DEVELOPMENT OF DATA PROCESSING ALGORITHMS FOR THE UPGRADED LHCb VERTEX LOCATOR

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER FOR THE DEGREE OF MASTER OF SCIENCE BY RESEARCH IN THE FACULTY OF SCIENCE AND ENGINEERING

2016

Toby Seán Nonnenmacher

SCHOOL OF PHYSICS AND ASTRONOMY
THE UNIVERSITY OF MANCHESTER
# Contents

Abstract 9

1 Introduction 14

2 LHCb 16
   2.1 The LHC ......................................................... 16
   2.2 LHCb .............................................................. 19

3 The LHCb Upgrade 22
   3.1 Upgraded components of the LHCb experiment ............... 22
      3.1.1 The Vertex Locator ........................................ 22
      3.1.2 The Dipole Magnet .......................................... 25
      3.1.3 The Tracking System ...................................... 25
      3.1.4 The Ring Imaging Cherenkov Detectors .................. 27
      3.1.5 The Calorimeter System .................................. 29
      3.1.6 The Muon Detectors ....................................... 31
      3.1.7 The Trigger .................................................. 33
   3.2 The LHCb VELO upgrade ................................... 34
      3.2.1 The current VELO ............................................ 35
      3.2.2 The VELO upgrade modules .............................. 37
      3.2.3 VeloPix: the VELO upgrade ASIC ....................... 38
      3.2.4 The VELO upgrade microchannel substrate ............. 40
      3.2.5 The VELO upgrade DAQ board ............................ 40
      3.2.6 The MiniDAQ-1 board ...................................... 41

4 Data Scrambling 43
   4.1 Motivation for scrambling algorithms ....................... 43
List of Figures

2.1 The CERN accelerator complex. Figure taken from Reference [2]. . 17

2.2 The LHC long term schedule. Long Shutdown II, during which
LHCb will see a major upgrade, is scheduled to begin in December
2018. Data taking is then scheduled to continue in 2021. Figure
taken from Reference [3]. . . . . . . . . . . . . . . . . . . . . . . . . 18

2.3 Cross-sectional view of the LHCb detector and its sub-detectors
along the x = 0 plane. Figure taken from Reference [6]. . . . . . . 20

3.1 Cross-sectional view of the upgraded LHCb detector and its sub-
detectors along the x = 0 plane. Figure taken from Reference [12]. . 23

3.2 Scale view of the LHCb magnet with units in mm. The interaction
point is located behind the magnet. Figure taken from Reference [14]. 24

3.3 Layout of the TT, showing its dimensions and the geometry of the
layers. Figure taken from Reference [15]. . . . . . . . . . . . . . . . 26

3.4 Cross-section of the straw tube module with with units in mm.
Figure taken from Reference [16]. . . . . . . . . . . . . . . . . . . . . . 27

3.5 Cross-section of the RICH1 detector in the y-z plane. Figure taken
from Reference [14]. . . . . . . . . . . . . . . . . . . . . . . . . . . . 28

3.6 Schematic of the internal cell structure of the HCAL. Figure taken
from Reference [14]. . . . . . . . . . . . . . . . . . . . . . . . . . . . 30

3.7 Cross-sectional view of the muon system, including iron filters. The
calorimeters separate M1 from M2-5. Figure taken from Reference
[14]. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32

3.8 Chart detailing the flow of the LHCb trigger in 2012. Figure taken
from Reference [17]. . . . . . . . . . . . . . . . . . . . . . . . . . . . 33
3.9 (top) Layout of the current VELO modules along the beam line in the x-z plane. (bottom) View of current VELO sensors in the transverse plane in both the closed (left) and open (right) positions. Figure taken from Reference [14]. 

3.10 A module for the upgraded VELO. Pixel sensors are bump-bonded to 3 ASICs and cooled to below $-30\,\degree C$ with bi-phase CO$_2$ which is circulated through microchannels etched in the silicon substrate. Figure taken from Reference [19].

3.11 The VeloPix data path. Hit information is grouped into SuperPixel (SP) packets, propagated downwards, and collected in the End-of-Column (EoC). A router distributes the data equally towards the 4 output serializers. Figure taken from Reference [19].

3.12 The microchannel prototype. A silicon wafer with the etched microchannels is bonded to a Pyrex plate.

3.13 The PCIe40 Data Acquisition (DAQ) board. This configuration provides 48 serial input and 48 output links at $5.12\,\text{GB} / \text{s}$ and a bi-directional link for Timing and Fast Control (TFC) signals. The output bandwidth is $\sim 100\,\text{GB} / \text{s}$. Figure taken from Reference [19].

3.14 The MiniDAQ-1 board is a reduced version of the final PCIe40 DAQ board. It is used for the development of the acquisition software and testing of the VeloPix. With this setup, the data is generated in a prototype board, sent through GWT serializers, converted to optical, and recorded in MiniDAQ-1. Finally, the data is sent to a PC through a $10\,\text{GB} / \text{s}$ optical link. The figure is of the setup at Manchester University.

4.1 Schematic view of the 128 bits of a single GWT frame, with an extended description for one SPP. A frame of data runs from Most Significant Bit (MSB) on the left to Least Significant Bit (LSB) on the right. Each SPP is composed of a Bunch Cross ID (BCID), Super Pixel Address, and Hitmap.
4.2 Evolution of the output of the additive scrambler when supplied with a single input signal, simulated in ModelSim. The x axis represents time, with a single clock cycle having a period of 6.25 ps. A single SPP of 30 bits of output data is shown, with raised outputs corresponding to 1s and lowered outputs corresponding to 0s. With a single constant input signal, the output continues to change over more than 30 clock cycles.

4.3 Evolution of the output of the intermediate (multiplicative) scrambler when supplied with a single input signal, simulated in ModelSim. The x axis represents time, with a single clock cycle having a period of 6.25 ps. Raised and lowered output bits correspond to 1s and 0s respectively. With a single constant input signal, the output changes over two clock cycles, and then remains constant.

4.4 Section of the logic gate diagram for the additive scrambler showing the XOR logic gates. The number of inputs in each case is shown on the left of the gate. The square boxes represent registers where the signals are stored in between clock cycles. This scrambling method requires an intermediate register labelled $lfsr_q$.

4.5 Section of the logic gate diagram for the intermediate scrambler. The data input is stored in the feedbackRegister. On each rising edge of the clock, the data is sent to a set of 30 XOR logic gates with 3 inputs apiece, and then sent to the output register ($RX\_DATA\_O$).

4.6 Section of the logic gate diagram for the VeloPix scrambler showing the XOR gates. The data is sent through variable-input XOR logic gates. An example 29-input gate is shown.

4.7 Histogram of the transitions for each scrambler. The number of transitions in the data is shown and for each scrambler fitted with a Gaussian.

4.8 Graphs displaying the difference in number of 1s and 0s in every frame. The results for the three scramblers are given, top, while the results for three randomly generated sets of data are given, bottom. Additionally, calculated $1\sigma$ and $2\sigma$ bands are shown on each graph.

4.9 Histogram showing the frequency of lengths of runs of 1s and 0s respectively. Results are shown for each scrambler, the data prior to scrambling, and a set of randomly generated data.
4.10 Histogram showing the frequency of lengths of runs of 1s and 0s indiscriminately on arithmetic and logarithmic scales respectively. . 56

5.1 Schematic view of the firmware framework for the VELO Upgrade. The GWT frame is built in the LLI_GWT component, whose main functionality is to find the GWT header in the incoming bit stream and lock to it. The next block descrambles the Super Pixel Packets and performs the error checking. The Router aligns all the packets according to their BCID. The TFC processing block implements the BCID-veto signals. The next block tags with one bit those clusters without neighbours in order to save CPU time in the High Level Trigger. Finally the Event ID builds the output frame according to some rules, in order to optimize the output bandwidth. Figure taken from [19]. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 60

5.2 Oscilloscope data. The x axis denotes the number of samples, ranging from 0 to 2000 for each row. The y axis denotes the voltage of the scope, ranging from 0.4 to −0.4 volt. The first four graphs correspond to the first four 128 bit frames sampled. The fifth row shows 500 frames overlaid, leaving only the unchanging four bit header and parity clearly visible towards the right of the figure. Figure taken from [21]. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 61

5.3 Data showing the effect of a bug in the GWT. All bits other than the four at the centre of the frame have been delayed by one clock cycle. Simulated data in which the two SPPs closest to the header were identical were used to demonstrate this. Figure taken from [21]. 63

5.4 Graph showing the frequency of the number of samples prior to a transition through 0 in the oscilloscope data. The number of samples before a transition is given on the x axis. Frequency is given on the y axis on a logarithmic scale. Peaks corresponding to integer multiples of 15.625 are clearly visible. . . . . . . . . . . . . . 64

6.1 A simplified schematic of the router. The 5 MSB router is shown (left) along with the complementary 4 LSB router (right). Figure taken from [19]. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 68
List of Figures

6.2 Diagram of a 1-input by 2-output 1-bit comparator (left) and 2-input by 2-output 1-bit comparator (right). Figure taken from [19]. 69

6.3 Diagram of the 5 MSB router logic showing the $1 \times 2$ and $2 \times 2$ comparators. Figure taken from [19]. 70

6.4 Schematic of the data formatting to be introduced between the router and the MEP. The differently numbered global data correspond to different bunch crossings. Figure taken from [22]. 73

6.5 Diagram of the path travelled by the data from the GWT Serialiser to the MEP. The number of bits per frame of data and transmission clock frequency are shown at each stage. The three elements passed between the router and post router are shown: the data and the number of SPPs for a given BCID are passed from the router, and the DV signal is sent back to the router. 75
Abstract

The LHCb detector will see a major upgrade during LHC Long Shutdown II, which is planned for 2019/20. The silicon Vertex Locator subdetector will be upgraded for operation under the new run conditions. The detector will be read out using a data acquisition board based on an FPGA. The work presented in this thesis is concerned with the development of the data processing algorithms to be used in this data acquisition board. In particular, work in three different areas of the FPGA is covered: the data processing block, the low level interface, and the post router block. The algorithms produced have been simulated and tested, and shown to provide the required performance. Errors in the initial implementation of the Gigabit Wireline Transmitter serialized data in the low level interface were discovered and corrected. The data scrambling algorithm and the post router block have been incorporated in the front end readout chip.

The University of Manchester
Toby Seán Nonnenmacher
Master of Science by Research
Development of Data Processing Algorithms for the Upgraded LHCb Vertex Locator
September 7, 2016
Declaration

No portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.
Copyright statement

1. The author of this dissertation (including any appendices and/or schedules to this dissertation) owns any copyright in it (the “Copyright”) and s/he has given The University of Manchester the right to use such Copyright for any administrative, promotional, educational and/or teaching purposes.

2. Copies of this dissertation, either in full or in extracts, may be made only in accordance with the regulations of the John Rylands University Library of Manchester. Details of these regulations may be obtained from the Librarian. This page must form part of any such copies made.

3. The ownership of any patents, designs, trade marks and any and all other intellectual property rights except for the Copyright (the “Intellectual Property Rights”) and any reproductions of copyright works, for example graphs and tables (“Reproductions”), which may be described in this dissertation, may not be owned by the author and may be owned by third parties. Such Intellectual Property Rights and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property Rights and/or Reproductions.

4. Further information on the conditions under which disclosure, publication and exploitation of this dissertation, the Copyright and any Intellectual Property Rights and/or Reproductions described in it may take place is available from the Head of School of Physics and Astronomy.
Acknowledgements

I owe a great deal of thanks to a number of people, without whom this thesis could not have been written:

I would like to thank Chris Parkes for his tireless supervision, unmatched proof-reading skills, and his care and attention in helping me with this thesis.

I would also like to thank Marco Gersabeck, Stefano De Capua, Alex Bitadze, and Graham Miller for their help with all the work presented in this thesis. Thanks are owed to Nick Mead and Ben Jeffreys for their particular work on the testing of scrambling algorithms.

Special thanks go to Chris Burr and Davide Porzio, for their advice on the writing of this thesis, for sharing an office with me, and for patiently answering my questions all year long.

Moreover, I would also like to thank the members of the HEP group in Manchester for being friendly, welcoming, and ready to answer any questions; and the members of the LHCb collaboration for all the help they have offered.

My year in Manchester has been one of the best experiences of my life. This is entirely down to Michael Byrne and all my other Manchester friends: students, bridge players, and quizzers alike.

My parents and sister have always supported me and my physics ambitions, and, without them, this year would not have been possible.
A special mention must go to Isi Bogod for all her love and support. Her constant encouragement has kept me going all year, especially when things were at their most difficult.

Finally, I would like to express my thanks to Pablo Rodriguez. Pablo was a brilliant supervisor, and helped me with every step of the work presented here. I miss him greatly, and only wish I had known him longer.
1 Introduction

The Large Hadron Collider (LHC) is the world’s largest and most powerful particle accelerator. Large Hadron Collider Beauty (LHCb) detector is one of the four main experiments at the Large Hadron Collider, and is dedicated to precision measurements of properties of particles including $b$ or $c$ quarks. The experiment is designed to make precise measurements of the Standard Model in the search for beyond the Standard Model physics. The Vertex Locator is the subdetector of LHCb closest to the beam line. It is a silicon strip detector that provides high precision measurements of tracks originating from the primary vertex and from the decays of $b$ and $c$ quarks. These tracks are used to precisely determine the position of the primary and secondary vertices, thus determining the flight distance of the hadrons containing the $b$ and $c$ quarks. The Vertex Locator has been used successfully for the entirety of Run I (2010-2013), and is now being used in Run II (2015-2018), of the LHC.

In preparation for LHC Run III, the LHCb detector will undergo a Phase I upgrade. The Vertex Locator, amongst other subdetectors, will be upgraded to facilitate increased instantaneous luminosity through faster readout. This will include upgrades to both the hardware used in the Vertex Locator, and the data processing algorithms it makes use of. The data acquisition board of the upgraded Vertex Locator will be based upon Field Programmable Gate Array (FPGA) technology. The work covered in this thesis is on the development of algorithms used in the data acquisition board for data processing. Work from different areas of the FPGA, including the data processing block, low level interface, and router block are explored. Results of their testing, operation and efficiency are discussed. The order in which these areas of work are presented corresponds to the order in which data travelling
through the FPGA is affected by them.

Chapter 2 provides an introduction to the Large Hadron Collider as a whole, and the LHCb experiment in particular. Chapter 3 explores the upgrade to the experiment including an outline of the upgraded subdetectors and a detailed description of the upgraded Vertex Locator. In Chapter 4 the work done on scrambling algorithms in the Vertex Locator data processing block is described. This includes a discussion of the motivations for using a scrambler, a description of the scrambling algorithms used, and an analysis of their efficiencies. Chapter 5 describes work relating to the decoding of the Gigabit Wireline Transmitter data in the Low Level Interface. Included in this chapter are: an overview of data reception in the Low Level Interface, an analysis of the quality of oscilloscope data taken, and an explanation of the writing and testing of algorithms to fix bugs in the data produced by the Gigabit Wireline Transmitter serialization. In Chapter 6, the work performed on the router of the FPGA is explored. This chapter contains a discussion of the operation of the router and the post router produced in this work. The algorithms written for the post router block governing reformatting of data and changing of transmission frequency are analysed. Finally in Chapter 7, a conclusion of the work covered in this thesis is presented.
The work presented in this thesis concerns a subdetector of the Large Hadron Collider Beauty experiment (LHCb). This chapter is an introduction to the LHCb experiment and its design. In Section 2.1 an overview of the Large Hadron Collider (LHC) and the main experiments at the European Organisation for Nuclear Research (CERN) is provided. The Large Hadron Collider Beauty experiment is discussed in greater detail in Section 2.2.

2.1 The LHC

The Large Hadron Collider (LHC) [1] is a circular collider, and at a circumference of 27km, is the world’s largest and also the most energetic accelerator. It is installed 100m below the ground and located at CERN on the outskirts of Geneva in a tunnel that previously contained the Large Electron Positron Collider (LEP). The LHC is designed for proton-proton collision ($pp$) at centre of mass energies up to a maximum of 14 TeV and at a luminosity of $10^{34}$ cm$^{-2}$s$^{-1}$. The layout of the accelerator complex is shown in Figure 2.1.

Collisions occur at four designated interaction points around the ring, where the four major experiments are situated: A Toroidal LHC ApparatuS (ATLAS) [1], Compact Muon Solenoid (CMS) [1], A Large Ion Collider Experiment (ALICE) [1], and LHC-beauty (LHCb) [1]. ATLAS and CMS are both general purpose detectors designed to look for the Standard Model Higgs boson and physics beyond the Standard Model. LHCb is specifically designed for the study of heavy flavour physics. In addition to $pp$ collisions, the LHC can also collide heavy ions, and ALICE is a dedicated heavy ion experiment. Three smaller
Figure 2.1: The CERN accelerator complex. Figure taken from Reference [2].
Figure 2.2: The LHC long term schedule. Long Shutdown II, during which LHCb will see a major upgrade, is scheduled to begin in December 2018. Data taking is then scheduled to continue in 2021. Figure taken from Reference [3].

specialised experiments share the interaction points of LHCb, CMS, and ATLAS respectively: Monopole and Exotics Detector At the LHC (MoEDAL) [1], TOTal Elastic and diffractive cross section Measurement (TOTEM) [1], and LHC-forward (LHCf) [1].

The LHC has been in operation since 2009; the first period of data used in physics analysis ran from 2010 to 2013, known as Run I. Operations began again with Run II in 2015 following an extended break referred to as Long Shutdown I. The current period of data taking is scheduled to end in December 2018, with another extended break, Long Shutdown II, being used for a major upgrade to LHCb and minor upgrades to ATLAS and CMS. The full LHC upgrade schedule is shown in Figure 2.2.

In order to reach the design energy of the beams, a number of smaller accelerators are required. These progressively increase the energy of the particle beams prior to their injection into the main beam pipe. The protons are created by the ionisation of hydrogen atoms in an electric field, before being accelerated to 50MeV in LINAC2. These protons are then injected into the Proton Synchrotron Booster, Proton Synchrotron, and Super Proton Synchrotron, where they are accelerated to 1.4 GeV, 25 GeV, and 450 GeV respectively. The beams are then accelerated from the injection energy to the required collision energy before being collimated and colliding at the interaction points. While the beams are colliding stably, the experiments take data; subsequently, the beam currents decay. After approximately ten hours,
the beam currents are too low for efficient data taking, so the beams are dumped, and the injection process begins anew.

2.2 LHCb

The LHCb detector is a single arm spectrometer designed for heavy flavour physics. It specialises in the study of hadrons containing $b$ and $c$ quarks for the primary purpose of the search for new physics in measurements of CP violation and rare decays. The detector has a forward angular coverage of 250 mrad in the horizontal plane, and was designed as a forward spectrometer because beauty hadrons are primarily produced in boosted pairs, close to the beam axis, travelling in the same direction. This corresponds to a pseudorapidity range of $1.6 < \eta < 0.49$, where $\eta = -\ln(\tan(\frac{\theta}{2}))$, and $\theta$ is the angle with respect to the beam axis. The full detector is shown in Figure 2.3.

The high centre of mass energy allows for the production of all species of $b$-hadrons, in contrast to the electron-positron collisions at Belle [4] and BaBar [5]. The dominant production of $b\bar{b}$ at the LHC is gluon fusion, with the pair being strongly boosted in the direction of the higher momentum parton.

The design luminosity of LHCb was $2 \times 10^{32} \text{ cm}^{-2}\text{s}^{-1}$, lower than CMS and ATLAS, in order that bunch crossings result in a single proton-proton interaction. In practice, since 2011, LHCb has operated at approximately twice its design luminosity. The luminosity is reduced by focusing the beams less than for ATLAS and CMS. This reduces radiation damage, simplifies data analysis, and decreases the detector occupancy. Additionally, at LHCb, luminosity levelling is achieved by partially separating the beam at the interaction point; as the beam current decays, the luminosity is maintained by increasing the overlap of the beams, resulting in a uniform luminosity over time, up to the point of complete overlap.

To carry out its physics programme, the LHCb detector requires excellent identification of muons, protons, pions, and Kaons, and also detects electrons and photons. Additionally, it requires a trigger that
Figure 2.3: Cross-sectional view of the LHCb detector and its sub-detectors along the $x = 0$ plane. Figure taken from Reference [6].
2. **LHCB**

can discriminate a large range of final states of interest from high multiplicity hadronic backgrounds, and an efficient and versatile data acquisition system.

The detector itself is made up of a large number of sub-components: the LHCb magnetic field is produced by a warm dipole magnet with an integrated field of 4 Tm. The interaction point is surrounded by the silicon Vertex Locator (VELO) [7], which provides high precision measurements of track and vertex coordinates. The remainder of the tracking system is composed of the Tracker Turicensis (TT) [8] and three tracking stations (T1-T3) that compose the Inner (IT) [8] and Outer (OT) [9] trackers. The dipole magnet lies directly upstream of the TT and downstream of T1-T3; the VELO lies beyond the influence of the magnet, while the TT is within the magnet’s fringe field. Two Ring Imaging Cherenkov Detectors (RICH) [10] are used for identification of pions, Kaons, and protons. The Calorimeter system is made up of scintillating tiles along with lead absorbers in the ECAL, and iron absorbers in the HCAL. These are used for the identification and measurement of the energy of electrons and photons in the ECAL [8], and hadrons in the HCAL [8]. To perform muon identification, five muon detectors (M1-M5) are used, with M1 in front of the calorimeter system, and M2-M5 behind it [11]. These subdetectors are described in greater detail in Sections 3.1.1 to 3.1.7

Objects within LHCb are described using a right handed Cartesian Coordinate system. The z axis along the beam pipe, with the positive direction running from the VELO through the detector. The y axis is vertically upwards, and the x axis horizontal, with the origin taken as the interaction point.
3 The LHCb Upgrade

During Long Shutdown II, LHCb will be upgraded in preparation for Run III. The upgrade will allow data collection of 50 fb$^{-1}$ integrated luminosity over 10 years of operation. This will require an instantaneous luminosity of approximately $10^{33}$ cm$^{-2}$s$^{-1}$. The current LHCb hardware is not optimal for acquiring data under these conditions; Subdetectors will have to be re-designed with due consideration given to occupancy rate, radiation hardness, material, cost, and, of course, the required precision. In Section 3.1, the sub-detectors and trigger of the upgraded LHCb experiment will be discussed, and compared to the corresponding current systems. The VELO will be discussed in greater depth in Section 3.2 as it is the primary focus of this work.

3.1 Upgraded components of the LHCb experiment

3.1.1 The Vertex Locator

The current VELO is the closest sub-detector to the beam, made of pairs of silicon modules arranged on either side of the beam line. During beam injection, the modules are separated by more than 5 cm in order to avoid radiation damage. Once the beams are in stable collision, they are brought to within 8 mm of the beam. The modules are used in pinpointing the position of the primary pp collision vertex and measuring the particle trajectories close to the interaction point. To improve its performance and reduce radiation damage, the VELO is cooled to $-20$ °C. The subdetector is contained in a secondary vacuum separated from the rest of the detector by a thin aluminium sheet to
Figure 3.1: Cross-sectional view of the upgraded LHCb detector and its sub-detectors along the $x = 0$ plane. Figure taken from Reference [12].
avoid radio frequency effects of the beams.

In order to maintain or improve its physics performance under the upgrade requirements of greater radiation resistance and the ability to handle a higher data rate, the VELO sensors and electronics will have to be completely replaced. The upgraded VELO [13] will be based on hybrid pixel detectors, using a radiation hard application specific integrated circuit (ASIC) called VeloPix, which will be able to deal with the high data rates. In contrast to the current VELO, the upgrade cooling will be integrated with the module, using a series of microchannels through which carbon dioxide may flow. The new detector will reuse large amounts of the current mechanical infrastructure. The upgraded VELO is discussed in further detail in Section 3.2.
3.1.2 The Dipole Magnet

The LHCb dipole magnet will remain unchanged during the upgrade. The field produced by the magnet is used to bend the trajectories of charged particles in the horizontal plane. The radius of curvature of a particle then allows its momentum to be measured. The magnet is made up of two conical, saddle-shaped coils, which are placed symmetrically to each other in the magnet yoke. The coils deliver a maximum magnetic field of 1.1 T, and an integrated field of 4 T m for 10 m long tracks. The magnet, shown in Figure 3.2 fully covers the acceptance regions of the TT and T1 and is at its maximum in strength in-between them.

The VELO is magnetically shielded, such that particle tracks in that region are not bent. The RICH photodetectors are similarly shielded in order to avoid the deflection of photoelectrons. The polarity of the magnet is changed multiple times per year to allow sources of measurement bias due to asymmetric detector efficiencies to be controlled or eliminated.

3.1.3 The Tracking System

Beyond the VELO, the upgraded tracking system of LHCb will be composed of the Upstream Tracker (UT) between the VELO and magnet, and the Scintillating Fibre (SciFi) tracker downstream of the magnet. The UT will replace the TT, due its greater radiation hardness, detector occupancy, and compatibility with the new 40 MHz readout. The UT comprises four silicon micro-strip layers, improving the coverage of the LHCb acceptance. In the TT four layers were formed of vertically mounted half modules containing seven sensors per module. The first and fourth layers have vertical strips while the second and third layers have strips rotated by $-5^\circ$ and $+5^\circ$ respectively. The UT will maintain this x-u-v-x geometry, but with higher granularity, and removing gaps on the detection area.

Downstream of the magnet, the SciFi will replace the current tracking stations T1-T3, which compose the Inner and Outer Tracker. 99% of the surface of the current tracking stations is covered by the OT de-
Figure 3.3: Layout of the TT, showing its dimensions and the geometry of the layers. Figure taken from Reference [15].
3. THE LHCB UPGRADE

Figure 3.4: Cross-section of the straw tube module with with units in mm. Figure taken from Reference [16].

ductor formed of straw drift tubes, shown in Figure 3.4. The tubes are 4.9 mm in diameter and filled with a mixture of Argon (70%), Carbon Dioxide (28.5%), and Oxygen (1.5%). Only the innermost 1% of the current tracking stations forms the IT, where the high track density due to the Lorentz boost of the b-hadrons requires a high granularity. The IT has the same x-u-v-x geometry as the TT and UT.

The complete downstream tracking system will be replaced by a single Scintillating Fibre Detector covering an area of 360 m². Using a single detector technology in place of the combined IT and OT has the advantage of simplifying track reconstruction. Moreover, the cost of construction, operation, and maintenance of a single technology is lower, and the need for new OT electronics is removed. The detector is composed of 2.5 m long scintillating fibres of diameter 250 µm. Scintillation light is detected by Silicon Photomultipliers (SiPM) contained at the top and bottom of the detector. The x-u-v-x geometry of the IT and OT will be retained in the SciFi tracker layers.

3.1.4 The Ring Imaging Cherenkov Detectors

LHCB currently uses two RICH detectors, RICH1 and RICH2, which provide the excellent identification of charged hadrons needed to distin-
3. THE LHCb UPGRADE

Figure 3.5: Cross-section of the RICH1 detector in the y-z plane. Figure taken from Reference [14].

guish pions from Kaons. When particles pass through radiators with refractive index $> 1$, at velocities greater than the phase velocity of the radiator, Cherenkov photons are released, and their angle of emission is measured by the detectors. This angle, $\theta_c$, is given by

$$\cos(\theta_c) = \frac{c}{nv}$$

(3.1)

where $c$ is the speed of light, $n$ the refractive index of the radiator, and $v$ the velocity of the particle.

In the upgrade, the overall structure of the RICH detectors will remain unchanged. RICH1 lies downstream of the VELO, and covers the full detector acceptance, along with the momentum range from 1 GeV/c to 60 GeV/c. The detector is shown in Figure 3.5. The radiator used is $C_4F_{10}$ which has a refractive index of 1.0014 for a wavelength of 400 nm. The Cherenkov photons are focused on Hybrid Photon Detectors (HPD). In the HPD, the incident Cherenkov photon
emits a photoelectron, which is then accelerated onto a silicon detector by a high voltage field. The HPDs operate for wavelengths in the range 200 to 600 nm and are shielded from the field of dipole magnet by iron blocks.

RICH2 lies downstream of the location of the SciFi tracker and covers a reduced acceptance, and higher momentum range from 15 GeV/c to > 100 GeV/c. For these momenta, a $C_4F_{10}$ radiator with refractive index 1.0005, at 400 nm is used. The optical system of the RICH2 is oriented in the horizontal plane, in contrast to the optical system of RICH1 which is situated in the vertical plane in the current detector. As a part of the upgrade, the photon detectors will be replaced with a new generation, the multianode photomultipliers (MaPMT) which can operate at the 40 MHz rate.

### 3.1.5 The Calorimeter System

The full calorimeter system of the current experiment is made up of four parts: The Scintillator Pad Detector (SPD), the PreShower detector (PS), and the Electromagnetic and Hadronic Calorimeters (ECAL and HCAL). Together, the detectors measure the positions and energies of electrons, hadrons, and photons. The calorimeter system also performs particle identification and is used in the first level of the trigger. The PS, which is used to suppresses the charged pion background, and the SPD, which uses identification of charged particles to suppress the neutral pion background will both be removed in the upgrade. The detection technology of the ECAL and HCAL will not be changed.

The ECAL uses *shashlik* technology: It is formed of a stack of interleaving layers of lead absorber and polystyrene scintillator, with a light collection system of optical fibers running perpendicular to the layers. The ECAL provides information on photons, electrons, and neutral pions. Layers are arranged perpendicular to the beamline with thickness 42 cm, ensuring that electrons and photons are contained within the calorimeter. The energy resolution of the ECAL is dependent on
incident particle energy:

$$\frac{\sigma_E}{E} = \frac{a}{\sqrt{E}} \oplus b \quad (3.2)$$

With $\oplus$ representing quadrature sum, $a = \sqrt{0.09}$ GeV, and $b = 0.008$

The ECAL is found downstream of the PS with a horizontal and vertical acceptance of 300 mrad and 250 mrad respectively. The ECAL is divided into three sections with varying cell sizes in order to produce a more uniformly distributed hit density over its full area.

The HCAL, is located downstream of the ECAL and functions in a similar fashion, using alternate layers of iron absorber and polystyrene scintillator, shown in Figure 3.6. The scintillating tiles in the HCAL, in contrast to the ECAL, run parallel to the beamline, and the granularity of the HCAL is smaller due in response to hadronic showers having greater width than electromagnetic showers. The calorimeter is subdivided into two sections, rather than three as in the SPD, PS, and ECAL. As for the ECAL, the HCAL energy resolution follows Equation 3.2, with different parameters: $a = \sqrt{0.69}$ GeV and $b = 0.09$. 

Figure 3.6: Schematic of the internal cell structure of the HCAL. Figure taken from Reference [14].
In the upgrade, the readout electronics of the calorimeters will be re-designed, to ensure longer lifetimes at high luminosity.

3.1.6 The Muon Detectors

The current muon system is made up of five muon stations: M1, located upstream of the calorimeters, and M2-M5, located downstream and followed by an iron muon filter. These stations are depicted in Figure 3.7. For the upgrade the M1 station will be removed, as the increased luminosities would make it extremely difficult to associate hits at M1 with muon track segments on the other stations.

The stations are positioned perpendicular to the beam line, and have an acceptance of $20 - 306$ mrad and $16 - 258$ mrad in the horizontal and vertical directions respectively. Muons with energy greater than $6 \text{ GeV/c}$ go through all five layers. M1-M3 have a transverse momentum resolution of approximately $20$ percent horizontally, while M4 and M5 are only used to confirm that the muon candidate has entered every layer.

The inner region of M1 contains a gas mixture of Argon, $CO_2$, and $C_4F_{10}$ in the ratio $45 : 15 : 40$, and triple gas electron multipliers (GEM). M2 to M5 and the remaining regions of M1 contain multiwire proportional chambers (MWPC) also containing Argon, $CO_2$, and $C_4F_{10}$, but this time in the ratio $40 : 55 : 5$. The MWPCs contain a system of anode wires between two plates, across which a potential difference can be applied. Charged muons ionize the gas, and the ionization electrons are collected by the wire to measure the muon position. Other than the removal of the M1 station, the main modifications on the muon chambers for the upgrade are: the redesign of readout electronics, to be able to operate at a $40 \text{ MHz}$ rate, and the installation of additional shielding before M2, in order to reduce the contamination due to hadrons.
Figure 3.7: Cross-sectional view of the muon system, including iron filters. The calorimeters separate M1 from M2-5. Figure taken from Reference [14].
3. THE LHCb UPGRADE

3.1.7 The Trigger

The LHCb trigger is designed to reduce the 40 MHz bunch crossing rate to approximately 3 kHz, available to be stored and analysed. Most of the events recorded by the detector will not be in the flavour physics of greatest interest, with branching fractions of b-hadrons to particular decay channels typically being on the order of $10^{-3}$ or less, and not all collisions producing $b\bar{b}$ pairs. The full trigger is made up of a hardware trigger (L0) and two high level C++ software triggers (HLT1 and HLT2). A crucial feature of the LHCb upgrade is the removal of the L0 trigger, leaving a purely software trigger.

During Run I and Run II, the L0 trigger has used information from the ECAL, HCAL, and muon system to reduce the rate to 1.1 MHz, which corresponds to the maximum rate at which the full detector can...
3. THE LHCb UPGRADE

be read out by the electronics. In particular, events with daughter tracks with high transverse momentum and displaced secondary vertices are selected. The current HLT further reduces the rate prior to the final final samples being written to disk. The HLT runs a C++ application on the Event Filter Farm, an array of multi-processor PCs with 29,000 logical CPU cores. The full flow of the trigger including the L0 trigger and HLT is given in Figure 3.8.

The proposed upgrade trigger will consist of an all-software structure providing flexibility to the selection strategies, in particular allowing more efficient triggering on low momentum tracks. The optional Low Level Trigger (LLT) is a software version of L0, applying cuts based on the transverse momentum of photons, electrons, hadrons, and muons. The LLT will be followed by the upgraded HLT, where tracks are reconstructed with precision similar to the offline reconstruction, and selection criteria are applied to reduce the data rate. The performance of the VeloPixel, UT, and SciFi tracker will allow the complete tracking sequence in the HLT to be performed in less than 10 ms. The data rate will be reduced to a rate of approximately 20 – 100 kHz that can be processed offline. The LLT is currently not considered necessary, as the upgraded HLT will be able to read the full data rate, and would reduce the efficiency for hadronic channels, but it will act as a safety net. LHCb will be the first hadron collider experiment to deploy a trigger exclusively in software.

3.2 The LHCb VELO upgrade

The changes to the LHCb experiment in the upgrade with the greatest impact upon the VELO are:

- Increased instantaneous luminosity of $2 \times 10^{33} \text{cm}^{-2}\text{s}^{-1}$.
- Reading out of the Front End (FE) at the bunch crossing frequency (40 MHz).
- The full implementation of event selection in the software.

The goals of the VELO detector upgrade [18] are to cope with the more demanding conditions of the upgraded LHCb experiment, while
maintaining or improving the performance of the current VELO. Simulations show that this can be achieved by reducing the distance to the interaction point. The combination of this, together with the luminosity increase, will cause the peak radiation dose received by the VELO to be increased by a factor 10, and the data bandwidth by a factor 40.

These running conditions cannot be handled with the technologies used in the current VELO detector; therefore, the upgrade modules must be completely redesigned. In place of the current, microstrip based modules, the new modules (see Fig. 3.10) will be based on hybrid pixel sensors, which have two main advantages: higher radiation tolerance and fast, robust pattern recognition due to a simpler geometry and lower channel occupancy. The increased data bandwidth requires new DAQ boards and more efficient cooling. Improved cooling is also required to counter the effect of higher radiation damage.

3.2.1 The current VELO

The current VELO is a subdetector made of 21 pairs of silicon modules built on either side of the beam line throughout the range $-0.2 \, m < z < 0.8 \, m$. This gives an angular acceptance of $15 - 300 \, \text{mrad}$ and, at full closure, the pairs overlap to ensure alignment and avoid acceptance gaps. The modules in the range $-0.2 \, m < z < 0.0 \, m$ help pinpoint the position of the primary vertex. The total number of stations was determined by the constraint that a track originating from a vertex with $|z| < 106 \, \text{mm}$ with $1.6 < \eta < 4.9$ must pass through a minimum of three sensors. A schematic of the setup of the VELO is given in Figure 3.9.

An individual module is made up of two semicircular microstrip silicon sensors of diameter 84 mm: The R sensor measuring the radial position of the track has 512 strips in each $45 ^\circ$ sector, arranged in concentric arcs around the beam line. The distance between strips varies from $38 \, \mu\text{m}$ (closest to the beam) to $102 \, \mu\text{m}$ (edge of the sensor). The $\phi$ sensor measuring the track’s azimuthal position is split into an inner and outer region. The 683 shorter strips in the inner region vary linearly in pitch from $38 - 78 \, \mu\text{m}$, while the pitch of the 1683 longer strips
Figure 3.9: (top) Layout of the current VELO modules along the beam line in the x-z plane. (bottom) View of current VELO sensors in the transverse plane in both the closed (left) and open (right) positions. Figure taken from Reference [14].
3. THE LHCB UPGRADE

Figure 3.10: A module for the upgraded VELO. Pixel sensors are bump-bonded to 3 ASICs and cooled to below $-30 \, ^\circ C$ with bi-phase CO$_2$ which is circulated through microchannels etched in the silicon substrate. Figure taken from Reference [19].

...in the outer region varies between 39 – 79 $\mu$m. The VELO is cooled to $-20 \, ^\circ C$, to reduce radiation damage. The subdetector is contained in a secondary vacuum, separated from the rest of the detector by a thin 300 $\mu$m aluminium sheet, to avoid radio frequency effects of the beams.

3.2.2 The VELO upgrade modules

A key element of the upgraded VELO modules is the VeloPix ASIC. It has a $256 \times 256$ pixel matrix with a pixel of size $55 \times 55 \, \mu$m. Three such ASICs are flip-chipped to a common sensor tile. The process of flip-chipping, or controlled collapse chip connection, involves the connection of the ASICs to the sensor tile using solder bumps on the ASIC pads. The chip is then flipped with its top side facing down, and the solder reflows to complete the connection. Each module will include four sensor tiles, two per side, arranged in an L-shape.
The substrate of the modules is a large, silicon plate, which will additionally act as a cooling plate. Microchannels will be etched into the plate and bi-phase CO$_2$ will be circulated at high pressure and at a temperature of $-30$ °C. The silicon plate will be soldered to a Kovar connector, which will act as the cooling and mechanical interface between carbon fibre frame and the substrate.

52 modules in total will be arranged in 26 pairs, with a pitch of 25 mm. The sensors will be placed closer to the beam axis, at 5 mm from the beam, instead of 8 mm. The geometry of the detector will cause the radiation dose to be highly non-uniform across a single sensor. By the end of its life, the tip of the sensor will have accumulated a dose of $8 \times 10^{13}$ 1 MeV $n_{eq}$/cm$^2$, while the opposite corner of the same sensor will have accumulated a dose smaller by a factor of 105.

3.2.3 VeloPix: the VELO upgrade ASIC

The VeloPix ASIC [13] is a development from the Timepix3 ASIC, with modifications to bring it in line with the VELO upgrade requirements. The features it shares with the Timepix3 are the $256 \times 256$ matrix and $55 \times 55$ µm pixel size, its manufacture on 130 nm technology, and a data driven read-out. Additionally, both ASICs are 3-side buttable, to be flip-chipped to a common sensor, with $2 \times 4$ pixel information gathered in a SuperPixel (SP) structure.

The VELO upgrade imposes demanding requirements in terms of radiation damage and data readout speed. The Timepix3 can be considered a general purpose ASIC, able to provide Time-Over-Threshold information, a measure of the charge collected, and Time-Of-Arrival information, the time-stamp of the hit, simultaneously. The VeloPix will provide only binary information of the hit to be able to cope with the 900 MHits/s at the upgrade luminosity.

The VeloPix will handle 10 times the data rate of the Timepix3 ASIC, with $4 \times 5.12$ GB s$^{-1}$ serial links, which can individually be switched off when unused to save power consumption. The SP architecture shares the address and time-stamp, to create a reduction of 30 percent in the amount of data to be transmitted. The information making up the
Figure 3.11: The VeloPix data path. Hit information is grouped into SuperPixel (SP) packets, propagated downwards, and collected in the End-of-Column (EoC). A router distributes the data equally towards the 4 output serializers. Figure taken from Reference [19].

SP propagates through the column, towards the periphery as shown in Fig. 3.11. For each row, an arbiter decides what information will be propagated: that coming from above, or that generated in the SP itself. The SP information arrives in non-chronological order at the periphery and must be time-sorted in the data acquisition boards (see Section 6.1).
3.2.4 The VELO upgrade microchannel substrate

The module substrate will be a 400 µm thick silicon plate made out of two silicon wafers, one of which has microchannels etched to flow bi-phase CO$_2$ at $-30$ °C. In addition to the perfect coefficient of thermal expansion (CTE) match between the ASICs and the substrate, a silicon substrate provides excellent heat conductivity with almost no temperature gradients across the module and a low contribution to the material budget.

A full scale silicon-on-Pyrex prototype has been produced with a layout close to the finished design, for testing of the cooling performance. A picture of this prototype is shown in Fig. 3.12. 12 ASICs were glued onto the silicon side of the prototype which was powered up to 60 W, twice the expected power dissipation in a module. The microchannel prototype was able to absorb the 60 W of heat with a CO$_2$ flow of 0.5 g s$^{-1}$, and a pressure drop of 4 bar between the inlet and the outlet.

3.2.5 The VELO upgrade DAQ board

A common DAQ board named $PCIe^40$ (see Fig. 3.13) was developed, based on an Altera ArriaX FPGA. The FPGA will be mounted on a
3. THE LHCB UPGRADE

Figure 3.13: The PCIe40 Data Acquisition (DAQ) board. This configuration provides 48 serial input and 48 output links at 5.12 GB s$^{-1}$ and a bi-directional link for Timing and Fast Control (TFC) signals. The output bandwidth is $\sim 100$ GB s$^{-1}$. Figure taken from Reference [19].

PCIe $\times$ 16 Generation 3 card. The card provides up to 48 input and 48 output serial links, along with a bi-directional link for Timing and Fast Control (TFC). The PCIe socket will provide up to 100 GB s$^{-1}$ of bandwidth.

3.2.6 The MiniDAQ-1 board

A prototype version of the PCIe40 board, MiniDAQ-1, has been available since 2014. The MiniDAQ-1 board implements an Altera StratixV FPGA with roughly half the resources of the final ArriaX but maintaining software compatibility. It implements 24 DAQ serial links and a bi-directional output link at 5.12 GB s$^{-1}$ and 10 GB s$^{-1}$ respectively. This board was used to develop the data processing code specific to the VELO, and test the use of the VeloPix ASIC as the serializer. The setup to test the serializer is shown in Fig. 3.14.

The VeloPix implements a serializer called Gigabit Wireline Transmitter (GWT), which is slightly different to the GigaBit Transceiver.
(GBT) serializer implemented by the other LHCb detectors. The GWT sends the data in a 128 bit frame, with fixed header and data lengths. The primary advantage of the GWT is that it is significantly less power demanding than the GBT. It also allows for a slightly larger bandwidth of $5.12 \text{ GB s}^{-1}$ compared with $4.8 \text{ GB s}^{-1}$ for the GBT. The downside of the GWT frame is that it features less redundancy, with the only error checking mechanism being the frame’s header and four parity bits.

The sensitivity of the GWT serializer to Single Event Upsets (SEU) was tested at a heavy ion test-beam. No unexpected behaviour was found, with the number of SEUs recorded being randomly distributed in time, and proportional to the ionization power of the beam. The worst case scenario, corresponding to the ion failing to induce a bit flip in the data, but rather affecting the Phase Lock Loop (PLL) of the clock, was tested: after such an event, the GWT serializer always recovers the normal transmission of data in less than $1.5 \mu\text{s}$. 

Figure 3.14: The MiniDAQ-1 board is a reduced version of the final PCIe40 DAQ board. It is used for the development of the acquisition software and testing of the VeloPix. With this setup, the data is generated in a prototype board, sent through GWT serializers, converted to optical, and recorded in MiniDAQ-1. Finally, the data is sent to a PC through a $10 \text{ GB s}^{-1}$ optical link. The figure is of the setup at Manchester University.
4 Data Scrambling

In this Chapter, work performed on the scrambling algorithms in the data processing block of the data acquisition board is presented. Section 4.1 introduces the need for scrambling algorithms. This is followed by Section 4.2 in which an explanation of the algorithms explored in this work is given. Results are presented in Section 4.3 and conclusions in Section 4.4.

4.1 Motivation for scrambling algorithms

An introduction to the new data acquisition board (DAQ) of the upgraded VELO was given in Section 3.2.5. Data is transferred from the front-end chips of the VELO to the DAQ, using 20 optical links per module, in order to perform the data processing away from the high radiation levels of the detector chamber. The transceivers of the DAQ and the prototype MiniDAQ are the components of the hardware that send and receive the data. In order to work properly they must be addressed with both the data and the associated clock. For sending, the data and the clock must be provided; for reception, the transceiver itself provides the clock as recovered from the data stream. The major advantage of this method is that it is no longer necessary to provide a fine-tuned clock, which takes into account all the possible delays. The disadvantage is that the data have to comply with certain criteria. Every time a transition from 0 to 1 or 1 to 0 occurs in the data, the clock resynchronises. Without transitions, there is a finite probability that the clock will become desynchronised. This probability becomes larger for longer chains of data lacking transitions. Consequently, a high frequency of transitions in the data is required in order to keep
the clock synchronised.

The GWT Serialiser in the ASIC forms a 128 bit frame, as shown in Figure 4.1 comprising of a 4 bit header, 4 single bit parity flags, and 4 Super Pixel Packets (SPP), each composed of 30 bits. The 4 parity bits correspond to the individual parities of the SPPs. The GWT data is not balanced between 1s and 0s and long chains of 0s are present, since the hitmap is largely empty. A scrambling algorithm is therefore introduced in the VeloPix in order to improve the balance of 1s and 0s and increase the number of transitions in the data. The scrambling algorithm may change any bits in the 4 SPPs, but leaves the header and parity alone, such that the start of each frame can be identified. The data is then descrambled in the first block of the low level interface (LLI) section of the TELL40 FPGA.

The requirement of the scrambling algorithm is to produce enough transitions between 1s and 0s in the data to render desynchronisation events extremely unlikely. This is very similar to the process of 8b/10b encoding, which improves balance of 1s and 0s by converting each string of 8 bits into 10 bit symbols. 8b/10b encoding is highly effective, reducing the difference in 1s and 0s in a string of greater than 20 bits to less than 3. However, in this application, a major limiting factor
on the data processing is the enormous quantity of data that must be transferred from the front end VELO. A method increasing the total number of bits, such as 8b/10b encoding, is therefore not feasible. Instead, a custom method of scrambling, in which the number of bits transferred remains the same, but balance of 1s and 0s is improved, is required. The specific scrambling algorithms used are discussed in the following section.

4.2 Scrambling algorithms

4.2.1 Description of the Algorithms

In this project, three different scrambling algorithms were analysed:

1. **Additive Scrambler**: This was the first scrambler used in the miniDAQ. It implemented a scrambling algorithm based on a constant, and the previous and current inputs. The additive scrambler had a dependence on time and required two sets of two-input XOR gates. At every iteration of the algorithm, scrambled data was produced by applying a complicated sequence of XOR functions on each bit of unscrambled data with unscrambled bits from the same frame, scrambled bits from previous frames, and a constant sequence of bits.

2. **The Intermediate Scrambler [20]**: Like the additive scrambler, this scrambler’s implementation required the previous and current input, but unlike it, a constant was only required for the first iteration. For subsequent iterations, scrambled data was produced by taking XOR functions of each bit only with other unscrambled bits and previous scrambled bits. The major difference between the intermediate and additive scramblers was that the intermediate scrambler did not have a time dependence.

3. **The VeloPix Scrambler**: This was the new scrambler to be used in the ASIC for the upgraded VELO. It was multiplicative and similar in nature to the intermediate scrambler, with differences stemming from their different motivations: the intermediate
scrambler was purely designed for software emulation, with the requirement of time independence. The VeloPix scrambler had the same requirements but, furthermore, needed to be compatible with additional requirements of the new ASIC. This included the constraint of less combinatorial logic in the VeloPix scrambler.

A corresponding descrambling algorithm was written for each scrambling algorithm, and used to verify that they were working correctly. This was of particular importance as it is crucial that any scrambled data be recoverable. In the case of the additive scrambler, the descrambling algorithm was identical to the scrambling algorithm, while the two were different in the case of the intermediate and VeloPix scramblers. Simulations of the scramblers and descramblers were run over hundreds of thousands of unscrambled frames of data, produced with Monte Carlo simulation [20]. The scramblers and descramblers were run on the data back to back in ModelSim, a multi-language environment for simulation of hardware description languages. The input and output data were compared to ensure they were identical. The scramblers were also run in C++ using the same input data. It was also verified that the output was identical for C++ and ModelSim. It was subsequently possible to run the scramblers over much larger data samples, a task at which C++ is more time efficient. Lastly, a desynchronisation event was simulated in the C++ code. As expected, no data for the Additive Scrambler was recoverable after the desynchronisation, while for the Intermediate Scrambler and VeloPix Scrambler only the first two frames following the event could not be recovered.

Comparison of the performances of the algorithms

The additive scrambler depends not only on the input data, but also on the initial point in time that the scrambler was started. As a consequence the descrambler has to be synchronised with the scrambler, and they both have to start at the same time. This behaviour is demonstrated graphically by simulating the additive scrambling algorithm while inputting a single constant. The evolution of the output of this scrambler with time (x-axis) is shown in Fig. 4.2. This simulation ver-
4. DATA SCRAMBLING

Figure 4.2: Evolution of the output of the additive scrambler when supplied with a single input signal, simulated in ModelSim. The x axis represents time, with a single clock cycle having a period of 6.25 ps. A single SPP of 30 bits of output data is shown, with raised outputs corresponding to 1s and lowered outputs corresponding to 0s. With a single constant input signal, the output continues to change over more than 30 clock cycles.

This indicates that even if the input signal always remains the same, the output changes because the number of clock cycles since the start of simulation continues to increase. Moreover, if the data does desynchronise, then all subsequent data is unrecoverable until a common reset signal is supplied. In this case, the signal supplied was a single SPP composed entirely of 0 bits.

The intermediate scrambler, and the VeloPix scrambler, do not suffer from the same problem. The intermediate scrambler output from a single constant input can be seen in Fig. 4.3. The equivalent output for the VeloPix scrambler looks the same as it is also a multiplicative scrambler. Due to being multiplicative, the intermediate and VeloPix scramblers, after a desynchronisation event, lose only two frames of data before the scrambler and descrambler resynchronise automatically.
4. DATA SCRAMBLING

Figure 4.3: Evolution of the output of the intermediate (multiplicative) scrambler when supplied with a single input signal, simulated in ModelSim. The x axis represents time, with a single clock cycle having a period of 6.25 ps. Raised and lowered output bits correspond to 1s and 0s respectively. With a single constant input signal, the output changes over two clock cycles, and then remains constant.

Logic gate diagrams, produced in programmable logic-device design software Altera Quartus II, can be used to compare the resource usage of the different scrambling algorithms. In Fig. 4.4 an extract from the logic gate diagram for the additive scrambler is shown. For an input of 120 bits, the additive scrambler uses two sets of two-input XOR gates, with each set containing 30 gates. For the same number of inputs, the Intermediate scrambler, given in Fig. 4.5, uses only one set of three-input XOR gates. This set again contains 30 gates. Similarly, the VeloPix scrambler, shown in Fig. 4.6, uses one set of 30 XOR gates, but the number of inputs to each XOR gate varies from gate to gate.

4.2.2 Simulation of the scrambling algorithms

The scrambling algorithms were coded into C++, in order to simulate them and compare their efficiencies. C++ was chosen on account of being faster and more flexible than ModelSim and more straightforward with regard to producing plots of the outputs. The header and parity bits were not included in data analysis as they are not acted on by the
4. DATA SCRAMBLING

Figure 4.4: Section of the logic gate diagram for the additive scrambler showing the XOR logic gates. The number of inputs in each case is shown on the left of the gate. The square boxes represent registers where the signals are stored in between clock cycles. This scrambling method requires an intermediate register labelled $lfsr_q$.

scrambling algorithms. Analysis was performed on:

- The number of transitions per frame (120 bit signal)
- A comparison of the lengths of runs of 1s and 0s
- The frequency of the lengths of runs of 1s and 0s
4. DATA SCRAMBLING

Figure 4.5: Section of the logic gate diagram for the intermediate scrambler. The data input is stored in the feedbackRegister. On each rising edge of the clock, the data is sent to a set of 30 XOR logic gates with 3 inputs apiece, and then sent to the output register (RX_DATA_O).
Figure 4.6: Section of the logic gate diagram for the VeloPix scrambler showing the XOR gates. The data is sent through variable-input XOR logic gates. An example 29-input gate is shown.
4.3 Results

The expected mean number of transitions in a randomly generated series of 120 bits would be 60. The mean number of transitions in the raw data, before scrambling, was 54.0. Figure 4.7 shows that all three scramblers increase the mean number of transitions to 60.0. The additive scrambler has a standard deviation of 7.35. The two multiplicative scramblers have lower standard deviations: 5.46 for the intermediate scrambler, and 5.45 for the VeloPix scrambler.

Figure 4.8 gives the change in the balance of 1s and 0s for each frame of 120 bits (having excluded the header and parity) over time. For unscrambled data, the difference in 1s and 0s would be off the scale of the graph, as the hitmap is largely empty. All three scramblers reduce the difference in 1s and 0s almost immediately, relative to the unscrambled data. In the first 5000 frames, the scramblers all perform similarly. Thereafter, the mean differences for the additive scrambler and intermediate scrambler both fluctuate more than that of the VeloPix scrambler. For comparison, three purely random sets of data were gen-
4. DATA SCRAMBLING

erated, and $1\sigma$ and $2\sigma$ bands were calculated and included on both graphs. The random data was generated using the Python "random" library to select 1s and 0s with equal probability. The predicted mean is 0, and the performance of each scrambler is consistent with the theoretical prediction of the $1\sigma$ and $2\sigma$ bands shown. The deviation from the predicted mean is fully consistent for both the random data, on the right, and the scramblers, on the left.

Two histograms of the frequency of chains of 0s and chains of 1s are given in Fig. 4.9 for the data prior to and after scrambling with each algorithm. That the raw data contains many more 0s than 1s can be observed from the data before scrambling on each histogram. For each scrambler, the number of 0s was decreased and the number of 1s was increased. The performances of the three different scramblers were almost identical in improving the balance between 1s and 0s. A set of purely randomly generated data was included for comparison: the result of all three scramblers was very similar to random data.

Subsequently, a histogram, given in Fig. 4.10, was plotted for each scrambler, showing the frequency of all chains in total, independent of the parity of the individual bits. All three scramblers perform approximately equally and reduce the frequency of long chains significantly. For example, the number of chains of length 7 is approximately halved. This increase in transitions in the data significantly reduces the probability of desynchronisation of the clock. The result of all three scramblers was again very similar to random data.
Figure 4.8: Graphs displaying the difference in number of 1s and 0s in every frame. The results for the three scramblers are given, top, while the results for three randomly generated sets of data are given, bottom. Additionally, calculated $1\sigma$ and $2\sigma$ bands are shown on each graph.
Figure 4.9: Histogram showing the frequency of lengths of runs of 1s and 0s respectively. Results are shown for each scrambler, the data prior to scrambling, and a set of randomly generated data.
Figure 4.10: Histogram showing the frequency of lengths of runs of 1s and 0s indiscriminately on arithmetic and logarithmic scales respectively.
4.4 Conclusions and Outlook

Chapter 4 has been concerned with the need for scrambling algorithms in the DAQ of the upgraded VELO. Transceivers in the DAQ or MiniDAQ must be able to recover the clock from the data stream. To avoid desynchronisation, frequent transitions between 1s and 0s in the data are required. A scrambling algorithm is required in order to obtain DC-balance (equal number of 1s and 0s) and minimise the length of runs of identical bits. This augments the number of transitions in the data, facilitating the data and clock to remain synchronized.

Three different scramblers were analysed (together with their corresponding descramblers): the original Additive Scrambler, the multiplicative Intermediate Scrambler, and the new, multiplicative VeloPix Scrambler. The Additive Scrambler was incapable of recovering any data following a desynchronisation event until after a reset signal, while the multiplicative scramblers only lost a small number of frames of data. In order to comply with the requirements of the ASIC, the VeloPix scrambler differed from the Intermediate Scrambler in, amongst other things, a need for less combinatorial logic. The performance of the multiplicative scramblers was in essence the same: they were both effective in introducing a greater balance of 1s and 0s to the data, and increasing the mean number of transitions to that expected in completely randomly generated data.

At time of writing, the VeloPix scrambling algorithm has been confirmed for use in the VeloPix. The fabrication of the chip is currently under way, with initial prototypes expected to be available in the coming months. The corresponding descrambling algorithm has been included in the Low Level Interface.
5 Decoding the GWT

This Chapter contains a description of work performed on GWT serialized data. Section 5.1 gives an introduction to the processes performed on data in the Low Level Interface. This is followed by Section 5.2 in which the analysis of data taken with an oscilloscope and methods to remove bugs in those data are discussed. The results and conclusions of that work are given in Section 5.3 and Section 5.4 respectively.

5.1 Reception of the Data Stream in the Low Level Interface

The data stream from the front end ASIC travels through the GWT serializer and arrives at the MiniDAQ prototype through a serial optical fibre. The transceiver component that decodes these data, produces a frame 32 bits in width. In order to comply with the bandwidth of the VeloPix, the transceiver must process the 32 bit words at 160 MHz. The 160 MHz clock is provided externally as a reference clock for the transceiver, and is generated in an on-board Phase-Locked Loop (PLL).

The output of the transceiver is fed into the Low Level Interface specific to the GWT (LLI_GWT), which searches for the header pattern ("0101") in the four Most Significant Bits (MSB) and produces a 128bit frame.\(^1\) The data provided by the transceiver contains the header in the four Least Significant Bits (LSB), but all the VELO code is implemented as though the header were in the MSB. For this reason the first block in the LLI_GWT performs an inversion of the bit order.

\(^1\)A second pattern called header idle ("0110") is also valid, however all the work in this chapter uses the header "0101"
5. DECODING THE GWT

The next block in the LLI_GWT checks how many consecutive headers have appeared. This block raises the lock signal after 23 consecutive headers and the lock is lost following 5 missing headers in 64 frames. This block also sends a bit_slip command to the transceiver once the lock signal is raised. When the transceiver detects a rising edge in this signal it delays the bit stream by one single clock cycle. The effect of this is that the header moves one bit towards LSB positions. The bit_slip is used until the header reaches the four LSB positions.

When the data leave the LLI_GWT they are correctly formatted, following the GWT data scheme shown in Fig. 4.1, in Section 4.1. According to this scheme the data frame has 128 bits, with a constant header in the four MSB. The next four bits correspond to the parities of the four Super Pixel Packets (SPP) contained in the frame. The remaining 120 bits corresponds to four SPP, each being 30 bit long. Every SPP is sub-divided into several parts with different meanings. The 9 MSB correspond to the Bunch Cross IDentification number (BCID), which is a counter that increases every 25 ns. The BCID can also be interpreted as the time-stamp of the SPP. The following 13 bits identify the address of the SPP inside a chip. Finally, the remaining 8 bits are the hitmap of the SPP. This means that each hit in the 8 pixels belonging to this particular SPP will be represented with a "1" in the hitmap.

The main part of the data processing receives the data from the Low Level Interface (LLI) and performs the operations depicted in Fig. 5.1.

5.2 Testing the Data and Bug Fixing

The scrambled GWT data is decoded in the LLI before the VeloPix descrambler is applied, as discussed in Chapter 4.

The formatted data may be read out from the LLI using an oscilloscope or an electro-optical converter. To better understand the decoding of the GWT pattern in the MiniDAQ, a few million (128 bit) frames of data were sampled at the Dutch National Institute for Subatomic Physics (Nikhef) using a high end oscilloscope [21]. Using this
Figure 5.1: Schematic view of the firmware framework for the VELO Upgrade. The GWT frame is built in the LLI_GWT component, whose main functionality is to find the GWT header in the incoming bit stream and lock to it. The next block descrambles the Super Pixel Packets and performs the error checking. The Router aligns all the packets according to their BCID. The TFC processing block implements the BCID-veto signals. The next block tags with one bit those clusters without neighbours in order to save CPU time in the High Level Trigger. Finally the Event ID builds the output frame according to some rules, in order to optimize the output bandwidth. Figure taken from [19].

In this work, the quality of the oscilloscope data was investigated and algorithms to solve the problems in the GWT data were written and tested in C++ and VHDL.
Figure 5.2: Oscilloscope data. The x axis denotes the number of samples, ranging from 0 to 2000 for each row. The y axis denotes the voltage of the scope, ranging from 0.4 to −0.4 volt. The first four graphs correspond to the first four 128 bit frames sampled. The fifth row shows 500 frames overlaid, leaving only the unchanging four bit header and parity clearly visible towards the right of the figure. Figure taken from [21].

5.2.1 Quality of the Oscilloscope Data

The oscilloscope data was taken at a sampling rate of 80 Gsamples s⁻¹, with a single 128 bit frame made up of 2000 samples and corresponding to 25 ns. Four frames of analogue data followed by 500 overlaid frames are shown in Figure 5.2. The voltage varies from 0.4 V for a 1 to −0.4 V for a 0. Each bit is composed of an average of 15.625 samples. Due to the limitations of the oscilloscope, the transitions between 0s and 1s are not recorded as being instantaneous. The fifth graph shows the samples taken of 500 individual frames plotted on top of each other. The only constant bits of the frame are the eight making up the header and parity, which can be seen clearly on the right of the frame. The
header is not in the LSB position at the very edge of the frame. Rather it has been transposed by 16 bits, demonstrating the first of the two bugs to be fixed.

The test of the quality of the analogue data, was to investigate the possibility of converting it to a binary string without losing data. Two algorithms for conversion of the oscilloscope data were constructed and tested in C++:

1. **Set Average Algorithm**

   Every bit of the frame corresponds to 15.625 samples. In this algorithm, the value of the $n$th bit is found using the voltage of the mid-point of each bit rounded to the nearest sample in accordance with Equation 5.1.

   \[
   \text{Sample no.} = \text{round} \left( \frac{1000}{128} \times (n - 0.5) \right)
   \]  

   (5.1)

   This ensures that every frame contains 128 bits, while uncertainty in the analogue data may cause some of those bits to be assigned incorrectly.

   2. **Transition Distance Algorithm**

   This algorithm uses the distance between transitions from negative to positive voltage or positive to negative voltage in the oscilloscope data to work out how many bits there are before a transition from 0 to 1 or 1 to 0 respectively. The number of bits before transition is given by the number of samples from one zero in the analogue data to the next, divided by the average number of samples per bit as shown in Equation 5.2.

   \[
   \text{No. bits prior to transition} = \text{round} \left( \frac{\text{Number of samples between zeros}}{15.625} \right)
   \]  

   (5.2)

   Uncertainties in the analogue data may lead to chains of 1s or 0s that are too long or short by one bit.

   The extent of the inaccuracies in the digital data provided by these algorithms must be established. If a small number of frames containing
errors are produced and can be identified, this may be considered an acceptable loss. If, on the other hand, a large number of frames are inaccurate, the oscilloscope data is unusable for translation to digital.

5.2.2 Removal of the Bugs in the Data

Two algorithms to counteract the bugs in the data stream were constructed and tested in C++ before being implemented in VHDL. The first, and simpler, of the two bugs sent out every 16 bits of data in the wrong order, starting with the fourth to fifteenth most significant bits and followed by the first to third. This bug was found because the header bits were not found in the LSB position in the data. The first algorithm solved this by transposing the bits back into their original order. The second bug caused the four central bits of the frame to be held one clock cycle ahead of the rest of the frame. This bug was found by using data in which the third and fourth SPPs in a frame were identical. An illustration of this bug is given in Figure 5.3. The second algorithm maintained delayed data output by a clock cycle while maintaining a copy of the four central bits from the previous frame and inserting them into the same position in the subsequent frame. The algorithms were initially tested on simulated data with the intention of subsequently testing them on the digitised oscilloscope data. Both algorithms were tested on their own, and in conjunction with the VeloPix scrambler and descrambler.
5. DECODING THE GWT

Figure 5.4: Graph showing the frequency of the number of samples prior to a transition through 0 in the oscilloscope data. The number of samples before a transition is given on the x axis. Frequency is given on the y axis on a logarithmic scale. Peaks corresponding to integer multiples of 15.625 are clearly visible.

5.3 Results

The accuracy of the digital data, produced by the action of the Set Average and Transition Distance algorithms on the oscilloscope data, was very poor. For the Set Average algorithm, each frame of data produced contained 128 bits as required, but more than 50% of the frame headers were incorrect, implying that well over half of the frames contained inaccuracies. For the Transition Distance algorithm, all the headers were correct and readily identifiable, but more than 60% of the frames contained more or fewer than 128 bits as extraneous bits were added or taken away. In both cases, the accuracy of the data produced was so poor as to render it unusable, and the bug fixing algorithms were only applied to simulated data. To further investigate the quality of the oscilloscope data, a histogram of the number of samples in between two transitions through 0 was plotted, and is given in Figure 5.4.
The average number of samples between transitions through 0 in the oscilloscope data (corresponding to transitions from 1 to 0 or 0 to 1 in the digital equivalent) is 15.625. The peaks visible in the data correspond to chains of 1s and 0s of integer length, and are found at approximate multiples of this average. The peaks become shorter with increasing number of samples, because longer chains are less frequent, due to the actions of the VeloPix scrambler. The poor quality of the oscilloscope data can be seen in the lack of gaps between peaks, i.e. the length of chains is indistinct due to the uncertainty in the data. Some chains of length 3, for example, are composed of more samples than some chains of length 4, or fewer samples than some chains of length 2. It is therefore clear that the inaccuracy of the digital data produced is down to the poor quality of the oscilloscope data, and not the algorithms used to convert it, as chains of bits of different lengths are indistinguishable. This testing verifies the need to read data through an electrical/optical device instead of an oscilloscope.

Both algorithms to fix bugs in the GWT readout performed well. The algorithms, which were run on simulated data in conjunction with the VeloPix scrambling and descrambling algorithms, were successful in removing the bugs in the C++ and ModelSim. The algorithm dealing with the reordering bug delayed the data by a single clock cycle, while fixing the four unsynchronised central bits required two clock cycles: the greatest efficiency possible.

### 5.4 Conclusions and Outlook

Chapter 5 has been concerned with the data produced by the decoding of the GWT pattern in the MiniDAQ. The quality of data taken with an oscilloscope was tested, and two bugs in the data output were resolved. The oscilloscope data was instrumental in discovering these bugs, and two algorithms were constructed to convert the data to digital. The digital data produced was not of usable quality. This was found to be due to the poor quality of the analogue data, in which some chains of bits of different lengths were indistinguishable from each other. In
future, data will have to be read out using an electro-optical converter.

Two algorithms counteracting the bugs in the data stream were constructed and tested. Both were successful and efficient in removing the bugs, and worked in conjunction with the VeloPix scrambler. The algorithms are ready to be implemented in the Low Level Interface of the MiniDAQ.
6 The Router and Post Router

This Chapter contains a description of work performed on the Router section of the PCIe40 board. Section 6.1 gives an explanation of the Router block and its function. This is followed by Section 6.2 in which the work done on algorithms for the post router block is discussed. Section 6.3 and Section 6.4 give the results and conclusions of that work, respectively.

6.1 The Router

The router is the section of the PCIe40 card that is responsible for the time reordering of the data packets. The VeloPix ASIC readout sends out the data with a variable latency, causing the data packets to arrive at the FPGA non-chronologically. Therefore, before cluster finding and event building can take place, a time reordering process is required by the FPGA. Indeed, the main function of the PCIe40 is the time ordering of data and assembly of data packets with all hits from a given Bunch Crossing Identity (BCID) which can then be sent on to the event builder farm. The PCIe40 must cope with the high rate of data coming from the 12 VeloPix ASICs in each module, the peak data rate of the hottest module being 61.2 Gbit/s. Fast data processing and large amounts of available memory are required inside the processor, in order to avoid any data loss. Additionally, there must be enough buffer space in reserve to deal with any large bursts of data.

The design of the algorithm implemented on Altera Arria X devices uses a two-stage process. The router comprises two smaller routers, of which a simplified scheme is given in Fig. 6.1. The first stage is a 5-bit router with 20 inputs (corresponding to the signals coming from
Figure 6.1: A simplified schematic of the router. The 5 MSB router is shown (left) along with the complementary 4 LSB router (right). Figure taken from [19].
6. THE ROUTER AND POST ROUTER

Figure 6.2: Diagram of a 1-input by 2-output 1-bit comparator (left) and 2-input by 2-output 1-bit comparator (right). Figure taken from [19].

The purpose of this router is to route the data packets using the 5 most significant bits (MSB) of the 9-bit timestamp (BCID) of the super pixel packets (SPP). The second stage is a 4-bit router with 32 inputs. It is formed of 32 identical blocks, each consisting of 1-input by 2-output, 1-bit comparator elements and RAM writing blocks. This router is responsible for ordering of the data according to the remaining 4 least significant bits (LSB).

The router is primarily formed by 2-by-2 and 1-by-2 comparators. The latter are the less complex and a simplified representation is shown in Fig. 6.2 (left). A data packet goes into the comparator, subject to whether the bit of the BCID being analysed is a 0 or a 1. Subsequently, the data packet is sent out through one of the two available outputs. The BCID of each data packet is checked one bit at a time, with five bits read by the 5 MSB router and the remaining four bits read by the 4 LSB router. 2-by-2 comparators are more complex, but the fundamental function is the same. Each of the two inputs effectively goes to a 1-by-2 comparator, which determine if the the bit of the BCID being analysed is equal to 0 or 1. Subsequently the data packet is sent to one of the first in, first out (FIFO) queues. The FIFO queue to which the data packet is sent depends on whether the bit is a 0 or a 1 (see Fig. 6.2 (right)). The reason these FIFO queues are necessary is that the two bits from the BCID being analysed in each input may be the
same. Both data packets must be sent out of the comparator through the same output and cannot leave simultaneously. The FIFO acts as a buffer, allowing the data packets to be sent out without data loss due to output saturation. To avoid any data loss, the buffers must have an abundance of storage to avoid buffer overflow. After the data packets go into their corresponding FIFO queues, two blocks verify which of the queues is closer to saturation to determine the order in which they send their data out. When the whole BCID has been checked by the two routers, all data corresponding to a given BCID will have gathered in the same queue.

A schematic of the basic logic of the 5 MSB router is given in Fig. 6.3. Each of the columns of the router deciphers one of the BCID bits of the data packets passing through that column, and orders those packets, sending those with the largest number of 0s and 1s to the top and bottom respectively. This translates into 32 outputs at the end of
6. THE ROUTER AND POST ROUTER

the 5 MSB router with the topmost output having purely 0s in its 5 MSB and the bottommost output having purely 1s in its 5 MSB. Therefore, from each output of the 5 MSB router come packets with 16 different possible BCIDs \(2^4 = 16\). Those 4 remaining bits of the 9 bit BCID must be determined and the data ordered according to their values. The following step orders the data coming from each of those 32 outputs by connecting them to a block made up of a 1-input by 2-output, 1-bit comparator and RAM writing blocks. Depending on the value of the first LSB of the BCID the data packet goes through one of the two available outputs. The path taken by each data packet depends on the 4 LSB of the BCID, leaving 16 possible paths and, therefore, 16 outputs. Even though there are 16 outputs, there is only ever one input link, so there are never two outputs sending out data simultaneously.

At the end of its path, the data packet reaches a FIFO queue. The 16 queues per block, store all the data packets that come from a single bunch crossing, i.e. have the same BCID. A device verifies which of the 16 FIFO buffers contains the most data, which is deposited in RAM where each of the memory slots is filled with the data from one buffer. Finally, after the data packets reach the RAM, they are sent to the next block of the data processor, the multi-event packet builder (MEP). The reason that the simpler design employed for the 4 LSB router is not used for the whole 9 bits is that the size of the required buffers and logic delays caused by handling 20 inputs and 9 bits are too large for the FPGA used.

6.2 Data Formatting in the Post Router

After their time in the router, the data packets are sent out from the RAM to the MEP building block of the data processor. The data format that must be read into the MEP is different to that of the data leaving the router, and it is non-trivial to change this. In this work, algorithms to provide the correct data formatting were constructed and tested. A post router block was created between the router and MEP.
6. THE ROUTER AND POST ROUTER

to be home to this reformatting.

The data is sent out from the data processing blocks and enters the router in frames of 120 bits at a frequency of 40 MHz. These frames go through comparators in the router, and are stored in RAMs as up to $256 \times 24$ bits each, equivalent to a maximum of 32 BCIDs. This data is transmitted to the post router 8 24 bits words at a time, at a frequency of 160 MHz. The MEP requires the data to be sent out of the post router in frames of 256 bits at a frequency of 39 MHz. One of the two main parts of this work was, therefore, to convert the data transmission frequency using a dual clock FIFO queue. Secondly, in order to perform functions on the data, the MEP block requires data corresponding to a new BCID to have a 32 bit global header. This global header is composed of 12 bits denoting the BCID event and a 20 bit Global Data Length (GDL). An additional 64 free bits must also be introduced for each event, to contain Event Identity information in the Timing and Fast Control and MEP blocks. The format of the data required by the MEP is shown in Figure 6.4. Algorithms to convert the data to its new format and change the data transmission frequency were written in VHDL and tested in ModelSim and Altera Quartus II.

6.2.1 Reformatting the Data Frames

Frames of 256 bits, containing up to 8 32 bit SPPs, enter the post router block from the RAM. The algorithm written in this work recognises when data corresponding to a new bunch crossing is received and counts how many SPPs correspond to that BCID, before data from the next bunch crossing will enter. The final frame corresponding to a BCID is full, only if the total number of SPPs corresponding to that BCID is a multiple of 8. Otherwise, the remainder of the final frame is empty. The data in the each frame is moved by 96 bits in the LSB direction, and headed by the 64 free bits and global header. If the BCID data package contains 6 or more SPPs in its final frame, then an extra frame must be added to hold the free bits or global header. This condition is given in Equation 6.1.
Figure 6.4: Schematic of the data formatting to be introduced between the router and the MEP. The differently numbered global data correspond to different bunch crossings. Figure taken from [22].
\[(\text{Number of Samples}) \mod 8 \geq 6 \quad (6.1)\]

After the introduction of the 64 free bits and global header, if the final frame of the BCID was not full, padded data was added to make it up to 256 bits. As each frame of data requires one clock cycle to be read in or out, the cases requiring the addition of a frame take a clock cycle longer to read out than read in. Therefore, a crucial part of the algorithm checks in advance to see if an extra clock cycle will be required, and sends a Data Valid (DV) signal back to the router to hold the data there for one cycle until the post router can accept it. If this were not implemented, the post router would become overloaded and data would be lost.

### 6.2.2 Changing the Transmission Frequency

The clock that governs data entering the post router runs at a frequency of 160 MHz, while data leaving the post router must enter the MEP at 39 MHz. This was achieved using a variable clock FIFO queue. Data enters the queue at the initial 160 MHz, and is stored there until it is ready to leave at 39 MHz. As the transmission clock of the FIFO queue is slower than the reception clock, another DV signal had to be implemented to stop the buffer from becoming saturated. As the buffer nears its fullest state, a signal is sent back to the router that holds the data until such a point as the FIFO queue becomes empty enough to resume receiving data.

### 6.3 Results

The dual clock FIFO buffers and reformatting algorithms were simulated and tested in ModelSim and subsequently Altera Quartus II. They were run on simulated data ensuring that bunch crossings containing various numbers of SPPs were tested. They were found to work successfully, producing data at a frequency of 39 MHz and including the 64 free bits and global header required. The resulting data was
Figure 6.5: Diagram of the path travelled by the data from the GWT Serialiser to
the MEP. The number of bits per frame of data and transmission clock
frequency are shown at each stage. The three elements passed between
the router and post router are shown: the data and the number of
SPPs for a given BCID are passed from the router, and the DV signal
is sent back to the router.
suitable for the functions of the MEP. A schematic of the path of data from the GWT Serialiser to the MEP, through the post router block produced in this work is given in Figure 6.5. Due to the use of the two DV signals, no data is lost in the post router block, and the number of additional clock cycles required to ensure this was negligible.

6.4 Conclusions and Outlook

Chapter 6 has been concerned with the transmission of data between the router and MEP. A post router block was constructed and algorithms within it were written to reformat the data to an MEP usable form and change the transmission frequency without loss of data. This block was successful in producing data of the format required by the MEP at the required frequency. The algorithms of the block were found to be efficient, requiring only a few additional clock cycles to perform the reformatting.

The post router block has been included in the router section of the VeloPix code following further work by Pablo Rodriguez and Graham Miller. The algorithms constructed in this work will be used in between the router and MEP blocks in the PCIe40 card.
7 Conclusion

The upgrade of the Vertex Locator is a major part of the Phase I upgrade to LHCb in preparation for Run III. The increased demands on data readout speed of the VELO require a complete overhaul of its data processing algorithms. This thesis describes work on constructing and testing these algorithms for the upgraded VELO FPGA board. In particular, this work has focused on algorithms in three main areas of the FPGA: the data processing block, the low level interface, and the router block.

The new VeloPix scrambling algorithm for the DAQ was simulated and tested against previous scrambling algorithms. This scrambler is needed within the DAQ to allow transceivers to recover the clock from the data stream, and avoid desynchronisation events. In total, three scramblers were analysed: the new VeloPix scrambler, the multiplicative Intermediate Scrambler, and the older Additive Scrambler. The VeloPix scrambler was used due to its compliance with the specific requirements of the ASIC and was implemented in the data processing block. It was found to work very well, recovering most data after a desynchronisation event and being effective in introducing a greater balance of 1s and 0s to the data, to reduce the number of desynchronisation events.

Data, produced in the LLI following decoding of the GWT pattern in the MiniDAQ, were examined. The quality of data taken with an oscilloscope was tested, and solutions were found to two bugs in the data output. Digitisation of the analogue scope data was attempted, but the digital data produced was not of usable quality. This was found to be due to the poor quality of the analogue data, which was analysed in detail. Two algorithms counteracting the bugs in the data stream were created and simulated. These successfully removed the bugs and
worked efficiently in combination with the VeloPix scrambler.

The path of data between the router and MEP blocks in the FPGA was explored. A post router block was constructed and reformatting algorithms were written to produce data that could be read in by the MEP. Additionally, algorithms were written to allow the post router different reception and transmission frequencies. This block was wholly successful in producing data of the format required by the MEP at the required frequency. The post router block was been included in the router section of the VeloPix code.

Much of the work covered in this thesis will be used in the upgraded Vertex Locator. The work on fixing bugs in the GWT serialized data in the LLI will be used in the prototype board, and is not relevant for the VeloPix chip. The VeloPix scrambling algorithm and post router section have both been confirmed for inclusion in the chip. The VeloPix chip is currently in fabrication, with initial prototypes expected to be available in the coming months. The upgraded VELO, which includes the chip, will be installed during Long Shutdown II (2018-2021). Subsequently in 2021, it will begin taking data for Run III of the LHC.
References


