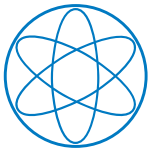
Physik-Department



Dissertation

Search for $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ at the Belle Experiment and Development of the Read-Out System for the Pixel Detector of the Belle II Experiment

Dmytro Levit

Technische Universität München

Technische Universität München

Physik-Department

Search for $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ at the Belle Experiment and Development of the Read-Out System for the Pixel Detector of the Belle II Experiment

Dmytro Levit

Vollständiger Abdruck der von der Fakultät für Physik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender: Prof. Dr. Andreas Weiler
Prüfer der Dissertation: 1. Prof. Dr. Stephan Paul
2. Prof. Dr. Christian Kiesling

Die Dissertation wurde am 12.12.2018 bei der Technischen Universität München eingereicht und durch die Fakultät für Physik am 23.01.2019 angenommen.

Abstract

The thesis describes development of a read-out system of the pixel detector at Belle II and search for a rare D^+ decay in full data sample of Belle experiment. Channel $D^+ \to K^- K_S^0 \pi^+ \pi^+ \pi^0$ is sensitive to new physics models that manifest themself in CP-violating isospin transition amplitudes $\Delta I = \frac{3}{2}$, which are prohibited in the standard model. This channel allows us to measure CP violation in the $\Delta I = \frac{3}{2}$ amplitude. I measured branching fraction with weak statistical significance 2 σ, and therefore report a 95 % upper exclusive limit on the branching fraction

$$BF(D^+ \to K^- K_S^0 \pi^+ \pi^+ \pi^0) < 7.23 \cdot 10^{-5}.$$

In the second part of the thesis, I describe an FPGA-based read-out system for the DEPFET pixel detector of the new Belle II experiment. The system controls the pixel detector and processes its data as a stand-alone data acquisition system for detector development and characterization and as a high-performance read-out system at Belle II.

Zusammenfassung

Die Dissertation befasst sich mit dem Aufbau des Datenerfassungssystems für den Pixeldetektor in Belle II und der Suche nach einem seltenen Zerfall des D^+ Mesons in dem gesamten Datensatz des Belle Experiments. Der Zerfall $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ könnte Physik jenseits des Standardmodels ausschließen. Durch diesen Zerfall könnten wir die, im Standardmodell verbotenen, CP-verletzenden $\Delta I = \frac{3}{2}$ Amplituden messen. Ich habe das Verzweigungsverhältnis mit schwacher statistischer Signifikanz $2\,\sigma$ gemessen und die 95 % obere Ausschlussgrenze für das Verzweigungsverhältnis

$$\mathrm{BF}(D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0) < 7.23 \cdot 10^{-5}$$

bestimmt. Im zweiten Teil meiner Arbeit beschreibe ich ein FPGA-basiertes Auslesesystem für den Belle II Pixeldetektor. Das System steuert den Detektor und verarbeitet seine Daten als Datenerfassungssystem für Detektorenentwicklung und -charakterisierung und als schnelles Auslesesystem in Belle II.

Contents

Chapter 1.

Introduction

The visible universe is made out of matter and not a matter-antimatter mixture. We know this fact from studying matter-antimatter composition of the cosmic rays. Also, we do not observe matter-antimatter annihilation signature in high-energy cosmic gamma rays.

There are two production mechanisms of the cosmic rays: the primary and the secondary production. The primary-produced rays originate in the stellar objects that radiate its constituents into the universe. For example, supernovae explosions or gas acceleration in the active galactic nuclei produce the rays of charged particles. These rays carry the information about the composition of the object.

If primary rays collide with cosmic background radiation in space they produce matter and antimatter rays. This type of rays is called the secondary rays. We can identify primary and secondary rays by their energy spectrum which differs for each type of the cosmic rays. Therefore, by measuring energy of the cosmic rays we can remove contribution of the secondary rays and calculate the matter-antimatter content of the universe [1].

The AMS-02 collaboration used this approach to measure the matter-antimatter composition of the visible universe. The experiment measured the amounts of matter and antimatter in the cosmic rays reaching Earth [2]. The measured fraction of the antimatter in the cosmic rays is consistent with secondary antimatter production in space from the collisions of the cosmic rays. This measurement proves that the visible universe consists only of matter.

This raises the question about the origin of matter-antimatter asymmetry in the universe. Scientists believe that most matter and all antimatter produced early in the universe annihilated with each other approximately 400000 years after the big bang. The photons from matter-antimatter annihilation form the cosmic background radiation. An excess of matter survived and formed our visible universe.

Because the universe is matter dominated, we can describe matter-antimatter asymmetry

$$A_{univ} = \frac{M - \overline{M}}{M + \overline{M}} \tag{1.1}$$

as a ratio of matter that is still around today $M - \overline{M}$ to the sum of matter and antimatter produced by the big bang $M + \overline{M}$.

Under assumption of baryon number conservation we know that all baryons, which survived annihilation in the early universe, have to be present today in form of baryonic matter. We estimate quantity of matter which survived annihilation from baryon density,

n_B, measured by AMS-02. We estimate the sum of matter and antimatter from the density of photons, n_γ, in the cosmic microwave background measured by the Planck collaboration [3]. With these quantities we estimate the matter-antimatter asymmetry in the early universe

$$A_{univ} = \frac{n_B}{n_\gamma} \approx 6 \times 10^{-10} \text{[1]}. \tag{1.2}$$

This asymmetry originates in the baryogenesis process that produced matter. In 1966 Sakharov proposed three conditions that are necessary during baryogenesis to produce the matter-dominated universe [4]. The Sakharov conditions are that baryogenesis processes:

1. proceed in thermal inequilibrium,
2. violate baryon number, and
3. violate discrete charge-parity (**CP**) symmetry.

The first condition is fulfilled because baryogenesis takes place in the expanding universe. Violation of baryon number has not yet been observed experimentally; but the standard model of particle physics allows for the baryon number violation at energies not currently reachable experimentally. Violation of the CP symmetry has been observed experimentally in the decays of kaons and beauty hadrons. The standard model incorporates CP violation through the structure of the weak interaction. However, the matter-antimatter asymmetry of the universe is 10 orders of magnitude larger than what the standard model could explain.

This discrepancy between observation and theory tells us that our understanding of the universe is not complete. So we search for new sources of CP violation beyond the standard model. A possible source of CP violation beyond the standard model may be found in the decays of the charm quark. Standard model prediction of the CP violation of the charm quark is two orders of magnitude below the current experimental sensitivity. This leaves room for discovering CP violation effects beyond the standard model predictions in decays of charmed hadrons.

This thesis starts with description of a search for non-standard-model sources of the CP violation in decay of D^+. I measured the branching fraction of the decay $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$using full data set of $1\,\text{ab}^{-1}$ recorded at the Belle experiment. This decay is sensitive to a subset of the CP-violating models beyond the standard model that have the weak $\Delta I = \frac{3}{2}$ structure.

While having the largest data sample, many measurements done at Belle still do not reach the desired sensitivity because the standard model predicts low rate for these processes. For example, the measurement of the CP violation in radiative D^0 decays to vector mesons at Belle is consistent with no CP violation [5]. But the sensitivity of the measurement does not reach standard model prediction because statistical uncertainty dominates the measurement's precision.

To improve sensitivity to effects of physics beyond the standard model, we designed an upgrade to the Belle experiment, the Belle II. The Belle II has the goal to increase data set

by factor 50 and improve sensitivity by factor 10. The goal will be achieved by increasing luminosity of the accelerator by factor 40 and improvements in the detectors.

The upgrade of the experiment adds a new pixel detector as the innermost subdetector. The detector will increase the impact parameter resolution of primary vertices. This is needed to compensate lower Lorentz boost of the collision products due to decreased beam energy asymmetry in the upgraded than in Belle. Another task of the detector is the reconstruction of low-momentum pion tracks. The low-momentum pions, which are commonly produced in the D^* decays, add an important discrimination parameter for selection of charmed meson.

As the innermost subdetector, the pixel detector is most sensitive for the beam background. High granularity of the detector combined with the long integration time results in high data rate of 20 GB/s. High data rate requirement reflects in the design of the data read-out system of the detector.

The second part of the thesis describes the development of the read-out system for the pixel detector of the Belle II experiment. The detector's design places non-trivial requirements for the read-out system. The system must

- synchronize and operate detector modules,
- sustain high data rates,
- process data in real time,
- configure and monitor front-end electronics.

All these requirements are addressed in the system design.

The thesis has following structure. The first part of the thesis describes the measurement of the branching fraction for the decay $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$. Next chapter describes the source of CP violation in the standard model. Chapter three describes a way for measuring a non-standard model contributions to CP violation in charm decays. Chapter four introduces the Belle experiment where the measurement is done. Chapter five describes the measurement performed using the Belle data sample.

The second part of the thesis describes the development of the read-out system for the pixel detector. Chapter six describes the design of the Belle II experiment. Chapter seven describes the vertex detector which consists of the silicon strip and pixel detector. Chapter eight describes the design of the read-out system of the pixel detector. Chapter nine gives the examples of the system use in different setups.

Chapter 2.

The Standard Model and the CP Violation

2.1. Discrete Space-Time Symmetries in the Standard Model

Symmetries describe how interactions behave under transformations of system's parameters. Discrete symmetries are a subset of symmetries that mirrors parameters of the system. Two discrete symmetries, the charge conjugation symmetry **C** and the parity symmetry **P**, are important for studying standard model predictions for behavior of particles and antiparticles.

Charge conjugation symmetry transforms particle to its antiparticle. Parity symmetry transforms a particle's handedness: a left-handed particle becomes a right-handed particle and vice-versa. A combination of charge conjugation and parity symmetries, the **CP**, transforms a left-handed particle to a right-handed antiparticle and vice versa.

Three interactions in the standard model behave differently under these transformations. The strong and electromagnetic interactions are invariant under **C**, **P**, and **CP** symmetries. We can express this through the decay rates $\Gamma_{L,R}$ for particles and $\overline{\Gamma}_{L,R}$ for antiparticles with the given handedness:

$$\Gamma_L = \Gamma_R = \overline{\Gamma}_L = \overline{\Gamma}_R \tag{2.1}$$

The invariance of the interaction is measured as asymmetries between two states:

$$A = \frac{\Gamma_1 - \Gamma_2}{\Gamma_1 + \Gamma_2} \tag{2.2}$$

For strong and electromagnetic interactions these asymmetries wanish

$$\underbrace{\frac{\Gamma_L - \Gamma_R}{\Gamma_L + \Gamma_R}}_{A_P} = \underbrace{\frac{\Gamma_L - \overline{\Gamma}_L}{\Gamma_L + \overline{\Gamma}_L}}_{A_C} = \underbrace{\frac{\Gamma_L - \overline{\Gamma}_R}{\Gamma_L + \overline{\Gamma}_R}}_{A_{CP}} = 0. \tag{2.3}$$

Bosons of the weak force interact only with left-handed particles and right-handed antiparticles. This maximally violates charge conjugation and parity symmetries

$$\underbrace{\frac{\Gamma_L - \cancel{\Gamma_R}}{\Gamma_L + \cancel{\Gamma_R}}}_{A_P} = \underbrace{\frac{\Gamma_L - \cancel{\overline{\Gamma}_L}}{\Gamma_L + \cancel{\overline{\Gamma}_L}}}_{A_C} = 1, \tag{2.4}$$

but nearly conserves CP symmetry

$$A_{CP} = \left| \frac{\Gamma_L - \overline{\Gamma}_R}{\Gamma_L + \overline{\Gamma}_R} \right| = 10^{-3} \text{ to } 10^{-1}. \tag{2.5}$$

Study of CP asymmetry lead to the discovery of the CKM matrix which transforms CP eigenstates into the mass eigenstates. This chapter reviews measurements that lead to discovery of CP violation and introduce CP violation in the framework of the CKM theory.

2.1.1. Parity Symmetry Violation

The parity operator mirrors spacial coordinates of the system. In 1956, Wu, using a method proposed by Lee and Yang [6], showed that weak force violates parity [7]. Wu and her colleagues studied β-decay of polarized Co^{60} nuclei:

$$Co^{60} \rightarrow Ni^{60} + e^- + \overline{\nu}_e + 2\gamma \tag{2.6}$$

Since electron has positive parity, a reflection of the system should not change its angular distribution. The spin of the Co nucleus, which was aligned to an external magnetic field, provided a reference frame for the measurement. The experiment monitored the rate of the electrons parallel and antiparallel to the spin of the Co nucleus decay and showed that mirroring the reference frame changes the decay rate. This proved that the weak force violates parity symmetry.

2.1.2. Charge Conjugation Symmetry Violation

The charge conjugation operator mirrors all charges of a particle, turning it into its antiparticle. In 1957, Garwin, Lederman, and Weinrich showed that the weak force violates **C** symmetry [8]. They measured the angular distributions of electrons and positrons produced in muon and antimuon decays and calculated the helicity for electrons and positrons. If **C** were symmetry of the weak interaction, electrons and positrons would have the same helicity. The experiment showed that electrons and positrons prefer opposite helicities. This proved that the weak interaction violates **C** symmetry.

2.2. CP Violation

Discovery that the weak interaction violates **C** and **P** maximally lead to the development of the V-A theory of the weak interaction [9, 10]. The V-A current of the weak interaction theory

$$V - A \equiv \underbrace{\overline{\Psi}\gamma^\mu\Psi}_{V} - \underbrace{\overline{\Psi}\gamma^\mu\gamma^5\Psi}_{A} = \overline{\Psi}\gamma^\mu(1-\gamma^5)\Psi, \tag{2.7}$$

with $\overline{\Psi}$ is an antifermion, Ψ is a fermion, has a left-handed projection operator

$$P_L = (1-\gamma^5) \tag{2.8}$$

in its structure. This operator projects a fermion field Ψ to a left-handed fermion field Ψ_L, and an antifermion field $\overline{\Psi}$ to a right-handed antifermion field $\overline{\Psi}_R$. The structure of the theory predicts that the weak force interacts only with left-handed fermions and right-handed antifermions

$$V - A = \overline{\Psi}_R \gamma^\mu \Psi_L. \tag{2.9}$$

Feynman and Gell-Mann also predicted that a combination of parity transformation and charge conjugation, **CP**, is a conserved symmetry of the weak interaction [10].

Tests of CP symmetry require measurement of the differences in the behavior of matter and antimatter. In 1964, Cronin and Fitch discovered violation of the CP symmetry in decays of neutral kaons. They measured the decay of the pure CP eigenstate into a different CP eigenstate. The effect is caused by tiny difference in the masses of the neutral mesons. In 1972, Kobayashi and Maskawa extended the flavour-mixing matrix discovered by Cabibbo with a third generation of quarks to explain the measured CP violation in the framework of the quantum field theory. This theory is known as the CKM theory[1]. The first quark of the third generation, the beauty quark, was discovered in 1975.

The CKM theory predicts another type of CP violation, direct CP violation, which shows up as an asymmetry in the rates of a decay mode and its CP-conjugate. In 1999, the KTeV experiment at Fermilab and the NA48 experiment at CERN discovered direct CP violation in the decays of neutral kaons — in the same neutral kaon decays in which Cronin and Fitch discovered indirect CP violation. In 2001, the Belle and BaBar experiments confirmed the CKM predictions for direct CP violation in neutral B meson decays.

2.3. CP Violating Amplitudes

We can describe CP violation in the amplitude formalism. The amplitude of a decay can be written as

$$A(X \to f) = |A| e^{i \arg A}. \tag{2.10}$$

The amplitude consists of the magnitude |A| and the phase arg(A). The amplitude for the corresponding CP-conjugated decay can be written as

$$\overline{A}(\overline{X} \to \overline{f}) = \left|\overline{A}\right| e^{i \arg \overline{A}}. \tag{2.11}$$

Because their magnitudes must be equal

$$|A| = |\overline{A}|, \tag{2.12}$$

we have freedom to redefine arg(A) as two phases: a CP-even phase δ that is not affected under CP transformation and a CP-odd phase ϕ that changes its sign under CP transformation

$$\arg A = \delta + \phi, \tag{2.13}$$

$$\arg \overline{A} = \delta - \phi. \tag{2.14}$$

[1] This theory is named after its creators: Cabibbo, Kobayashi, and Maskawa.

Both amplitudes have then the form

$$A(X \to f) = |A| e^{i(\delta+\phi)} \tag{2.15}$$

$$\overline{A}(\overline{X} \to \overline{f}) = |A| e^{i(\delta-\phi)}. \tag{2.16}$$

If there are several amplitudes which contribute to the same decay, then it is possible to observe the interference of the amplitudes experimentally through the decay rates

$$\Gamma(X \to f) = |A|^2 = |A_1 + A_2|^2 \tag{2.17}$$

$$= |A_1|^2 + |A_2|^2 + |A_1 A_2| \left(e^{i((\phi_1+\delta_1)-(\phi_2+\delta_2))} + e^{i(-(\phi_1+\delta_1)+(\phi_2+\delta_2))} \right) \tag{2.18}$$

$$\Gamma(\overline{X} \to \overline{f}) = \left|\overline{A}\right|^2 = \left|\overline{A}_1 + \overline{A}_2\right|^2 \tag{2.19}$$

$$= |A_1|^2 + |A_2|^2 + |A_1 A_2| \left(e^{i((-\phi_1+\delta_1)-(-\phi_2+\delta_2))} + e^{i(-(-\phi_1+\delta_1)+(-\phi_2+\delta_2))} \right). \tag{2.20}$$

We can define phase differences as $\Delta\phi = \phi_1 - \phi_2$, $\Delta\delta = \delta_1 - \delta_2$, and magnitude ratio $R = \frac{|A_2|}{|A_1|}$. Then, we can construct an observable for the strength of the CP violation in a decay channel which can be measured experimentally:

$$A_{CP} = \frac{\Gamma(X \to f) - \Gamma(\overline{X} \to \overline{f})}{\Gamma(X \to f) + \Gamma(\overline{X} \to \overline{f})} = \frac{-2R sin\Delta\phi sin\Delta\delta}{1 + R^2 + 2R cos\Delta\phi cos\Delta\delta}. \tag{2.21}$$

The relation shows that CP is violated if and only if

- more than one amplitude contributes to the decay, R > 0,
- the CP-even phase difference $\Delta\delta$ is not zero,
- the CP-odd phase difference $\Delta\phi$ is not zero.

CP violation requires CP-odd phase which comes from complex couplings which are complex-conjugated by CP conjugation

$$x = |x| e^{i\phi_x} \xrightarrow{CP} |x| e^{-i\phi_x} = \bar{x}. \tag{2.22}$$

2.4. Cabibbo-Kobayashi-Maskawa Matrix

Complex couplings arise as effective couplings between up- and down-type fields in the standard model Lagrangian. We show this by describing the field transformation into the eigenstates of the diagonalized mass matrix in the mass term of the Lagrangian and then transforming the weak term of the Lagrangian to use the eigenstates of the matrix.

In general, the standard model defines two 3×3 mass matrices for up- and down-type fields, M_u and M_d, that Yukawa coupling requires for the Higgs mechanism:

$$\mathscr{L}_m = \overline{U} M_u U H + \overline{D} M_d D H. \tag{2.23}$$

We can diagonalize the mass matrices with the change-of-basis matrices V_u and V_d

$$M'_u = V_u^\dagger M_u V_u = diag(m_u, m_c, m_t) \tag{2.24}$$

$$M'_d = V_d^\dagger M_d V_d = diag(m_d, m_s, m_b). \tag{2.25}$$

We then rewrite the mass term as coupling of the eigenstates of the diagonalized matrices

$$\mathscr{L}_m = \overline{U} M_u U H + \overline{D} M_d D H = \underbrace{\overline{U} V_u}_{\overline{U}'} M'_u \underbrace{V_u^\dagger U}_{U'} H + \underbrace{\overline{D} V_d}_{\overline{D}'} M'_d \underbrace{V_d^\dagger D}_{D'} H = \overline{U}' M'_u U' H + \overline{D}' M'_d D' H. \tag{2.26}$$

Now, we transform the weak term of the Lagrangian

$$\mathscr{L}_W \sim \overline{D} \gamma^\mu (1 - \gamma_5) U \mathrm{W}^-{}_\mu \tag{2.27}$$

into eigenstates of the diagonalized mass matrix $\overline{D}'$ and U'

$$\mathscr{L}_W \sim \underbrace{\overline{D} V_d}_{\overline{D}'} V_d^\dagger \gamma^\mu (1 - \gamma_5) V_u \underbrace{V_u^\dagger U}_{U'} W_\mu = \overline{D}' \gamma^\mu (1 - \gamma_5) V_d^\dagger V_u U' W_\mu. \tag{2.28}$$

The matrix $V_d^\dagger V_u$ is the effective weak coupling of the up- and down-type fields.

In 1963, Nicola Cabibbo used similar approach to describe how to preserve the universality of the coupling in the V–A theory of weak interactions [11]. The mass eigenstates of the d and s quarks[2] are rotated into weak eigenstates d' and s':

$$\begin{pmatrix} d' \\ s' \end{pmatrix} = \begin{pmatrix} cos\Theta & sin\Theta \\ -sin\Theta & cos\Theta \end{pmatrix} \begin{pmatrix} d \\ s \end{pmatrix}. \tag{2.29}$$

The unitary 2×2 rotation matrix is called the Cabibbo matrix; and Θ, the Cabibbo angle. Though the Cabibbo matrix described the weak decays known at that time, it could not describe the CP violation observed by Cronin and Fitch. This is the consequence of Cabibbo matrix not containing complex amplitudes that are needed for CP violation.

In 1972, Kobayashi and Maskawa studied different structures of the weak interaction [12]. They showed that a complex n×n matrix has n^2 real parameters. Out of n^2, (2n - 1) parameters can be absorbed as unphysical quark field phases, which leaves $(n - 1)^2$ free parameters. By using Cabibbo's approach, we can redefine $\frac{1}{2}n(n-1)$ as mixing angles. Remaining $\frac{1}{2}(n-2)(n-1)$ parameters are complex phases.

The Cabibbo's 2×2 rotation matrix has one free rotation angle. Rotation by the Cabbibo angle absorbs complex phase of the fermion fields. This means that weak interaction described by a 2×2 Cabibbo matrix conserves CP symmetry. Last statement contradicts experimental evidences.

A 3×3 unitary matrix has 4 free parameters of which 3 are independent rotation angles. The remaining free parameter is the complex phase that cannot be rotated away. Presence of a free phase violates CP.

[2] Only three quarks – u, d, and s – were known at the time.

The 3×3 unitary matrix

$$\begin{pmatrix} \mathrm{u}' \\ \mathrm{c}' \\ \mathrm{t}' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} \mathrm{d}' \\ \mathrm{s}' \\ \mathrm{b}' \end{pmatrix} \tag{2.30}$$

described in equation 2.28 is the effective weak coupling matrix $V_d^\dagger V_u$. This matrix is called the CKM matrix. The rank of the matrix necessitates existence of the third generation of quarks. In 1977, Lederman et al. discovered the first member of the third generation, the beauty quark, in proton-nucleus collisions [13].

CKM matrix is currently the only observed source of CP violation in standard model. We can calculate size of CP violation that arises from effective weak coupling in the CKM theory and compare it with CP asymmetry of the universe.

In 1985, Jarlskog showed that CP violation in the CKM theory comes from a mismatch between up- and down-type matrix [14] with CP asymmetry quantified as

$$A_{univ}^{CP} \approx 2\, m_{\mathrm{t}}^4\, m_{\mathrm{b}}^4\, m_{\mathrm{c}}^2\, m_{\mathrm{s}}^2\, \frac{|J|}{\Lambda^{12}} \tag{2.31}$$

where J is the Jarlskog invariant, Λ is an energy scale of the process. Jarlskog determinant is a convention-independent measure of CP violation in standard model

$$J = Im(V_{ij} V_{kl} V_{il}^* V_{kj}^*) = (3.172^{+0.094}_{-0.098}) \times 10^{-5}\,[15]. \tag{2.32}$$

where V_{ab} are CKM matrix elements. Normalized to the electro-weak scale (100 GeV), the standard model contribution to CP asymmetry of the universe is

$$A_{univ,CKM}^{CP} \approx 10^{-20}\,[16]. \tag{2.33}$$

Compared to measured CP asymmetry of the universe, given in equation 1.2, the standard model contribution is 10 orders of magnitude below the CP violation which is needed to satisfy Sakharov conditions for the baryogenesys. This comparison shows that we need additional sources of CP violation of the non-CKM structure to explain matter-dominated universe.

Chapter 3.

CP Violation in Charm Sector as a Probe for the New Physics

In the previous chapter we estimated the baryon asymmetry using parameters from the standard model. We learned these parameters from measurements of CP violation in weak decays. But CP violation in the standard model is not enough to explain matter-antimatter asymmetry from baryogenesys.

To find missing CP violation, we look for CP violation in decays, where the standard model predicts none. For example, we look in decays of charm quark. It is challenging to estimate absolute magnitude CP violation in charm. Charm mass is in the region, where no perturbative QCD method applies. This makes it not possible to reliably calculate expected CP violation. But we can use other standard model predictions that can be identified by either presence or absence of CP violation. One of these predictions, which I describe in this chapter, is the isospin structure of weak decay. Also, up to now we observed CP violation only in decays of down-type quarks s and b contained in K and B mesons. It would be interesting to measure CP violation in decays of charm quark, because charm quark is the only up-type quark that can have CP violation in hadron decays.

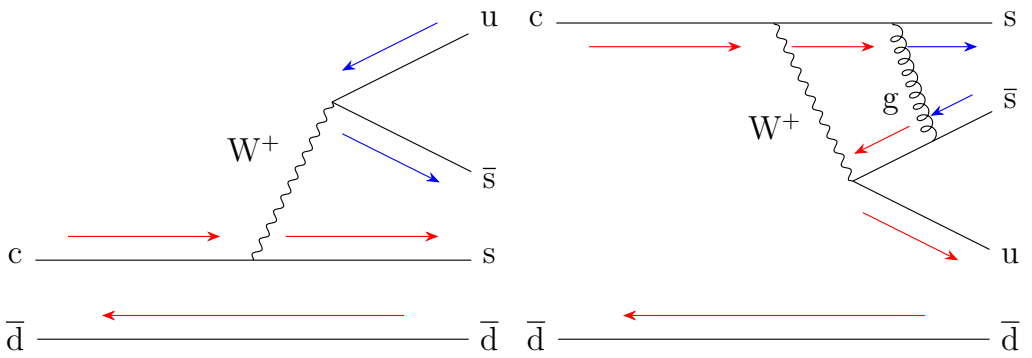
3.1. Measurement of CP Violation in Charm

The direct CP violation in the charm sector is expressed by the difference of the CP asymmetries in D mesons decaying to a two-kaon and a two-pion final states

$$\Delta A_{CP} = A_{CP}(\mathrm{D} \to \mathrm{KK}) - A_{CP}(\mathrm{D} \to \pi\pi) \tag{3.1}$$

The combined fit of the ΔA_{CP} parameter to the latest measurements of the CP violation in the charm decays yields no CP violation at 9.3 % confidence level [17]. The CP violation from the tree and penguin amplitudes in the singly-Cabibbo suppressed decays is, however, suppressed in the standard model at the level 10^{-3} [18]. Therefore, the current measurement precision allows for an additional non-standard model source of the CP violation.

Grossman et al. [19] propose a test for the subset of the new physics models, which realize the CP-violating $\Delta I = \frac{3}{2}$ decay amplitudes in the singly-Cabibbo suppressed D decays. For example, the model by Hochberg and Nil predicts a new scalar coupling which



(a) Color-allowed tree-level operator

(b) Color-suppressed tree-level operator

Figure 3.1.: Representative Feynman diagrams for tree-level operators. Arrows show color flow in the diagram

has the $\Delta I = \frac{3}{2}$ structure [20]. In this analysis we search for the new physics contributions to the CP violation in $\Delta I = \frac{3}{2}$ decay amplitude in the singly-Cabibbo suppressed decays

$$D^+ \to K^* \overline{K}^* \pi. \tag{3.2}$$

This section shows why the $\Delta I = \frac{3}{2}$ decay amplitude can only be realized by the physics outside of the standard model based on reference [19].

I will focus on the $\Delta C = 1$ decays in this thesis. These are decays of charm meson in the initial state into a flavourless final state. The effective Hamiltonian of the decay is

$$H_{\text{eff}} = \frac{G_F}{\sqrt{2}} \left(\sum_{p=d,s} V_{cp}^* V_{up} (C_1 Q_1^p + C_2 Q_2^p) - V_{cb}^* V_{ub} \sum_{i=3}^{6} C_i Q_i + C_{8g} Q_{8g} \right) + h.c. \tag{3.3}$$

where G_F is the Fermi constant, C_i are Wilson coefficients and Q_i are effective field operators [21].

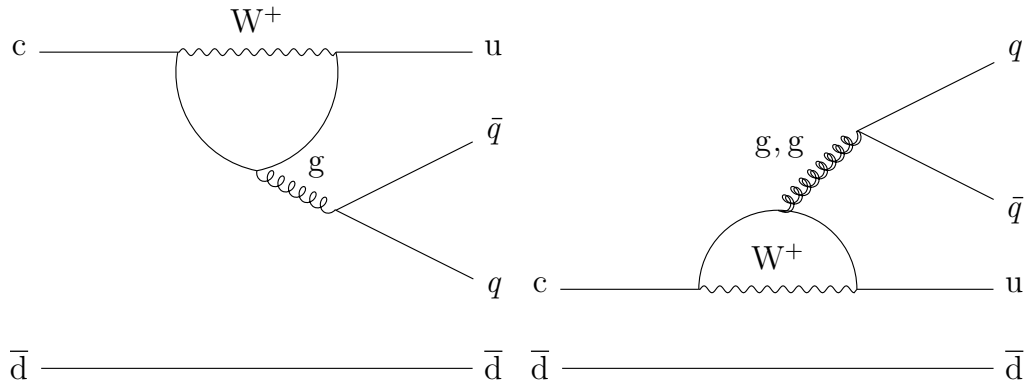
3.1.1. Tree-Level Operators

There are two types of tree-level operators in the Hamiltonian: a color-allowed operators Q_1, and a color-suppressed operators Q_2

$$Q_1^p = (\overline{p}_\alpha c_\alpha)_{V-A} (\overline{u}_\beta p_\beta)_{V-A}, \tag{3.4}$$

$$Q_2^p = (\overline{p}_\alpha c_\beta)_{V-A} (\overline{u}_\beta p_\alpha)_{V-A} \tag{3.5}$$

where α, β is summation over quark colors and p = d,s. The operators describe decay of charm to down-type quarks d and s. Figure 3.1 shows representative Feynman diagrams for a color-allowed and a color-suppressed tree operators. Diagrams show that different colors of $\bar{s}u$ need a gluon that causes α_s suppression in the leading order for the



(a) Color-allowed QCD penguin operator (b) Color-suppressed QCD penguin operator

Figure 3.2.: Representative Feynman diagrams for QCD penguin operators

color-suppressed operator, while the colors of $\bar{\mathrm{s}}\mathrm{u}$ are free for the color-allowed operator. Therefore, the color-allowed operator is enhanced by number of colors.

We can calculate the isospin structure of the operator using isospin decomposition. In this parametrization (u, d), ($\overline{\mathrm{d}}$, -$\bar{\mathrm{u}}$), (D^+, D^0), (K^{*+}, K^{*0}), ($\overline{\mathrm{K}}^{*0}$, K^{*-}) form isospin doublets, (c) forms an isospin singlet, and (π^+, π^0, π^-) form an isospin triplet. Operators have isospin structure

$$\sum_{p=d,s}\sum_{i=1,2} Q_i^p : \left|\tfrac{1}{2},-\tfrac{1}{2}\right\rangle_{\mathrm{d}} \otimes |0\rangle_{\mathrm{c}} \otimes \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle_{\mathrm{u}} \otimes \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle_{\bar{\mathrm{d}}} + |0\rangle_{\bar{\mathrm{s}}} \otimes |0\rangle_{\mathrm{c}} \otimes \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle_{\mathrm{u}} \otimes |0\rangle_{\mathrm{s}} = \tag{3.6}$$

$$\left(1+\sqrt{\tfrac{2}{3}}\right)\left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle + \sqrt{\tfrac{1}{3}}\left|\tfrac{3}{2},\tfrac{1}{2}\right\rangle, \tag{3.7}$$

where ket notation $|I,I_3\rangle$ contains total isospin I and third component of the isospin I_3. The decomposition shows that both tree-level operators have isospin structure $\Delta I = \frac{1}{2}$ and $\Delta I = \frac{3}{2}$.

3.1.2. Penguin Operators

The standard model penguin operators are

$$Q_{3,5} = (\bar{\mathrm{u}}\mathrm{c}) \sum_{q=\mathrm{u,d,s}} (\bar{q}q)_{\mathrm{V\mp A}}, \tag{3.8}$$

$$Q_{4,6} = (\bar{\mathrm{u}}_\alpha \mathrm{c}_\beta) \sum_{q=\mathrm{u,d,s}} (\bar{q}_\beta q_\alpha)_{\mathrm{V\mp A}}, \tag{3.9}$$

$$Q_{8g} = -\frac{g_s}{8\pi^2} m_c \bar{\mathrm{u}} \sigma_{\mu\nu}(1+\gamma_5) G^{\mu\nu} \mathrm{c}. \tag{3.10}$$

The operators $Q_{3,5}$ are the color-enhanced QCD penguin operators. The operators $Q_{4,6}$ are the color-suppressed QCD penguin operators. Figure 3.1 shows representative Feynman diagrams for these operators. Color-suppressed operator produce $q\bar{q}$ in a color-singlet

state, therefore, additional gluons are needed to couple to $q\bar{q}$. This suppresses the color-suppressed relative to the color-enhanced operator.

Penguin operators have isospin structure

$$Q_{3-6} : \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle_{\mathrm{u}} \otimes |0\rangle_{\mathrm{c}} \otimes \left(- \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle_{\mathrm{u}} \otimes \left|\tfrac{1}{2},-\tfrac{1}{2}\right\rangle_{\bar{\mathrm{u}}} + \left|\tfrac{1}{2},-\tfrac{1}{2}\right\rangle_{\mathrm{d}} \otimes \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle_{\bar{\mathrm{d}}} \right) = \sqrt{2} \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle \tag{3.11}$$

that only contains isospin transitions $\Delta I = \frac{1}{2}$. The operator Q_{8g} is the magnetic penguin operator. It has isospin structure

$$Q_{8g} : \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle_{\mathrm{u}} \otimes |0\rangle_{\mathrm{c}} = \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle \tag{3.12}$$

that also only contains isospin transitions $\Delta I = \frac{1}{2}$.

The fact that all standard model penguin operators have isospin structure $\Delta I = \frac{1}{2}$ implies that only tree-level amplitudes have $\Delta I = \frac{1}{2}$ transitions in charm decays.

3.1.3. CP Violation in $\Delta I = \frac{3}{2}$ Amplitudes

Grossman et al. used isospin decomposition to construct a pure $\Delta I = \frac{3}{2}$ amplitude from the amplitudes in a singly-Cabibbo suppressed D^+ decay [19]. We begin with calculating isospin structure of the final states. The decomposition of the final states as eigenstates of isospin is

$$\left|\mathrm{K}^{*+}\overline{\mathrm{K}}^{*0}\pi^0\right\rangle = \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle \otimes \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle \otimes |1,0\rangle \tag{3.13}$$

$$= \sqrt{\tfrac{1}{2}}\,|2,1;1\rangle + \sqrt{\tfrac{1}{2}}\,|1,1;1\rangle \tag{3.14}$$

$$\left|\mathrm{K}^{*+}\mathrm{K}^{*-}\pi^+\right\rangle = \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle \otimes \left|\tfrac{1}{2},-\tfrac{1}{2}\right\rangle \otimes |1,1\rangle \tag{3.15}$$

$$= \tfrac{1}{2}\,|2,1;1\rangle - \tfrac{1}{2}\,|1,1;1\rangle + \sqrt{\tfrac{1}{2}}\,|1,1;0\rangle \tag{3.16}$$

$$\left|\mathrm{K}^{*0}\overline{\mathrm{K}}^{*0}\pi^+\right\rangle = \left|\tfrac{1}{2},-\tfrac{1}{2}\right\rangle \otimes \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle \otimes |1,1\rangle \tag{3.17}$$

$$= \tfrac{1}{2}\,|2,1;1\rangle - \tfrac{1}{2}\,|1,1;1\rangle - \sqrt{\tfrac{1}{2}}\,|1,1;0\rangle\,, \tag{3.18}$$

$$\tag{3.19}$$

where ket notation $\left|I,I_3;I_{\mathrm{K}^*\overline{\mathrm{K}}^*}\right\rangle$ contains total isospin I, third component of the isospin I_3, and total isospin of the $\mathrm{K}^*\ \overline{\mathrm{K}}^*$ pair $I_{\mathrm{K}^*\overline{\mathrm{K}}^*}$.

Then, we calculate isospin projections of the effective operators $O_{\frac{1}{2}}$ and $O_{\frac{3}{2}}$ for isospin transitions $\Delta I = \frac{1}{2}$ and $\Delta I = \frac{3}{2}$ in D^+ decay:

$$O_{\frac{1}{2}} \left|\mathrm{D}^+\right\rangle = \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle \otimes \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle = |1,1\rangle\,, \tag{3.20}$$

$$O_{\frac{3}{2}} \left|\mathrm{D}^+\right\rangle = \left|\tfrac{3}{2},\tfrac{1}{2}\right\rangle \otimes \left|\tfrac{1}{2},\tfrac{1}{2}\right\rangle = -\tfrac{1}{2}\,|1,1\rangle + \tfrac{\sqrt{3}}{2}\,|2,1\rangle\,. \tag{3.21}$$

Next, we calculate matrix elements for the operators in a basis of total isospin eigenstates separated by total isospin of the $K^*\overline{K}^*$ pair:

$$\left\langle 1,1;0\middle|O_{\frac{1}{2}}\middle|D^+\right\rangle \equiv B_1', \tag{3.22}$$

$$\left\langle 1,1;0\middle|O_{\frac{3}{2}}\middle|D^+\right\rangle \equiv -\tfrac{1}{2}B_3', \tag{3.23}$$

$$\left\langle 1,1;1\middle|O_{\frac{1}{2}}\middle|D^+\right\rangle \equiv B_1, \tag{3.24}$$

$$\left\langle 1,1;1\middle|O_{\frac{3}{2}}\middle|D^+\right\rangle \equiv -\tfrac{1}{2}B_3, \tag{3.25}$$

$$\left\langle 2,1;1\middle|O_{\frac{1}{2}}\middle|D^+\right\rangle \equiv 0, \tag{3.26}$$

$$\left\langle 2,1;1\middle|O_{\frac{3}{2}}\middle|D^+\right\rangle \equiv \tfrac{\sqrt{3}}{2}C_3. \tag{3.27}$$

We combine isospin decomposition of the final states with matrix elements for operators $O_{\frac{1}{2}}$ and $O_{\frac{3}{2}}$ to calculate amplitudes for D^+ decay to each final state:

$$A^{+00} \equiv \left\langle K^{*+}\overline{K}^{*0}\pi^0\middle|O_{\frac{1}{2}}+O_{\frac{3}{2}}\middle|D^+\right\rangle = \tfrac{1}{2}\left(\sqrt{2}B_1+\tfrac{\sqrt{3}}{\sqrt{2}}C_3-\tfrac{1}{\sqrt{2}}B_3\right), \tag{3.28}$$

$$A^{+-+} \equiv \left\langle K^{*+}K^{*-}\pi^+\middle|O_{\frac{1}{2}}+O_{\frac{3}{2}}\middle|D^+\right\rangle = \tfrac{1}{2}\left(-B_1+\sqrt{2}B_1'+\tfrac{\sqrt{3}}{2}C_3+\tfrac{1}{2}B_3-\tfrac{1}{\sqrt{2}}B_3'\right), \tag{3.29}$$

$$A^{00+} \equiv \left\langle K^{*0}\overline{K}^{*0}\pi^+\middle|O_{\frac{1}{2}}+O_{\frac{3}{2}}\middle|D^+\right\rangle = \tfrac{1}{2}\left(-B_1-\sqrt{2}B_1'+\tfrac{\sqrt{3}}{2}C_3+\tfrac{1}{2}B_3+\tfrac{1}{\sqrt{2}}B_3'\right). \tag{3.30}$$

By adding the amplitudes we construct a pure $\Delta I = \frac{3}{2}$ amplitude

$$\sqrt{2}A^{+00}+A^{+-+}+A^{00+} = \sqrt{3}C_3. \tag{3.31}$$

To test standard model, we can measure CP asymmetry in C_3

$$A^{CP}(C_3) = \frac{|C_3|^2-|\overline{C_3}|^2}{|C_3|^2+|\overline{C_3}|^2} = \frac{\left|\sqrt{2}A^{+00}+A^{+-+}+A^{00+}\right|^2-\left|\sqrt{2}\overline{A}^{-00}+\overline{A}^{-+-}+\overline{A}^{00-}\right|^2}{\left|\sqrt{2}A^{+00}+A^{+-+}+A^{00+}\right|^2+\left|\sqrt{2}\overline{A}^{-00}+\overline{A}^{-+-}+\overline{A}^{00-}\right|^2}. \tag{3.32}$$

Because standard model only allows $\Delta I = \frac{3}{2}$ transitions in tree amplitudes, CP asymmetry will always cancel for this eignestate in standard model. Any non-zero CP violation in this eigenstate is a clear sign for physics beyond the standard model.

To have non-zero CP violation, the interference terms in

$$\left|\sqrt{2}A^{+00}+A^{+-+}+A^{00+}\right|^2 = 2\left|A^{+00}\right|^2+\left|A^{+-+}\right|+\left|A^{00+}\right|^2+2\left|A^{+00}\right|\left|A^{+-+}\right|cos\Delta\delta_{+-+}^{+00} \tag{3.33}$$

$$+2\left|A^{+00}\right|\left|A^{00+}\right|cos\Delta\delta_{00+}^{+00}+2\left|A^{+-+}\right|\left|A^{00+}\right|cos\Delta\delta_{00+}^{+-+}, \tag{3.34}$$

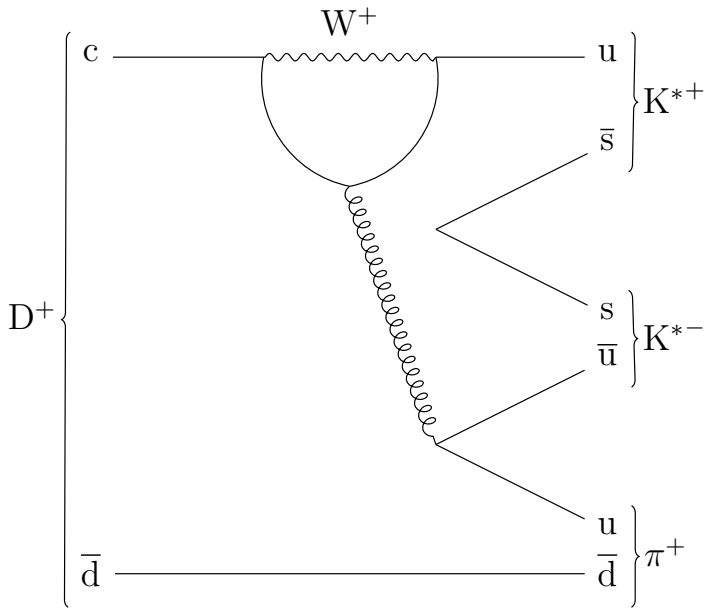


Figure 3.3.: One of the possible D^+ decay amplitude in the channel of interest

where $\Delta\delta^{+00}_{+-+}$, $\Delta\delta^{+00}_{00+}$, $\Delta\delta^{+-+}_{00+}$ are CP-even phase differences between amplitudes, must not vanish.

3.2. Decay $D^+ \to K^*K^*\pi$

One of the decay channels proposed by Grossman et al. in [19] for the search of the $\Delta I = \frac{3}{2}$ decay amplitude is the channel

$$D^+ \to K^*K^*\pi.$$

This channel has a common final state for the decay modes $K^{*+}\overline{K}^{*0}\pi^0$, $K^{*+}K^{*-}\pi^+$, and $K^{*0}\overline{K}^{*0}\pi^+$:

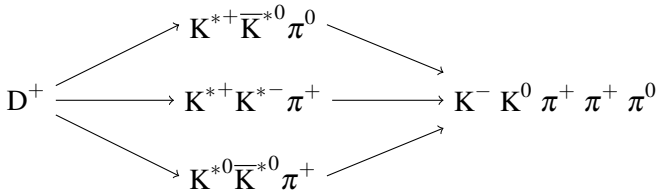


Figure 3.3 shows one of the possible diagram. This channel has the advantage that the common final state allows us to reconstruct relative phases of the decay amplitudes by analysing their overlap in the phase space of the five-body final state.

The decay $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ offers a practical way of measuring new physics contributions on the theory side. However, this decay has never been observed before. Therefore, in this work I will focus on measuring the branching fraction of this decay at Belle experiment.

To estimate expected branching fraction we consider decay mode $D^+ \to K^- K^0_S \pi^+ \pi^+$ with branching fraction $(0.234 \pm 0.018)\,\%$. We expect that additional π^0 would only affect the available phase space of the decay. We estimate that branching fractions of the decay would have similar ratio as the ratio of their phase space volume V per particle

$$\frac{BF(D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0)}{BF(D^+ \to K^- K^0_S \pi^+ \pi^+)} \sim \frac{\sqrt[5]{V_{D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0}}}{\sqrt[4]{V_{D^+ \to K^- K^0_S \pi^+ \pi^+}}} \approx 0.55. \tag{3.35}$$

Alternatively, we compare branching fractions for two D^0 decays that have similar kinematics

$$\frac{BF(D^0 \to K^- K^+ \pi^+ \pi^- \pi^0)}{BF(D^0 \to K^- K^+ \pi^+ \pi^-)} = \frac{(3.2 \pm 2.0)\,\%}{(2.4 \pm 0.1)\,\%} = 1.3 \pm 0.8. \tag{3.36}$$

Precision of the branching fraction measurement for decay $D^0 \to K^- K^+ \pi^+ \pi^- \pi^0$ leaves room for suppression by one order of magnitude. Therefore, we expect branching fraction of the decay $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ to be in the order of 10^{-4}.

Chapter 4.

Belle Experiment

The Belle experiment was a solid angle spectrometer at the KEKB B factory at the High Energy Accelerator Research Organization KEK [1], in Tsukuba, Japan. KEKB is an electron-positron collider that operates at and near the $\Upsilon(4S)$ resonance. The $\Upsilon(4S)$ resonance is the first $b\bar{b}$ resonance above the threshold to decay to a pair of B–$\overline{B}$ mesons, to which it decays 96 % of the time. A main goal of the Belle experiment was to study the properties of the b quark.

Figure 4.1 shows the cross section for inclusive hadron production in e^-e^+ collisions. There are resonant bottomonium states Υ, where $\Upsilon(4S)$ is the first resonance that can produce B–$\overline{B}$ mesons. The $\Upsilon(4S)$ resonance is above a flatter source of nonresonant cross-section that is the source for $q\bar{q}$ production. The cross section for charm production is

$$\sigma(e^-e^+ \to c\bar{c})_{\Upsilon(4S)} = 1.3\,nb$$

where a D meson is produced 90 % of the time. This is comparable with beauty production at the $\Upsilon(4S)$ resonance

$$\sigma(e^-e^+ \to b\bar{b})_{\Upsilon(4S)} = 1.1\,nb.$$

Therefore, the KEKB produces comparable number of the D mesons, which we use to study CP violation in the D^+ meson decays.

4.1. KEKB B Factory

The KEKB B factory is asymmetric in energy. It consists of a linear accelerator and two accelerator rings. Electrons and positrons are produced in the linear accelerator and accelerated to 3 GeV. Then, they are injected into the ring accelerators. The rings have the circumference 3016 m and share the same tunnel. One ring accelerates electrons to 8 GeV. The other ring accelerates positrons to 3.5 GeV.

The rings intersect at an interaction point where the beams collide. The asymmetric beam energies give the collision products a Lorentz boost in the direction of the electrons. The Lorentz boost is important for precise measurement of decay times of the collision products.

During 10 years of operation, KEKB delivered 1 ab^{-1} integrated luminosity. This corresponds to 7.7×10^8 $B\overline{B}$ pairs and a comparable number of D mesons.

[1] 高エネルギー加速器研究機構 [Kō Enerugī Kasokuki Kenkyū Kikō] – High Energy Accelerator Research Organization

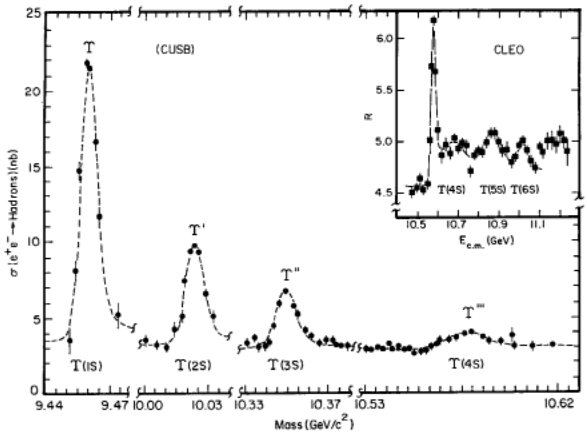


Figure 4.1.: Cross-section for inclusive hadron production by e^-e^+ annihilation in the energy region of the upsilon system [22].

4.2. Belle Detector

The Belle detector is an almost-4π, general-purpose spectrometer that covers the polar angle from 17° to 150° around the interaction region of KEKB. The high acceptance allows us to reconstruct full decay chain of D mesons by detecting their decay products.

The detector has a cylindrical coordinate system with longitudinal coordinate z, which is aligned with positron beam, radial coordinate r, and azimuth angle ϕ. The direction of the z-axis is in electron flight direction. The origin of the coordinate system is at the interaction point. The detector consists of the central barrel region and two end cap regions: forward end cap in the positive-z region and backward end cap in the negative-z region.

Figure 4.2 shows the layout of the Belle experiment. Subdetectors cover interaction point cylindrically. A particle passes silicon vertex detector, central drift chamber, time-of-flight detector, electromagnetic calorimeter, and K_L^0 and muon detector in the barrel region. In end cap region, a particle passes silica aerogel counter detector in the forward end cap, electromagnetic calorimeter, and K_L^0 and muon detector in both forward and backward end caps. The extreme forward calorimeter, which is installed on the front face of the compensation magnet's cryostat, measures luminosity by measuring rate of the electron-positron scattering. A full technical description of the detector is available in [23].

4.2.1. Silicon Vertex detector

The silicon vertex detector, with fine spacial resolution, measures position of the primary decay vertices. The hits in the silicon vertex detector are also used in track reconstruction.

The detector is comprised of double-sided silicon strips using CMOS technology. One side of the detector contains P^+ strips, the other side contains N^+ strips that are perpendicular to P^+ strips. Charged particles, which cross the detector, produce electron-hole

Figure 4.2.: Layout of the Belle detector

pair in depleted silicon bulk. Electrons and holes drifting to opposite sides of the detector produce electric signal in the strips. We reconstruct position of charged particles by correlating signals from P^+ and N^+ strips in time.

Two silicon vertex detectors existed in the Belle experiment. The first detector, SVD1, consisted of 3 detector layers and operated from the start of the experiment until October 2003 [23]. The detector suffered from radiation damage, which decreased the signal-to-noise ratio by 30 %. So the detector was replaced by one with radiation-hard components. This second detector, SVD2, consists of 4 detector layers and operated until the end of the experiment [24]. It covered a slightly larger polar angle and, due to a faster shaper, reduced the dead time — from 6.4 % in SVD1 to 1 % in SVD2. The upgrade lead to the improvement of the vertex location resolution.

4.2.2. Central Drift Chamber

The central drift chamber is used to determine momentum of a particle by measuring a track over a long leverarm [23]. The specific energy loss of a track in the drift chamber gives information about its particle type.

The central drift chamber consists of 13 cylindrical layers and is filled with a helium-ethane mixture. Charged particles ionize gas in the chamber. Ion then drift towards a sensitive wire where they induce electric signal. We reconstruct a two-dimensional projection of the track by combining hits that are close in space and in time. Two types of layers are installed alternatively: layers parallel to the z axis and layers at a small angle to the z axis. Timing coincidence of hits in these layers gives information about the z coordinate of the track in addition to the r and ϕ coordinates. The curvature of the track in constant magnetic field allows us to measure momentum of the particle.

4.2.3. Particle Identification Detectors

The primary goal of the silica aerogel Cherenkov counter and the time-of-flight detectors is to precisely identify species of the charged tracks.

The aerogel ring-imaging Cherenkov detector detects Cherenkov light, emitted by a charged track, in the aerogel medium [23]. This subdetector covers the barrel and the forward end-cap regions. The light yield measured by the detector is proportional to the shape of the Cherenkov light cone, which depends on the particle's mass and momentum. We can calculate mass of a particle using the light yield and momentum measured via the central drift chamber.

The time-of-flight detector covers the barrel region. It uses two layers of fast scintillator counters to measure the time a particle needs to travel the distance between the layers. With velocity and momentum, we can calculate mass of the particle and determine its species.

4.2.4. Electromagnetic Calorimeter

The electromagnetic calorimeter measures energy deposited in the detector by the particle. The calorimeter consists of the 8736 Cs(Tl)I crystals which cover full acceptance range of the Belle detector. The 30 cm deep Cs(Tl)I crystals used in Belle cover 16.2 units of radiation length. This allows them to fully absorb most electromagnetically interacting particles.

High segmentation of the calorimeter, while increasing the complexity and reducing the efficiency of the detector, is optimized for reconstruction of the photons. Photons leave very compact shower in the crystal. Typically, a photon candidate leaves more than 80 % of its energy in one crystal. To account for the case when the photon traverses more than one crystal, this condition is relaxed. The condition which we use in Belle considers the ratio of energies which the photon leaves in a central 3×3 cluster and in a central 5×5 cluster. We require this ratio to be

$$E9oE25 = \frac{E(3\times3)}{E(5\times5)} > 0.85$$

to select good photon candidates.

Information from particle identification detectors, specific energy loss and momentum measured in the central drift chamber, and energy, deposited in the electromagnetic calorimeter, flow into calculation of the likelihood ratios for a track hypothesis. I use the likelihood ratios in the reconstruction software to quantify the probability of a track being a specific particle.

Chapter 5.

Search for the Decay $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$

5.1. Analysis Overview

To calculate branching fraction of a decay channel, we must know three quantities:

- signal yield, obtained from the fit to reconstructed data,
- reconstruction efficiency, and
- number of D mesons in data sample.

But because D mesons are not resonantly produced at Belle, we cannot estimate their number in the data sample with sufficient precision. Therefore, I measured branching fraction relative to a well-known channel $D^+ \to K^- K^0_S \pi^+ \pi^+$ with branching fraction

$$\mathrm{BF}(D^+ \to K^- K^0_S \pi^+ \pi^+) = (0.234 \pm 0.018)\,\%. \tag{5.1}$$

As a cross-check, I reconstructed another four-particle D decay channel $D^+ \to \pi^- K^0_S \pi^+ \pi^+$. The normalizing channel has the same number and same types of charged tracks as the 5-particle decay mode, which I want to measure. This brings the advantage that systematic uncertainties of the measurement associated with measuring charged tracks cancel in the branching fraction calculation.

The analysis is performed on all decay channels in the same manner. First, I simulate all decay channels using Monte Carlo simulation. Events in the simulation are distributed according to available phase space. Then, I select events from phase space simulated data, signal Monte Carlo, and from full experiment simulation, generic Monte Carlo, using loose selection criteria. With these events, I train a boosted decision tree classifier that is used to obtain final data sample. I use final data sample to calculate efficiencies and define signal and background shapes for all decay channels. The efficiencies are used in the normalized branching fraction measurement

$$\mathrm{Br(channel)} = \left(\frac{\mathrm{yield}_{channel}}{\varepsilon_{channel}} \Big/ \frac{\mathrm{yield}_{norm}}{\varepsilon_{norm}} \right) \times \mathrm{Br(norm)}. \tag{5.2}$$

The final data sample is used to study our sensitivity to the signal. I first fit invariant mass of the simulated signal and background events independently to create models for their shape. Then, I add both models and fit the final data sample to obtain the signal yield. In addition, I explore sensitivity of the measurement by fitting data samples that contain different signal fractions.

5.2. Monte Carlo Simulations

To study selection efficiencies and signal shapes, I simulate signal decays using Monte Carlo technique. This simulation is called signal MC. For the 5-particle decay channel, I simulate a prompt D^+ decay, where D^+ immediately decays to final-state particles, and three subchannels, where D^+ first decays to $K^*K^*\pi$ which then decays to final-state particles. For every channel, including $K^*K^*\pi$ sub-channels, I generate 10^7 events with EvtGEN with kinematics of the decays distributed according to available phase space [25]. Detector response is simulated with GEANT3 software [26].

I study background shapes and composition on the full experiment simulation, called generic MC. I reconstruct three independent data sets and use one data set for preparing event selection and two other data sets for verification of the event selection on the background distribution.

5.3. Event Selection

I start decay reconstruction with event selection using loose selection criteria. The event selection algorithms focus on rejecting as many of background events as possible to reduce data size. I use same selection criteria in all decay channels. Due to common selection criteria, charged tracks have same kinematic distribution, which leads to cancellation of systematic uncertainties in normalizing branching fraction. The selection algorithm for different decay channels differ only in number and types of the reconstructed final-state particles.

Table 5.1 lists cuts used in event selection. First, I select events based on their 2nd Fox-Wolfram moment R_2 defined as

$$H_n = \sum_{i,j=1}^{N} \frac{|\vec{p}_i||\vec{p}_j|}{s} P_n(cos\Omega_{ij}), \tag{5.3}$$

$$R_2 = \frac{H_2}{H_0} \tag{5.4}$$

with $\vec{p}_{i,j}$ are four-momenta of the particles, P_2 is the Legendre polynomial of degree 2, and Ω_{ij} is the opening angle between particles [27]. The 2nd Fox-Wolfram moment discriminates event by their sphericity. I discard events with the normalized moment below 0.1 to remove jet-like events.

Neutral pions are reconstructed from two good photon candidates with E9oE25 ratio, described in section 4.2.4, greater than 0.85. I require χ^2 of the mass constrained fit to a pair photons to be smaller than 16 and momentum of π^0 to be greater than 0.22 GeV.

I reconstruct K^0_S from a $\pi^+\pi^-$ pair using the standard cut of the nisKsFinder algorithm to select good kaons [28]. NisKsFinder is a machine learning algorithm that uses 13 kinematic parameters of the K^0_S candidate and its daughter $\pi^+\pi^-$ to select candidates with high efficiency.

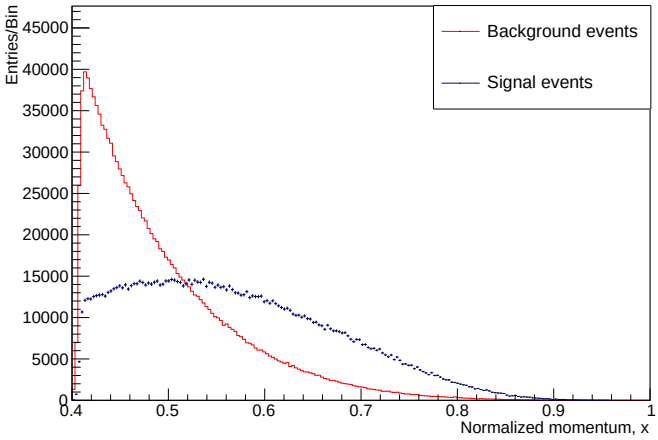


Figure 5.1.: Distribution of the normalized momentum for signal and background events in the $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ mode.

I select charged tracks using likelihood values, calculated by particle identification for every track and every mass hypothesis. To separate pions from kaons, I use the likelihood ratio $L_{K/\pi}$ that is calculated from probabilities of a track for being a pion L_π or a kaon L_K

$$L_{K/\pi} = \frac{L_K}{L_K + L_\pi}. \tag{5.5}$$

I additionally require charge tracks to originate from region close to interaction point and to have at least 2 hits in the vertex detector to reject poorly reconstructed tracks.

I combine all particles to a D^+ candidate. I require invariant mass of the D^+ candidate to be limited to a region around nominal D^+ mass. I fit D^+ vertex using all charged tracks of the D^+ candidate [29]. I require that the fit does not fail. This removes many purely combinatorial background events.

The normalized momentum of the D^+ candidate

$$x = \frac{P_D}{\sqrt{\frac{s}{4} - M_D^2}} \tag{5.6}$$

must be greater than 0.4. Figure 5.1 shows normalized distribution of normalized momentum for signal and background events. Background events peak at low value of x, while signal events peak at 0.5. Cutting on 0.4 reduces a large fraction of background events.

5.3.1. Geometric Acceptance and Selection Efficiency

I calculate geometric acceptance and signal selection efficiency by comparing total number of events generated in signal MC with number of detectable events and number of correctly reconstructed events. The number of detectable events is the number of decays

Parameter	Selection criteria	Comment
	Events	
R_2	> 0.1	2nd Fox-Wolfram moment
	π^0	
$\chi^2_{\pi^0}$	< 16	χ^2 of the mass fit
$\lvert\vec{p}_{\pi^0}\rvert$	> 0.22 GeV/c	
E9oE25, γ for π^0	> 0.85	Energy ratios for daughter photons
	Charged tracks	
L_e	< 0.6	probability of a track being an electron
L_p	< 0.6	probability of a track being a proton
$L_{K/\pi}$ for $K^\pm$	< 0.6	
$L_{K/\pi}$ for $\pi^\pm$	> 0.6	
dr	< 10 mm	impact parameter of a track in r direction
dz	< 15 mm	impact parameter of a track in z direction
Number of SVD hits	> 2 per charged track	
	D^+	
m_{D^+}	$(1.8196 < m_{D^+} < 1.9196)$ GeV/c^2	
x	> 0.4	normalized momentum

Table 5.1.: Selection criteria used in the skim module

where all final state-particles of the D^+ decay hit Belle detector. The number of correctly reconstructed events is calculated by matching reconstructed candidates and their daughter particles with MC truth.

I calculate geometric acceptance as a ratio of detectable particles to the total number of events:

$$A = \frac{N_{\text{detectables}}}{N_{\text{total}}}. \tag{5.7}$$

Geometric acceptance is limited by detector geometry and cannot be improved in analysis.

I calculate selection efficiency as a ratio of correctly reconstructed candidates to detectable events:

$$\varepsilon_{\text{sel}} = \frac{N_{\text{correct}}}{N_{\text{detectables}}}. \tag{5.8}$$

I calculate detection efficiency as a product of selection efficiency and geometric acceptance:

$$\varepsilon_{\text{det}} = A \cdot \varepsilon_{\text{sel}}. \tag{5.9}$$

Table 5.2 lists number of reconstructed events in generic MC. I reconstruct no signal events in the 5-particle channel because this channel has not been measured yet and is not

Decay	Number of reconstructed events		
	Total	Signal	Background
$D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$	1'960'396	0	1'960'396
$D^+ \to K^- K^0_S \pi^+ \pi^+$	15'104'297	453'042	14'651'255
$D^+ \to \pi^- K^0_S \pi^+ \pi^+$	118'150'470	2'609'381	115'541'089

Table 5.2.: Number of reconstructed events in generic MC

Decay	Acceptance, %	ε_{sel}, %	ε_{det}, %
$D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$	18.6	16.08	2.99 ± 0.0056
$K^{*+} K^{*-} \pi^+$	20	17.46	3.50 ± 0.0120
$K^{*+} \overline{K}^{*0} \pi^0$	17.1	12.07	2.07 ± 0.0090
$K^{*0} \overline{K}^{*0} \pi^+$	19.5	17.34	3.39 ± 0.0120
$D^+ \to K^- K^0_S \pi^+ \pi^+$	38.9	37.7	14.60 ± 0.0120
$D^+ \to \pi^- K^0_S \pi^+ \pi^+$	37.6	42.9	16.10 ± 0.0130

Table 5.3.: Geometric acceptance and selection efficiencies

a part of generic MC. Therefore, for efficiency calculation I use data from signal MC for all channels.

Table 5.3 lists acceptances and efficiencies for all channels. The efficiency of the $K^{*+}\overline{K}^{*0}\pi^0$ subchannel is significantly lower than the efficiencies of other subchannels. This effect is caused by π^0 momentum cut and different momentum distributions of π^0 in different subchannels. Section 5.4.6 describes this effect in more details.

5.4. Post-Skim Event Selection

Due to very loose cuts in the skim module, the ratio of signal events to background events at the expected branching fraction 10^{-4} is $7.5 \cdot 10^{-4}$. This ratio is too low to measure the signal fraction. Therefore, additional event selection is done on the pre-selected data.

For the final event selection, I train boosted decision trees for background separation, optimize cuts on the kinematic variables, and use best candidate selection to keep a single candidate per event.

5.4.1. Background Reduction with Boosted Decision Trees

I use machine learning algorithm 'boosted decision tree', the BDT, to suppress background events [30]. I separate signal events from the signal MC and background events from the generic MC in two data samples with the size 1/4 and 3/4 of initial data samples. The

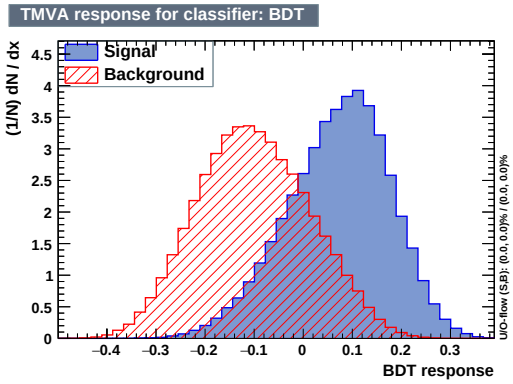


Figure 5.2.: BDT response

smaller sample is used for training of the BDT. The larger sample is used for verification and efficiency calculations.

I train the BDT on 31 kinematic parameters available from the D^+ candidates and their daughter particles. Appendix A lists parameters with the variable importances provided by the TMVA.

Figure 5.2 shows BDT response for the 5-particle channel. It shows that BDT learns differences between signal and background events and can discriminate them. I use BDT response as a parameter in the final event selection.

5.4.2. Best Candidate Selection

Additional background suppression is performed using best candidate selection. On average, selection algorithm selects 2 background candidates per event. By selecting the best candidate, I can eliminate 35 % of the candidates. For use in the best candidate selection, I considered 4 possible parameters:

- BDT response,
- vertex fit p-value,
- deviation of the D^+ candidate's invariant mass from the PDG value, and
- normalized momentum x.

I order all candidates in an event by each of these parameters. For each of the parameters, I select the D^+ candidate with the best value.

Figure 5.3 shows invariant mass distribution of the background events after best candidate selection. Selection, based on deviation of the invariant mass, produces a peaking shape in the distribution. Selection, based on the BDT response, produce peaking shape of the background distribution in the normalizing channels. These parameters are not suitable as a selection criteria. Other parameters produce comparable shapes of the invariant mass distribution in the background events.

Parameter	Signal efficiency, %
X	80.2
Vertex fit p-value	79.8
BDT response	75.6
D^+ mass deviation	89.4

Table 5.4.: Retention efficiency of the best candidate selection cuts

Best-candidate selection also affects shape of the invariant mass distribution of the partially reconstructed signal events that contain a wrong photon in π^0. These events are important because the shape of their invariance mass distribution is used as part of signal PDF in the fits. Figure 5.4 shows invariant mass distributions of the partially reconstructed signal events with different best-candidate selection criteria. Selection, based on the BDT response and the vertex fit p-value, preserves a broad Gaussian shape. Selection, based on the normalized momentum, produces irregular shape. This eliminates normalized momentum as a selection criteria.

Table 5.4 lists signal retention efficiencies for the best candidate selection algorithm with different selection parameters. Best candidate selection based on the vertex fit p-value has higher signal retention fraction than the one based on the BDT response. Therefore, I use best candidate selection algorithm that selects events with the best vertex fit p-value.

5.4.3. Final Cuts

Final event selection selects events with successful vertex fit, cuts on the normalized momentum, and on the BDT response. I do not cut on the BDT response in normalizing channels because it disturbs shape of the invariant mass distribution. If multiple candidates for an event are selected after applying the cuts, a candidate is selected with the best p-value of the vertex fit.

A 2D scan determines optimal values for a cut on the normalized momentum and the BDT response. As the figure-of-merit for the cut, I use following quantity:

$$r = \frac{N_{signal}}{\sqrt{N_{background}}}. \tag{5.10}$$

Figure 5.5 shows results of the scan. The scan determined the largest signal significance at normalized momentum 0.4 and BDT response 0.15.

Figure 5.6 shows invariant mass distributions for normalizing channels obtained after applying the optimized cuts. Signal peaks are clearly visible in the distributions. Figure 5.7 shows invariant mass distribution of the 5-particle channel with 530 signal events, randomly selected from the signal MC and full background from the generic MC. With calculated signal detection efficiency, 530 signal events correspond to branching fraction that is 10 times lower than for the channel $D^+ \to K^- K^0_S \pi^+ \pi^+$.

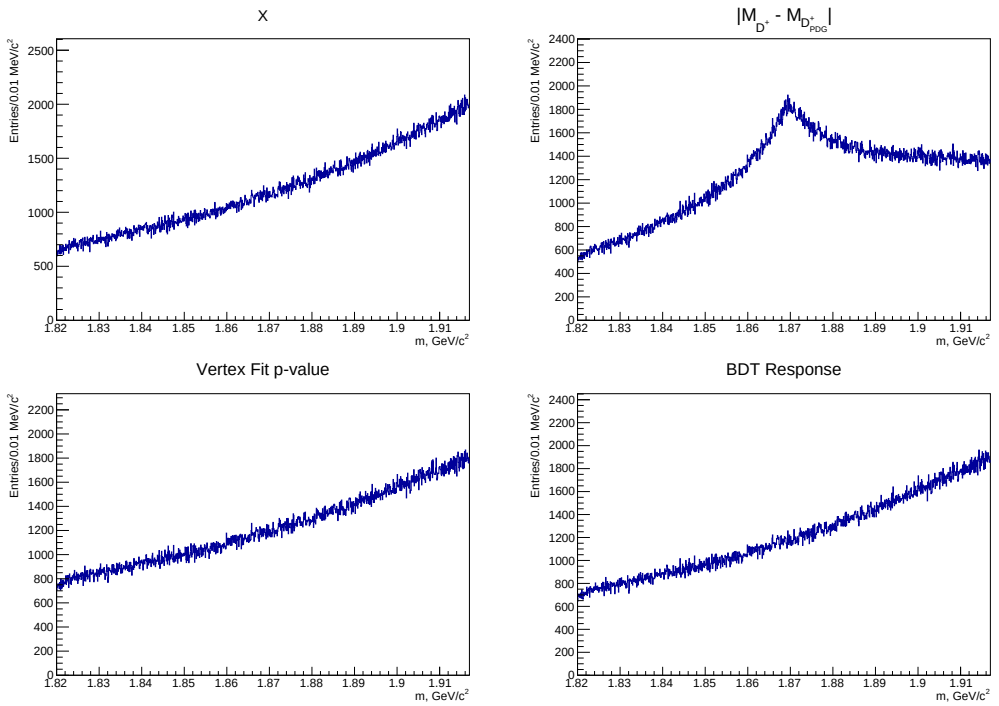


Figure 5.3.: Invariant mass distributions of the background events of the $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ decay after best candidate selection cuts

Channel	Cuts	Event Type	Cut efficiency, %	Total signal efficiency, %
$D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$	best(P_{Vtx}) & $P_{Vtx} > 0$ & X > 0.4 & ω η-Veto & BDT > 0.15	signal background	6.6407 0.6883	0.5129
$D^+ \to K^- K^0_S \pi^+ \pi^+$	best(P_{Vtx}) & $P_{Vtx} > 0$ & X > 0.4 & ω η-Veto	signal background	76.4034 46.4912	12.9822
$D^+ \to \pi^- K^0_S \pi^+ \pi^+$	best(P_{Vtx}) & $P_{Vtx} > 0$ & X > 0.4 & ω η-Veto	signal background	68.4190 34.8713	14.3882

Table 5.5.: Cut efficiencies

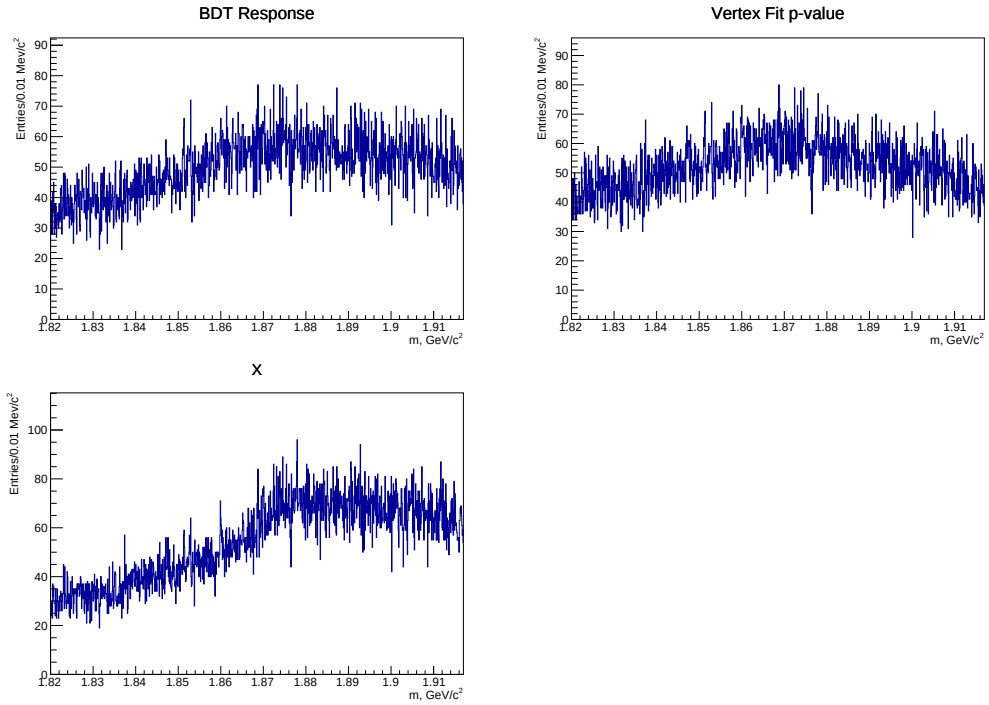


Figure 5.4.: Invariant mass distribution of the partially reconstructed events from the signal MC after the best candidate selection cuts

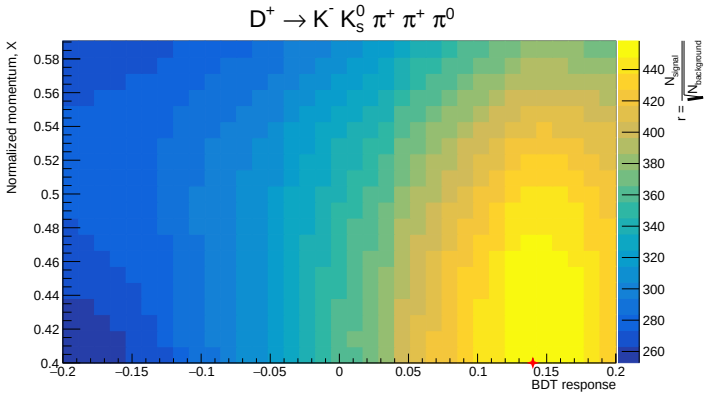


Figure 5.5.: Results of the cut scans over the BDT response and the normalized momentum parameters. Red marker shows optimal cut position

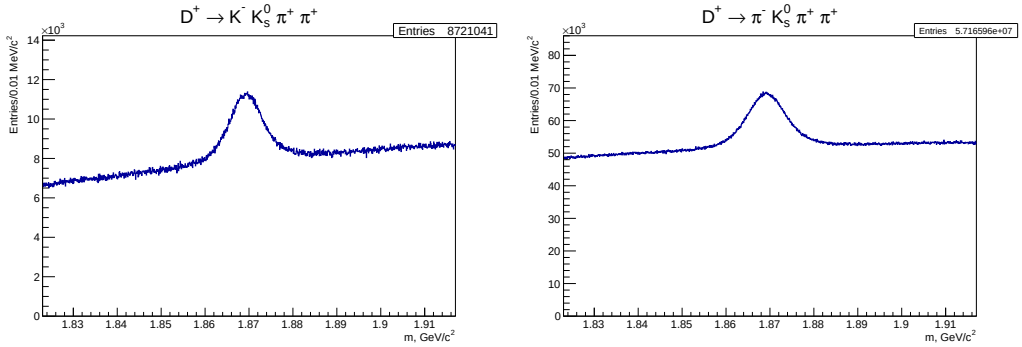


Figure 5.6.: Invariant mass distribution of the normalizing channels after applying background reduction cuts, generic MC

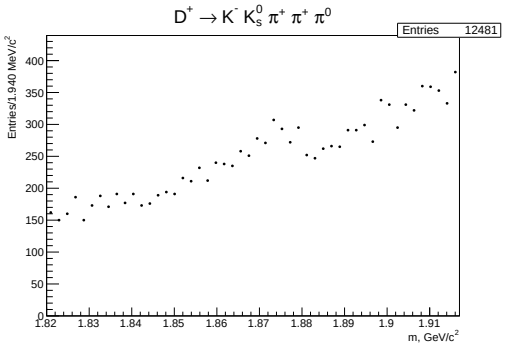


Figure 5.7.: Invariant mass distribution for the five-body mode. It shows 530 signal events from the signal MC, which correspond to the BR $2.28 \cdot 10^{-4}$, and full background data sample from the generic MC

Table 5.5 lists cut efficiencies and total signal efficiencies. I included the ω η-veto into the cuts to suppress the peaking background component. The veto is described in section 5.4.5.

5.4.4. Background Studies

The events reconstructed from the generic MC contain only background events for the decay $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$. This allows us to study background composition by matching final-state particles with their MC truth.

I classified background events by number of daughter particles in D^+ candidate which have common mother particle. The classes are

- 2: two daughters with a common mother,
- 2+2: two daughters with a common mother, other two daughters with another common mother,

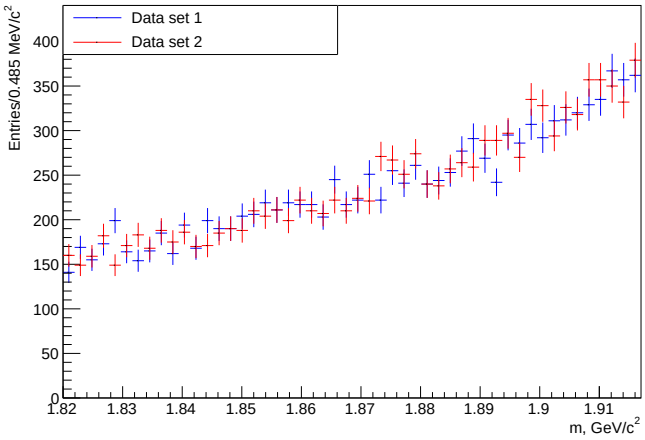


Figure 5.8.: Comparison of the invariant mass distributions of the background events in the validation data sets

- 3: three daughters with a common mother,
- 3+2: three daughters with a common mother, other two daughters with another common mother,
- 4: four daughters with a common mother,
- 5: five daughters with a common mother, and
- combinatorial: all daughter particles originate from different mother particles.

Figure 5.9 shows invariant mass distribution of the background events separated into these classes. All classes show smooth shape without strong peaking components.

Table 5.6 lists the most prominent background contributions. The most prominent channels that contribute to background are K^* and D^{*+} decays. The D^{*+} contribution explains rising shape of background distribution towards higher masses. Because subchannels of the 5-particle decay channel contain K^*, suppressing background contribution from K^* may affect composition of subchannel amplitudes. Therefore, I do not suppress them and fit the invariant mass distribution by a polynomial shape.

Figure 5.8 shows invariant mass distributions of the background events in the validation data sets. Both data sets show good agreement on the shape of the distribution.

5.4.5. Peaking Component in the Background Distribution

Later tests showed that background invariant mass distribution has a small peaking component at D^+ mass. The peaking component is not easily recognized in the invariant mass distribution but causes a constant offset to the signal fraction parameter in the fit.

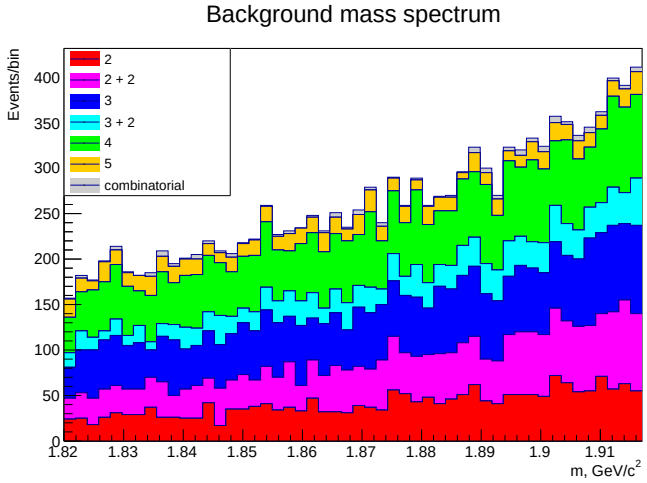


Figure 5.9.: Invariant mass distribution of the classified background events

Class	Decay	Number of Events	Fraction of Events in Bin
2+2	$K^{*+} \to \pi^+ K^0_S$	701	0.14
2+2	$\omega(782) \to \pi^- \pi^+ \pi^0$	640	0.13
4	$D^{*+} \to K^- \pi^0 \pi^0 \pi^+ \pi^+$	363	0.10
2	$K^{*+} \to \pi^+ K^0_S$	349	0.16
3	$K^{*0} \to \pi^- K^+$	317	0.06

Table 5.6.: Most prominent background channels

This background contribution originates from the decay $D^+ \to K^- \pi^+ \pi^+ \pi^+ \pi^- \pi^0$ which is included in generic MC as two channels:

$$D^+ \to \overline{K}^{*0} \omega \pi^+$$
$$D^+ \to \overline{K}^{*0} \eta \pi^+$$

The decay has a poorly measured branching fraction and was removed from the PDG listing in the 2004 edition [31]. To suppress this channel, I apply a veto for this decay and assign a systematic uncertainty that is proportional to uncertainty of the branching fraction for the decay $D^+ \to K^- \pi^+ \pi^+ \pi^+ \pi^- \pi^0$. After vetoing this decay, the peaking component also disappears.

ω–η Veto

The ω–η veto suppresses the peaking component from the $D^+ \to K^- \pi^+ \pi^+ \pi^+ \pi^- \pi^0$ decay. Figure 5.10 shows the squared invariant mass distribution of the combinations of the final state particles used in the veto. I use π^+ and π^- from the K^0_S, the π^+'s which are ranked

Parameter	Value, %
$\varepsilon_{\text{signal}}$	0.6719±0.0024
$\varepsilon_{\eta\text{-}\omega}$	0.0560±0.0020
$\varepsilon_{\text{veto signal}}$	76.3390±0.4034
$\varepsilon_{\text{veto }\eta\text{-}\omega}$	31.9580±2.1629

Table 5.7.: Reconstruction and veto efficiencies for the signal channel and $\eta - \omega$ background modes

by their energies, as well as the π^0. There are two peaks that show peaking structure in background distributions. The peaks are at the ω and the η masses.

The ω–η veto cuts out events around nominal mass of the resonances within 3σ of the peak widths.

Because branching fractions of η–ω channels are poorly known, this measurement can improve them. I estimate number of events in the η–ω channel from signal yields, obtained in the fits to data with and without the veto. I express the signal yield **Y** as following:

$$Y^{\text{sig}}_{\text{no veto}} = f \cdot N \cdot \varepsilon_{\text{signal}} + (1-f) \cdot N\varepsilon_{\eta\text{-}\omega} \tag{5.11}$$

with **N** is the initial number of events in the data set which are reconstructed as the 5-particle final state, **f** is the fraction of 5-particle signal events in the signal yield, $\varepsilon_{\text{signal}}$ is the efficiency of selecting 5-particle signal events, and $\varepsilon_{\eta\text{-}\omega}$ is the efficiency of selecting $\eta - \omega$ background events. Next, I express signal yield with the veto applied:

$$Y^{\text{sig}}_{\text{veto}} = f \cdot N \cdot \varepsilon_{\text{signal}} \cdot \varepsilon_{\text{veto signal}} + (1-f) \cdot N\varepsilon_{\eta\text{-}\omega} \cdot \varepsilon_{\text{veto }\eta\text{-}\omega} \tag{5.12}$$

with $\varepsilon_{\text{veto signal}}$ is the efficiency of the veto for selecting 5-particle signal events from the un-vetoed data set, and $\varepsilon_{\text{veto }\eta\text{-}\omega}$ is the efficiency of the veto for selecting $\eta - \omega$ background events from the un-vetoed data set.

By calculating selection efficiencies in MC and the signal yields from the fits to the data, reconstructed with and without $\eta - \omega$ veto, I can express the fraction of the 5-particle mode in the data set as:

$$\text{f} = \frac{\varepsilon_{\eta\text{-}\omega}\left(Y^{\text{sig}}_{\text{no veto}} \cdot \varepsilon_{\text{veto }\eta\text{-}\omega} - Y^{\text{sig}}_{\text{veto}}\right)}{Y^{\text{sig}}_{\text{veto}}(\varepsilon_{\text{signal}} - \varepsilon_{\eta\text{-}\omega}) - Y^{\text{sig}}_{\text{no veto}}(\varepsilon_{\text{signal}} \cdot \varepsilon_{\text{veto signal}} - \varepsilon_{\eta\text{-}\omega} \cdot \varepsilon_{\text{veto }\eta\text{-}\omega})}. \tag{5.13}$$

Table 5.7 lists the reconstruction and veto efficiencies calculated from the signal MC.

5.4.6. Efficiency Corrections

Lower efficiency of $\text{K}^{*+}\overline{\text{K}}^{*0}\pi^0$ subchannel can be explained by lower momentum of π^0 generated in this channel. Figure 5.11 shows distribution of the reconstructed π^0 momentum in the sub-channels of the $\text{D}^+ \to \text{K}^-\text{K}^0_S\pi^+\pi^+\pi^0$decay. The π^0 momentum cut cuts

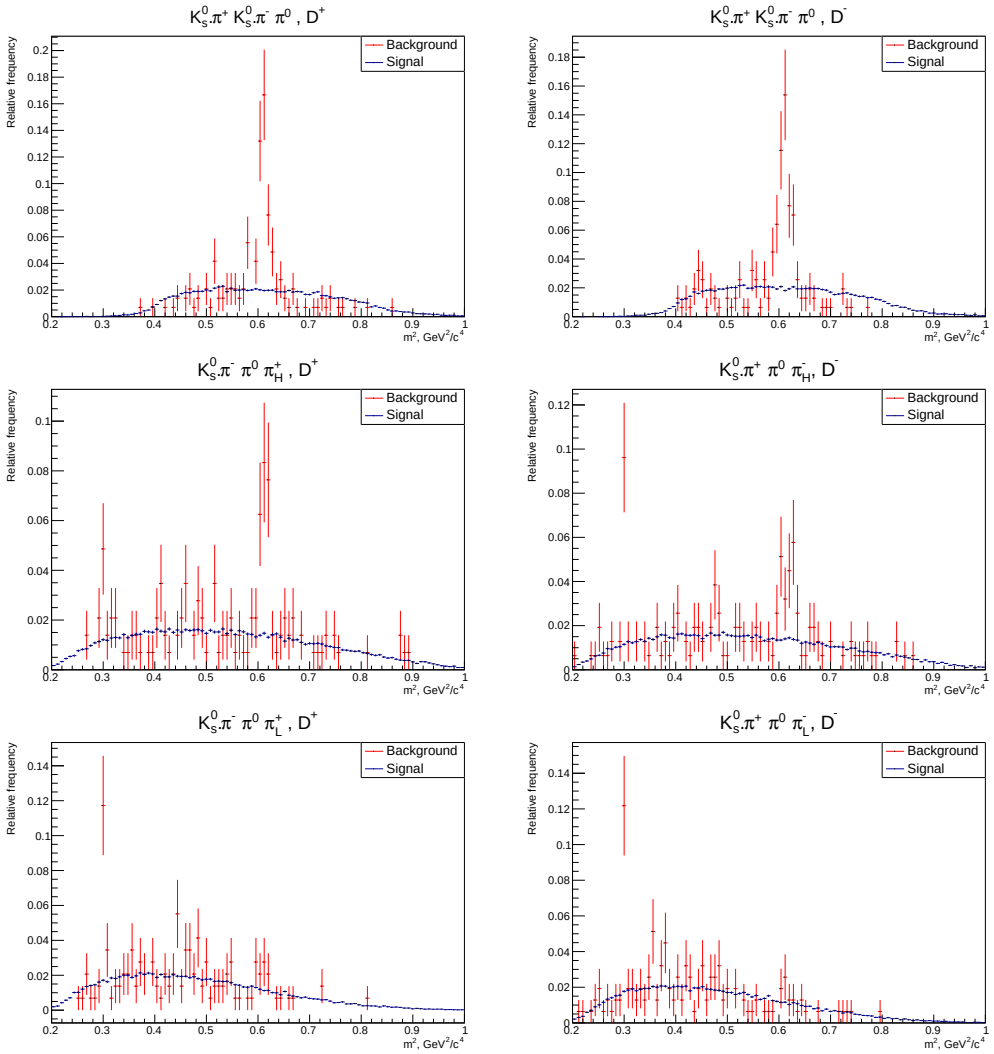


Figure 5.10.: Distributions of the squared invariant masses for different combination of the final state particles. $K^0_S.\pi$ denotes the pion from the reconstructed K^0_S. $\pi^+{}_{H/L}$ denotes energy ranking of the pion. ”H” means pion with the highest energy. ”L” means pion with lower energy

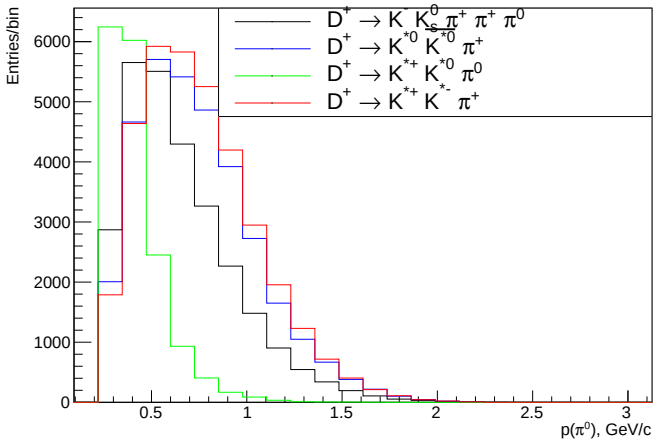


Figure 5.11.: The distribution of the reconstructed π^0 momentum

larger portion of the signal in the $K^{*+}\overline{K}^{*0}\pi^0$ sub-channel than in other sub-channels. This leads to lower efficiency in this channel in respect to other sub-channels.

To quantify this effect, I calculate effective efficiency for each subchannel using a 2-dimensional slice of the 8-dimensional phase space of this decay. For the calculation, I use 2-dimensional distributions of the invariant masses for a combination of $\pi^+\pi^+\pi^0$ and $K^-K^0_S\pi^0$.

First, I calculate the efficiency map for the prompt 5-particle decay by dividing the reconstructed events by the generated events. Figure 5.12 shows the efficiency map for the 5-particle decay mode. The efficiency map shows almost uniform efficiency distribution over the whole 2-dimensional slice of the phase space.

Figure 5.13 shows the same two-dimensional slice of the phase space for every sub-channel. The $K^{*+}\overline{K}^{*0}\pi^0$ subchannel shows different phase space distribution than other subchannels. I use these distributions to calculate weighted efficiencies for every subchannel. I calculate weighted efficiency by summing efficiencies weighted by the number of events in the same bin:

$$\varepsilon_w = \sum_i \varepsilon_i \cdot \frac{N_i}{\sum_j N_j} \tag{5.14}$$

Table 5.8 lists weighted efficiencies for every subchannel and effective efficiency. I calculate effective efficiency as a mean of the weighted efficiencies. The effective efficiency deviates for different subchannel from 0.5284 to 0.6253. This is the contribution of effective efficiency to the systematic uncertainty of the measurement.

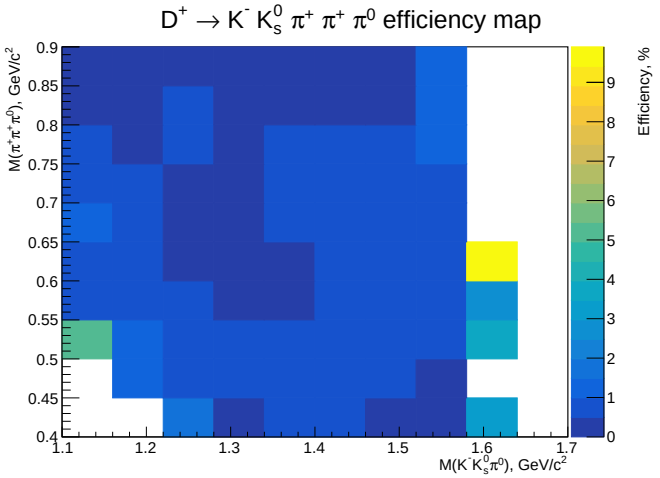


Figure 5.12.: Efficiency map for the prompt 5-particle decay mode

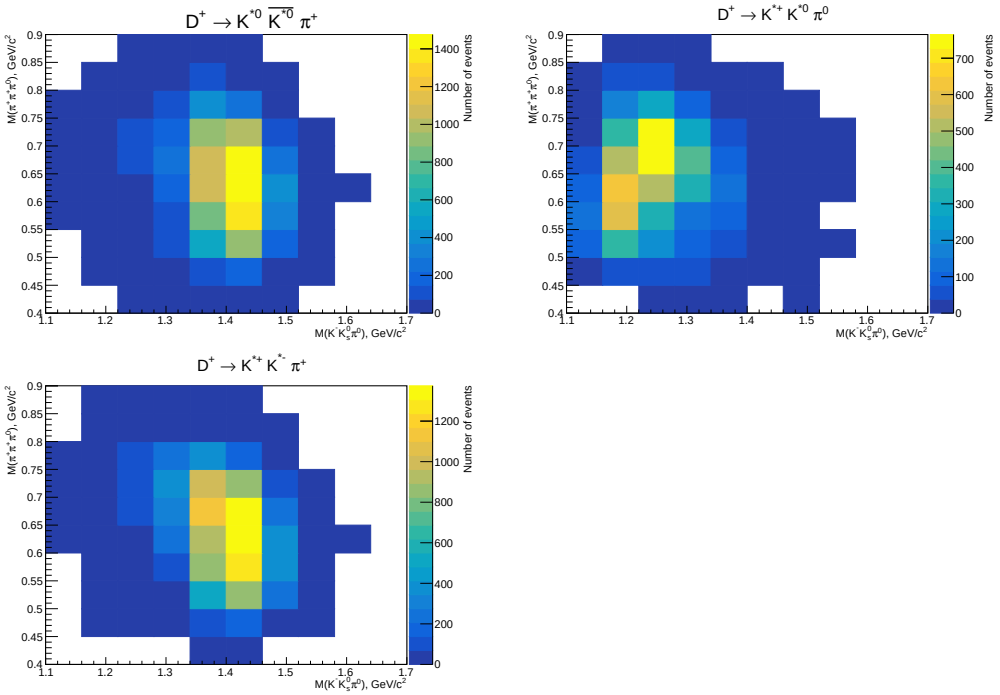


Figure 5.13.: Two-dimensional slice of the phase space of the sub-channels used for calculation of the weighted efficiencies

Sub-channel	Weighted efficiency, %	Deviation from mean, % of mean value
$K^{*+}K^{*-}\pi^+$	0.5284	-0.0338
$K^{*0}\overline{K}^{*0}\pi^+$	0.5328	-0.0294
$K^{*+}K^{*0}\pi^0$	0.6253	+0.0631
Effective efficiency	$0.5622^{+0.0631}_{-0.0338}$	

Table 5.8.: Weighted efficiencies for the sub-channel

5.5. Fit to the Data

I fit a model to invariant mass distribution of events that pass final event selection to calculate fraction of signal events in data. This requires knowledge of the shapes, produced by signal and background events. I study invariant mass distributions produced in Monte Carlo simulations to parametrize shapes for signal and background events separately. Then, I combine signal and background shapes and test combined model with artificial data samples that contain full background data sample and variable number of signal events. Finally, before fitting combined model to full data sample, I cross check my parametrization with 10 % of data.

All fits are done by maximizing unbinned log likelihood using the Bayesian Analysis Toolkit framework [32].

5.5.1. Signal Shape

I model the signal shape with Gaussian and double-sided crystal ball distributions. Crystal ball distribution describes asymmetric Gaussian distribution with a power-law tail. The distribution was first used in the crystal ball experiment to describe photon energy deposition in scintillator crystals [33]. Similar type of energy deposition is also found in this analysis because I reconstruct π^0 from photons detected in electromagnetic calorimeter. The crystal ball distribution is constructed of a Gaussian core with mode $\overline{m}$ and standard deviation σ, and a power-law tail with transition point α in units of σ and a power-law exponent n. Double-sided crystal ball distribution

$$DCB(m) \propto \begin{cases} \left(\frac{n_L}{\alpha_L}\right)^{n_L} \cdot e^{-\frac{\alpha_L^2}{2}} \cdot \left(\frac{n_L}{\alpha_L} - \alpha_L - \frac{m-\overline{m}}{\sigma}\right)^{-n_L} & ,if \frac{m-\overline{m}}{\sigma} < -\alpha_L \\ e^{\frac{(m-\overline{m})^2}{2\sigma^2}} & ,if -\alpha_L < \frac{m-\overline{m}}{\sigma} < \alpha_H \\ \left(\frac{n_H}{\alpha_H}\right)^{n_H} \cdot e^{-\frac{\alpha_H^2}{2}} \cdot \left(\frac{n_H}{\alpha_H} - \alpha_H + \frac{m-\overline{m}}{\sigma}\right)^{-n_H} & ,if \frac{m-\overline{m}}{\sigma} > \alpha_H. \end{cases} \quad (5.15)$$

is constructed similar to the crystal ball distribution with exponential tails on both high-mass side with transition point α_H and low-mass side with transition point α_L.

Gaussian distributions are of the form

$$\text{Gauss(m)} \propto e^{-\frac{(m-\overline{m})^2}{2\sigma^2}}, \quad (5.16)$$

with $\overline{m}$ is the mode, and σ is the standard deviation. One Gaussian distribution has its standard deviation comparable with the standard deviation of the double-sided crystal ball distribution. Another distribution is by factor 10 broader. I use it to model shape of the distribution that comes from signal events with a misreconstructed π^0.

The full signal model is the weighted sum of two Gaussian distributions and the double-sided crystal ball distribution

$$\text{signal(m)} = (1 - f_{\text{Gauss,broad}}) \cdot (f_{\text{DCB}} \cdot \text{DCB(m)} + (1 - f_{\text{DCB}}) \cdot \text{Gauss}_{\text{narrow}}(m)) \quad (5.17)$$
$$+ f_{\text{Gauss,broad}} \cdot \text{Gauss}_{\text{broad}}(m) \quad (5.18)$$

with f_{DCB} and $f_{\text{Gauss,broad}}$ are the weight parameters with the range between 0 and 1. The model has in total 12 free parameters. Figure 5.14 shows good agreement between the signal model and signal events from signal MC.

5.5.2. Background Shape

I fit background shape with the polynomial of degree 2:

$$\text{bkg}(m) \propto c_0 + m \cdot c_1 + m^2 \cdot c_2 \quad (5.19)$$

with c_0, c_1, and c_2 are the coefficients of the polynomial. Modelling of the background distribution showed strong correlation of both free parameters. Therefore, the background model has a single free parameter c_1. Figure 5.15 shows good agreement between the background model and background events reconstructed from generic MC.

5.5.3. Combined Fits

The combined fit is constructed as the weighted sum of the signal and background models:

$$\text{model(m)} = \text{f}_{\text{signal}} \cdot \text{signal(m)} + (1 - \text{f}_{\text{signal}}) \cdot \text{background(m)} \quad (5.20)$$

The total number of the free parameters for the combined model is 13.

For the combined fit, I don't fix parameters, but leave them float in the range of $\pm 5\,\sigma$ around the MC determined values which were determined during the fits to the pure signal and background data samples. The parameter's prior distribution is set to the posterior distribution of the parameter calculated in the fits to the pure signal and background data samples. The prior for the signal fraction parameter is set to a flat distribution. This gives us the estimation for the variation of the parameters in the fit. Variation of parameters is normally treated as a systematic uncertainty and now it is included in the statistic uncertainty.

Figure 5.16 shows result of the fit to the invariant mass of the background events plus 530 signal events which correspond to estimated branching fraction $2.28 \cdot 10^{-4}$. Figure 5.17 shows the posterior distribution of the signal fraction parameter for this fit. The posterior distribution of the signal fraction parameter from the fit has Gaussian shape and agrees within 1 σ interval with the true value.

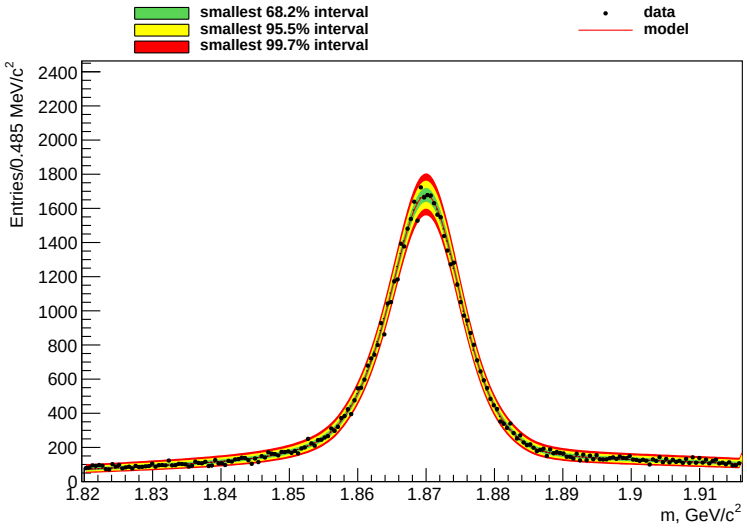


Figure 5.14.: Fit of the signal model to the invariant mass distribution of the signal events. χ^2/NDF of the fit is 1.23. Green, yellow, and red bands indicate approximately 1, 2, and 3 σ prediction based on the posterior distribution of the model. Following plots use the same convention

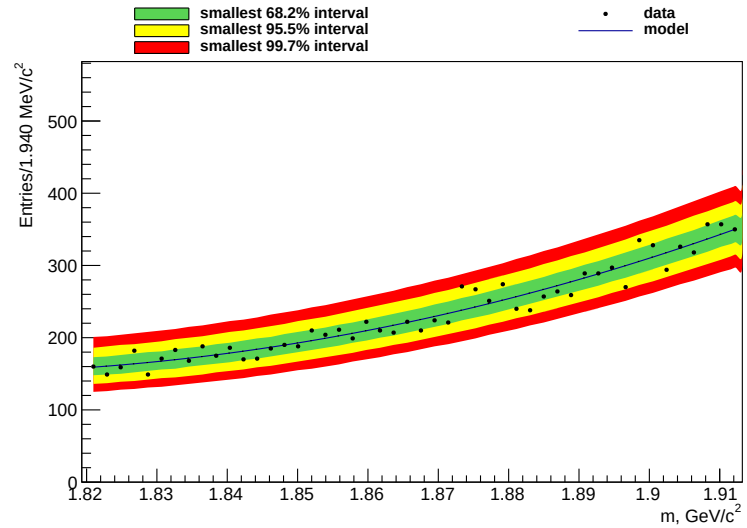


Figure 5.15.: Fit of the background model to the invariant mass distribution of the background events. χ^2/NDF of the fit is 1.00

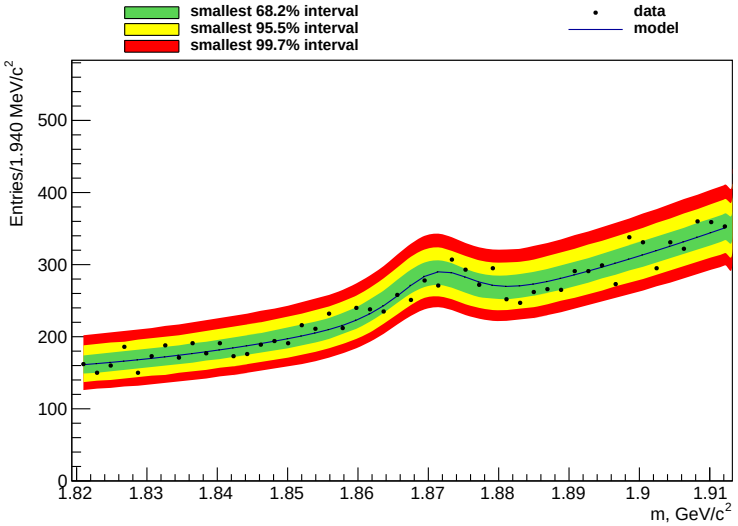


Figure 5.16.: Fit of the combined model to the invariant mass of the full background data sample plus 530 signal events which correspond to BR = $2.28 \cdot 10^{-4}$. χ^2/NDF of the fit is 0.82

Figures 5.18 show examples for update of our knowledge about parameters by the fit. I set DCB fraction parameter to a Gaussian prior that is determined from MC. The parameter is allowed to vary within 5 σ of its value. Fit slightly updates this parameter to match the shape of signal distribution in data.

I left signal fraction parameter unconstrained in the fit. This is indicated by the flat prior in the update plot. Posterior distribution of the parameter, sampled by the fitter, follows marginalized likelihood distribution of this parameter.

5.5.4. Sensitivity Scans

To estimate our sensitivity to the signal, I emulate possible branching fractions by adding a variable number of signal events to the full background data sample. Then, I fit two models to data to calculate a maximum of the likelihood function: a background-only model with the fixed signal fraction to 0 as a null hypothesis, and a combined model with the floating signal fraction parameter. Because I use the Bayesian approach by setting strong prior which drives the posterior distributions for all parameters, except for the signal fraction, setting the signal fraction to 0 reduces the number of degrees of freedom by approximately 1. Significance of the fit results is calculated using the Wilks' theorem with one degree of freedom difference between models as

$$s = \sqrt{-2\ln\left(\frac{L_0}{L_{sig}}\right)} \tag{5.21}$$

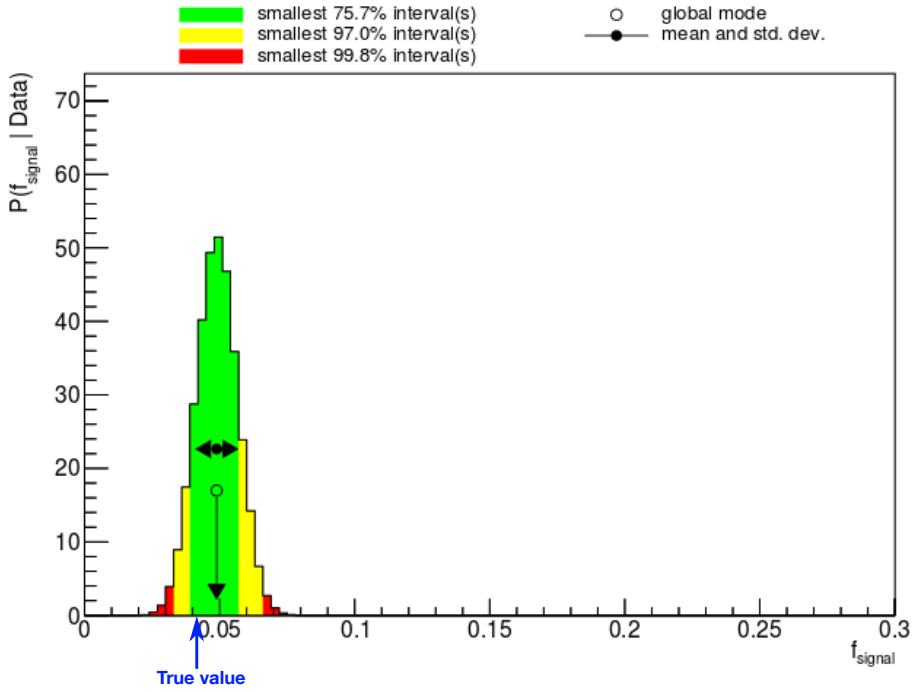
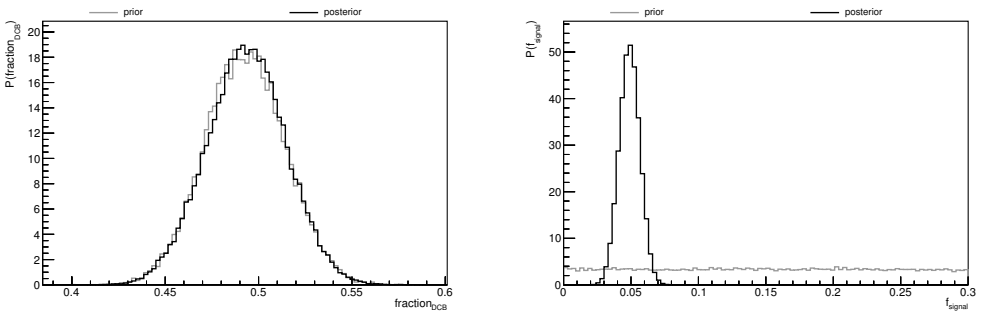


Figure 5.17.: Posterior distribution of the signal fraction parameter



(a) Update of the DCB fraction parameter (b) Update of the signal fraction parameter

Figure 5.18.: Update of the knowledge about parameters by the fit

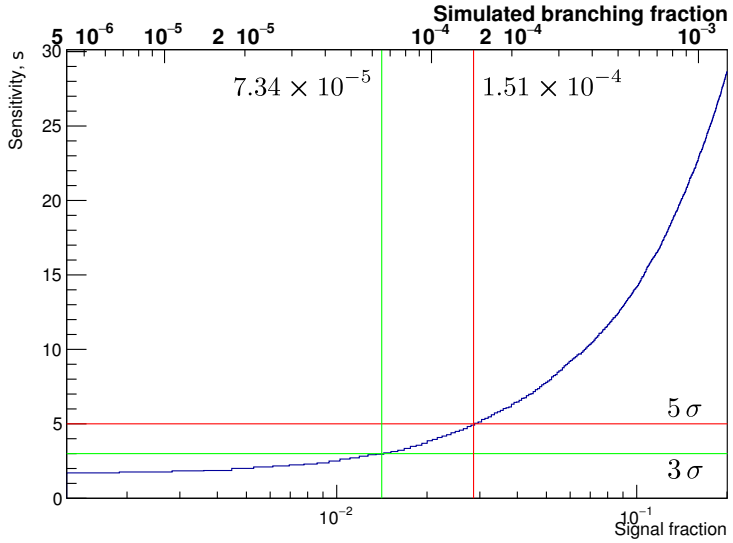


Figure 5.19.: Sensitivity of the combined fit

Figure 5.19 shows sensitivity plot obtained from the sensitivity scan. The sensitivity of 5σ is reached at 350 events which correspond to BR = $1.51 \cdot 10^{-4}$.

For the sensitivity values below 5σ, I calculate the Bayesian upper limit at 95 % confidence level. This value is calculated by determining signal fraction parameter which corresponds to the value 0.95 of the cumulative distribution function of the parameter's posterior distribution shown in figure 5.17. Figure 5.20 shows the exclusion plot at the 95 % confidence level of the measured signal fraction.

Figure 5.21 shows linearity of the fit obtained from the sensitivity scans. The signal fraction measured in the sensitivity scan is plotted against the true signal fraction injected into the data set. The error of the measured signal fraction is the standard deviation of the posterior distribution of the signal fraction parameter. The true signal fraction is within $1\ \sigma$ of the measured signal fraction over the full region used in this study.

5.6. Cross Checks

I performed three types of cross checks. First, I repeated the linearity scans with the mass range shrunk by 0.005 and 0.01 GeV/c^2 from both ends of the invariant mass range. The partially reconstructed signal component has more events in the outer bins of the invariant mass histogram than the fully reconstructed component. Therefore, the change of the mass range also changes the fraction of both components. I take this effect into account by obtaining the fraction of the partially reconstructed signal component from the signal-only fit within the given ranges. Then, I use the obtained fraction in the linearity scans. Figure 5.22 shows the result of the scan. The linearity scans show, that the sensitivity of the fit remains linear in the region of the expected signal fraction.

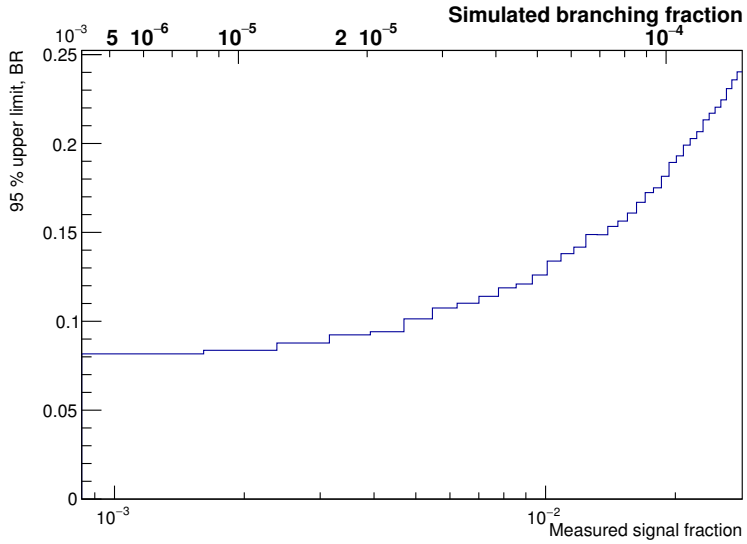


Figure 5.20.: 95 % upper limit on the measured signal fraction

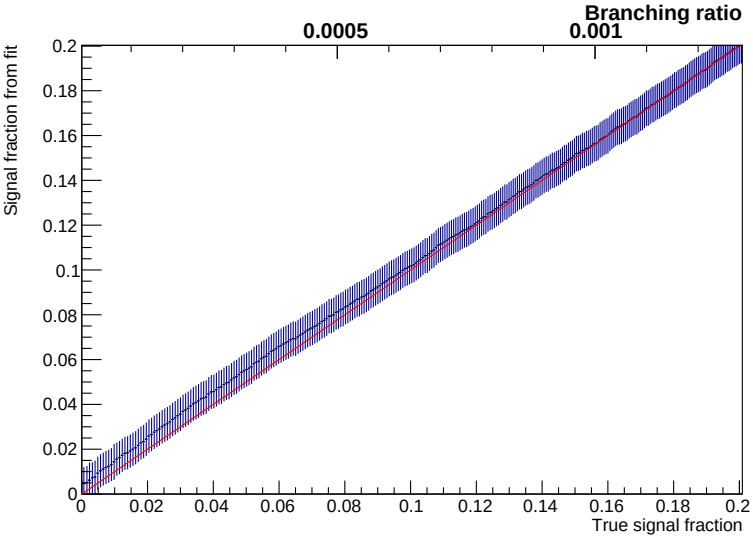
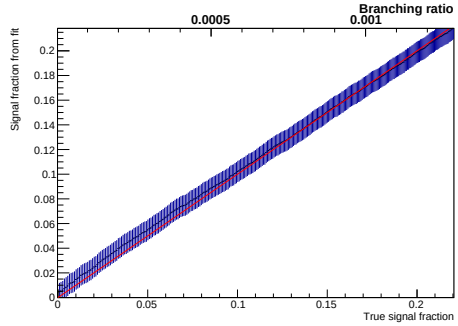
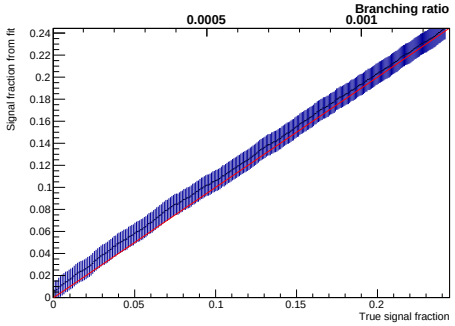


Figure 5.21.: Linearity test of the toy study. The red line shows the line with slope 1 and the intercept at (0, 0)

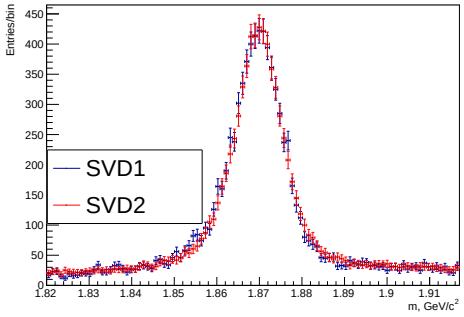


(a) Mass ranges reduced by 0.005 GeV/c^2

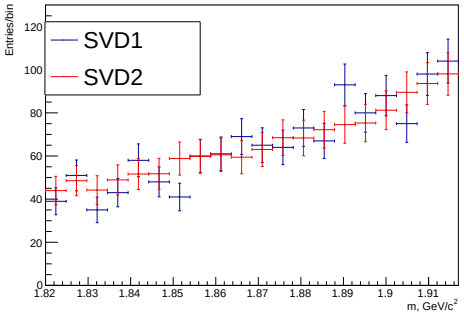


(b) Mass ranges reduced by 0.01 GeV/c^2

Figure 5.22.: Linearity test of the toy study with reduced mass ranges. The red line shows the line with slope 1 and the intercept at (0, 0)



(a) Signal events



(b) Background events

Figure 5.23.: Comparison of signal and background shapes of the invariant mass distribution recorded with SVD1 and SVD2

Next, I checked signal and background shapes of the invariant mass distribution recorded with SVD1 and SVD2. Figure 5.23 compares normalized invariant mass distributions for signal and background events. Both signal and background distributions agree in shape for SVD1 and SVD2 data.

Finally, I checked the effect of different fit shapes on the quality of the fit. Figure 5.24 shown the fit of the model that consists of the sum of 3 Gaussian distributions to the signal events. Figures 5.25 and 5.26 show the fit of the models that consist of polynomials of degree 1 and 3 to the background events. Figures 5.27 shows the combined fits with the polynomial of degree 1 as a background shape and the sum of 3 Gaussian distributions as a signal shape. All fits show worse agreement with data than the combined model described earlier.

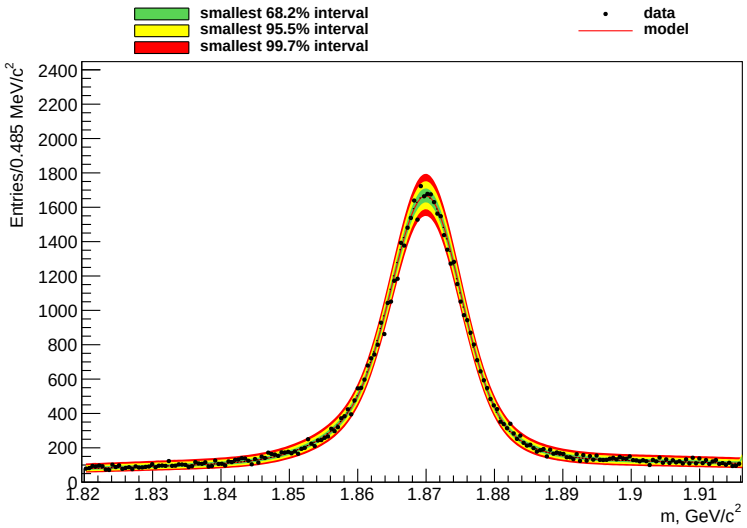


Figure 5.24.: Fit of the signal events with the sum of 3 Gaussian distributions. χ^2/NDF of the fit is 1.27

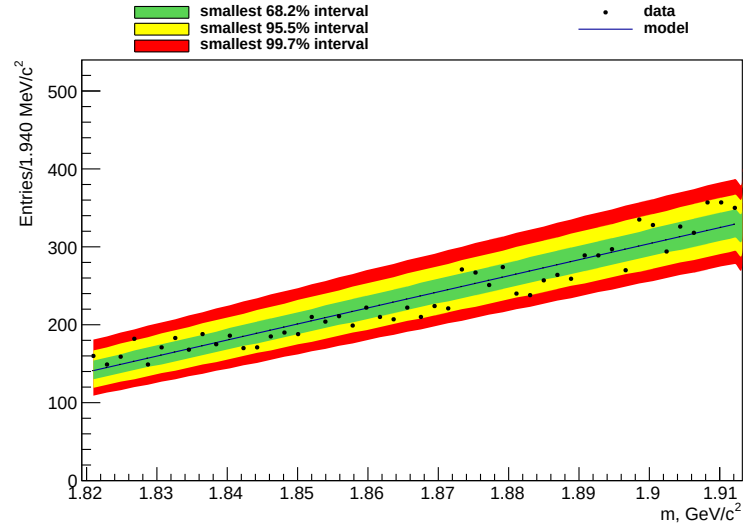


Figure 5.25.: Fit of the background events with the polynomial of degree 1. χ^2/NDF of the fit is 1.12

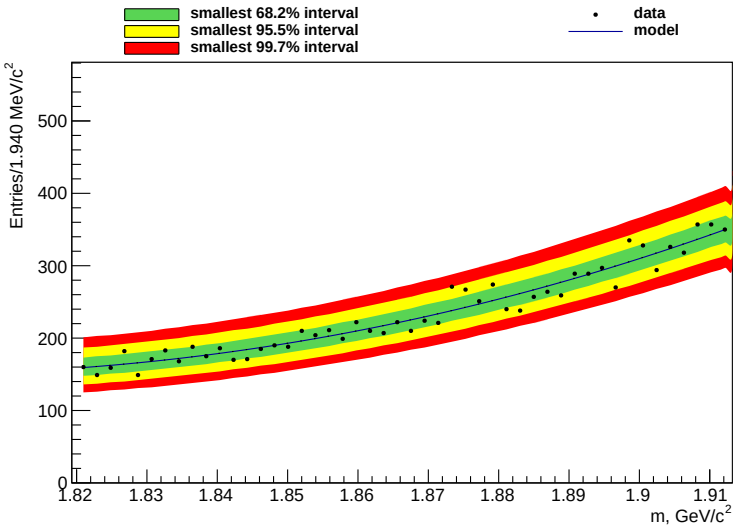


Figure 5.26.: Fit of the background events with the polynomial of degree 3. χ^2/NDF of the fit is 1.00

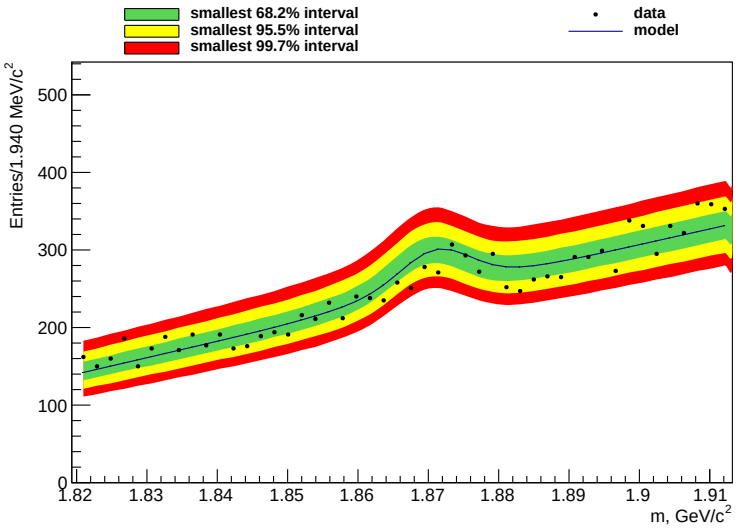


Figure 5.27.: Combined fit of the 530 signal events and background. Signal events are fitted with the sum of 3 Gaussian distributions. Background events are fitted with the polynomial of degree 1. χ^2/NDF of the fit is 1.32

Channel	Yield in data	Yield in generic MC	$\frac{\textit{Yield in data}}{\textit{Yield in generic MC}}$
$D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$	828	984	0.842 ± 0.029
$D^+ \to K^- K^0_S \pi^+ \pi^+$	521'975	667'824	0.782 ± 0.001
$D^+ \to \pi^- K^0_S \pi^+ \pi^+$	3'368'992	4'200'106	0.802 ± 0.001

Table 5.9.: Yield comparison between data and generic MC in the experiment 55

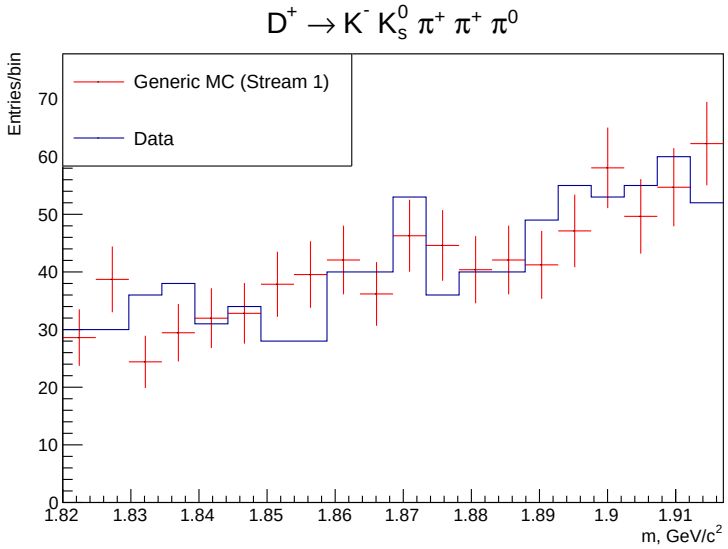


Figure 5.28.: Normalized invariant mass distribution of the experiment 55 for the $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ channel in generic MC and Belle data

5.7. Cross Checks with 10 % Data Sample

I reconstructed approx. 10 % of Belle data sample and compared invariant mass distributions with data from generic MC to cross check my reconstruction algorithm. Table 5.9 lists yields in MC and data. My reconstruction algorithm reconstructs 20 % fewer events in data than in simulation. Similar scaling effect has been already observed by another Belle analysis. Because all channels are scaled similarly, this effect does not influence the measurement.

Figures 5.28, 5.29, and 5.30 compare invariant mass distributions in the data and in the generic MC scaled to the yield in data. Channels $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ and $D^+ \to \pi^- K^0_S \pi^+ \pi^+$ show good agreement between Belle data and generic MC.

Channel $D^+ \to K^- K^0_S \pi^+ \pi^+$ showed large discrepancy between data and MC in the height of the signal peak relative to the background level. This discrepancy comes from higher branching fraction of the channel in generic MC than the branching fraction measured by the FOCUS collaboration [34]. The branching fraction in the generic MC is 6.43×10^{-3}, while measured fraction is $(2.28 \pm 0.18) \times 10^{-3}$. Figure 5.29 shows invari-

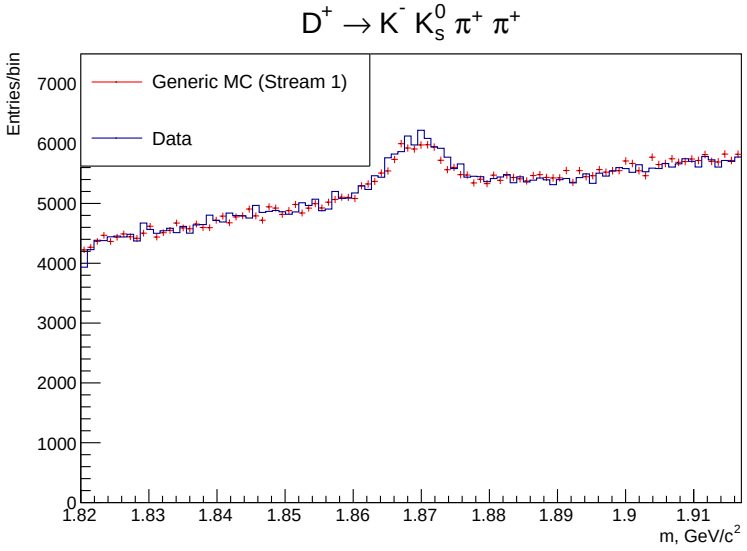


Figure 5.29.: Invariant mass distribution of the experiment 55 in the $D^+ \to K^- K^0_S \pi^+ \pi^+$ channel with the signal component in the generic MC scaled to the measured branching fraction

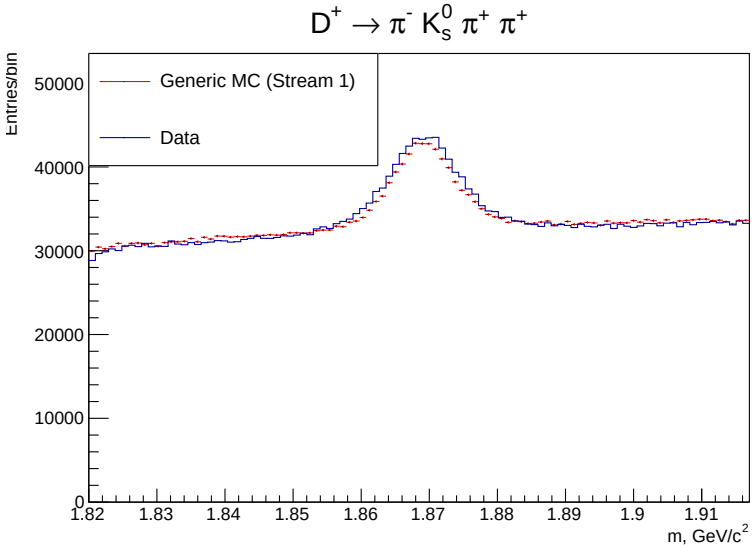


Figure 5.30.: Normalized invariant mass distribution of the experiment 55 in the $D^+ \to \pi^- K^0_S \pi^+ \pi^+$ channel

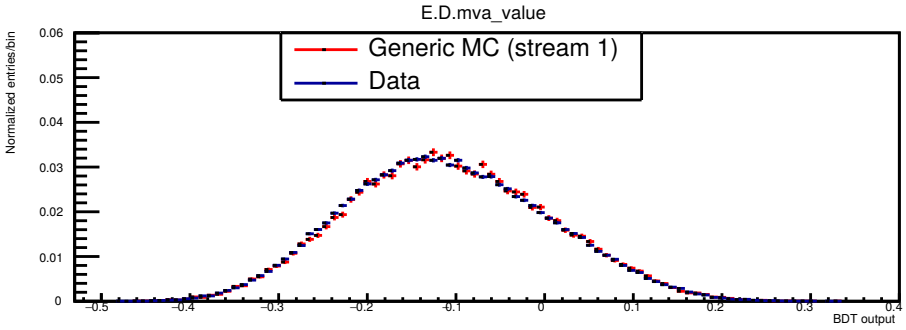


Figure 5.31.: BDT responses in the channel $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$

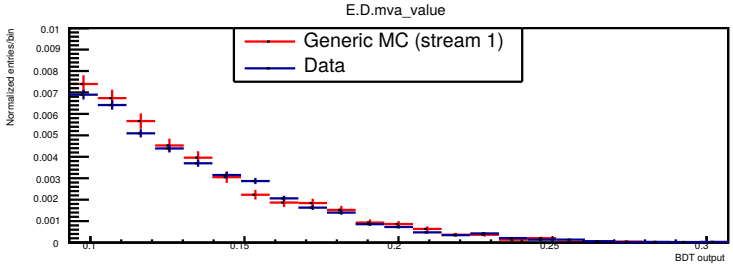


Figure 5.32.: Zoom to the tail of the BDT responses in the channel $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$

ant mass distributions in the channel $D^+ \to K^- K^0_S \pi^+ \pi^+$ already with signal component in generic MC scaled to match measured branching fraction for this channel.

Because BDT uses 31 kinematic variables to calculate its response, its response is sensitive to variations of input parameters. Figures 5.31-5.34 show comparison of the BDT responses for all channels. All channels show good agreement between data and generic MC. This indicates that MC agrees well with real data.

5.8. Systematic Uncertainties

Table 5.10 lists the systematic uncertainties.

Systematic uncertainty of the fit shape is extracted from the marginalized posterior distribution of the fraction parameter. This is possible because I let all parameters float

Source of uncertainty	Value, %
π^0 reconstruction	2.30
BDT cut	1.09
π^0 momentum cut, section 5.4.6	$^{+11.22}_{-6.00}$

Table 5.10.: Sources of the systematic uncertainties

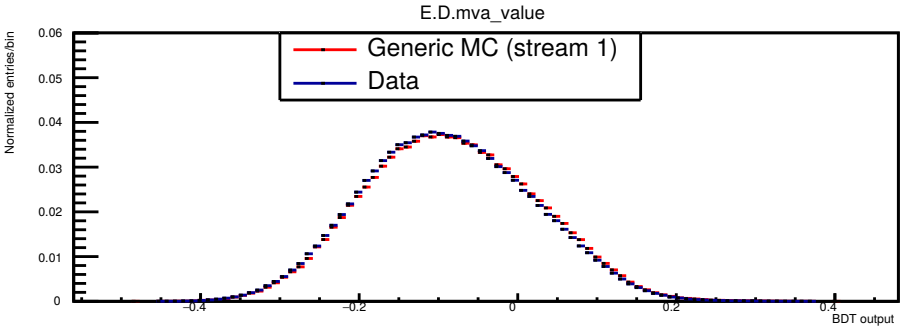


Figure 5.33.: BDT responses in the channel $D^+ \to K^- K^0_S \pi^+ \pi^+$

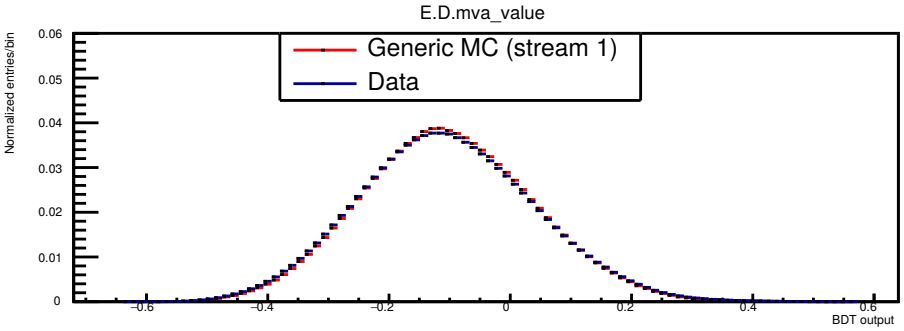


Figure 5.34.: BDT responses in the channel $D^+ \to \pi^- K^0_S \pi^+ \pi^+$

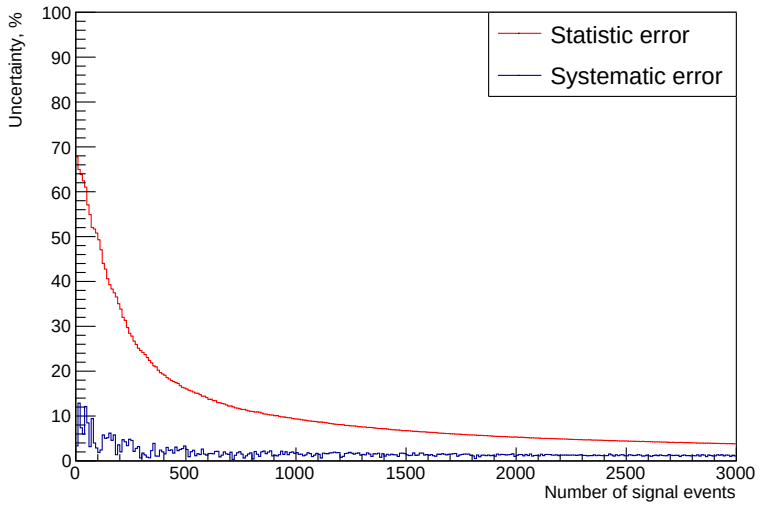


Figure 5.35.: Behaviour of systematic and statistic uncertainties as a function of the signal yield

in the fit. Therefore, their distribution reflects the true variation of the parameters. I separate statistic and systematic contributions to the uncertainty by fitting the invariant mass distribution with only the signal yield parameter floating. The uncertainty of the parameter calculated in this fit corresponds to only statistical uncertainty. The systematic uncertainty is calculated by subtracting the uncertainty of the fit with one free parameter from the uncertainty of the parameter with 14 free parameters in quadrature:

$$\sigma_{systematic} = \sqrt{\sigma^2_{14 parameters} - \sigma^2_{1 parameter}} \tag{5.22}$$

Figure 5.35 shows systematic uncertainty due to the fit shape and statistic uncertainty as functions of the signal yield. Statistic uncertainty decreases with increasing signal yield. Systematic uncertainty shows no dependence on the signal yield once the fitter can clearly recognize signal.

The systematic uncertainty due to π^0 reconstruction is assigned to 2.3 % for π^0 momentum range between 0 and 0.5 GeV/c as determined from the $\tau^- \to \pi^- \pi^0 \nu_\tau$ decay [35]. Because the number of the charged tracks is the same in the main and the normalizing channels, systematic uncertainties due to the tracking performance cancel [36].

Inefficiency in the reconstruction of low-momentum π^0 leads to the spread in the reconstruction efficiency in different sub channels. Conservative estimation of the spread is discussed in the section 5.4.6 and is listed in table 5.10 as "π^0 momentum cut".

The systematic uncertainty due to the BDT cut is estimated from the toy Monte Carlo studies by repeating the cut on a random set of signal events. Figure 5.36 shows cut efficiency distribution. Cut efficiency shows irregular distribution around central value. I calculate systematic uncertainty for cut efficiency as full width at half maximum of this distribution.

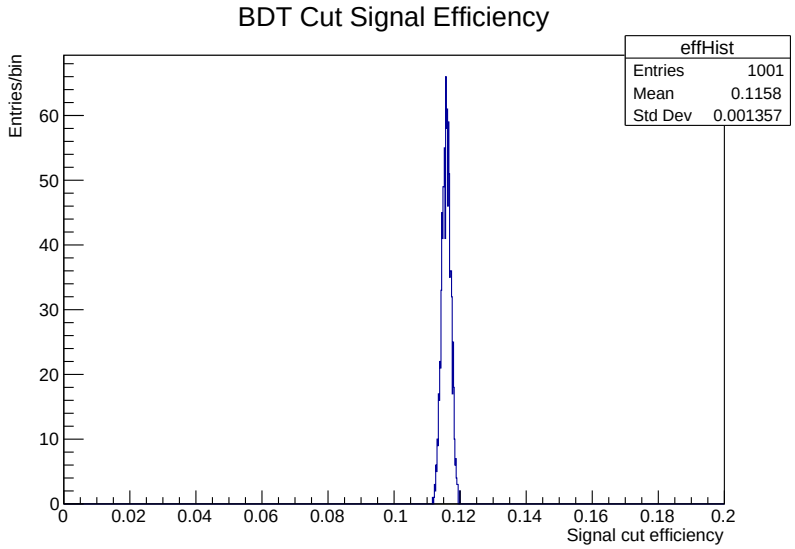


Figure 5.36.: Efficiency distribution due to the BDT cut

5.9. Measurement of Branching Fractions

I reconstructed full Belle data set of 1 ab^{-1} using the techniques outlined in the previous sections. I fit channels $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ and $D^+ \to K^- K^0_S \pi^+ \pi^+$ using unbinned maximum likelihood fit. Due to large data set in channel $D^+ \to \pi^- K^0_S \pi^+ \pi^+$, I fit this channel using binned maximum likelihood fit. Figures 5.37, 5.38a, and 5.38b show the data and the best fit for the signal and two normalization channels. Peaks in the normalizing channels are clearly visible. In the fit to $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$, most data points are within 68.2 % interval around central value of the model. Marginalized posterior distribution of the signal fraction parameter show Gaussian shape and reaches 0 within 99.8 % interval. This indicates low statistical significance of the fit.

Table 5.11 lists the signal yields obtained from the fit to the data. Table 5.12 lists the branching fraction of the signal channel calculated from the signal yields of the normalization channels. As a cross check, I calculated number of $D^{\pm}$ mesons in the data set from signal yields of the normalization channels. Number of $D^{\pm}$ mesons agrees withing statistical uncertainties in both channels.

The fit to the data yields 183 ± 79 signal events in the decay channel $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ with statistical significance 1.9 σ. I use channel $D^+ \to K^- K^0_S \pi^+ \pi^+$ as normalization channel to cancel systematic uncertainties associated with reconstruction of charged tracks. The relative branching fraction of the 5-particle channel is

$$\frac{BF(D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0)}{BF(D^+ \to K^- K^0_S \pi^+ \pi^+)} = 0.033 \pm 0.014. \tag{5.23}$$

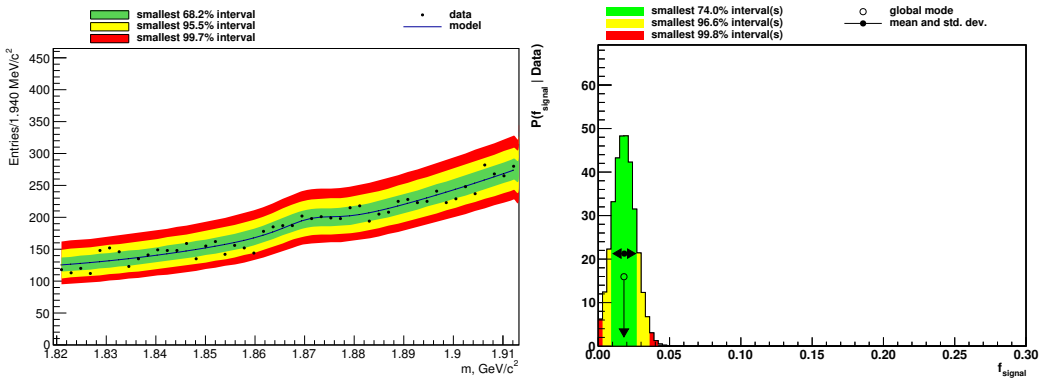
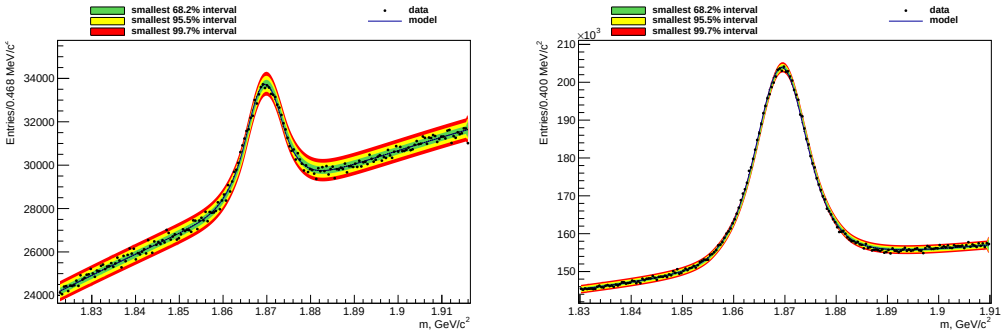


Figure 5.37.: Fit to the data in the $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$channel and the posterior distribution of the signal fraction parameter



(a) Fit to the data in the $D^+ \to K^- K^0_S \pi^+ \pi^+$channel

(b) Fit to the data in the $D^+ \to \pi^- K^0_S \pi^+ \pi^+$channel

Channel	Reconstructed events	Signal fraction	Signal yield	Significance of the fit
$D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$	10'024	0.0183±0.0079	183±79	1.9 σ
$D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ (without veto)	14'682	0.0260±0.0068	382±99	3 σ
$D^+ \to K^- K^0_S \pi^+ \pi^+$	5'929'690	0.0249±0.0002	141'594±1429	
$D^+ \to \pi^- K^0_S \pi^+ \pi^+$	33'181'670	0.0617±0.0002	2'048'504±5606	

Table 5.11.: Signal yields in the reconstructed channels

Normalization channel	Number of $D^\pm$ mesons	Absolute branching fraction of the 5-particle channel	95 % upper CL on the BF
$D^+ \to K^- K^0_S \pi^+ \pi^+$	$(4.78 \pm 0.39) \cdot 10^8$	$(7.44 \pm 3.30) \cdot 10^{-5}$	$1.32 \cdot 10^{-4}$
$D^+ \to \pi^- K^0_S \pi^+ \pi^+$	$(4.79 \pm 0.18) \cdot 10^8$	$(7.45 \pm 3.22) \cdot 10^{-5}$	$1.30 \cdot 10^{-4}$

Table 5.12.: Branching fraction of the $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$channel

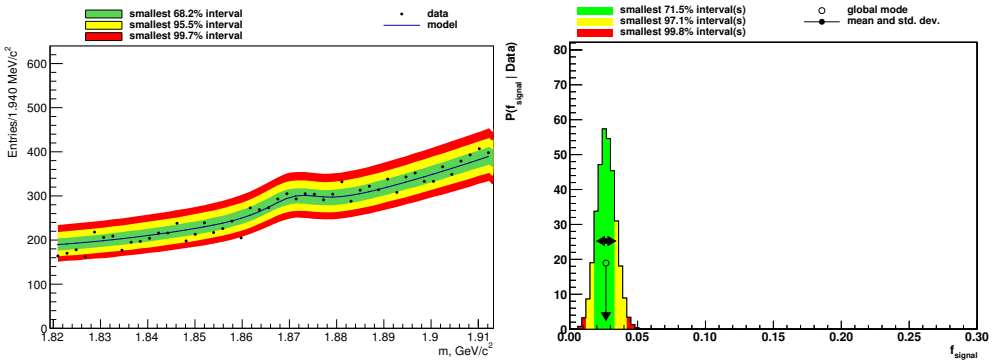


Figure 5.39.: Fit to the data in the $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$channel without $\eta - \omega$ veto and the posterior distribution of the signal fraction parameter

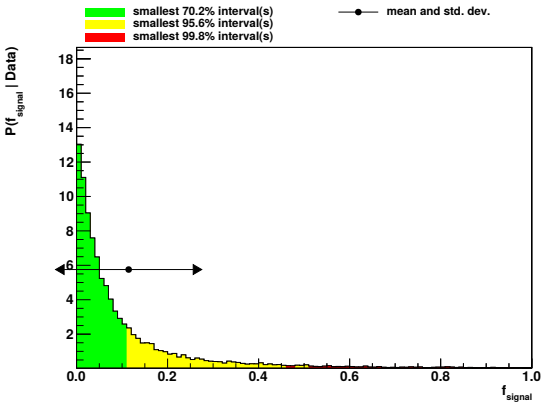


Figure 5.40.: Posterior distribution of the fraction of the 5 particle final state mode in the signal yield

Under assumption that $BF(D^+ \to K^- \pi^+ \pi^+ \pi^+ \pi^- \pi^0) = 0$, this corresponds to absolute branching fraction

$$BF(D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0) = (7.44 \pm 3.30) \cdot 10^{-5}. \tag{5.24}$$

Because statistical significance of the signal channel is below $5\,\sigma$, I give the 95 % upper confidence level for the branching fraction

$$BF(D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0) < 1.32 \cdot 10^{-4}. \tag{5.25}$$

In addition to measuring the branching fraction of the signal channel, I estimated the fraction of the decay $D^+ \to K^- \pi^+ \pi^+ \pi^+ \pi^- \pi^0$ by the relation of the signal yields with and without the $\eta - \omega$ veto as described in the section 5.4.5. Figure 5.39 shows the data and the best fit to the data for the signal channel reconstructed without applying the $\eta - \omega$ veto. The fit to the data calculates 382±99 signal events with the statistical significance

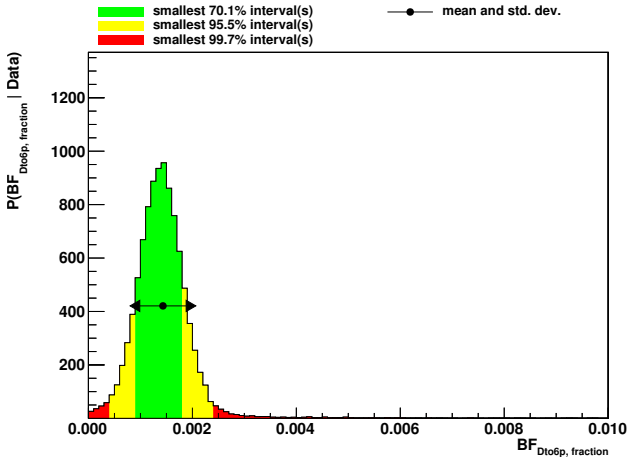


Figure 5.41.: Posterior distributions of the branching fraction of the channel $D^+ \to K^-\pi^+\pi^+\pi^+\pi^-\pi^0$

$3\,\sigma$. I calculated following fraction of the 5-particle channel in the reconstructed signal with the expression 5.13 and the efficiencies listed in the table 5.7

$$f_{D^+ \to K^- K_S^0 \pi^+\pi^+\pi^0} = 0.045 \pm 0.101. \tag{5.26}$$

Figure 5.40 shows posterior distribution of this parameter. Signal fraction for decay $D^+ \to K^- K_S^0 \pi^+\pi^+\pi^0$peaks at 0 which means no significant contribution is measured in this channel.

Figure 5.41 shows marginalized posterior distribution of the branching fractions calculated using the yield ratios normalized to the decay channel $D^+ \to K^- K_S^0 \pi^+\pi^+$. The distribution shows Gaussian shape and reaches 0 within 99.7 % interval. The branching fraction of the $D^+ \to K^-\pi^+\pi^+\pi^+\pi^-\pi^0$ decay mode is

$$\mathrm{BF}(D^+ \to \overline{K}^{*0}\eta\pi, D^+ \to \overline{K}^{*0}\omega\pi) = (1.46 \pm 0.63) \cdot 10^{-3}. \tag{5.27}$$

The calculated branching fraction of the $D^+ \to K^-\pi^+\pi^+\pi^+\pi^-\pi^0$ decay mode agrees with the measurement in [31] within statistical uncertainties. The upper 95 % confidence level for this channel is

$$\mathrm{BF}(D^+ \to \overline{K}^{*0}\eta\pi, D^+ \to \overline{K}^{*0}\omega\pi) < 2.28 \cdot 10^{-3}. \tag{5.28}$$

The 95 % upper confidence level on the branching fraction for the 5-particle channel is

$$\mathrm{BF}(D^+ \to K^- K_S^0 \pi^+\pi^+\pi^0) < 7.35 \cdot 10^{-5}. \tag{5.29}$$

5.10. Improvement of the Measurements

There are several ways to improve the measurements. The size of the Belle data sample is large enough to measure the decay $D^+ \to K^-\pi^+\pi^+\pi^+\pi^-\pi^0$ with sufficient statistical significance if the branching fraction of the decay is in the order of 10^{-3}. The measurement

requires optimization of the selection criteria for this particular channel. For example, I can select only pions which are not part of the K^0_S candidate.

The sensitivity of the measurement for the decay $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ is limited by the size of data sample of the Belle experiment. Therefore, an increase of the data sample by factor 50 in Belle II experiment combined with improvements in the detector design have the potential for improving sensitivity to the decay by factor 10. Next chapters describe the upgrade of the detector and, specifically, the design of the read-out system for the pixel detector of the Belle II experiment.

At this point I switch topic of the thesis from data analysis to design of the read-out system for the pixel detector of the Belle II experiment. This part describes Belle II experiment, pixel detector, and focuses on description of the data read-out system.

* * *

This part gives full description of the read-out system of the pixel detector for the Belle II experiment. This includes work done in collaboration with my colleagues. I would like to outline their contributions to the system here.

Igor Konorov leads the project. He designed most of the system's hardware, clustering algorithm, data processing core with overlapping trigger support, and subevent builder.

I worked with Dominic Gaisbauer to design UCF protocol, used in the intra-system communication. I implemented synchronization of clock phases and Dominic designed logical layer of the protocol UCF.

I worked with Yunpeng Bai to perform radiation hardness certification of the optical transmitters. Yunpeng designed the ATCA carrier board.

Stefan Huber contributed to the development of the data processing core with the overlapping trigger functionality. He designed the DHI board and commissioned the system for phase 2 of the Belle II experiment.

I worked with Martin Gottwald to test correctness of the cluster algorithm in the FPGA.

Chapter 6.

Belle II Experiment

The Belle experiment stopped its operation in 2010 to undergo an upgrade with the goal to increase data set by factor 50. The new experiment is called Belle II. All detectors of the Belle experiment are upgraded and a new pixel subdetector is added to improve impact parameter resolution of the detector. To reach the goal luminosity, the KEKB accelerator is also upgraded. The peak luminosity of the upgraded accelerator, SuperKEKB, will increase by factor 40.

Figure 6.1 shows the layout of the Belle II experiment. The detector is built around the interaction point of the asymmetric electron-positron collider SuperKEKB. Belle II uses the same coordinate system as Belle. The barrel region of Belle II is covered by a two-layer DEPFET pixel detector, a four-layer silicon strip detector, a central drift chamber, time-of-propagation counters, an electro-magnetic calorimeter, and a K_L^0 and muon detector. The end-caps of Belle II is covered by an aerogel ring imaging Cherenkov detector, an electromagnetic calorimeter and a K_L^0 and muon detector.

6.1. SuperKEKB Collider

The SuperKEKB collider is an upgrade of the electron-positron collider KEKB [37]. The target goal of the SuperKEKB collider is achieving peak luminosity of $8 \times 10^{35} cm^{-2} s^{-1}$, which is 40 times higher than peak luminosity of the KEKB. Table 6.1 compares operating parameters of the KEKB and SuperKEKB accelerators.

The luminosity in an electron-positron collider is expressed as

$$L = \frac{\gamma_{\pm}}{2er_e}\left(1+\frac{\sigma_y^*}{\sigma_x^*}\right)\left(\frac{I_{\pm}\xi_{y\pm}}{\beta_y^*}\right)\left(\frac{R_L}{R_{\xi_{y\pm}}}\right) \tag{6.1}$$

where γ is the Lorentz factor, r_e is the electron radius, $\sigma_{x/y}^*$ is the beam size at the interaction point, I is the beam current, β_y^* is the vertical beta function at the interaction point, $\xi_{y\pm}$ is the vertical beam-beam parameter, $R_{L/\xi}$ are the reduction factors [37]. Compared with the KEKB, the SuperKEKB achieves factor 20 increase in the luminosity by squeezing beams in vertical direction to reduce vertical beta function β_y^* and by increasing the crossing angle, which causes interaction cross-section of the beams to decrease. This technique is called the nano beam scheme [39]. SuperKEKB uses superconducting focus quadrupole magnets to shape the beams at the interaction region [40]. Another factor 2 is achieved by increasing beam currents.

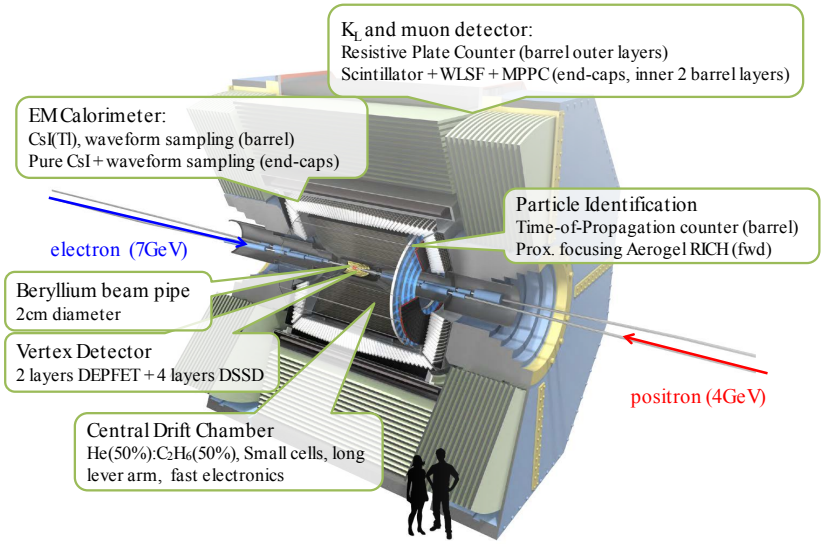


Figure 6.1.: Layout of the Belle II experiment

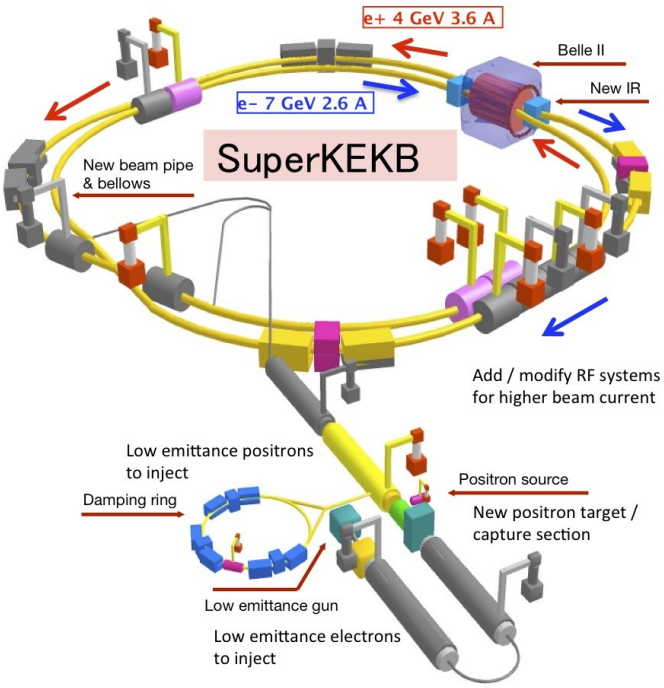


Figure 6.2.: Layout of the SuperKEKB accelerator

Parameter	KEKB [38]		SuperKEKB [37]	
	LER	HER	LER	HER
Circumference, m	3016			
Luminosity, $cm^{-2}s^{-1}$	2.1×10^{34}		8×10^{35}	
Number of bunches	1585		2500	
E, GeV	3.5	8	4	7
I_{beam}, A	1.6	1.1	3.6	2.6
Crossing angle, mrad	22		83	
Horizontal betatron function at the IP β_x^*, cm	150	150	3.20	2.50
Vertical betatron function at the IP β_y^*, mm	5.9	5.9	0.27	0.30
Horizontal beam-beam parameter ξ_x	0.120	0.99	0.0028	0.0012
Vertical beam-beam parameter ξ_y	0.120	0.89	0.0881	0.0807
Horizontal emittance ε_x, nm	18	24	2.3	4.6
Vertical emittance ε_y, pm			8.64	11.5

Table 6.1.: Comparison of the parameters of the KEKB and SuperKEKB colliders

Figure 6.2 shows the layout of the collider. Electron bunches are produced in the pre-injector by the RF [1] gun. Bunches, which are used for the positron production, are accelerated to 4 GeV towards the tungsten conversion target. Positrons are injected into the damping ring, which reduces emittance by radiation damping. The low emittance positron bunches are then accelerated to 4 GeV in the linear accelerator and injected into the low-energy ring of the SuperKEKB. The electron bunches bypass the tungsten target. They are accelerated to 7 GeV and are injected into the high-energy ring of the SuperKEKB.

The short beam lifetime of approximately 600 s at the designed beam current is compensated by the continuous injection scheme. In this scheme, electron and positron bunches are alternately injected into the accelerator's rings with frequency 25 Hz for each ring. This challenges detector design, because detectors should be able to cope with increased beam background levels when freshly injected bunches cross the interaction point.

Lower energy asymmetry between electron and positron bunches than in the KEKB collider results in a lower boost factor $\beta\gamma$ of the collision products. This affects primary decay vertex separation of produced B–$\overline{B}$ meson pairs in the laboratory frame. Therefore, vertex location resolution must be improved in Belle II.

6.2. Central Drift Chamber

The Belle II central drift chamber reconstructs charged tracks and measures their momenta, provides particle identification, and trigger signal for charged tracks [41].

The central drift chamber consists of 56 wire layers. Figure 6.3 shows the layout of the Belle II central drift chamber compared with the Belle configuration. Eight innermost layers build the inner superlayer with the smaller radial cell size of 10 mm. Smaller cell

[1] Radio frequency

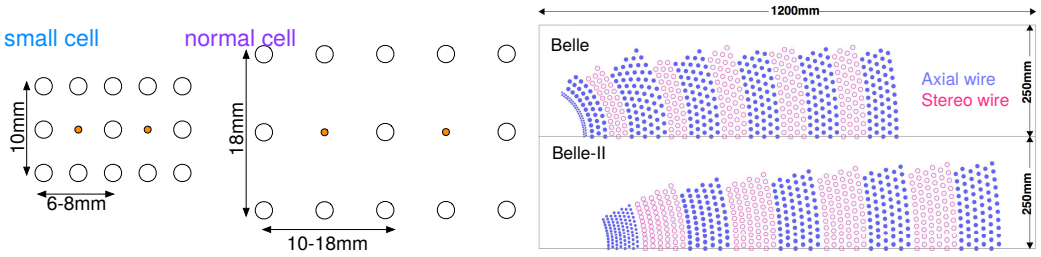


Figure 6.3.: Layout of the Belle II CDC compared with the Belle configuration [42]. Large white circles represent aluminum wires, smaller filled circles represent gold-plated tungsten wires

size reduces cell occupancy, which is higher in inner layers due to proximity to the interaction point. Remaining layers build 8 superlayers with 6 layers per superlayer and the larger radial cell size of 18 mm.

The superlayers are divided into the axial and the stereo superlayers. Axial superlayers are arranged parallel to the z-axis of the experiment. Stereo layers are rotated in respect to the z-axis to provide better resolution in z direction.

6.3. Aerogel Ring Imaging Cherenkov Detector

The aerogel ring imaging Cherenkov detector, the ARICH, is a part of the particle identification system in Belle II. The detector measures Cherenkov light produced by charged particles in the silica aerogel radiator [43]. Figure 6.4 shows the principle of particle identification in the detector. Cerenkov light propagates in a cone with opening angle Θ_C, which is proportional to mass of the particle m, its momentum p, and refractive index of the aerogel n

$$m = \frac{p}{c}\sqrt{n^2 cos^2\Theta_C - 1}. \tag{6.2}$$

Opening angle is obtained by reconstructing an ellipse built from the interception of the light cone and the sensitive plane of the detector. By measuring particle's momentum and opening angle, it is possible to identify particle's species.

The ARICH detector is installed in the forward end-cap of the Belle II detector outside of the central drift chamber. The detector consists of 124 pairs of aerogel tiles and 420 144-channel hybrid avalanche photo detectors.

A double layer technique first used in the aerogel Cherenkov counter of the Belle experiment is also used in the ARICH detector to increase the light yield [45]. This effect is achieved by combining two radiator layers with different refractive indices. Figure 6.5 shows difference in light propagation between the single- and double-layer detectors. Adjusting the refractive index of the first radiator slightly higher than that of the second radiator causes the light cones from the double-layer radiators to focus on the detector's sensitive area. This technique increases photon yield in the aerogel radiator by 40 % and keeps the thickness of the radiator constant [45].

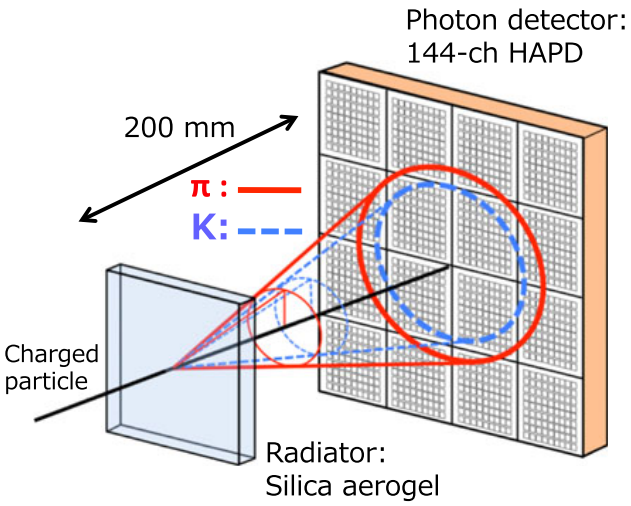


Figure 6.4.: Principles of the particle identification in the ARICH detector [44]

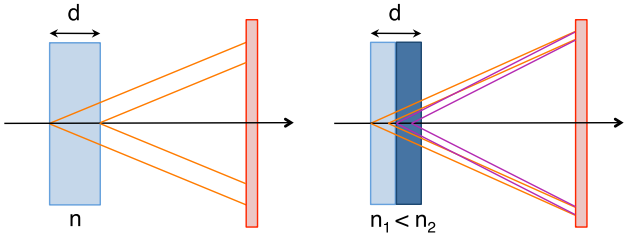


Figure 6.5.: Focusing of the Cerenkov light in the single- and double-layer radiators [44]

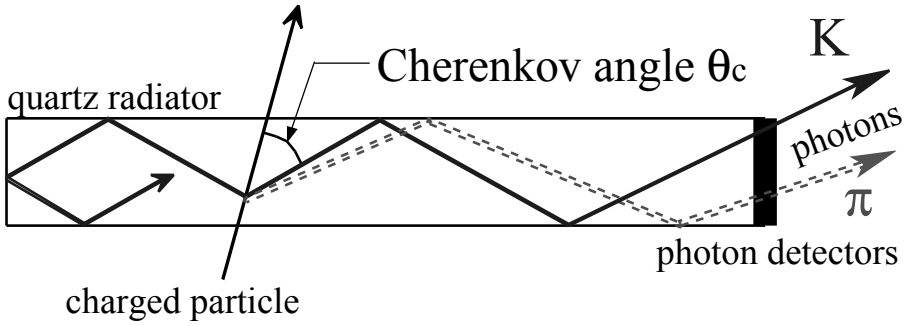


Figure 6.6.: Light propagation in the TOP detector [41]

6.4. Time-of-Propagation Counter

The time-of-propagation counter, the TOP, performs particle identification in the barrel region of the Belle II detector. Particle identification is based on measuring propagation time of photons in the quartz radiator. Propagation time is proportional to the Cherenkov angle, which depends on momentum and mass of a particle, and the position of the particle. Momentum and position of a particle are measured by the central drift chamber. In addition, precise knowledge about the time of the interaction is required. This information is distributed by the global trigger decision logic of the Belle II experiment. Typical propagation time of the Cherenkov light in the radiator is 100 ps. Therefore, the time resolution of 50 ps is required to distinguish between pions and kaons with multi-GeV momenta [46].

Figure 6.6 shows light propagation in the TOP detector. Particles with the same momenta but different mass produce light cones with different Cherenkov angles Θ_C. The walls of the radiator reflect photons until they reach the photosensor located at the end of the radiator. The other end of the radiator contains a focusing mirror, which reflects the signal back towards the two-dimensional photosensor. The read-out electronics analyzes time evolution of the signal on the photosensor and estimates probability of a particle to be of a specific type.

6.5. Electromagnetic Calorimeter

The electromagnetic calorimeter, the ECL, detects particles, which interact electromagnetically, and measures their energy. With fast detector response, the electromagnetic calorimeter is an important input to the global trigger decision logic. The detector is also used in the electron identification, for luminosity measurements, and for K^0_L detection.

The electromagnetic calorimeter in Belle II reuses crystals from the electromagnetic calorimeter of the Belle experiment. The crystals cover polar angle region between 12.4° and 155.1° with two 1° gaps between the end caps and the barrel regions [47].

The main difference to the Belle experiment is an upgrade of the analog, digitizing, and DAQ electronics to improve energy and time resolution of the detector. New analog

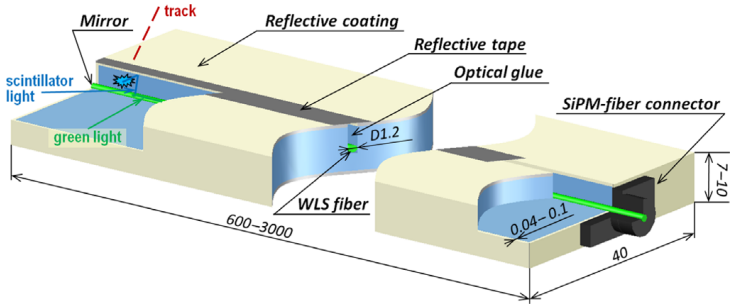


Figure 6.7.: Layout of the scintillator strip [49]

electronics is needed because Tl doped CsI crystals have long scintillation decay time that causes pile-up noise. With increased beam background, this effect would lead to long dead times and decrease of detection efficiency. New electronics reduces shaping time of analog signals and employs digital signal processing to recover pile-up signals.

6.6. K_L^0 and Muon Detector

The outermost detector in Belle II is the K_L^0 and muon detector. The detector covers polar angle from 20° to 155°. The detector is integrated with the magnetic flux return of the solenoid. The sensitive detector layers are installed in the gaps between iron plates of the magnetic flux return. The iron plates of the magnetic flux return provide enough material, in which K_L^0 particles can shower hadronically [41].

The detector consists of two subdetectors: the barrel and two end cap detectors. The outer layers of the barrel subdetector reuse the existing resistive plate chambers from the Belle experiment [48]. The detector module contains perpendicular sensitive strips that measure a 2D position of the track. The disadvantage of resistive plates is long dead time during recovery after a discharge. Therefore, resistive plate chambers can only be used in the well shielded environment of the outer barrel region.

In the end cap regions and the inner layers of the barrel region, the expected background rate in SuperKEKB would reduce efficiency of the resistive plate chambers to below 50 % [41]. These layers are built with organic scintillator with faster recovery time [49]. Figure 6.7 shows the layout of the scintillator strip. A strip contains a wavelength-shifting fiber that collects scintillation light produced by particles in scintillating material. One side of the wavelength-shifting fiber is coated by the reflective layer, the opposite side is connected to the silicon photomultiplier. The silicon photomultiplier, which operates in the Geiger mode, detects scintillation light collected by the fiber. The strips in the neighboring layers are installed perpendicular to each other to measure 2D coordinate of the track.

Chapter 7.

Vertex Detector

The innermost detector of the experiment is the vertex detector. The detector consists of two independent sub-detectors: the four-layer double-sided silicon strip detector and the two-layer pixel detector.

The detector's goals and constraints emerge from its position in Belle II. As the closest detector to the interaction point of the SuperKEKB accelerator, the vertex detector is ideally suited for measuring precise position of the decay vertices. Additionally, data from the silicon strip detector is used in tracking and in the online data reduction for the pixel detector. Energy loss measurements in the pixel detector also make it possible to detect and identify charged low-momentum particles, which do not reach other detectors.

In order to improve track resolution in the experiment, detectors should minimize secondary scattering. This is done by limiting material budget of the sensors and the front-end electronics. Additional layers of the vertex detector will improve vertex parameter resolution by factor 2 to 15 μm for high momentum particles as shown in figure 7.1.

Detector	Layer number	Ladders	Sensors per Ladder	Radius [mm]
PXD	1	8	2	14
PXD	2	12	2	22
SVD	3	7	2	39
SVD	4	10	3	80
SVD	5	14	4	104
SVD	6	17	5	135

Table 7.1.: Layout of the vertex detector

Table 7.1 summarizes the layout of the vertex detector. The vertex detector layers are arranged cylindrically around the interaction point, as shown in figure 7.2 with layers numbered in radial direction.

Two inner layers consist of 20 pixel detector ladders. A ladder consists of two sensors, the half ladders, which are aligned along the z-axis. Four outer layers consist of 7, 10, 14, 17 ladders. The ladders in layer 3 are built of 2 sensors, in layer 4 of 3 sensors, in layer 5 of 4 sensors, and in layer 6 of 5 sensors.

Proximity to the interaction point also results in higher radiation background. Simulations of the background in Belle II show that the PXD is expected to absorb a dose of 15-18 Mrad and the SVD a dose of 4.5 Mrad during the lifetime of Belle II [52]. Ionizing radiation causes lattice defects in the semiconductor detectors, which change operation

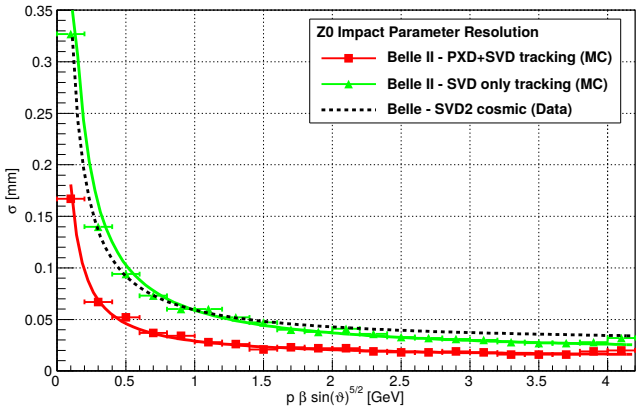


Figure 7.1.: Monte-Carlo simulation of the expected improvement of the impact parameter resolution in Belle II compared to the Belle experiment [50]

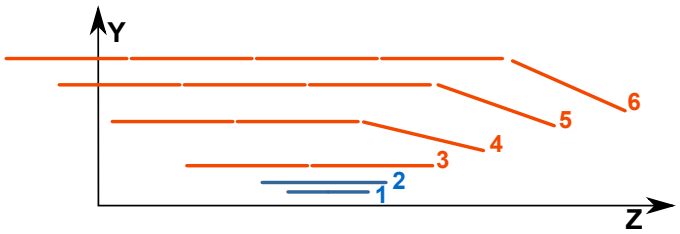


Figure 7.2.: Layer numbering of the vertex detector [51]

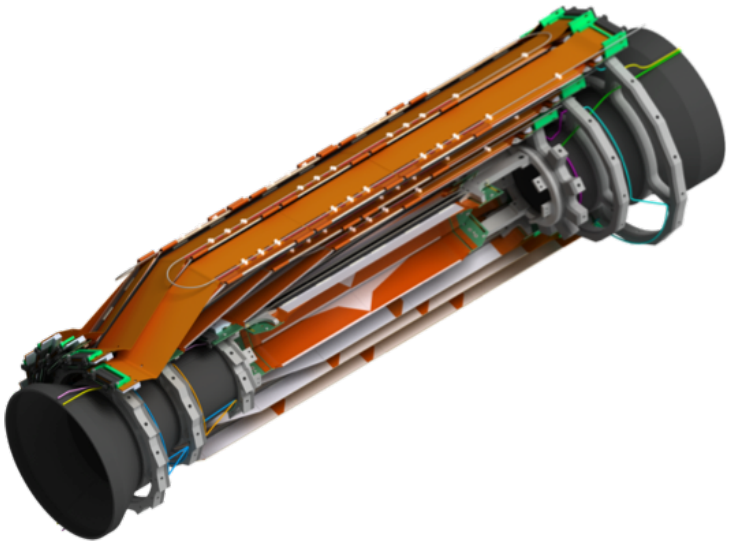


Figure 7.3.: CAD drawing of the silicon strip detector [53]

Form factor	Nr. of strips p/n side	Strip pitch, μm p/n side	Active area, mm^2
Small	768/768	50/160	4716
Large	768/512	75/240	7073
Trapezoidal	768/512	50-75/240	5890

Table 7.2.: Specifications of the sensors used in the silicon strip detector

potentials in semiconductor and increase leakage currents. This requires adjustments of the operating parameters of the detector and reduces efficiency of the particle detection. The front-end electronics must adapt to the new operation conditions and be radiation hard by design. High radiation background will result in high occupancy, which requires faster read out times and better data reduction algorithms.

7.1. Silicon Strip Detector

The read-out module of the silicon strip detector is a ladder. A ladder consists of double-sided silicon sensors and a flexible fan-out circuit. The ladders are installed in four layers and cover the polar angle from 17° to 150°. Figure 7.3 shows the CAD drawing of the fully assembled detector.

The sensors are produced in three form factors: small, large and trapezoidal. Table 7.2 lists dimensions and parameters of the sensors.

The sensor consists of p- and n-doped strips implanted orthogonally on both sides of the bulk silicon. The n-type strips are surrounded by the p-type strips for isolation. The

aluminum electrodes are installed on top of the strips carry signals to the front-end electronics, which digitizes the signals.

Bulk silicon is fully depleted by reverse biasing with high voltage. A charged particle crossing the depleted region excites electron-hole pairs. Holes drift to the p-side and electrons drift to the n-side. Drift of charged particles induces electric signal in both type of strips. Measuring these signals allows us to determine position of the track in 2 dimensions by finding coinciding strips. Additionally, if charge is shared by several strips of the same type, the resolution can be further improved by using center-of-mass algorithm.

Silicon sensors are connected to the flexible PCBs, the origami, made from 3 layers of Polyamide. A 1 mm thick layer of Airex material, which provides electrical and thermal isolation, separates the sensor and the origami [54]. Two pitch adapters connect origami with both sensor sides. The origami houses APV25 shaper chips, which are thinned down to 100 μm to reduce material budget [55]. Because APV25 chips have high thermal dissipation in order of 350 mW, they are actively cooled by CO_2 cooling system. The temperature is measured by fiber optical sensors installed in the Airex layer [56]. The material budget of the fully assembled detector in the sensitive region is, on average, 0.6 % of radiation length X_0 per layer [52].

7.2. Pixel Detector

The sensor technology used in the pixel detector is the depleted p-channel field effect transistor, DEPFET. DEPFET is an active pixel sensor technology, proposed by J. Kemmer and G. Lutz in 1987 [57]. Pixel structure of the DEPFET combines a sensor, an amplifier and an analog memory element. The structure allows us to build very thin detectors with low power consumption and low heat dissipation. The technology is also radiation hard by design and has an excellent signal-to-noise ratio. This makes the technology suitable for constructing vertex detectors with high granularity and small spacial resolution.

7.2.1. DEPFET Technology

Figure 7.4 shows schematic layout of the DEPFET pixel. A MOSFET transistor is integrated on top of the silicon substrate. The silicon substrate is depleted by applying a high (approx. -80 V) depletion voltage to a p-contact on the opposite side to the MOSFET. The potential minimum in the depleted substrate is shifted to the region 1 µm below the gate contact of the MOSFET by an additional phosphorus implantation. This region is called the internal gate. The n^+ doped region close to the internal gate called the clear contact. The clear contact is separated from the internal gate by a polysilicon layer that acts as a potential barrier. An additional clear gate operated at a constant voltage lowers this potential. Applying positive pulse to the clear contact, changes potential barrier, so that thermal energy of the electrons trapped in the internal gate is sufficient to overcome this barrier.

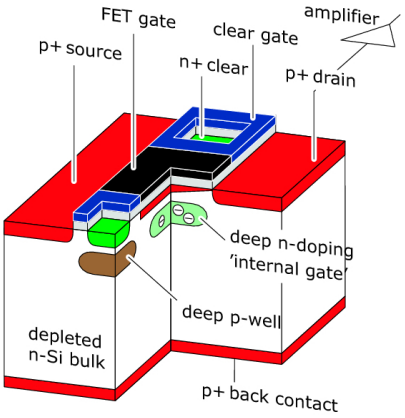


Figure 7.4.: Schematic layout of the DEPFET pixel by P. Lechner

Typical operation of the DEPFET pixel is described below. The incident particle produces electron-hole pairs in the depleted silicon. Holes drift to the p-contact on the backside and are removed from the detector. Electrons drift to the potential minimum in the internal gate, where they are stored. Once the MOSFET gate is activated, current flows from the source to the drain contacts. Charge, collected in the internal gate, modulates current through the MOSFET. Modulation is described by the charge amplification coefficient:

$$g_q = \frac{\Delta I}{\Delta Q} \tag{7.1}$$

where I is current flowing through the transistor, and Q is charge accumulated in the internal gate. The charge amplification coefficient depends, in general, on material properties and geometry of the transistor.

Signal is measured as source-drain current. Measurement does not destroy charge in the internal gate. Once the measurement is finished, the internal gate must be cleared by applying positive voltage to the clear gate.

7.2.2. Noise

The measurement precision depends on equivalent noise charge. The equivalent noise charge of the DEPFET is expressed as:

$$ENC = \sqrt{\alpha \frac{2kT}{g_q} C_{tot}^2 A_1 \frac{1}{\tau} + 2\pi\alpha_f C_{tot}^2 A_2 + qI_L A_3 \tau} \tag{7.2}$$

where A_1, A_2, A_3 are the production-specific parameters [58]. The first term in the equation 7.2 describes thermal noise. Thermal noise is proportional to the capacitance, C_{tot},

and inversely proportional to the integration time, τ, and charge amplification, g_q. The second term describes the 1/f noise and is proportional to the capacitance, C_{tot}. The last term describes the shot noise caused by leakage current, I_L.

The equation 7.2 allows us to estimate intrinsic noise of the detector. Charge amplification of the DEPFET in the Belle II pixel detector is 400 pA/e$^-$; integration time in Belle II is in order of 100 ns [58]. Capacitance is given through the capacitance of the internal gate, which can be made very small. Leakage current is very small for the new detectors, but increases with irradiation. For a 50 μm-thick sensors, measured noise is 50 e$^-$. This gives signal-to-noise ratio of 80 for the expected signal level of 4000 e$^-$ [58].

7.2.3. Intrinsic Electronic Shutter

An important feature of the DEPFET technology is intrinsic electronic shutter, the gated mode. DEPFET, operated in the gated mode, protects accumulated electrons, collected in the internal gate, and removes new electrons, generated while detector is in the gated mode. This feature will be used to make the detector immune to injection noise caused by freshly injected beam bunches. Injected bunches cool down by emitting synchrotron radiation. This process takes approx. 4 ms out of 20 ms injection cycle. During this time, they generate signal in the detector, which leads to increase in the detector occupancy. This effect would lower efficiency of track reconstruction by 20 % and cause data loss. Gated mode will blind the detector only during transitions of the noisy bunches through the interaction point and restore detector operation after noisy bunches pass the detector. Time, required for switching into the gated mode, is 1 ms.

Figure 7.5 shows working principle of the intrinsic electronic shutter. Gate potential is lowered when the pixel is activated during the normal operation of the DEPFET. If the clear potential is raised, then thermal energy of the electrons is sufficient to overcome the potential barrier between the clear and internal gate. This operation removes electrons from the internal gate and effectively clears the pixel.

In the gated mode, the gate remains turned off, which leaves the internal gate at higher potential. Additionally, the clear potential is raised, so that electrons cannot overcome potential barrier anymore. This traps accumulated electrons in the internal gate. At the same time, the clear potential becomes the most attractive potential in the pixel. Therefore, all newly produced electrons do not drift to the internal gate but to the clear gate and are immediately removed from the pixel.

7.2.4. Matrix Design

A pixel detector unit in Belle II is called the half ladder. Figure 7.6 shows the layout of the half ladder. A half ladder consists of a DEPFET matrix with the thickness 50 μm and a support frame with the thickness 400 μm, which carries steering chips.

The half ladders are produced using the silicon-on-insulator technology. Figure 7.7 shows production steps of the detector. In the first step, the sensor wafer is implanted with doping agents and oxidized. In the second step, the sensor wafer is bonded with the handling wafer and ground to the sensor thickness of 50 µm. In the third step, DEPFET

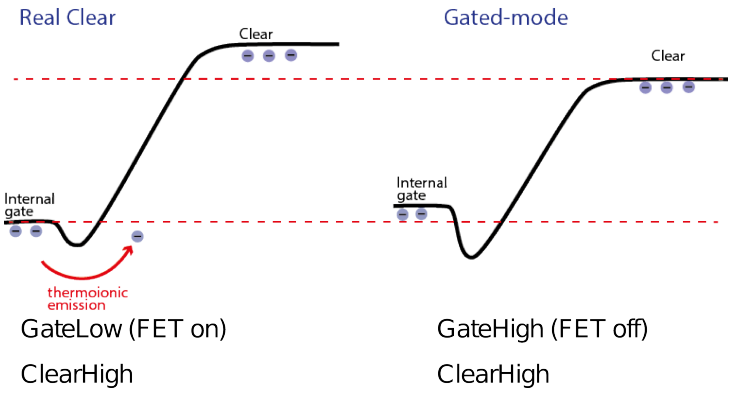


Figure 7.5.: Potentials in a DEPFET pixel during normal operation (left) and during the gated mode (right)

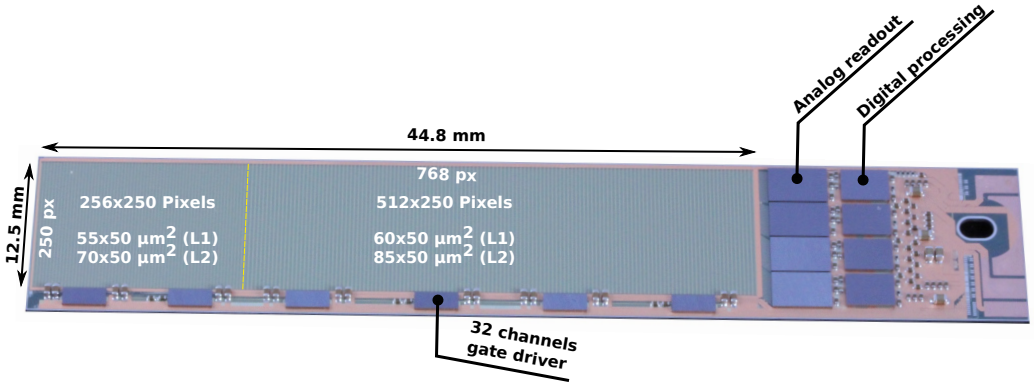


Figure 7.6.: Layout of a half ladder

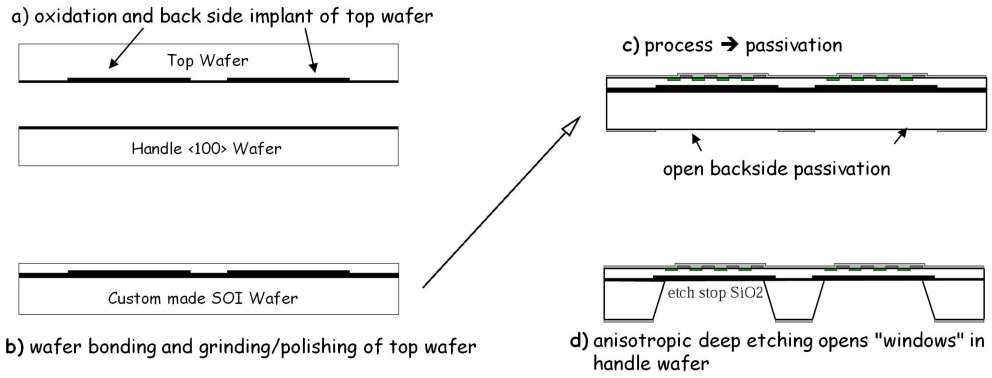


Figure 7.7.: Half ladder thinning steps [58]

Layer	Inner region, μm^2	Outer region, μm^2	Active area, mm^2
1	50 x 55	50 x 60	12.5 x 44.8
2	50 x 70	50 x 85	12.5 x 61.44

Table 7.3.: Pixel dimensions and active detector area

structures are produced in the sensor wafer and wafers are coated with passivation layer, except for the area below the DEPFET structures on the handling wafer. In the fourth step, silicon below the sensor is removed using deep anisotropic etching. The SOI technology reduces material budget of the detector and provides a monolithic support frame for the front-end ASICs[1] outside of the sensitive area. This brings combined material budget of a half ladder to 0.11 % X_0.

A half ladder in Belle II contains a 768x250 matrix of DEPFET pixels. The matrix is subdivided in two regions along z axis: the inner region with small pixels and the outer region with large pixels. The inner region contains 256x250 pixels; the outer region contains 512x250 pixels. Table 7.3 lists pixel dimensions in the detector. This division is necessary to preserve nearly constant angular resolution.

DEPFET matrix is operated in a rolling-shutter mode: only four rows of the pixels are active at any given moment. Every pixel in these four rows is activated, measured and cleared simultaneously. This is done by connecting all pixels in the four rows to the same control lines, gate and clear, as shown in figure 7.8. This concept reduces number of control lines and speeds-up matrix read-out by factor 4 at the cost of increasing the number of ADC channels.

Rolling shutter mode reduces power consumption of the matrix because only four activated rows consume power. Low power consumption results in an estimated thermal dissipation of 0.5 W per half ladder [59].

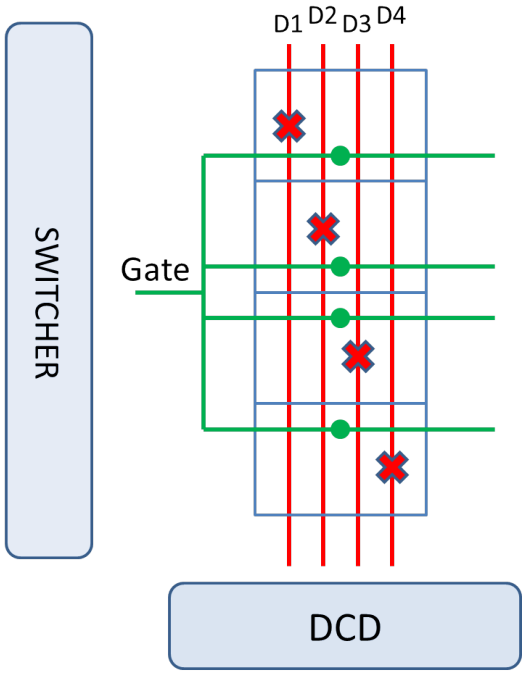


Figure 7.8.: Four-fold read-out implementation in the pixel detector [58]

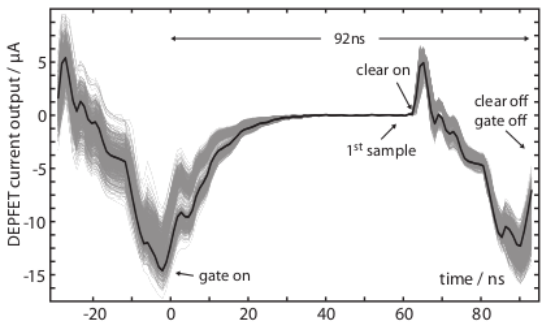


Figure 7.9.: Typical drain current cycle of the DEPFET pixel [60]

7.2.5. Read-Out Cycle

Figure 7.9 shows typical read-out cycle of the DEPFET channel. The cycle starts with activation of the pixel by lowering gate potential. Current on the drain line rises until it reaches stable plateau. Rise time of the current is limited by the capacitance of the drain line. Stable current at this plateau is measured by the ADC [2]. After digitization finishes, internal gate is cleared by activation of the clear potential. Front end electronics then activates next row, and the cycle repeats with another pixel that is connected to the same drain line.

Minimum period of the cycle is limited by sum of rise and settle times of the drain current, time required to receive current in the ADC, and clear time. The cycle period used in Belle II is 104.8 ns. With 192 electrical rows this defines total integration time of the detector of 20.12 µs.

7.2.6. Front-end Electronics

The front-end ASICs consist of six high-voltage gate drivers, the switchers, four multi-channel ADCs, the drain current digitizers, and four data handling processors, the DHPTs. The switchers, which control matrix operation, are operated by a DHPT. The drain current digitizers receive drain current from active pixels, digitize current, and send samples to DHPTs. The DHPTs synchronize switchers with ADCs, read digital samples, perform data reduction and send data to the read-out electronics.

SwitcherB

The switcherB ASICs, switcher for Belle II, drive high voltage gate and clear lines of the DEPFET pixels [61]. Each switcherB operates 32 electrical rows or 128 geometrical rows. Therefore, six ASICs are required for steering full matrix on a half ladder.

Switchers contain a low-voltage control block, high-voltage channels, two voltage regulators and a configuration block. The low-voltage control block is built as a 32-bit shift register and is controlled by four external signals: a clock, an input to the shift register, SERIN, gate and clear strobes. Switcher operate with clock frequency of 9.54 MHz. DHPT synchronizes switcher operation by shifting a pulse into the shift register of the first switcher. The pulse propagates through the shift register with switcher clock and activates the corresponding row of the matrix. This effectively implements rolling shutter. The gate and clear strobes, also generated by DHPT, define timing for activation of the corresponding DEPFET control line in the currently active matrix row. The output of the shift register, the SEROUT, of one switcher is connected to the SERIN of the next switcher. This propagates the rolling shutter sequence to the next switcher.

[1] Application-specific integrated circuit

[2] Analog-to-digital converter

Drain Current Digitizer

Drain current of the DEPFET pixels is digitized by the 256-channel ADC ASIC, the drain current digitizer for Belle II. DCD is produced in UMC 180 nm CMOS technology using radiation hard design techniques [62]. Resolution of the ADC is 8 bit with sampling rate 9.54 MHz. Dynamic range of the ADC is between 20 µA in the fine mode and 40 µA in the coarse mode. Power consumption of the ASIC in the operation mode is below 2 W.

The DCD is built out of an analog block, which consists of 256 analog channels, and a digital read-out and control block. Main elements of the analog channel are a two-bit offset DAC [3], a resistive current receiver, an analog common mode correction circuit, a calibration current injection circuit, and a pipeline ADC.

The offset DAC is used for compensation of pedestal dispersion caused by radiation damage to the DEPFET structure. The DAC consists of three adjustable current sources that inject current into the input line. The DHPT loads offset DAC configuration as a two-bit value every gate cycle (104.8 ns).

The ADC uses redundant signed-digit cyclic conversion algorithm. This algorithm compares input current to a positive and a negative thresholds, and then generates a redundant two-bit result. If the signal is not within the thresholds, then the reference voltage is added or subtracted. The signal current is then multiplied by two. Resulting signal is stored in a current memory cell and is used in the next conversion cycle.

Calibration of the current receiver and the ADC is performed by injecting external current into the channel. Two current sources can be used for calibration: the internal current source in the DCD and the external current source connected to the monitor pin of the ASIC.

The digital block of the DCD consists of the ADC data converter, high speed data serializer, ADC control block, JTAG [4] configuration interface, and offset DAC control block.

The ADC data converter decodes redundant binary representation of the conversion result to a binary representation. Data are then serialized and sent over 64 output pads to the data handling processor. At the operational frequency of 305 MHz DCD generates 19.5 Gb/s, which sums to 78 Gb/s per half ladder.

The JTAG interface is used for configuration and for testing functional blocks of the ADC.

Data Handling Processor

The data handling processor is the ASIC that synchronizes matrix operation with the digitization process, performs first digital data processing and reduction, and sends data over high-speed serial links to the data read-out system.

Figure 7.10 shows the layout of the DHPT. DHPT contains switcher sequencer, offset DAC controller, data processing path, command receiver, temperature sensor, and slow control JTAG controller.

[3] Digital-to-analog converter

[4] Joint Test Access Group

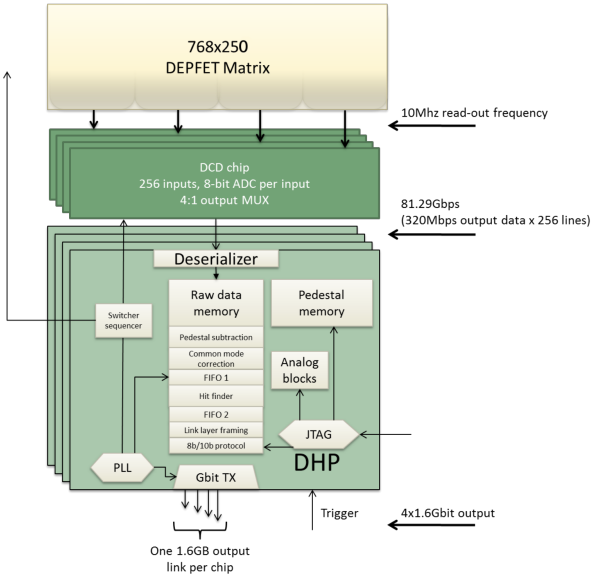


Figure 7.10.: Block diagram of the DHPT [63]

The switcher sequencer generates switcher control sequence. The sequencer is implemented as a state machine that reads sequence from a memory block and drives the sequence on the output pins. The sequence can be stored in two independent memory blocks: one for normal operation and one for gated mode operation. Sequence memory is programmed by the configuration interface. Switched between sequences is controlled by the external control command.

The offset DAC controller is responsible for loading offset DAC values into the DCD. The offset DAC values are stored in a memory block also programmed by the configuration interface.

Data processing in the DHPT starts with receiving data from the DCD. Data are buffered in the dual-port memory. Upon receiving the read command, data are read from the memory by the read process. The read process is always delayed in respect to the write process by a configurable delay. This delay compensates trigger delay in the trigger distribution path.

Data processing consists of pedestal subtraction, common mode correction, and zero suppression. First, the pre-calculated offsets, the pedestals, are subtracted from the signal amplitude. The common mode correction core then calculates offset common for all pixels in the same row and subtracts it from the amplitude. The hit finder then uses zero suppression algorithm, which discards pixels with amplitudes below a configurable threshold.

Data are then packed into frames in the Aurora protocol format [64]. Framed data are serialized and sent over the high-speed serial link, operated at 1.52 Gb/s, to the read-out system.

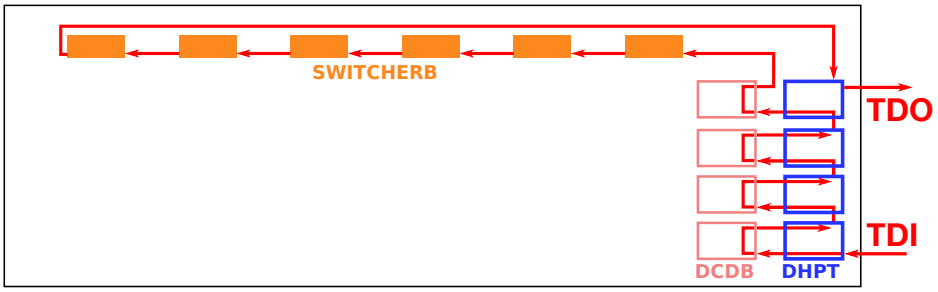


Figure 7.11.: JTAG chain of the pixel detector with all ASICs included in the chain

The DHPT contains a temperature sensor block that measures the on-die temperature of the ASIC or the temperature of the silicon frame of the detector [65]. Designed resolution of the temperature sensor is 0.1°C. Temperature sensor is operated by the configuration interface.

The slow control JTAG interface provisions configuration registers and control bus for operation of the temperature sensor. A DHPT contains a digital selector that includes or excludes attached DCD and switchers from the JTAG chain depending on the DHPT configuration.

Configuration Interface

All front-end ASICs are configured using a common JTAG interface. JTAG interface consists of four signals. The Test-Clock, TCK, and the Test-Mode-Select, TMS, are provided by the master and are shared by all ASICs. The Test-Data-In, TDI, is the slave input; the Test-Data-Out, TDO, is the slave output. The TDO pin of one slave is connected to the TDI pin of the next slave. In this way, ASICs build a daisy chain.

Figure 7.11 shows the layout of the JTAG chain with all ASICs included in the chain. The chain layout is not static. After powering up the detector, the JTAG chain includes only DHPT ASICs. The layout of the JTAG chain can be changed by changing configuration of the DHPT. Each DHPT can include or exclude attached DCD and switcher into the chain.

Chapter 8.

Data Read-Out System of the Pixel Detector

Before describing data acquisition of the pixel detector, it is useful to give an overview of the Belle II data acquisition system to highlight the need for a separate data processing path for pixel detector data. Designed trigger rate of Belle II is 30 kHz. Table 8.1 gives the expected event size in Belle II, listed as individual contributions of subdetectors. Events, generated by the pixel detector, are larger by factor 20 than events of the remaining detectors combined.

Figure 8.1 shows the layout of the Belle II data acquisition system. The layout of the data acquisition system also reflects this data rate asymmetry. The system consists of a two-stage event builder. The system, except for the pixel detector, employs a common read-out scheme. Data processing starts in the COPPER[1] cards. COPPER cards include several FPGA-based interfaces to the front-end electronics of the detector [66]. Each COPPER card also includes an Atom CPU that performs initial data processing and sends data to the read-out PC over the Ethernet network.

One read-out PC receives data from several COPPER cards, reduces data size by removing redundant information, combines data that belong to the same event, and sends data to the first event building stage over 10 Gb/s Ethernet network. The event assembly is done in the input computers to the high-level trigger subsystem. All read-out PCs send data, which belong to the same event, to the same high-level trigger input computer. The 10 Gb/s network switch, which commutates data from the read-out PCs to the high-level trigger input, is a part of the event building algorithm.

The high-level trigger analyses events online to decide, if the events contain interesting information and have to be preserved. The online selection node reduces pixel detector data based on the decision, which is met by the high-level trigger and the SVD-only tracking. Reduced pixel detector data and events from the high-level trigger are then sent over another 10 Gb/s Ethernet network to the event builder 2. The event builder 2 assembles events and sends them to storage.

Limitations of the system in respect to the pixel detector start to be visible already in the COPPER read-out. Bandwidth of the COPPER card, 1 Gb/s, is insufficient to read-out a pixel detector module that is capable to send 4×1.25 Gb/s. Even if we would be able to read pixel detector with COPPERs, network performance degradation, caused by heavy load, will still limit performance of the system. Therefore, we avoid using software in the read-out system of the pixel detector for data processing and implement all algorithms in hardware.

[1] COmmon Pipelined Platform for Electronics Readout

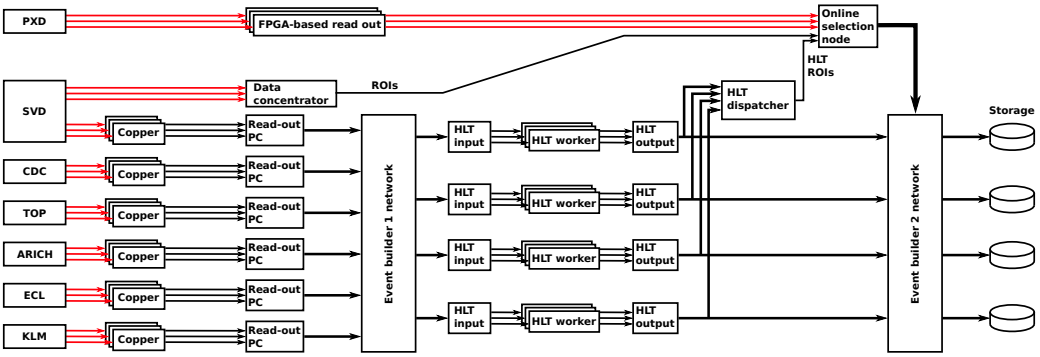


Figure 8.1.: Layout of the Belle II data acquisition system

Detector	Event size, kB
Pixel detector	800
Silicon vertex detector	14.9
Central drift chamber	6
Time-of-propagation counter	4
Aerogel RICH	2.8
Electromagnetic calorimeter	12
K_L^0 and muon detector	2.6
Belle II	842.3

Table 8.1.: Event size in each sub-detector [67]

8.1. Requirements for the Pixel Detector Read-out System

Available memory in DHPT limits average detector occupancy to 3 %. With this occupancy, the pixel detector can generate up to 20 GB/s of data. Even if processed separately, storing all pixel detector events would require a significant increase of the storage capacity.

The detector has low time resolution due to long integration time of 20 μs. This results in high occupancy because most hits in the detector are generated by beam background. These data do not contribute to track reconstruction and can be removed from the data stream.

Therefore, we focused the design of the read-out system for the pixel detector on online data filtering in hardware. The read-out system makes decision, which pixels to filter, based on the hit topology and on the track position.

The high-level trigger farm analyses tracks in software to identify events from non-background interactions that must be stored. The high-level trigger also identifies regions in the pixel detector that intersect with tracks in the event. This information allows us to efficiently reduce data by factor 10. We expect to gain an additional factor 3 by using fast track extrapolation from the SVD.

The average event processing time in the high-level trigger is 2 s. During this time the read-out system buffers events in the memories. This requires buffer space in the order of 40 GB.

8.1.1. Data Processing

Main technology used in the data acquisition system is FPGA[2]. It has several key advantages over conventional data processing in software. First, modern FPGAs have multi-gigabit serial links, which enable fast data exchange between processing modules. These links may be used for system synchronization. FPGA provides access to high density external memories with a high bandwidth interface. Memories are used as buffers, which reduce peak data rate. This also equalizes data flow in time.

Finally, main advantage of FPGAs is real time parallel data processing. Unlike data processing by the CPU, FPGA is an inherently parallel technology. We can easily replicate algorithms in FPGA, therefore processing of several data streams in FPGA takes the same time as processing only one stream. By combining parallelism and pipeline algorithms, for example by chaining smaller algorithms that process a small chunk of data at a time, it is possible to create efficient data acquisition systems.

The data acquisition system of the pixel detector consists of two FPGA-based systems: the Data Handling Hub or DHH, and the Online Selection Node or ONSEN. Figure 8.2 shows one eighth of the data acquisition system. Both systems comply to ATCA[3] standard [68]. The DHH system consists of 48 FPGA cards in the compact AMC [4] form factor, which are shown in figure 8.3, and 8 FPGA cards in the double AMC form factor. Fourty

[2] Field-programmable gate array

[3] Advanced Telecom Computing Architecture

[4] Advanced mezzanine card

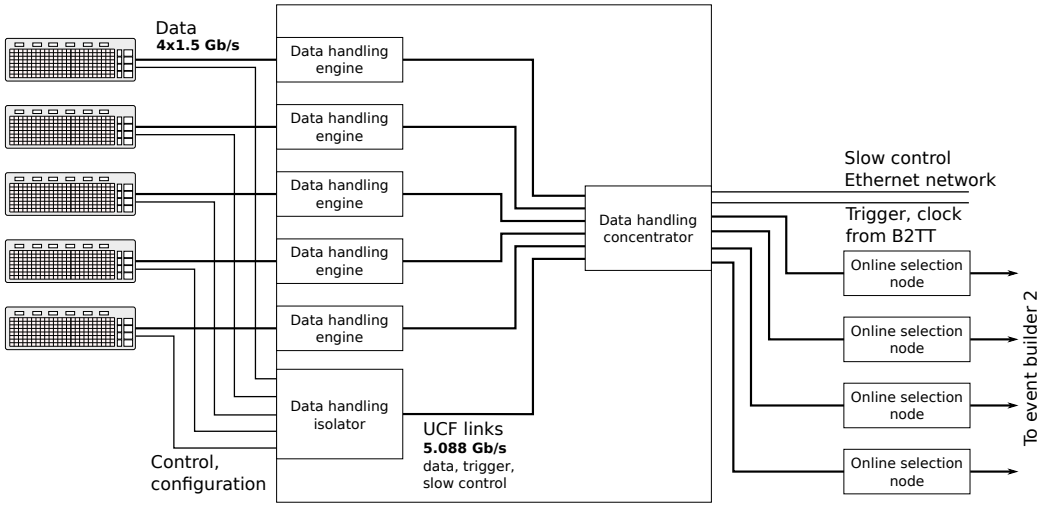


Figure 8.2.: One eighth of the pixel detector data acquisition system in Belle II

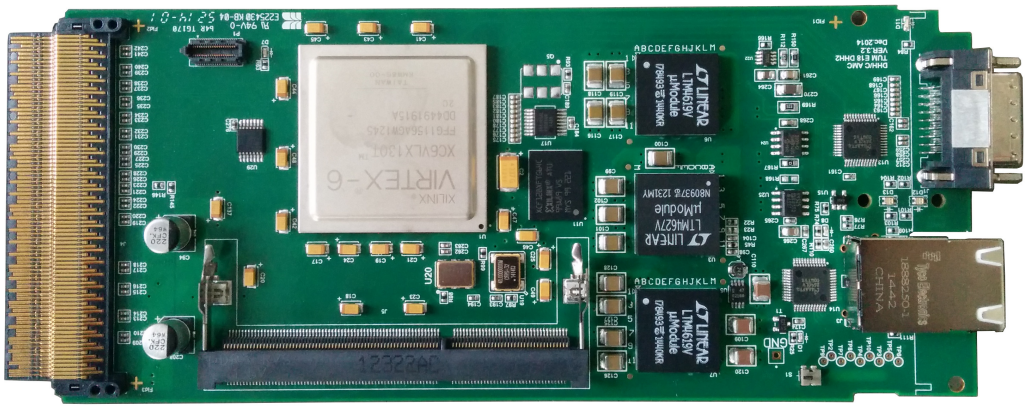


Figure 8.3.: DHH module in the compact AMC form factor

modules are called data handling engines or DHE. They read data from detector modules over four optical high-speed links. Eight modules, called the data handling isolators or DHI, control and configure front-end ASICs of the detector. The DHIs galvanically isolate signal lines of each detector. The remaining 8 modules are called data handling concentrators or DHC. Each DHC is connected to 5 DHE, the Belle II trigger distribution system, slow control network, and the ONSEN system. The DHC distributes trigger and slow control to the DHEs and DHI, receives detector data from the DHEs, and sends them to the online selection nodes. The ONSEN system is built of 34 FPGA cards in the AMC form factor, which are shown in figure 8.4. Two modules, the merger nodes, are dedicated to receiving a per-event filter decision, the high-level trigger. 32 modules, the selectors, receive data from the DHH system, regions of interest, and trigger information from the merger nodes, filter data, and send filtered data to the second event builder stage.

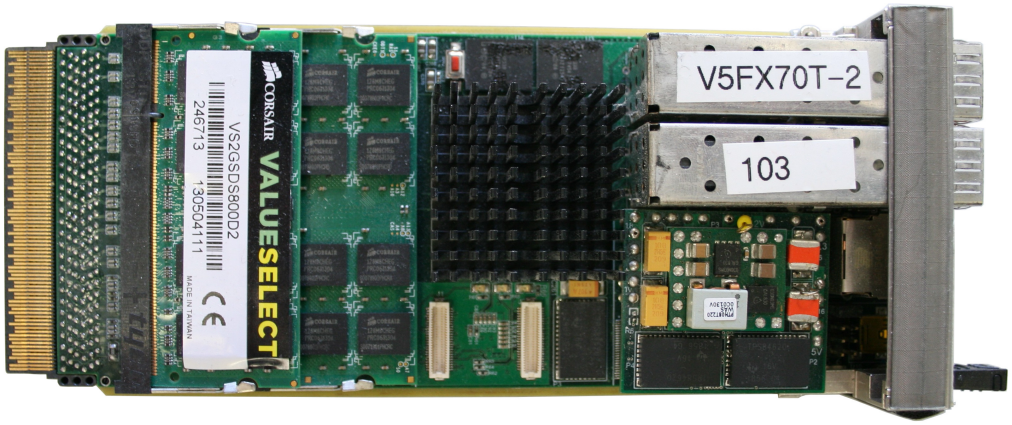


Figure 8.4.: ONSEN module in the AMC form factor

The DHH system receives trigger and synchronization information from the Belle II time and trigger distribution system, the B2TT [69]. The trigger information is converted into the read command that is sent to the DHPTs. DHPTs send data frames in reply. DHH system receives data, prepares data for data reduction, builds subevents, and sends them to the ONSEN system for data filtering. The system also mirrors outgoing data stream to a stand-alone PC for data quality monitoring.

ONSEN system filters data according to filtering conditions received from outside. Two filtering conditions are used by ONSEN: high-level trigger, and regions of interests. The high-level trigger is calculated online using information available in event builder 1. The high-level trigger is a binary decision for keeping or rejecting the event. Regions of interest are rectangular regions in the pixel detector, which are calculated by extrapolating the reconstructed charged tracks back to the pixel detector planes. The high-level trigger is generated in the event reconstruction software running on the incomplete events after the first event builder stage. Regions of interest are generated in the event reconstruction software, and, independently, in the SVD-only hardware track reconstruction system, the Data Concentrator [70].

Two algorithms are available for data filtering with regions of interest. The hit-based algorithm rejects hits outside of the regions of interest, while saving hits inside the regions. The cluster-based algorithm saves all pixel clusters that intersect regions of interest. Next, data are sent to the second event builder stage for merging with the rest of the event.

Estimated data reduction factor of the system is 30 [71]. The high level trigger is expected to trigger 1/3 of the events. An additional factor 10 comes from data reduction using regions of interest. Detailed description of the ONSEN system is given in [71].

8.2. Data Handling Hub System

The data handling hub system plays a key role in the pixel detector read-out by connecting the detector to data acquisition, slow control, and trigger distribution systems. There are three main tasks dedicated to the DHH system. The DHH system reads data from the detector and supervises read-out process. This task requires receiving trigger signal from the global trigger distribution system, synchronous trigger distribution to the read-out modules, and reading of the front-end ASICs.

The next task is preparing data for data reduction. Data from the detector are synchronized between four streams, based on the event number and timing of the trigger signal. If several triggers overlap within 20 μs integration window, data are shared between triggers. Events are then processed by the clustering algorithm that reconstruct clusters from hits, which are adjacent to each other. These hits are combined to a cluster that gets a unique ID. By analyzing these clusters, we can identify and save low-energy pions, which do not reach outer detectors and would be removed in the ONSEN system. The subevent builder combines events with the same event number from 5 read-out modules to a subevent and sends subevents to the ONSEN system.

The last main task of the DHH system is control and configuration of the front-end ASICs. Data handling engines and data handling isolators in the final system provision a JTAG master interface to the detector. The interface is accessible from the slow control EPICS network. FPGA cards also carry a precision current source for absolute ADC calibration.

Two versions of the system were developed. A stand-alone version suitable for the laboratory setups consists of a single FPGA card on a VME[5] carrier board. The system is optimized for detector development and characterization. This system reads detector data and sends them out without processing over the Ethernet network to a PC.

The final system is built in the ATCA form factor. The system consists of 8 ATCA carrier cards, and 8 rear-transition modules, which carry external interfaces. The carrier card houses 7 FPGA cards in the compact AMC form factor. The system is build in two layers that follows data flow through the system. Five first-layer cards, the data handling engines, provide data interface to the sensor, and a high-speed link to the subevent builder. The subevent builder, the data handling concentrator, provides interface to the B2TT system, to the slow control Ethernet network, the data quality monitoring Ethernet network, and to the online selection nodes.

8.3. Hardware Components

Hardware used in the DHH system evolved from the stand-alone prototype system in the VME form factor to the final modules in the compact AMC form factor and the ATCA carrier boards. The prototype module was developed in 2012 to evaluate different aspects of the system before implementing the system in the final form factor. After fixing the

[5] Versa Module Europa bus

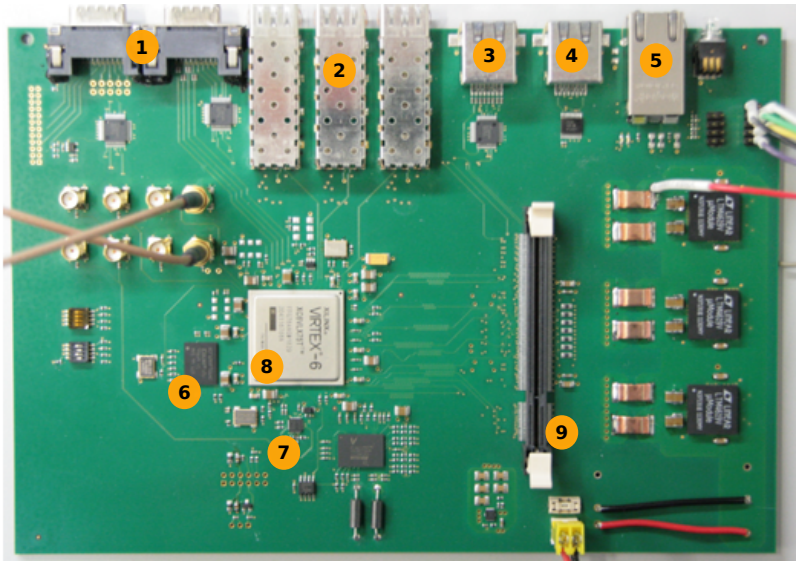


Figure 8.5.: The DHH prototype module

design requirements, 10 modules in the compact AMC form-factor were produced in 2013. The final modules were produced in 2014 with small modifications in the module design.

8.3.1. DHH Prototype Module

Figure 8.5 shows the prototype module. Following components are highlighted on the photograph:

1. Infiniband connectors
2. SFP cages
3. RJ45 connector for trigger interface
4. RJ45 connector for JTAG interface
5. RJ45 connector for Ethernet connection
6. Flash chip for storing firmware
7. Clock synthesizer and jitter cleaner Si5338
8. Virtex-6 FPGA
9. DDR2 SODIMM slot

The card is built in the VME form-factor. The internal logic of the module is implemented in the Xilinx Virtex-6 VLX75T-2 FPGA. The card is equipped with the detector interface

that consists of two Infiniband connectors as an interface to the detector, three SFP+ cages for connecting the card to the ONSEN system, an RJ45 connector for B2TT connection, an RJ45 connector for the Ethernet communication, and a DDR2 SODIMM socket.

The detector interface consists of the four high-speed link inputs, and fast control lines: reference clock, trigger, frame sync, reset, and four JTAG lines. Additionally, there are two analog lines: DHP-IO voltage sense and current source output for DCD calibration. All digital signals except for the reference clock are connected to the FPGA. The reference clock for detector and FPGA is generated in the programmable clock synthesizer Si5338.

The LVDS buffers drive output signals on the detector interface to protect detector from being powered over the control lines. The incoming DHP-IO voltage sense line is connected to a comparator input that controls output of the driver. If the DHPT is turned off, the sense voltage falls below the comparator threshold. This switches outputs of the LVDS buffer in high-impedance state.

8.3.2. Programmable Clock Synthesizer Si5338

The programmable clock synthesizer Si5338 can generate a high-quality clock signal in a wide frequency range with deterministic phase relation to the reference clock. The clock synthesizer receives clock from FPGA and synthesizes reference clock for front-end ASICs and FPGA.

The clock synthesizer consists of a PLL and programmable clock dividers. The PLL generates high-quality clock signal, which is phase aligned to the input clock. This reduces jitter of the output clock. The output signal of the PLL, which has frequency around 2.5 GHz, is divided in the clock dividers to obtain clock with desired frequencies.

The configuration of the PLL and the clock dividers of the Si5338 is stored in 512 registers. The registers are programmed by the FPGA over the I^2C control interface. These registers are not stored between power-cycles and need to be programmed after every time the module is powered.

8.3.3. DHH AMC Modules

Final DHH modules are designed in the compact AMC form factor. The design of the AMC cards follows the design of the DHH prototype. The differences are listed below.

The AMC cards use Xilinx Virtex-6 VLX130T-2 FPGA, which contains approximately twice as much logic blocks as the FPGA used in the prototype board [72]. All external interfaces except for the detector interface are moved to the AMC connector. They are accessible from the carrier boards.

The Ethernet PHY chip is removed in favour of the SGMII[6] SFP copper module, shown in figure 8.6. This module consists of a PHY chip, a transformer, and an RJ45 connector. This greatly simplifies the design of the card because the SGMII interface requires only 2 differential pairs in contrast to 27 signals in the GMII[7] interface used in the prototype.

[6] Serial Gigabit Media Independent Interface

[7] Gigabit Media Independent Interface

Figure 8.6.: SGMII SFP adapter module

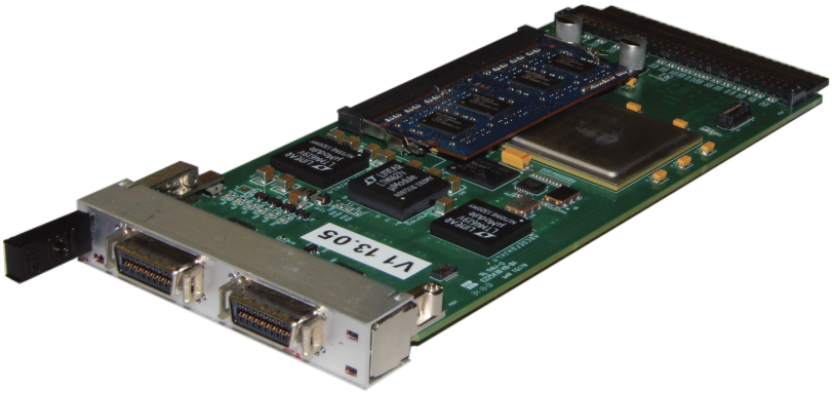


Figure 8.7.: DHH AMC v. 1 card

Alternatively, the SGMII SFP copper modules can be replaced by the fiber optics modules, if signal integrity must be improved.

The B2TT command line is connected in parallel to the FPGA and to the clock and data recovery chip ADN2814. The chip contains a PLL that can recover clock information from the DC-balanced 8b/10b-encoded B2TT control signal. The recovered clock is phase adjusted to data and has improved jitter characteristics in comparison with the clock signal provided by B2TT system.

The DDR2 SODIMM slot is replaced by the DDR3 SODIMM slot on the AMC card. The DDR3 memory provisions higher memory densities and higher bandwidth than DDR2 memory.

Two versions of the AMC cards were produced. Figure 8.7 shows version 1 of the AMC card. The card is designed with two Infiniband connectors on the detector interface. This version is compatible with the DHP v. 0.2 and DHPT ASICs. The precision current source, used for the DCD calibration, is produced as a piggyback module that is installed on top of the AMC card.

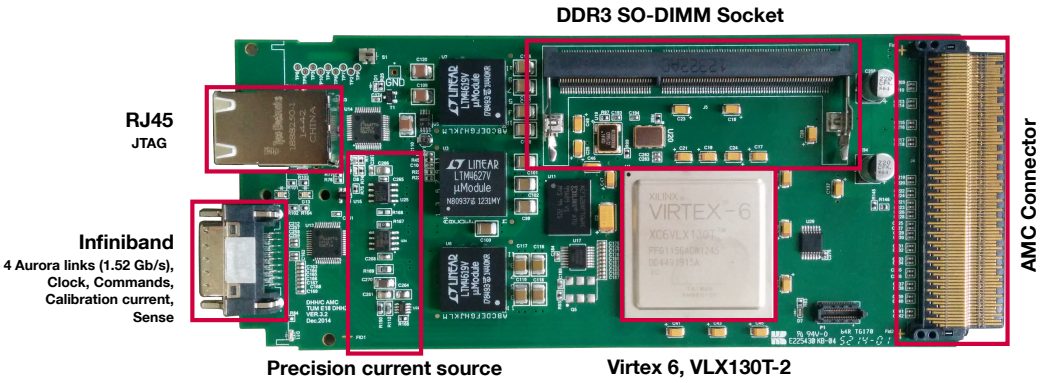
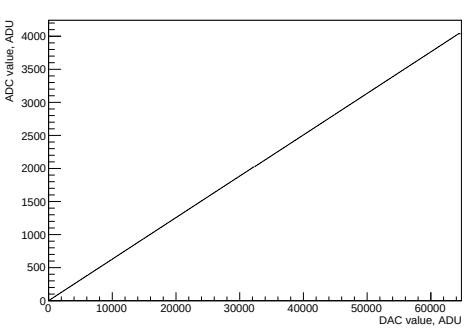
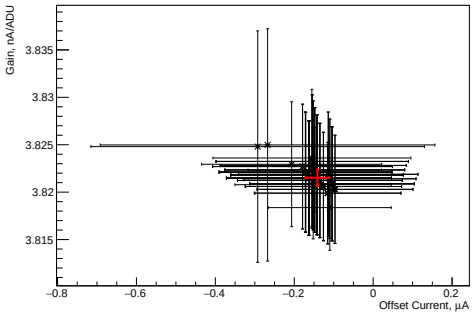


Figure 8.8.: DHH AMC v. 3 card



(a) Linearity of the current source



(b) Spread of the gain and pedestal offset. Red marker shows the weighted mean

Figure 8.9.: Measurement of the current source

Figure 8.8 shows version 3 of the AMC card. The card is designed with one Infiniband connector and one RJ45 connector. Using a CAT7 Ethernet cable with smaller width instead of one Infiniband cable allows us to relax space requirements for the cable routing in the inner region of Belle II. The Infiniband connector carries high-speed digital signals: data, clock, command, and analog lines: DCD current source and DHPT sense lines. The RJ45 connector carries JTAG lines to the detector. The precision current source is integrated on the module to provide better electric contact. We added the signal buffer on the JTAG lines of the FPGA to improve the signal quality.

Current Source

The current source can inject current in the range 0-500 μA with the step of 8 nA. The DAC in the current source is programmed by FPGA over the SPI interface. The ADC monitors the voltage level on current source output. The ADC is read out by the same SPI interface. Appendix F shows the schematics of the current source.

We tested the linearity of the current source using the on-board ADC and a 10 kOhm resistive load. We scanned full dynamic range of the DAC and measured voltage drop at the resistive load with the ADC. Figure 8.9a shows the result of the scan in one DHE module. The current source shows linear behavior throughout full dynamic range of the DAC.

We repeated the scan with all available DHE modules to measure the deviation of the gain and of the current pedestal offset. During the scan we fitted the voltage curve with a first degree polynomial and histogrammed parameters obtained by the fit. Figure 8.9b shows the result of the measurements. Parameters on all cards agree within uncertainties of the measurement. Mean gain is

$$\bar{g} = (3.82152 \pm 0.00087)\,\frac{nA}{ADU} \tag{8.1}$$

with mean offset

$$\bar{I} = (-0.141 \pm 0.030)\,\mu A. \tag{8.2}$$

Because precision of the ADC is 80 nA in the fine mode, this resolution is sufficient for absolute calibration of the ADC [73].

DDR3 Adapter Cards

We designed two extension cards with the DDR3 SODIMM connector to extend the functionality of the DHH AMC card.

Figure 8.10 shows a card with 8 LEMO connectors: four inputs and four outputs. The card carries active components to convert signals between the SSTL[8] 1.5 V and the NIM[9] standards. The card is used in the laboratory setups for synchronization of the detector read-out and external electronics, for example a laser is used in detector characterization.

Figure 8.11 shows a card with two Infiniband connectors. The card converted the signals between the SSTL 1.5 V and LVDS standards. We used the card to control and configure switchers on the Hybrid6 DEPFET module during the beam test in January 2014. Section 9.1.1 describes the beam test setup in details.

8.3.4. VME Carrier Board

The DHH AMC cards operates in a carrier board that provides power and connectors for external interfaces. We designed the carrier card, shown in figure 8.12, in the VME form factor. The VME connectors on the carrier card provide 5 V power from the VME crate and mechanical support in the crate. The carrier card provides AMC connector for the AMC card, JTAG connector for configuration of the FPGA on the AMC card, RJ45 connector for connection to the B2TT system, and 16 general purpose SFP cages.

This type of the carrier board is used in the laboratory setups and for the beam tests due to its simple design.

[8] Stub Series Terminated Logic

[9] Nuclear Instrumentation Module

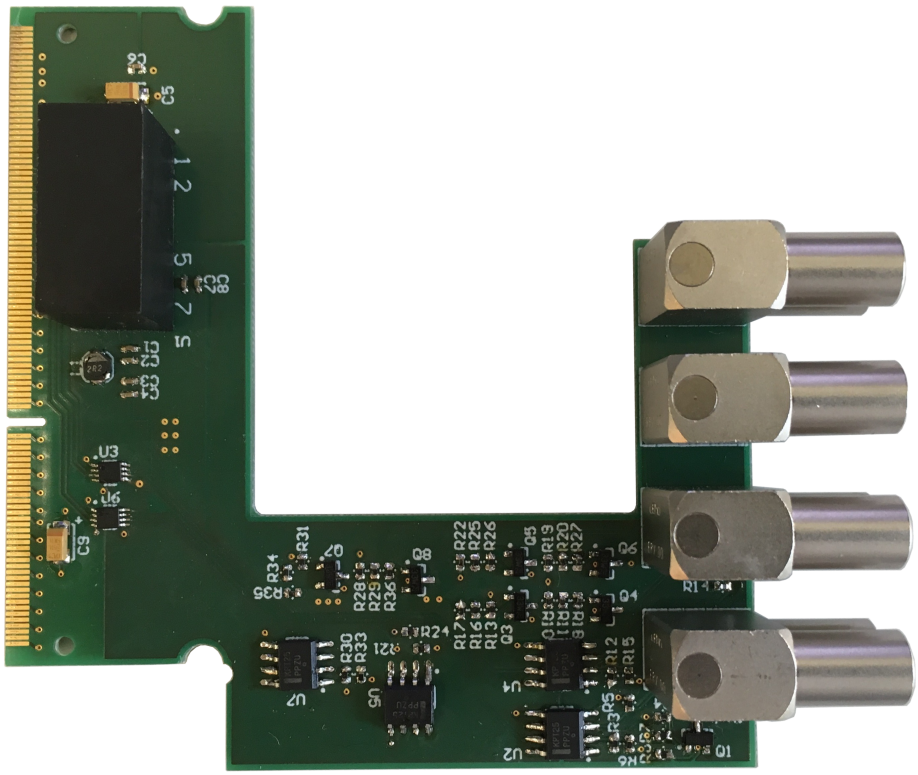


Figure 8.10.: DDR3 to LEMO adapter

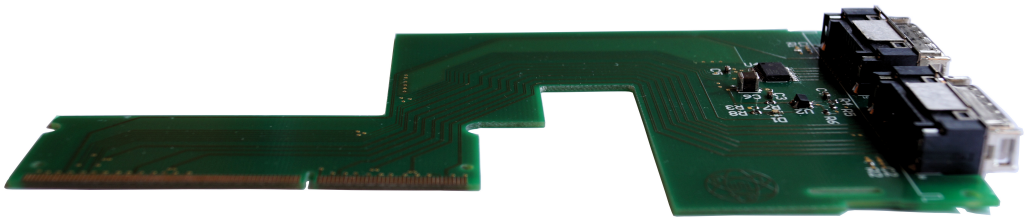


Figure 8.11.: DDR3 to Infiniband adapter

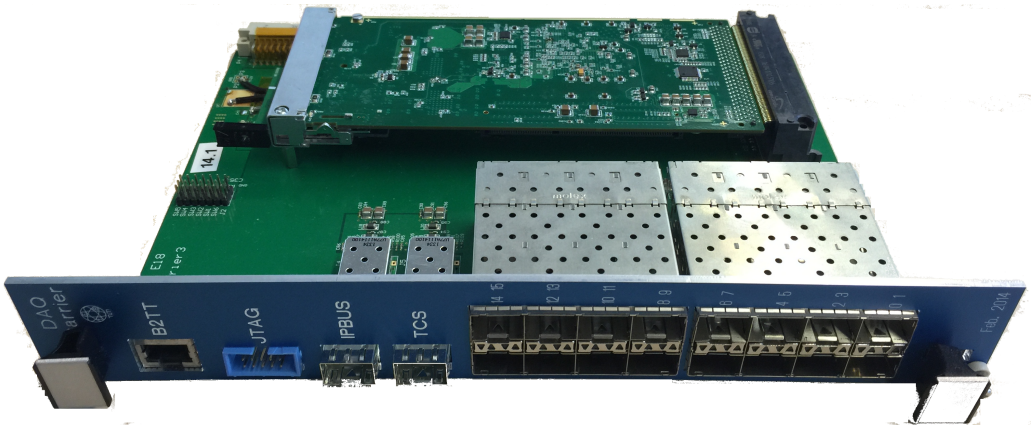


Figure 8.12.: Carrier board in the VME form factor with installed DHH card

8.3.5. ATCA Carrier Board and Rear-Transition Module

The DHH system at Belle II is built in the ATCA form factor. One functional unit of the DHH system consists of five AMC DHH cards, configured as DHEs, an AMC DHH card, configured as a DHC, a Data Handling Insulator, an ATCA carrier board, and a rear-transition module.

Figure 8.13 shows the ATCA carrier board. The carrier board provides 8 AMC slots, a slot for the IPMI[10] controller, and a connector to the rear-transition module, which provides additional interfaces. Appendix H gives detailed information about the carrier board.

Figure 8.14 shows the rear-transition module. The module provides external interfaces for the AMC cards. The DHC card is connected to four SFP+ cages for communication with ONSEN system, two SFP+ cages for communication with slow control and data quality monitoring networks, and to a RJ45 connector for communication with B2TT system. Two 12-channel optical receivers connect high-speed data links from the detectors to DHEs.

8.3.6. Optical Transmitters

During tests of the detectors, we observed degradation of signal quality of the high-speed links from the detector. Amplitude of the voltage swing at the FPGA is 70 mV for some links, while the minimal differential peak-to-peak input voltage for Virtex-6 GTX receivers is 125 mV [74]. Degradation of the signal quality caused increase in transmission errors and drop of the Aurora link. To improve signal quality, we decided to replace copper links with fiber optics that has lower signal attenuation. As a side effect, we replaced the combination of Infiniband and RJ45 cables with a single Camera Link cable [75]. We designed an additional module, the data handling insulator, which provides 5 Camera Link interfaces for detector control.

[10] Intelligent Platform Management Interface

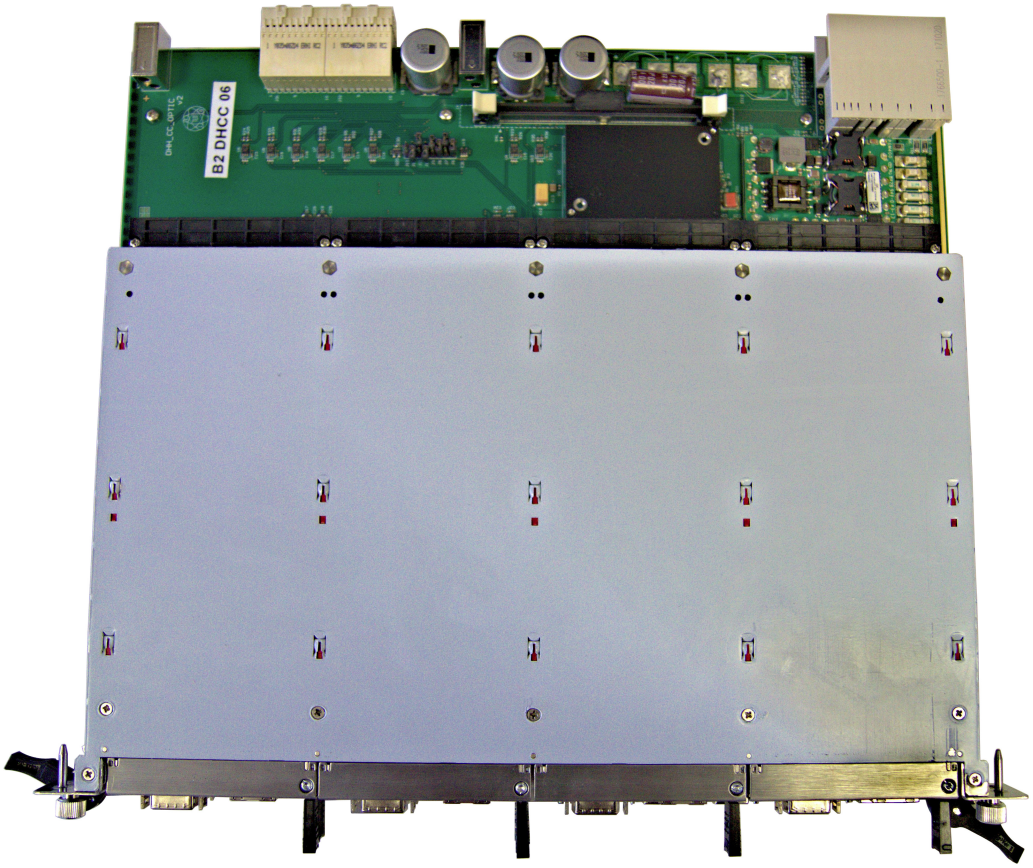


Figure 8.13.: ATCA carrier board

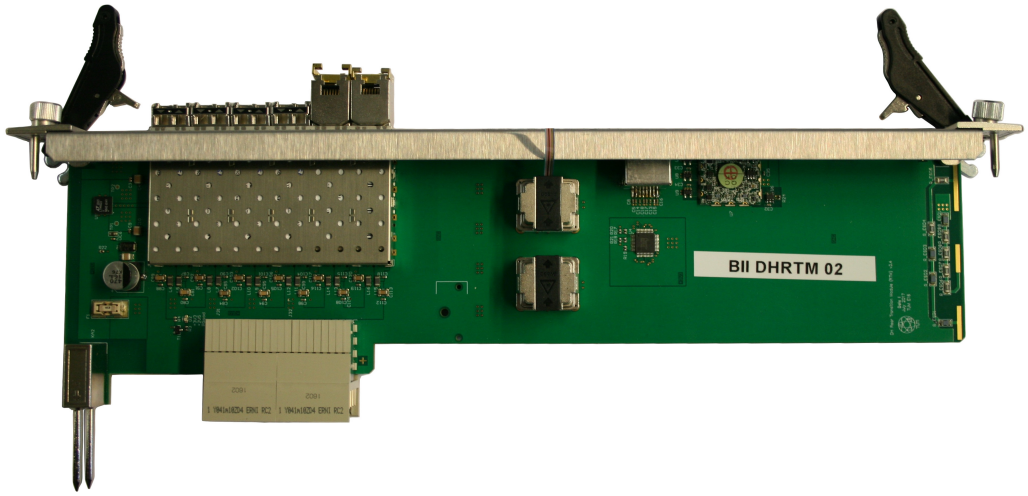


Figure 8.14.: Rear-transition module

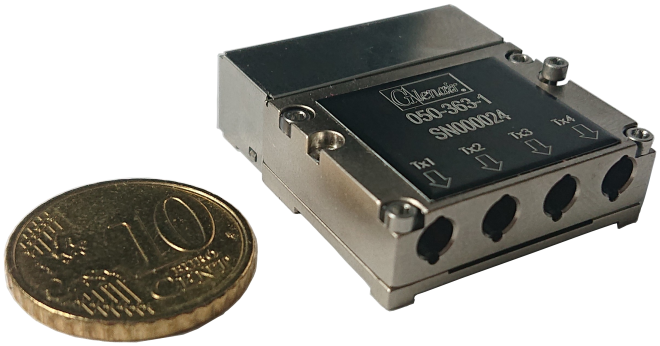


Figure 8.15.: Glenair 4-channel transmitter 050-363

Replacement of the link medium requires the installation of the optical transmitters very close to the pixel detector. The only available space is the dock box region located in a ring outside of the central drift chamber [76].

There are two requirements that must be fulfilled for installation of the optical transmitters. The dock box region is shared with the DC/DC converters for the SVD. Therefore, available space limits dimensions of the transmitters. This requirement eliminates VTTx transmitters, developed in the framework of the versatile link project, aimed at developing radiation hard optical transceivers for high-luminosity upgrade of the LHC[11] experiments, from the list of potential candidates [77]. Finally, the dock box region is not shielded from ionizing radiation and neutrons, produced electrons colliding with beam pipe walls.

We identified two possible candidates, which fulfill the dimension requirement. The first candidate is a 4-channel multi-mode fiber transmitter 050-363 from Glenair, which is shown in figure 8.15. The second candidate is the 12-channel multi-mode fiber transmitter from the competitor company.

We exposed both transmitter types to gamma and neutron radiation to qualify them for 10 years operation at Belle II conditions. Figure 8.16 shows the layout of the setup used during radiation tests. The device under test, the DUT is installed on a PCB inside the radiation environment. The DUT receives data from the FPGA card over the passive copper cable and retransmits the signal back to the FPGA board via an optical fiber link. The FPGA generates pseudo-random data, transmits data to the DUT at 2 Gb/s using Aurora protocol, receives data, and checks data for errors. The control PC monitors error counters, link status, and current consumption of the transceivers over Ethernet network. In addition, the PC directly monitors transmitter conditions, such as currents and temperature, by reading them from the DUP with FPGA over a dedicated copper I^2C link.

Transmitters were irradiated in steps. Between steps, we power cycled the transmitter to emulate operational cycle at the experiment.

Belle II simulations predict maximum neutron flux $1.15 \times 10^{11}\, cm^{-2} yr^{-1}$ for the ARICH detector, which surrounds the forward dock box region [78]. We use this flux as an upper

[11] Large Hadron Collider at CERN

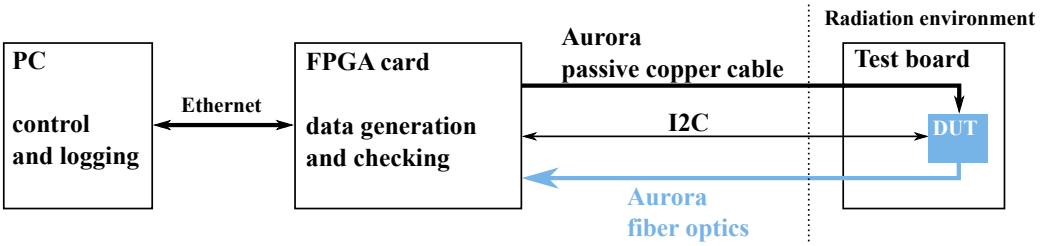


Figure 8.16.: Layout of the radiation tests of the optical transmitters

limit for our tests. We did the neutron illumination at the NECTAR source in FRM II reactor[12] in Garching. We exposed both transmitter types to 1.8 MeV neutrons for 5435 s at the neutron rate of $2.3 \times 10^8 cm^{-2} s^{-1}$. The dose, absorbed by transmitters, corresponds to 10 years' equivalent dose of neutrons at Belle II. During the illumination time, both devices operated stably and no errors or link drop events were recorded. This measurement concludes that both devices are immune to neutron irradiation in the environment of the Belle II experiment.

Belle II simulations in the 15th background campaign calculate maximal total dose in the central drift chamber at 250 Gy per yr. We irradiated both types of devices in the Gammacell 220 chamber at the Helmholtz center in Munich. The Gammacell 220 uses Cobalt-60 as a radiation source with rates in the order Gy/min. The Glenair transmitter received a total dose 2500 Gy, which corresponds to 10 years operation at the Belle II conditions. No errors were detected during irradiation. Two devices of the second transmitter type stopped working after receiving 766 Gy and 911 Gy. The high-speed link drop coincided with the loss of I^2C communication and the reduction in current consumption of the transmitter. This measurement concludes that only the Glenair transmitter fulfills Belle II requirements in radiation hardness.

Based on the results of the irradiation tests, we selected Glenair 050-363 transmitter for installation at Belle II.

8.3.7. Data Handling Insulator

During the development of the ATCA carrier board we faced a problem with galvanic isolation of the DHE modules. There is not enough space for 5 isolating DC/DC converters on the ATCA carrier board. High-speed connections between DHE and DHC have to be galvanically isolated as well. This requires designing a loop through the RTM, which involves 2 optical transceivers per channel. This solution was also declined because of the limited space on the RTM. We decided to implement all electrical detector interfaces on a separate AMC module, the data handling isolator.

Figure 8.17 shows the data handling isolator. The module is built in the double-width compact AMC form factor. The DHI provides 5 galvanically isolated electrical detector channels. Each channel provides a single camera link connector. The camera link cable

[12] Forschungs-Neutronenquelle Heinz Maier-Leibnitz (Research Neutron Source Heinz Maier-Leibnitz)

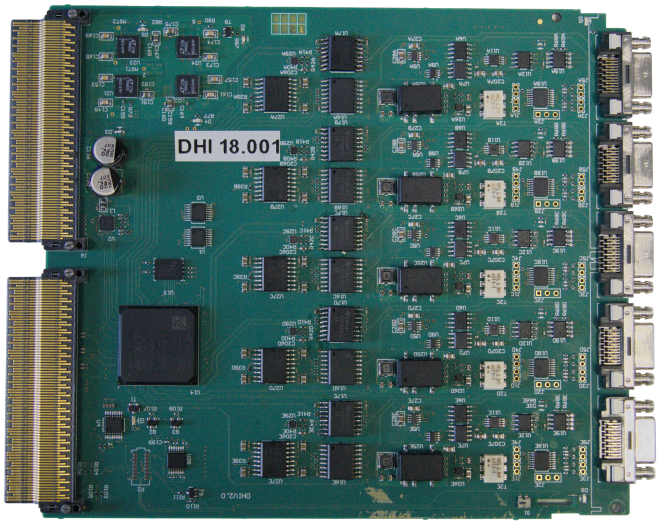


Figure 8.17.: Data Handling Isolator card

carries slow control signals, command signal, analog current source signal for the pixel detector module, and power lines for the optical transmitter installed in the dock box. The channels are controlled by a single Artix-7 FPGA installed on the DHI. This FPGA takes over detector slow control and synchronization functionality from the DHE modules.

8.4. System Clock

Clock signal is the most important signal in the synchronous systems. Because clock drives digital components, noise on the clock signal propagates through the system and is convoluted with intrinsic noise of the system. Clock signal is particular important in the pixel detector read-out because stability of the clock influence stability of the high-speed links in the system. Therefore, clock distribution has to be designed with extreme care. We use an external phase-locked loop to reduce noise on the input clock.

The DHH system uses two clock sources. The stand-alone system usually does not have an external reference and uses an internal 127.21 MHz oscillator. The system installed in Belle II has to run synchronously with other detectors. Therefore, the Belle II time and trigger distribution system provisions common RF clock with frequency 127.21 MHz. The B2TT system derives this clock from the SuperKEKB's main 508.84 MHz clock through synchronous division by 4. The DHC receives this clock and delivers it synchronously to the DHE and DHI cards. The DHE and DHI use this clock to synthesize the detector clock GCK, which frequency is related to the B2TT clock by factor 3/5. Because synchronization of the clocks with the odd frequency relation is not a trivial task, I developed a method to synchronize phases of the GCK and B2TT clocks.

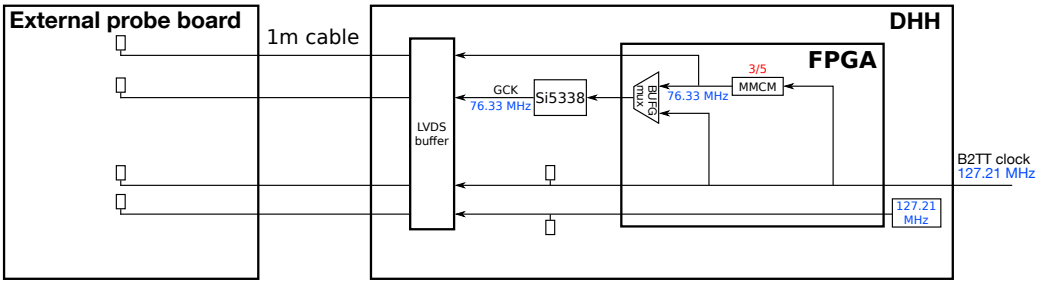
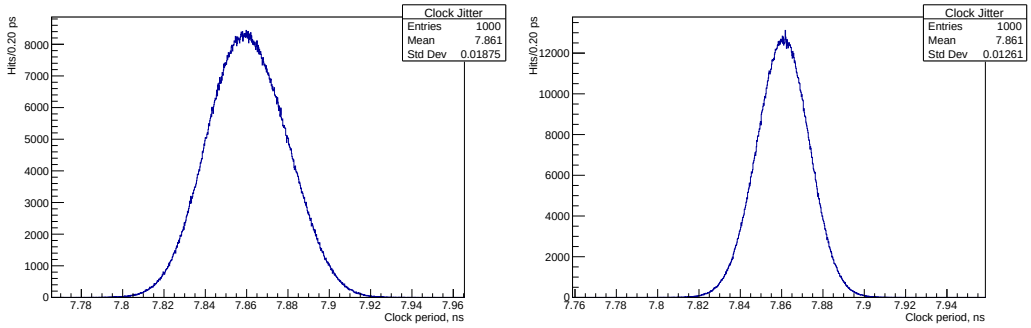


Figure 8.18.: Layout of the clock measurement setup



(a) Jitter of the B2TT clock

(b) Jitter of the oscillator clock

Figure 8.19.: Jitter of the input clocks at the FPGA output

8.4.1. Quality of the Clock

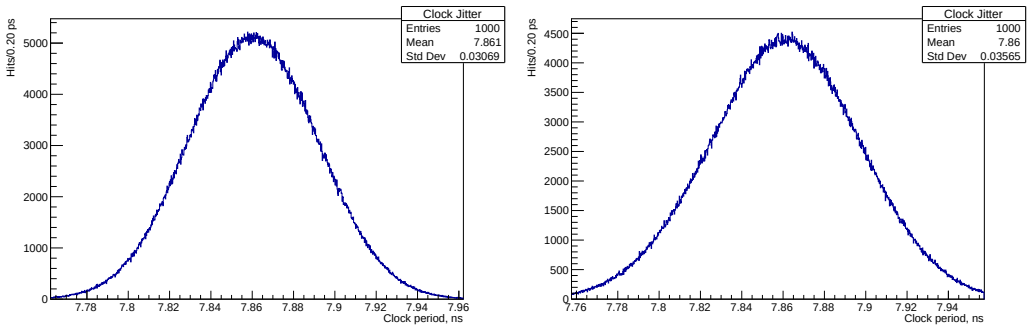
Figure 8.18 shows the layout of the clock measurement setup. There are two 127.21 MHz input clocks to the FPGA: the oscillator on the DHH card and the external B2TT clock. The FPGA drives the clock synthesizer Si5338 and four LVDS lines connected to the signal buffer. The Si5338 drives the synthesized 76.326 MHz clock, called the GCK, as an LVDS signal to the signal buffer. The signal buffer sends LVDS signals to the external board through 1 m Infiniband and CAT7 cables where the real-time oscilloscope samples the signals. The oscilloscope also samples input clocks at the FPGA's outputs on the DHH card.

Depending on firmware configuration, FPGA can output input clock signals unchanged or synthesize 76.326 MHz clock for Si5338 and the measurement board.

Clock Jitter

The jitter of the clock is a measure for fluctuation of the clock period. I measure jitter by histogramming clock period using the real-time oscilloscope and fitting the histogram with the appropriate distribution.

First, I measured the jitter of the input clocks. Figure 8.19 shows jitter of the input clocks at the outputs of FPGA. The jitter of the oscillator clock is 12.6 ps, the jitter of



(a) Jitter of the B2TT clock

(b) Jitter of the oscillator clock

Figure 8.20.: Jitter of the input clocks at the external board

the B2TT clock is 18.8 ps. This is expected because B2TT clock experience degradation through transmission over a CAT7 cable, which acts as a low-pass filter.

To qualify the effect of the LVDS buffer and clock transmission over the cable I measured the clock jitter at the external board. B2TT clock is delivered to the external board over 1 m Infiniband cable, the oscillator clock is delivered over 1 m CAT7 cable. Figure 8.20 shows jitter of the input clocks at the external board. Jitter of the oscillator clock is 35.7 ps, the jitter of the B2TT clock is 30.7 ps. Because the clock buffer has the same effect on signal degradation in both measurements, difference between the measurements at the DHE and at the external board comes only from difference in the characteristics of the cables. This shows that the Infiniband cable has lower attenuation than the CAT7 cable.

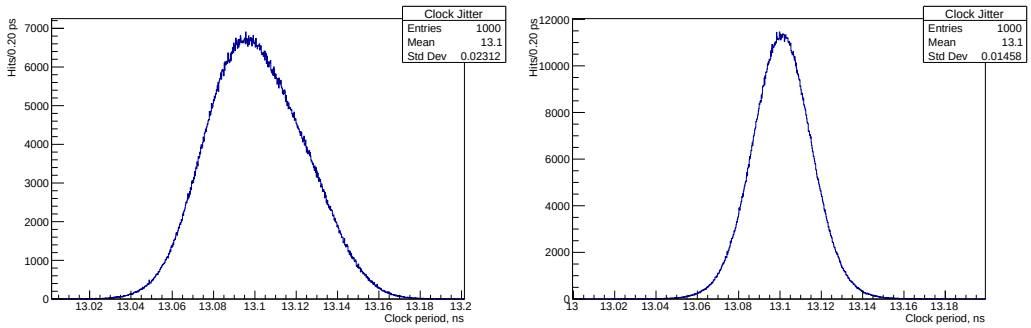
Next, I studied quality of clock synthesis in FPGA and Si5338. All measurements are done at the external board.

Figure 8.21a shows jitter of the GCK, synthesized in FPGA. Jitter of the GCK, synthesized in FPGA, is 23 ps. While jitter of the GCK is lower than jitter of the B2TT clock, the jitter distribution of the GCK, synthesized in FPGA, is skewed. This points at the imperfect implementation of the PLL in the Virtex-6 FPGA.

Figure 8.21b shows jitter of the GCK synthesized in Si5338 from the B2TT clock. Jitter of the GCK, synthesized in Si5338, is 15 ps. The measurement shows dramatic improvement of the clock quality and the shape, if compared with the GCK synthesized in FPGA. Figure 8.22 shows measurement of the GCK, synthesized in the Si5338 from the 76 MHz clock, synthesized in FPGA. The measurement shows further jitter improvement of the GCK, compared with the clock synthesized from the B2TT clock, by 1 ps. The symmetric shape of the jitter distribution also shows that Si5338 is immune against skewed period distortion of the input clock.

Phase Stability

For the measurement of phase stability of Si5338, I used two DHH cards driven by the same B2TT clock. Figure 8.23 shows the distribution of phase difference between two



(a) GCK synthesized in the FPGA (b) GCK synthesized in the Si5338

Figure 8.21.: Jitter of the GCK, synthesized from the B2TT clock

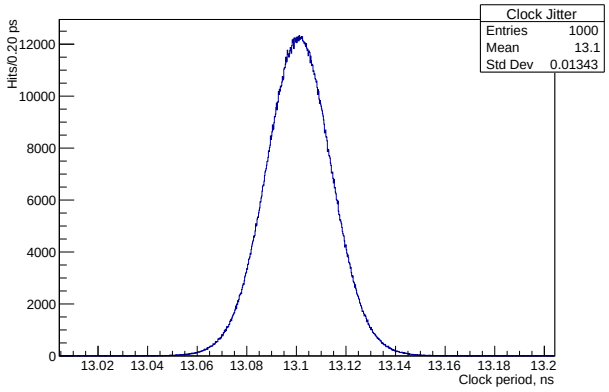


Figure 8.22.: Jitter of the GCK, synthesized in the Si5338 from the 76 MHz clock, synthesized in the FPGA

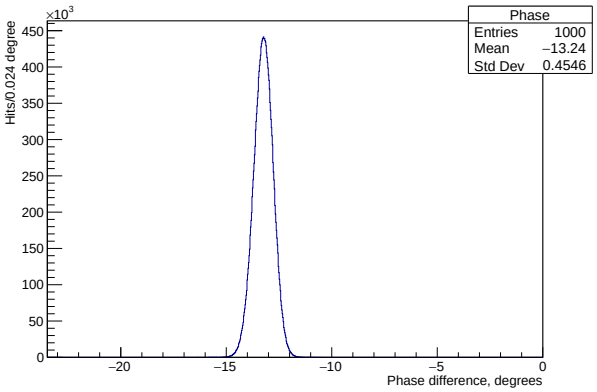


Figure 8.23.: Distribution of the phase difference between the GCK generated by two DHE cards

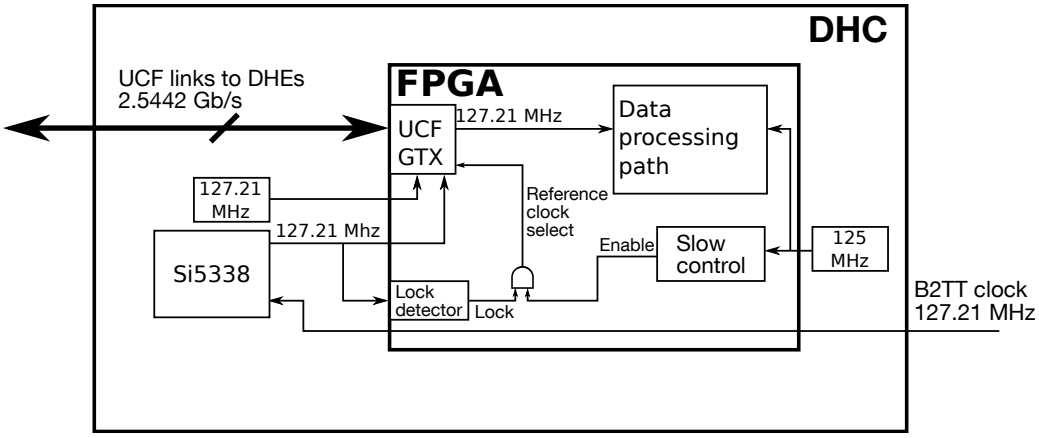


Figure 8.24.: Layout of the clock distribution in the DHC

clocks. The phase remains stable within 0.45°. I run the measurement for 24 hours to test stability of the phases on two DHH cards. The 24 h test did not record any jumps of the phase difference between both cards. This measurement indicates that DHH cards can keep reliable phase to each other.

The phase between 127.21 MHz and GCK synthesized in Si5338 is not unique. It depends on synchronization of the clock dividers in Si5338, which we cannot resynchronize from outside. Section 8.4.3 describes the method to solve this problem.

8.4.2. Clock Processing in the DHC

Figure 8.24 shows the layout of clock distribution in the DHC. There are two sources of the reference clock signal in the DHC: the free running oscillator and the B2TT clock. Both clocks have the same frequency 127.21 MHz. The B2TT clock is connected to the Si5338 clock synthesizer, which works in a jitter cleaner mode. The output clock signal of the Si5338 chip has frequency 127.21 MHz. Both clocks are connected to the reference clock ports of high-speed serial transceivers in the UCF links. The UCF protocol embeds clock information into the serial data stream that is sent to DHEs and DHI.

Selection of the main clock source is needed to keep system operational, if the B2TT clock is unavailable. By default, the free running oscillator is selected. The output of the jitter cleaner is connected to the lock detector circuit, implemented as a clock manager. Once the B2TT clock is stable and the jitter cleaner is configured, the lock detector permits clock switching. The switching is done by sending the GTX reset command over the slow control.

The Ethernet interfaces, used by slow control, UDP link to the local DAQ, Aurora links to ONSEN, and data processing logic operate with 125 MHz reference clock.

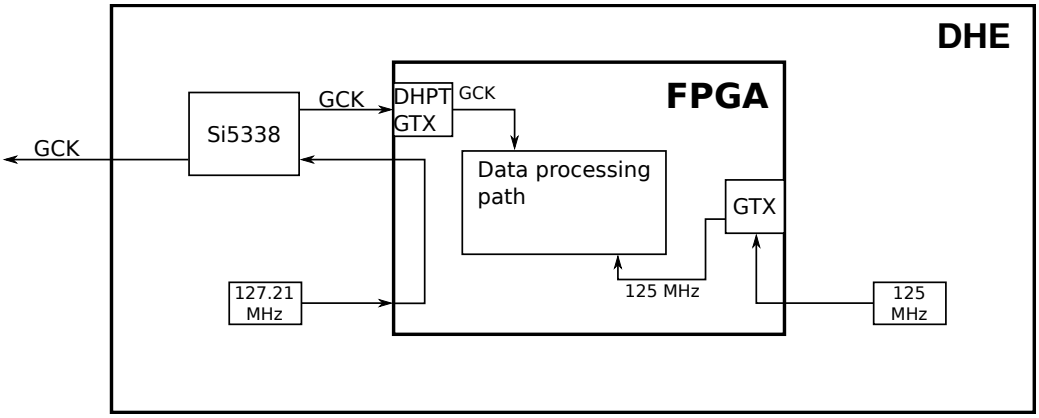


Figure 8.25.: Clock distribution in the standalone system

8.4.3. Clock Processing in the DHE

Standalone Setup

Figure 8.25 shows the layout of clock distribution in the standalone system. The clock goes through FPGA unchanged to the Si5338. The Si5338 multiplies the clock by factor 3/5 to obtain the GCK clock. The GCK is provided to DHPTs and as the reference clock to high-speed transceivers of the FPGA, which receive data from the DHPT. The transceivers generate a user clock from the reference clock to use it in data processing and control logic of the FPGA.

The interface to the downstream systems, Ethernet or Aurora, runs with an independent 125 MHz oscillator as a reference clock. The 125 MHz clock is also used in the system-wide timer for measurement of time intervals: for example, in the rate measurements and artificial trigger generation.

Belle II Setup

Figure 8.26 shows the layout of clock distribution in the DHE at Belle II. In the DHE, the UCF link provides the clock, which is guaranteed to be phase aligned to the DHC clock. The clock is recovered from the UCF data stream and has frequency 127.21 MHz.

Because period of the input clock is not an integer multiple of the GCK, Si5338 cannot synthesize GCK without phase ambiguity to the input clock. Therefore, I designed a method to synthesize the 76.326 MHz clock with no phase ambiguity in FPGA.

For this method we first synthesize a 3xB2TT clock in the clock manager of the FPGA from the UCF clock. This clock is phase synchronous to the UCF clock. Next, this clock is connected to the input of the BUFR primitive. The BUFR is configured for division of the input clock by factor 5. This gives us the clock with frequency of the GCK, but not be yet aligned to the UCF clock. Finally, we need a synchronization signal from the DHC, sent over a UCF link. The synchronization signal, which is phase aligned to the UCF clock, clears the divider counter in the BUFR asynchronously. This synchronizes the output

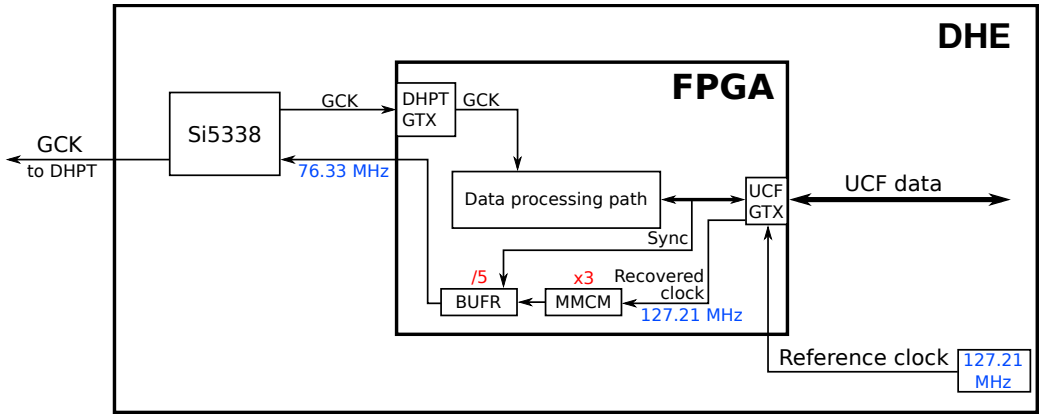


Figure 8.26.: Clock distribution in the DHE system for Belle II

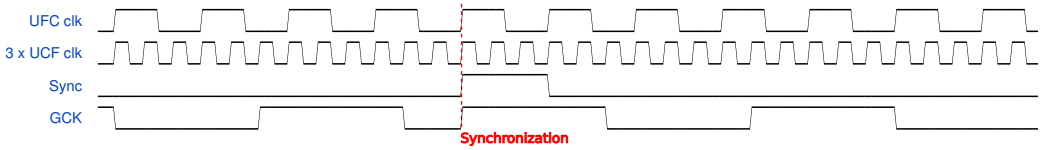


Figure 8.27.: Waveform of the GCK synchronization to the UCF clock

clock of the BUFR to the phase of the UCF clock. Figure 8.27 shows the waveform of the synchronization process.

The output of the BUFR drives Si5338 that operates in the jitter cleaner mode. Because the input and output clocks have the same frequency, the PLL operates in the synchronous mode and preserves phase of the input clock. Therefore, the final GCK is also synchronized to the UCF clock and to the B2TT clock.

The DHI uses an FPGA from a different family than the DHE. Nevertheless, clock processing in the DHI works in similar way as in the DHE.

8.5. Data Handling Engine

The data handling engine is part of the DHH system that communicates directly with the front-end electronics on the half-ladder. The DHE cards provide command, data, and slow control interfaces to the detector, data and trigger processing logic, and an outgoing data link to data acquisition systems.

8.5.1. Standalone System for Laboratory Setups and Beam Tests

The system development started with the standalone system, which is designed to assist detector development and characterization. Therefore, it implements full data acquisition chain in a single module. This system was also used for studying communication inter-

faces to the front-end ASICs and defining requirements for the final system, which will be installed in Belle II.

The standalone system is designed as a mobile system. The system can be easy connected to a conventional PC over two independent Ethernet interfaces: one for data, and one for the slow control. The system can run with internal or external trigger source. Internal trigger source is used for detector development in laboratory setups. External trigger source synchronizes with other detectors at beam test setups.

Firmware for the standalone system consists of trigger processing logic, detector control interface, and data processing chain. The trigger processing logic receives triggers from outside or generates artificial triggers with configurable rate. The detector control interface generates synchronous commands for the DHPT. The data processing chain receives and combines detector data frames to build events. Events are sent out over Ethernet to a PC or over an Aurora channel to the ONSEN system.

Trigger Processing Logic

Trigger is a signal that initiates event processing. The trigger processing logic in the DHE is separated in trigger receivers and trigger processor.

Figure 8.28 shows the layout of the trigger system in the DHE. Two types of trigger signals are available in the system: external trigger and auxiliary trigger. Trigger signals contain a trigger pulse, which defines timing of the trigger, and a unique event number. Both types of trigger signals are connected to the trigger processor, which generates commands for the front-end electronics and the event builder core. Selection of trigger type is done via slow control.

External trigger sources are the TLU receiver, which is used with the EUDET telescope at beam tests, and the B2TT receiver for Belle II trigger and timing system [79]. Electrically, they use the same IO lines. Selection of the external trigger source is done at the synthesis stage.

Auxiliary trigger sources are IPbus trigger, artificial trigger core, and trigger core of the current source scan. Auxiliary trigger sources are connected in a chain and follow these rules. IPbus trigger, generated by the slow control, is multiplexed with the current source scan trigger. During scans, IPbus trigger is disabled to allow for uninterrupted scan. The output of the current source scan core is the input to the artificial trigger core. Triggers, generated in this core and by the current source scan, are forwarded to the trigger processor. The multiplexer selects event number from the source that generated the trigger pulse.

TLU Receiver The TLU core that we use in the DHE was developed in the University of Bonn for the DHH emulator, the predecessor read-out system of the DHE. The core has two parameters: division factor of the TLU clock and event number length. These parameters are configured via slow control.

Clock division parameter defines time between trigger is generated and the full event number is received. This time may be in the order of the DEPFET read-out time. Therefore, the core generates two signals: a trigger pulse immediately upon receiving the trigger

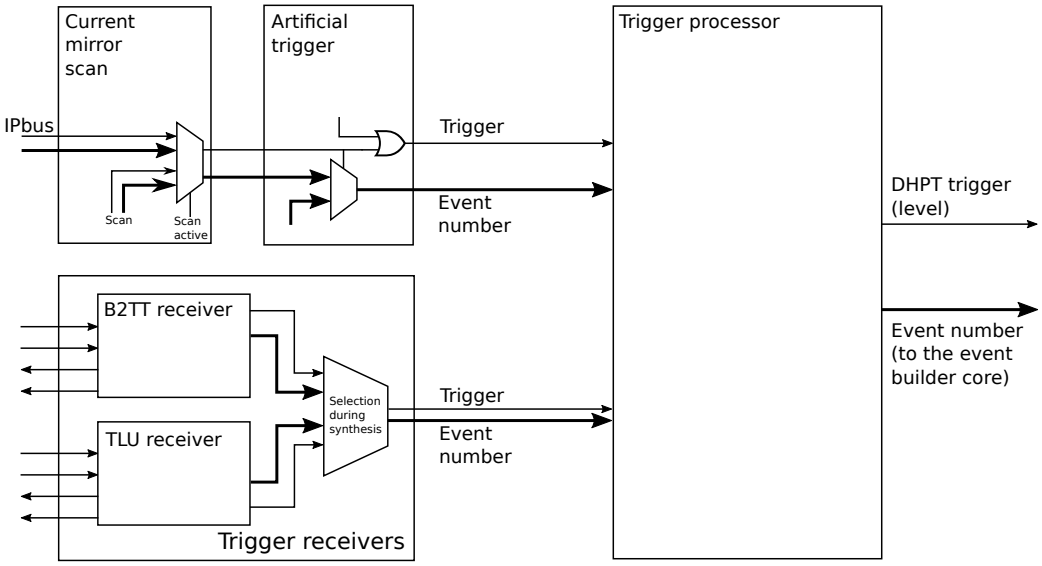


Figure 8.28.: Layout of the trigger system in the DHE

signal, and another upon receiving full event number. As a consequence, the DHE starts to read data by the first signal. In the mean time, data are buffered in the FIFO of the input links. Only after receiving full event number, the DHE starts to process the event.

It is important to mention, that TLU interface uses DC-coupled LVDS signals. Therefore, the interface is not compatible with the DHH AMC cards and VME carrier boards on the hardware level. User has to replace capacitors with 0 Ohm resistors on each line of the trigger interface in order to make DHE compatible with TLU. The cable between TLU and the carrier board must be a cross-over cable.

B2TT Receiver B2TT sends data as a synchronous 8b/10b encoded 254 Mbps data stream divided into 2560-bit frames. Each frame corresponds to a revolution cycle of the SuperKEKB accelerator. A frame consists of 16 160-bit packets. Triggers are transmitted as an 8b/10b code word with the most significant bit set to '1'. Decoded value of the trigger code gives precise time of the trigger and its type: 3 bits represent position of the trigger within the 5 clock cycles of the code word, 4 bits represent the trigger type.

Another function of B2TT is synchronization of read-out electronics. System distributes information about current run and experiment numbers, synchronizes absolute time, and resets electronics. B2TT also provides a backchannel that allows us to throttle trigger link, if the read-out system cannot accept any more triggers.

We use B2TT receiver core provided by the KEK group for receiving B2TT messages. Because data encoding embeds clock into data stream, we extract clock from B2TT data stream in the external clock recovery chip on the AMC card. The clock recovery chip outputs data and clock signals to FPGA. This improves quality of data signal and provides clock signal for the receiver core that is phase aligned to data. The receiver core decodes code words and supplies decoded information to the firmware.

IPbus Trigger Users can generate triggers from slow control. It is implemented as a register mapped into address space of the slow control IPbus core described in subsection 8.8.1. Every write access on this register generates a trigger. The value, written into the register, becomes the event number associated with the trigger.

There are few disadvantages of using this trigger source. Performance of the PC limits maximum trigger frequency. Generation of triggers may interfere with slow control programs, if they run as a different software process. Also, IPbus transaction frames may be lost in the network. These problems lead to the development of the artificial trigger core.

Artificial Trigger Core The artificial trigger core generates triggers in the firmware of the DHE. The core has following parameters: trigger frequency, number of triggers, first event number. The core consists of a final state machine that controls trigger generation and a timer.

Core parameters are programmed before trigger generation is activated. Final state machine runs with the system clock, the GCK. The timer runs with an independent clock with known frequency. The timer divides input clock and generates pulses with a desired period. If the core is activated, then a trigger pulse is generated for every pulse of the timer. The core increments event number for every timer pulse.

The core asserts busy signal while it processes triggers. This signal is connected to slow control. After the core succeeded to generate required number of triggers, it goes into the idle mode. User can then re-activate the core with different parameters.

Trigger for Current Source Scans The current source scan core is designed to increase speed of DCD calibration using the current source integrated into the DHE. The core implements full scan logic in the firmware. This removes communication overhead of the slow control system, which adds up with every trigger.

The core takes full control over trigger generation and operation of the current source. Before the core is activated, user has to set range, step size, settling delay for the DAC, and number of event to be recorded in every step.

The core starts with programming desired current value into the current source over the SPI interface. Once the value is programmed, the core waits for settling time before first trigger is generated. The core then generates a trigger pulse and increments event number. In addition, current value is sent to the event builder core, which merges this information into the outgoing data stream. The core then waits for a handshake from the event builder core. The handshake indicates that the core has finished processing current event. The scan core then generates a new trigger or increments the DAC value, if the necessary number of events has been generated for current DAC value. The scan finishes when DAC value goes out of scan range and the core goes into the idle state.

Trigger Processor The trigger processor receives trigger information, which is used to generate read command for detector front-end. The core synchronizes detector operation in respect to the external reset signal and provides trigger identification information for the event builder core.

7	6	5	4	3	2	1	0
RST	$\overline{RST}$	$VETO$	$\overline{VETO}$	TRG	$\overline{TRG}$	$FSYNC$	$\overline{FSYNC}$

Figure 8.29.: Control command

1	1	1	0	0	0	$FSYNC$	$\overline{FSYNC}$

Figure 8.30.: Memory dump command

0	0	0	1	1	1	$FSYNC$	$\overline{FSYNC}$

Figure 8.31.: Idle command

The core can receive trigger pulse and an event number from one of two sources: an artificial trigger or an external trigger interface. The actual trigger source is selected by slow control. Upon receiving trigger signal, the core generates read command for detector front-end. The read command is asserted for a pre-defined time period configured by slow control. Length of the read command defines number of detector gates, or 4 detector rows, which front-end will send to the DHE. Because GCK is 8 times faster than gate switching frequency, length of the read command can be calculated using the expression

$$N_{\text{GCK, cycles}} = N_{\text{gates}} * 8 - 7. \tag{8.3}$$

Reduction by 7 GCK cycles is required due to operation conditions of the detector control interface described below.

The core generates periodical pulses, frame sync, that synchronize running shutter mechanism in the detector front-end. The core initially synchronizes to an external reset signal and then operates by counting number of clock cycles elapsed since last sync. Time period is configured by slow control in GCK cycles.

For data processing, the core memorizes frame number of the last received detector frame and gate number that is active when trigger arrives. Because DHE and DHPT clocks are synchronous, the core counts active gates in respect to frame sync. This information is sent to the event builder core, which merges it into the data stream. One should take into account that active gate information in data may have a fixed offset to the active gate of the running shutter in the detector. This offset is a configuration parameter of the front-end and is programmable via slow control.

Detector Control Interface

DHE operates detector front-end electronics using serial command interface. Control signals over the serial command interface are encoded as 8-bit words using broken Manchester encoding. Data words are sent synchronously with the GCK.

Manchester encoding provides DC-balanced signals by having equal number of '1's and '0's for any combination of control signals. Figures 8.29-8.31 show format of the Manchester-encoded control words.

Reset signal or **RST** resets the DHPT ASIC. There are two types of reset signals – **short** and **long** – that differ by the length of the command. Short reset is sent in a single control

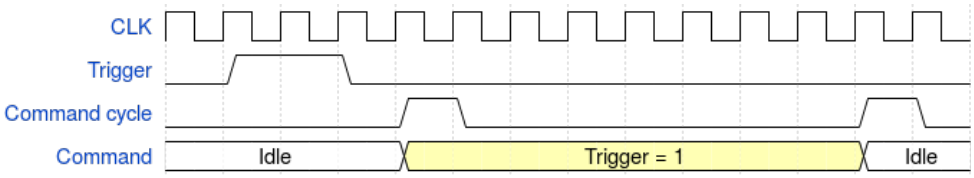


Figure 8.32.: Trigger signal is sent as one word

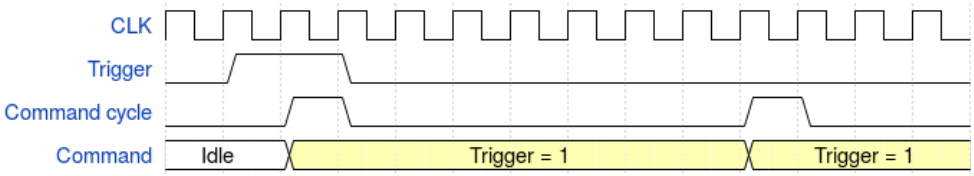


Figure 8.33.: Trigger signal is sent as two words

word. Short reset resets only logic, responsible for data transmission between DHPT and DHE. This reset type is used for recovering high speed links. Long reset is a repetition of short resets for few consecutive commands. Long reset resets all DHPT logic except configuration memory. Control logic in the DHE takes care of the timing required for generation of the corresponding reset commands. Both resets are issued by slow control in the standalone system.

The **VETO** signal activates logic in the DHPT that is responsible for gated mode operation of the detector. The signal can be controlled by slow control or by a sequencer described in subsection 8.8.2.

The trigger signal or the **TRG** is the read command generated by the trigger processor core. The signal enables data processing logic in the DHPT.

The frame sync signal or the **FSYNC** defines start of detector frame and resynchronizes running shutter. The signal is also generated by the trigger processor core.

The memory dump command forces DHPT to send out content of the memory without any further data processing.

The idle character is used to define boundaries of the command word. The character is only sent if all control signals are deasserted.

Sending a control word takes 8 clock cycles. This defines time precision of the control interface. Control signals are resynchronized to command word period by memorizing all asserted control signals during one command period and clearing them at the beginning of the next period. Therefore, a control signal has to be kept in high state for **N * 8 - 7** cycles to be sent in exactly N control words. Otherwise, number of control words, in which the signal is sent, depends on the exact timing of the signal. Figure 8.32 shows a situation when a trigger signal is asserted for two cycles within one command. The signal is sent as one command in this case, regardless of the signal width. Figure 8.33 shows a situation when a trigger signal is asserted for two cycles within two commands. The signal is sent as two consecutive commands.

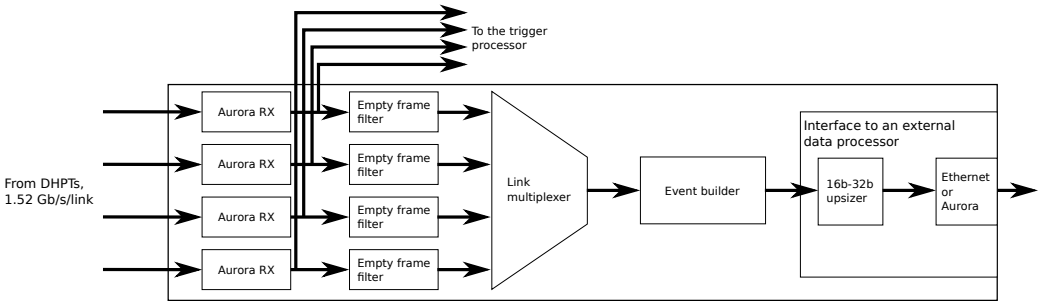


Figure 8.34.: Layout of the data processing chain in the DHE for the laboratory setups

Data Processing Chain

Data processing chain in the DHE for laboratory setups forwards detector data to data acquisition without modification. The data processing chain consists of receiving data, adding event information and headers to data, and sending data to data acquisition: ONSEN system or PC. Figure 8.34 shows the layout of the data processing chain. Data are received by 4 unidirectional Aurora cores. Data are filtered from empty frames. Filtered data streams are multiplexed on a frame-by-frame basis into one stream that is the input to the event builder. The event builder assembles an event by collecting data that belong to the same trigger. The core also adds service information to an event. The event builder sends finished events to the external data processor: over Aurora link to the ONSEN system or over Ethernet to a PC.

Data Receiver Data are received by the data receiver. The data receiver includes 4 unidirectional Aurora receiver cores, 4 empty frame filters, and a stream multiplexer.

The Aurora receiver core uses a simplex RX Aurora core with 16-bit wide framing interface without backchannel [64]. The core takes over low-level link operation, link synchronization, and data decoding.

Frames, received by the Aurora core, are sent to the empty frame filter. The empty frame filter removes frames that are sent by the front-end electronics, if no read command is sent during last detector integration period or no hits pass zero suppression. Empty frames carry only frame header and frame number and are discarded. The data frames, which pass the filter, are written into a large FIFO which has capacity to store the largest possible memory dump frames. Frames are stored in the FIFO until they are read by the event builder.

All data including empty frames are also sent to the trigger processor core. The trigger processor core extracts frame number from data and forwards it to the event builder core to be included in the event information.

The stream multiplexer combines four data streams into one stream. Streams are combined frame-by-frame using following algorithm. A frame from a stream is read until the end of the frame is reached or until no data are received for a predefined time interval. If

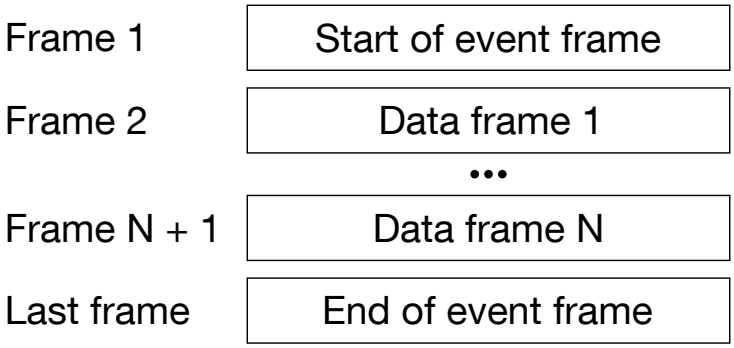


Figure 8.35.: DHE event structure

one of the above conditions happen or if there are no data to read, the multiplexer switches to the next stream.

Event Builder Event builder merges frames from four data streams, which belong to the same trigger, into a single event. Figure 8.35 shows the event structure. An event starts with the start of event frame, followed by an arbitrary number of data frames, and finishes with the end of event frame. The start of event frame contains full timing, status information about the event, and full event number. Data frames contain detector data with added headers that identify event. The headers contain data type, link ID, and lower half of the event number. The end of event frame contains summary of the event such as event size and error conditions that appeared during processing of this event. The event builder core calculates a 32 bit CRC from the frame's payload. This word is appended at the end of each frame and is used to detect data corruption. Full description of the data format is given in [80].

Processing of the event starts upon receiving trigger signal. At the same time, the event builder receives event number and event timing. With this information, the event builder generates start of event frame. The event builder core waits for detector data to arrive. If no data are received over a configurable period, a special frame, the **ghost frame**, is generated. The **ghost frame** indicates absence of detector's data in the event. The **ghost frame** is always followed by the end of event frame.

If detector's data are received, data frames are complemented with the headers. Data frames, which are not aligned to 32-bit words, are padded using the last start-of-row word in the data frame. We chose a start-of-row word as a padding word because it does not add information about hits and is easily identifiable in data as a repetition of the previous start-of-row word. Data frames are read until no more data are received for a configurable period. The event is then completed by generating an end of event frame.

The usage of the timeout for event separation limits trigger and data rate firmware capability. The timeout shall be longer than one frame read-out time and maximum time to process data. For example, the detector sends data over 4 DHPT links. With the length of the detector's integration cycle and the trigger command set to 20 us, the minimum dead time between triggers is 120 us for the average detector occupancy of 3 %.

Interface to an External Data Processor Full events are transferred to the external data processor. Before data are sent to the external interface, width of the data interface is changed from 16 to 32 bits in the upsizer core. The external data processor is the ONSEN system, the read-out PC, or both. The interface to the ONSEN system is an optical bidirectional Aurora link with flow control. Aurora link can operate at 3.125 or 6.25 Gb/s line rates. The interface to the read-out PC is a unidirectional UDP [13] connection over the Ethernet network. Selection of the external interface is done during the firmware synthesis.

For UDP communication we use the 1G Ethernet UDP/IP stack by A. Fiergolski and P. Fall and a custom frame splitter core [81]. The UDP core implements UDP, IP [14], and ARP [15] protocols. Destination IP address and UDP port are programmed by slow control.

Because the maximum Ethernet frame size is limited by the maximum transmission unit, the MTU, in the network hardware, frames, which exceed the MTU, are split into smaller chunks. The DHE uses default MTU for Ethernet networks of 1500 bytes. The splitting is done by the frame splitter core. The frame splitter core adds a header to the all chunks except for the first one. The header indicates separation into chunks and provides sequential chunk ID for re-assembly of the frame at the destination. Appendix E describes the chunk format.

8.5.2. DHE System for Belle II

DHE system for the pixel detector at Belle II consists of 40 DHE modules. Because modules need to operate synchronously, all modules are synchronized to the same external clock and have only one common trigger source. Because triggers are distributed in time according to the Poisson distribution, minimum time between triggers at the designed trigger rate of 30 kHz is less than the integration time of the pixel detector. Therefore, a new data processing chain for the read-out system is needed for a dead-time free operation. The DHE also implements the cluster reconstruction algorithm. The cluster reconstruction algorithm is used to classify clusters generated by low momentum pions using energy loss in the detector. Clustering algorithm allows us to identify and mark these tracks to avoid losing them in online data reduction.

Synchronization Channel

At Belle II, DHC synchronizes up to five DHE modules. Synchronization is done using the advanced operational mode of the high-speed serial links in the Virtex-6 FPGA with a custom Unified Communication Framework or the UCF [82]. A UCF synchronization channel operates at the line rate 2.544 Gb/s and transmits trigger information, slow control and data. Section 8.7 gives detailed description of the framework.

The synchronization channel transmits real-time control signals with fixed latency between DHC and DHE. Real-time control signals are trigger pulse, trigger type, trigger

[13] User Datagram Protocol
[14] Internet Protocol
[15] Address Resolution Protocol

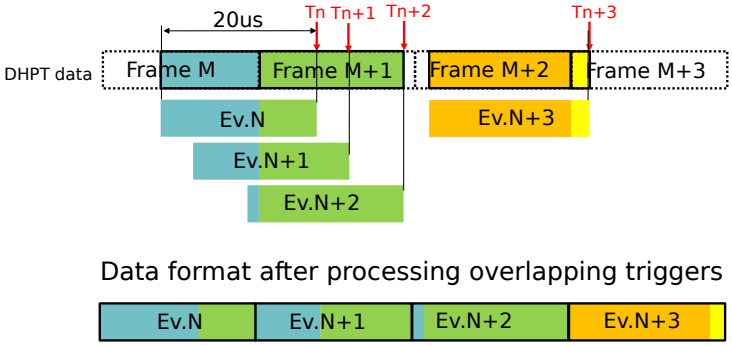


Figure 8.36.: Read-out of overlapping triggers

time stamp, trigger number, injection veto, reset, and accelerator synchronization signal. Appendix G describes format of the trigger frames.

Data Processing Chain with Overlapping Trigger Support

Data processing chain for the pixel detector at Belle II must operate at 30 kHz trigger rate without causing dead time. Triggers may overlap in time, if a new trigger is received while processing of the new trigger is not finished. Figure 8.36 shows this situation. Trigger reads hits that correspond to data integrated by the matrix over the previous cycle of 20 μs. A new trigger, received during the read-out of the previous trigger, shares part of data with the previous trigger. Because front-end sends data only once, the data processing chain in the DHE has to copy hits from the overlapping regions into several events. This causes output data rate to increase by factor 1.2.

The data processing chain used for the laboratory setups requires separation in time between triggers for event building. Therefore, a new data processing chain has been developed that satisfies requirements of the experiment.

Figure 8.37 shows layout of the data processing chain. Main element of the new data chain is dual-port memory, implemented as block RAM or external DDR3 memory. Memory is divided into four banks by the number of the front-end streams. Each bank is used as a ring buffers. Received data are stripped of headers and written as a continuous data stream to a memory bank. The trigger processor calculates expected start and end of event in row coordinates of the detector matrix. The frame receiver receives data, stores them to memory, and calculates memory pointers in row coordinates for the start- and the end-of-event. Pointers are transmitted to the event creators that assemble an event from data, stored in memory banks.

Hits in events are re-mapped to restore geometrical hits coordinates that are scrambled by routing of the drain lines on the detector. The remapped hits are sent through the clustering engine.

The clustering engine recovers clusters: hits with adjacent coordinates that are formed by tracks, which hit more than one pixel. If one may assume that connected hits are produced by the same track, cluster reconstruction allows us to measure flight direction

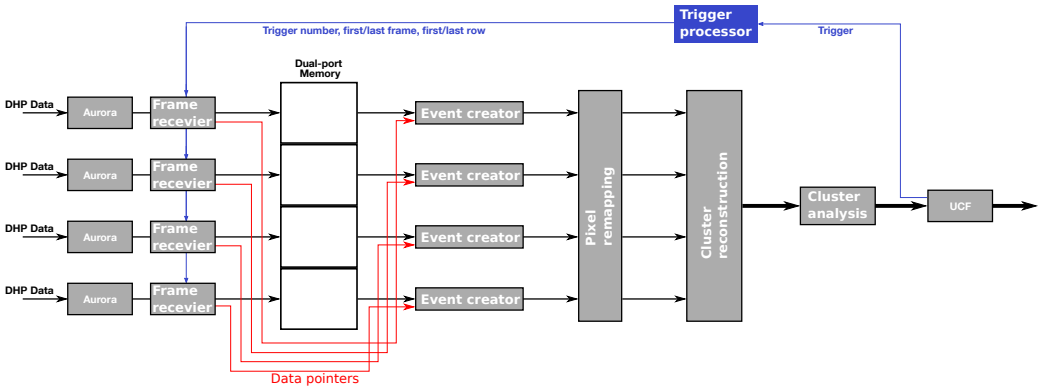


Figure 8.37.: Data processing chain in the DHE for Belle II

of the particle and its energy loss in the detector. Flight direction helps to discriminate between beam background particles, which would produce long tracks parallel to the beam direction, and decay products with high transverse momentum. Energy loss in the detector depends on particle's energy and can be used for particle identification.

Particle identification in the pixel detector plays the essential role in recovery of the low-momentum pions. Low-momentum pions are produced in the D^* decays. Due to their low momentum, magnetic field inside of Belle II confines these pions to the vertex detector volume. This prevents them from leaving hits in the outer detectors. Therefore, the high-level trigger cannot produce ROIs for these hits: the hits would be rejected by the ONSEN system and would lead to low reconstruction efficiency of the D decays.

Shape and amplitude sum of the reconstructed clusters are analyzed as an aid to the online data reduction system. The cluster analysis based on a machine learning algorithm is described in [83]. Analyzed clusters are then sent to the DHC over the UCF stream.

Alternatively, data can bypass cluster reconstruction and analysis. In this case, data are sent to the DHC after pixel remapping.

Following sections describe all data processing cores, except for the pixel remapping core. Design of the pixel remapping core is not finalized yet and, therefore, is not discussed here.

Frame Receiver Figure 8.38 shows the layout of the frame receiver. The frame receiver consists of four cores: frame, start, end, and merge state machines.

The frame FSM[16] receives detector data from the Aurora core. The FSM removes empty frames and frame headers from data stream and decodes data in frame number and row coordinate. Remaining data are written to the dual-port memory. The frame FSM distributes current memory address and decoded coordinates to the start and end state machines.

The start and end FSMs calculate pointers to the start and to the end of event. This requires knowledge of the event boundaries. The trigger processor calculates event bound-

[16] Finite-state machine

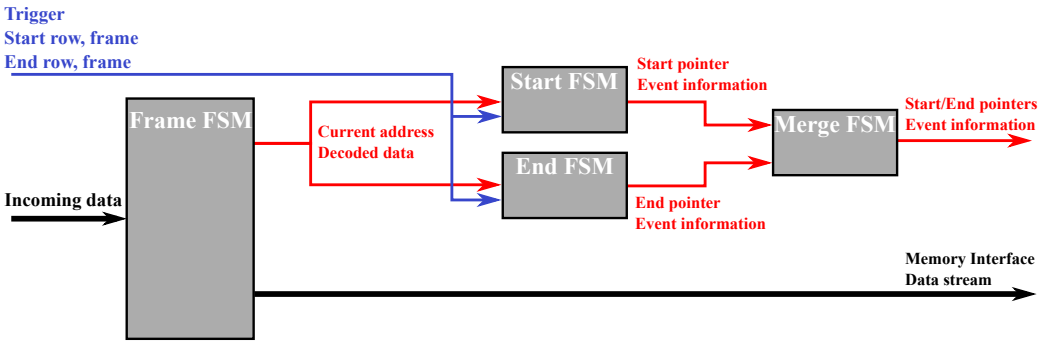


Figure 8.38.: Layout of the frame receiver

aries using start frame number, start row number, end frame number, and end row number. The start and end state machines store this information into an input FIFO.

Event processing starts with reading event boundaries from the input FIFO. As first data word of the event is processed by the frame FSM, the start FSM writes pointer to this data and event information to the output FIFO. Event information consists of event number, frame number of the first frame, data type, and error flags.

The end FSM works similar to the start FSM. As frame FSM receives data outside event boundaries, the end FSM writes a pointer to the last data inside the event and event information to the output FIFO.

Both FSMs check data for consistency to protect data processing chain from erroneous data. For example, if no data are received from the front-end for a predefined period, the error flag in the event information is raised to indicate this condition.

The start and the end FSM operate independently ensuring overlapping trigger functionality. Once the start or end condition for an event is met, the corresponding FSM reads event boundaries for the next event from the input FIFO. Number of overlapping triggers per event is only limited by depth of the FIFO.

The merge FSM merges pointers and event information from the start and the end state machines. The merge FSM reads data from the output FIFOs and checks synchronization between the start and the end FSMs by comparing event numbers in the event information. If both event numbers match, data are merged and sent to the corresponding event creator. If synchronization is broken, an error flag is raised in the information sent to the event creator. The merge FSM then tries to restore synchronization by reading out only the FIFO with the smallest event number until event numbers match again.

Event Creator The event creator cores assemble event frames separately for each DHPT from data stored in the dual-port memory. The core receives data pointers and event information from the merge FSM. Using event information, the core generates headers according to the DHE data format [80]. The core then reads data stored between the start and the end pointers from the memory. After finishing reading data, the core appends a 32 bit checksum calculated from the headers and the payload to the end of the frame. The frame is then transmitted to the event builder.

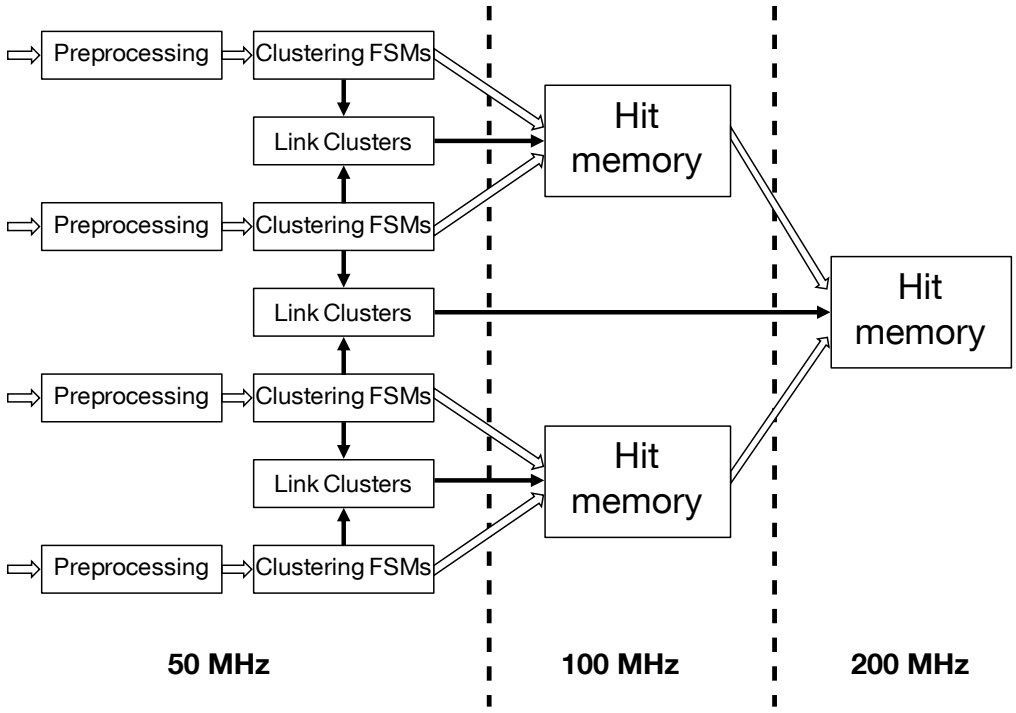


Figure 8.39.: Data flow in the cluster reconstruction algorithm by I. Konorov

Cluster Reconstruction Algorithm

Cluster reconstruction algorithm finds adjacent hits and connects them by assigning a common ID to these hits, the cluster ID. There are two possible reconstruction topologies: each of the possible 8 direct neighbours or only non-diagonal neighbours can be connected to the hit. The clustering algorithm implemented in the DHE connects only non-diagonal neighbours to a hit. A diagonal neighbour, which is not connected to the hit through another hits, is considered an unrelated cluster.

Cluster reconstruction is implemented as a pipeline algorithm. Figure 8.39 shows data flow in the algorithm. First, coordinates of the hits are decoded from data and clusters are reconstructed in four data streams independently. Clusters with adjacent hits in neighboring data streams are linked to the same cluster ID. In the next step, pairs of neighbouring streams are merged in the memories. Finally, clusters in two remaining streams are merged into an output stream. The operational frequency doubles in every step to account for increase in data rate due to aggregation of the data streams.

Figure 8.40 shows the layout of cluster reconstruction for a data stream. The cluster reconstruction algorithm consists of data pre-processing, 32 finite-state machines, cluster ID update bus, memories for cluster ID remapping and hit storage, and event re-assembly stage.

The pre-processing module receives zero-suppressed data, decodes them, and sends hit coordinate to the FSMs and to the hit memory. The FSMs assign a cluster ID to a hit and

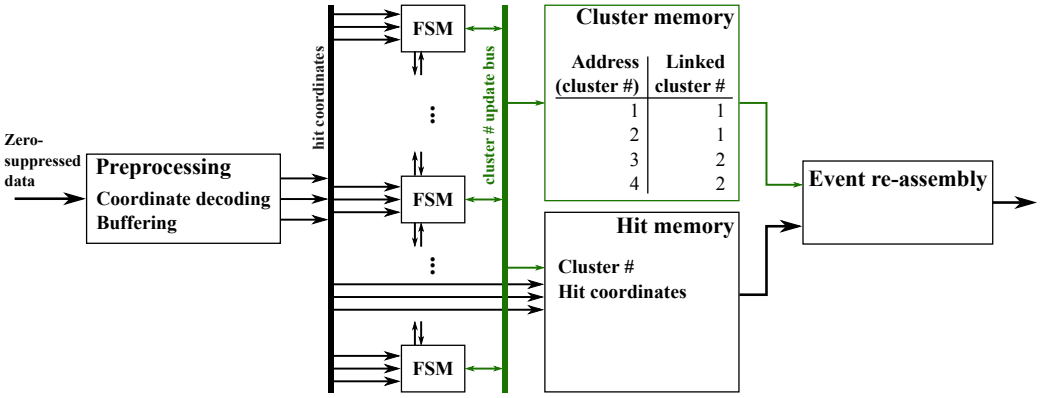


Figure 8.40.: Algorithm for cluster reconstruction

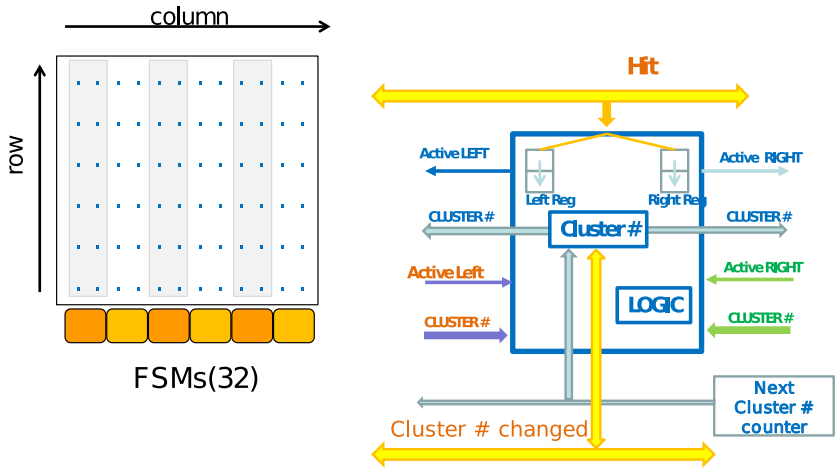


Figure 8.41.: Layout of cluster reconstruction and clustering FSM by I. Konorov

send this information to the cluster ID bus and to their direct neighbours. The cluster ID bus broadcasts change of cluster ID to all FSMs and to the cluster memory. In addition, the next free cluster ID is broadcasted on the bus. The event re-assembly core performs remapping of cluster ID for connected clusters, reads hits from the hit memory, and re-assembles the event.

The clustering FSM is the central element of the algorithm. Figure 8.41 shows the structure of the FSM. The FSM tracks the state of two detector columns. Upon receiving a new hit, the FSM checks the state of the neighbour columns either in the same FSM or in the neighbour FSM. If there are no active neighbour columns, the FSM assigns next cluster ID to a hit, becomes active, and broadcasts this change on the cluster ID bus. If the neighbour column is active, the FSM assigns cluster ID from the active column to the hit and becomes active. The FSM becomes inactive if there is no hit in the following row.

If two neighbour columns are active, one hit merges two independent sub-clusters. The FSM has a special rule for this situation. The FSM compares cluster IDs of the sub-

clusters and assigns the smaller cluster ID to the hit. Then, the FSM writes the smaller cluster ID into the cluster memory at the address of the larger cluster ID, and broadcasts this change on the cluster ID bus. This action has two consequences. First, other active FSMs, which have the larger cluster ID, update their cluster ID to the lower one and use it for the following hits. Second, remapping of the cluster ID is done through a look-up to the cluster memory.

Once all data in the event are processed, the event re-assembly starts. To avoid using search algorithm every time when a hit is processed, the cluster memory needs to be updated to point to the final cluster ID. If one starts the update from the smallest cluster ID in the table, then this procedure requires at most two read cycles and one write cycle of the cluster memory for every sub-cluster. First read retrieves the ID of neighbouring sub-cluster, which is always equal or smaller than the ID of the current sub-cluster. Previous statement is always true because of the sub-cluster merging algorithm in the FSM. Second read at the address of the neighbouring sub-cluster retrieves final cluster ID because this ID has been already updated. Final cluster ID is written at the address of the current sub-cluster and is used to remap hits that belong to this sub-cluster.

Example

Figure 8.42 shows an example of the hit pattern reconstruction and the cluster memory content. The algorithm reconstructs clusters from left to right and from top to bottom. The first hit in the first row is assigned the cluster ID **1**. The second hit in the first row is not connected to the first hit, therefore it is assigned the cluster ID **2**. The first hit in the second row is connected to the cluster **1**. This hit is assigned to the same cluster ID. The procedure continues until the fourth hit in the third row. Here, clusters **2** and **3** merge. Therefore, the value in the cluster memory at the address **3** is changed to **2**. Clusters **1** and **2** merge in the last row and the cluster memory is updated.

Finally, the cluster memory has to be updated. We start with cluster **2**. Read at the address **2** returns **1**. Read at the address **1** returns **1**. Nothing needs to be updated for cluster **2**. Next, we continue with cluster **3**. Read at the address **3** returns **2**. Read at the address **2** returns **1**, which, as we know, is the final cluster ID. We update value at the address **3** to **1**. Now, when we read hits that belong to sub-cluster **3**, a single look-up to the cluster memory is sufficient for remapping this hit to the final cluster ID.

Test of the Algorithm

We tested the cluster reconstruction algorithm in FPGA with randomly generated shapes [84]. Figure 8.43 shows the layout of the test system, used for testing the algorithm. The test setup consists of software, which generates clusters and reconstructs clusters as a reference implementation, and implementation of the algorithm in the FPGA board. Same data are processed in FPGA, where they are sent over an Ethernet link, and by the reference algorithm in the software. Evaluation software then compares results of the cluster reconstruction in FPGA and in software.

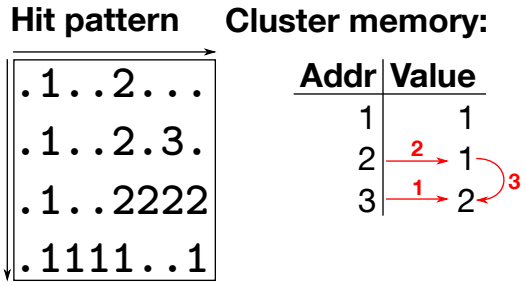


Figure 8.42.: Example of cluster reconstruction. Numbers in the hit pattern represent position of the hits with the corresponding cluster ID. Red numbers and arrows shows the order of memory update

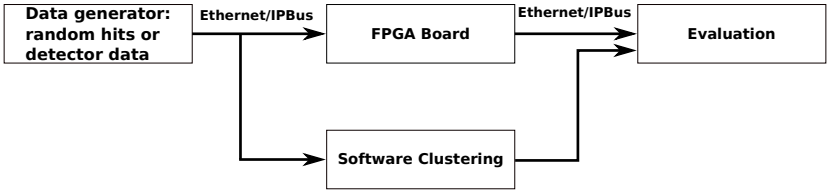


Figure 8.43.: Layout of the test system for the cluster algorithm

This method allowed us to thoroughly test cluster reconstruction algorithm in FPGA. The tests revealed problems in the initial implementation that were fixed in the final version of the core.

8.6. Data Handling Concentrator

Data handling concentrator is the second data processing layer in the DHH system. The full DHH system uses 8 DHC modules: one DHC module per ATCA carrier board.

DHC plays central role in the system architecture. The DHC collects data from 5 DHE modules, assembles them into subevents, and distributes subevents to 4 online selection nodes. Subevent assembly averages data rates on the outgoing links. The effect is maximized, if a DHC reads both inner and outer layer detectors. In addition, DHC sends a fraction of events over a UDP link to the local pixel detector DAQ for data quality monitoring.

DHC is also connected to B2TT and slow control Ethernet network. DHC distributes trigger and real-time control signals, which steer data read-out in DHEs, over a dedicated channel of the UCF link, and synchronizes DHE clock phases to the phase of the B2TT clock. The UCF master on the DHC implements Ethernet hub in the firmware that distributes Ethernet frames to DHEs and forwards Ethernet frames from DHEs to the network.

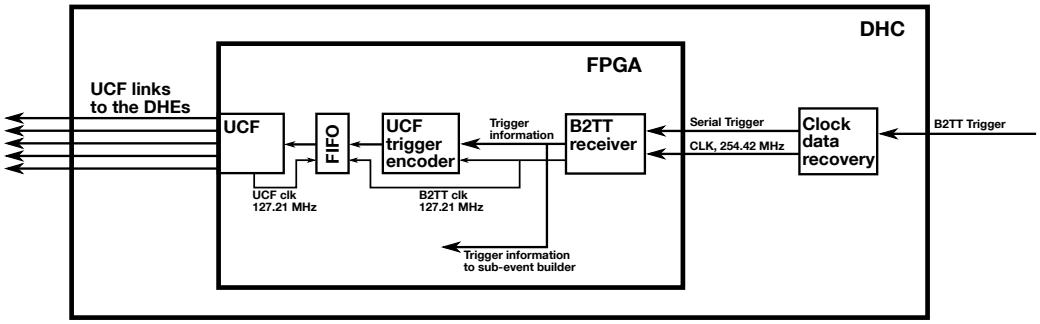


Figure 8.44.: Layout of the trigger processing in the DHC

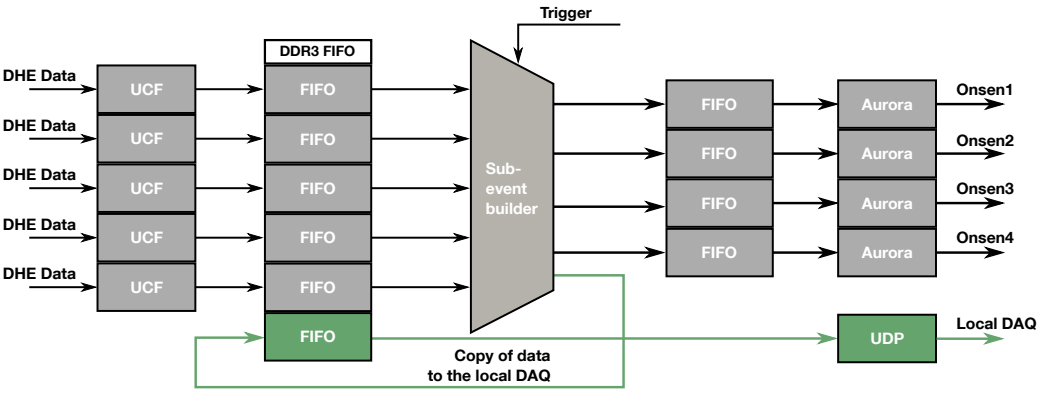


Figure 8.45.: Layout of the data processing in the DHC

8.6.1. Trigger Processing

The only source of trigger information in the DHC is the B2TT link. DHC receives serial trigger signal, decodes it, and distributes it to DHE modules. Figure 8.44 shows the layout of the trigger processing in the DHC.

DHC receives and decodes trigger information similar to the B2TT receiver in the DHE as described in section 8.5.1. Recovered trigger information is written with the RF clock to the UCF trigger encoder and to the subevent builder.

The UCF trigger encoder synchronizes trigger information to the running shutter of the pixel detector. This process proceeds in a three-gate cycle so that phases of the RF and the GCK clocks align at the end of the cycle. One gate of the running shutter takes 13.3 cycles of the RF clock. Trigger encoder subdivides three-gate cycle in three periods of the length 12, 14, and 14 clock cycles, which correspond approximately to one gate. Trigger is then sent once per period and arrives at the DHE during correct gate. UCF distributes trigger information to 5 DHE modules over high-speed links through a stream that guarantees fixed latency for data transmission.

8.6.2. Data Processing

Figure 8.45 shows the layout of the data processing chain in the DHC. DHE data are received over UCF links. Received frames are buffered in large FIFOs, implemented using external DDR3 memory. The subevent builder core assembles subevents by combining all DHE events with the same event number. Complete subevents are then stored in one of the four output FIFOs. Each FIFO is read by an ONSEN link. In addition, a portion of subevents are copied to the sixth FIFO in DDR3 memory. This FIFO is read by the UDP core that sends data to the local DAQ over an Ethernet link.

FIFO Implementation in External Memory

The large FIFO, implemented in the external memory, is an important element of the subevent builder algorithm. FIFO buffers DHE events before they are processed by the subevent builder.

The FIFO core is built around the memory interface generator core, the MIG [85]. The MIG core takes over low-level control of the DDR3 memory and provisions transaction-based interface to the user. The throughput of the DDR3 memory with the MIG core is measured at 3 GB/s.

Figure 8.46 shows the layout of the FIFO core. The core supports up to 16 FIFO instances. Data flow is optimized for throughput and performance. Every four FIFO interfaces are multiplexed into a memory stream. Four memory streams share a single memory port. Address space of the memory is divided into 4 regions called banks. One bank stores information, which belong to four FIFOs. FIFO interface runs with the slower clock, 125 MHz, while the memory streams and the memory port run with 200 MHz.

First, 32-bit data words are encoded in 33-bit words. The most significant bit of the word indicates whether the word is a data or a control word. Control words define start and end of a frame.

Up to 14 33-bit words and the service information are then combined to a 256-bit word to match the width of the MIG user interface. Service information carries port ID and a count of the 33-bit words. Four FIFO streams are merged into a single memory write stream by a multiplexer. The multiplexer reads two 256-bit words, which belong to a single memory transaction, from a FIFO stream in a round-robin manner and stores them in a FIFO, which is read by a memory write stream. Memory write streams read data from the FIFO, write them to the corresponding memory bank, and update the write pointer for the bank.

Four write and four read streams share interface to the memory. Arbitration between streams is implemented as round-robin algorithm for write and read streams separately. Read streams have lower priority than write streams to prevent data loss. Read streams are allowed to read memory only, if there are data in the bank, indicated by the difference between read and write pointer for the bank, and write streams are idle.

Data, read from the memory, are written to the correct FIFO stream as 33-bit words according to its stream ID. In the next step, data are decoded into a 32-bit stream by recovering frame information from control words.

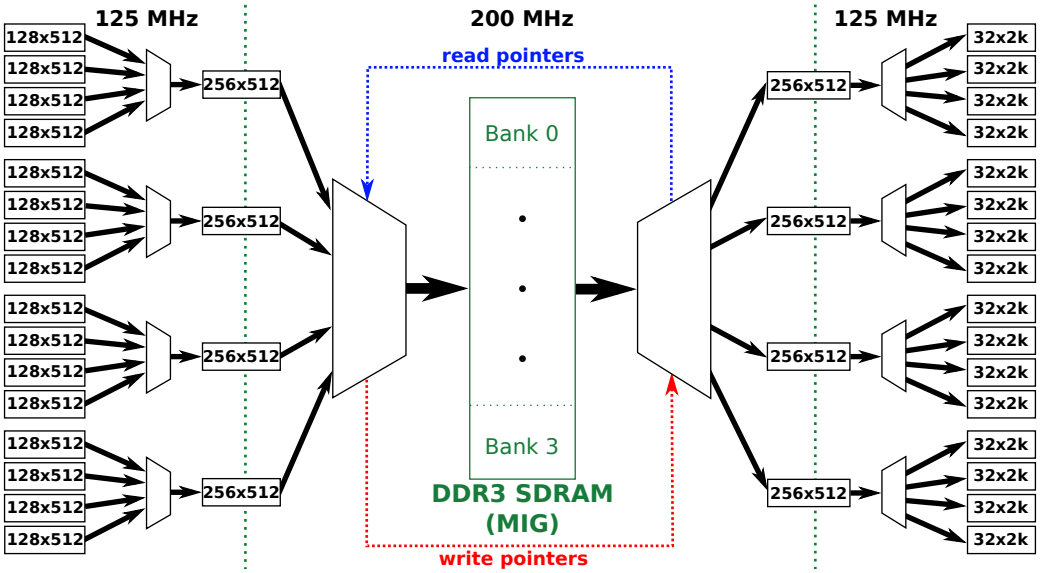


Figure 8.46.: Data flow in the DDR3 FIFO core

Sub-Event Builder Algorithm

The subevent builder in the DHC combines events with the same event number from up to 5 DHEs. Processing of a subevent begins with receiving trigger information from B2TT. Trigger information is stored in a FIFO in the subevent builder. The subevent builder reads trigger information from the FIFO and then reads DHE events, associated with this trigger, from the memory FIFO. The subevent builder writes complete subevents into one of the output FIFOs, where data are buffered before being sent to ONSEN.

The trigger information FIFO in the subevent builder regulates trigger reception in the DHC. If the FIFO is full, the subevent builder cannot accept new triggers and blocks trigger reception by raising BUSY flag in the B2TT interface. Once the fill level of the FIFO drops, the BUSY flag clears and DHC can receive new triggers again.

Upon reading new trigger information, subevent builder generates a start of subevent frame. This frame contains full event information: event number, trigger type, time tag, experiment number, and information about active DHE links. The state machine then reads FIFOs with DHE events in the order of increasing link index. FIFOs that correspond to inactive DHE links are skipped. The algorithm expects every DHE to generate an event in response to a trigger signal. The state machine reads full DHE event from a corresponding memory FIFO. If the event is not yet received, the algorithm blocks until data are available. Processing of an event starts with reading a start-of-event frame and ends with reading an end-of-event frame. The state machine checks event numbers in the DHE frames and compares them with the event number received from B2TT. Mismatch is reported to the slow control. After reading the end-of-event frame, the state machine reads event from the next DHE. After events from all DHEs are processed, the state machine

generates an end-of-subevent frame that summarizes information about the subevent. The cycle then repeats for the next event from the trigger FIFO.

Subevent Distribution

Subevents, which are distributed to the ONSEN system, follow the distribution pattern of the high-level trigger. Each ONSEN system receives high-level trigger for every forth event. This is done as a preparation for final event building, because subevents with the same event numbers will be sent only to pre-defined ONSEN nodes and not distributed randomly. In this case, the event building algorithm in the event builder 2 needs to merge two data streams instead of 8. The deterministic event distribution also offers an opportunity for an additional consistency check based on the source of events.

Subevent distribution is done in the subevent builder of the DHC. The subevent builder decides about the destination of the subevent based on the least significant bits of the event number. Mapping of the event number and the ONSEN links is stored in the look-up table. Frames, which belong to the subevent, are then written to the correct output FIFO.

This mechanism needs synchronization of the look-up table between the high-level trigger and the DHC. This is done by initializing the look-up table in the DHC over slow control.

8.7. Unified Communication Protocol

Serial high-speed links between DHE and DHC are implemented using the unified communication framework or the UCF. The UCF combines four link types, used in modern data acquisition systems: data, slow control, clock, and trigger distribution. These link types are aggregated into one physical high-speed link. Reduction in the number of high-speed links, used for communication between components of the data acquisition systems, reduces complexity of the hardware and the system's costs.

Design of the UCF originates in the SODA project and the switched enabling protocol developed for the PENeLOPE experiment [86, 87]. The principles that went into foundation of the framework are technology agnostic. Therefore, the algorithm can be implemented in FPGAs of different manufacturers than Xilinx with small adaptations for the hardware interface. The UCF is currently used in the PENeLOPE experiment, the COMPASS experiment, the NA64 experiment, and the pixel detector read-out in the Belle II experiment.

Figure 8.47 shows the layout of the UCF. The implementation consists of one master and one or more slave devices. The master device manages operation of the UCF links and is the source of the clock reference for the slaves. Data sent to slaves is encoded with 8b/10b encoding that embeds clock phase information into data stream. This allows slave devices to synchronize their clock to the reference clock of the master device.

The framework supports a star-like topology and a point-to-point topology. Figure 8.48 shows possible UCF topologies.

The star-like topology is used by systems with moderate data rates. The master is connected to multiple slaves with a single optical high-speed link through a passive optical

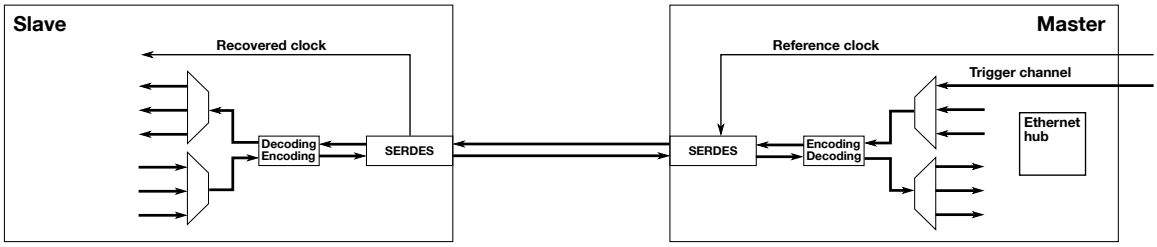
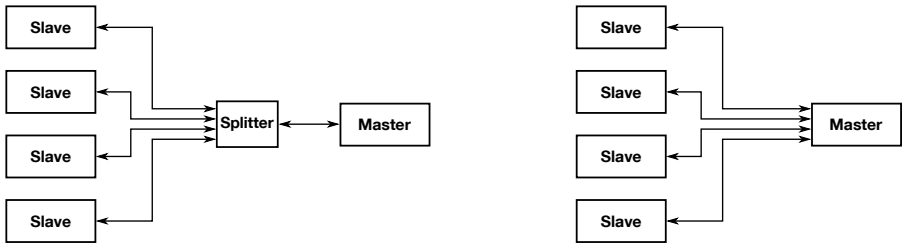


Figure 8.47.: Layout of the UCF. Only one link is shown



(a) Star-like topology (b) Point-to-point topology

Figure 8.48.: Possible UCF topologies

splitter. The splitter splits strong incoming signal into many outgoing signals to all slaves. Therefore, all slaves always receive the same data. Master controls send function of the slaves using time-division multiplexing algorithm to keep only one active slave at a time. The active slave activates its optical transmitter and sends data, while other slaves keep their transmitters deactivated. This topology is described in details in [82].

The topology used in the Belle II pixel detector read-out system is the point-to-point topology. This topology connects every slave to the master with a dedicated high-speed link. Therefore, full bandwidth of the high-speed link is available to every slave.

UCF offers multiple logical streams that are transmitted over a single high-speed link. The protocol provisions arbitration for the streams based on the stream's priority. As an optional feature, UCF implements an Ethernet hub. The hub is interfaced to the Ethernet network on the master FPGA on one side and to the UCF slave streams on the other side. The hub connects slaves to the Ethernet network.

8.7.1. Clock Phase Synchronization

Synchronization of the clock phases uses advanced mode of operation of the high-speed serial transceivers in the Virtex-6 FPGA. Clock synchronization requires high-quality reference clocks with the same frequency on the master and the slave sides. For the Virtex-6 FPGA, precision of the reference clock frequency must be better than 1000 ppm [88].

8b/10b Encoding and Clock Recovery

Clock synchronization relies on the recovery of the clock embedded in the 8b/10b encoded data stream. This is possible due to the properties of the 8b/10b encoding. A byte of data that consists of 8 bits is encoded with 10 bits. The increase in size by 2 bits increases the number of possible combinations by factor 4. Therefore, every byte can be encoded by 2 different 10-bit characters. Characters are selected so that number of '1's and '0's in a character is either equal or differs by one. This guarantees at least one transition every 5 clock cycles. The 8b/10b state machine also tracks running disparity that counts number of sent '0's and '1's. If the difference is not 0, the state machine selects a character with opposite disparity in the next cycle. This keeps transmission line DC balanced.

The 8b/10b decoding adds 12 special characters, not decoded as data. This characters are called control symbols. Control symbols are used to transmit control information across the link. One of the control character, called a 'comma' symbol, is used for synchronization of the link and aligning data stream to the word boundary.

Predictable maximal transition density of the encoding makes it possible to recover clock phase from the data stream. For clock recovery, one can use purely analog methods like PLLs, which can align phase of the output clock to the data stream, or the digital methods, which determine best phase point on a known data stream.

The GTX transceiver in the Virtex-6 FPGA uses clock data recovery based on the phase interpolator [88]. The clock data recovery state machine determines phases of transition and sampling regions for a data stream using base clock, generated by the PLL. The phase interpolator circuit generates sampling clock that is phase synchronous with data. This clock is then used as a recovered clock in FPGA.

Clock Distribution in the Transceivers

Clock synchronization algorithm rely on clock distribution network in the transceivers. Both transmitter and receiver parts of the transceiver consist of four clock domains each. These are serial, physical medium attachment, physical coding sublayer, and user design domains.

Serial domain sends or receives serial data stream. Data are converted between parallel and serial interfaces at the transition between the physical medium attachment domain and the serial domain. Clocks of the serial clock domain and the physical media domain must be perfectly phase aligned to avoid metastability at the domain crossing. Because frequency of the serial clock is very high, transceiver generates and distributes these two clocks using internal resources, which are not accessible to FPGA designer.

The physical medium attachment sublayer, the PMA, in the transmitter is needed to correct link polarity. The PMA in the receiver performs correction of the link polarity, comma detection, comma alignment, and 10b/8b decoding. The physical coding sublayer, the PCS, provisions interface to user design. The interface to user design can change the width of data transferred between user design and the PCS. The PCS in the transmitter also performs 8b/10b encoding of the data.

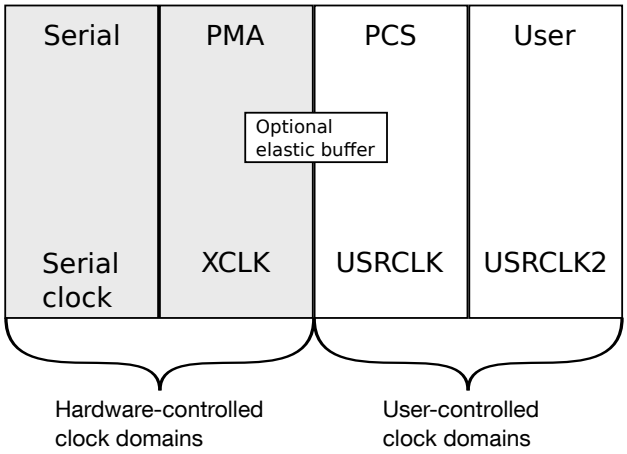


Figure 8.49.: Layout of the clock domains in the high-speed serial transceivers

Data width conversion on the data interface between user design and the PCS is done with two clocks, the USRCLK2 and USRCLK. These clocks have integer frequency ratio to each other that matches the ratio of the data interface widths in both clock domains. Both clocks must have the same clock source and be phase aligned to keep clock domains in synchronization. Designer has full control over generation and distribution of these clocks.

Because the width of the data interface between the PCS and the PMA domains is the same, frequencies in both clock domains match. There are two mechanisms for resolving phase ambiguity in the transceiver. There is a FIFO between clock domains, the elastic buffer, that is used, among other functions, for clock domain crossing. But the elastic buffer introduces not-constant latency in the data path. This is not desired in synchronous systems and should be avoided.

To use transceiver without elastic buffer, we need to synchronize phases of the PCS and the PMA clocks. This can be done with integrated phase alignment circuit. The circuit aligns phase of the PMA clock, the XCLK, to the phase of the PCS clock, the USRCLK, and then keeps clock alignment. The requirements for the phase alignment circuit are described in [88].

Clock Synchronization Algorithm in the UCF

Clock synchronization algorithm is different for transmitter, and for receiver. We synchronize serial data stream to external reference clock in the transmitter. We synchronize recovered clock to the incoming data stream in the receiver.

The external reference clock in the DHH system is the B2TT RF clock. UCF synchronizes serial data stream to the external reference clock in the master FPGA using the phase alignment circuit of the transceiver. The phase alignment circuit is operated by the state machine. Figure 8.50 shows the timing diagram of the phase alignment circuit operation. To operate the phase alignment state machine, UCF executes following steps:

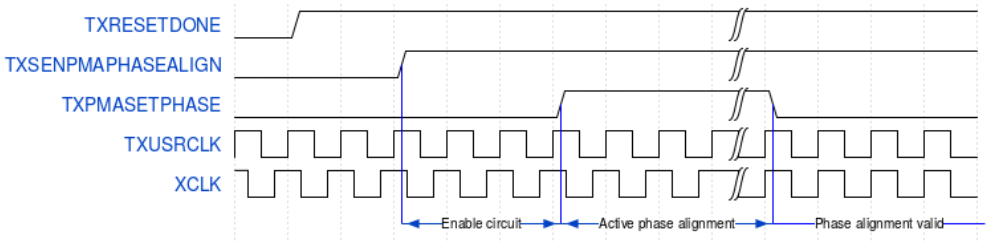


Figure 8.50.: Timing diagram of phase alignment circuit operation

1. Enable phase alignment circuit when transceiver comes out of reset state indicated by ”RESETDONE” signal by driving ”TXENPMAPHASEALIGN” signal high.
2. Wait for 32 cycles of TXUSRCLK2 clock.
3. Drive the signal ”TXPMASETPHASE” high for a pre-defined period to activate active phase alignment. The period depends on the value of the TX PLL divider in the GTX and is deterministic.
4. Keep the signal ”TXENPMAPHASEALIGN” high to let the phase alignment circuit keep the phase of the XCLK clock locked to the USRCLK clock.

After clocks on the master are synchronized, master starts to send the synchronization sequence. The synchronization sequence consists of the comma character, K28.5, and at least one other control character. Length of the synchronization sequence equals the width of the user interface in bytes to avoid ambiguities in the receiver.

Phase synchronization in receiver begins with receiving 8b/10b encoded data. Clock, embedded in data, is recovered in the receiver as the serial clock. The serial clock is divided to obtain the PMA clock that drives user logic, the PCS clock domain, and the serial-to-parallel converter. Deserialized data has 20-fold ambiguity due to possible bit misalignments. The ambiguity is there because clock divider in the PMA cannot lock on a pattern in data.

Standard mechanism to resolve this problem in the receiver is using comma alignment block. We don’t use comma alignment block in the receiver to avoid adding latency to the system. UCF uses stochastic approach to achieve synchronization of the parallel clock with data stream:

1. Reset receiver. Receiver resynchronizes on a random bit in data stream.
2. Wait until receiver comes out of reset state.
3. Check received data again. Probability to achieve correct synchronization is 1 out of 20.
4. Repeat until valid data are received.
5. Mark link as established.

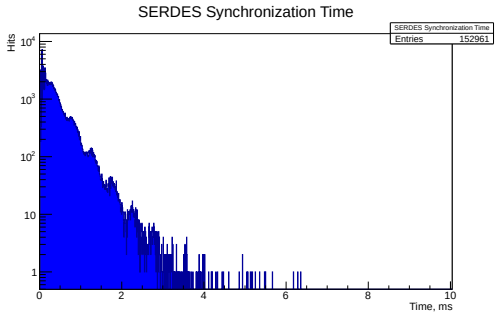


Figure 8.51.: Distribution of the receiver synchronization time

Because locking of the clock dividers in the receiver is a random process, the period necessary for clock synchronization is not deterministic. We measured clock synchronization time on a test setup. The test setup consists of two independent DHH AMC cards connected by an optical high-speed link. The master FPGA synchronizes its transmitter clock to the clock of an oscillator installed on the board and then broadcasts synchronization sequence. The slave FPGA contains a core that measures time required for clock synchronization. We observed, that difference in the clock phases between and after reset depends on the width of the reset signal. To avoid being stuck in the same phase, we randomized the width of the reset signal using a linear feedback shift register. Figure 8.51 shows the distribution of the synchronization time, measured during these tests. We reliably achieved synchronization in less than 7 ms in all trials.

8.7.2. Link Operation

The link between master and slave is operated through the communication between the transmitter and receiver state machines of the UCF. The master transmitter state machine takes over link initialization and data transmission. The state machine in the slave receives data and writes them into the corresponding data stream. The theory of operation of the master and the slave state machines are described in this subsection.

Data Transmission

Data are transmitted between FPGAs as data streams of the UCF link. One physical UCF link between FPGAs provisions up to 65 data streams using the subset of the AXI4-stream interface [89]. Data streams have fixed priorities. UCF handles priorities based on a preemption algorithm described later in this section The stream with the highest priority has fixed latency. This stream is used for distribution of trigger and real-time control information.

UCF manages streams using time division multiplexing algorithm. Streams are assigned a stream ID that defines priority of the stream. The stream with higher stream ID has higher priority.

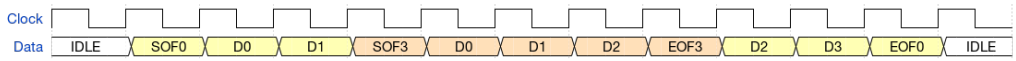


Figure 8.52.: Timing diagram of the preemption algorithm in the UCF. Frame sent on the stream 0 is preempted by the frame on the stream 3

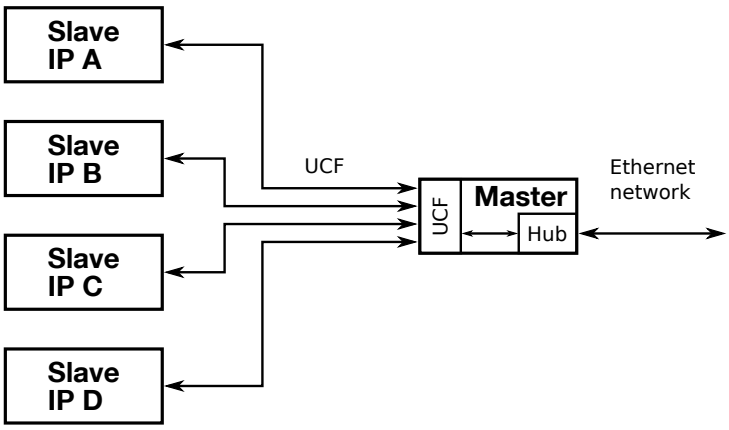


Figure 8.53.: Layout of network forwarding with the Ethernet hub core

UCF uses preemption algorithm, if a stream with higher priority must send data while a stream with lower priority is sending a frame. In this case, UCF interrupts transmission of the stream with lower priority to transmit data of the stream with higher priority. Figure 8.52 shows timing diagram of UCF preemption. If a stream with an even higher priority needs to send data, then UCF interrupts transmission again, and the stream with the highest priority is granted permission to send data. When the stream with the highest priority finishes sending data, UCF grants access to the high-speed link to the interrupted stream with the next-highest priority. The procedure repeats until no interrupted streams are left. Then, UCF enters IDLE state and waits for new data.

8.7.3. Ethernet Hub

UCF allows us to extend IPbus slow control in a transparent way with only DHC being connected to an Ethernet network. This is done by forwarding Ethernet packets through a UCF link using Ethernet hub core.

Figure 8.53 shows the layout of network forwarding with the Ethernet hub core. The core is a part of the UCF master core. The Ethernet hub core emulates functionality of commercially available Ethernet hubs. The hub forwards data between physical Ethernet network and UCF streams, reserved for slow control. This brings the advantage that slow control cores in the slave FPGAs are accessible in the same way as if they were directly attached to Ethernet network.

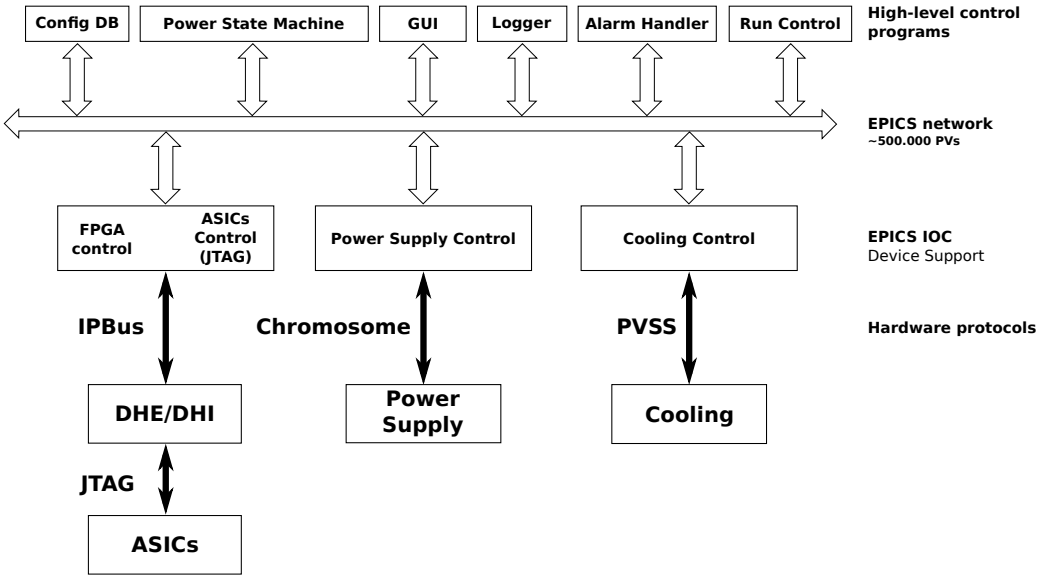


Figure 8.54.: Layout of the slow control system

8.8. Slow Control

Main function of the slow control system is configuration and monitoring of the read-out process. Also, being the only system with direct connection to the detector front-end, slow control of the DHH system takes over the task of configuration and monitoring of the front-end ASICs. This is done via a dedicated JTAG interface in the DHE and in the final system in the DHI. In following sections the term ”slow control” includes the slow control of the DHH system and the slow control of the front-end ASICs.

Functionality in the slow control system is distributed among independent programs that communicate with each other using EPICS[17] framework [90]. EPICS programs communicate with each other over an Ethernet network by accessing process variables provided by the processes. Process variables define message exchange interface between processes. Programs that communicate with hardware are called input-output controllers. Other processes that implement high-level functionality can access hardware by connecting to the input-output controllers through EPICS network.

Figure 8.54 shows the layout of the pixel detector slow control system. The input-output controllers provide access to the DHH hardware and front-end ASICs, power supply units, or cooling system. High-level control includes run control interface, logger, graphical user interface, power state machine, and configuration data base. Run control interface synchronizes system state with the global run control of the Belle II experiment. Logger records process variables for offline analysis. Graphical user interface offers interactive control for expert users. Power state machine implements procedures for switching de-

[17] Experimental Physics and Industrial Control System

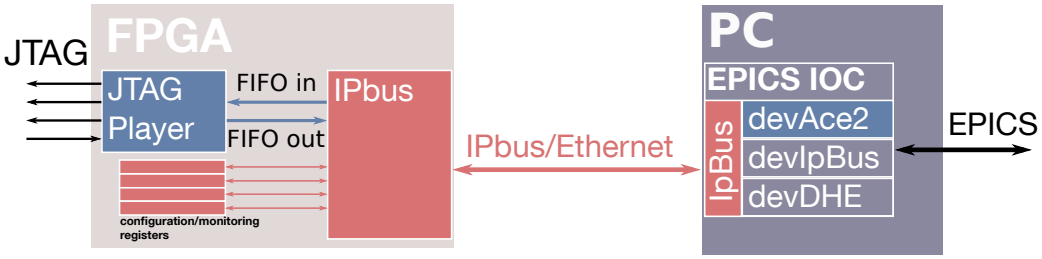


Figure 8.55.: Schematic layout of the DHH slow control

tector on and off. Configuration data base stores configuration parameters for the detector that are read by the power state machine.

Figure 8.55 shows the schematic layout of the DHH slow control. Architecture of the slow control system follows system-on-a-chip architecture. The periphery is memory mapped into the common address space on the bus. Unlike system-on-a-chip architecture, the slow control system does not include a CPU on the FPGA. The FPGA is accessible over Ethernet network through IPbus protocol [91]. The EPICS input-output controller communicates with FPGA firmware through IPbus transactions to the memory-mapped register in the firmware. The input-output controller is a bus master that implements different aspects of high-level control functionality and interfaces hardware resources to the EPICS network.

8.8.1. IPbus

IPbus protocol and implementation were developed by the CMS collaboration for upgrade of the trigger system of the experiment. The protocol uses UDP [18] communication that is simple enough for implementation in the FPGA.

The IPbus system consists of two partners. The active partner that initiates transactions is called IPbus master. The passive partner that receives and processes transactions is called IPbus slave. Master is implemented in software on a PC. Slaves can be implemented in software or in FPGA firmware. Here, the term ”IPbus slave” always refer to the slave implementation in firmware.

The IPbus slave provides Ethernet communication for FPGA control and monitoring. The core converts IPbus transactions, received from Ethernet, to a set of control signals, called the bus, which are used to access FPGA resources.

There are 2 versions of the IPbus protocol between IPbus master and IPbus slave available. Version 1 always operates with one packet at a time. This limits data throughput between FPGA and PC. If a UDP packet is lost in the network, the master must be able to handle this situation.

Version 2 supports sending several packets while waiting for slave to process them. This shortens time between packets and improves utilization of the network bandwidth. Version 2 of the protocol also implements reliable packet delivery on top of UDP. Another

[18] User datagram protocol

interesting feature of version 2 is the possibility of obtaining IP address from the central server using RARP [19] protocol [92].

Still, both versions of the protocol require not more than one master process. More master processes can lead to interference in the slave.

Early laboratory and beam test versions of the firmware used version 1 of the protocol because it was the only version available when we started development of the firmware. We now use version 2 of the protocol for all setups. We tested both versions of the protocol and tests showed that version 2 behaves identical to version 1 under the same conditions.

Addressing

Network communication using IPbus protocol requires 3 addresses: a port number that is unique on the host, an IP address that is unique in the subnetwork, and a MAC address that is unique in the network segment of the host. The firmware of the DHH modules store all possible IP addresses for IPbus instances in the look-up table. The firmware selects particular IP address by reading the look-up table at the address, stored as bits 6-0 of the User access register [93]

$$\text{IP} = \text{LUT[USERACCESS(6 downto 0)]}. \tag{8.4}$$

Because this register is initialized during the last step of firmware generation, this has the advantage that we can change address without modifying firmware. As version 2 of IPbus supports dynamical address assignment, the RARP mechanism using FPGA's unique identifier, the DNA [93], may be used in the future.

MAC address is calculated from 24-bit prefix of the organizational unique identifier of the FPGA manufacturer, Xilinx Inc, and 24 least significant bits of the IP address

$$\text{MAC} = \text{0x000A35} \ \& \ \text{IP(23 downto 0)}. \tag{8.5}$$

Bus Slaves

IPbus master communicates with FPGA and other ASICs through bus slaves that are memory-mapped into the IPbus address space. The bus slave interface is compatible with the Wishbone interconnect used by the open source community [94].

FPGA firmware interacts with IPbus using register-based interface. This interface allows us to read a 32-bit value from the firmware or to write a 32-bit value to the firmware.

8.8.2. IPbus Cores

JTAG Core

The JTAG core is used for communication with front-end ASICs of the detector. Figure 8.56 shows the layout of the core. Building blocks of the core are control block, bitstream decoder, JTAG player, and calibration unit. The core has a register based interface for control and configuration, and a FIFO interface for bitstream transmission.

[19] Reverse address resolution protocol

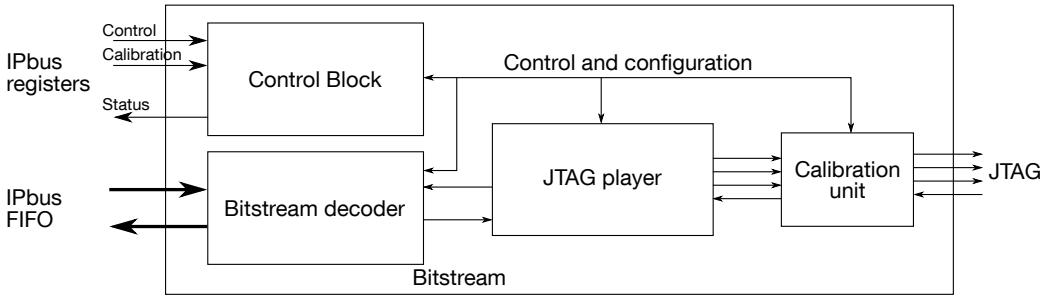


Figure 8.56.: Layout of the JTAG core

The control block decodes control registers as control signals that operate bitstream processing. When bitstream processing is ongoing, the block monitors processing and updates the status register.

The bitstream is received by the IPbus FIFO slave. Bitstream decoder reads the bitstream from the FIFO. Data are then extracted and sent to the JTAG player. If IPbus master requests a JTAG read transaction, the bitstream decoder compresses data that were read from JTAG and writes them to the IPbus FIFO, accessible by master.

The JTAG player executes bitstream instructions by driving data into JTAG IO lines, specified in bitstream instructions, and reading incoming data. Appendix C describes the bitstream format.

The JTAG player has a register that sets frequency divider for the JTAG clock, TCK. Maximum achievable frequency of the interface is

$$f_{\mathrm{TCK,\,max}} = f_{core}/2. \tag{8.6}$$

The divider divides JTAG clock twice per cycle: in the high and in the low state. This reduces JTAG frequency by factor $\mathbf{2 \cdot div}$:

$$f_{TCK} = \frac{f_{core}}{2 \cdot div}. \tag{8.7}$$

JTAG signals generated by the player are connected to the calibration unit. The calibration unit consists of 4 64-bit shift registers. The depth of each shift registers is configured independently by slow control. This is used to adjust phase of a signal. The granularity of the calibration is $\mathbf{1/f_{core}}$.

To operate the core, we need a master that keeps track of logical structure of the JTAG chain and generates bitstream according to chain configuration. Because this task is non-trivial for FPGA implementation, master is implemented in software. Typical operation loop of the core is:

0. configure and reset the core,
1. write length of the bitstream into the bitstream size field of the control register,
2. write compressed bitstream into the FIFO,

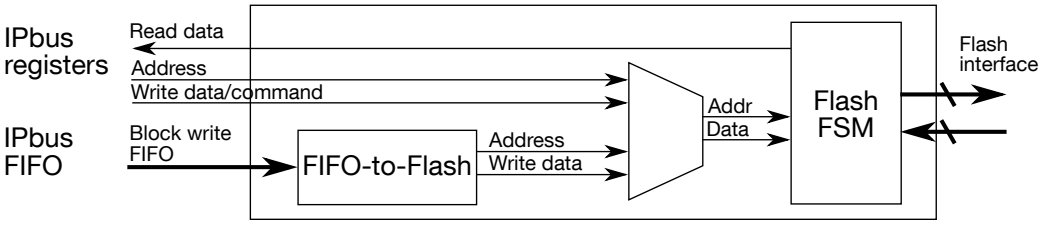


Figure 8.57.: Layout of the flash memory programming core

3. poll status register until End-Of-File or error field is activated,
   - if error flag is active, reset the core,
4. if read operation was requested, read the fill level of the read FIFO,
5. read the specified number of words from the read FIFO.

Flash Memory Programming Core

The flash memory programming core is developed to update system's firmware through slow control. The core is accessible through IPbus and offers much faster programming time than is achievable using standard JTAG programmer. Typical programming time with the core is approx. 1 minute, while with the JTAG programmer is tens of minutes.

The flash memory programmer core supports asynchronous common flash memory interface [95]. The core supports three commands: read, write, and block write. Figure 8.57 shows the layout of the core.

The core consists of the flash state machine and the FIFO-to-flash conversion core. The flash state machine converts commands into transactions of the common flash interface. The core has three registers:

- address register,
- write-enabled data and command register sets single data word or command for the state machine,
- read-enabled data register for data that is read from flash.

The core executes read and write commands directly from IPbus registers. The block write command changes mode of operation. The core switches address and data lines to a FIFO that is filled with data blocks by IPbus master. Upon finishing block write, the lines are switched back to IPbus registers.

I^2C Core

The I^2C protocol is used on the DHH card for configuration of the programmable clock synthesizer Si5338. We use the open source core developed by Richard Herveille as the I^2C bus master. Control registers of the core are described in the specification [96].

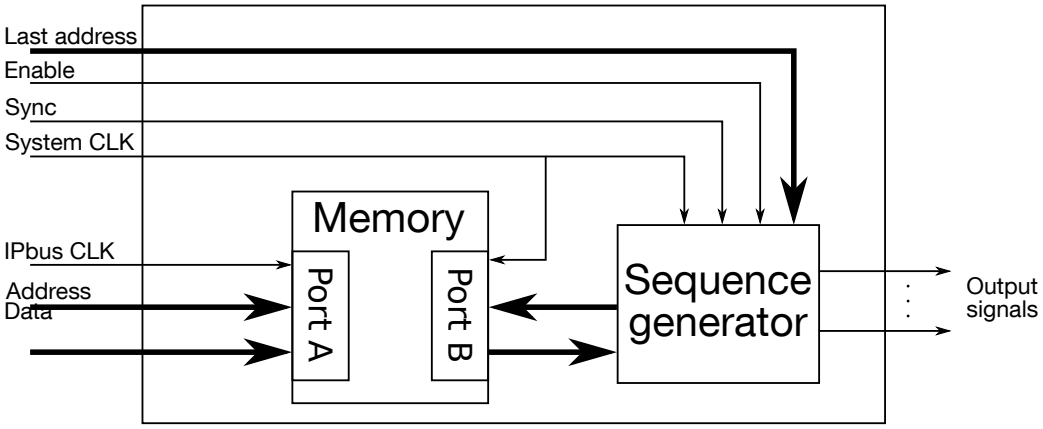


Figure 8.58.: Layout of the sequencer core

Sequencer

The sequencer is a core that generates digital pulses following a sequence stored in memory. The sequencer generated switcher control sequence for the Hybrid 6 board, which does not provide connections between DHPTs and switchers. Another application of the sequencer is synchronization of the read-out with external measurement equipment in the laboratory setups. One of the setups use the sequencer for characterization of the gated mode operation of the pixel detector. Chapter 9 describes these setups in more details.

Figure 8.58 shows the layout of the sequencer core. The core consists of a dual port memory and a sequence generator.

The dual port memory is a 4 bit wide 16 kWords deep block RAM. One port of the memory programs memory using a block RAM IPbus slave. Another port is read by the sequence generator. To program memory, set the initial address and write data to the data register. Consecutive write transactions on the data register increment address.

After memory is programmed, the core executes following algorithm:

1. Idle state. Outputs are in low.
2. Activate sequencer over slow control.
3. Wait for frame sync to start operation. Set address to 0.
4. Read data.
5. Update output signal with new data.
6. Increment address. If last address is reached, set address to 0.
7. Go to 4.

8.8.3. Control Software

Control software is implemented as an EPICS input-output controller. The IOC loads libraries that translate EPICS requests into IPbus transactions to the FPGA. Libraries implement following functionalities:

- 1-to-1 mapping of the EPICS PVs to the IPbus registers for real-time control and monitoring,
- actions that require access to registers in a pre-defined order, and
- high-level JTAG implementation for programming and monitoring of the front-end ASICs.

I use the EPICS asynchronous driver framework, asyn, to implement the libraries [97]. Asyn provides implementation of the EPICS message exchange interface and thread synchronization primitives that are used to prevent simultaneous access to hardware. Appendix D lists configuration parameters of the IOC.

Direct Register Access

Direct access to FPGA registers is implemented in the **devIpBus** library developed by Alan Campbell. The library provisions 1-to-1 mapping of the EPICS PVs to the IPbus registers and bit fields.

At the start of the IOC, the library creates EPICS PVs from the data base configuration. The IOC reads all read-enabled IPbus registers and initializes the PVs with the read values.

Once the IOC is initialized, the library starts to serve EPICS requests. The library does not send requests immediately, but caches them first. Cached requests are sent every 5 ms. Caching improves utilization of the IPbus link by sending larger Ethernet frames at the cost of time resolution of the control system.

Procedural Access

Procedural access library is called **devDHE**. This library implements tasks that require a sequence of IPbus transactions for performing a task. Tasks are triggered by a write access to a corresponding EPICS PV.

The library

- executes reset commands,
- programs clock synthesizer,
- sets phases between FPGA clock and the GCK,
- programs flash memory on the DHH card, and
- triggers FPGA reboot.

Index	Configuration		Comment
	Source clock, MHz	Target clock, MHz	
0	127.21	62.5	Test clock
1	127.21	65	Test clock
2	127.21	67.845	Test clock
3	127.21	71.2376	Test clock
4	127.21	76.326	Clock for B2TT-less systems
5	127.21	127.21	DHC clock
6	76.326	76.326	DHE/DHI clock
7	63.605	63.605	Test clock
8	127.21	72	Test clock
9	127.21	73	Test clock
10	127.21	74	Test clock
11	127.21	75	Test clock

Table 8.2.: List of the available clock configurations

Reset Commands

There are four reset commands accessible through the library. Short and long resets send corresponding reset commands to the DHPT and reset data processing chain in DHE. UDP reset resets the UDP core in DHC. GTX reset resets gigabit transceivers in DHE and DHC. All resets are initiated by writing an arbitrary value into the corresponding EPICS PV.

Clock Synthesizer Configuration

The clock synthesizer Si5338 is configured over I^2C interface using the I^2C IPbus slave. Configurations are generated in the ClockBuilder Desktop software by Silicon Labs. Table 8.2 lists available clock configurations. A configuration is selected using the value, written into the clock configuration EPICS PV. The library then loads configuration into Si5338 and initializes it with the procedure, described in the user guide of Si5338 [98].

Configuration of the GCK Phase

Programming of the GCK phase changes phase relation between the GCK, generated for the DHPT, and the reference clock, generated for the FPGA. This allows us to calibrate phase relation between the GCK and the command line of the FPGA to improve stability of the system with DHPT v. 1.0–1.1. The procedure is initiated by writing a desired phase in units of degrees into the corresponding EPICS PV. The library calculates raw parameters for Si5338 and loads them into the Si5338. Then the soft reset of the Si5338 is performed to apply new settings. Because reset causes interruption of the clock to FPGA and detector front-end, the procedure resets gigabit transceivers in FPGA and then sends long reset command to FPGA and detector front-ends to restore synchronization between them.

Flash Memory Programming Controller

Flash programming controller is implemented as a master for the flash memory programming core in FPGA. The controller can read and decode firmware files and implements several high-level commands that allows us to read and write data from the on-board NOR flash.

The controller is designed to accept data in the output format of Xilinx toolchain, the Intel MCS-86 Hexadecimal Object format [99]. A transaction is initiated by writing the MCS file as an array of 8-bit values into the corresponding EPICS PV. The file is then stored in the temporary file in the IOC's local file system. The controller reads the file, decodes it, sets flash memory in asynchronous mode that is required for writing to flash, erases flash memory, and writes data there.

FPGA reboot

Reboot procedure initiates restarts FPGA by loading FPGA's firmware from flash memory. First, the address of the firmware in flash must be set. The default address is 0x00000000. Then, the procedure can be initiated by writing an arbitrary value into the EPICS PV that triggers reboot.

Reboot procedure first sets flash memory in the synchronous mode. Then, the controller activates reboot state machine that issues a soft reboot command to FPGA.

8.8.4. JTAG Control Software

JTAG control library manages JTAG chain configuration and JTAG registers of the ASICs. Following requirements strongly affected design of the library:

1. dynamic configuration of the chain. DHPTs can include DCDs and switchers in the JTAG chain or exclude them from the chain. Configuration of the JTAG chain is defined by a single bit in the JTAG register of DHPTs. JTAG controller must track this bit and update chain configuration when the bit changes.

2. non-compliance with the JTAG standard. Some registers in the DCD v. 2 and 3 are not compliant with the JTAG standard because registers sample data on the falling edge of the JTAG clock and write data on the rising edge of the clock. This creates an incompatibility between ASICs, which can be fixed in software to some extent. Some registers of the DCD have inverted bit order. Software must convert data before they are loaded into hardware. The ID register of the DCD does not adhere to the JTAG standard, which makes it difficult to automatize discovery of the JTAG chain configuration. The ID register of the switchers is non compliant with the JTAG standard, because the least significant bit of the register is '0'.

3. long configuration registers with large number of bit fields. Configuration registers in the ASICs, which are accessed by a single JTAG transaction, are subdivided into bit fields. Bit fields control single function of an ASIC and are independent of each other. To change a parameter, all other fields must be kept unchanged. A

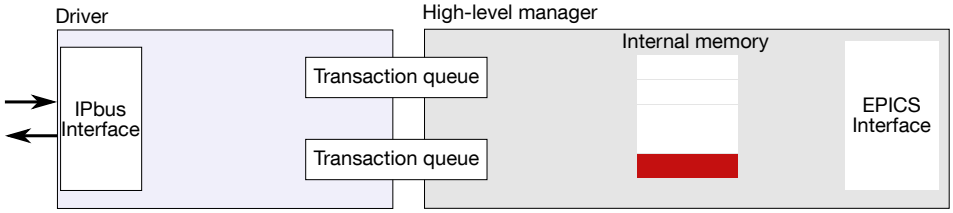


Figure 8.59.: Layout of the high-level JTAG controller

mechanism is needed to keep and recall values of all bit fields. ASICs in a half ladder have more than 10000 bit fields, which the library must manage.

The library **devACE2** implements a software JTAG controller that addresses these requirements. Figure 8.59 shows the layout of the controller. The controller consists of two parts: the high-level manager, and the low-level driver. The high-level manager handles EPICS interface and such abstractions as ASICs, registers, bit fields. The low-level driver handles bitstreams and communication with hardware. To avoid blocking slow control during long transactions, the low-level driver runs asynchronously to the high-level manager. They communicate with each other using asynchronous transaction queues.

High-level Manager

Figure 8.60 shows typical hierarchy of the high-level abstractions. The JTAG chain consists of ASICs, identified by their type and position in the JTAG chain. The ASIC model prepares JTAG transactions for the driver. If a transaction must access ASIC that is not yet included in the chain, the manager extends the chain by configuring the corresponding DHPT. Then, the manager issues initial transaction to the included ASIC.

An ASIC model consists of registers that correspond to physical JTAG registers in the ASICs. A register is either subdivided into bit fields, which can be accessed independently of each other, or is a standalone register without bit fields. A register with bit fields contains two sets of fields: a cached set that stores values before they are written to hardware, and a current set that stores values synchronized with hardware. Bit fields and stand-alone registers are accessible from the EPICS network as signed 32-bit integer records. Update of cached bit fields does not initiate a JTAG transaction, but only changes its value in the memory.

To initiate JTAG transaction, each register with bit fields contains two more fields: **dispatch read** and **dispatch write**. Writing an arbitrary value into the dispatch fields triggers a corresponding JTAG transaction. The manager indicates completion of the transaction by inverting value that was written into the dispatch field:

$$0 \Rightarrow 1$$
$$1 \ldots Int_{max} \Rightarrow 0.$$

After the write access finishes, bit fields of the cached set are copied into the bit fields of the current set. After the read access finishes, bit fields of the cached and the current sets are updated with the read values.

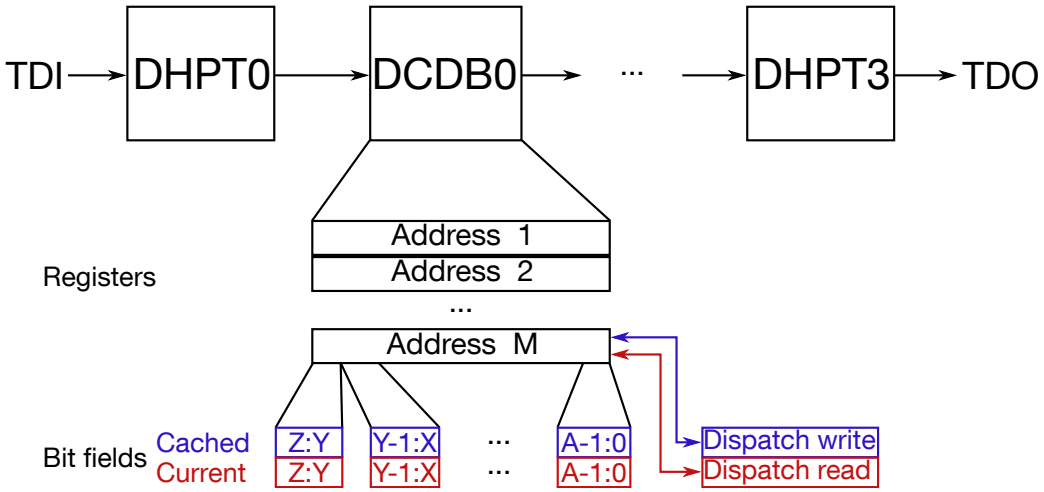


Figure 8.60.: High-level abstractions in the JTAG controller

A stand-alone register has only a **dispatch read** field. Update of the register immediately triggers a JTAG transaction. Completion of the read transaction is indicated in the same way as for the registers with bit fields.

In addition to the normal registers, DHPT model provisions 3 8-bit array records. These arrays are mapped to DHPT memories. The high-level manager divides data, written into these arrays, into 16-bytes words and writes them into ASIC.

Low-level Driver

The low-level driver initializes JTAG chain, optimizes transaction execution, generates bitstream out of single transactions, communicates with FPGA, and decodes data that are read from JTAG interface. The driver also generates a long TCK pattern (over 120'000 cycles), which is used in the temperature measurement core of DHPT. The driver operates on hardware abstractions and physical structure of the JTAG chain: transaction between states of the JTAG final state machine and position of an ASIC in the physical chain [100].

Chain initialization

Initialization of the JTAG chain proceeds in two steps. Because initialization requires active communication with the ASICs, digital part of the ASICs must be powered during this stage. Unpowered or broken ASICs, or broken JTAG chain are usual causes of failures during initialization.

The fist step calibrates phase of the read line, TDO, in respect to the JTAG clock line TCK. Calibration is done by shifting a known pattern into the chain and reading it back for all possible phases. The driver then compares actual pattern that is read back with expected pattern and decides, if the phase is good. If phase 0 is a good phase, then the operational phase is set to 0. Otherwise, the driver chooses the middle of the first interval of good phases as the operational phase.

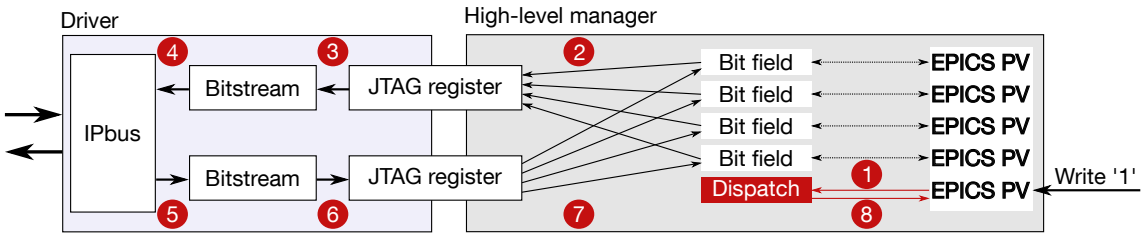


Figure 8.61.: Example of a JTAG transaction

If the calibration step doesn't fail, the second step, chain discovery, is performed. The chain discovery allows us to use unchanged software to operate detector modules with different ASIC configuration. The chain discovery uses following algorithm. First, reset the JTAG controllers in the ASICs. This forces them to load the ID register, if ID register is supported, or the bypass register with value '0' otherwise. Then, send a long pattern consisting of '1's to the JTAG chain. Analyze data, read from JTAG, to determine type and number of ASICs in the chain. DCD and DHPT do not load ID code after reset, but the 1-bit bypass register. This makes it impossible to distinguish DCD from DHPT. Therefore, it is recommended to use chain discovery only immediately after powering up the detector, when the only ASICs present in the JTAG chain are DHPTs.

JTAG Transaction

Figure 8.61 shows a typical JTAG transaction. The transaction consists of following steps:

0. Before a transaction is initiated, users can modify the fields in the IOC's memory by writing to EPICS PVs associated with the fields.

1. User initiates transaction by writing a value into the dispatch PV of the JTAG register. The JTAG register model assembles register content from cached bit fields.

2. The ASIC model writes transaction into the queue to the driver. The transaction includes value of the register, register's instruction code, address of the ASIC, and return action. The return action defines action that is executed upon finishing the transaction.

3. The driver reads transactions from the queue. Transactions are then converted into bitstream for the JTAG core. If a transaction accesses the same register as the previous JTAG transaction, the driver skips instruction code to reduce size of the bitstream. Transactions are processed until one of the following conditions is satisfied:

   - the queue is empty,
   - a read transaction is requested,
   - a reset or reinitialization of the chain is required after the transaction, or
   - the size of the bitstream reaches the limit that prevents FIFO overflow in the JTAG core.

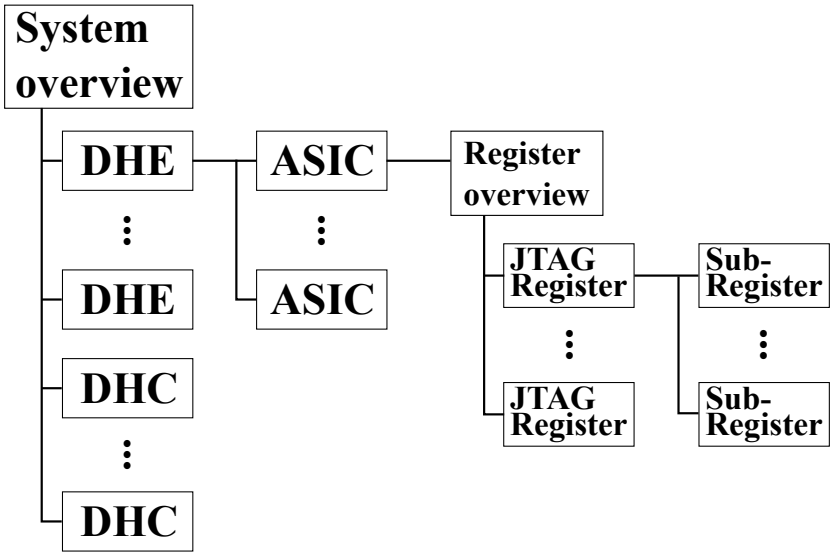


Figure 8.62.: Layout of the graphical user interface for the DHH system

4. The driver transmits bitstream to the FPGA over IPbus. The driver then periodically polls the status register of the JTAG core that indicates end of bitstream processing or an error condition. If no error or End-Of-File bits are asserted after 250 ms, driver marks all transactions in bitstream as failed.

5. If a read transaction does not fail, the driver reads bitstream from the JTAG core. Bitstream contains data and mask that indicates valid bits in data.

6. The driver decodes value of the bit fields. Status of the transaction and decoded data are sent to the high-level manager.

7. The JTAG register model updates current set of the bit fields:
   - read transaction: bit fields are updated with data received from the ASIC;
   - write transaction: cached set of the bit fields is copied into current set.

8. The manager informs user that transaction is complete by changing value of the dispatch PV.

Graphical User Interface

Graphical user interface is implemented in the Control System Studio, the CSS [101]. The CSS includes data provider plugins to access data from different sources, for example from an EPICS PV or from a data base, and an operator interface plugin BOY to visualize data [102].

Figure 8.62 shows the layout of the graphical user interface developed for the system. The layout reflects hierarchical structure of the DHH system and the detector. The graph-

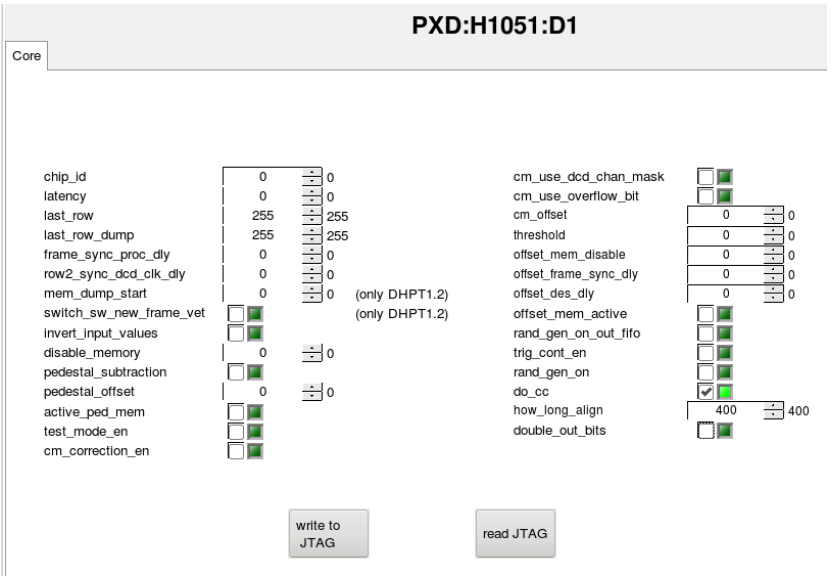


Figure 8.63.: Register OPI for the DHPT core JTAG register

ical user interface consists of operator interface panels, the OPIs, which contain widgets, for example buttons or input fields that are used for interaction with the user.

The work with the system starts by opening the system overview OPI. This OPI contains a list of all available DHE, DHC, and DHI modules in the system. A click on the module's name opens an OPI with settings for this module.

The DHE and DHI OPIs contain a list of ASICs for the half ladder that is connected to this DHE. A click on the ASIC's name opens register overview OPI for this ASIC. The register overview OPI contains a list of JTAG register, which can be opened through a click to a register's name. Figure 8.63 shows the register OPI for the DHPT core register. The register OPI contains a list of bit fields in the JTAG register. Every bit field has two widgets: an input field updates the cached value and a display field shows the current value in hardware.

Chapter 9.

Tests of the Read-Out System

The DHH system was used in the series of beam tests at CERN [1] and DESY [2], during detector development and characterization in laboratory setups, and in phase 2 of the Belle II experiment. This chapter illustrates typical system operation by the example of the laboratory setup for the gated mode test of the DEPFET detector, combined vertex detector beam test at DESY in January 2014, and detector operation during phase 2 of the Belle II experiment.

9.1. Combined Beam Test at DESY

The combined beam test at DESY in January 2014 was dedicated to integration of the pixel and the silicon strip detectors with the prototype of Belle II data acquisition system. The test was conducted at the beam test area TB24/2 of the DESY II accelerator [103]. The goal of the beam test was to test communication between subsystems in the read-out path of the detectors, between the read-out electronics of the detector and the Belle II DAQ, and to study performance of online tracking, alignment software, and online data reduction in the ONSEN system.

9.1.1. Pixel Detector and the DHH System at the Beam Test

Figure 9.1 shows the sensor used in the beam test. The sensor has a matrix with 192x480 pixels with the pixel size 50x75 μm^2. The matrix is operated by four switchers and three pairs of the DHP v. 0.2 and DCDB v. 2. The Hybrid 6 board houses the detector and provisions connectors for power, control and configuration interfaces, and data links.

The detector, used at the beam test, has a major difference to the detector, which is designed for operation at Belle II: DHP cannot operate switchers. This is done to test gated mode operation. This mode is not supported by the DHP 0.2 installed on the detector, but is supported by the next version of the ASIC, the DHPT, which had not yet been available by the time of the test. Instead, the switchers have to be controlled and configured by external electronics. The Hybrid 6 board houses two additional Infiniband connectors for switcher control signals and JTAG configuration interface.

[1] Conseil Européen pour la Recherche Nucléaire

[2] Deutsche Elektronen-Synchrotron

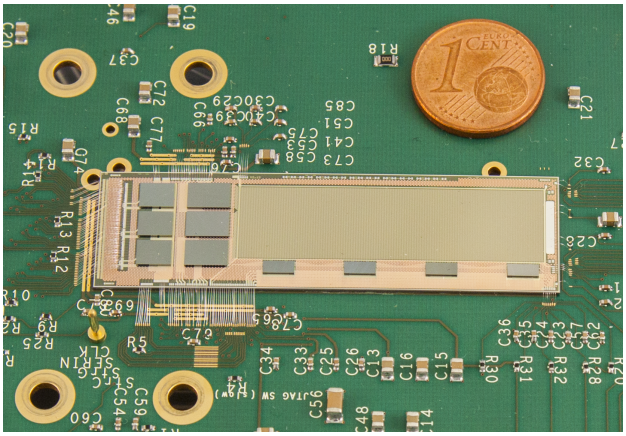


Figure 9.1.: DEPFET sensor used in the beam test at DESY in 2014. Courtesy of M. Schnell

To ensure synchronous operation of the switchers and the DHP-DCDB pairs, all ASICs should be driven by the same clock. Therefore, it was natural to implement switcher control functionality in the DHE. We developed an adapter card, shown in figure 8.11, which is installed into the DDR3 slot of the DHE card. The card adds two Infiniband connectors, which allow us to transmit 16 differential signals. We added the sequencer core, described in section 8.8.2, to operate running shutter sequence in switchers, and the second JTAG player to configure switchers. The sequencer was operated by the same clock as the DHP interface to keep detector's ASICs synchronized.

To transmit trigger information with deterministic latency and fixed clock phase from the DHC to the DHE, we used a custom protocol to transmit trigger over the high-speed serial link. The data rate of trigger transmission link was 2.5 Gb/s. The protocol used during this beam test went into foundation of the UCF protocol and trigger distribution logic in the DHH system. The DHE transmitted pixel data to the DHC over an Aurora link operated at the data rate 6.25 Gb/s.

9.1.2. Read-out Chain

Figure 9.2 shows the layout of the test setup. The setup consisted of a fast scintillation detector, a six-layer beam test EUDET telescope, the pixel detector, and a three-layer silicon strip detector [104]. The pixel detector and the silicon strip detector were installed between the planes of the beam telescope inside of the persistent current superconducting magnet that generated magnetic fields up to 1 T [103]. Strong magnetic field allowed us to study performance of the detectors and online tracking algorithms in conditions similar to Belle II experiment [105]. The beam test telescope helped to establish correct detector alignment.

The fast scintillation detector generated a trigger pulse, if beam particles hit the detector. The trigger logic unit, the TLU, which received trigger pulses, was configured to detect coincidence of the trigger pulses from planes of the scintillation detector. Upon detection

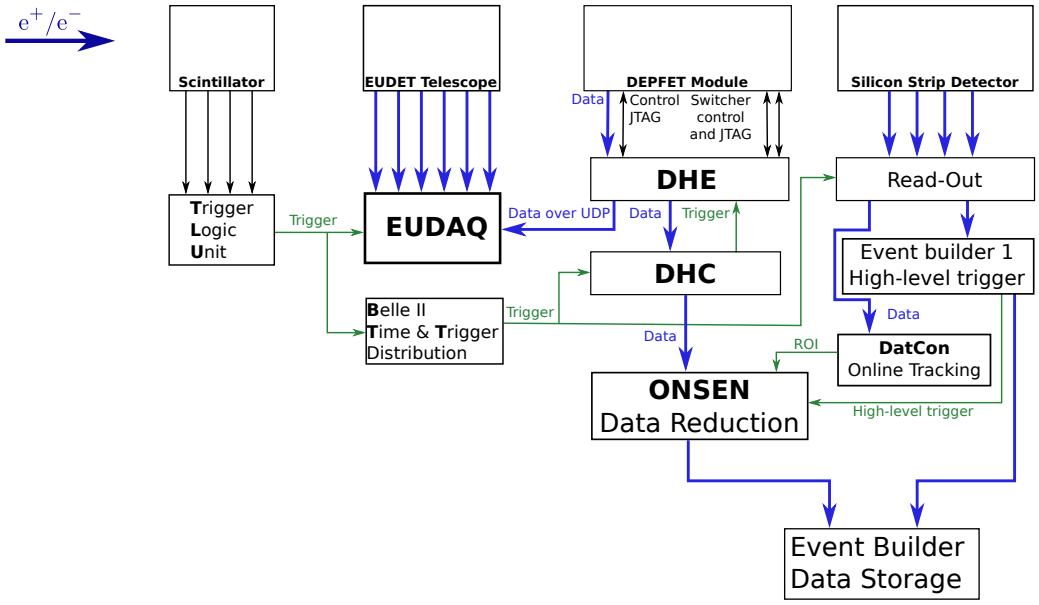


Figure 9.2.: Layout of the test setup at DESY in January 2014

of the coincidence, the trigger logic unit generated a trigger signal and distributed it to the EUDET DAQ and to the Belle II time and trigger distribution system. B2TT distributed triggers to the pixel detector and the silicon strip detector read-out electronics. In this way, events, recorded by the EUDET telescope and the detectors under test, were assigned the same trigger number and could be merged offline. Triggers distributed by TLU and B2TT initiated read-out process in the read-out electronics of the detectors. Because DHE firmware was not capable of processing overlapping events, dead time between triggers was set to 200 μs. This limited maximum achievable trigger rate to 5 kHz.

DHE combined data from the pixel detector data streams to events and sent them to the DHC and to the EUDET DAQ. EUDET DAQ merged pixel detector events with EUDET events. These data were used for monitoring detector alignment and for studies of the pixel detector performance. Data, sent to the DHC, were forwarded to the ONSEN system where data reduction was performed. Detailed description of the ONSEN setup is given in [106].

The read-out electronics of the silicon strip detector sent events to the FPGA-based tracking system DATCON and to the event builder 1. Event builder 1, the software event builder, combined data from four SVD ladders and sent them to the high-level trigger farm. The high-level trigger system reconstructed tracks from SVD data and generated high-level triggers for the ONSEN system, which contains coordinates of the beam intersection with the pixel detector. DATCON system performed similar calculation in FPGA and sent its decision to the ONSEN system as well.

ONSEN system buffered pixel detector events in its memory until it received high-level trigger for the event. The high-level trigger was then merged with regions-of-interest

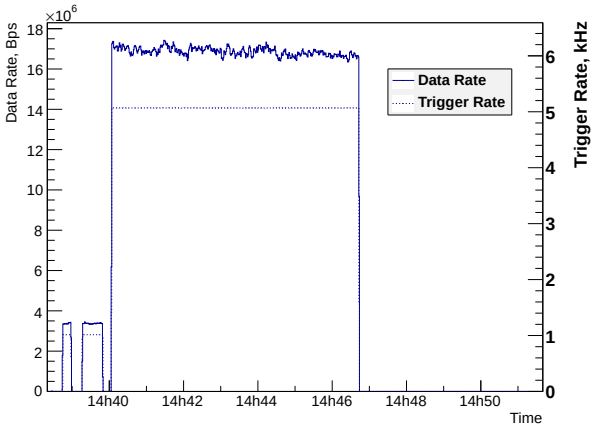


Figure 9.3.: Results of the high trigger rate test

from DATCON. Resulting information was used for filtering pixel detector data. After filtering was finished, ONSEN sent information to the event builder 2 that merges filtered pixel detector data with SVD data. Finished events were stored on hard drives for offline analysis.

9.1.3. Results of the Beam Test

During the beam test, we demonstrated combined operation of the system and recorded more than 20 million events with our read-out chain. We tested online data reduction with calculated and artificial ROIs [106]. Offline analysis of unfiltered data confirmed correctness of the online filtering algorithm, implemented in the ONSEN system.

Average trigger rate during the beam test was 300 Hz with average data rate of 200 kB/s. This corresponds to average detector occupancy of 0.4 %.

We also performed a high trigger rate test to measure trigger rate capability of the pixel detector read-out chain. Figure 9.3 shows data and trigger rates recorded during the test. DHE sustained data rate of 17 MB/s with maximum trigger rate of 5 kHz. This data rate corresponds to an average detector occupancy of 1.8 %. Because we tried to stress the system, no high-level triggers were generated during the test. Therefore, buffers in the ONSEN system were not read-out and were gradually filled with data. This limited test time to 6 minutes.

9.2. Laboratory Setup for Gated Mode Test

One example that shows flexibility of the DHH system is the setup for testing of the gated mode of the pixel detector. Gated mode is intrinsic electronic shutter. Detector in the gated mode guides created free electrons towards reset gate while simultaneously preserving accumulated charge. Main goal of the test is to study gated mode-related detector

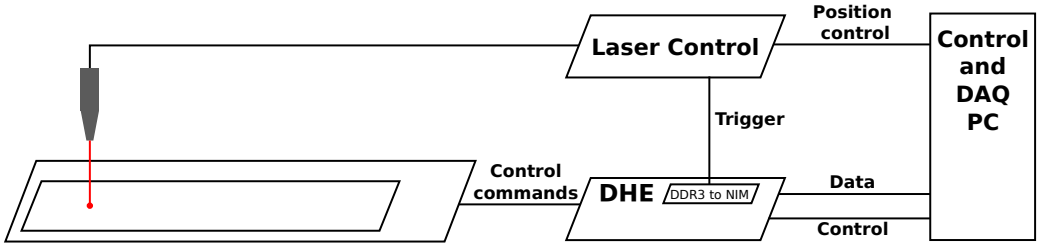


Figure 9.4.: Layout of the setup for the gated mode test

performance. Another important goal is determination of optimum control sequence to minimize time, necessary for switching detector into the gated mode and back to normal operation.

Detector goes into the gated mode through a special configuration of DEPFET control lines, which are steered by switchers. Switcher ASICs receive control signals from the DHPT sequencer. There are two sequencers in DHPT, which are configured over JTAG. DHPT uses the first sequencer for normal detector operation. The second sequencer, which contains gated mode sequence, is activated upon receiving a VETO command from the DHE. In these tests, a laser beam is turned on at a specific point in the read-out cycle to illuminate DEPFET matrix right before or during gated mode. These tests require precise synchronization between laser pulse and the read-out system.

Figure 9.4 shows the layout of the setup for the gated mode test. The setup consists of the control and DAQ computer, the DHE card, and the infrared laser with position control. Computer software initiates tests, configures DHE and position of the laser, records detector data sent by the DHE over Ethernet, and merges detector data with test settings for offline analysis. DHE controls detector operation by configuring ASICs and sending control commands to DHPT through an Infiniband cable. In addition, DHE generates a trigger signal to the laser controller through a DDR3-to-NIM converter board shown in figure 8.10. Trigger signal switches laser beam on and off. Laser beam emulates effects of the noisy bunches in the SuperKEKB.

DHE executes all test steps and synchronizes laser with detector read-out. The sequencer core, described in section 8.8.2, controls operation of detector and laser. The core is synchronized to the frame sync signal and has memory for 8192 four-bit control words. Because the core executes a new command every GCK cycle, memory capacity is sufficient for reading 5 consecutive detector frames. This allows us to study how detector performance behaves after gated mode is switched off.

Bits in the control word control independent functions

- bit 0 activates or deactivates laser beam,
- bit 1 sends a trigger command to DHPTs,
- bit 2 initiates detector read-out process, and
- bit 3 sends a VETO command to DHPTs, which activates gated mode sequencer.

Control sequence is programmable by slow control. This allows us to verify different detector operation modes by using different sequences.

We studied two modes of detector operation during the gated mode. The first mode is gated mode without read out. In this mode, switcher clock stops and switchers are commanded to set all gates in the gated mode. Switchers loose synchronization with running shutter in this mode and have to be resynchronized after clock is available again. The second mode is the gated mode with read out. In this mode, switchers remain synchronous with running shutter.

Detailed results of the tests are published in [107]. Measurements showed that minimal period of the gated mode is 1 μs for the gated mode without read out and 2 μs for the gated mode with read out. Measurements also showed that under certain conditions switcher may require up to 40 μs to resynchronize with the DHPT in the gated mode without read out. Because we want to minimize dead time, gated mode with read out is the only acceptable mode of operation. Another measurement showed that pedestal current through DEPFET pixels in the gated mode exhibit large fluctuations. These fluctuations increase detector occupancy. This situation would lead to loss of data due to high occupancy. Therefore, it is important to avoid reading these rows. These results will define implementation of the gated mode controller that will be implemented for detector operation at Belle II.

9.3. Phase 2 of the Belle II Experiment

Phase 2 of the Belle II experiment is the commissioning run of the experiment. Phase 2 was running from February to July 2018 started with no beam and cosmic ray trigger. On April 26th, SuperKEKB collided electron-positron beams for the first time. Since then, the experiment recorded 500 pb^{-1} of integrated luminosity.

There were several goals to achieve during phase 2. First, the experiment has to demonstrate combined operation of all subdetectors in Belle II. This goal requires integration of the detectors in the Belle II DAQ and B2TT system. Next, the SuperKEKB collider was prepared and tuned for operation under designed conditions. To achieve this goal, the collider gradually increased beam currents to 0.7 A and decreased horizontal betatron function β_y of the beam to 2 mm. Data, recorded during phase 2, are used for dark matter searches and for validation of the reconstruction algorithms in the known decay modes.

9.3.1. Pixel Detector in the Phase 2

The Belle II detector in phase 2 consists of all Belle II subdetectors, described in the chapter 6, except for the full vertex detector. The vertex detector is represented by one sector of the final vertex detector, built out of two pixel detector ladders and four silicon strip detector ladders. Figure 9.5 shows the expanded view of the inner detector configuration in phase 2. Besides pixel and silicon strip detectors, there are detectors of the BEAST II[3] experiment [108]. BEAST II measures and characterizes beam background to

[3] Beam Exorcism for a Stable ExperimenT II

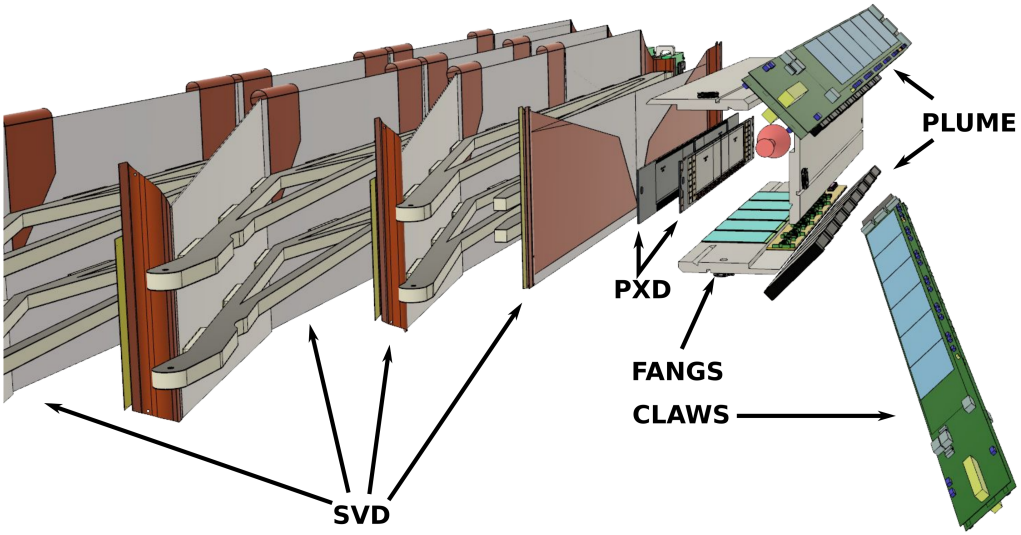


Figure 9.5.: Expanded CAD view of the inner detectors in phase 2

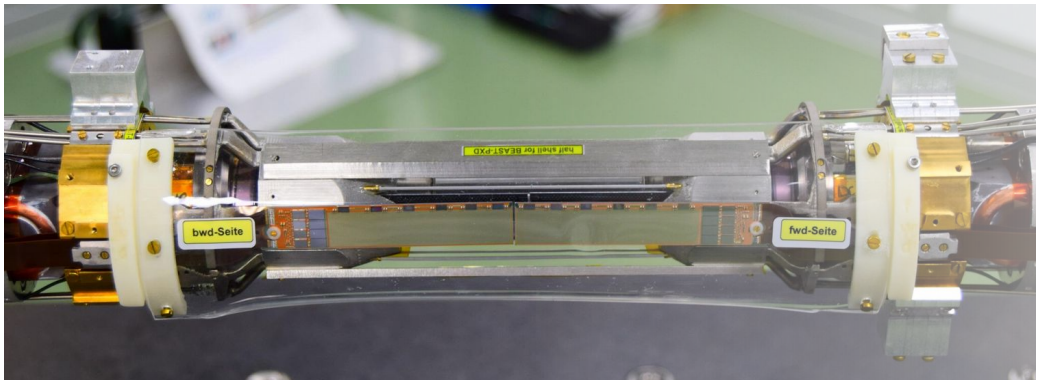


Figure 9.6.: Two-layer pixel detector for phase 2 mounted on the cooling block. Courtesy of C. Marinas

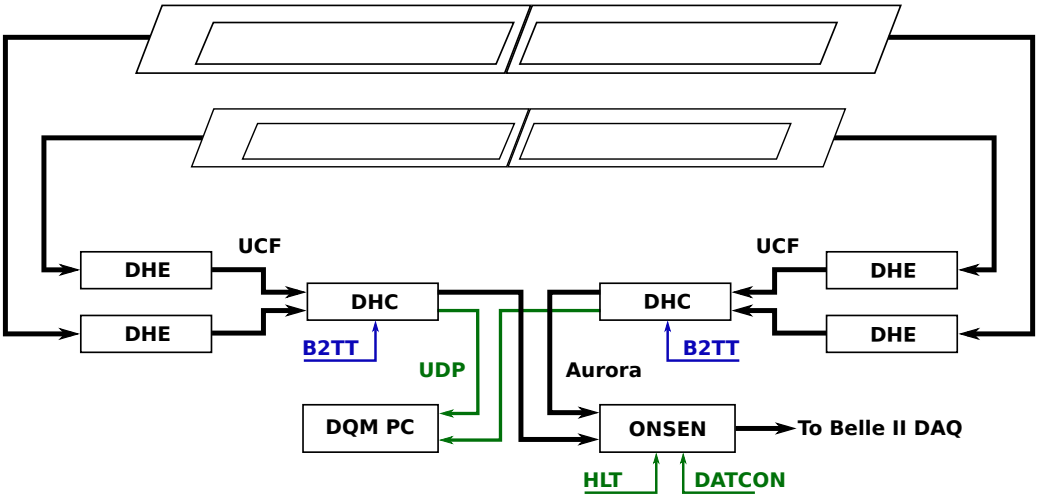


Figure 9.7.: Layout of the pixel detector readout chain during phase 2 of the experiment

provide feedback for SuperKEKB optimizations. The BEAST II subdetectors, installed in the vertex detector regions, are the scintillator detector CLAWS, the double-sided pixel detector PLUME, and the hybrid pixel detector, based on the FE-I4 technology, FANGS.

Figure 9.7 shows readout chain of the pixel detector in phase 2. B2TT initiates data read-out by distributing trigger signal, generated in the trigger decision logic, to DHC modules. DHC distributes trigger signal over the UCF links to DHE modules, which send read command to detector front-end ASICs. Detectors are connected to the DHEs through a DHI prototype card. DHI prototype card converts DHE interface, which consists of Infiniband and RJ45 cables, to a CameraLink detector interface. DHI prototype uses interface conversion to implement galvanic isolation and signal conditioning. Detectors transmit data to DHEs over optical fiber links. DHE combines data from four links to an event and sends events to DHC over a 2.5 Gb/s UCF link. The DHC merges events with the same trigger number from two DHEs and builds subevents out of them. Then, DHC distributes subevents to ONSEN system over a 6.25 Gb/s Aurora link and to the data quality monitoring computer over a 1 Gb/s Ethernet UDP link. The DQM system monitors data consistency and detector performance. ONSEN stores subevents in the memory, while waiting for high-level trigger decision and regions-of-interest from DATCON. Once high-level trigger decision is received, ONSEN starts data reduction process, based on information from HLT and DATCON. ONSEN sends reduced data to the event builder 2 for merging with the rest of event.

9.3.2. DHH System in the Phase 2

DHH system in phase 2 consists of four DHE modules, four DHI prototype boards, and two DHC modules. The DHE and DHC modules were installed in two ATCA carrier boards described in the section 8.3.5. The ATCA carrier boards used rear-transition modules, which house optical data receivers for receiving detector data, optical transceivers for

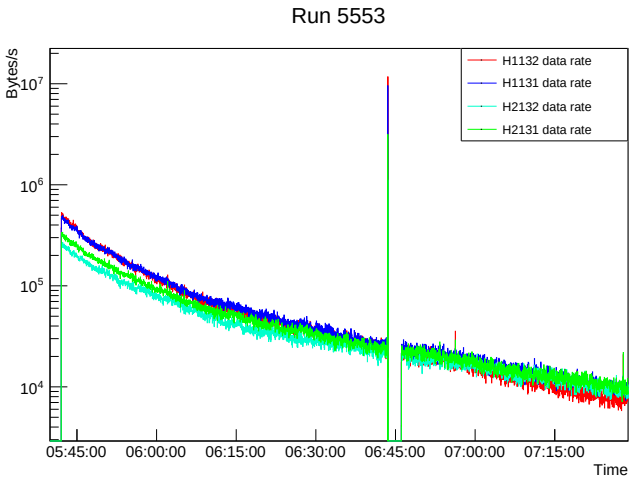


Figure 9.8.: Increase of detector data rate in all DHEs in the run 5553

ONSEN links, Ethernet transceivers for DQM data link and slow control, and the B2TT interface.

We used two versions of DHE firmware in phase 2. Firmware versions have different types of data processing chain.

The first version contains data processing chain, which is used in the stand-alone firmware, with a UCF link for receiving triggers from DHC, sending events to DHC, and for slow control. This type of firmware was considered stable and was used most of the time. Data processing chain of the lab firmware required dead time of 200 μs between triggers for processing events. The firmware operated stably at the average trigger rates of 300 Hz.

We observed a rare problem when a DHE didn't send out event to the trigger. This lead to the loss of synchronization in the subevent builder in DHC, and, consequently, ONSEN system stopped the run due to inconsistency in data. The problem was caused by the sharp rise in detector data rate. Figure 9.8 shows increase of detector data rate in all detector modules during one run. As all detectors had seen increase in data rate at the same time, we consider the problem to be caused by instability of the beam. Its cause is still being investigated by accelerator experts. Because of high detector occupancy, data processing chain in the DHE was unable to finish event processing within 200 μs. Therefore, DHE ignored triggers, arrived while processing of the previous trigger was not finished.

The second version is the firmware with support for overlapping triggers as described in section 8.5.2. This type of firmware lacked pixel remapping, cluster reconstruction, and cluster analysis cores during phase 2. The firmware used in phase 2 buffered data in block RAM of the FPGA instead of the external DDR3 memory.

We tested the firmware with pre-defined test patterns and during physics runs. We used a test pattern that emulates detector occupancy of 0.4 % during the tests. The occupancy was limited by the available bandwidth on the UCF link between DHEs and DHC. We set trigger dead time to 1 μs during the test to guarantee minimum trigger separation of

one detector gate. We performed the test with 50 kHz pulse trigger and 30 kHz Poisson trigger. We observed no errors in data during these tests. This qualifies the firmware for use in physics runs. During the physics runs, we observed a minor problem in the data format, but no obvious problems otherwise.

Operation of the DHH system in phase 2 of the Belle II experiment was an important step towards the final system. The problem with high-occupancy events pointed out the necessity of the data streams resynchronization in the DHC firmware. The first operation of the overlapping trigger firmware under real conditions was successful. Data rate of the UCF links between DHEs and DHC will be raised to 5.1 Gb/s for phase 3 of the Belle II experiment. Higher link rates will allow us to read detector data with higher occupancy in phase 3.

Chapter 10.

Summary and Conclusions

I measured branching fraction of the decay $D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0$ using full data sample of the Belle experiment of 1 ab^{-1}. To measure branching fraction, I modeled the signal and two normalization channels using Monte-Carlo simulated data. Then, I compared results of the simulation to 10 % of Belle data sample to validate my reconstruction algorithm. The shape of invariant mass distributions and distribution of the kinematic parameters agree in data and simulation.

Despite my estimation for the branching fraction, statistical significance of the signal in the fit is only 2 σ. Therefore, I report the 95 % upper exclusion limit for the branching fraction of the decay channel

$$BF(D^+ \to K^- K^0_S \pi^+ \pi^+ \pi^0) < 7.23 \cdot 10^{-5}. \tag{10.1}$$

I also estimated branching fraction of the similar decay channel $D^+ \to K^- \pi^+ \pi^+ \pi^+ \pi^- \pi^0$ that consists of two subchannels $D^+ \to \overline{K}^{*0} \eta \pi$ and $D^+ \to \overline{K}^{*0} \omega \pi$. In this decay channel, the pion pair $\pi^+ \pi^-$ do not originate from the decay of K^0_S. It is important to note here, that selection criteria in my analysis are not optimized for this decay mode. The measurement with statistical significance 3 σ yields branching fraction

$$BF(D^+ \to \overline{K}^{*0} \eta \pi, D^+ \to \overline{K}^{*0} \omega \pi) = (1.46 \pm 0.63) \cdot 10^{-3} \tag{10.2}$$

and the upper 95 % confidence level

$$BF(D^+ \to \overline{K}^{*0} \eta \pi, D^+ \to \overline{K}^{*0} \omega \pi) < 2.28 \cdot 10^{-3}. \tag{10.3}$$

Both results agree with the measurement in [31].

The Belle data sample is enough to measure branching fraction of the $D^+ \to K^- \pi^+ \pi^+ \pi^+ \pi^- \pi^0$ decay mode with high statistical significance if its branching fraction is in the order of 10^{-3}. This would require, in the simplest case, selection of the pion pair $\pi^+ \pi^-$ which are incompatible with K^0_S hypothesis. Further improvements to the selection criteria would increase the signal-to-background ratio.

To improve sensitivity of the measurements, we contributed to the Belle II experiment by developing the FPGA-based read-out system for the pixel detector. We built two types of the system: a stand-alone system, and a high-performance two-level system for integration into the Belle II data acquisition system.

The stand-alone read-out system is used for detector development and detector characterization. This system contains a full data processing core and a detector control interface necessary. A computer and an FPGA board is sufficient to operate the detector. We extended the system with adapter boards to enable new modes of detector operation, for example to generate signals for tests of the gated mode with external light source. This flexibility makes the stand-alone read-out system well suitable for laboratory setups.

We used the experience, collected during detector operation with the stand-alone read-out system, to build a full scale system. We optimized the full scale read-out system for high-performance operation at Belle II. We separated system's functionality into three subsystems: data processor DHE, subevent builder DHC, and detector control and configuration interface DHI.

The beam tests gave us the opportunity to operate the read-out system with the ONSEN system and the prototype of the Belle II DAQ system. Operation of the system at the beam tests helped to determine final requirements for the read-out system. This contributed to successful commissioning of the system in phase 2 of the Belle II experiment.

Operation of the system in phase 2 allowed us to test the firmware with the overlapping trigger functionality and outlined the problems of the system. The subevent builder needs a mechanism to check for synchronization between DHE data streams and to resynchronize the streams in the case, if synchronization is lost. We expect event sizes and trigger rates to increase in phase 3. Bandwidth of the UCF link has to be increased as well to fulfill system's requirements. The gated mode operation with the full-scale system must be tested within the Belle II DAQ system. This requires additional support in the B2TT hardware and in the B2TT receiver core. However, in phase 2 we did not observe any critical problems in the design of the read-out system.

During development of the read-out system, we used FPGA as central piece of the system. We designed a new FPGA-based event building algorithm and a synchronous serial protocol UCF. The FPGA-based event builder offers some advantages over software solutions. The hardware of the FPGA-based event builder is very compact in comparison to software-based event builders. Implementation of the event builder in FPGA allows us to process data in real time. This makes the algorithm deterministic, reduces its complexity, and improves reliability of the system. Bandwidth of the memory and of the high-speed links is utilized exclusively by the event builder, thus maximizing performance of the algorithm. The UCF protocol can carry several logical streams in a single link, thus reducing system costs. Due to its ability to carry a synchronous data stream, UCF is perfect for trigger and clock distribution systems. These algorithms are highly flexible and can be used as building blocks for future data acquisition systems.

Appendices

A. BDT Parameters

Table .1 lists parameters, used in the BDT, and their relative importance.

Name	Variable importance, %	Description
$L_{xy}(D)$	7.148	projection of the decay length in xy-plane
$E_{CMS}(\pi^0)$	5.940	energy of π^0 in the center-of-mass frame of D^+
$p_t(D)$	5.221	transverse momentum of D^+ candidate
$x(D)$	5.150	normalized momentum
$p_{vertexfit}$	4.978	vertex fit p-value
$cos(\Theta_{CMS})$	4.574	cosine of the angle between D^+ and z axis
$\chi^2(\pi^0)$	4.363	χ^2 of mass fit of π^0
R_2	4.206	2nd Fox-Wolfram moment
γ_L angle	3.992	angle in ECL of the low-energy photon from π^0
γ_H angle	3.935	angle in ECL of the high-energy photon from π^0
$p_t({\pi^+}_L)$	3.722	transverse momentum of low-energy π^+
$cos(\Theta(K^-))$	3.703	cosine of the angle between K^- and z axis
$m(K^0_S)$	3.648	mass of K^0_S
$E_{CMS}(K^0_S)$	3.647	energy of K^0_S in the center-of-mass frame of D^+
$p_t({\pi^+}_H)$	3.609	transverse momentum of high-energy π^+
$cos(\Theta({\pi^+}_L))$	3.576	cosine of the angle between low-energy π^+ and z axis
$cos(\Theta({\pi^+}_H))$	3.471	cosine of the angle between high-energy π^+ and z axis
$p_t(K^-)$	3.442	transverse momentum of K^-
$log_{10}(\frac{E(\gamma_L)}{E(\gamma_H)})$	3.084	log of energy ratio of π^0 photons
$E(\gamma_L)$	2.690	energy of the low-energy photon
$E(\gamma_H)$	2.487	energy of the high-energy photon
$N_{SVDhits,z}({\pi^+}_L)$	2.071	number of SVD hits in z-stripes for low-energy π^+
$L_e({\pi^+}_L)$	1.817	likelihood of low-energy π^+ being an electron
$N_{SVDhits,r\phi}({\pi^+}_H)$	1.722	number of SVD hits in rϕ-stripes for high-energy π^+
$N_{SVDhits,z}({\pi^+}_H)$	1.720	number of SVD hits in z-stripes for high-energy π^+
$N_{SVDhits,r\phi}({\pi^+}_L)$	1.634	number of SVD hits in rϕ-stripes for low-energy π^+
$N_{SVDhits,r\phi}(K^-)$	1.567	number of SVD hits in rϕ-stripes for K^-
$N_{SVDhits,z}(K^-)$	1.493	number of SVD hits in z-stripes for K^-
K^0_S vertex displacement	1.388	distance between D^+ and K^0_S vertices

Table .1.: List of the parameters used in the BDT training

B. EPICS Process Variables

B.1. Process Variables in the DHE

Name	**D**escription	**R**ange
ipbus_address:ID:cur	IPbus ID	
dheid:ID:set	DHE ID (appears in the frame header)	
	Firmware revision	
revision:VALUE:cur	32 bit DHE firmware revision	
revision_year:VALUE:cur	year of the DHE firmware revision synthesis	
revision_month:VALUE:cur	month of the DHE firmware revision synthesis	
revision_day:VALUE:cur	day of the DHE firmware revision synthesis	
revision_hour:VALUE:cur	hour of the DHE firmware revision synthesis	
revision_minute:VALUE:cur	minute of the DHE firmware revision synthesis	
ethaddr:VALUE:cur	Address of the Ethernet configuration in the firmware table	
	FPGA monitoring	
temperature:TEMP:cur	Temperature of the FPGA, C	
vccaux:VALUE:cur	VCCAUX of the FPGA, V	
vccint:VALUE:cur	VCCINT of the FPGA, V	
	Reconfiguration and clocks	
boot_address:VALUE:set	Address of the firmware in the DHE flash to be used during reboot	
mcs_array:A:set	PV for uploading the firmware update in the MCS format	
fpga_reboot:S:set	Reboot FPGA firmware from flash	0-1
half_rate:S:set	Set DHPT data links to work in half rate mode	0-1
clock_frequency:set	Program clock synthesizer with the selected frequency	
	Resets	
dhp_rst_long:S:set	Generate long reset for DHPT and DHE. Long reset lasts two cycles of the trigger command or longer	
dhp_rst_short:S:set	Generate short reset for DHPT and DHE. Short reset lasts one cycle of the trigger command	
dhpt_crst:S:set	DHPT power up reset. Active low. Signal uses spare line of JTAG Infiniband connector as power up reset for Hybrid5 / DHPT test board[1]	0-1

Name	**D**escription	**R**ange
rst_gtx:S:set	Reset GTX on the DHPT links	0-1
DHPT link monitoring and parameters		
dhp_pwdn_n:S:cur	Status of the DHPT sense line	
rxeqmix:VALUE:set	RXEQMIX for all DHPT channels. Register selects transfer function for the equalizer. See [88]	0-7
dfeeyedacmon1:AMPL:raw	Signal EYE amplitude for DHPT ch. 1	
dfeeyedacmon2:AMPL:raw	Signal EYE amplitude for DHPT ch. 2	
dfeeyedacmon3:AMPL:raw	Signal EYE amplitude for DHPT ch. 3	
dfeeyedacmon4:AMPL:raw	Signal EYE amplitude for DHPT ch. 4	
dhp1_channel_up:S:cur	DHPT ch. 1 up flag	
dhp2_channel_up:S:cur	DHPT ch. 2 up flag	
dhp3_channel_up:S:cur	DHPT ch. 3 up flag	
dhp4_channel_up:S:cur	DHPT ch. 4 up flag	
dhp_data:CNT:cur	DHPT data counter, B	
dhp_data:RATE:cur	DHPT data rate, 1/s	
dhp_frame:CNT:cur	DHPT frame counter	
dhp_pll_not_locked:S:cur	GTX PLL on the DHPT channel not locked flag	
dhp_stat_en:S:set	Enables DHPT statistics block	0-1
remapping:S:set	Enables pixel address remapping in DHE firmware	
DHC link monitoring		
dhc_channel_up:S:cur	DHC link channel up	
dhc_data:CNT:cur	DHC data counter, B	
dhc_data:RATE:cur	DHC data rate, 1/s	
dhc_frame:CNT:cur	DHC frame counter	
dhc_pll_not_locked:S:cur	GTX PLL on the DHC channel not locked	
dhc_stat_en:S:set	Enables DHC statistics block	0-1
trg:CNT:cur	Trigger counter	
trg:RATE:cur	Trigger rate, 1/s	
trg_missing:CNT:cur	Counter for the triggers ignored due to high FIFO level	
trg_missing:RATE:cur	Rate for the ignored triggers, 1/s	
trg_stat_en:S:set	Enables trigger statistics block	0-1
cm_adc:VOLT:cur	Current mirror ADC reading, ADU	
cm_dac:VOLT:set	Current mirror DAC setting, ADU	0-65535
cm_scan_dac_min:VALUE:set	Minimum DAC value for the current mirror scanner core[1]	0-65535
cm_scan_dac_max:VALUE:set	Maximum DAC value for the current mirror scanner core[1]	0-65534

Name	**D**escription	**R**ange
cm_scan_dac_step:VALUE:set	Step size for the current mirror scanner core[1]	0-65535
cm_scan_dac_to_trg_dly:VALU	Delay between setting DAC in the current mirror and generating a trigger. Set to 500[1]	0-65535
cm_scan_start:S:set	Start the scan flag[1]	0-1
cm_scan_finished:S:cur	Scan core not busy flag[1]	
cm_scan_trg_per_dac:VALUE:s	Number of triggers generated for a single DAC settings[1]	0-255
DHPT interface parameters		
trg_dly:VALUE:set	Delay between receiving trigger and sending trigger to DHPT	1-32767
trg_en:S:set	Enable trigger processing in DHE	0-1
trg_len:VALUE:set	Length of the DHPT trigger signal in GCK clock cycles	1-32767
frame_timeout:VALUE:set	Frame timeout in GCK clock cycles. Used for event separation, not used in overlapping trigger mode	0-100000
fck_len:VALUE:set	Frame length in GCK clock cycles	1-32767
fck_strobe_length:VALUE:set	Width of the FCK signal in GCK clock cycles. Always set to 1 in the DHPT mode	1
invert_trg:S:set	Invert the DHPT trigger signal	0-1
trg_ipbus_freq:VALUE:set	Frequency of the software trigger, Hz[1]	0-10000
ipbus_trg_start_nr:VALUE:set	Initial trigger number	0-($2^{32}-1$)
nr_of_ipbus_trg:VALUE:set	Expected number of triggers. Set to 0 to generate unlimited number of triggers	0-($2^{32}-1$)
ipbus_trg_en:S:set	Enable artificial trigger core	0-1
use_ipbus_trg:S:set	Use artificial trigger input to trigger processor	0-1
veto:S:set	Set level of the VETO signal for DHPT[1]	0-1
mem_dump:S:set	Send one memory dump command	0-1
use_dhpt:S:set	Use DHPT manchester encoded interface. If set to 0 then DHP 0.2 interface (4 LVDS line) is used	0-1
UDP data link configuration		
udp_eth_dst_hi:VALUE:set	2 higher bytes of the PC MAC address	0-65535
udp_eth_dst_low:VALUE:set	4 lower bytes of the PC MAC address	0-($2^{32}-1$)
udp_ip_dst:VALUE:set	PC IP address in the form 0xAABBCCDD = AA.BB.CC.DD[1]	0-($2^{32}-1$)
udp_ip_src:VALUE:set	DHE IP address in the form 0xAABBCCDD = AA.BB.CC.DD[1]	0-($2^{32}-1$)

[1] Exists only in lab firmware

Name	**D**escription	**R**ange
udp_port_dst:VALUE:set	PC UDP port[1]	0-65535
udp_port_src:VALUE:set	DHE UDP port[1]	0-65535
JTAG core parameters		
jtag_alternatingBypass:S:set		0-1
jtag_idle120000clk:S:set	Generate 120000 idle JTAG clocks. Used in temperature measurement procedure for the DHPT	0-1
jtag_reinit_chain:S:set	Re-initialize JTAG chain	0-1
jtag_chain_initialized:S:cur	JTAG chain initialized flag	
jtag_port:VALUE:cur	UDP Port of the JTAG memory server. JTAG memory server is deprecated and will be removed	
jtag_queue_size:VALUE:cur	Number of unprocessed transactions in the JTAG driver	
tck_divider:VALUE:set	Clock divider for the JTAG player. Divides JTAG clock by factor $2 \cdot (N+1)$	0-255
tck_dly:VALUE:set	Delay of the TCK line, IPbus clock cycles	0-31
tdi_dly:VALUE:set	Delay of the TDI line, IPbus clock cycles	0-31
tms_dly:VALUE:set	Delay of the TMS line, IPbus clock cycles	0-31

Name	**D**escription	**R**ange
dhc_board_id:ID:set	DHC ID (appears in the frame header)	0-15
FPGA monitoring		
dhc_temperature:TEMP:cur	Temperature of the FPGA, C	
dhc_vccint:VALUE:cur	VCCINT of the FPGA, V	
Firmware revision		
revision:VALUE:cur	32 bit DHC firmware revision	
revision_year:VALUE:cur	year of the DHC firmware revision synthesis	
revision_month:VALUE:cur	month of the DHC firmware revision synthesis	
revision_day:VALUE:cur	day of the DHC firmware revision synthesis	
revision_hour:VALUE:cur	hour of the DHC firmware revision synthesis	
revision_minute:VALUE:cur	minute of the DHC firmware revision synthesis	
ethaddr:VALUE:cur	Address of the Ethernet configuration in the firmware table	
Firmware configuration		
def_b2tt_flag:S:cur	B2TT core included	
def_ddr3:S:cur	DDR3 FIFO included	
def_dhe_gen:S:cur	DHE data generator included	
def_nmb_dhe:VALUE:cur	Number of DHE links	
def_onsen:S:cur	ONSEN links included	
def_opencore_udp:S:cur	UDP core included	
def_sel_onsen_625:S:cur	ONSEN link data rate. '1' - 6.25 Gb/s, '0' - 3.12 Gb/s	
def_useictrl:S:cur	External IDELAYCTRL used in B2TT	
Resets		
dhc_rst:S:set	Reset data processing cores	0-1
dhc_rst_gtx:S:set	Reset GTX	0-1
dhc_rst_long:S:set	Send long reset to DHEs	0-1
dhc_rst_short:S:set	Send short reset to DHEs	0-1
dhc_rst_udp:S:set	Reset UDP core	0-1
trg_mismatch_rst:S:set	Reset trigger mismatch counters	0-1
disable_onsen_flow:S:set	Disable data transmission to ONSEN	0-1

Name	**D**escription	**R**ange
use_b2tt:S:set	Trigger source selection. ’0’ - artificial trigger generator, ’1’ - B2TT	0-1
clock_frequency:set	Program clock synthesizer with the selected frequency	
dhc_ACC_TRG_CNT:CNT:cur		
dhc_b2clkup:S:cur	Status of the B2TT clock	
dhc_b2ttup:S:cur	Status of the B2TT link	
dhc_crc_error_cnt_0:VALUE:cur	Counter of the CRC errors for DHE0	
dhc_crc_error_cnt_1:VALUE:cur	Counter of the CRC errors for DHE1	
dhc_crc_error_cnt_2:VALUE:cur	Counter of the CRC errors for DHE2	
dhc_crc_error_cnt_3:VALUE:cur	Counter of the CRC errors for DHE3	
DDR3 FIFO monitoring		
dhc_ddr3_ready:S:cur	Initialization of the DDR3 controller finished	
dhc_ddr3_fill_level_0:VALUE:cur	Fill level of the DDR3 FIFO 0	
dhc_ddr3_fill_level_1:VALUE:cur	Fill level of the DDR3 FIFO 1	
dhc_ddr3_fill_level_2:VALUE:cur	Fill level of the DDR3 FIFO 2	
dhc_ddr3_fill_level_3:VALUE:cur	Fill level of the DDR3 FIFO 3	
dhc_ddr3_fill_level_4:VALUE:cur	Fill level of the DDR3 FIFO 4	
dhc_ddr3_fill_level_5:VALUE:cur	Fill level of the DDR3 FIFO 5	
dhc_ddr3_mem0_full:S:cur	DDR3 FIFO 0 full flag	
dhc_ddr3_mem1_full:S:cur	DDR3 FIFO 1 full flag	
dhc_ddr3_mem2_full:S:cur	DDR3 FIFO 2 full flag	
dhc_ddr3_mem3_full:S:cur	DDR3 FIFO 3 full flag	
dhc_ddr3_mem4_full:S:cur	DDR3 FIFO 4 full flag	
dhc_ddr3_mem5_full:S:cur	DDR3 FIFO 5 full flag	
dhc_dhc_busy:S:cur	Trigger FIFO full	
dhc_dhe_channel_up:S:cur	Channel up flag for the DHE links	
dhc_dhe_sync:S:cur	DHE receiver synchronized flag	
dhc_hlt_busy:S:cur	HLT generator busy	
dhc_onsen_channel_up:S:cur	Channel up flag for the ONSEN links	
dhc_rec_trg_cnt:CNT:cur	Counter of the processed triggers in the DHC	
dhc_refclk_sel:S:cur	Source of the reference clock. ’0’ - local oscillator, ’1’ - B2TT	
dhc_trg_busy:S:cur		

Name	**D**escription	**R**ange
dhc_trg_cnt:VALUE:cur		
dhc_trigger_rate:VALUE:set	Desired trigger rate in Hz for the pulse trigger generator	1-(2^{32} – 1)
trg_mismatch_dhc:VALUE:cur		
trg_mismatch_dhe0:VALUE:cur		
trg_mismatch_dhe1:VALUE:cur		
trg_mismatch_error:S:cur		
dhc_dhe_mask:VALUE:set		
dhc_first_row:VALUE:set		
dhc_hit_distance:VALUE:set		
dhc_ipbtrigger:S:set		
dhc_ipbtrigger_en:S:set		
dhc_memdump:S:set	Send memory dump command to DHEs	0-1
dhc_number_triggers:VALUE:set		
dhc_onsen_busy_mask:S:set		
dhc_trigger_interval:VALUE:set		
dhc_udp_busy_mask:S:set		
dhc_hlt_busy_mask:S:set		

UDP data link to DQM

dhc_udp_dst_ip:VALUE:set	PC IP address in the form 0xAABBCCDD = AA.BB.CC.DD	0-(2^{32} – 1)
dhc_udp_dst_port:VALUE:set	PC UDP port	0-65535
dhc_udp_src_ip:VALUE:set	DHC IP address in the form 0xAABBCCDD = AA.BB.CC.DD	0-(2^{32} – 1)
dhc_udp_src_port:VALUE:set	DHC UDP port	0-65535
mcs_array:A:set	PV for uploading the firmware update in the MCS format	

C. JTAG Bitstream Format

Figure .1 shows composition of the JTAG bitstream. The bitstream consists of a header, an arbitrary number of write, read, and mode set instructions, and an end of stream instruction. Table .4 lists bitstream commands.

The read, write, and mode set instructions have following format:

$$\text{0xII 0xLLLLLLLL 0xDD..0xDD,} \tag{.4}$$

where **0xII** is an 8-bit instruction code, **0xLLLLLLLL** is a 32-bit data length in bits, and **0xDD** are data. Data bit length and data are transmitted as **little endian**.

Instruction	Description	Instruction code
Header	Indicates start of stream	**0x00 0x00**
Mode set	Drive TMS line	**0x02**
Write	Shift data to TDI	**0x03**
Read	Shift data to TDI, capture TDO	**0x04**
End of stream	Indicates end of stream	**0x07 0x00 0x00**

Table .4.: Bitstream instructions

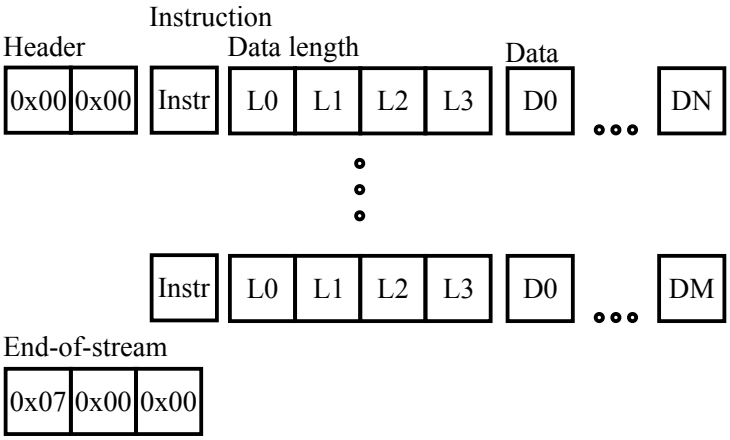


Figure .1.: Composition of the JTAG bitstream

D. Parameters of the EPICS IOC

The start script for the EPICS IOC typically consists of following elements:

1. Define macros
2. Load and register the type definitions
3. Initialize of the EPICS modules. DHE IOC uses following modules:
   - **Logger**: communication with the central logging facility
   - **devIpBus**: direct access to the IpBus registers
   - **devDHE**: special routines which access IpBus registers, for example resets, I2C
   - **devACE**: JTAG controller
4. Load data base files with the PV definitions
5. Start the IOC

It is useful to define macros at the beginning of the script. The user should define every parameter which is used several times in the script. This eliminates inconsistencies and allows users fast modification of the script. The macro definition uses following syntax:

```
epicsEnvSet("macroName", "macroValue")
```

The loading and registering of the type definitions is usually done by the following command and does not need to be modified:

```
dbLoadDatabase("dbd/dhhioc.dbd")
dhhioc_registerRecordDeviceDriver(pdbbase)
```

Module initialization is done with the followind syntax:

```
devIpBusConfigurePort       "ipbusID"       "file://pathToConnectionXML" "priority"
devACEPortDriverConfigure   "jtag:ipbusID"  "file://pathToConnectionXML" "numberOfSwitchers" \
                                            "switcherDHPT" "DCDVersion"
devDHEConfigure             "dhe:ipbusID"   "file://pathToConnectionXML"
```

Table .5 describes the parameters.

Name	Description	Possible values
ipbusID	ID of the IPbus slave as defined in connection.xml	
priority	EPICS processing priority	0 – priority 50 (medium) 1-90
numberOfSwitchers	Number of switcher ASICs on the detector module	0-6
switcherDHPT	Position of the DHPT which controls switchers. Counting starts from the DHPT which receives TDI signal from FPGA	1-4
DCDVersion	Version of the DCD ASICs installed on the detector module. The DCD version also defines the version of the DHP ASIC supported by the module	2 – DCD v. 2, DHP v 0.2 3 – DCD v. 3, DHPT, includes workaround for DCD JTAG bug 41 – DCD v. 4.1, DHPT 42 – DCD v. 4.2, DHPT

Table .5.: Parameters of the EPICS modules

E. Format of the UDP Frames

The DHE lab firmware and the DHC firmware splits large frames into chunks with the maximum size of the MTU setting of the network interface. The splitter core sends the first chunk unchanged. For the second and the following chunks, the core prepends a header of the form **0xFFFFXXXX**, where **XXXX** is the consecutive chunk number. The numbering of the chunks is necessary for frame assembly in the PC, because the UDP protocol does not guarantee the order of the frame reception. Figure .2 shows the example of a frame which is split in three chunks.

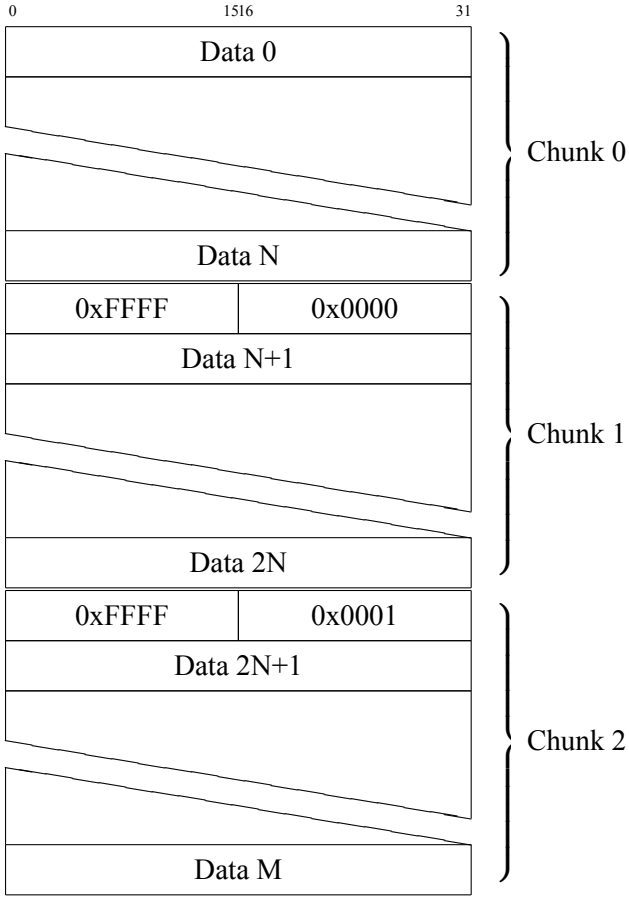


Figure .2.: Example of the frame splitting

F. Schematics of the Current Mirror

Figure .3 shows the schematics of the precision current source on the DHH AMC module v. 3.

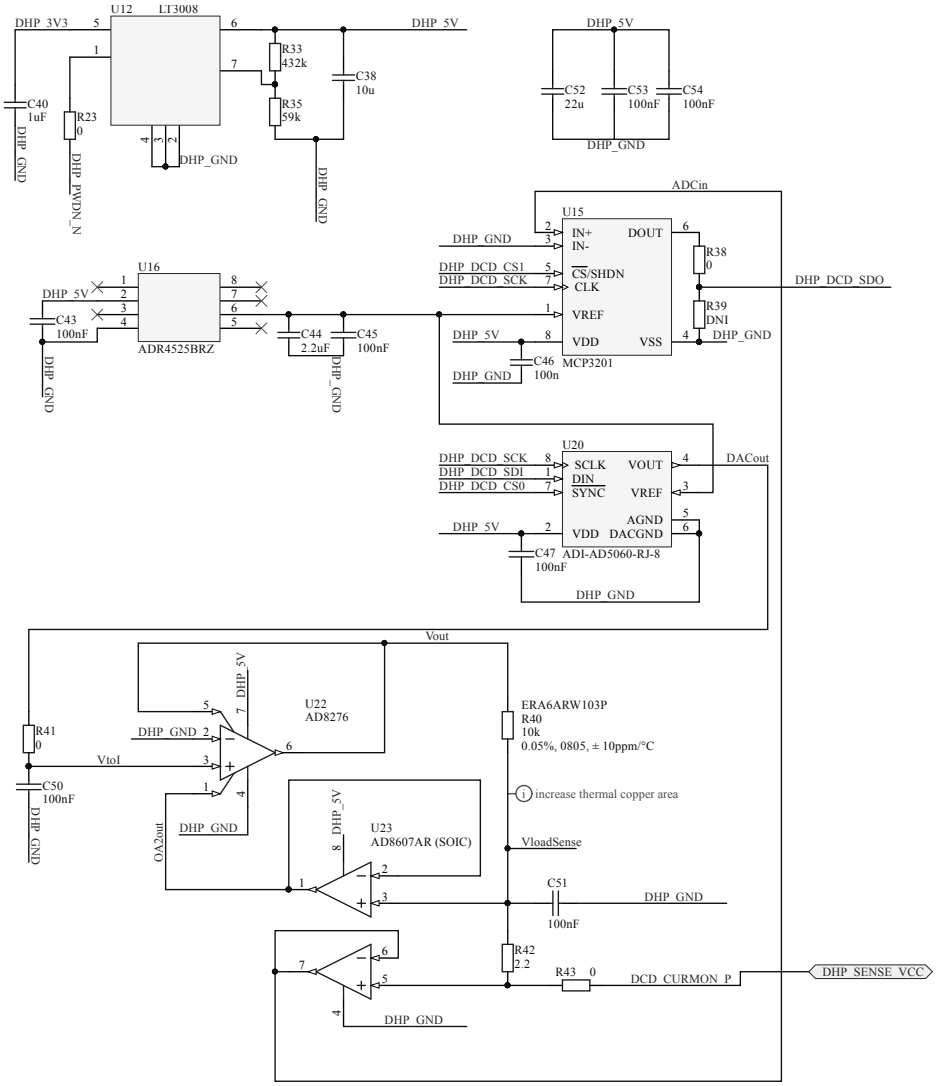


Figure .3.: Schematics of the precision current source

G. Format of the UCF Trigger Frame

Figure .4 shows the format of the trigger frame. The trigger frame contains following signals:

- Trigger, **T**
- Memory dump, **MD**: triggers memory dump of DHPT
- Long reset from slow control, **ILR**
- Short reset from slow control, **ISR**
- Front-end reset from B2TT, **FR**
- Run reset from B2TT, **RR**
- Revolution cycle from B2TT, **RV**: signal repeats every 10 μs to synchronize the read-out system with the accelerator cycle
- Every second revolution cycle, **RV2**: signal is derived from the REVO to be synchronous with the pixel detector cycle. The signal repeats every 20 μs. The reference time for this signal is the run reset
- Run reset synchronized to the revolution cycle signal, **SR**
- Event number
- Bits reserved for future use, **Res.**: tied to ground

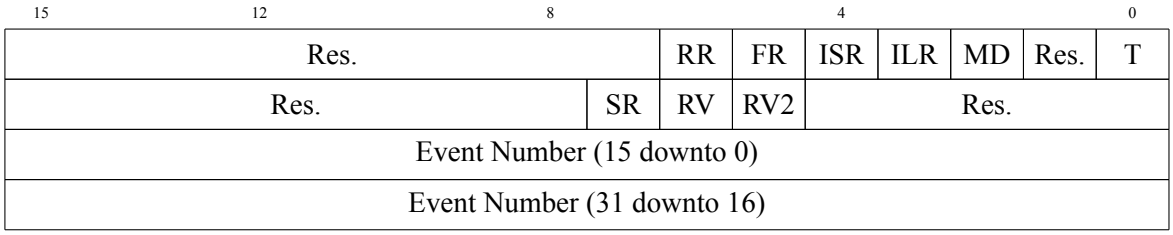


Figure .4.: Format of the UCF trigger frame

H. ATCA Carrier Board and Rear Transition Module

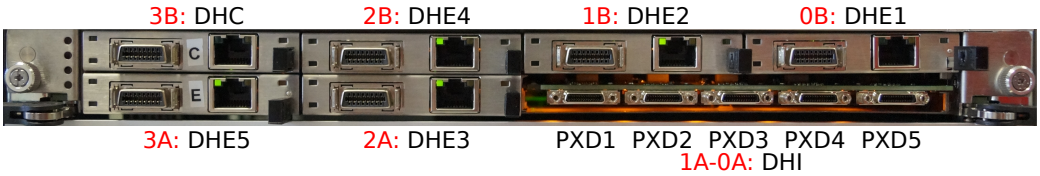


Figure .5.: Logical assignment of the functional modules in the ATCA carrier board. Slot numbers are written in red

Figure .5 shows logical assignment of the DHE, DHC, and DHI modules in the ATCA carrier board. Their positions are fixed, because routing of the signals between the slots determines functionality of the module, which is inserted into the slot. Indices of the DHE modules correspond to numbering of UCF links in the DHC. Indices of the Camera Link interfaces on the DHI correspond to UDP port of the IPbus instances. DHI-only control listens at port 50001. A control core, which controls detector PXD**N**, listens at port **50001**+**N**.

There are 8 jumpers on the carrier board that are used to keep JTAG chain functional, if one or more AMC slots are empty on the carrier board. These jumpers shorten TDI and TDO line of the empty slots. Table .6 lists mapping between jumpers and AMC slots.

AMC slot	Jumper
0A	J6
0B	J7
1A	J8
1B	J9
2A	J10
2B	J11
3A	J12
3B	J13

Table .6.: Mapping between AMC slots and jumpers

Figure .6 shows rear-transition module with external interfaces highlighted. These are

1. Digilent JTAG programmer with micro-USB interface,
2. RJ45 connector for B2TT interface to DHC,
3. outer 12-channel Avago miniPOD for receiving data from detectors. Connected to DHE4 and DHE5,
4. inner 12-channel Avago miniPOD for receiving data from detectors. Connected to DHE1, DHE2 and DHE3,
5. IPbus Ethernet transceiver, connected to DHC,

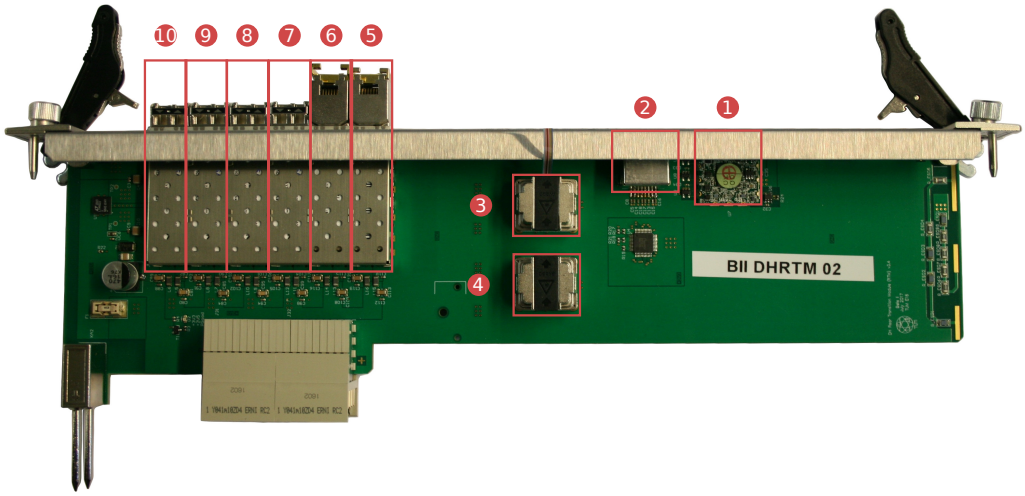


Figure .6.: Rear-transition module

6. Ethernet transceiver for UDP data link to local DAQ, connected to DHC,
7. link 1 to ONSEN,
8. link 2 to ONSEN,
9. link 3 to ONSEN, and
10. link 4 to ONSEN.

Table .7 lists mapping of optical links in the miniPODs to DHPT links in DHEs

miniPOD	Optical link	DHE	DHPT link in DHE
inner	0	1	1
	1	2	4
	2	1	3
	3	2	2
	4	2	1
	5	1	4
	6	2	3
	7	1	2
	8	3	1
	9	3	2
	10	3	3
	11	3	4
outer	0	4	1
	1	4	2
	2	4	3
	3	4	4
	8	5	1
	9	5	2
	10	5	3
	11	5	4

Table .7.: Mapping of the optical links in the Avago miniPODs to DHPT links in the DHE

Bibliography

> We don't know who this Wolfgang is, but the documentation says that the option was "required for Wolfgang's PhD thesis"
>
> "Local Guide to BibLaTeX" explaining the --wolfgang flag

[1] M. Tanabashi *et al.*, "Review of particle physics," *Phys. Rev. D*, vol. 98, p. 030001, Aug 2018. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevD.98.030001

[2] M. Aguilar *et al.*, "Antiproton flux, antiproton-to-proton flux ratio, and properties of elementary particle fluxes in primary cosmic rays measured with the alpha magnetic spectrometer on the international space station," *Phys. Rev. Lett.*, vol. 117, p. 091103, Aug 2016. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.117.091103

[3] Ade, P. A. R. *et al.*, "Planck 2015 results - XIII. Cosmological parameters," *A&A*, vol. 594, p. A13, 2016. [Online]. Available: https://doi.org/10.1051/0004-6361/201525830

[4] A. D. Sakharov, "Violation of CP Invariance, C asymmetry, and baryon asymmetry of the universe," *Pisma Zh. Eksp. Teor. Fiz.*, vol. 5, pp. 32–35, 1967, [Usp. Fiz. Nauk161,no.5,61(1991)].

[5] T. Nanut *et al.*, "Observation of $D^0 \to \rho^0\gamma$ and search for cp violation in radiative charm decays," *Phys. Rev. Lett.*, vol. 118, p. 051801, Jan 2017. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.118.051801

[6] T. D. Lee and C. N. Yang, "Question of parity conservation in weak interactions," *Phys. Rev.*, vol. 104, pp. 254–258, Oct 1956. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.104.254

[7] C. S. Wu *et al.*, "Experimental test of parity conservation in beta decay," *Phys. Rev.*, vol. 105, pp. 1413–1415, Feb 1957. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.105.1413

[8] R. L. Garwin, L. M. Lederman, and M. Weinrich, "Observations of the failure of conservation of parity and charge conjugation in meson decays: the magnetic moment of the free muon," *Phys. Rev.*, vol. 105, pp. 1415–1417, Feb 1957. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.105.1415

[9] E. C. G. Sudarshan and R. E. Marshak, "The nature of the four-fermion interaction," *Proc. of the Conference on Mesons and Newly-Discovered Particles, Padua-Venice*, Sept 1957.

[10] R. P. Feynman and M. Gell-Mann, "Theory of the fermi interaction," *Phys. Rev.*, vol. 109, pp. 193–198, Jan 1958. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.109.193

[11] N. Cabibbo, "Unitary symmetry and leptonic decays," *Phys. Rev. Lett.*, vol. 10, pp. 531–533, Jun 1963. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.10.531

[12] M. Kobayashi and T. Maskawa, "CP-Violation in the Renormalizable Theory of Weak Interaction," *Progress of Theoretical Physics*, vol. 49, no. 2, pp. 652–657, 1973. [Online]. Available: http://dx.doi.org/10.1143/PTP.49.652

[13] S. W. Herb *et al.*, "Observation of a dimuon resonance at 9.5 gev in 400-gev proton-nucleus collisions," *Phys. Rev. Lett.*, vol. 39, pp. 252–255, Aug 1977. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.39.252

[14] C. Jarlskog, "Commutator of the quark mass matrices in the standard electroweak model and a measure of maximal CP nonconservation," *Phys. Rev. Lett.*, vol. 55, pp. 1039–1042, Sep 1985. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.55.1039

[15] J. Charles *et al.*, "CP violation and the CKM matrix: assessing the impact of the asymmetric B factories," *The European Physical Journal C - Particles and Fields*, vol. 41, no. 1, pp. 1–131, May 2005. [Online]. Available: https://doi.org/10.1140/epjc/s2005-02169-1

[16] G. W. S. Hou, "Source of CP Violation for the Baryon Asymmetry of the Universe," in *Proceedings, 34th International Conference on High Energy Physics (ICHEP 2008): Philadelphia, Pennsylvania, July 30-August 5, 2008*, 2008.

[17] Y. Amhis *et al.*, "Averages of b-hadron, c-hadron, and τ-lepton properties as of summer 2016," *Eur. Phys. J.*, vol. C77, p. 895, 2017, updated results and plots available at https://hflav.web.cern.ch.

[18] J. Brod *et al.*, "Size of direct *CP* violation in singly Cabibbo-suppressed *D* decays," *Phys. Rev. D*, vol. 86, p. 014023, Jul 2012. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevD.86.014023

[19] Y. Grossman, A. L. Kagan, and J. Zupan, "Testing for new physics in singly Cabibbo suppressed D decays," *Physical Review D*, vol. 85, no. 11, p. 114036, 2012.

[20] Y. Hochberg and Y. Nir, "Relating Direct *CP* Violation in *D* Decays and the Forward-Backward Asymmetry in $t\bar{t}$ Production," *Phys. Rev. Lett.*, vol.

108, p. 261601, Jun 2012. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.108.261601

[21] G. Buchalla, A. J. Buras, and M. E. Lautenbacher, "Weak decays beyond leading logarithms," *Rev. Mod. Phys.*, vol. 68, pp. 1125–1244, Oct 1996. [Online]. Available: https://link.aps.org/doi/10.1103/RevModPhys.68.1125

[22] D. Besson and T. Skwarnicki, "Upsilon spectroscopy: Transitions in the bottomonium system," *Annual Review of Nuclear and Particle Science*, vol. 43, no. 1, pp. 333–378, 1993. [Online]. Available: https://doi.org/10.1146/annurev.ns.43.120193.002001

[23] A. Abashian *et al.*, "The Belle detector," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 479, no. 1, pp. 117 – 232, 2002, detectors for Asymmetric B-factories. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168900201020137

[24] H. Aihara *et al.*, "Belle SVD2 vertex detector," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 568, no. 1, pp. 269 – 273, 2006, new Developments in Radiation Detectors. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168900206011193

[25] D. J. Lange, "The EvtGen particle decay simulation package," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 462, no. 1, pp. 152 – 155, 2001, bEAUTY2000, Proceedings of the 7th Int. Conf. on B-Physics at Hadron Machines. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168900201000894

[26] R. Brun *et al.*, "GEANT: Detector description and simulation tool," CERN, Tech. Rep., 1993.

[27] G. C. Fox and S. Wolfram, "Observables for the analysis of event shapes in e^+e^- annihilation and other processes," *Phys. Rev. Lett.*, vol. 41, pp. 1581–1585, Dec 1978. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.41.1581

[28] H. Nakano *et al.*, "K_S^0 selection with NeuroBayes and nisKsFinder class (Belle Note 1253 ver. 1)," Tech. Rep., 2012.

[29] J.-I. Tanaka, "Kinematic Fitting. Belle Note 194," Tech. Rep., 2000.

[30] A. Hoecker *et al.*, "TMVA-Toolkit for multivariate data analysis," *arXiv preprint physics/0703039*, 2007.

[31] S. Barlag *et al.*, "A reanalysis of branching fractions of charmed mesons D0, D+ and Ds+," *Zeitschrift für Physik C Particles and Fields*, vol. 55, no. 3, pp. 383–390, Sep 1992. [Online]. Available: https://doi.org/10.1007/BF01565095

[32] A. Caldwell *et al.*, "BAT–The Bayesian analysis toolkit," *Computer Physics Communications*, vol. 180, no. 11, pp. 2197–2209, 2009.

[33] M. Oreglia, "A Study of the Reactions $\psi' \to \gamma\gamma\psi$," Ph.D. dissertation, SLAC, 1980. [Online]. Available: http://www-public.slac.stanford.edu/sciDoc/docMeta.aspx?slacPubNumber=slac-r-236.html

[34] J. M. Link *et al.*, "A Measurement of the Branching Ratios of D^+ and D_s^+ Hadronic Decays to Four-Body Final States Containing a K_S," *Phys. Rev. Lett.*, vol. 87, p. 162001, 2001.

[35] S. Ryu, "Study of π^0 efficiency using $\tau^- \to \pi^-\pi^0\nu_\tau$ (Belle Note 1224)," Tech. Rep., 2012.

[36] B. Bhuyan, "High P_T Tracking Efficiency Using Partially Reconstructed D^* Decays (Belle Note 1165)," Tech. Rep., 2010.

[37] Y. Ohnishi *et al.*, "Accelerator design at SuperKEKB," *Progress of Theoretical and Experimental Physics*, vol. 2013, no. 3, p. 03A011, 2013. [Online]. Available: http://dx.doi.org/10.1093/ptep/pts083

[38] Y. Funakoshi *et al.*, "Recent Progress of KEKB," in *Particle accelerator. Proceedings, 23rd Conference, PAC'09, Vancouver, Canada, May 4-8, 2009*, 2010, p. WE6PFP043. [Online]. Available: http://accelconf.web.cern.ch/AccelConf/PAC2009/papers/we6pfp043.pdf

[39] S. Collaboration, "SuperB: A High-Luminosity Asymmetric e+ e-Super Flavor Factory. Conceptual Design Report," *arXiv preprint arXiv:0709.0451*, 2007.

[40] N. Ohuchi *et al.*, "Design and Construction of the SuperKEKB QC1 Final Focus Superconducting Magnets," *IEEE Transactions on Applied Superconductivity*, vol. 25, no. 3, pp. 1–4, June 2015.

[41] T. Abe *et al.*, "Belle II Technical Design Report," *ArXiv e-prints*, Nov. 2010.

[42] N. Taniguchi, "Central Drift Chamber for Belle-II," *Journal of Instrumentation*, vol. 12, no. 06, p. C06014, 2017. [Online]. Available: http://stacks.iop.org/1748-0221/12/i=06/a=C06014

[43] M. Tabata *et al.*, "Silica aerogel radiator for use in the A-RICH system utilized in the Belle II experiment," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 766, pp. 212 – 216, 2014, rICH2013 Proceedings of the Eighth International Workshop on Ring Imaging Cherenkov Detectors Shonan, Kanagawa, Japan, December 2-6, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168900214004264

[44] S. Iwata *et al.*, "Particle identification performance of the prototype aerogel RICH counter for the Belle II experiment," *Progress of Theoretical and Experimental Physics*, vol. 2016, no. 3, p. 033H01, 2016. [Online]. Available: http://dx.doi.org/10.1093/ptep/ptw005

[45] T. Iijima *et al.*, "A novel type of proximity focusing RICH counter with multiple refractive index aerogel radiator," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 548, no. 3, pp. 383 – 390, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168900205010843

[46] Y. Maeda, "Status of installation and commissioning for the Belle II time-of-propagation counter," *Journal of Instrumentation*, vol. 12, no. 08, p. C08005, 2017. [Online]. Available: http://stacks.iop.org/1748-0221/12/i=08/a=C08005

[47] B. Shwartz *et al.*, "Electromagnetic calorimeter of the Belle II detector," *Journal of Physics: Conference Series*, vol. 928, no. 1, p. 012021, 2017. [Online]. Available: http://stacks.iop.org/1742-6596/928/i=1/a=012021

[48] J. G. Wang, "RPC performance at KLM/BELLE," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 508, no. 1, pp. 133 – 136, 2003, proceedings of the Sixth International Workshop on Resistive Plate Chambers and Related Detectors. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168900203013354

[49] T. Aushev *et al.*, "A scintillator based endcap KL and muon detector for the Belle II experiment," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 789, pp. 134 – 142, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016890021500385X

[50] Z. Drasal and K. Prothmann, "Optimization of the Belle II vertex detector," *PoS*, vol. HQL2012, p. 073, 2012.

[51] A. Moll *et al.*, *Belle II Note Number 0010: The vertex detector numbering scheme*, September 2016.

[52] L. Vitale, "The Belle II silicon-strip vertex detector," *PoS*, p. 017, 2015.

[53] G. B. Mohanty, "Belle II Silicon Vertex Detector," *Nucl. Instrum. Meth.*, vol. A831, pp. 80–84, 2016.

[54] A. Airex, "Airex® t92 easy processing structural foam, data sheet 01.2016 (gf-tds-021)."

[55] L. Jones, *APV25-S1: User guide version 2.2*. Chilton: RAL Microelectronics Design Group, 2001. [Online]. Available: https://cds.cern.ch/record/1069892

[56] D. Moya and I. Vila, “Structural and environmental monitoring of tracker and vertex systems using Fiber Optic Sensors,” Tech. Rep. arXiv:1203.0109, Mar 2012, comments: LCWS11 International Workshop in Future Linear colliders. [Online]. Available: https://cds.cern.ch/record/1428663

[57] J. Kemmer and G. Lutz, “New detector concepts,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 253, no. 3, pp. 365 – 377, 1987.

[58] T. B. I. pixel detector collaboration, *The PXD Whitebook*, June 2017.

[59] H. Ye *et al.*, “Thermal mock-up studies of the DEPFET pixel vertex detector for Belle II,” 2016.

[60] M. Koch, “Development of a Test Environment for the Characterization of the Current Digitizer Chip DCD2 and the DEPFET Pixel System for the Belle II Experiment at SuperKEKB,” Ph.D. dissertation, Bonn U., 2011. [Online]. Available: http://inspirehep.net/record/1231385/files/CERN-THESIS-2011-084.pdf

[61] I. Peric *et al.*, “DCDB and SWITCHERB, the readout ASICS for Belle II DEPFET pixel detector,” in *Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2011 IEEE*, Oct 2011, pp. 1536–1539.

[62] ——, “DCD - The Multi-Channel Current-Mode ADC Chip for the Readout of DEPFET Pixel Detectors,” *IEEE Transactions on Nuclear Science*, vol. 57, no. 2, pp. 743–753, April 2010.

[63] M. Lemarenko and L. Germic, *DHPT 1.0 Manual*, October 2014.

[64] Xilinx, *LogiCORE IP Aurora 8B/10B v6.2, User Guide 766*.

[65] R. Casanova, O. Alonso, and Ángel Diéguez, *TOPUB*, January 2015.

[66] T. Higuchi *et al.*, “Modular pipeline readout electronics for the SuperBelle drift chamber,” *IEEE Transactions on Nuclear Science*, vol. 52, no. 5, pp. 1912–1917, Oct 2005.

[67] S. Y. Suzuki *et al.*, “The Three-Level Event Building System for the Belle II Experiment,” *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 1162–1168, June 2015.

[68] P. I. C. M. Group, “Advancedtca® base specification,” Tech. Rep. 3.0, April 2015.

[69] M. Nakao, “Timing distribution for the Belle II data acquistion system,” *Journal of Instrumentation*, vol. 7, no. 01, p. C01028, 2012.

[70] M. Schnell, “Development of an FPGA-based Data Reduction System for the Belle II DEPFET Pixel Detector,” Ph.D. dissertation, Bonn U., 2015.

[71] B. Spruck *et al.*, "The Belle II pixel detector data acquisition and reduction system," *Nuclear Science, IEEE Transactions on*, vol. 60, no. 5, pp. 3709–3713, 2013.

[72] Xilinx, *Virtex-6 Family Overview, Data Sheet 150*.

[73] I. Perić, *Production ASICs: DCDB4.1 and DCDB4.2 Reference Manual*, May 2016.

[74] Xilinx, *Virtex-6 FPGA Data Sheet: DC and Switching Characteristics, DS 152*.

[75] A. I. Association, *Camera Link: Specifications of the Camera Link Interface Standard for Digital Cameras and Frame Grabbers*. Automated Imaging Association, 2007. [Online]. Available: https://books.google.de/books?id=9LP1ZwEACAAJ

[76] L. Bosisio *et al.*, "SVD Radiation and Environmental Monitoring: General Requirements, Belle II Note 43," May 2015.

[77] J. Troska *et al.*, "Versatile transceiver and transmitter production status," *Journal of Instrumentation*, vol. 8, no. 12, p. C12030, 2013. [Online]. Available: http://stacks.iop.org/1748-0221/8/i=12/a=C12030

[78] S. R. Dejong, "Study of thermal neutron flux from SuperKEKB in the Belle II commissioning detector," Ph.D. dissertation, 2017.

[79] H. Perrey, "An EUDET / AIDA pixel beam telescope for detector development," in *Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2012 IEEE*, 2012, pp. 1996–1998.

[80] B. Spruck *et al.*, "PXD DAQ Data Formats," May 2016.

[81] A. Fiergolski and P. Fall, "1G eth UDP / IP Stack." [Online]. Available: https://opencores.org/project,udp_ip_stack

[82] D. Gaisbauer *et al.*, "Unified communication framework," *IEEE Transactions on Nuclear Science*, vol. 64, no. 10, pp. 2761–2764, Oct 2017.

[83] S. Baehr *et al.*, "Online-Analysis of Hits in the Belle-II Pixeldetector for Separation of Slow Pions from Background," *Journal of Physics: Conference Series*, vol. 664, no. 9, p. 092001, 2015. [Online]. Available: http://stacks.iop.org/1742-6596/664/i=9/a=092001

[84] Gottwald, Martin, "Evaluation of the new Clustering Algorithm for Belle II," Bachelor's Thesis, Technische Universität München, 2013.

[85] Xilinx, *Virtex-6 FPGA Memory Interface Solutions, User Guide 406*.

[86] I. Konorov *et al.*, "SODA: Time distribution system for the PANDA experiment," in *2009 IEEE Nuclear Science Symposium Conference Record (NSS/MIC)*, Oct 2009, pp. 1863–1865.

[87] D. Gaisbauer *et al.*, "Self-triggering readout system for the neutron lifetime experiment PENeLOPE," *Journal of Instrumentation*, vol. 11, no. 02, p. C02068, 2016. [Online]. Available: http://stacks.iop.org/1748-0221/11/i=02/a=C02068

[88] Xilinx, *Virtex-6 FPGA GTX Transceivers, User Guide 366*.

[89] ARM, *AMBA 4 AXI4-Stream Protocol*, 1st ed.

[90] L. R. Dalesio, J. O. Hill, M. Kraimer, S. Lewis, D. Murray, S. Hunt, W. Watson, M. Clausen, and J. Dalesio, "The experimental physics and industrial control system architecture: past, present, and future," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 352, no. 1-2, pp. 179–184, 1994.

[91] R. Frazier *et al.*, "Software and firmware for controlling CMS trigger and readout hardware via gigabit Ethernet," *Physics Procedia*, vol. 37, no. 0, pp. 1892 – 1899, 2012, proceedings of the 2nd International Conference on Technology and Instrumentation in Particle Physics (TIPP 2011).

[92] R. Finlayson *et al.*, "A Reverse Address Resolution Protocol," Internet Requests for Comments, RFC Editor, RFC 903, June 1984. [Online]. Available: http://www.rfc-editor.org/rfc/rfc903.txt

[93] Xilinx, *Virtex-6 FPGA Configuration User Guide, UG360*.

[94] R. Herveille *et al.*, "Wishbone system-on-chip (soc) interconnection architecture for portable ip cores," *OpenCores Organization*, 2002.

[95] "Common flash interface (cfi)," JEDEC, Standard, September 2003.

[96] R. Herveille", *"I^2C-Master Core Specification"*, July 2003.

[97] M. R. Kraimer *et al.*, "EPICS: Asynchronous driver support," in *Proc. Int. Conf. Accelerator and Large Experimental Physics Control Systems*, 2009, pp. 074–5.

[98] S. Labs, *Si5338 I^2C-Programmable Any-Frequency, Any-Output Quad Clock Generator*.

[99] Intel, *Hexadecimal Object File Format Specification*, 1 1988.

[100] "Ieee standard for test access port and boundary-scan architecture," *IEEE Std 1149.1-2013 (Revision of IEEE Std 1149.1-2001)*, pp. 1–444, May 2013.

[101] J. Hatje *et al.*, "Control system studio (css)," 01 2007.

[102] X. H. Chen and K. U. Kasemir, "BOY, A Modern Graphical Operator Interface Editor and Runtime," *Conf. Proc.*, vol. C110328, pp. 1404–1406, 2011.

[103] R. Diener *et al.*, "The DESY II Test Beam Facility," 2018.

[104] H. Jansen *et al.*, "Performance of the EUDET-type beam telescopes," *EPJ Techniques and Instrumentation*, vol. 3, no. 1, p. 7, Oct 2016. [Online]. Available: https://doi.org/10.1140/epjti/s40485-016-0033-2

[105] T. Bilka *et al.*, "Demonstrator of the Belle II Online Tracking and Pixel Data Reduction on the High Level Trigger System," *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 1155–1161, June 2015.

[106] T. Geßler *et al.*, "The ONSEN Data Reduction System for the Belle II Pixel Detector," *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 1149–1154, June 2015.

[107] F. Müller, "Characterization and optimization of the prototype DEPFET modules for the Belle II Pixel Vertex Detector," July 2017. [Online]. Available: http://nbn-resolving.de/urn:nbn:de:bvb:19-210714

[108] P. M. Lewis *et al.*, "First Measurements of Beam Backgrounds at SuperKEKB," 2018.

ACKNOWLEDGEMENTS

First, I would like to thank my supervisors, Igor Konorov and Daniel Greenwald, who guided me through my thesis. Thank you, Igor, for always supporting me and teaching me about electronics. But the most important thing that I learned from you is to stay calm and do not panic. I enjoyed a lot working together with you. Спасибо Вам!

I am very grateful to Daniel for your scientific guidance and, especially, your help with my thesis. I enjoyed our discussions about physics and cooking. Thank you for always subtly pushing me to improve myself.

Next, I would like to thank Stefan Huber for being my friend and a good colleague. Thanks to your work, I had time to work on my analysis.

I would like to acknowledge my colleagues in the DEPFET collaboration, Florian Lütticke and Botho Paschen. Through work with you, I learned a great deal of how DEPFET really works. I have also found good friends in the Giessen group. Thank you, Thomas Geßler and Klemens Lautenbach-san, Florian and Botho, for having fun time together.

A big thank you to Danny van Dyk for explaining me theory behind isospin sum rules.

These acknowledgemnets would be incomplete without thanking E18. I find that our chair is a great place to work. I would like to thank professor Paul and all my colleagues for comfortable and scientific atmosphere at the chair. And a special thank you to Karin Frank for making our daily lives a bit easier.

Я хотел бы поблагодарить мою семью за поддержку все эти годы. Я благодарен моим родителям за принятое ими когда-то решение оставить все и переехать в Германию. Я надеюсь, что оправдал Ваши ожидания.

И, наконец, Света! Спасибо тебе за твою поддержку моей работы и твою любовь. Спасибо тебе за то, что всегда ждала меня дома, когда мне приходилось надолго уезжать. Я тебя тоже очень люблю и ценю!