

Fakultät für Physik und Astronomie
Ruprecht-Karls-Universität Heidelberg

Diplomarbeit
im Studiengang Physik
vorgelegt von
Ralf Muckerheide
aus Düsseldorf

November 2005

Entwicklung eines Serientests für den TDC-Auslesechip der LHCb Spurkammern

Die Diplomarbeit wurde von Ralf Muckerheide ausgeführt am
Physikalischen Institut
unter der Betreuung von
Herrn Prof. Dr. Ulrich Uwer

Entwicklung eines Serientests für den TDC-Auslesechip der LHCb Spurkammern

Zur Auslese der LHCb Spurkammern wurde in Heidelberg ein strahlenharter TDC-Chip entwickelt. Bevor die TDC-Chips im Experiment eingesetzt werden können, müssen alle produzierten Chips getestet werden.

Im Rahmen dieser Diplomarbeit wurde der notwendige Serientest konzipiert und aufgebaut. Um beim Test hohe Datennahmeraten zu gewährleisten, wurden alle Testalgorithmen unter Benutzung eines Field Programmable Gate Arrays (FPGA) in Hardware realisiert. Statt der TDC Rohdaten werden am Ende jedes Untertests nur noch die Testergebnisse an den Mess-PC weitergegeben.

Die FPGA-Schaltung ermöglicht zum einen die Überprüfung der aus dem Datenstrom extrahierten Status- und Übersichtsinformationen. Zum anderen ist es möglich, Trefferhäufigkeiten und Driftzeiten in einem Histogramm zu akkumulieren und auszuwerten. Die parallele Datenverarbeitung auf dem FPGA beschleunigt die Testprozedur erheblich, so daß der Test eines Chips nach etwa 25 s abgeschlossen ist.

Die in dieser Arbeit entwickelten und erfolgreich getesteten Algorithmen werden für die Serientests des OTIS Chips eingesetzt.

Development of a serial test of the TDC readout chip of the Tracking System for the LHCb Experiment

For the readout of the LHCb Tracking system a radiation hard TDC chip (OTIS) has been developed in Heidelberg. Before it can be used in the Experiment each TDC chip has to be tested for its functionality.

Within this thesis a procedure has been developed which is capable to perform these tests. To assure high data taking rates during the test phase all test algorithms have been implemented in hardware using a Field Programmable Gate Array (FPGA). Instead of the TDC raw data only the test results are transmitted to the readout PC at the end of each individual subtest.

The FPGA circuit allows to check the status and summary information that has been extracted from the raw data stream. In addition it accumulates hit rates and drifttimes in histograms. The parallel processing on the FPGA chip accelerates the test procedure such that the test of a single TDC chip is finished after 25 s.

The successfully developed and implemented test algorithms are in use in the current mass production test of the OTIS chip.

Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 1 | Einleitung | 1 |
| 1.1 | Der Large Hadron Collider (LHC) | 1 |
| 1.1.1 | Das LHCb-Experiment | 2 |
| 1.2 | Die Outer Tracker Ausleseelektronik | 4 |
| 1.2.1 | Steuerung der Elektronik | 5 |
| 1.3 | Der OTIS Time to Digital Converter | 7 |
| 2 | Anforderungen an den OTIS-TDC Serientest | 11 |
| 2.1 | Motivation | 11 |
| 2.2 | Die Funktionstest am OTIS | 12 |
| 2.3 | Schematischer Überblick über den Testaufbau | 16 |
| 3 | Entwurf der FPGA-Schaltung | 19 |
| 3.1 | FPGA Grundlagen | 19 |
| 3.1.1 | EDA-Entwurfswerkzeuge und Entwurfsablauf | 21 |
| 3.2 | Überblick über die Testprozedur | 23 |
| 3.2.1 | Der PCI-Block und Datensynchronisation | 27 |
| 3.2.2 | Der PCI-Controller | 28 |
| 3.2.3 | Der OTIS-Controller | 31 |
| 3.2.4 | Die Datenintegritätsprüfung | 33 |
| 3.2.5 | Der Histogrammierblock | 37 |
| 3.2.6 | Die Daten-Prozessierung | 43 |
| 4 | Messungen zum FPGA-Schaltungsentwurf | 49 |
| 4.1 | Die Fast Control auf dem FPGA | 49 |
| 4.2 | Die Datenintegritätsprüfung | 51 |
| 4.2.1 | Die OTIS-Datenkopf-Analyse | 51 |
| 4.2.2 | Der EventID-Tester | 52 |
| 4.2.3 | Der Latency-Tester | 53 |
| 4.3 | Der Histogrammer-Block | 55 |
| 4.3.1 | Der Driftzeithistogrammer | 55 |
| 4.3.2 | Der Channel-Map-Histogrammer | 56 |
| 4.4 | Die Datenprozessierung | 58 |
| 4.4.1 | Das Banyan-Netzwerk | 58 |
| 4.4.2 | Die DNL-Berechnung | 60 |

| | | |
|----------|---|-----------|
| 5 | Betriebspunktstudien zum OTIS | 61 |
| 5.1 | 2D-Spannungsscan | 61 |
| 5.1.1 | Testaufbau | 61 |
| 5.1.2 | Durchführung der Messung | 63 |
| 5.2 | Frequenzscan des OTIS | 71 |
| 5.3 | Messungen am OTIS 1.3 | 73 |
| 6 | OTIS-Wafer-Test | 75 |
| 6.1 | Der Testaufbau im Reinraum | 75 |
| 6.2 | Überblick über den Testablauf | 76 |
| 6.2.1 | Der General Connection Test | 77 |
| 6.2.2 | Der Channel-Map Test | 77 |
| 6.2.3 | Der DNL Test | 79 |
| 6.2.4 | Der Buffer Overflow Test | 80 |
| 7 | Zusammenfassung | 83 |
| A | Anhang | 85 |

Abbildungsverzeichnis

| | | |
|------|--|----|
| 1.1 | Experimenteller Bereich des LHCb-Experiments | 1 |
| 1.2 | Der LHCb-Detektor | 2 |
| 1.3 | Das äußere Spurkammesystem | 3 |
| 1.4 | Schema der Ausleseelektronik | 4 |
| 1.5 | Schematischer Überblick über das lokale TTC-System | 6 |
| 1.6 | Schematischer Überblick über die lokale Slow Control | 6 |
| 1.7 | Der OTIS-TDC | 7 |
| 1.8 | Blockdiagramm der DLL | 7 |
| 1.9 | Blockschaltbild des OTIS TDC | 8 |
| 1.10 | Ein Datensatz des OTIS | 9 |
| 1.11 | Kodierung der Driftzeiten | 9 |
| 2.1 | Pipeline-Register | 14 |
| 2.2 | Überlauf des Derandomizing Buffers | 15 |
| 2.3 | Überblick Serientestaufbau | 16 |
| 3.1 | Schematischer Aufbau des FPGA | 19 |
| 3.2 | Die Logik-Zelle | 20 |
| 3.3 | Die Stratix-Karte | 21 |
| 3.4 | Der Entwurfsablauf | 22 |
| 3.5 | Datenfluss durch das FPGA Design | 23 |
| 3.6 | Steuereinheit des FPGA | 25 |
| 3.7 | Schema vom Toplevel des Schaltungsentwurfs | 26 |
| 3.8 | Der PCI-Controller | 28 |
| 3.9 | Schema eines Zustandsautomaten des Befehlsdekoders | 29 |
| 3.10 | Der OTIS-Controller | 31 |
| 3.11 | Zustandsautomat zur Triggergenerierung | 32 |
| 3.12 | Die Daten-Integritätsprüfung | 33 |
| 3.13 | Schema der Statusbit-Analyseeinheit | 34 |
| 3.14 | Die Statusbit-Extraktion | 35 |
| 3.15 | Der Event-ID-Tester | 36 |
| 3.16 | Der Latency-Tester | 37 |
| 3.17 | Der Histogrammer | 38 |
| 3.18 | Bereitstellung der Driftzeitdaten | 38 |
| 3.19 | Erster Teil des Histogrammers | 40 |
| 3.20 | Der FIFO-Controller Zustandsautomat | 40 |

| | | |
|------|---|----|
| 3.21 | Simulation des Dateneinleseteils des Histogrammers | 41 |
| 3.22 | Zweiter Teil des Histogrammers | 42 |
| 3.23 | Simulation der Initialisierung und Datenauslese des Histogrammers | 43 |
| 3.24 | Das Banyan-Netzwerk | 43 |
| 3.25 | Die Basisebene des Banyan-Netzwerks | 44 |
| 3.26 | Die zweite Ebene des Banyan-Netzwerks | 44 |
| 3.27 | Die dritte Ebene des Banyan-Netzwerks | 45 |
| 3.28 | Die vierte Ebene des Banyan-Netzwerks | 46 |
| 3.29 | Berechnung der DNL auf dem FPGA | 47 |
| | | |
| 4.1 | Fast-Control-Signale des FPGA | 50 |
| 4.2 | Fast-Control des FPGA - zweiter Teil | 50 |
| 4.3 | Funktionstest des Header-Analysers | 51 |
| 4.4 | Zweiter Test des Header-Analysers | 52 |
| 4.5 | Funktionstest des EventID-Testers | 53 |
| 4.6 | Funktionstest des Latency-Testers | 53 |
| 4.7 | Die OTIS Monitor-Signale | 54 |
| 4.8 | Histogrammer Test | 55 |
| 4.9 | Channel-Map-Histogrammer Test | 57 |
| 4.10 | Channel-Map-Histogrammer Test am OTIS -even | 57 |
| 4.11 | Channel-Map-Histogrammer Test am OTIS -odd | 58 |
| 4.12 | OTIS 1.2 Histogramm | 58 |
| 4.13 | Banyan Funktionstest - Maxima | 59 |
| 4.14 | Banyan Funktionstest - Minima | 60 |
| 4.15 | DNL-Funktionstest | 60 |
| | | |
| 5.1 | Aufbau der Testumgebung | 61 |
| 5.2 | 2D-Spannungsscan am OTIS 1.2 (erster Teil) | 64 |
| 5.3 | 2D-Spannungsscan am OTIS 1.2 (zweiter Teil) | 65 |
| 5.4 | Verlauf von Vctrl | 66 |
| 5.5 | DNL pro Zeitbin $2.5\text{ V } V_{digital}, 2.5\text{ V } V_{analog}$ | 67 |
| 5.6 | DNL pro Zeitbin $2.9\text{ V } V_{digital}, 2.7\text{ V } V_{analog}$ | 67 |
| 5.7 | Odd-Even Asymmetrie des OTIS 1.2 | 68 |
| 5.8 | Bin 0-Abweichung | 69 |
| 5.9 | Verlauf der DNL | 70 |
| 5.10 | DNL/Abstand $V_{digital} - V_{analog}$ | 70 |
| 5.11 | Aufbau der Testumgebung mit externem Frequenzgenerator | 71 |
| 5.12 | Frequenzscan am OTIS | 72 |
| 5.13 | Aufbau der Testumgebung | 73 |
| 5.14 | Histogramm des OTIS 1.3 | 74 |
| 5.15 | Zebra-Plot OTIS 1.3 | 74 |
| | | |
| 6.1 | Testaufbau Reinraum | 75 |
| 6.2 | Ablauf des Serientests | 76 |
| | | |
| A.1 | Erster Teil des Toplevels | 86 |
| A.2 | Zweiter Teil des Toplevels | 87 |
| A.3 | Belegung der PCI-Register | 88 |

| | | |
|-----|--|----|
| A.4 | Teil 1 der Testadapter - Pinbelegung | 89 |
| A.5 | Teil 2 der Testadapter - Pinbelegung | 90 |
| A.6 | Teil 3 der Testadapter - Pinbelegung | 91 |
| A.7 | Testadapter AS30 | 92 |
| A.8 | Testadapter Bestückungsseite - Top | 93 |
| A.9 | Testadapter - Bottom | 93 |

Tabellenverzeichnis

| | | |
|-----|--|----|
| 2.1 | Funktionstests des OTIS | 12 |
| 3.1 | Das Befehlsregister | 30 |
| 3.2 | Der Datenkopf des OTIS 1.2 | 35 |
| 5.1 | Spezifikation der FPGA Signalpegel | 62 |
| 5.2 | Spezifikation der OTIS Signalpegel | 62 |
| 6.1 | Ablauf des General Connection Tests | 78 |
| 6.2 | FPGA Konfigurationsregister beim General Connection Test | 78 |
| 6.3 | Ablauf des Channel-Map Tests | 79 |
| 6.4 | Ablauf des DNL Tests | 80 |
| 6.5 | Ablauf des Buffer Overflow Tests | 81 |

Kapitel 1

Einleitung

1.1 Der Large Hadron Collider (LHC)

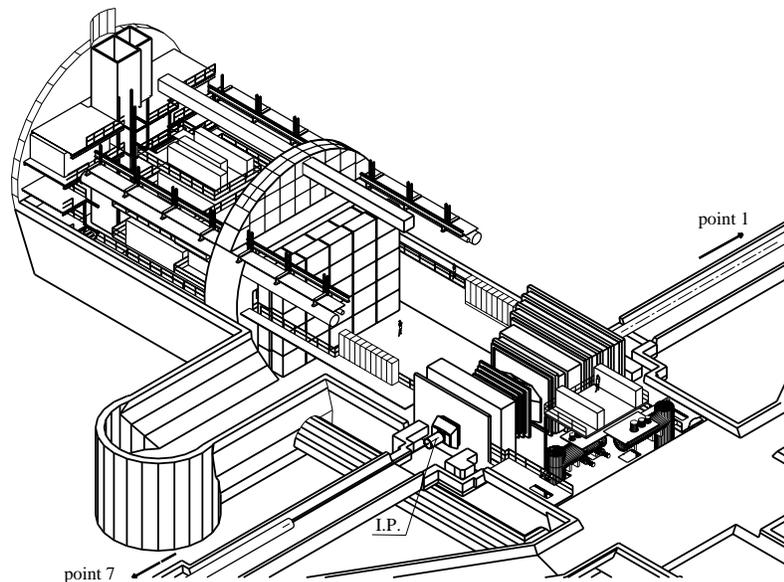


Abbildung 1.1: Experimenteller Bereich des LHCb-Experiments

Die Hochenergiephysik beschäftigt sich mit der Frage nach den kleinsten Bausteinen der Materie und den grundlegenden Kräften zwischen diesen Bausteinen. Im Standardmodell der Elementarteilchenphysik ist das heutige Wissen zu diesen Fragen zusammengefaßt. Zu den Elementarteilchen gehören zwölf Fermionen, die sich in sechs Quarks (up, down, charm, strange, top und beauty(bottom)) und sechs Leptonen (e , μ , τ , ν_e , ν_μ , ν_τ), sowie die zugehörigen Antiteilchen einteilen lassen. Im Standardmodell wird die Wechselwirkung zwischen diesen Teilchen über den Austausch von den sogenannten Austauschbosonen beschrieben. Hierbei handelt es sich um das Photon für die elektromagnetische Wechselwirkung, die W- und Z-Bosonen für die schwache Wechselwirkung und die Gluonen für die starke Wechselwirkung.

Bis heute konnte das Standardmodell in zahlreichen experimentellen Tests sehr erfolg-

reich bestätigt werden. Dies ist allerdings nur möglich, indem man mindestens 18 freie Parameter einführt, deren Werte gemessen werden müssen. Es erlaubt keine theoretischen Vorhersagen zu diesen Parametern. Weiterhin konnte bisher ein Teilchen das für die Theorie von großer Bedeutung ist, das Higgs-Teilchen, bisher noch nicht experimentell nachgewiesen werden. Von zukünftigen Experimenten erhofft man sich einerseits den Nachweis des Higgs-Teilches andererseits aber auch Hinweise auf mögliche Erweiterungen des Standardmodells, die es erlauben alle beobachteten Phänomene in einer Theorie mit möglichst wenig freien Parametern zu beschreiben.

Die oben beschriebenen Fragestellungen sind die Motivation für den Bau des Large Hadron Colliders (LHC) am CERN (Convention Européenne de la Recherche Nucléaire). Es handelt sich um einen ringförmigen Teilchenbeschleuniger mit einem Umfang von 27 km. Hier werden Protonen mit einer Schwerpunktsenergie von 14 TeV zur Kollision gebracht. Am LHC werden die Experimente ALICE, ATLAS, CMS, LHCb und TOTEM durchgeführt.

1.1.1 Das LHCb-Experiment

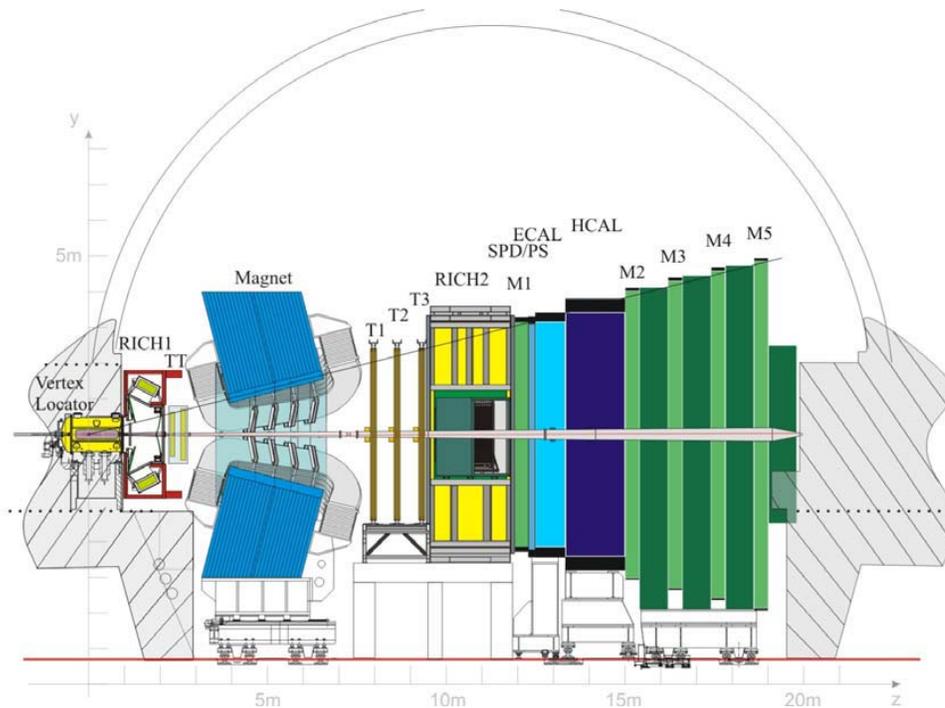


Abbildung 1.2: Der LHCb-Detektor. Das äußere Spurkammersystem (T1-T3) befindet sich hinter dem Magneten.

Die vorliegende Arbeit entstand im Rahmen des LHCb-Experiments. Ziel dieses Experiments ist die Untersuchung der CP-Verletzung und anderer seltener Phänomene, die bei den Zerfällen von B-Mesonen auftreten. Dieses ist im LHCb-Experiment mit bisher unerreichter Präzision möglich. LHCb wird bei einer Luminosität von $2 \times 10^{32} \text{ cm}^{-2} \text{ s}^{-1}$

betrieben. Man erwartet, daß in jedem Jahr der Datennahme ca. 10^{12} $b\bar{b}$ -Quarkpaare produziert werden. Aufgrund ihres hohen Longitudinalimpulses werden die entstehenden B-Mesonen unter kleinen Winkeln zum Strahlrohr produziert. Aus diesem Grund wird der LHCb-Detektor als Vorwärtsspektrometer realisiert. Der abgedeckte Raumwinkel liegt in der horizontalen Ebene (auch ablenkende Ebene genannt, da der Magnet die Bahnen der geladenen Teilchen horizontal ablenkt) zwischen 10 und 300 mrad, in der senkrechten zwischen 10 und 250 mrad [1].

Das Experiment wurde so optimiert [2], daß der Zerfall der B-Mesonen mit einer hohen Effizienz nachgewiesen werden kann. Der Detektor setzt sich zusammen aus dem Vertex-Locator (VELO), den inneren und äusseren Spurkammern (Inner Tracker (IT) und Outer Tracker (OT)), den elektromagnetischen und hadronischen Kalorimetern (ECAL bzw. HCAL), dem Ring-Imaging Cherenkov-Zähler (RICH) sowie dem Myonensystem (siehe Abbildung 1.2).

Hinter dem Wechselwirkungspunkt ist ein Dipolmagnet platziert, der die Flugbahn der bei der Kollision entstandenen Teilchen ändert. Durch das Vermessen der Spuren im Spurkammersystem kann der Impuls dieser Teilchen bestimmt werden. Das Spurkammersystem besteht aus zwei separaten Teilen: einer inneren und einer äusseren Spurkammer. Die innere basiert auf einer Siliziumstreifentechnologie. Der nicht von der inneren Spurkammer abgedeckte Bereich wird von der äusseren übernommen, welche auf der Strawtechnologie basiert. Der in dieser Arbeit getestete OTIS-TDC (Time to Digital Converter) ist ein zentraler Bestandteil der Ausleseelektronik des äusseren Spurkammersystems.

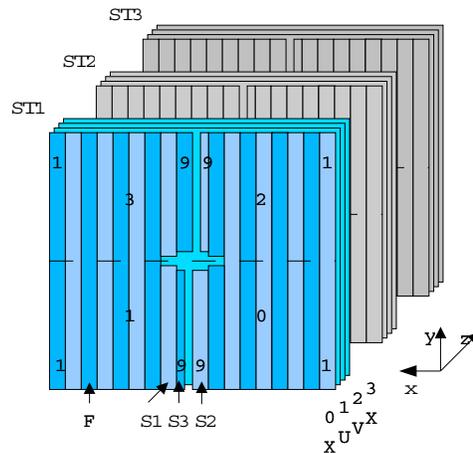


Abbildung 1.3: Drei Stationen bilden das äußere Spurkammersystem (T1-T3). In einer Station sind die vier Lagen entweder senkrecht (X), um $+5^\circ$ (U) und um -5° (V) angeordnet.

Das äussere Spurkammersystem wird als Driftkammerdetektor realisiert. Dieser besteht aus drei identischen Stationen, die sich wiederum aus vier Stereolagen zusammensetzen. Insgesamt wird vom gesamten Spurkammersystem eine Fläche von 300 m^2 durch die Straw-Detektoren abgedeckt. Die erste Lage ist senkrecht aufgehängt, die zweite um $+5^\circ$ und die dritte um -5° gedreht zu dieser. Die vierte Lage wird wiederum senkrecht

aufgebaut. Durch diese Stereolagen kann die Lage der Spuren entlang der y-Achse gemessen werden (siehe Bild 1.3). Jeweils 256 Strawtubes von 2.5 m Länge sind Teil eines 5 m langen Moduls.

Bei den Strawtubes handelt es sich um Driftröhrchen. Sie sind aufgebaut wie ein Proportionalzählrohr. Die zusätzlich Bestimmung der Driftzeit der Elektronen erlaubt es jedoch den Ort des Teilchendurchganges mit einer Auflösung von $200\mu\text{m}$ zu vermessen. Das Driftröhrchen besteht aus dem leitfähigen Strawtube mit einem Durchmesser von 5 mm und dem Anodendraht. Zwischen dem leitenden Straw und dem Anodendraht wird eine Hochspannung angelegt. Der Straw ist mit einem Zählgas (Ar/CO₂) gefüllt. Durchquert ein ionisierendes Teilchen das Zählgas, werden Elektronen und Ionen erzeugt. Das elektrische Feld zwischen Kathode und Anode bewirkt eine Stoßentladung von Elektronen, die auf den Anodendraht hin beschleunigt werden. Die Driftzeit der durch die Primärionisation erzeugten Elektronen zum Anodendraht beträgt typischerweise unter 50 ns. Die Driftzeitinformation ist wichtig, da sie erlaubt, den radialen Abstand der Spur vom Anodendraht mit einer Auflösung von $200\mu\text{m}$ zu vermessen. Die Signale von 55.000 Kanälen des Outer Tracker werden von ASD-Chips (Amplifier Shaper Discriminator) vorverstärkt und an den OTIS-TDC weitergeleitet.

1.2 Die Outer Tracker Ausleseelektronik

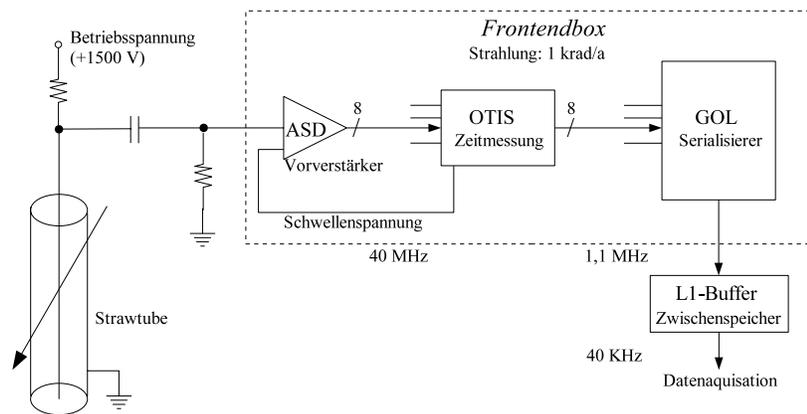


Abbildung 1.4: Schema der Ausleseelektronik

Abbildung 1.4 zeigt schematisch die in [3] entwickelte Auslesekette. Die Proton-Proton-Wechselwirkungsrate im LHC-Experiment beträgt 40 MHz, daß heißt alle 25 ns findet ein Bunch-Crossing (BX) statt. Für jedes Bunch-Crossing nimmt die Elektronik einen Meßwert auf. Die Entscheidung, ob ein Meßwert gespeichert wird, fällt in einem dreistufigen Triggersystem. Aufgrund der Strahlenbelastung von mehreren kRad für die Dauer des Experiments müssen die entwickelten Elektronikkomponenten strahlentolerant sein. Für das Auflösungsvermögen der Driftzeitmessung ist ein Wert unter einer Nanosekunde gefordert.

Die Auslese der äußeren Spurkammern erfolgt über folgende drei Chips:

- der ASDBLR-Vorverstärker (Amplifier Shaper Discriminator with BaseLine Restoration)
- der OTIS-TDC zur Bestimmung der Driftzeiten
- der GOL (Gigabit Optical Link) -Serialisierer zur Versendung der Daten über ein Glasfaserkabel zum L1-Buffer

ASD-Vorverstärker

Ein analoges Ladungssignal (von ca. 60 fC) aus den Driftkammern gelangt zunächst zu dem ASDBLR Vorverstärker Chip [5]. Hier wird das Signal von acht Detektorkanälen verstärkt und diskriminiert. Für die 32 Kanäle eines OTIS-Chips sind daher vier ASD-Vorverstärker vorgesehen. Liegt ein Ladungspuls über einer vorher festgelegten Schwelle, erzeugen sie ein differentielles Ausgangssignal für die OTIS-Karte [4].

OTIS TDC

Der Outer Tracker Time Information System (OTIS) Chip [14] arbeitet als Time to Digital Converter (TDC) und mißt den Zeitpunkt, zu dem ein diskriminierter Ladungspuls des Vorverstärkers anlag relativ zum Zeitpunkt der Wechselwirkung mit einer Genauigkeit von < 1 ns. Die Protonen kollodieren im LHC-Beschleuniger mit einer Rate von 40,08 MHz. Der OTIS generiert eine Driftzeit und leitet sie nach einer positiven L0-Triggerentscheidung an die optische Senderkarte (GOL-Aux-Karte [6]) weiter. Die maximale mittlere Rate, mit der die L0-Triggerentscheidungen akzeptiert werden können, beträgt 1,1 MHz.

Gigabit Optical Link

Der Gigabit Optical Link (GOL) [7] serialisiert die 8 Bit OTIS-Daten zu einem 1,6 GBit/s Datenstrom und speist sie über eine VCSEL-Diode (Vertical Cavity Surface Emitting Laser) in eine optische Faser ein. Mit den 128 Anschlüssen einer Frontendbox können die Kanäle eines Driftkammermoduls ausgelesen werden. Dazu sind in der Frontendbox acht Vorverstärkerkarten, vier TDC-Karten und eine optische Senderkarte untergebracht.

Die weitere Datenverarbeitung findet außerhalb des Detektorbereichs statt. Von einer optischen Empfängerkarte (O-RXCard [8]) werden die serialisierten Daten von bis zu zwölf optischen Senderkarten entgegengenommen und in parallele elektrische Signale zurücktransformiert. Zwei optische Empfängerkarten befinden sich auf der sogenannten TELL1-Karte [9], mit deren Hilfe die Daten im Outer-Tracker-Datenformat abgespeichert werden. Die physikalisch interessanten Daten werden von hier aus zur weiteren Auswertung mit maximal 40 kHz an eine PC-Farm (>1000 PCs) geschickt.

1.2.1 Steuerung der Elektronik

Die Elektronik wird von zwei Systemen kontrolliert. Das erste System wird synchron zum 40 MHz Takt betrieben und nennt sich „Timing an Trigger Control System“ (TTC-System) oder auch „Fast Control“-System. Es ist für die Verteilung der Triggersignale,

der Resetsignale, der Testpulse und der Taktsignale zuständig. Diese Signale werden an die Frontendboxen verteilt. Dort sorgen die optischen Senderkarten für die weitere Verteilung der Signale an die angeschlossenen Chips.

Das zweite System ist die „Slow Control“. Mit diesem können die Register in den OTIS-Chips gesetzt und weitere Einstellungen vorgenommen werden. Diese Einstellungen werden während einer Messung nicht verändert, weshalb das verwendete Protokoll nicht mit 40 MHz betrieben werden muß. In der Ausleseketten des äußeren Spurkammersystems wird eine I²C-Logik [16] für die Kommunikation mit dem OTIS verwendet.

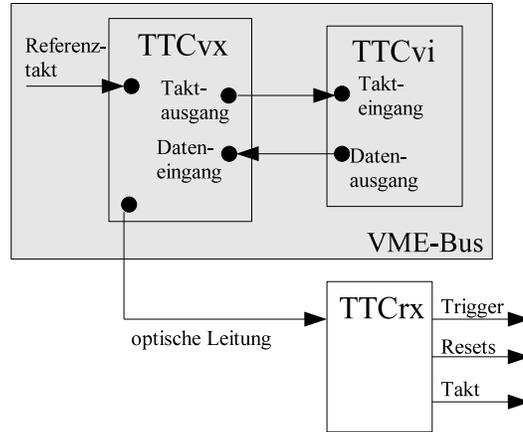


Abbildung 1.5: Schematischer Überblick über das lokale TTC-System

Am Physikalischen Institut existiert ein lokale TTC-System. Es besteht aus den in Abbildung 1.5 gezeigten Komponenten. Das TTCvx-Modul [10] nimmt einen Referenztakt entgegen und gibt diesen an das TTCvi-Modul [11] weiter. Hier wird nun ein Datensignal generiert, das alle Trigger-, Reset- und Taktsignale enthält. Dieses Datensignal wird an das TTCvx-Modul zurückgegeben und in ein optisches Signal umgewandelt. Die optischen Signale werden an den TTCrx-Chip [12] übertragen und von diesem auf einzelnen Pins als TTL-Signal zur Verfügung gestellt.

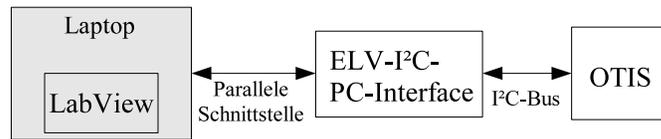


Abbildung 1.6: Schematischer Überblick über die lokale Slow Control

Für die Slow-Control des OTIS (Abbildung 1.6) dient ein externer Rechner, der mittels einer LabView-Applikation die einzelnen Register des OTIS beschreiben und auslesen kann. Die Kommunikation zwischen Rechner und OTIS findet über das I²C-Protokoll [16] statt. Dazu ist eine kommerzielle ELV-I²C-PC-Interface-Box zwischengeschaltet. Sie generiert aus dem parallelen Datenstrom von der Druckerschnittstelle des Rechners die seriellen I²C-Daten für den OTIS.

1.3 Der OTIS Time to Digital Converter

Der OTIS TDC (Time to Digital Converter) [14] ist ein Schlüsselement in der Auslekette des äußeren Spurkammersystems. Er wurde am ASIC-Labor der Universität Heidelberg entwickelt. Der Chip hat die Aufgabe, Driftzeiten in Relation zum LHC-Takt zu messen. Dies soll er mit einer Auflösung von unter 1 ns bewerkstelligen. Er wird als strahlenharter Chip in einem $0.25\mu\text{m}$ -CMOS-Prozess hergestellt. Es existieren verschiedene Versionen des Chips. Die erste Version 1.0 wurde im Mai 2002 submittiert. In dieser Arbeit wurde mit den Versionen 1.1, 1.2 und der vor wenigen Wochen aus der Produktion zurückgekommenen OTIS-Version 1.3 gearbeitet.

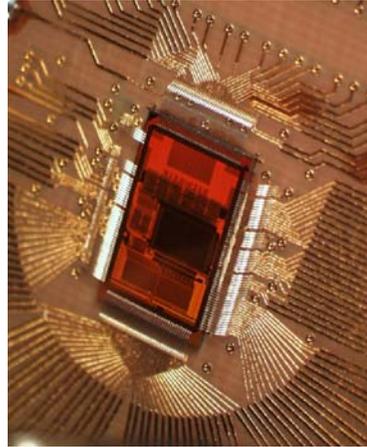


Abbildung 1.7: Der OTIS-TDC

Der OTIS wird mit der 40 MHz LHC-Frequenz getaktet. Eine DLL (Delay Locked Loop) teilt einen Taktzyklus zur Messung der Driftzeit in 64 Zeitbins. Das ergibt eine Auflösung von 390 ps. Die DLL ist dafür aus 64 Invertern aufgebaut (Abbildung 1.8). Durch sie propagiert der 40 MHz Systemtakt. Sobald ein Ladungspuls des Vorverstärkers einen Treffer anzeigt, wird vom Hit-Register geprüft, in welchem der Bins der 0-1-Übergang des 40 MHz Systemtaktes stattgefunden hat. Die Nummer des Bins wird als 6 Bit Driftzeit kodiert.

Phasendetektor und Charge Pump haben die Aufgabe, die Verzögerungszeiten der DLL nachzuregeln. Dies ist erforderlich, da die Verzögerungszeiten der Inverter abhängig

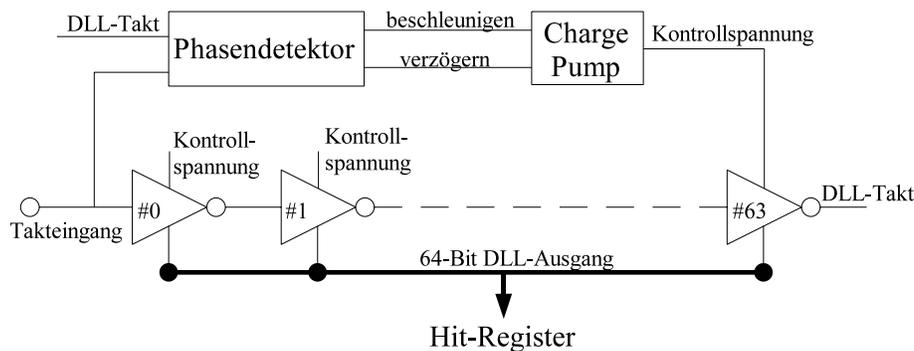


Abbildung 1.8: Blockdiagramm der DLL. Der Phasendetektor mißt die Phasendifferenz zwischen Referenztakt und seiner verzögerten Version. Die Durchlaufzeiten durch die Verzögerungselemente werden mit der Kontrollspannung reguliert.

von der anliegenden Versorgungsspannung und von der Betriebstemperatur sind. Der Phasendetektor in Kombination mit der Charge Pump verhindert, daß die Länge der DLL vom geforderten Wert von 25 ns abweicht. Die Beschleunigungs- bzw. Verzögerungspulsängen aus dem Phasendetektor variieren mit der Phasendifferenz zwischen Dll-Takt und der Frequenz am Takteingang. Die Kontrollspannung der Charge Pump wird entsprechend den Ausgabesignalen des Phasendetektors entweder erhöht oder erniedrigt. Dies resultiert in steigenden oder fallenden Propagationszeiten der Taktflanken durch die Inverter.

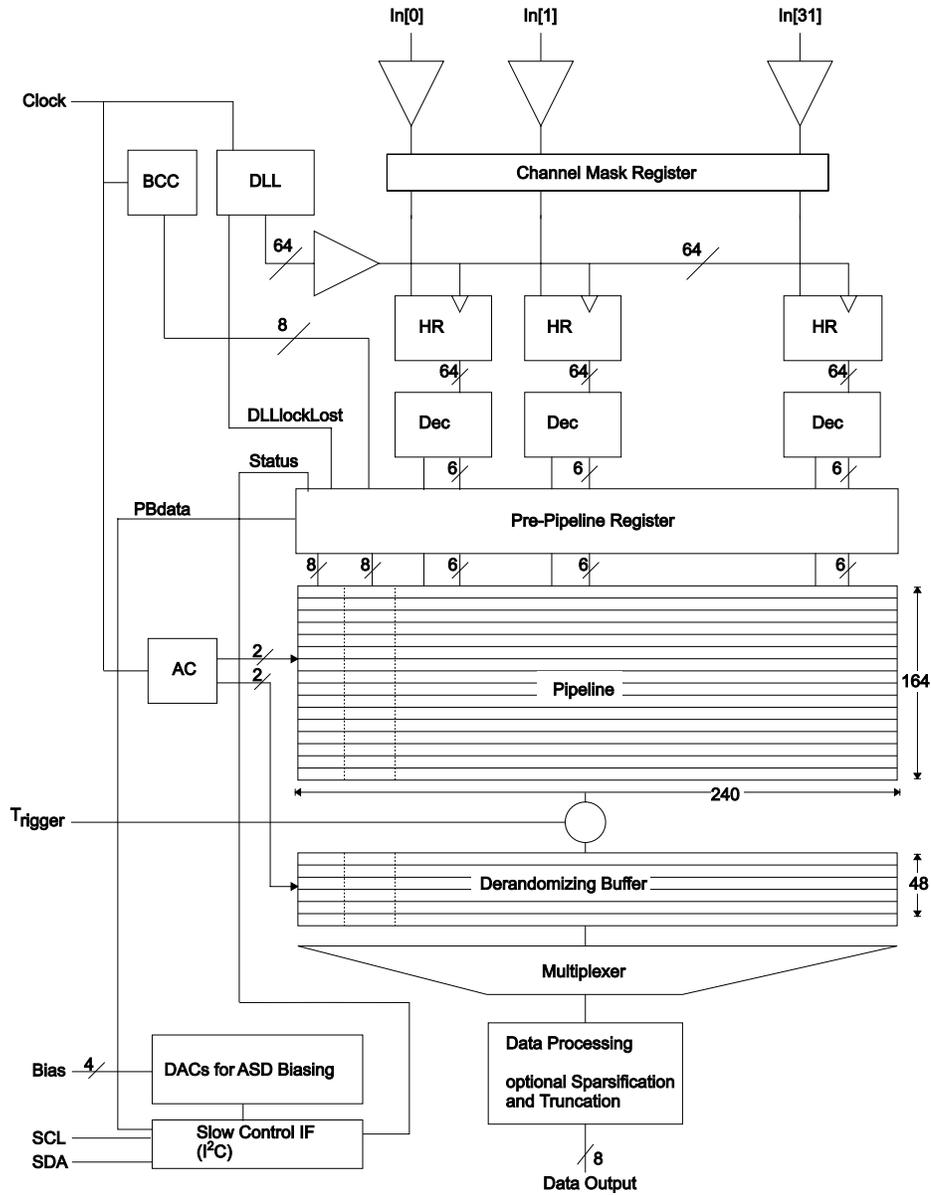


Abbildung 1.9: Blockschaltbild des OTIS TDC

Abbildung 1.9 zeigt ein Blockdiagramm des OTIS. Der Inhalt des Hit-Registers (HR) und des Bunch-Crossing Zählers (BCC) wird mit jedem Taktzyklus in das Pipeline-Register übertragen. Ein Schreibzeiger erhöht mit jedem Taktzyklus die Speicheradresse, in der die eingehenden Daten aus dem Hit-Register abgelegt werden. Sobald er auf Adresse 163 angelangt ist, springt er wieder auf die erste Position zurück.

Bekommt der OTIS nun vom TTC-System einen Trigger geschickt, so schaut er von seiner aktuellen Position aus in der Pipeline die im Latency-Register eingestellte Anzahl von Zeilen zurück. In dieser auszulesenen Adresse wird für jeden Kanal nach Treffern gesucht. Diese werden als TDC-Zeit an den 48 Zeilen tiefen Derandomizing Buffer weitergegeben.

Wird das Meßintervall des OTIS auf 75 ns eingestellt, werden zwei weitere Zeilen aus den Bunch-Crossings N-1 und N-2 nach Treffern durchsucht. Es besteht die Möglichkeit, die Suchtiefe auf ein, zwei oder drei Bunch-Crossings einzustellen, um Driftzeiten von maximal 25, 50 oder 75 ns zu messen.

Output Data Format (SingleHit Mode)

| | | | | | |
|--------------|--------|--------------|--------------|-----|---------------|
| Bit: | 0..31 | 32..39 | 40..47 | ... | 280..287 |
| Data: | Header | Drift Time 0 | Drift Time 1 | ... | Drift Time 31 |

Abbildung 1.10: Ein Datensatz des OTIS

Die so erzeugten TDC-Zeiten werden vom OTIS zu einem Datensatz zusammengefügt (siehe Abbildung 1.10). Die ersten vier Byte eines Datensatzes bilden den Datenkopf. In ihm befinden sich die Statusbits des OTIS und seine 12 Bit große Identifikationsnummer. In den folgenden 32 Byte stehen die Driftzeitinformationen. Sie setzen sich zusammen aus zwei Bit, die anzeigen, in welchem Bunch-Crossing ein Treffer stattgefunden hat, sowie aus der 6 Bit TDC-Zeit. Die Kodierung des Bunch-Crossings ist in Bild 1.11 gezeigt. Mit der Kodierung „10“ für das dritte Bunch-Crossing sind Driftzeiten von 0 bis 191 möglich, mit der Kodierung „00“ können Driftzeiten von 0 bis 63 erzeugt werden. Die Kombination „11“ bedeutet, daß kein Treffer gefunden wurde.

| Hit Position | Drift Time Encoding |
|--------------|---------------------|
| 1. BX | 00XXXXXX |
| 2. BX | 01XXXXXX |
| 3. BX | 10XXXXXX |
| No Hit | 11XXXXXX |

Abbildung 1.11: Kodierung der Driftzeiten

Kapitel 2

Anforderungen an den OTIS-TDC Serientest

2.1 Motivation

Ziel dieser Arbeit war die Erstellung eines geeigneten Testaufbaus für den OTIS TDC. Aufgrund der begrenzten Zeitspanne, die für den funktionalen Test von mehr als 3500 OTIS-Chips zur Verfügung steht, sollte die Testdauer pro Chip nicht mehr als eine Minute betragen. In dieser Zeit muß der Chip im Waferprober angefahren werden, alle funktionalen Tests durchlaufen und die Überprüfung der Linearität der Zeitmessung absolviert haben.

Es gab mehrere Beweggründe, die zentrale Steuerung und Datenverarbeitung des OTIS-Serientests in Hardware umzusetzen. Als erstes benötigt man ein zentral steuerbares Kontrollsystem für den OTIS, das aus Platzgründen im Reinraum nicht von mehreren externen Komponenten aufgebaut werden kann, wie das im bisherigen Elektroniklabor im Physikalischen Institut der Fall ist. Eine Implementierung auf einem FPGA erleichtert die Handhabung des Testaufbaus enorm, da sie die Eingabe von z.B. Auslesefrequenz oder das Setzen von Rücksetzsignalen per PC möglich macht. Ein weiterer Grund ist die Zeitlimitierung für den Chip-Test, der nach Lieferung der produzierten Wafer innerhalb weniger Wochen absolviert sein muß.

So ist die Bestimmung der differentiellen Nichtlinearität (DNL), die ein Maß für die Genauigkeit der Zeitmessung des TDC ist, auf dem FPGA deutlich schneller, als die bisherige Methode. Der OTIS besitzt 64 Zeitbins für die Messung von Driftzeiten, die alle innerhalb einer gewissen Tolleranzschwelle die gleiche Größe besitzen müssen. Um dies zu verifizieren, ist eine Testsequenz vorgesehen, in der zufällige Driftzeiten produziert werden. Der OTIS mißt den Zeitpunkt dieser Pulse relativ zu einem 40 MHz Referenztakt. Unter Vorgabe einer Triggerfrequenz werden diese Pulse vom OTIS als Driftzeit erkannt. Bildet man aus den Driftzeiten ein Histogramm, erwartet man in allen Zeitintervallen gleichverteilte Trefferzahlen. Im Falle einer Softwarelösung müssen alle Rohdaten zur Auswertung auf einen PC übertragen werden. Um eine aussagekräftige Statistik zu erhalten, wird hier eine Zahl von einer Millionen zufällig generierten Driftzeiten betrachtet, die in 64 Zeitbins histogrammiert werden. Das ergibt bei einer Datensatzgröße von 36 Byte pro Ereignis ein Gesamtdatenvolumen von ca. 34 MegaByte, das für die Erstellung eines Driftzeithistogramms über den PCI-Bus auf den PC zu übertragen ist. Da

der PCI-Treiber aufgrund seiner begrenzten Übertragungskapazität nur eine maximale Triggerrate von etwa 3 KHz zulässt, beträgt die Gesamtzeit für die Datenübertragung 6 Minuten. Man beachte, innerhalb dieser Zeit nur das Histogramm eines Eingangskanals des OTIS von insgesamt 32 aufgenommen wurde.

Die Hardware-Lösung, bei der die Rohdaten direkt vom FPGA empfangen und mit 40 MHz interner Taktrate weiterverarbeitet werden beschleunigt den Testablauf um ein vielfaches. Zieht man erneut das Beispiel für die Berechnung der DNL heran, werden sowohl Histogrammierung der Driftzeiten, als auch die Berechnung der DNL vom FPGA übernommen, bevor das Endergebnis an den PC weitergeleitet wird. Für Analysezwecke ist es zweckmäßig, das fertige Histogramm, sowie die DNL auszugeben. Die hohe Verarbeitungsgeschwindigkeit des FPGA erlaubt hier eine maximale, durch den OTIS limitierte Triggerrate von 1,1 MHz. Nach einer kompletten Messung müssen also ein Kilobyte für ein Histogramm (64×16 Bit) und ein Byte für den Wert der DNL über den PCI-Bus abgerufen werden¹. Die Limitierung ist nun also nicht mehr durch die Übertragungsgeschwindigkeit des PCI-Busses, sondern durch die Geschwindigkeit des OTIS festgelegt. So wird eine Gesamtdauer von unter einer Sekunde erreicht, d.h. die Messung wird um den Faktor 365 schneller. Da zum erfolgreichen Bestehen eines kompletten Tests die DNL von mindestens vier Kanälen analysiert werden soll, kommt die Softwarelösung nicht in betracht.

2.2 Die Funktionstest am OTIS

Zunächst wird ein Überblick über die zu testenden Funktionalitäten des Chips gegeben. Die Tests sind in Tabelle 2.1 aufgelistet. Sie lassen sich in zwei Kategorien unterteilen. Zur ersten Kategorie gehören die Tests zur digitalen Funktionalität des OTIS. Der Test zur Genauigkeit der Driftzeitmessung des OTIS bildet die zweite Kategorie.

| Funktionstests | |
|-----------------------|-----------------------------------|
| 1. | I ² C-Programmierung |
| 2. | Vorverstärker-Schwellenspannungen |
| 3. | Chip-Identifikationsnummer |
| 4. | Chip-Counter |
| 5. | Arbeitsbereich der DLL |
| 6. | Latenzzeit |
| 7. | Integrität des Speichers |
| 8. | Ansprechbarkeit aller Kanäle |
| Leistungstest | |
| 9. | Charakterisierung der Zeitmessung |

Tabelle 2.1: Auflistung aller Tests, die stattfinden sollen

Im folgenden wird auf die einzelnen Tests genauer eingegangen:

¹Die Busbreite von 16 Bit pro Zeitbin wurde im FPGA Desing festgelegt und erlaubt daher pro Zeitbin die Aufnahme von bis zu 65535 Einträgen.

Die Funktionalen Tests

I²C-Programmierung: Um Einstellungen am OTIS vornehmen zu können, beispielsweise für den Auslesemodus oder die Latenzzeit, sind auf dem Chip mehrere Register vorgesehen. Diese lassen sich über das I²C-Protokoll ansprechen und beschreiben. Zu Beginn eines jeden Test wird als erstes überprüft, ob diese Grundfunktionalität gegeben ist.

Vorverstärker Schwellenspannungen: Im Outer Tracker stellt ein OTIS-Chip die Schwellenspannung für vier ASD Verstärker Diskriminator-Chips bereit. Auf dem OTIS sind dazu vier Digital-Analog-Wandler integriert. Ihr Variationsintervall für die Ausgangsspannung reicht von 0 Volt bis 2,5 Volt, bei einem Auflösungsvermögen von 8 Bit. Demzufolge läßt sich die Stufengröße berechnen zu:

$$V = \frac{2,5V}{2^8} = 9,8 mV \quad (2.1)$$

Das Setzen der Register auf dem OTIS für die Schwellenspannung erfolgt über I²C. Im Serientest muß gezeigt werden, daß die Schwellenspannungen mit den entsprechenden Einträgen in den DAC-Registern des OTIS linear anwachsen.

Konfiguration der Identifikationsnummer des OTIS: Die 12 Bit Identifikationsnummer beschreibt die Position des Chips innerhalb des Detektors. Das Setzen der Bits wird im Serientest überprüft. Dazu ist eine Routine vorgesehen, die zwei unterschiedliche Bitmuster für die Identifikationsnummer erzeugt. Diese Bitmuster müssen sich bei einem funktionierenden Chip in seinem Datenkopf wiederfinden lassen.

Event- und Bunch-Clock-Zähler: Der OTIS-Chip zählt alle angenommenen und abgelehnten Trigger, sowie die Zahl der ausgehenden Datensätze. Während die ersten beiden Zählwerte nur per I²C-Protokoll aus den OTIS-Registern auslesbar sind, werden die vier Least Significant Bits (LSBs) des 16 Bit-Zählers für die ausgehenden Datensätze im OTIS-Datenkopf als Event-ID hinterlegt. Mit dem Event-ID-Reset läßt sich der Zähler auf Null zurücksetzen. Ein Trigger wird immer abgelehnt, wenn der Derandomizing Buffer voll ist. Es gilt:

$$n_e = EventID + n_a \quad \begin{array}{l} n_e = \text{Zahl der empfangenen Trigger} \\ n_a = \text{Zahl der abgelehnten Trigger} \end{array}$$

Ausserdem besitzt der OTIS den 12 Bit breiten Bunch-Clock-Zähler, der synchron zur LHC-Frequenz mit jedem Takt um eins inkrementiert wird. Die Differenz des Zählers zwischen zwei OTIS-Datensätzen entspricht demnach dem Abstand zwischen zwei Triggerentscheidungen. Die ersten acht Bit des Bunch-Crossing-Zählers werden ebenfalls in den OTIS-Datenkopf geschrieben und sind mit dem Bunch-Crossing-Reset zurücksetzbar. Das richtige Inkrementieren beider Zähler ist im Serientest nachzuweisen.

DLL-Arbeitsbereich: Die als „Full Custom Design“ implementierte DLL bildet den Kern des OTIS. Damit sie zur Driftzeitmessung herangezogen werden kann, muß die durch sie propagierende Taktfrequenz phasenstarr zur LHC-Frequenz sein. Hierzu dient der

Phasenkomparator, der je nach Phasendifferenz die Spannungsgeregelten Verzögerungselemente über die Kontrollspannung (im weiteren mit V_{ctrl} bezeichnet) nachregelt, bis LHC-Takt und OTIS-Takt übereinstimmen. Der Chip befindet sich dann im sogenannten „locked“ Zustand. Je nach anliegender Betriebsspannung oder Betriebstemperatur ändert sich der Wert der Kontrollspannung. In einem OTIS-Register kann man eine untere und eine obere Grenze vorgeben, innerhalb derer sich die Kontrollspannung bewegen darf. Ein Über- bzw. Unterschreiten dieser Grenzen wird durch das „DLL-Lock-Lost Bit“ angezeigt, dessen Wert über den OTIS-Datenkopf mit ausgegeben wird.

Latenzzeit: Im Pipeline-Register des OTIS werden die eingehenden Daten zwischengespeichert. Nach Empfang der Triggerentscheidung mit einer Latenzzeit von maximal $4.1\mu s^2$, werden die Daten entweder verworfen oder an den Derandomizing Buffer weitergeleitet. Die Vorgänge des Ein- bzw. Auslesens des Pipeline-Registers werden von den Lese- und Schreibzeigern kontrolliert. Sie zeigen an, welche Adresse im Speicher gerade ausgelesen oder beschrieben wird. Mit jedem Taktzyklus erhöht sich die Adresse der beiden Zeiger bis sie nach Speicherplatz 163 wieder auf Adresse null zurückspringen. Die Latenzzeit gibt die Zahl der Zeilen zwischen dem Schreib- und dem Lesezeiger an. Der Lesezeiger läuft dem Schreibzeiger um die eingestellte Latenzzeit hinterher.

Der OTIS besitzt zwei Ausgänge für die Monitor-Signale. Diese können verschiedene interne Signale des OTIS nach Außen geben. Unter anderem besteht so die Möglichkeit, die Nulldurchgänge des Lese- und Schreibzeigers über diese Signale darzustellen und zu überprüfen.

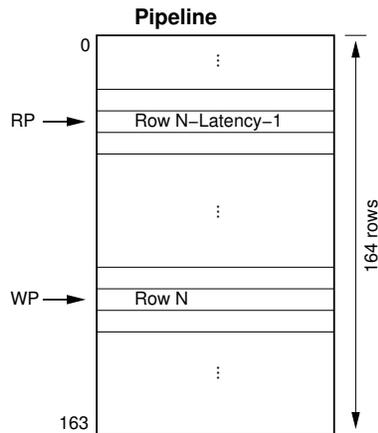


Abbildung 2.1: Das Pipeline-Register [15]. Schreib- und Lesezeiger haben den Abstand Latency

Memory-Selftest: Der Memory Selftest stellt eine interne Testroutine des OTIS dar. Ein regelmäßiges Muster wird aus dem Playback-Register in alle Speicherzellen geschrieben. Nach dem Auslesen der Zellinformationen werden die erhaltenen Bitwerte mit dem vorgegebenen Muster verglichen. Der Test startet automatisch nach jedem Power-Up-Reset und L0-Reset. Das Ergebnis der Testauswertung wird im OTIS-Datenkopf hinterlegt. Innerhalb der Serientestprozedur wird der Status des Memory-Selftest in mehreren Se-

²das entspricht und 164 Taktzyklen und damit genau der Tiefe des Pipeline-Registers

quenzen abgefragt.

Buffer Overflow: Der Derandomizing Buffer ist dem Pipeline-Register nachgeschaltet und hat eine Tiefe von 48 Zeilen. Das Buffer-Overflow-Signal im OTIS-Datenkopf wird aktiviert, wenn der Derandomizing Buffer vollläuft. Im 1 BX-Modus (25 ns Meßintervall) ist der Speicher nach 49 konsekutiven Triggern gefüllt. Hierbei ist zu berücksichtigen, daß gleichzeitig zur Datenentgegennahme nach 36 Taktzyklen wieder ein Datensatz rausgeschrieben wurde³. Im Serientest wird der OTIS im 3 BX-Auslesemodus betrieben. In diesem Modus werden nach dem ersten Trigger die Inhalte von drei Zeilen des Pipeline-Registers übertragen. In der dritten Zeile wird nach einem Treffer gesucht. Die Driftzeit setzt sich dabei zu $T+50\text{ ns}$ ($0\text{ ns} \leq T \leq 25\text{ ns}$) zusammen, wobei T die 64 Bit Driftzeit der zuletzt übertragenen Zeile darstellt.

Mit jedem folgenden konsekutiven Trigger wird jeweils nur eine weitere Zeile übertragen. Nach einem konsekutiven Trigger mit einer Länge von 46 Taktzyklen sind alle Speicherplätze belegt.

Im Serientest wird ein Überlauf des Buffers künstlich provoziert und das entsprechende Errorbit überprüft. Ausserdem wird getestet, wieviele Datensätze der Speicher maximal aufnehmen kann, bevor eine Fehlermeldung des OTIS ausgegeben wird.

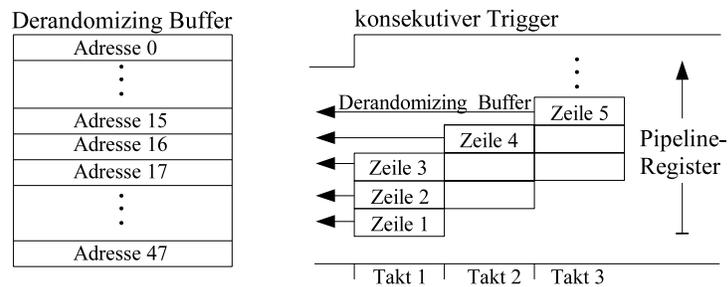


Abbildung 2.2: Verhalten des OTIS bei Empfang von konsekutiven Triggern im 3 BX Auslesemodus. Nach dem ersten Taktzyklus werden im Derandomizing Buffer drei Datensätze aus dem Pipeline-Register empfangen, mit jedem folgenden Taktzyklus ein weiterer. nach 46 Taktzyklen ist der Speicher voll.

Ansprechbarkeit aller Kanäle: In diesem Test soll die Funktionalität aller Kanäle des OTIS nachgewiesen werden. Dazu wird auf dem FPGA ein Histogramme der Trefferverteilung über alle Kanäle gebildet. Die Treffer werden von einem Pulser erzeugt und statistisch gleichverteilt auf alle 32 Hiteingänge des OTIS gelegt. Im Detektorbetrieb besteht die Möglichkeit, einzelne Kanäle des OTIS auszumaskieren. Diese Funktion wird bereitgestellt, da im Detektor ein defekter Kanal, beispielsweise hervorgerufen durch einen Kurzschluss in einem der Straw-Tubes, bei weiteren Messungen nicht mehr berücksichtigt werden soll.

Konsistenz der Header-Einstellungen: Weitere Informationen im Header, auf die nicht weiter eingegangen wird, sind der Auslesemodus (Single-Hit bzw. Multi-Hit), der Truncation-

³Die Daten werden Byteweise mit jedem Takt aus dem Derandomizing Buffer rausgeschrieben.

Modus und der Playback-Modus. Diese Werte werden für den Serientest konstant bei einer Einstellung belassen und sollten sich in den Header-Statusbits wiederfinden.

Der Performance Test

Die Leistungscharakteristik des Chips spiegelt sich im Wert für die differentielle Nicht-linearität (DNL) wieder. Sie beschreibt das maximale Auflösungsvermögen des TDC-Chips. Hierzu wird untersucht, wie sich die einzelnen Zeitbins hinsichtlich ihrer Länge unterscheiden. Idealerweise liefert der OTIS eine Zeitauflösung von 390 ps. Es wird allerdings erwartet, daß sich die Inverter der DLL-Kette hinsichtlich ihrer Signalverzögerungszeit leicht unterschiedlich verhalten und somit verschieden große Zeitbins entstehen. Bedingt wird dies unter anderem durch die verschiedenen Ausdehnungen der PMOS und NMOS Transistoren innerhalb der Inverter. Dieser Effekt ist auf die Art des Herstellungsprozesses des OTIS rückführbar und wird hier nicht weiter erläutert.

2.3 Schematischer Überblick über den Testaufbau

Ein schematische Aufbau des im ASIC-Labor stattfindenden Tests ist in Abbildung 2.3 zu sehen. Der Test des OTIS findet im Waferprober statt. Die einzelnen Chips werden

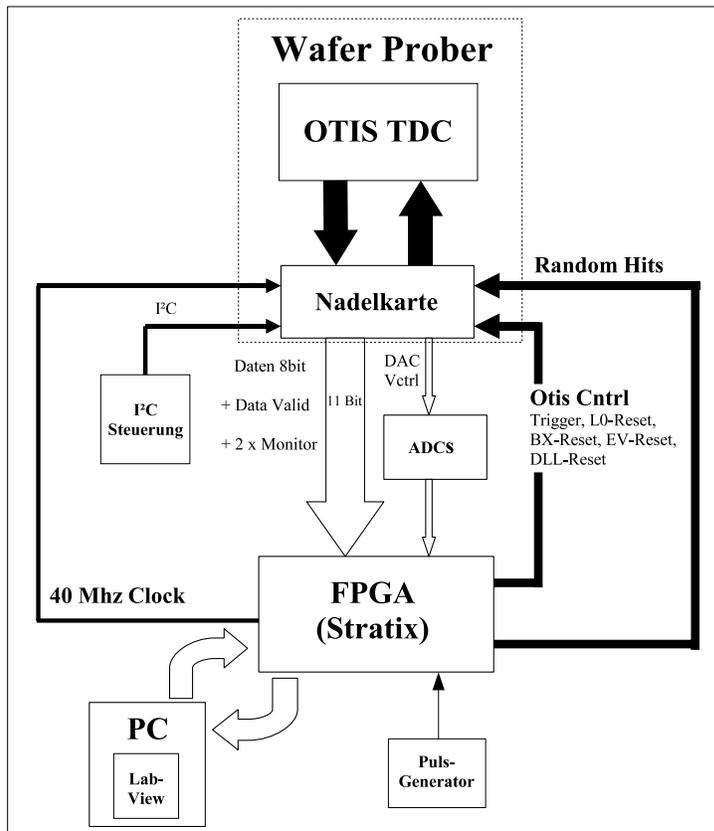


Abbildung 2.3: Aufbau der Testumgebung für den Serientest des OTIS

auf dem Wafer über eine Nadelkarte angesteuert. Diese sorgt für die Kommunikation zwischen dem OTIS und den Einheiten, die seine Signale empfangen bzw. die seine Steuersignale und Testpulse bereitstellen.

Steuerung des OTIS

Die Steuerung des OTIS erfolgt durch zwei Komponenten:

- die Slow-Control zur asynchronen Programmierung und Auslese der OTIS-Statusregister
- die Fast-Control, die synchron zum 40 MHz-Takt betrieben wird

Letzteres wurde in den bisherigen Versuchsaufbauten am Physikalischen Institut von mehreren externen Modulen in einem VME-Rahmen aufgebaut. Sie dienen zur Erzeugung von Systemtakt, Resetsignalen, Triggerentscheidungen, sowie von Testpulsen. In dieser Arbeit wurde das TTC-System auf dem FPGA implementiert, um den Aufbau kompakter zu gestalten.

Folgende Einstellungen am OTIS können so vorgenommen werden:

1. Wahl des Auslesemodus, z.B. der Meßbereich von 25, 50 oder 75 ns
2. Setzen von Schwellenspannungen für die ASD-Verstärker
3. Vorgabe einer unteren und einer oberen Grenzspannung für die DLL (Delay Locked Loop)
4. Einstellung der Funktion der Monitor-Anschlüsse
5. Maskieren der Eingangskanäle
6. Einstellung der Latenzzeit in Taktzyklen
7. Umschalten zwischen normalem Betriebs- und Debugmodus
8. Programmierung der Playback-Daten

Neben der Übertragung der Einstellwerte für die Statusregister, lassen sich während des Betriebs verschiedene Informationen des OTIS über I²C zurücklesen. Unter anderem erhält man die Werte für die Zahl der abgelehnten Trigger, die Zahl der akzeptierten Trigger, die Zahl der ausgegebenen Datensätze und die Identifikationsnummer.

Pulserzeugung

Des weiteren werden externe Pulse von einem Pulsgenerator⁴ bereitgestellt, die über den FPGA je nach Einstellung auf alle geraden oder alle ungeraden Kanäle des OTIS verteilt werden. Diese Pulse kommen asynchron zum OTIS-Taktsignal und verhalten sich damit für den OTIS wie zufällige Treffer. So können über eine große Anzahl von Ereignissen gemittelt alle Driftzeiten gleichverteilt vom OTIS empfangen werden. Dadurch wird die Messung der DNL ermöglicht.

⁴dieser Pulsgenerator wurde später durch einen Oszillator auf der FPGA-Karte ersetzt.

OTIS Rückgabewerte

Schließlich werden mit dem FPGA die Informationen des OTIS zurückgelesen und ausgewertet. Dazu gehören die Kontrollspannung des OTIS (V_{ctrl}) und die Schwellenspannungen (DAC-Werte) für die Verstärker Diskriminatoren. Hinzu kommen die Driftzeitinformationen, die in einem 8 Bit-Datenstrom übermittelt werden, sowie das Data-Valid-Signal. Neben den Driftzeitdaten wird der Datenkopf, der Auskunft über den Status des OTIS gibt, mit dem Datenstrom übermittelt. Als letztes werden die Informationen von den Monitor-Anschlüsse ausgelesen. Je nach Voreinstellung über die Slow-Control geben Sie Aufschluß über:

- die Länge der voreingestellten Latenz in Taktzyklen
- den Füllstand des Derandomizing Buffers
- die erfolgreiche oder fehlgeschlagene Absolvierung des Memory-Selftest
- die Länge eines OTIS-Datensatzes
- den korrekten Arbeitspunkt der DLL

Einige der Informationen werden zusätzlich im Datenkopf des OTIS übertragen und vom FPGA extrahiert und ausgewertet. Lediglich die Latenz ist nur über die Monitor-Anschlüsse erfassbar, weshalb beim späteren Testen des Chips genau diese Information direkt abgegriffen wird.

Kapitel 3

Entwurf der FPGA-Schaltung

3.1 FPGA Grundlagen

Ein FPGA (Field Programmable Gate Array) ist ein frei konfigurierbarer Logikschaltkreis. Er ist besonders geeignet für die Aufgabenstellung in dieser Diplomarbeit, da im Unterschied zu Mikroprozessoren nicht nur eine sequentielle Abarbeitung von Algorithmen möglich ist (Computing-in-Time), sondern Informationen massiv parallel verarbeitet werden können (Computing-In-Space). In dieser Diplomarbeit ist die Schaltung auf dem *StratixTM EP1S25*-FPGA der Firma AlteraTM realisiert.

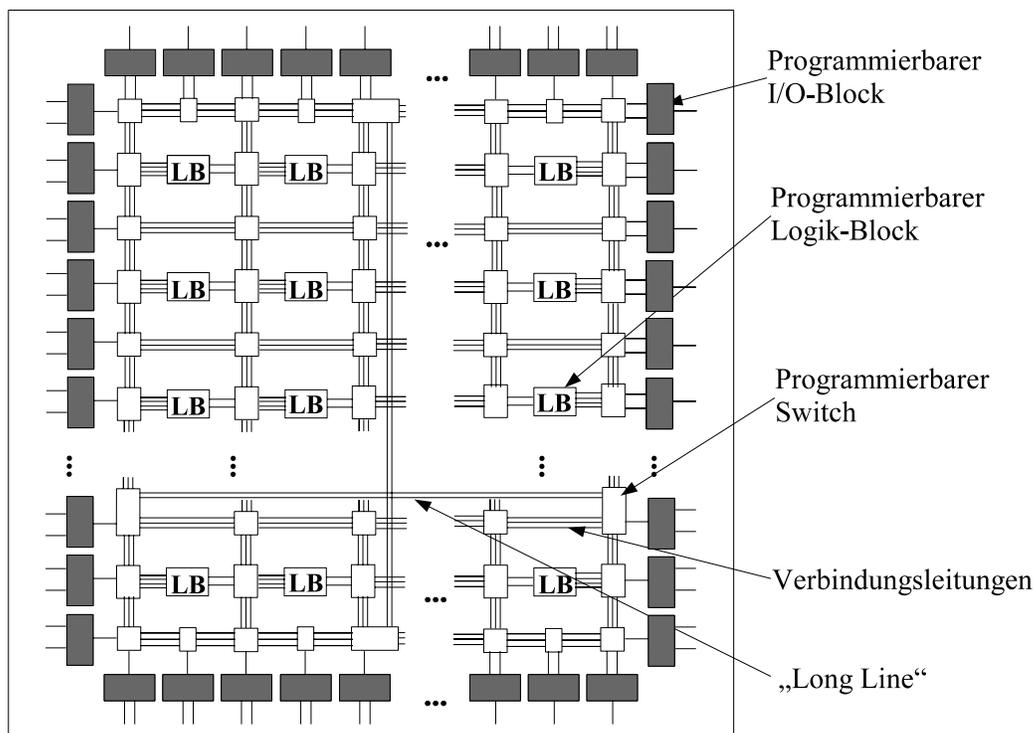


Abbildung 3.1: Schematischer Aufbau des FPGA

Der FPGA besteht zum größten Teil aus vielen identischen konfigurierbaren Logikblöcken, die in einer Matrixstruktur angeordnet sind. Diese können je nach Anforderung an die zu realisierende Schaltung untereinander verdrahtet werden. An den äusseren Seiten sind die I/O Blöcke platziert, über die Signale vom FPGA an externe Empfänger weitergegeben bzw. von externen Quellen empfangen werden können. Die Verdrahtungen werden mittels programmierbarer Verbindungspunkte, den sogenannten Switches, konfiguriert. Da jeder dieser Switches eine Signalverzögerung hervorruft, sind zusätzlich „Long Lines“ eingefügt, um große Distanzen ohne dazwischengeschaltete Switches überbrücken zu können. Ein Logikblock ist wiederum aus identischen logischen Zellen aufgebaut¹. Neben diesen Grundelementen befinden sich auf dem Stratix-FPGA weitere Bauteile, die hier nicht eingezeichnet sind, wie z.B. PLLs (Phase Locked Loops), sowie verschiedene RAM- (Random Acces Memory) und DSP-Blöcke (Digital Signal Processing).

Der Stratix-FPGA verfügt über 25500 logische Zellen, von denen in dieser Arbeit etwa 17800 (70 %) verwendet wurden. In diesen logischen Zellen werden boolesche Funktionen umgesetzt, d.h. jede logische Zelle kombiniert die Eingangssignale zu einer Ausgangsfunktion. Daher werden sie häufig als Funktionsgeneratoren bezeichnet. Dies steht im Kontrast zur Realisierung der booleschen Funktion durch die entsprechende Anordnung von Logikgattern. Es existieren zwei Architekturformen, aus denen die booleschen Funktionen generiert werden können: die LUT-Architektur (Look-Up-Table) und die MUX-Architektur (Multiplexer).

Erstere basiert auf der Verwendung von statischem RAM Speicher, in den sämtliche Kombinationen von Eingangssignalen und den dazugehörigen Ausgangssignalen abgelegt sind.

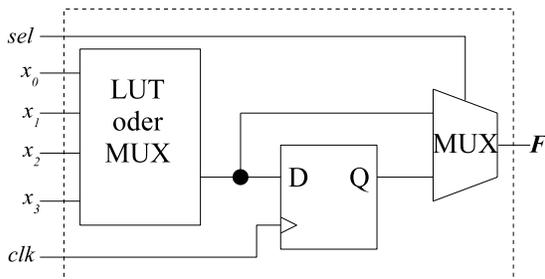


Abbildung 3.2: Die Logik-Zelle als Basis eines FPGA Logikblocks

Man benötigt demnach für eine Funktion mit n Variablen 2^n Speicherzellen, in die während der Konfigurationsphase des FPGA die Funktionswerte hineingeschrieben werden.

Bei der MUX-Architektur werden die Eingangssignale x_0 bis x_3 an den Selektions-eingang eines Multiplexers angeschlossen. Ändern sich diese Eingangssignale, so wird ein neuer Dateneingang des Multiplexers selektiert und durchgeschaltet. Je nachdem welche Bitkombinationen an den Eingängen des Multiplexers vorliegen, können verschiedene boolesche Funktionen realisiert werden. Man spricht in diesem Fall von 2^n -Input Multiplexern, die jede beliebige Funktion mit n Variablen durch Anlegen der 2^n Dateneingangssignale erzeugen können.

Um die Möglichkeit zu haben, die Ausgangssignale der Funktionsgeneratoren auf eine bestimmte Taktfrequenz zu synchronisieren, wird ein D-Flipflop-Register in Form eines

¹Beim Stratix-FPGA besteht ein Logikblock aus zehn solcher Zellen.

D-Latches zur Verfügung gestellt. Damit man zwischen kombinatorischer und taktsynchroner Logik hin- und herschalten kann, ist am Ausgang der Logikzelle ein Multiplexer zwischengeschaltet.

Den grössten Teil der Chipfläche (etwa 90 %) belegen die Verdrahtungsressourcen. Der längstmögliche Weg, den ein Signal während einer Taktperiode nehmen kann (kritischer Pfad), legt die maximale Arbeitsfrequenz der Schaltung fest. Zur Berechnung des kritischen Pfades werden die aufaddierten Laufzeiten durch die Logikgatter und durch die Verdrahtungen berücksichtigt. Eine hohe Auslastung der FPGA-Ressourcen kann daher die Einhaltung einer vorgegebenen Taktfrequenz schwierig machen.

Der in dieser Arbeit verwendete FPGA befindet sich auf einer PCI-Steckkarte, auf der neben dem Stratix-Chip EP1S25 unter anderem noch diverse Anschlüsse aufgebracht sind. Weiterhin sind externe DDR SDRAM Bausteine auf dieser Karte platziert. Die Karte wurde als „PCI-Development-Kit, Stratix Edition“ [18] erworben und ist auf dem Foto in Abbildung 3.3 dargestellt.

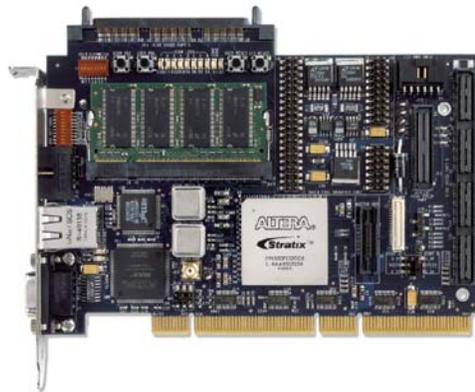


Abbildung 3.3: Die Stratix-Karte

3.1.1 EDA-Entwurfswerkzeuge und Entwurfsablauf

In diesem Abschnitt werden die zum Entwurf und Test verwendeten Werkzeuge vorgestellt. Kern des Entwurfsprozess ist die Beschreibung des Schaltplans auf einer höheren Abstraktionsebene, die RTL (Register Transfer Level) genannt wird. Als Entwicklungsumgebung wurde der *HDL-Designer 2005.1* verwendet. Dieser erlaubt die graphische Eingabe der Schaltung und ermöglicht so die Kombination von selbstdefinierten Bausteinen mit Gattern aus einer Bibliothek zu komplexen Schaltungen. Der erstellte VHDL-Code (Very High Speed Hardware Description Language) wird anschließend mit Hilfe von *ModelSim SE 6.0d* simuliert. Eine Simulation erfordert eine zuvor erstellte Testumgebung, welche die für die Schaltung notwendigen Stimuli bereitstellt. So ist es möglich, das Verhalten der Schaltung und die resultierenden Signale zu analysieren.

Das Programm *Precision Synthesis 2005a.46* erstellt im nächsten Schritt aus der RTL-Beschreibung der Schaltung eine VHDL-Gatternetzliste, die anschließend weiterverar-

beitet werden kann. Dieser Schritt wird als erste Synthese des VHDL-Codes bezeichnet. In diesem Schritt findet eine Übersetzung der Teile des Codes, die auf Verhaltensbeschreibungen der Schaltung basieren (z.B. Zustandsautomaten) in eine Anordnung von logischen Bauelementen, statt.

Alle verwendeten Programme sind Teile aus dem Programmpaket von Mentor GraphicsTM.

Die Benutzerumgebung zur Umsetzung der Gatternetzliste in eine physische Schaltung wird von dem jeweiligen FPGA-Hersteller vorgegeben, in diesem Fall handelt es sich um *Quartus II 4.2* von Altera. Dieses Werkzeug übernimmt die logische Synthese, das „Place and Route“ und die Laufzeitanalyse. Mit der logischen Synthese werden verschiedene Algorithmen durchlaufen zur Minimierung der Anzahl der Gates und zur Beseitigung redundanter Logik und um die FPGA-Architektur so effizient wie möglich auszunutzen. „Place and Route“ bedeutet die Platzierung der einzelnen Gatter auf der zur Verfügung stehenden Chipfläche und deren Verknüpfung untereinander. Die Laufzeitanalyse aller auf dem Chip verwendeten Signale wird von Quartus vorgenommen, um potentielle Signal-Laufzeitfehler aufzudecken. Waren alle Schritte erfolgreich, kann man den FPGA „programmieren“, d.h. die Schaltung auf dem FPGA implementieren.

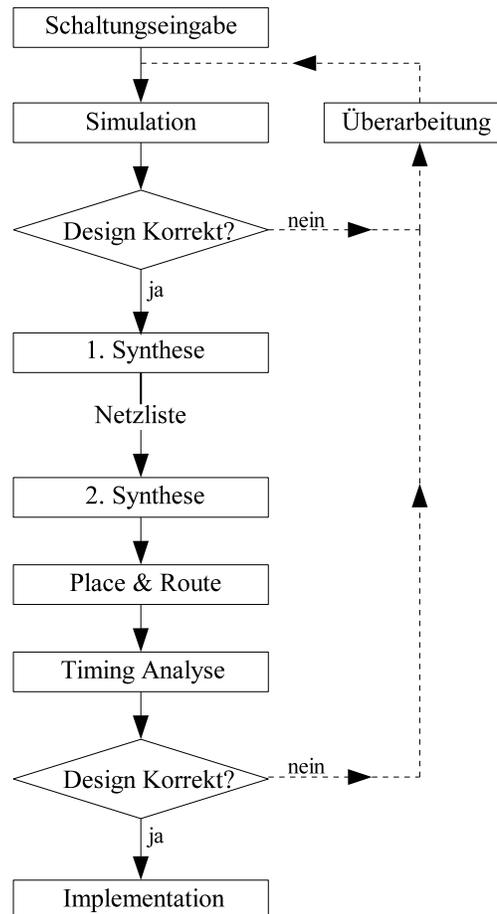


Abbildung 3.4: Der Entwurfsablauf

3.2 Überblick über die Testprozedur

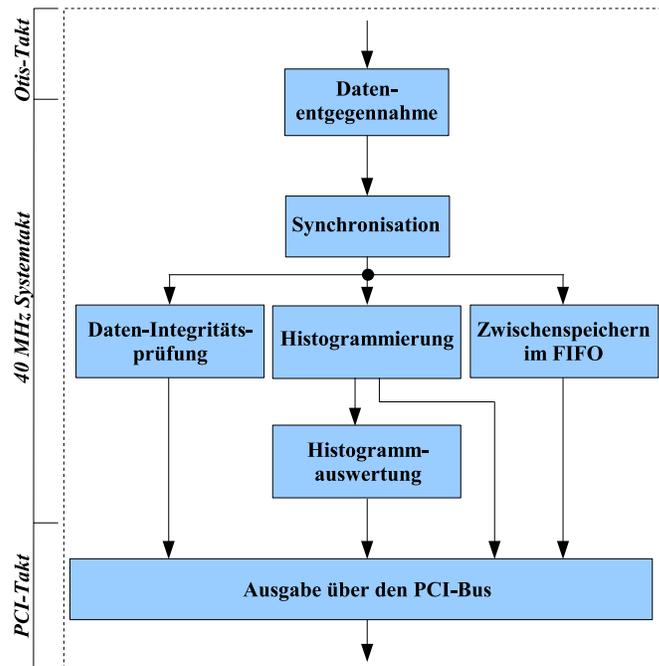


Abbildung 3.5: Datenfluss durch das FPGA Design

In diesem Teil wird der Datenfluß durch das Design vorgestellt. Allen Tests gemeinsam ist der Empfang und die Synchronisation der Daten (Abbildung 3.5). Danach kann der gesamte Datenfluß gleichzeitig an drei Zweige der Testprozedur weitergeleitet werden.

Im ersten Zweig wird der Datenstrom dazu verwendet, die Integrität der OTIS-Statusbits aus dem Datenkopf zu überprüfen. Als Ergebnis erhält man Zählwerte der Errorbitzähler für die verschiedenen OTIS-Funktionstests. Sie sagen aus, wie oft innerhalb einer Meßreihe die zu untersuchende Funktionen des OTIS fehlerhaft waren. Zusätzlich gibt dieser Teil des Designs Informationen darüber aus, in welchem aktuellen Status sich der OTIS gerade befindet.

Im zweiten Zweig des Datenflußdiagramms werden folgende Werte histogrammiert:

- die Driftzeiten für einen Kanal zur Berechnung der DNL
- die Anzahl der Treffer pro Kanal zum Testen der Kanalmaske
- die Zahl der Treffer für jede vorkommende Event-ID im OTIS-Datenkopf

Die drei erstellten Histogramme werden zur Auswertung vom FPGA an den PC übergeben. Die Daten aus den Driftzeithistogrammen gelangen gleichzeitig in den Block zur Histogrammauswertung. Hier wird die zum einen die differentielle Nichtlinearität berechnet und zum anderen die Maxima und die Minima aus den Histogrammen bestimmt.

Im dritten Zweig auf der rechten Seite findet die direkte Auslese der auf dem FPGA synchronisierten OTIS-Datensätze statt. Sie werden über ein FIFO über die PCI-Schnittstelle vom FPGA auf den PC übertragen. So ist es möglich, die OTIS-Datensätze in Echtzeit auf dem Bildschirm zu betrachten oder auf dem PC abzuspeichern.

Die Ergebnisse aller Teiltests werden an den PC weitergeleitet. Ein nachgeschaltetes Programm auf dem Rechner zur Auswertung der Informationen entscheidet, ob ein Chip für den Detektorbetrieb geeignet ist.

Innerhalb des Designs werden drei verschiedene Taktfrequenzen verwendet:

PCI-Takt: Sie wird vom PCI-Bus gespeist und kann sowohl mit 33 MHz als auch mit 66 MHz betrieben werden. Sie wird innerhalb des PCI-Megacores und des PCI-Interfaces verwendet.

40 MHz Systemtakt: Der Hauptteil der Schaltung wird genau wie der OTIS von einem 40 MHz Systemtakt betrieben. Er wird von einem 100 MHz Oszillator auf der Stratix-Karte erzeugt. Über eine PLL (Phase-Locked-Loop) auf dem FPGA wird er anschließend heruntergetaktet und auf die einzelnen Systemblöcke im FPGA Design verteilt. Zusätzlich erhält der OTIS-Chip seine Taktfrequenzfrequenz von der PLL.

OTIS-Takt: Der OTIS bezieht seine Taktfrequenz vom FPGA. Die vom OTIS ausgegebenen Daten kommen jedoch nicht synchron zum 40 MHz Takt am FPGA an, da die Datenübertragung vom OTIS zum Rechner über längere Kabellaufwege verläuft. An dieser Stelle ist es nützlich, das Last Dummy Out (LDO) des OTIS als Arbeitsfrequenz für die Datenengengnahme auf dem FPGA zu verwenden. Beim LDO handelt es sich um einen Ausgang am OTIS, der die Arbeitsfrequenz der DLL-Kette widerspiegelt. Dieser Takt ist immer synchron zu den ausgegebenen OTIS-Datensätzen. Damit läßt sich die Phasenverschiebung zwischen dem Datenstrom und dem 40 MHz Systemtakt des FPGA umgehen.

Die Steuereinheit auf dem FPGA

Unabhängig vom oben erwähnten Teil des Datenverlaufs ist eine zweite Einheit innerhalb des Schaltungsentwurfs zur Steuerung von FPGA-Design und OTIS realisiert. Das Schema wird in Abbildung 3.6 aufgezeigt. Hier werden benutzergesteuerte Befehle über die PCI-Schnittstelle des FPGA entgegengenommen und an die ausführenden Unter-einheiten im FPGA-Design weitergeleitet. Weiterhin wird die „Fast-Control“ von dieser Einheit gesteuert und verschiedene Befehle direkt an den OTIS und an das Nadelkarten-Adapterboard gesendet. Schließlich übernimmt diese Einheit die Verteilung der Pulser-signale.

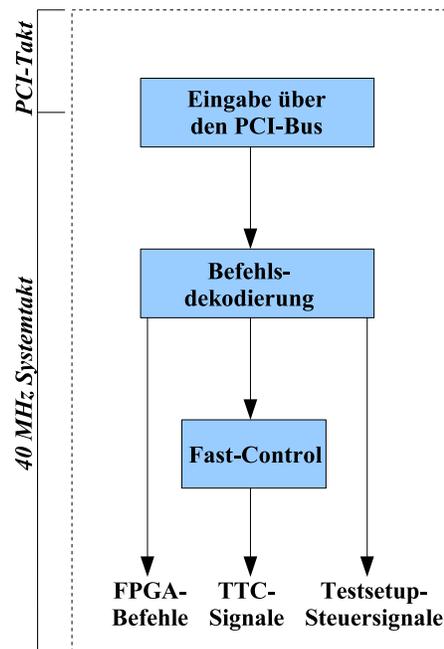


Abbildung 3.6: Steuereinheit des FPGA

Überblick über das Toplevel

In Abbildung 3.7 ist ein schematischer Überblick von der höchsten Ebene (Toplevel) der Schaltung dargestellt. Hier sind die im Design implementierten funktionalen Einheiten, die wiederum darunterliegende Ebenen enthalten, zu größeren zusammenhängenden Blöcken zusammengefasst. Gelb gekennzeichnet sind die Busse, die für das Setzen von Registern verantwortlich sind und mit deren Hilfe man Einstellungen am Design vornehmen kann. Ebenso ist der Bus gelb markiert, der die vom Design fertig prozessierten Daten wieder über den PCI-Bus an den PC zurückgibt. Der mit rot gekennzeichnete Bus überträgt die vom Benutzer erteilten Befehle auf die einzelnen Blöcke. Alle Signale zwischen PCI-Interface und dem restlichen Design werden durch den PCI-Busverteiler zugeordnet. Insgesamt sind acht verschiedene Blöcke erkennbar. Die eigentlichen Tests finden in den Blöcken der Datenintegritätsprüfung, der Histogrammierung und der Daten-Prozessierung statt. Der mit OTIS-Controller bezeichnete Block ersetzt das TTC-System für den OTIS aus dem bisherigen lokalen Aufbau am Physikalischen Institut. Oben links in der Übersicht treffen die Signale des OTIS im Design ein. Weiterhin sind die Ausgabesignale des FPGA dargestellt. Dazu gehören die Steuersignale für den OTIS, sowie für die Nadelkarte (z.B. zum Einschalten der Betriebsspannung oder zur Erzeugung bestimmter Bitmuster in der OTIS-Identifikationsnummer). Ausserdem wird das Signal des Pulsers so verteilt, daß entweder alle ungeraden oder alle geraden Kanäle des OTIS mit Signalen versorgt werden.

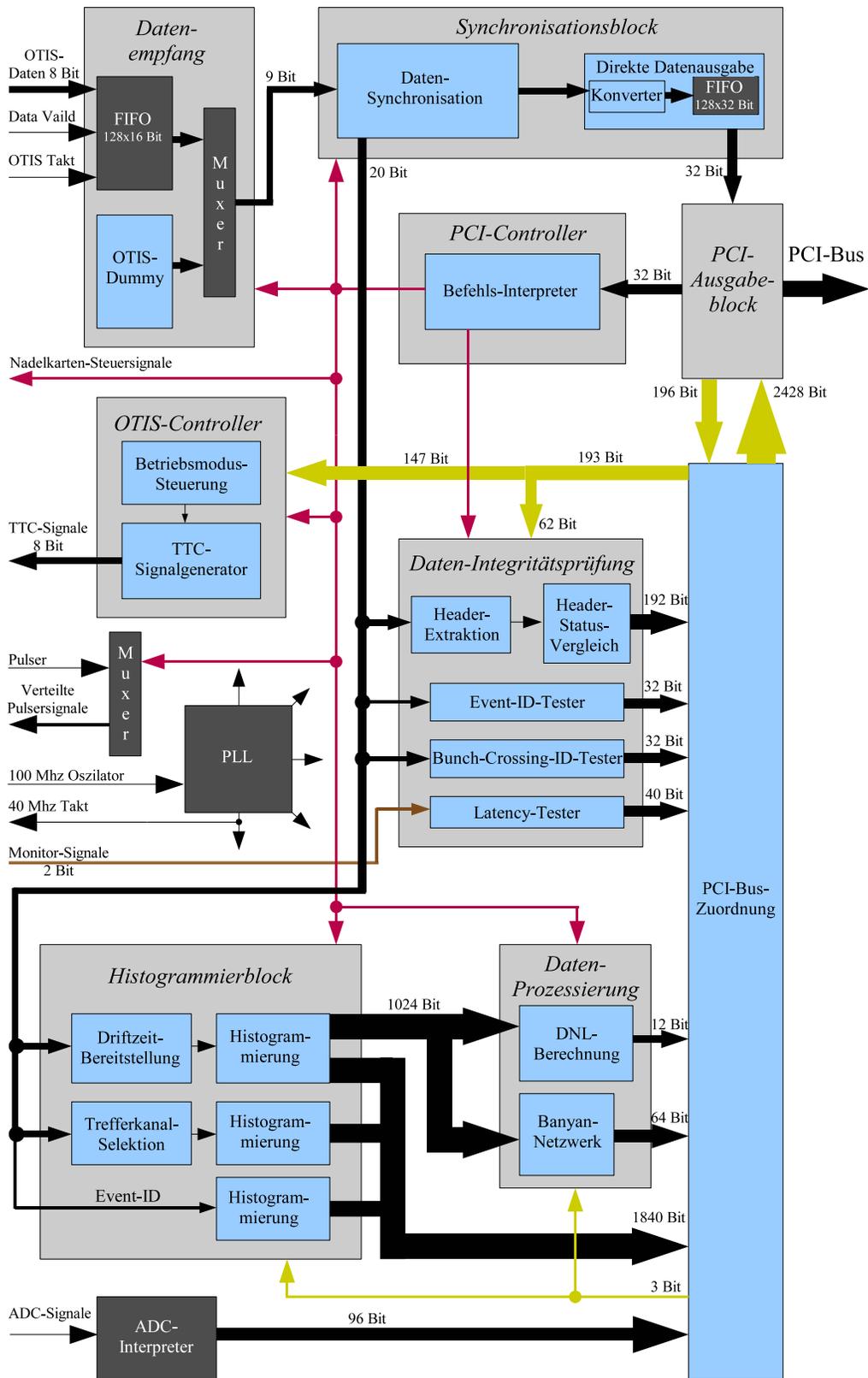


Abbildung 3.7: Schema vom Toplevel des Schaltungsentwurfs

3.2.1 Der PCI-Block und Datensynchronisation

Datenentgegennahme

Zunächst werden die Daten wie in Abbildung 3.7 zu sehen ist vom FPGA entgegengenommen und in ein sogenanntes „Dual Ported FIFO“ geschrieben. Hier wird ein Wechsel vom OTIS-Takt auf den internen 40 MHz Systemtakt vorgenommen. Wahlweise ist es möglich die OTIS-Daten zur Simulation vom OTIS-Dummy erzeugen zu lassen. Der OTIS-Dummy simuliert das Verhalten des tatsächlichen OTIS-Chips, indem er einen 36 Byte langen Datenstrom mit Datenkopf und einstellbaren Driftzeiten generiert. Diese Option hat sich bei der Fehlersuche während der Implementierung des Designs als sehr nützlich erwiesen. Seine Verwendung erlaubt innerhalb der Schaltungssimulation beispielsweise eine Voraussage der zu erwartenden Einträge in den Driftzeithistogrammen und der daraus berechneten DNL im Daten-Prozessierungsblock.

Datensynchronisation

Nach der Datenentgegennahme erfolgt die Datensynchronisation, d.h. aus dem kontinuierlichen Datenstrom werden die Driftzeiten der 32 Kanäle des OTIS mitsamt des Datenkopfes selektiert. Bedingung für die Weiterleitung der Daten ist, daß die Positions-ID des OTIS mit dem im Konfigurationsregister voreingestellten Wert übereinstimmt. Ist dies nicht der Fall, wird der Datenstrom abgelehnt. Des weiteren wird das „Data-Valid“-Signal für die Synchronisation benutzt. Das Signal stammt direkt vom OTIS und markiert einen gültigen Datensatz. Erwartet wird, daß es für die Länge eines Datensatzes aktiv ist. Sind alle Bedingungen erfüllt, wird der validierte Datensatz im Synchronisationsblock mit dem „Event-Frame“-Signal markiert und weitergeleitet. Für das Erleichtern der Fehlersuche beim Entwurf des Designs ist ausserdem ein zweiter Betriebsmodus vorhanden, der alle Daten aus dem Datenstrom weiterleitet, die nicht den Wert null haben. Er wird als „Non-Zero-Mode“ bezeichnet. Auch diese Funktion hat sich bei der Schaltungsanalyse als sehr hilfreich erwiesen.

Direkte Datenausgabe an den PC

Nach der Synchronisation gibt es zwei Möglichkeiten für den weiteren Datenverlauf. Zum einen besteht die Option der Direkten Datenausgabe an den Endnutzer. Dazu durchlaufen die Daten als erstes den nachgeschalteten Konverter, der das Datenformat von 36×8 Bit in 9×32 Bit umwandelt. Anschließend werden die Daten im Synchronisations-FIFO zwischengespeichert und an das PCI-Interface weitergegeben, um einen Taktwechsel auf den 33 MHz-Takt des PCI-Busses zu vollziehen.

Datentransfer zu den Testeinheiten

Die zweite Möglichkeit des Datenverlaufs besteht darin, die synchronisierten Daten an das restliche Design weiterzuleiten, um sie analysieren zu können. Die Arbeitsweise der nachfolgenden Blöcke wird im nächsten Abschnitt erläutert. Wenn alle Analysevorgänge

der OTIS-Daten auf dem FPGA abgeschlossen sind, tritt der PCI-Ausgabeblock in Erscheinung.

Der PCI-Ausgabeblock

Der PCI-Block besteht aus dem PCI-Interface und dem PCI-Megacore wurde zusammen mit dem Synchronisationsblock aus [13] in leicht abgeänderter Version übernommen. Die beiden Komponenten des PCI-Blocks, das PCI-Interface und das PCI-Megacore sorgen für die Kommunikation zwischen FPGA und PCI-Bus. Beim PCI-Megacore handelt es sich um eine „Intellectual Property“ (IP), die als fertige logische Einheit von Altera gegen eine Lizenzgebühr bereitgestellt wird. Auf die Funktionsweise des sehr komplexen PCI-Teils soll hier nicht weiter eingegangen werden. Zu erwähnen bleibt lediglich, daß vom PCI-Block unter anderem ein Adressraum von 64×64 Bit, also 4096 Bit zur Verfügung gestellt wird. Innerhalb dieses Adressraums können nun Daten abgespeichert werden, die entweder als Rückgabewerte vom Design vorliegen oder die vom Benutzer vorgegebenen Einstellungen für das Design beinhalten. Sie werden im folgenden mit PCI-Register bzw. Konfigurationsregister bezeichnet.

3.2.2 Der PCI-Controller

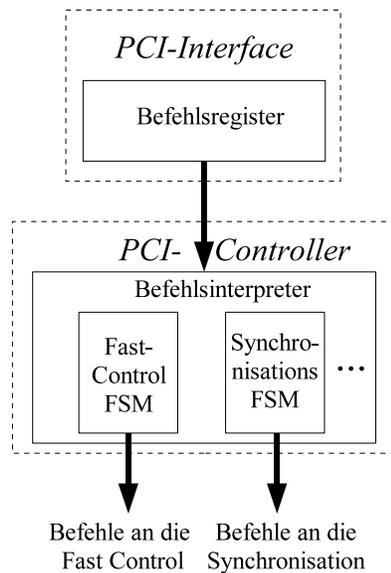


Abbildung 3.8: Der PCI-Controller

Der PCI-Controller hat die Aufgabe, die von einem Benutzer gegebenen Befehle zu empfangen und zu übersetzen. Für die Entgegennahme der Befehle befindet sich im Konfigurationsregister des PCI-Teils auf Adresse 1023 ein 32 Bit breiter reservierter Bereich. Ein vom Anwender abgegebener Befehl wird als Dezimalwert über das Befehlsregister übertragen und an den Befehlsinterpretierer weitergeleitet. Im Befehlsinterpretierer befinden sich im wesentlichen Zustandsautomaten, die je nach Befehl verschiedene Steuersignale

aktivieren. Es lassen sich einerseits die verschiedenen Untereinheiten im FPGA steuern, andererseits Befehle direkt auf den OTIS bzw. auf die Nadelkarten-Adapterkarte geben. In Tabelle 3.1 sind alle vorhandenen Befehle aufgelistet.

Die Befehle sind den verschiedenen Blöcken zugeordnet. Die Befehle an die „Fast Control“-Einheit dienen dazu, verschiedene Resets und Triggersignale an den OTIS zu schicken. Den Block zur Datenentgegennahme kann man mit den Befehlen 35 und 36 zwischen Dummy-Modus und OTIS-Daten umschalten lassen. Am Synchronisationsblock läßt sich für Testzwecke der Non-Zero-Modus an bzw. ausschalten. Da sich die Datenströme für die unterschiedlichen OTIS-Versionen leicht unterscheiden, besteht ausserdem die Notwendigkeit, dies dem Synchronisationsblock mitzuteilen.

Einige Befehle richten sich an alle Untereinheiten der Schaltung. Dazu gehört zum einen die Anweisung das gesamte FPGA Design mittels „System-Reset“ zurückzusetzen und zum anderen der Start -bzw. Stoppbefehl für die Messungen. Beim Start einer Messung wird die Trigger-State Machine der Fast-Control-Einheit aktiviert.

Abbildung 3.9 zeigt als Beispiel den Zustandsautomaten des Befehlsinterpreters, der für die Wahl des Betriebsmodus der Fast Control auf dem FPGA verantwortlich ist. Die Hexadezimahlwerte x“2D“ bis x“30“ entsprechen den Befehlen 45 bis 48 in Tabelle 3.1. Sobald das Befehlsregister einen neuen Wert an diese Zustandsmaschine überträgt, wechselt sie in den gewünschten Zustand. In den vier Zuständen wird jeweils der binäre Wert eines 2 Bit-Busses definiert. Dieser ist an die Fast-Control-Einheit im FPGA angeschlossen, die je nach Kombination der aktiven Bits dieses Busses in einen der vier möglichen Betriebsmodi wechselt.

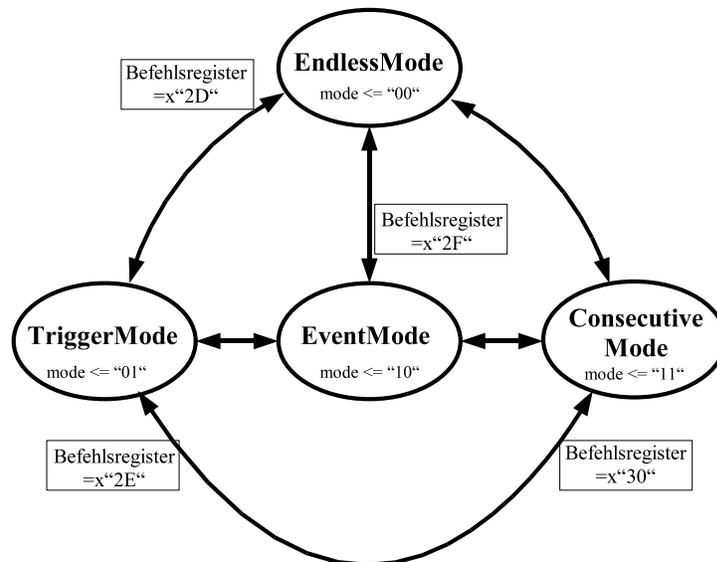


Abbildung 3.9: Schema eines Zustandsautomaten des Befehlsdekoders

| Befehlsnr. | Befehl | Beschreibung |
|------------|----------------------------|--|
| 20 | PowerUp-Reset | Befehle an die „Fast Control“-Einheit |
| 22 | Level0-Reset | |
| 23 | Bunch-Counter-Reset | |
| 24 | Event-Counter-Reset | |
| 25 | All Resets | |
| 27 | DLL-Reset | |
| 45 | Endless Mode | |
| 46 | Trigger Mode | |
| 47 | Event Mode | |
| 48 | Consecutive Mode | |
| 35 | Enable Dummy Mode | Befehle für die Datenentgegennahme |
| 36 | Disable Dummy Mode | |
| 26 | Clear FIFO | Befehle an den Synchronisationsblock |
| 37 | Enable Non-Zero-Mode | |
| 38 | Disable Nono-Zero-Mode | |
| 41 | OTIS-Version 1.0 | |
| 42 | OTIS-Version 1.1 or higher | |
| 31 | Histogramm one channel | Befehle an den Histogrammierblock |
| 32 | Histogramm all channels | |
| 52 | Set Hitmask Even | Befehle an die Steuereinheit des Pulsers |
| 53 | Set Hitmask Odd | |
| 60 | Enable Pulser | |
| 61 | Disable Pulser | |
| 21 | System-Reset | Befehle an alle Blöcke des Designs |
| 29 | Start Measurement | |
| 30 | Stop Measurement | |
| 54 | Enable Bank-Select | Befehle an den OTIS |
| 55 | Disable Bank-Select | |
| 56 | Enable EDC | |
| 57 | Disable EDC | |
| 50 | Position-ID Even | Befehle an die Nadelkarte |
| 51 | Position-ID Odd | |
| 58 | Enable LVDS | |
| 59 | Disable LVDS | |
| 62 | Shut Down Power Supply | |
| 63 | Turn On Power Supply | |

Tabelle 3.1: Das Befehlsregister

3.2.3 Der OTIS-Controller

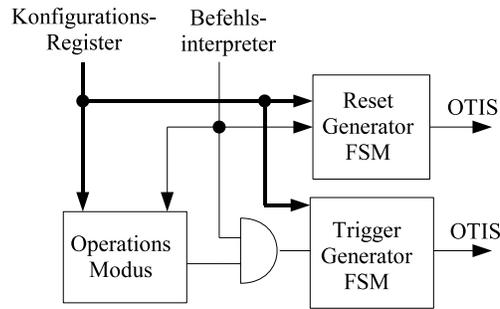


Abbildung 3.10: Der OTIS-Controller

Reset-Signale

In diesem Teil werden Steuersignale für den OTIS erzeugt. Hierzu gehören die Resetsignale. Das PowerUp-Reset, das beim OTIS-Serientest zu Beginn bei jedem Chip gesetzt wird, dient der Initialisierung aller Konfigurationsregister des OTIS. Um dies zu erreichen, muss das PowerUp-Reset-Pad des OTIS für mindestens $100\ \mu\text{s}$ auf Masse heruntergezogen werden. Einmal gesetzt ist die I²C-Schnittstelle des OTIS nun bereit, Daten zu empfangen. Ohne Neuprogrammierung über I²C nach einem solchen Reset stellt der Chip nur noch ein Minimum an Funktionen zur Verfügung. Es müssen daher für einen sinnvollen Test alle Register neu beschrieben werden.

Nach der Neuprogrammierung der Register folgt üblicherweise das ebenfalls in dieser Einheit bereitgestellte Level0-Reset. Es sorgt dafür, daß die programmierten Parameter wie Auslesemodus oder Latenz übernommen werden und setzt gleichzeitig die Lese- bzw. Schreibzeiger des Pipeline Registers und des Derandomizing Buffers zurück. Schließlich werden zur Vorbereitung auf die Datennahme alle State Machines des Chips in einen definierten Zustand gebracht.

Weitere Reset-Signale sind für den Event-Zähler sowie für den BunchCrossing-Zähler des OTIS vorgesehen. Als letztes gibt es das DLL-Reset. Es ist ausschließlich für Testzwecke des OTIS im Labor geeignet und findet im LHCb Experiment keine Verwendung. Der Grund warum das DLL-Reset-Signal gesondert behandelt wird ist, daß nach einem solchen Reset die DLL $1\ \mu\text{s}$ benötigt, um wieder den Arbeitspunkt zu erreichen, in dem erst wieder gültige Daten bereitgestellt werden können. Gesetzt werden sollte es, wenn der Arbeitsbereich der DLL aus den gültigen Grenzen herausgelaufen ist, was auch im Fehlerbit des OTIS-Datenheaders angezeigt wird.

Der Befehlsinterpreter liefert dem Zustandsautomaten für die Erzeugung der Resetsignale in Abbildung 3.10 die Anweisungen für die auszuführenden Operationen. Mit Hilfe des Konfigurationsregisters wird dem Resetgenerator mitgeteilt, wie viele Taktzyklen ein gesetztes Resetsignal aktiv bleiben soll. Es ist möglich jeden Resetsignal einzeln zu schicken oder alle Signale in einer bestimmten Abfolge hintereinander auszugeben.

Triggerezeugung

Der OTIS-Controller-Block erzeugt weiterhin die Triggersignale, die den OTIS dazu veranlassen, einen gültigen Datensatz anzunehmen. Die Triggerrate ist einstellbar und wird über die Konfigurationsregister an diesen Block weitergegeben. Der Wert für die Triggerrate beträgt für den Serientest 1 MHz. Der Trigger wird synchron zum 40 MHz-Systemtakt erzeugt. Der Zustandsautomat ist in Abbildung 3.11 schemenhaft dargestellt. Die Pfeile haben die Bedeutung einer Signalzuweisung. Der Wartezustand wird verlassen, sobald das „trigger enable“-Signal aktiv wird. Dieses Signal ist gekoppelt an den Befehl „Start Measurement“. Im Start-Zustand wird das 32 Bit-Signal „count“ auf den Wert für die Triggerrate gesetzt, die vorher vom Anwender ins Konfigurationsregister geschrieben wurde. Anschließend wird der Wert im nächsten Zustand solange runtergezählt, bis er null ist. Zu diesem Zeitpunkt ist die Bedingung erfüllt, ein Triggersignal der Länge von 25 ns zu erzeugen. Von hier aus findet im nächsten Taktzyklus ein Wechsel in den Start-Zustand statt. Die „Idle“-Position wird dann angenommen, wenn ein Design-Reset oder ein Abbruch der Messung erfolgt ist.

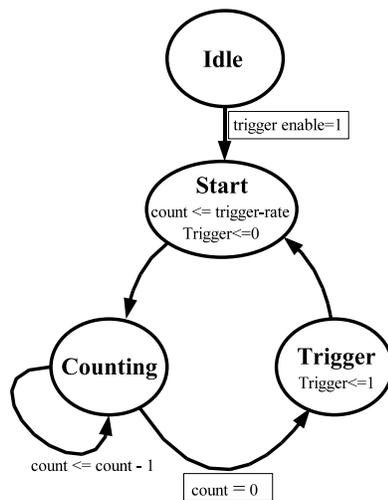


Abbildung 3.11: Zustandsautomat zur Triggeregenerierung

Wahl des Betriebsmodus

Die Wahl des Betriebsmodus für die Histogramm-Messungen ist die dritte wichtige Funktion des OTIS-Controllers. Es wird zwischen vier verschiedenen Modi unterschieden. Der mit „Endless Mode“ bezeichnete Befehl lässt die „Fast Control“ solange Trigger an den OTIS schicken, bis die Messung mit dem Befehl „Stop Measurement“ wieder abgebrochen wird. Will man der Fast-Control eine Triggerzahl vorgeben, die für eine automatische Beendigung der Messung zu erreichen ist, muß der „Trigger Mode“-Befehl erteilt werden. Soll die Bedingung für einen automatischen Abbruch der Messung nicht die Zahl der Trigger, sondern die Zahl der zurückgelesenen Datensätze sein, steht der Befehl „Event Mode“ zur Verfügung. Um aufeinanderfolgende Trigger für den „Buffer Overflow“-Test zu generieren, erteilt man den in Tabelle 3.1 als letzten aufgeführten Befehl „Consecutive

Mode“ an den FPGA.

Über das Konfigurationsregister bekommt die Einheit zur Steuerung des TTC-Betriebsmodus die Werte für eine Abbruchbedingung einer Messung mitgeteilt. Der nachfolgende Triggeregenerator arbeitet nur dann, wenn zwei Bedingungen des Und-Gliedes in Abbildung 3.10 erfüllt sind. Erstens muß der Befehl für den Start einer Messung gesetzt sein und zweitens darf die Abbruchbedingung des jeweiligen Operationsmodus noch nicht erreicht sein.

Sämtliche Signale die von der Fast Control an den OTIS geschickt werden sind LVDS-Signale (Low Voltage Differential Signals). Dieser Standard garantiert eine hohe Störsicherheit bei der Signalübertragung. Im Unterschied zu gewöhnlichen „single ended“-Lösungen treten hier keine Probleme bezüglich der Potentialdifferenzen mehr auf.

3.2.4 Die Datenintegritätsprüfung

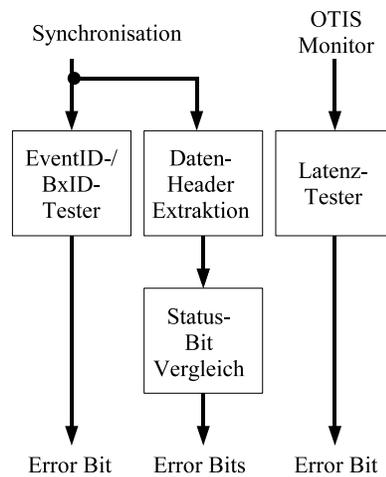


Abbildung 3.12: Die Daten-Integritätsprüfung

Unter dem Namen Datenintegritätsprüfung sind diejenigen Einheiten zusammengefaßt, die ihre Informaionen aus dem OTIS-Datenkopf bzw. aus den Monitorsignalen gewinnen. Damit wird der Bereich der digitalen Funktionalität des OTIS abgedeckt. Die Einheit gliedert sich in drei Komponenten. Die erste Komponente besteht aus einem Zustandsautomaten für die Extraktion der Daten aus dem OTIS-Datenkopf. An sie schließt sich der Block zur Auswertung der so gewonnenen Information an. Auf diese Weise ist es möglich, zu überprüfen, ob am OTIS vorgenommene Einstellung über die Slow control richtig übernommen werden können. Zusätzlich werden die Ergebnisse der internen OTIS-Testroutinen analysiert. Dazu werden die vom OTIS empfangene Bits mit Vorgabewerten verglichen. Auftretende Fehler der Vergleiche werden gezählt und ins PCI-Register geschrieben. Damit man den momentanen Status des OTIS während des Betriebs mitverfolgen kann, werden gleichzeitig die erhaltenen Informationen aus dem OTIS-Datenkopf an den Anwender weitergeleitet.

Der Latency-Tester stellt die zweite Komponente dieser Einheit dar. Hier werden die eingehenden Monitorsignale des OTIS ausgewertet. In der dritten Komponente werden die Event- und Bunch-Crossing Zähler des OTIS auf ihre korrekte Arbeitsweise hin über-

prüft.

Die OTIS-Statusbit-Analyseeinheit

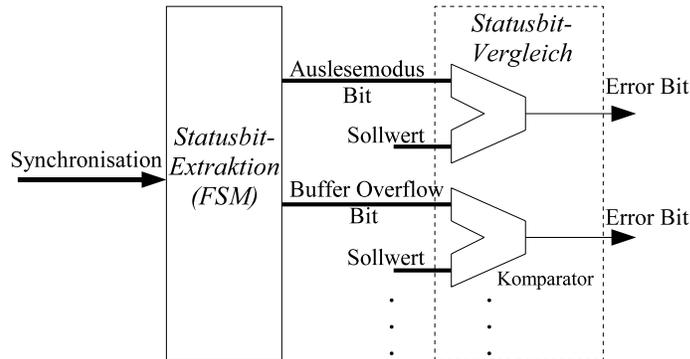


Abbildung 3.13: Schema der Statusbit-Analyseeinheit

Die für die Analyse der OTIS-Funktionen notwendigen Bits werden aus dem Datenkopf herausgesucht und im nachfolgenden Block analysiert. Bild 3.13 zeigt den prinzipiellen Aufbau dieser Einheit. Zuerst erreichen die 8 Bit-Daten aus der Synchronisation den Zustandsautomaten zur Extraktion der Header-Bits. Dort werden die einzelnen Statusbits aus dem Datenstrom herausgelesen. Diese Bitwerte aus dem Datenstrom, die Auskunft über den momentanen Status der einzelnen Funktionseinheiten des OTIS anzeigen, werden anschließend auf sechs verschiedene Signalleitungen separiert. In Abbildung 3.14 ist der Zustandsgraph für die Extraktion des Headers illustriert. Je nach OTIS-Version ist das Verlassen des Wartezustands an mehrere Bedingungen geknüpft. Ein Kriterium ist das Auffinden der OTIS-Identifikationsnummer im Datenstrom. Eine weitere Bedingung fordert ein aktives „Event-Frame“-Signal. Es zeigt einen von der Synchronisation als gültig eingestuftens Datensatz an. In den nächsten beiden Zuständen werden danach die gesuchten Bitwerte aus den Daten auf externe Signalleitungen geschrieben. Der Aufbau eines Datenkopfes ab Version 1.1 des OTIS ist in Tabelle 3.2 auf Seite 35 dargestellt. Man erhält damit die Statuswerte für folgende OTIS-Funktionen:

- Auslesemodus
- Zahl der Bunch Crossings
- Playback Modus
- Truncation Modus
- Error Bit
- Buffer Overflow Bit

Mit der Zahl der Bunch Crossings ist in diesem Fall die Länge des Meßintervalls des OTIS gemeint. Bit 18 ist das Oder aus dem Memory-Selftest-Bit, dem DLL-Lock-Lost Bit. Sollte einer dieser internen OTIS-Funktionstests des OTIS fehlschlagen, nimmt das Bit den Wert 1 an. Das Bit für den Truncation Modus gibt an, ob die OTIS-Datensätze nach 900 ns abgeschnitten werden sollen.

Header Data Format (OTIS1.2)

| Byte | 1 | | 2 | | 3 | | 4 | |
|---------|--------|-----|-----|--------|-----|----------|--------|-----|
| | MSB | LSB | MSB | LSB | MSB | LSB | MSB | LSB |
| Bit No. | 0..11 | | 12 | 13..17 | | 18,19 | 20..23 | |
| Data | TDC ID | | 1 | RO | Err | Event-ID | | BX |

| | | | | | |
|-----------|--------------|------------|----------|----------------------|-----------------|
| Read Mode | Number of BX | Truncation | Playback | Error (SFT, DLL SEU) | Buffer Overflow |
|-----------|--------------|------------|----------|----------------------|-----------------|

Tabelle 3.2: Der Datenkopf des OTIS 1.2

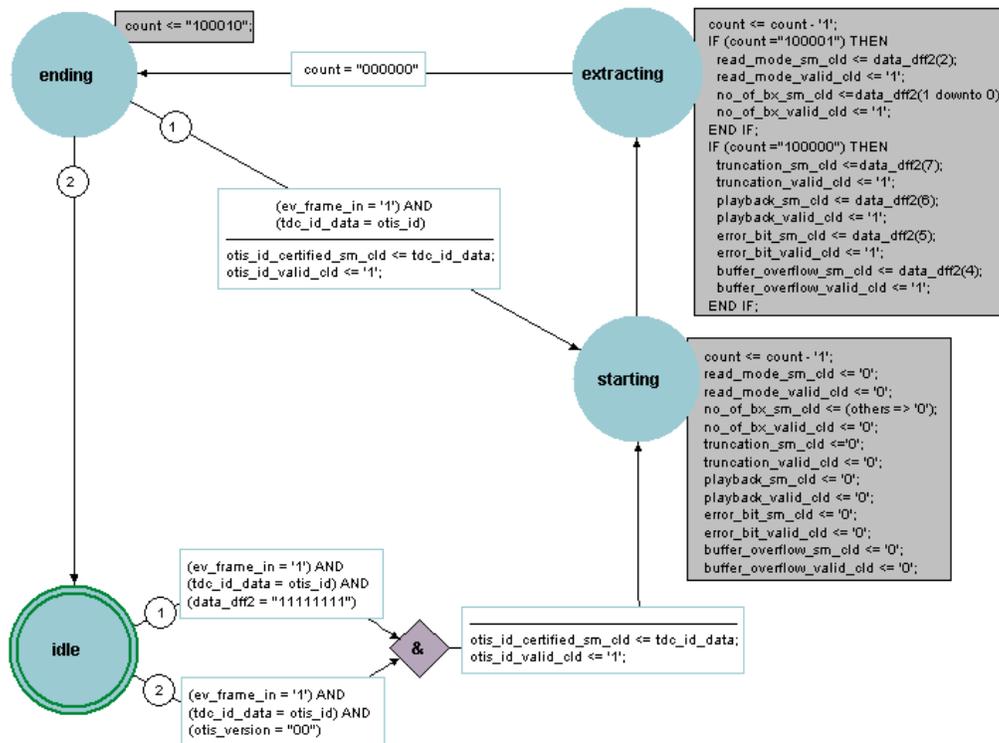


Abbildung 3.14: Die Statusbit-Extraktion. Der Wartezustand wird verlassen, sobald ein vom Synchronisationsblock als gültig markierter Datensatz mit korrekter Identifikationsnummer eintrifft.

Der Truncation Modus spielt nur eine Rolle, wenn der OTIS im Multi-Hit-Modus betrieben wird, in dem die Datensätze auch länger als 900 ns sein können. Auf den Multi-Hit Modus des OTIS wird nicht weiter eingegangen, weil diese Funktion im Serientest nicht überprüft wird.

Nach der Extraktion der einzelnen Bits des Datenkopfs folgt der Vergleich mit den Erwartungswerten. Die Erwartungswerte für den Header stehen im Konfigurationsregister. Bei Abweichungen von den Erwartungswerten gehen die entsprechenden Errorbits für einen Taktzyklus an. Die Häufigkeit, mit der dies während einer Messung geschieht, wird von 32 Bit Zählern registriert und in den PCI-Registern abgespeichert. Zeigt also ein Chip nicht das gewünschte Verhalten, wird dies in der Anzahl der gezählten Fehler im FPGA-Registern verdeutlicht. Bei erfolgreichem Test eines OTIS-Chips stehen am Ende alle Fehlerzähler für die Statusbits auf Null.

Der EventID- und der Bunch-CrossingID-Tester

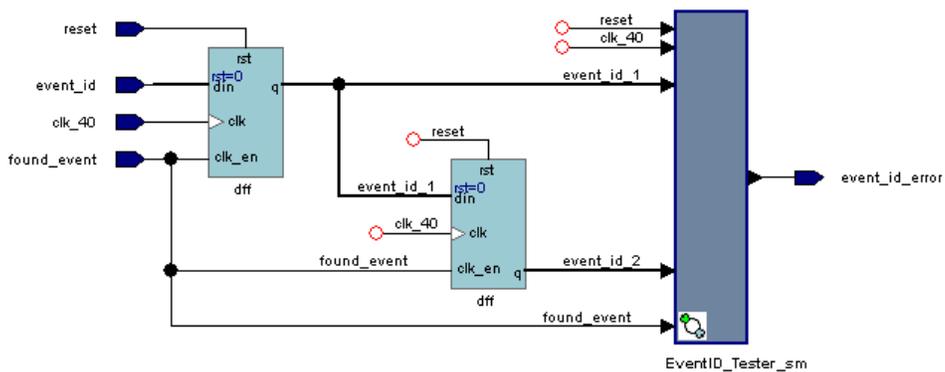


Abbildung 3.15: Der Event-ID-Tester

Zwei weitere Komponenten der Datenintegritätsprüfung sind der Event-ID- und der Bunch-Crossing-ID Tester. Überprüft werden soll, ob sich die Zähler mit jedem 40 MHz Taktzyklus um eins inkrementieren. Die Schaltung für den Test des EventID-Zählers ist in Abbildung 3.15 gezeigt. Das Found-Event-Signal stellt neben dem Event-Frame-Signal eine weitere Markierung eines Datensatzes dar, mit dem Unterschied, daß dieses Signal nur für einen Taktzyklus am Ende eines Datensatzes gesendet wird. Bereitgestellt wird es vom Synchronisationsblock. Es aktiviert beide D-Flipflops für einen Taktzyklus, um die Zählerwerte der Bunch-Crossings N und N+1 an eine Zustandsautomat weiterzuleiten. Die Zustandsautomat überprüft das korrekte Inkrementieren der Event-IDs. Bei Abweichungen werden Fehlerbits generiert, die wiederum im PCI-Controller Block bis zum Ende der Messung gezählt und ins Konfigurationsregister geschrieben werden. Auf dieselbe Art und Weise wurde der Bunch-Crossing Tester implementiert. Der einzige Unterschied besteht darin, daß die Bunch-Crossing-ID im Header des OTIS nicht 4 Bit, sondern 8 Bit breit ist.

Der Latency-Tester

Die letzte Einheit in diesem Block stellt der Test für die einstellbare Latenz des OTIS dar. Er empfängt die beiden Monitor-Signale des OTIS, welche die zuvor per I²C-Protokoll eingestellte Funktion darstellen. Im Zustandsgraphen in Abbildung 3.16 ist die Verhaltensweise dieser Einheit zu erkennen. Sobald der Lesezeiger den Wert eins annimmt, werden die Taktzyklen gezählt, bis der Schreibzeiger ebenfalls auf eins steht. Der Abstand zwischen fallender Flanke des Lesezeigers und steigender Flanke des Schreibzeigers stellt genau die Latenz des OTIS dar. Der so erhaltene Wert wird mit der erwarteten Latenzzeit verglichen. Die ganze Prozedur wiederholt sich, wenn der Lesezeiger die 164 Zeilen des Pipeline Registers durchlaufen hat und wieder von vorne anfängt. Mögliche Fehler werden daher alle 4.1 μ s gezählt.

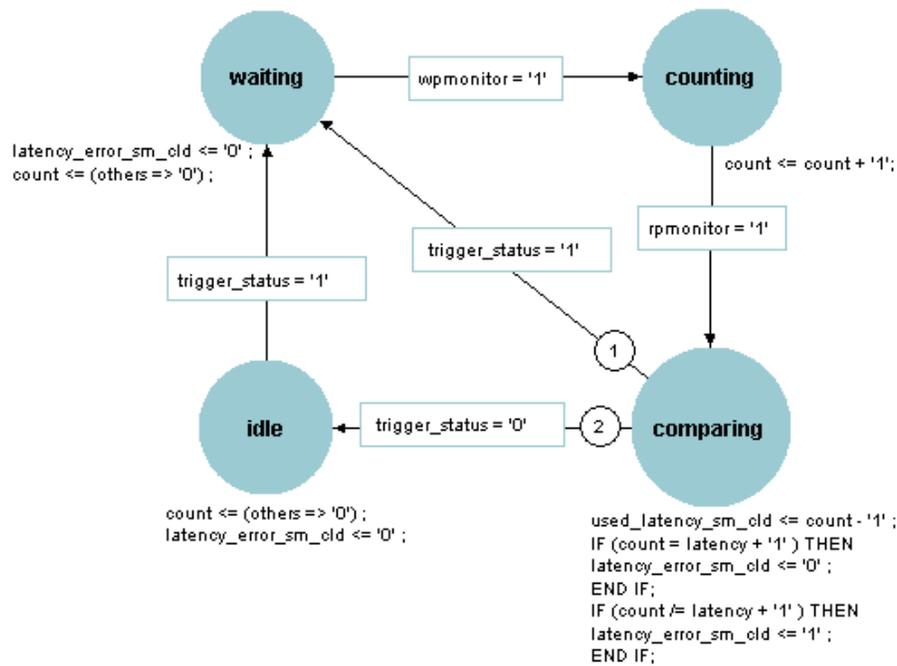


Abbildung 3.16: Der Latency-Tester

3.2.5 Der Histogrammierblock

Um bestimmte Eigenschaften des OTIS (DNL) bzw. bestimmte Funktionen (Ansprechbarkeit aller 32 Kanäle) zu testen, muß eine große Zahl von Ereignissen ausgewertet werden. Das Bilden von Histogrammen auf dem FPGA führt zu einer entscheidenden Beschleunigung des Testablaufs.

In Abbildung 3.17 ist der Histogrammer-Block schemenhaft dargestellt. Der Block zur Histogrammierung ist modular aufgebaut und läßt sich daher für verschiedene Zwecke verwenden. Aufgenommen werden die Histogramme für die Driftzeiten, für die Chan-

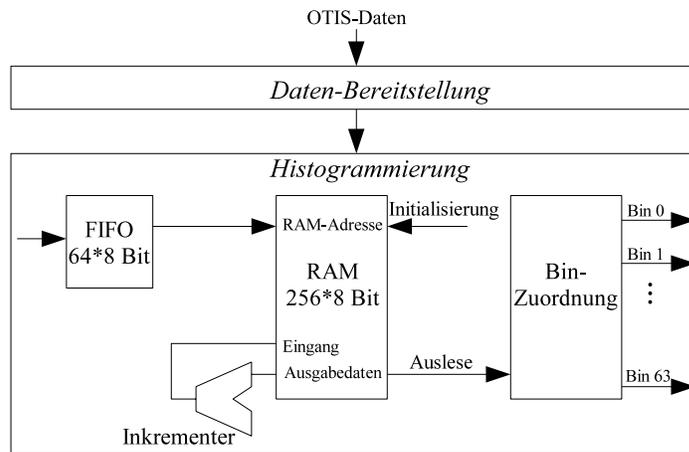


Abbildung 3.17: Der Histogrammer

nel Map und für die Event-IDs. Die gewählte Speichertiefe des RAM-Bausteins erlaubt die Histogrammierung von 256 verschiedenen Werten. Davon werden bei der Driftzeit-Histogrammierung 64 Zeilen benötigt, für die Channel-Map-Histogrammierung 32 und 16 für die Histogrammierung der Event-ID. Je nachdem welcher Wert histogrammiert werden soll, muss dem Histogrammierblock eine entsprechende Einheit zur Bereitstellung der Daten vorgeschaltet sein.

Bereitstellung der Daten

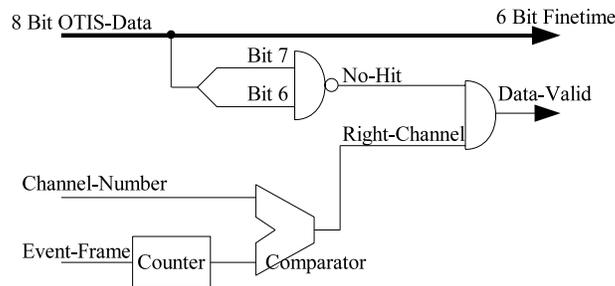


Abbildung 3.18: Bereitstellung der Driftzeitdaten

Exemplarisch ist hier gezeigt, wie die Driftzeitdaten für den Histogrammer-Block bereitgestellt werden. Bei der Bereitstellung der Driftzeitdaten ist zu beachten, daß entweder alle Driftzeiten oder nur die eines bestimmten Kanals aus dem Datenstrom herausgesucht werden. Die Wahl des Kanals wird über das Konfigurationsregister bestimmt und dem Block mit Hilfe des „Channel-Number“-Signals mitgeteilt (siehe Abbildung 3.18).

Ist die zum voreingestellten Kanal gehörende Driftzeit extrahiert, muß entschieden werden, ob es sich dabei um einen gültigen Treffer handelt. Dazu werden in der Schaltung die ersten beiden Bits der 8 Bit-OTIS-Daten analysiert. Sie zeigen an, ob ein Treffer stattgefunden hat und wenn ja in welchem Bunch-Crossing. Sind beide Bits aktiv, wurde

kein Treffer erzeugt. Das „Data-Valid“-Signal hinter dem negierten Und-Glied bleibt auf null und die Driftzeit wird nicht ins Histogramm aufgenommen. Statt dessen wird das Ereignis als „kein Treffer“ in einem 32 Bit Zähler registriert. Innerhalb des Testaufbaus tritt dieser Fall ein, wenn dem OTIS ein Trigger geschickt wird, obwohl zeitgleich kein Pulserignal an seinem Treffereingang anliegt. Dies ist möglich, da sich Trigger und der für die Erzeugung von Treffern zuständige Pulser völlig asynchron zueinander verhalten. Liegt allerdings ein Treffer vor, so wird dies der nachfolgenden Histogrammer-Einheit über das „Data-Valid“-Signal mitgeteilt.

Die nachfolgende Histogrammer-Einheit erfüllt zwei verschiedene Aufgaben. Sie stellt zum einen die Verteilung der Driftzeiten in den verschiedenen Zeitbins des OTIS in einem Histogramm dar. Der OTIS unterteilt ein LHC-Bunch Crossing in 64 verschiedene möglichst gleichgroße Zeitintervalle oder Zeitbins. Es soll für alle Chips im LHC-Detektor gewährleistet sein, daß sich die Breiten der Zeitintervalle nicht zu stark unterscheiden, um die Anforderung an die Genauigkeit der Zeitmessung zu erfüllen. Um das zu verifizieren, werden die OTIS-Eingänge mit zufälligen Treffern von einem Pulser versorgt. Ziel ist es, eine gleichmäßige Verteilung der Treffer auf alle Zeitbins zu erhalten.

Um zu sehen, ob die 32 Kanäle des OTIS ansprechbar sind, ist im OTIS-Serientest eine Prozedur geplant, in der zunächst nur Pulse auf die ungeraden Kanäle gegeben werden. Im zweiten Schritt wiederholt sich die Messung für die geraden Kanäle. Im zu erstellenden Histogramm dürfen für den ersten Fall also alle geraden Kanäle keine Treffer enthalten und die ungeraden statistisch verteilt gleich viel Einträge sehen. Für den zweiten Teil muß das Umgekehrte gelten. Wichtig für die Bereitstellung der Daten bevor sie in ein Histogramm geschrieben werden ist also nur, ob ein Treffer stattgefunden hat oder nicht und in welchem Kanal dies geschehen ist. Unbeachtet bleiben die in 6 Bit dekodierten Driftzeiten aus den Datensätzen. Auch spielt es keine Rolle, in welchem Bunch-Crossing ein Treffer registriert wurde.

Die Methode der Histogrammierung

Nach Verlassen der Datenvalidierung folgt nun die nächste Stufe der Histogrammierung. Da dieser Teil modular konstruiert wurde, kann man für die Bildung der verschiedenen Histogramme denselben Block dreimal verwenden. Man kann ihn grob einteilen in einen Dateneinleseteil und einen Datenausgabe -und Initialisierungsteil (vgl. Abbildung 3.19 und 3.22).

Der Dateneinleseteil

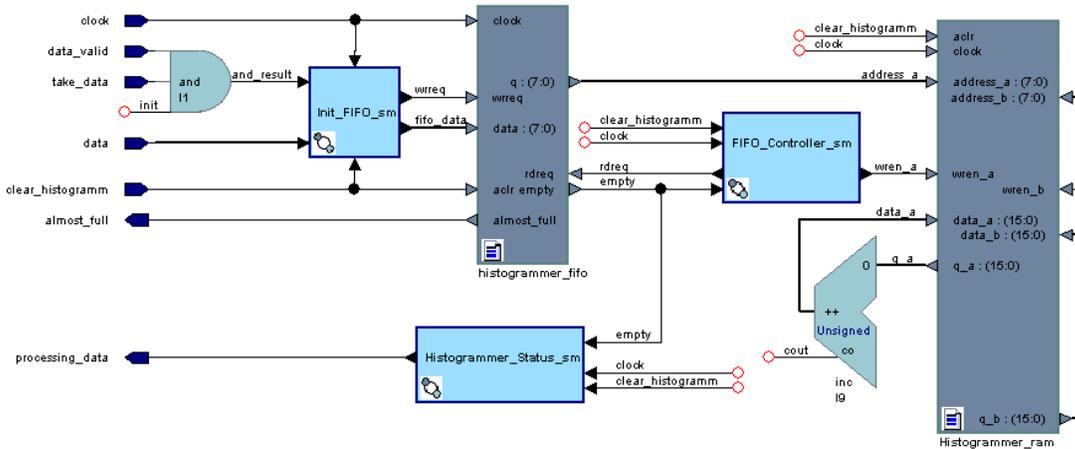


Abbildung 3.19: Erster Teil des Histogrammers

Zu Beginn werden die Daten zur Pufferung in ein 8 x 256 Bit großes FIFO geschrieben. Es gibt drei Bedingungen für die Aufnahme eines Datensatzes ins FIFO:

- Der RAM-Baustein darf sich nicht im Zustand der Initialisierung befinden
- Das „data_valid“-Signal muß für die Länge eines Datensatzes auf eins stehen
- Der Befehl „Start Measurement“ aus dem PCI-Controller muß gesetzt sein

Sobald sich ein Datensatz im FIFO befindet, wird dies dem „FIFO-Controller“- Zustandsgraphen (vgl. Abbildung 3.20) über das „empty“-Signal angezeigt. Es folgt nun ein vom Zustandsautomaten initiiertes Lesezugriff auf das FIFO und zwei Takte später ein Schreibzugriff auf den RAM-Baustein.

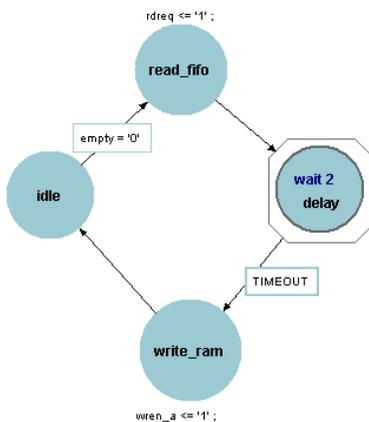


Abbildung 3.20: Der FIFO-Controller Zustandsautomat

Der Dual-Portet RAM Baustein ist der Hauptbestandteil in dieser Einheit. Er wird vom „ALTE-RA Megawizard“, einer Bibliothek, die speziell auf diesen FPGA zugeschnittene logische Einheiten beinhaltet, zur Verfügung gestellt.

Genutzt werden die auf dem FPGA-Chip platzierten M-RAM-Bausteine, die jeweils 576 Byte groß sind [20]. Dual-Portet RAM bedeutet, daß gleichzeitige Lese- und Schreibzugriffe an unterschiedlichen Adressen möglich sind. In dieser Schaltung wurde eine Konfiguration des RAM-Speichers

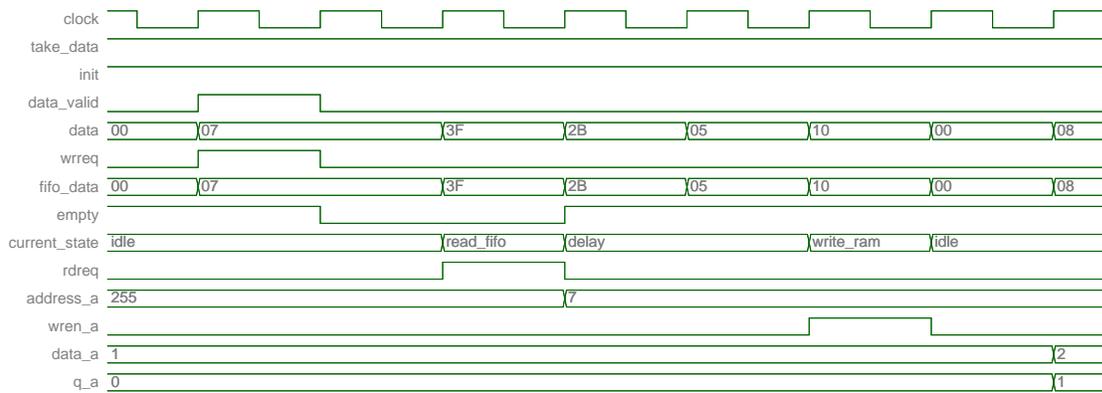


Abbildung 3.21: Simulation des Dateneinleseteils des Histogrammers. Das „data-valid“-Signal aus der vorgeschalteten Einheit zur Datenbereitstellung markiert eine eingehende Driftzeit. Das dreifache Und-Glied stößt einen Schreibbefehl an das FIFO an („wrreq“), welches den FIFO-Controller über das „empty“-Signal in den „Read-FIFO“-Zustand bringt. Zwei Takte später initiiert der FIFO-Controller einen Schreibbefehl an den RAM-Baustein („wren-a“): Adresse 7 erhält einen zusätzlichen Eintrag.

von 256×16 Bit gewählt. Das heißt, daß die Adressierung über eine 8 Bit breite Datenleitung erfolgt, wobei in jeder Adresse Werte bis 65535 stehen können. Für das Beispiel der Driftzeithistogrammierung wird jedes Zeitbin von 0 bis 63 durch die ersten 64 Adressen des RAM-Bausteins vertreten. Wenn ein Treffer in einem Bin des OTIS stattgefunden hat, wird zunächst die Adresse im RAM-Baustein ausgelesen, die der Nummer des Bins entspricht. Haben in diesem Bin vorher schon Treffer stattgefunden, wird die Anzahl zwei Takte nachdem die Adresse angelegen wurde ausgegeben und anschließend inkrementiert. Liegt nun der neue Wert am Dateneingang an, wird vom Zustandsautomaten ein Schreibbefehl für das RAM angestoßen, um den neuen Wert abzuspeichern. Mit Hilfe des RAM-Bausteins besteht also die Möglichkeit, die Einträge für die 64 Zeitbins des OTIS in den zugehörigen Adressen nacheinander aufzusummieren. Weil die Driftzeiten in sechs Bit kodiert sind, man bei der Adressierung von 256 Adressen allerdings einen 8 Bit-Wert erwartet, wurden die Driftzeitdaten im vorherigen Validierungsblock mit Nullen in den ersten beiden MSBs (Most Significant Bits) aufgefüllt.

Betrachtet man die Histogrammierung der Channel-Map, stellt hier jeder 8 Bit-Wert die Zahl 0 bis 32 für die 32 Kanäle dar. Ebenso gilt dies für die Event-ID, für die die Werte 0 bis 15 in die ersten 16 Adressen des RAMs geschrieben werden.

Eine Simulation der Signale des Histogrammers ist in Abbildung 3.21 gezeigt.

Initialisierung und Datenausgabe

Im zweiten Teil der Histogrammeinheit (vgl. Abbildung 3.22) findet erstens die Initialisierung und zweitens die Ausgabe der Daten des RAM-Bausteins statt. Um alle Werte, die im RAM stehen zu löschen, genügt in dem Fall kein einfaches Zurücksetzen mehr. Um eine neue Messung zu starten, muss vorher der gesamte Adressbereich des RAMs mit Nullen beschrieben werden, andernfalls werden die alten Werte wieder übernommen und

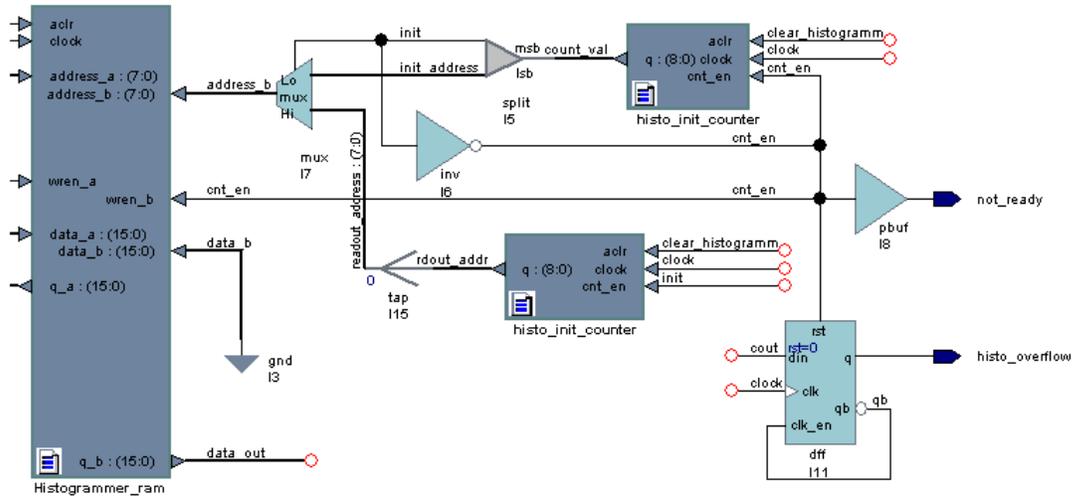


Abbildung 3.22: Zweiter Teil des Histogrammers

weiter inkrementiert. Hierzu aktiviert ein 8 Bit Zähler nach einem Design-Reset-Befehl des Anwenders² den Schreibzugriff auf den RAM-Baustein mit dem „wren_b“-Signal und zwar für 256 Taktzyklen. Dadurch werden alle Adressen mit den an „data_b“ anliegenden Nullen überschrieben. Während dieser Zeit ist das „init“-Signal inaktiv, so daß weder Daten vom FIFO aufgenommen werden, noch ein Schreibzugriff auf der Seite der Dateneinleseseite stattfinden kann.

Die Initialisierung endet, wenn das höchste des Zählers angeht. An dieser Stelle wird ein zweiter Zähler aktiviert, der in einer Schleife alle 256 Adressen des RAM-Bausteins solange durchläuft, bis wieder ein Reset des Designs erfolgt. In diesem Zustand ist das Modul bereit, Meßwerte aufzunehmen. Eine Simulation ist in 3.23 dargestellt.

Die Auslese der Daten aus dem RAM geschieht gleichzeitig zum Schreiben auf der Dateneinlese-Seite. Die zu jeder Ausleseadresse gehörigen Daten werden über das „data_out“-Signal zur nächsten Einheit weitergeleitet.

Für die Dauer, in der Schreib- oder Lesezugriffe auf das RAM absolviert werden, aktiviert der Zustandsautomat in Abbildung 3.19 das Signal „processing_data“. Dieses Signal wird über die Konfigurationsregister des PCI-Busses dem Anwender sichtbar gemacht.

Abschließend müssen die Daten, die das RAM ausgibt noch der richtigen Bin-Nummer im Histogramm zugeordnet werden. Dies geschieht in der Datenzuordnungseinheit von Abbildung 3.17. Es existieren 64 16 Bit-Busse, um die Einträge für die Zeitbins 0 bis 63 an die PCI-Register zu übertragen. Die Logik in der Datenzuordnungseinheit verteilt die Zählwerte der ausgelesenen RAM-Adressen³ auf diese 64 Busse. Es muß dabei berücksichtigt werden, daß der Wert einer Ausleseadresse erst zwei Takte nach einer gestellten Leseanfrage an das RAM ausgegeben wird. Bewerkstelligt wird diese Aufgabe von 64

²Dieser Befehl wird in Abbildung 3.22 mit dem Signal „clear histogramm“ erteilt.

³Diese Adressen sind identisch mit den Nummern der Zeitbins

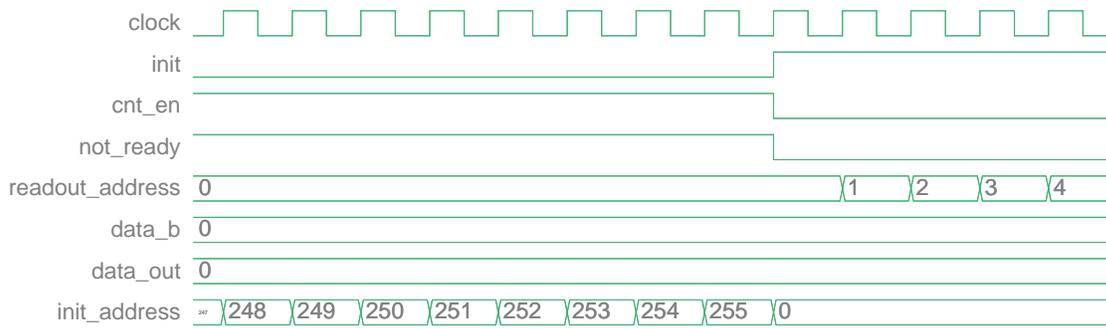


Abbildung 3.23: Simulation des Initialisierungs- und Auslesteils des Histogrammers. In der ersten Hälfte werden alle Adressen des RAM-Blocks bis Adresse 255 durchlaufen. Danach ist die Initialisierung abgeschlossen und das „not-ready“-Signal wird inaktiv. Mit dem „cnt-en“-Signal wird der Zähler für die Initialisierung ausgeschaltet. Gleichzeitig schaltet das aktive „init“-Signal den Zähler für die Datenausgabe ein.

Komparatoren, die immer dann einen Wert auf einen Bus schreiben, wenn die zu ihm gehörende Ausleseadresse am RAM angelegen hat.

Die gesamte beschriebene Einheit zur Histogrammierung existiert dreimal, da für die Driftzeiten, die Channel-Map und die Event-IDs drei verschiedene Histogramme erstellt werden müssen. Diese arbeiten parallel nebeneinander, so daß nach einer fertigen Messung alle drei Histogramme gleichzeitig ausgegeben werden.

3.2.6 Die Daten-Prozessierung

Das Banyan-Netzwerk

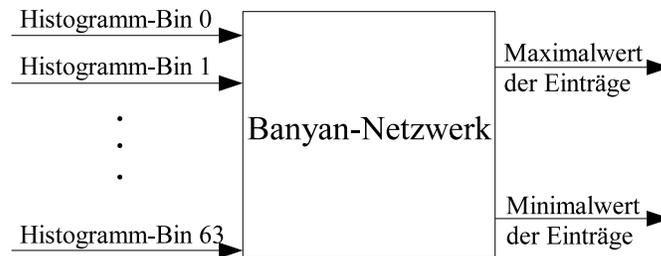


Abbildung 3.24: Das Banyan-Netzwerk

Für den Serientest des OTIS ist es notwendig, aus den Driftzeit-Histogrammen die maximalen und die minimalen Zählwert zu ermitteln. Mit diesen Informationen läßt sich feststellen, ob alle Driftzeiten vom OTIS generiert werden konnten. Ist das nicht der Fall, beträgt das Minimum für mindestens ein Zeitbin null. Ausserdem ist die Ausgabe der Extremwerte nützlich für den Schaltungsentwurf, weil man so die vom FPGA erzeugten Histogramme mit denen über Software berechneten Histogramme auf ihre Richtigkeit

hin untersuchen kann. Zum Heraussuchen von Minimal- und Maximalwert eines Histogramms eignet sich besonders das Banyan-Netzwerk [19].

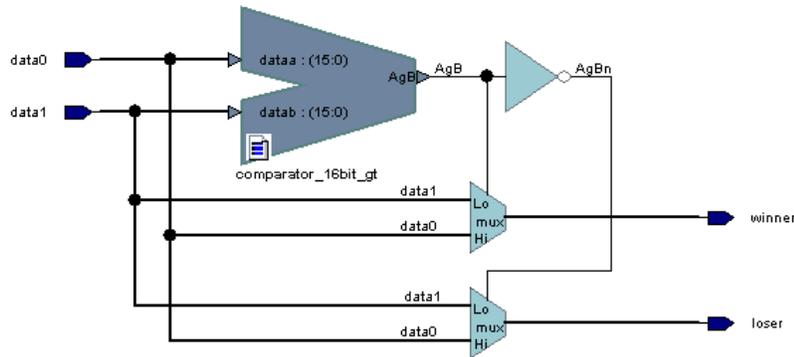


Abbildung 3.25: Die Basisebene des Banyan-Netzwerks

In der Telekommunikation sind Banyan-Netzwerke wichtige Grundschaltungselemente, um auf möglichst schnellem Weg n Eingangswerte nach ihrer Größe sortiert auszugeben. Charakteristisch für diese Art der Schaltung ist, daß sich ihre Strukturen ausgehend von einer Basisschaltung in darüberliegenden Ebenen und in aufeinanderfolgenden Stufen wiederholen.

In der untersten Ebene befindet sich das Basis-Schaltwerk (siehe Abbildung 3.25). Die Werte zweier eingehender 16 Bit Busse werden hier mittels Komparator verglichen und der größere Wert als Gewinner, der kleinere als Verlierer auf die Ausgänge geschrieben. Die zweite Ebene (vgl. Abbildung 3.26) enthält vier dieser „Basic Banyans“. Die ersten beiden Module bestimmen wieder die zwei größten und die zwei kleinsten Werte und geben sie an die nächsten zwei Module weiter. Aus den beiden Maximalwerten wird erneut der Gewinner bestimmt, aus den kleineren wieder der Verlierer. Diese Struktur wiederholt sich in der übergeordneten dritten Ebene (Abbildung 3.27)

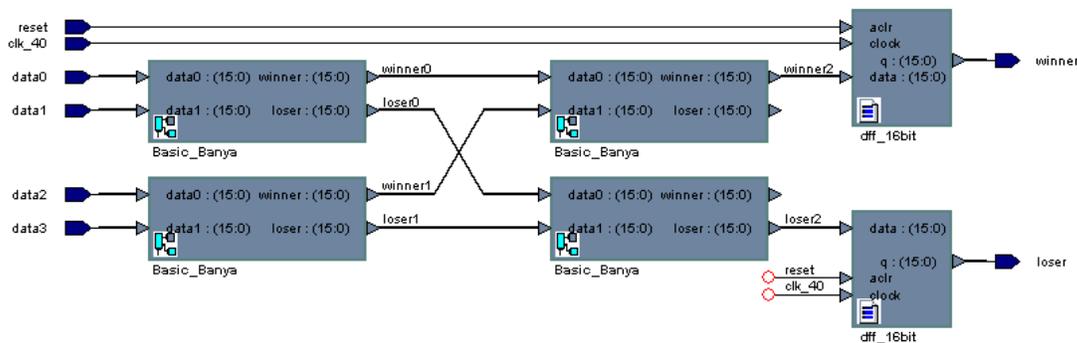


Abbildung 3.26: Die zweite Ebene des Banyan-Netzwerks

mit dem Unterschied, daß jetzt 16 Eingänge zur Verfügung stehen. Da hier 64 Werte zu vergleichen sind setzt sich die Struktur bis in die vierte Ebene fort (siehe Abbildung 3.28).

Allgemeiner werden bei n Eingängen $n/2$ Schaltwerke pro Stufe und $\log_2 n$ Stufen benötigt. Für die 64 verschiedenen Werte eines Driftzeithistogramms erhält man daher 32 Schaltwerke und sechs Stufen, bis alle Werte der Größe nach sortiert sind. Ausgegeben werden in diesem Fall nur der maximale und der minimale Wert für die Einträge aller Zeitbins aus dem Driftzeithistogramm.

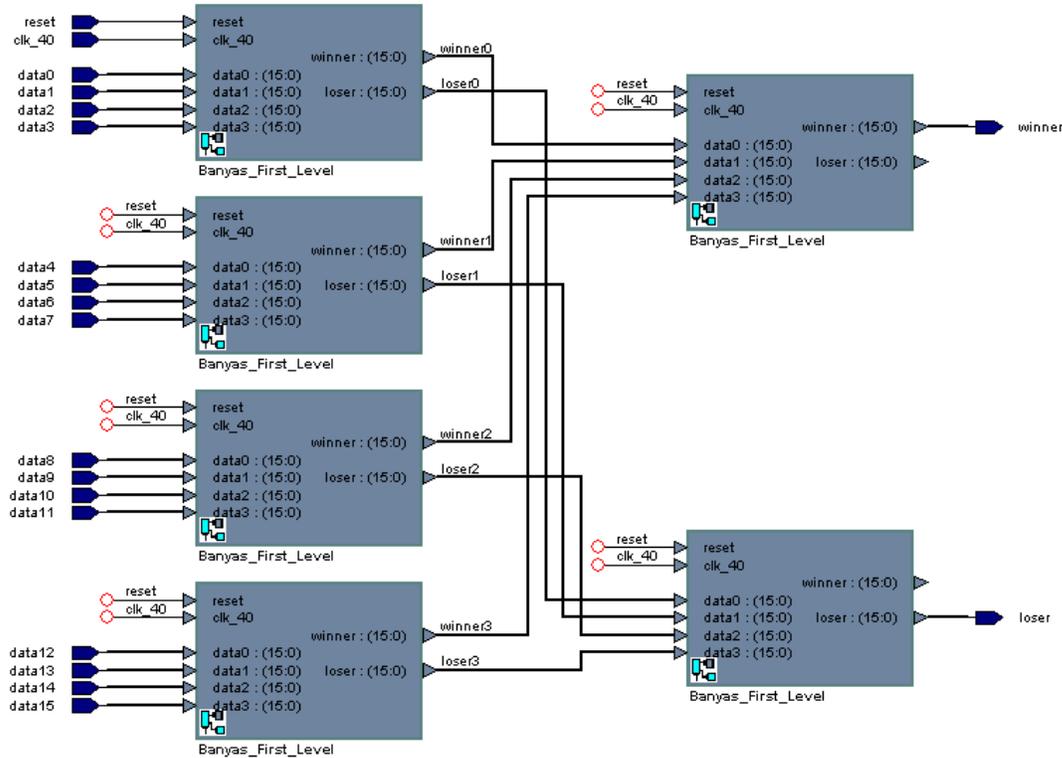


Abbildung 3.27: Die dritte Ebene des Banyan-Netzwerks

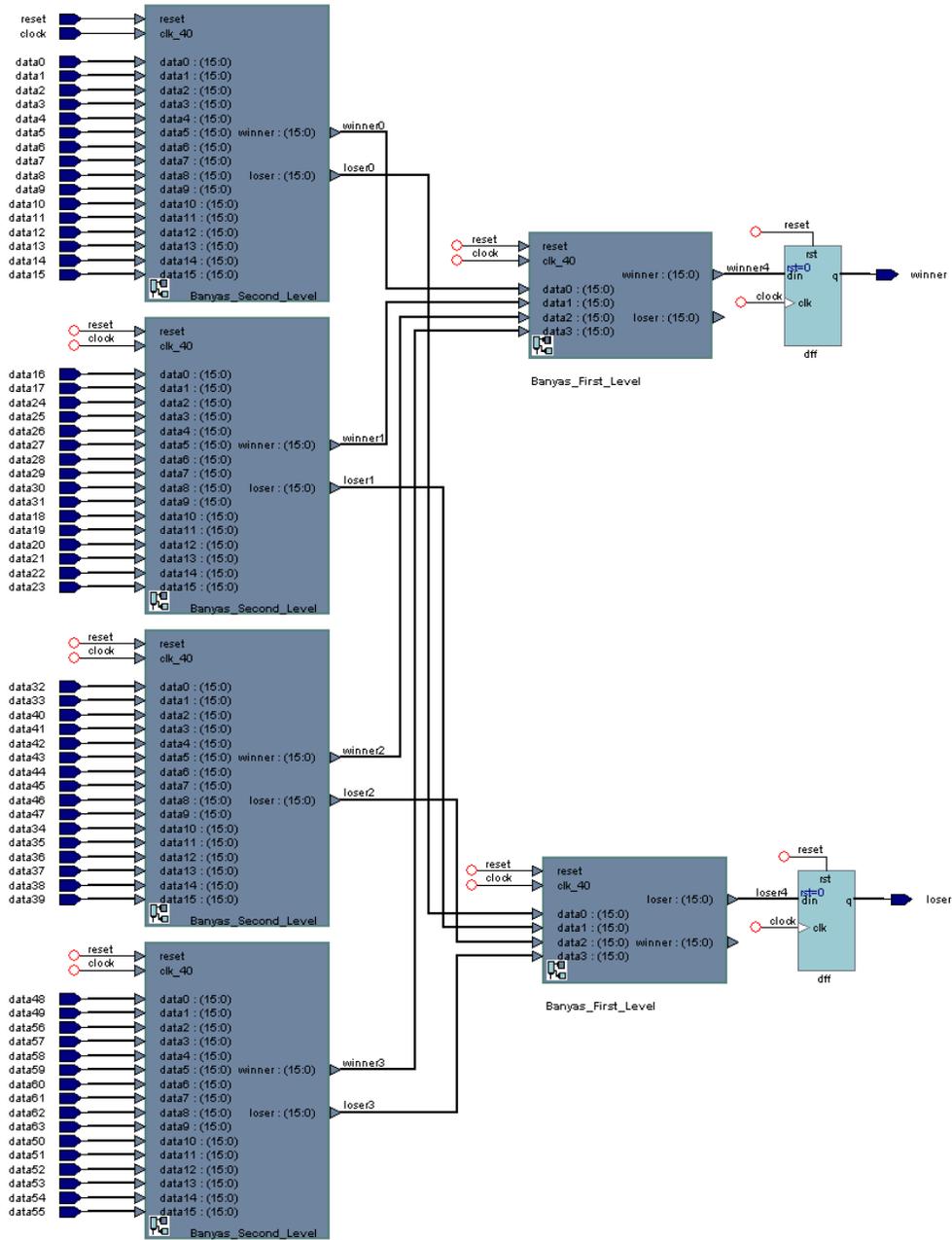


Abbildung 3.28: Die vierte Ebene des Banyan-Netzwerks

Die Berechnung der differentiellen Nichtlinearität

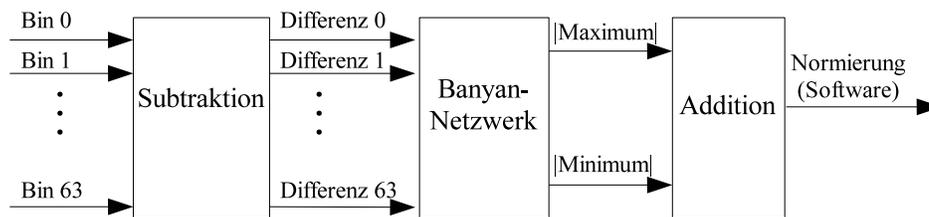


Abbildung 3.29: Berechnung der DNL auf dem FPGA. Der letzte Schritt der Normierung wird per Software vollzogen.

Der DNL-Wert spielt bei der Charakterisierung des Zeitmeßverhaltens des OTIS eine große Rolle. Analog zu ADCs (Analog to Digital Converter), die eine Spannung in einen digitalen Wert umwandeln, ordnet der TDC Zeitdifferenzen einen Digitalwert zu. Da die DLL-Kette ein Bunch-Crossing von 25 ns in 64 Teilstücke unterteilt, ist die theoretische Breite der Quantisierungsstufe auf 390 ps festgelegt. In einem Driftzeithistogramm ist die Anzahl der Einträge in einem TDC-Bin proportional zu seiner Breite. Die DNL für jedes Zeitbin wird nach Formel 3.1 berechnet. Dieser Wert gibt die Abweichung der gemessenen Breite des Zeitbins von der theoretischen Breite in Einheiten von Bins an. Die Summe aus dem Betrag des Maximums und des Minimums der DNL-Werte aller Zeitbins ergibt die endgültige DNL des OTIS.

$$\text{DNL}(i) = \frac{\text{Einträge in Bin } (i) - \text{Einträge in Bin } (i-1)}{(\sum \text{ aller Einträge }) / \text{Anzahl der Bins}} \quad (3.1)$$

$$\text{DNL} = \{|\max(\text{DNL}(i))| + |\min(\text{DNL}(i))|\} \quad (3.2)$$

Alle Rechenschritte bis auf die Division werden auf dem FPGA vollzogen (Abbildung 3.29). Nach Bildung der Differenzen der Einträge zwischen allen Zeitbins gelangen die Werte in ein Banyan-Netzwerk. Dieses liefert das Maximum und das Minimum aller Differenzen. Der letzte Schritt auf dem FPGA ist die Addition von Maximum und Minimum, bevor die Division durch die Gesamtzahl aller Einträge von einer für dieses Projekt programmierten Benutzeroberfläche durchgeführt wird.

Kapitel 4

Messungen zum FPGA-Schaltungsentwurf

Im folgenden wird der Test der verschiedenen Funktionseinheiten des FPGA vorgestellt. Zu den Funktionseinheiten gehören:

- die Einheit zur Datenintegritätsprüfung
- der Histogrammer-Block
- die Datenprozessierung

Um die korrekte Arbeitsweise der Einheit für die Datenintegritätsprüfung zu zeigen, wurde der OTIS-Dummy zu Hilfe gezogen. Er ist zur Simulation eines OTIS-Chips auf dem FPGA implementiert. Der OTIS-Dummy erlaubt die Vorgabe von Driftzeiten und von Bitmustern im Datenkopf. Auf diese Weise ist festgelegt, wie sich die nachfolgenden Funktionseinheiten auf dem FPGA verhalten. Die Tests zum Histogrammer-Block und zur Datenprozessierungseinheit wurden mit Hilfe einer OTIS-Testkarte vorgenommen. Auf der Testkarte sind sämtliche vom OTIS ausgegebene Signale abgreifbar. Ausserdem befinden sich auf ihr die Anschlüsse für die Slow-Control- und die Fast-Control-Signale.

4.1 Die Fast Control auf dem FPGA

Die Funktionsweise der Fast-Control auf dem FPGA wurde in Kapitel 3 erklärt. Sie erzeugt die Resetsignale, die 40 MHz-Taktfrequenz und die Triggersignale für den OTIS. Die Frequenz, mit der die Trigger erzeugt werden läßt sich auf dem FPGA einstellen. Auch die Länge der zu generierenden Resetsignale ist variabel.

Als Beispiel ist in Abbildungen 4.1 eine Abfolge von gesendeten Resets gezeigt. Auf dem FPGA besteht die Option, diese Signale mit einem einzigen Befehl hintereinander erzeugen zu lassen. Der OTIS erwartet an seinen Reseteingängen Signale mit negativer Logik. Als erstes generiert das Signal für das Level0-Reset für 500 ns eine logische Null, gefolgt von dem Bunch-Counter-Reset und dem Event-Counter-Reset. Damit sich die Signale nicht überlagern, ist nach jedem gesendeten Reset ein Wartezustand von 1.5 μ s

vorgesehen.

Abbildung 4.2 zeigt den vom FPGA erzeugten 40 MHz-Takt, der gleichzeitig die Arbeitsfrequenzen für die FPGA-Schaltung und für den OTIS bereitstellt. Das Triggersignal kommt Taktsynchron und besitzt eine Länge von 25 ns. Dieses veranlaßt den OTIS dazu einen Datensatz von 900 ns auszugeben, was durch das negative „Data-Valid“-Signal angezeigt wird.

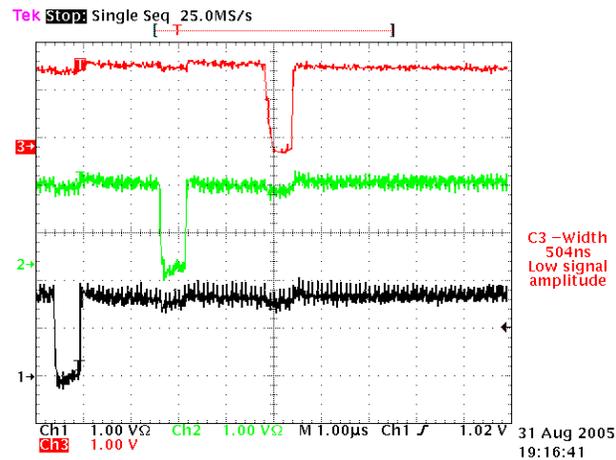


Abbildung 4.1: Die Abfolge der Resets nach dem FPGA-Befehl „All Resets“. Zunächst kommt das Level0-Reset (Kanal 1), gefolgt von dem Bunch-Counter-Reset (Kanal 2) und dem Event-Counter-Reset (Kanal 3). Jedes Reset wird für 500 ns aktiviert.

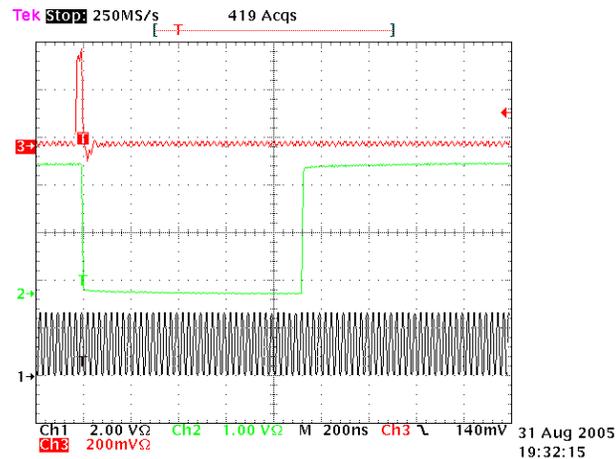


Abbildung 4.2: Dem vom FPGA generierte Trigger (Kanal 1) folgt ein 900 ns langer OTIS-Datensatz, angezeigt durch das „Data-Valid“-Signal (Kanal 2). Kanal 3 zeigt die vom FPGA bereitgestellte 40 MHz Taktfrequenz.

4.2 Die Datenintegritätsprüfung

4.2.1 Die OTIS-Datenkopf-Analyse

Die auf dem FPGA implementierte OTIS-Datenkopf-Analyseeinheit hat den Zweck, die digitale Funktionalität des OTIS zu überprüfen. Der Status des Digitalteils des OTIS findet sich in seinem Datenkopf wieder. Daher wird dieser im Serientest analysiert. Der FPGA hat die Aufgabe, Abweichungen vom gewünschten Verhalten des OTIS anzuzeigen. Dies geschieht über die Zählwerte der Fehlerbits aus der in diesem Abschnitt getesteten Einheit.

Die Überprüfung der Funktionalität der OTIS-Datenkopf-Analyseeinheit wurde mittels einer Datensimulation durch den OTIS-Dummy vorgenommen. Der OTIS-Dummy ist auf dem FPGA implementiert. Seine Ausgabedaten haben exakt die gleiche Struktur wie die eines OTIS-Chips. Der Datenkopf eines Chips enthält neben den Informationen über die internen OTIS-Einstellungen die Statusbits für den Füllzustand des Speichers und für die korrekte Arbeitsweise der DLL. Mit dem OTIS-Dummy ist es möglich, diese Werte beliebig einzustellen, um so verschiedene Verhaltensweisen des Chips simulieren zu können.

Zur Überprüfung dieser Einheit wurden zwei verschiedene Bitmuster in den Datenkopf des OTIS-Dummy geschrieben. Im ersten Fall wurden alle Header-Bits mit Nullen belegt, was identisch mit den erwarteten Werten im FPGA-Design war (siehe Abbildung 4.3). Der zweite Fall unterscheidet sich dahingehend, daß nun alle Header-Bits aktiviert wurden.

In der in Kapitel 3 vorgestellten Schaltung für die Datenintegritätsprüfung wurde veranschaulicht, wie diese Bits mit Hilfe einer Zustandsautomat aus dem Datenstrom extrahiert werden. Für jedes extrahierte Header-Bit existiert eine eigene Signalleitung bzw. ein eigener Bus. Der Statusbitvergleich mit den erwarteten OTIS-Rückgabewerten findet in der sich anschließenden Einheit statt. Die zu erwartenden Werte stehen im Konfigurationsregister und müssen zuvor durch den Anwender gesetzt worden sein.

Da beim ersten Test die Dummy-Daten mit den erwarteten Werten übereinstimmten, blieben alle Fehlerbits nach Durchlaufen von 10.000 Dummy-Datensätzen auf null.

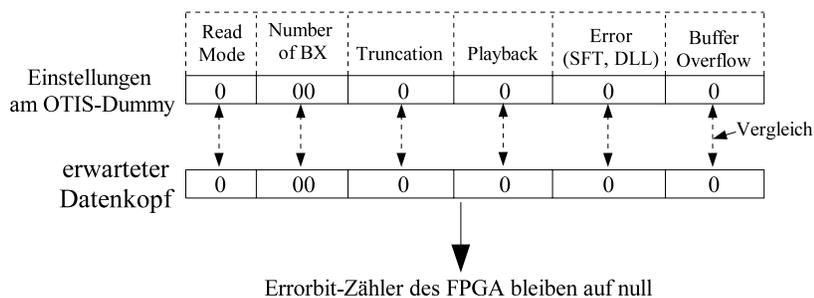


Abbildung 4.3: Test der OTIS-Datenkopf-Analyseeinheit der FPGA-Schaltung. Der Bitweise Vergleich der erwarteten Statusbits mit denen des OTIS-Dummy liefert eine Übereinstimmung. Die entsprechenden Ausgabewerte des FPGA für die Errorbit-Zähler blieben auf null.

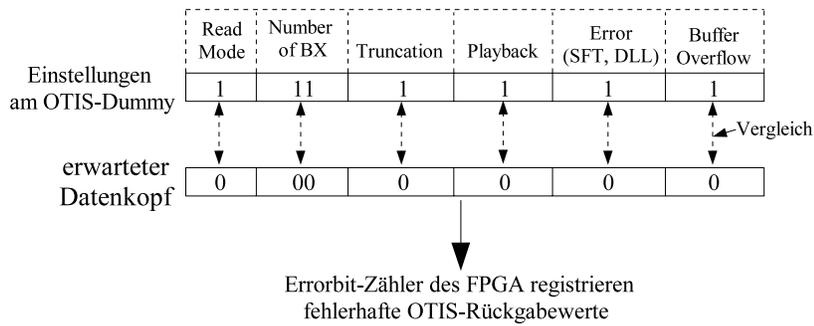


Abbildung 4.4: Zweiter Test der OTIS-Datenkopfanalyseeinheit. Alle Errorbit-Zähler des FPGA wurden aktiv

Im zweiten Fall (siehe Abbildung 4.4) wurden die Einstellungen für den erwarteten Datenkopf so gewählt, daß alle Errorbit-Zähler dieser Einheit auf dem FPGA aktiviert werden mußten. Nach Verarbeitung von 10.000 Dummydatensätzen zeigten genau diese Zähler 10.000 Fehler an. Die Funktionsweise dieser Einheit wurde in späteren Testläufen mit einem angeschlossenen OTIS-Chip mehrfach mit dem Testaufbau in Kapitel 5 verifiziert. Dazu wurde die direkte Datenausgabereinheit der FPGA-Schaltung benutzt. Sie erlaubt das Abspeichern der originalen OTIS-Datensätze auf dem Rechner, nachdem sie im FPGA-Design synchronisiert wurden. Die so erhaltenen OTIS-Header aus den Rohdaten erlaubten einen Vergleich mit den vom FPGA-Design erwarteten Statusbits. Die sich daraus ergebenden Voraussagen für die Werte der Bit-Errorzähler des FPGA-Designs stimmten mit den tatsächlichen Zählwerten überein.

4.2.2 Der EventID-Tester

Die vier Least Significant Bits (LSB) des 8 Bit Event-ID-Zählers auf dem OTIS-Chip werden in den Datenkopf eines Datensatzes geschrieben. Der momentane Zählwert dieser vier Bit wird im Synchronisationsblock aus dem OTIS-Datenstrom extrahiert. Mit jedem akzeptierten Trigger wird der Zähler um eins inkrementiert. Die in Kapitel 3 beschriebene Schaltung erkennt, wenn diese Funktionalität nicht mehr gegeben ist. Mögliche auftretende Fehler werden auf dem FPGA von einem 32 Bit Zähler registriert. Um zu zeigen, daß diese Einheit fehlerfrei arbeitet, wurden mehrere verschiedene Tests durchgeführt.

Zunächst fand der OTIS-Dummy seine Anwendung. Er besitzt analog zum OTIS-Chip einen Zähler für die Event-ID, der mit jedem Trigger hochzählt. Damit ist das Inkrementieren der Event-ID-Werte in den Daten garantiert. Entsprechend zeigten sich bei Durchführung einer Meßreihe keine Einträge in den ausgegebenen Event-ID-Fehlerbits des FPGA.

Der FPGA erlaubt gleichzeitig die Aufnahme eines Histogramms der Event-ID. Dieses muß in allen Bins gleich viel Einträge haben, wenn der OTIS-Dummy ein ganzzahliges Vielfaches von 16 Datensätzen erzeugt hat. Abbildung 4.5 zeigt eine solche Messung, wobei jetzt nicht mehr der OTIS-Dummy, sondern ein Chip ausgelesen wurde. Der genaue Testaufbau wird in Kapitel 5 beschrieben.

Die Fast Control auf dem FPGA wurde im „Trigger-Modus“ betrieben. Nach Durchführung

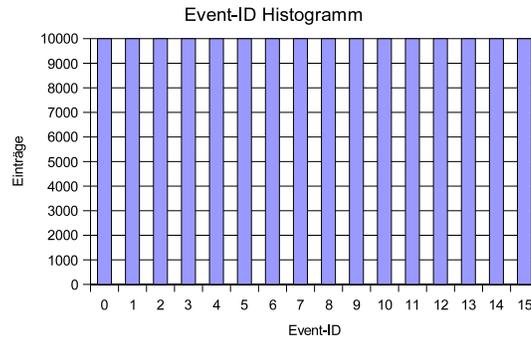


Abbildung 4.5: Event ID Histogramm. Während der Messung trat kein Fehler auf, weshalb alle 16 Bins die gleiche Anzahl an Einträgen enthalten.

einer Messung mit 160000 Triggern lieferte die Integritätsprüfung keine Fehlerwerte für den Event-ID-Zähler des OTIS. Das entsprechende Histogramm muß daher 10.000 Einträge in jedem Bin besitzen (Abbildung 4.5).

4.2.3 Der Latency-Tester

In diesem Teil wird die letzte Einheit der Datenintegritätsprüfung getestet. Die Monitor-Anschlüsse des OTIS zeigen die Länge der Latenzzeit über den Lese- und Schreibzeiger in Taktzyklen an (Siehe Abbildung 4.7). Die Nulldurchgänge der Zeiger wiederholen sich mit einer Zeitspanne von $4,1 \mu\text{s}$ oder alle 164 Taktzyklen.

Der Test wurde am OTIS-Chip mit dem Testaufbau aus Kapitel 5 durchgeführt.

Der Zustandsautomat des Latency-Testers ermittelt den Wert der Latenzzeit und ver-

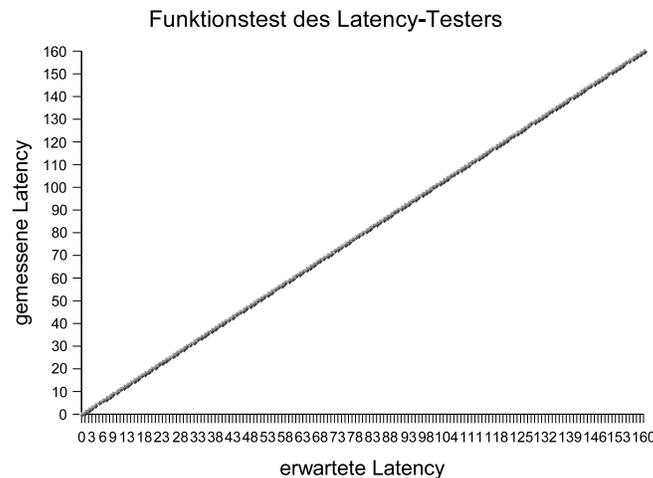


Abbildung 4.6: Funktionsüberprüfung des Latency-Testers auf dem FPGA. Alle am OTIS eingestellten Werte für die Latency konnten vom FPGA wieder richtig ausgegeben werden.

gleicht ihn mit einem Vorgabewert. Auf dem OTIS läßt sich die Latenzzeit per I²C einstellen. Der entsprechend erwartete Wert wird über das Konfigurationsregister dem FPGA-Design vom Anwender mitgeteilt. Abweichungen von Erwartungs- und gemessenen Wert werden mit einem Zähler aufgenommen. Zusätzlich wird der gemessene Wert vom FPGA ausgegeben. Die Latenzzeitwerte von 0 bis 160 wurden nacheinander am OTIS eingestellt und der zugehörige Ausgabewert des FPGA abgelesen. In Abbildung 4.6 ist der Verlauf der Messung gezeigt. Die vom OTIS ausgegebenen Latenzzeiten werden alle vom FPGA-Design richtig wiedergegeben. Stellt man die erwartete Latenzzeit bewußt einen Wert zu hoch oder zu niedrig ein, macht sich dies in der Anzeige der Errorbit-Zähler für die Latenzzeitmessung auf dem FPGA bemerkbar. Im Unterschied zu den anderen Errorbit-Zählern, werden Fehler mit einer konstanten Rate von 244 KHz aufgenommen¹. Damit ist dieser Teil der Messung unabhängig von der Triggerrate. Der Test startet und endet mit dem „Start Measurement“-Befehl, der gleichzeitig die Triggereinheit in der Fast-Control-Einheit des FPGA aktiviert.

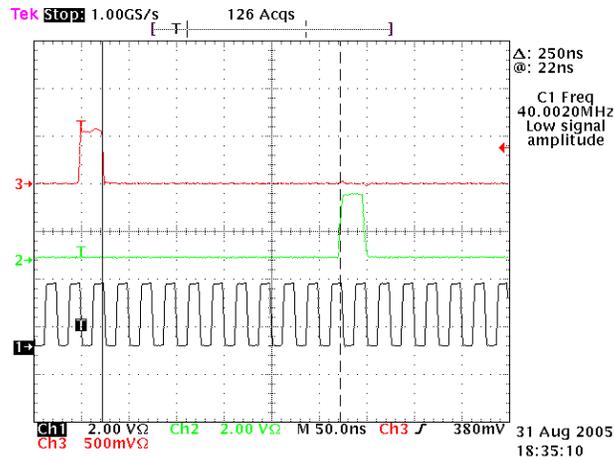


Abbildung 4.7: Nulldurchgänge der Schreib- und Lesezeiger im Pipeline-Register. Die Signale werden über die zwei Monitor-Pads des OTIS abgerufen. Kanal 1 stellt den 40 MHz-Takt des Last-Dummy-Out Ausgangs am OTIS dar. Der Abstand zwischen fallender Taktflanke des Lesezeigers (Kanal 3) und steigender Flanke des Schreibzeigers (Kanal 2) gibt die Latency in Taktzyklen an. Hier beträgt sie 10 Taktzyklen. Dieser Wert war gleichzeitig auf dem FPGA ablesbar

¹Das entspricht der Frequenz, mit der sich die Nulldurchgänge der Lese- und Schreibzeiger im Pipeline-Register wiederholen.

4.3 Der Histogrammer-Block

4.3.1 Der Driftzeithistogrammer

Nachdem die fertige Schaltung auf dem FPGA implementiert war, galt es sicherzustellen, daß die Histogrammierung die richtigen Daten für die einzelnen Zeitbins zurückgibt. Die Messungen wurden mit dem in Kapitel 5 beschriebenen Testaufbau am OTIS-Chip durchgeführt. Zunächst wurden die OTIS-Rohdaten über die direkte Datenausgabe des FPGA auf dem PC abgespeichert. Gleichzeitig wurden die Histogramm Daten aus dem FPGA übertragen. Das Vergleichshistogramm konnte man anschließend aus den Rohdaten mit Hilfe von Root-Skripten [17] erzeugen. Wie in Abbildung 4.8 zu sehen ist, stimmen beide Histogramme exakt überein. Sie wurden für Kanal 15 des OTIS aufgenommen. Insgesamt wurden hier zehn Million Trigger mit einer Rate von 1 MHz geschickt.

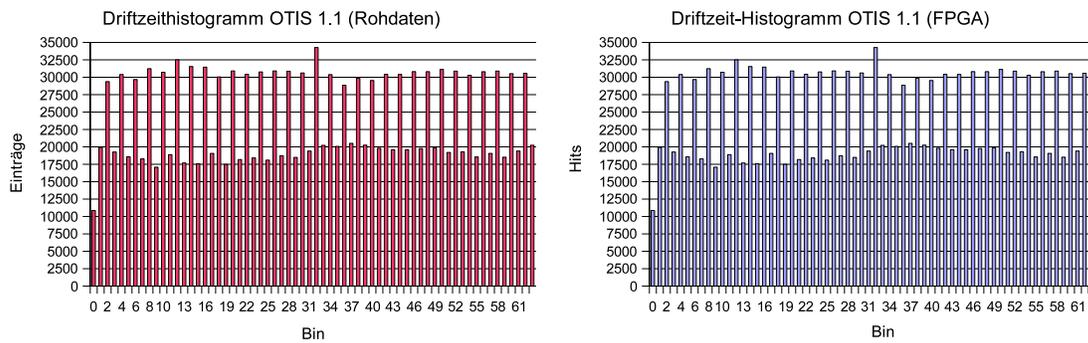


Abbildung 4.8: Vergleich der beiden Driftzeithistogramme aus den Rohdaten(links) und aus den FPGA-Daten(rechts) bei einer nominellen Digitalspannung von 2,57 V. Die Messungen wurden am OTIS 1.1 durchgeführt.

Die digitale Versorgungsspannungen am OTIS-Testboard wurde auf 2,57 V eingestellt. Die Analogspannungsversorgung zur Versorgung der DLL wurde auf dem Testboard von der Digitalspannung abgezweigt. Die Wahl der Betriebsspannungen ist wichtig, weil sie die Performance des Chips beeinflussen, wie spätere Messungen noch zeigen werden. Der Bereich, in dem die Kontrollspannung des OTIS liegt, zeigt an, ob sich die DLL fehlerhaft verhält. Sie befand sich bei dieser Messung mit einem Wert von 1,3 V innerhalb des Arbeitsbereichs des TDC.

Der Chip befand sich während der Messung im 3 BX-Auslesemodus. Bei der Bildung der Histogramme aus den 8 Bit Driftzeiten, wurden die zwei MSBs (Most Significant Bits) nicht berücksichtigt. Daher besitzten die Histogramme 64 Zeitbins, an Stelle der 192 Zeitbins, die man mit einem Meßintervall des OTIS von 75 ns maximal aufnehmen kann. Die zwei MSBs von jedem Driftzeit-Byte repräsentieren die Nummer des Bunch-Crossings innerhalb des Meßintervalls, in dem ein Ereignis stattgefunden hat. Diese Information ist für die Analyse des Zeitmeßverhaltens des OTIS nicht relevant. Der Driftzeit-Histogrammer nimmt die Ereignisse unabhängig von der Bunch-Crossing-Nummer auf, in der sie stattgefunden haben. Lediglich bei der Bereitstellung der Driftzeitdaten im vorgeschalteten Block spielen die beiden MSBs eine Rolle. Sie werden zur Entscheidung

herangezogen, ob ein Ereignis als Driftzeit ins Histogramm übernommen wird.

Bei dieser Messung betrug die Pulserrate ca. 2,1 MHz. Das Intervall, in dem Driftzeiten gemessen wurden war auf 75 ns eingestellt. Daraus läßt sich der Anteil der Einträge in einem Driftzeithistogramm an der Zahl der erzeugten Trigger ausrechnen zu:

$$n = \frac{\text{Meßintervall}}{\text{Pulserperiode}} = \frac{75 \text{ ns}}{476 \text{ ns}} = 0.158 \quad (4.1)$$

Mit der Zahl der gemessenen Histogramm-Einträge von ca. 1.572.000 erhält man einen Anteil von 0.157. Dementsprechend haben etwa 8.428.000 vom FPGA empfangene Datensätze keine gültige Driftzeit enthalten. Eine Driftzeit wird nur dann generiert, wenn innerhalb von 75 ns nach Beginn einer steigenden Triggerflanke ein Pulssignal auf dem Hiteingang des OTIS anliegt. Das gilt für den 3BX-Auslesemodus des OTIS, der hier eingestellt war. Im 1BX -bzw. 2BX - Modus liegen die Zeiten bei 25 und 50 ns. Das Ergebnis „kein Treffer“ wird in den Daten als Hexadezimalwert C0 angezeigt². Möchte man bei gleicher Triggerrate mehr gültige Treffer zurückbekommen, muß man die Pulserrate erhöhen.

4.3.2 Der Channel-Map-Histogrammer

Das Channel-Map Histogramm soll zeigen, daß nur die am OTIS gepulsten Kanäle Driftzeiteinträge erhalten. Da der OTIS 32 Kanäle besitzt, hat ein Channel-Map Histogramm 32 Bins.

In einem Datensatz folgt die 8 Bit Driftzeit des nullten Kanals nach dem vier Byte großen Datenkopf. Daran schließen sich in aufsteigender Reihenfolge die Driftzeiten der restlichen 31 Kanäle byteweise an.

Mit Hilfe des OTIS-Dummy läßt sich ein kompletter Datensatz beliebig konstruieren. So kann man für jeden Kanal innerhalb eines Datensatzes vorgeben, ob er eine gültige Driftzeit oder keinen Treffer ausgeben soll. Abbildung 4.9 zeigt, daß immer genau in denjenigen Kanälen Treffer gezählt wurden, die zuvor im OTIS-Dummy mit gültigen Driftzeiten versehen wurden. Die Zahl der Treffer in jedem Bin des Histogramms entsprach genau der Zahl der gesendeten Trigger.

Mit dem Aufbau in Abschnitt 5, konnte man die Funktionalität des Channel-Map-Histogrammers an einem OTIS-Chip überprüfen. Dieser Aufbau ermöglicht, entweder alle ungeraden oder alle geraden Kanäle eines OTIS-Chips gleichzeitig mit Pulsen zu versorgen. Das Umschalten zwischen beiden Einstellungen ist mit Hilfe des FPGA-Designs steuerbar. Zu diesem Zweck existieren zwei Befehle im Befehlsregister. Der OTIS wurde über die Slow-Control so programmiert, daß alle Kanäle für den Datenempfang aktiviert waren. In der ersten Messung (Abbildung 4.10) wurden die Signale des Pulsers über den FPGA auf alle geraden Kanäle umgeleitet, in der zweiten (Abbildung 4.11) auf alle ungeraden. Die resultierenden Histogramme zeigten die erwartete Verteilung der Einträge. Getriggert wurde mit 1 MHz, während die Eingangskanäle des OTIS mit Pulsen im

²Hexadezimal C0 entspricht dem Binärwert 11000000, wobei die beiden MSB (Most Significant Bits) die Driftzeit als „kein Treffer“ kodieren

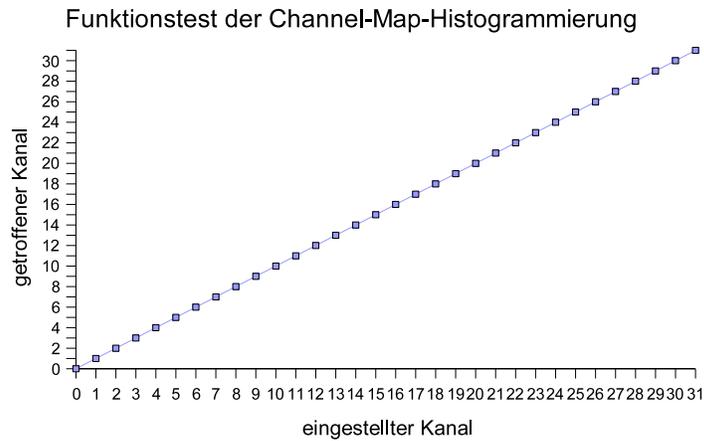


Abbildung 4.9: Test der Funktionalität des Channel-Map-Histogrammers mit dem OTIS-Dummy. Alle am Dummy mit einer Driftzeit belegten Kanäle zeigten in den Histogrammen an den richtigen Stellen die erwartete Anzahl an Einträgen.

8 MHz-Takt versorgt wurden. Die Fast-Control auf dem FPGA befand sich im Triggermodus und generierte 100.000 Trigger.

Die Bin-Einträge zeigen für beide Histogramme Werte von 62.000 ± 150 . Man sieht, daß es in beiden Histogrammen keine Treffer in den Kanälen gab, die nicht mit Pulssignalen versehen wurden. Die geringe Variation der Einträge in den getroffenen Kanälen werden durch statistische Schwankungen verursacht. Sie sind darauf zurückzuführen, daß die Pulssignale unterschiedliche Wegstrecken auf den verschiedenen Platinen zurücklegen müssen, bis sie an den Eingängen des OTIS eintreffen.

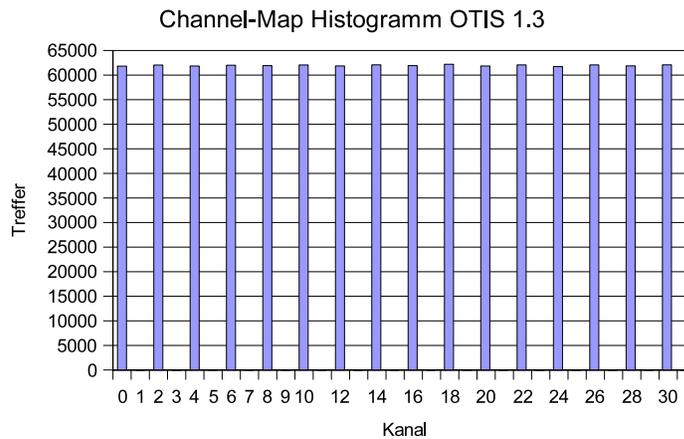


Abbildung 4.10: Channel Map Histogramm für die geraden Kanäle aufgenommen mit dem OTIS 1.3

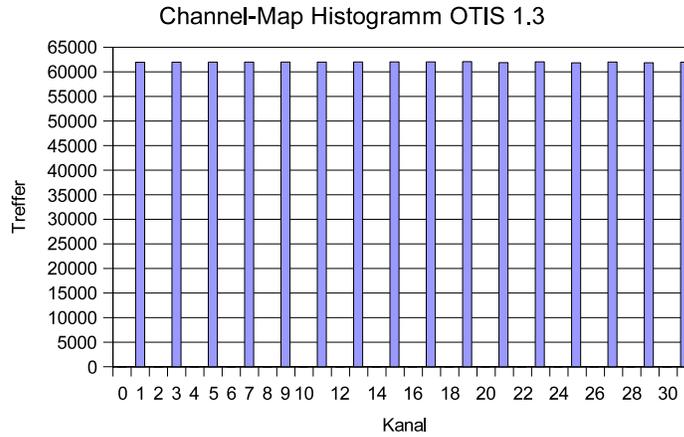


Abbildung 4.11: Channel Map Histogramm für die ungeraden Kanäle

4.4 Die Datenprozessierung

4.4.1 Das Banyan-Netzwerk

In der Datenprozessierungseinheit sind mehrere Blöcke integriert, deren korrekte Arbeitsweise noch nachgewiesen werden mußte. Insbesondere wurde überprüft, ob das Banyan-Netzwerk die richtigen Rückgabewerte für das Maximum und Minimum eines Driftzeithistogramms liefern konnte. Diese Funktionalität ist wichtig für den Serientest. Liegt das Minimum nach einer Messung in irgendeinem Driftzeitbin auf null, wird eine Fehlermeldung erzeugt.

Um die Funktionalität zu prüfen, wurden zunächst mehrere Histogramme bei verschie-

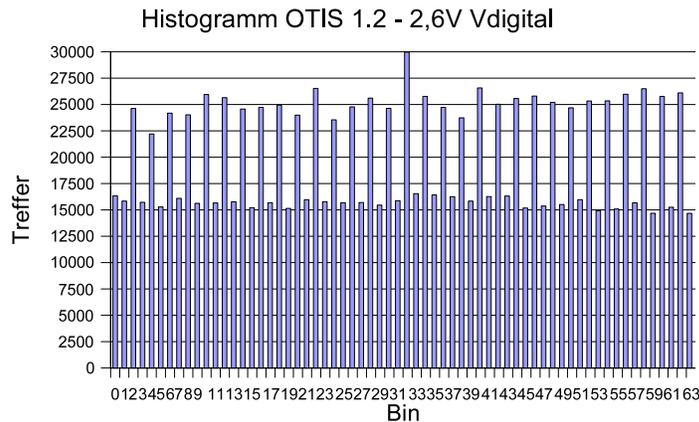


Abbildung 4.12: Histogramm des OTIS 1.2 aus der Meßreihe zur „Überprüfung des Banyan-Netzwerks. Maximalwert und Minimalwert aller Einträge in den Zeitbins stimmen mit den vom FPGA bestimmten überein.

denen Versorgungsspannungen des OTIS 1.2 mit dem FPGA aufgenommen. Abbildung 4.12 zeigt ein Histogramm aus der Meßreihe, das bei einer Digitalspannung von 2,6 Volt

aufgenommen wurde. Die Spannungsversorgung für den Analogteil des Chips wurde wiederum auf der Testkarte von der Digitalspannung abgezweigt.

Es wurden fünf Millionen Trigger geschickt und für alle Spannungen wieder fünf Millionen Datensätze empfangen. Getriggert wurde mit einem Megahertz, bei einer Pulserrate von 3,85 MHz. Alle Histogramme wurden für Kanal 15 des OTIS aufgenommen.

Die Maximum befindet sich in diesem und in allen übrigen Histogrammen im 32. Zeitbin. Dieser Effekt wurde bereits in [15] beschrieben. Es handelt sich um eine entwurfsbedingte Eigenschaft des Chips, die mit der Nachfolgendversion des OTIS 1.3 behoben wurde. Sie betrifft ausschließlich die Kanäle 0 bis 15. Für die Messung der Driftzeiten spielt dieser Effekt beim OTIS 1.2 jedoch keine Rolle.

Ausserdem ist in den Histogrammen dieser Meßreihe eine leichte Asymmetrie der Einträge zwischen geraden und ungeraden Zeitbins erkennbar. Dieses ist ein bereits bekanntes Merkmal des OTIS 1.2. Wie im folgenden Kapitel gezeigt wird, läßt sich diese Asymmetrie durch die richtige Wahl der analogen und der digitalen Versorgungsspannung ausgleichen.

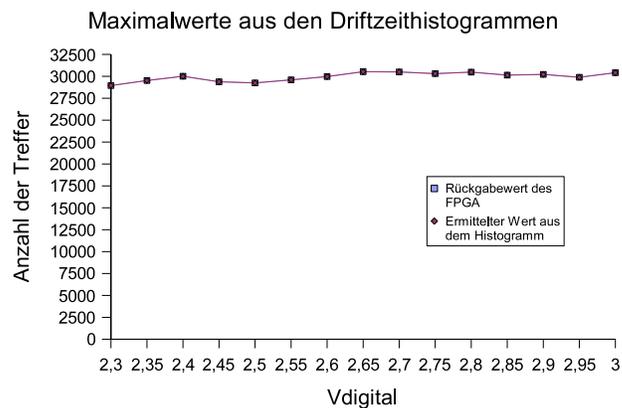


Abbildung 4.13: Messung zum Funktionstest des Banyan-Netzwerkes für die Maxima aus den Driftzeithistogrammen. Die Messungen wurden mit dem OTIS 1.2 durchgeführt.

Zusätzlich zu den Histogrammen selbst wurden die direkten Ausgabewerte des FPGA für die maximalen- und die minimalen Histogrammeinträge festgehalten. Zur Kontrolle konnte man die größten und kleinsten Zählwerte aus den Histogrammen ablesen, die vom FPGA erstellt wurden. Ein Vergleich der Ergebnisse mit den Ausgabewerten des Banyan-Netzwerkes auf dem FPGA zeigte eine völlige Übereinstimmung (Abbildungen 4.13 und 4.14).

Man erkennt aus den beiden Graphen, daß die Werte für die unterschiedlichen Versorgungsspannungen nur innerhalb der statistischen Schwankungen variieren.

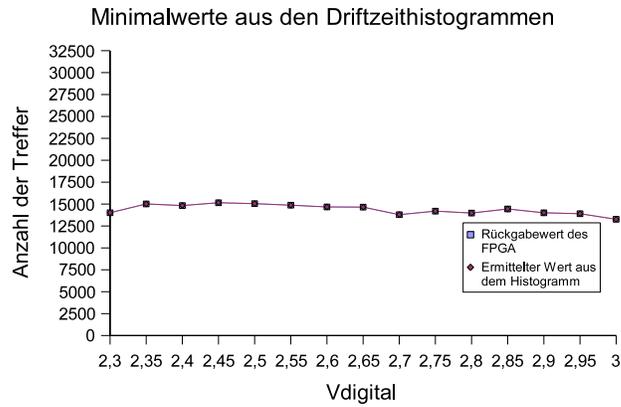


Abbildung 4.14: Messung zum Funktionstest des Banyan-Netzwerkes für die Minima aus den Driftzeithistogrammen mit dem OTIS 1.2.

4.4.2 Die DNL-Berechnung

Als letztes ist die korrekte Arbeitsweise der FPGA-Einheit zur DNL-Berechnung nachzuweisen.

Dazu wurden die identischen Meßreihen verwendet, mit denen schon das Banyan-Netzwerk getestet wurde. Aus diesen Histogrammen berechnet der FPGA den Absolutwert der DNL nach den Formeln in Abschnitt 3.2.6

Die Normierung des auf dem FPGA berechneten DNL-Wertes auf die durchschnittliche Zahl der Einträge pro Zeitbin wird von einem Softwareprogramm vollzogen. Eine Division stellt für eine Schaltung eine sehr komplexe Rechenoperation dar und verbraucht relativ viel Ressourcen. Daher wurde dieser Schritt nicht auf dem FPGA implementiert. Ein Vergleich der so erhaltenen DNL-Werte für die Histogramme dieser Meßreihe mit den Ausgabewerten des FPGA zeigte keine Abweichungen (Abbildung 4.15).

Die gemessenen DNL-Werte befinden sich für die Bereiche 2,3 Volt bis 3,0 Volt zwischen 1,2 und 1,4 Bins. Der Verlauf ist nahezu konstant.

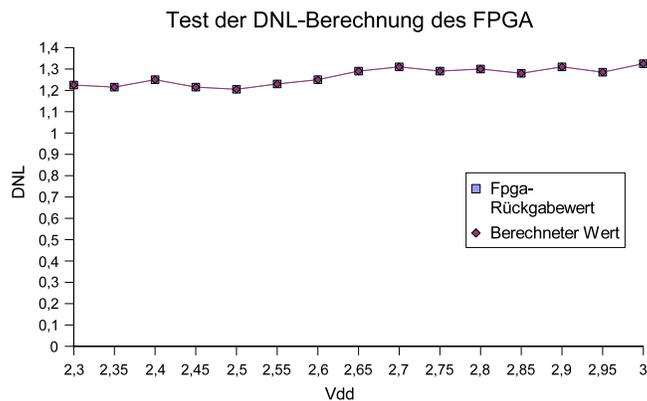


Abbildung 4.15: Funktionaler Test der DNL-Berechnung auf dem FPGA

Kapitel 5

Betriebspunktstudien zum OTIS

5.1 2D-Spannungsscan

In den bisherigen Messungen mit der OTIS-Testkarte waren die Digitalspannungen und die Analogspannungen des OTIS mit einem konstanten Abstand zueinander eingestellt. Die Analogspannung wurde nicht separat angeschlossen, sondern auf der Testkarte von der anliegenden Digitalspannung abgezweigt.

Vergleichsmessungen mit dem im ASIC-Labor vorhandenen OTIS-Teststand¹ zeigten unterschiedliche Zeitmeßverhalten des OTIS, sobald man den Chip mit zwei getrennt anliegenden Spannungen versorgte und den Abstand zwischen diesen Spannungen variierte.

Aus diesem Grund wurde der Chip mit verschiedenen Kombinationen von Digital- und Analogspannungen vermessen. Ziel war es, einen optimalen Arbeitspunkt für den OTIS-TDC festzulegen.

5.1.1 Testaufbau

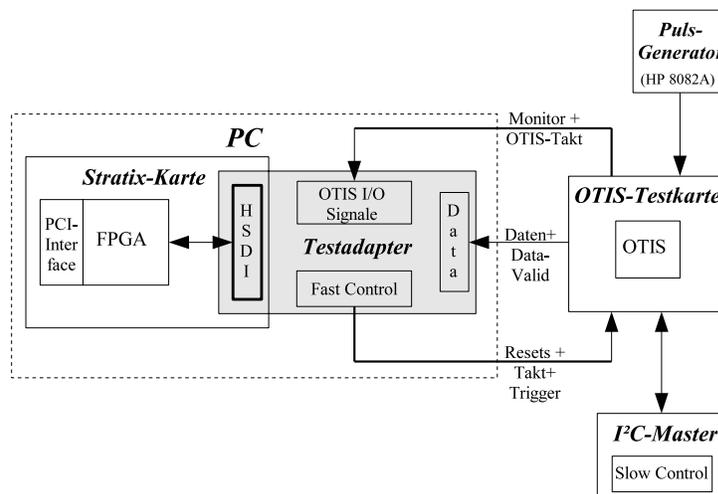


Abbildung 5.1: Aufbau der Testumgebung

¹Dieser Aufbau ist mit einem Logic-Analyser realisiert [15].

Im Unterschied zum Serientestaufbau sind hier einige Komponenten, wie z.B. die Nadelkarte oder der Nadelkartenadapter, noch nicht integriert. Die Stratix-PCI-Karte ist zusammen mit dem in dieser Diplomarbeit entworfenen Testadapterkarte AS30 in einem PC untergebracht. Die Testadapterkarte verbindet Ein- und Ausgabesignale des FPGA über einen 160-poligen HSDI-Anschluß (High Speed Differential Interface) mit der OTIS Testkarte V1². Der HSDI-Anschluß unterstützt differentielle bidirektionale 16 Bit Schnittstellen und erlaubt einen maximalen Datentransfer von 840 Mbits/Sekunde.

| LVTTL | | | | | | | |
|----------------------------------|-----------------|-------|------|-------------------------|------|-------|---------|
| | High-Pegel | | | Low-Pegel | | | Einheit |
| | Min. | Max. | Typ. | Min. | Max. | Typ. | |
| Eingang | 1,7 | 4,1 | - | -0,5 | 0,7 | - | V |
| Ausgang | 2,4 | - | - | - | 0,45 | - | V |
| LVDS (100 Ω Terminierung) | | | | | | | |
| | Offset Spannung | | | Differentielle Spannung | | | Einheit |
| | Min. | Max. | Typ. | Min. | Max. | Typ. | |
| Eingang | 0,1 | 1,1 | - | 0,3 | 1,0 | - | V |
| Ausgang | 1,125 | 1,375 | 1,2 | 0,25 | 0,55 | 0,375 | V |

Tabelle 5.1: Spezifikation der FPGA Signalpegel

| I ² C | | | | | | | |
|---------------------------------|----------------|------|------|----------------------|------|------|------|
| | logic 0 | | | logic 1 | | | Unit |
| | Min. | Max. | Typ. | Min. | Max. | Typ. | |
| input | -0.7 | 1.1 | 0.0 | 1.5 | 7.0 | 2.5 | V |
| output | — | — | 0.0 | — | — | 2.5 | V |
| CMOS | | | | | | | |
| | logic 0 | | | logic 1 | | | Unit |
| | Min. | Max. | Typ. | Min. | Max. | Typ. | |
| input | -0.7 | 1.1 | 0.0 | 1.4 | 3.3 | 2.5 | V |
| output | — | — | 0.0 | — | — | 2.5 | V |
| LVDS (100 Ω termination) | | | | | | | |
| | offset voltage | | | differential voltage | | | Unit |
| | Min. | Max. | Typ. | Min. | Max. | Typ. | |
| input | 0.0 | 2.5 | 1.2 | 0.1 | 2.5 | 0.2 | V |
| output | — | — | 1.02 | — | — | 1.38 | V |

Tabelle 5.2: Spezifikation der OTIS Signalpegel

Signalpegel Am FPGA sind die Signaltypen für alle I/O-Pins frei einstellbar. In diese Arbeit wurden LVTTL-(Low Voltage Transistor Transistor Logik) und LVDS-Signale verwendet (Low Voltage Differential Signal). Die Signalpegel stimmen mit den Spezifikationen des OTIS überein. In Tabelle 5.1 sind die Pegel der verwendeten FPGA-Signale

²Für Messungen mit dem OTIS 1.2 wurde die Testkarte durch die neuere Version V2 ersetzt.

dargestellt. Auf der OTIS-Testkarte sind alle Signale des TDCs über Stecker abgreifbar. Der OTIS unterstützt 3 verschiedene Typen von I/O-Anschlüssen: I²C, CMOS (Complementary Metal Oxide Semiconductor) und LVDS. Die Signalpegel für diese Anschlüsse sind in Tabelle 5.2 aufgelistet.

Signalerzeugung und Datenempfang Es werden einerseits die 8 Bit Daten des OTIS, sowie die in Abbildung 5.13 unter OTIS I/O zusammengefaßten Signale zum FPGA übertragen. Zur letzten Gruppe gehören die Monitorsignale des OTIS und das Last Dummy Out³. Später wurden bei der Erweiterung des Testaufbaus zusätzliche Steuersignale für den Nadelkartenadapter über diesen Stecker ausgegeben. Die bisher erwähnten Signale gehören zum Typ Low Voltage TTL. Die Fast Control verwendet LVDS-Signale für die Übertragung. Sie sind mit den folgenden Funktionen belegt:

- alle Resets: PowerUp-Reset, DLL-Reset, L0-Reset, BunchCrossingID-Reset, EventID-Reset
- Trigger mit konfigurierbarer Frequenz
- 40 MHz Systemtakt für die Bereitstellung der Arbeitsfrequenz des OTIS

Zur Steuerung der Slow-Control ist wiederum ein Laptop mit der I²C-Steuerung angeschlossen. Die zufälligen Treffer generiert ein externer Pulsgenerator. Da er asynchron zur Arbeitsfrequenz des OTIS läuft ist sichergestellt, daß die Treffer auf alle Zeitbins des OTIS statistisch gleichverteilt werden.

5.1.2 Durchführung der Messung

Um zu untersuchen, wie sich die Asymetrie der OTIS-Zeitbins in den Histogrammen mit der Variation von $V_{digital}$ und V_{analog} ändert, wurde mit diesem Meßaufbau ein Scan über beide Versorgungsspannungen durchgeführt. Die Triggerfrequenz betrug 1 MHz, wobei 1 Million Trigger gesendet wurden. Der OTIS arbeitete im 3 BX-Auslesemodus, während der Pulser Signale mit einer Frequenz von 4,2 MHz bereitstellte. Nach Formel 4.1 wird bei diesen Einstellungen mit jedem dritten Trigger eine Driftzeit am OTIS erzeugt. Die Spannungen $V_{digital}$ und V_{analog} wurden direkt am Chip gemessen. In den Abbildungen 5.2 und 5.3 sind die aufgenommenen Histogramme dargestellt. Von links nach rechts variiert die Analogspannung um 200 mV, bei konstanter Digitalspannung. Von oben nach unten bleibt die Analogspannung konstant, während die Digitalspannung in 200 mV-Schritten abfällt.

Verlauf der Kontrollspannung

Zunächst wird der Verlauf der gemessenen Kontrollspannung für alle Messungen betrachtet. Abweichend hohe Werte der Kontrollspannung sind ein Indiz dafür, daß die DLL nicht korrekt arbeitet. In diesem Fall lassen sich keine Schlüsse aus einem aufgenommenen Dirftzeithistogrammen ziehen. In Abbildung 5.4 sind alle Werte aufgetragen.

³Das Last Dummy Out stellt den OTIS-Takt bereit.

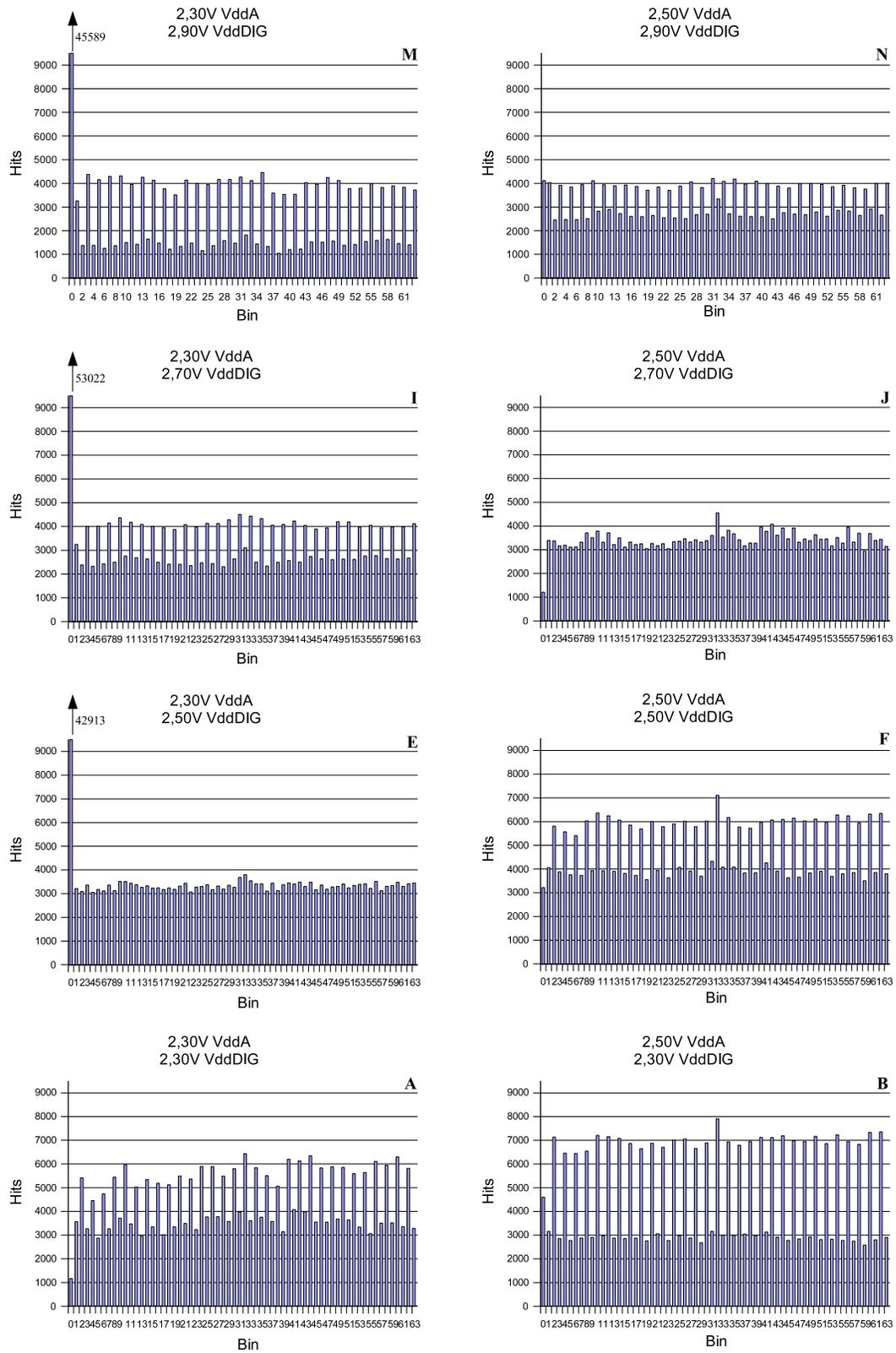


Abbildung 5.2: 2D-Spannungsscan am OTIS 1.2 (erster Teil)

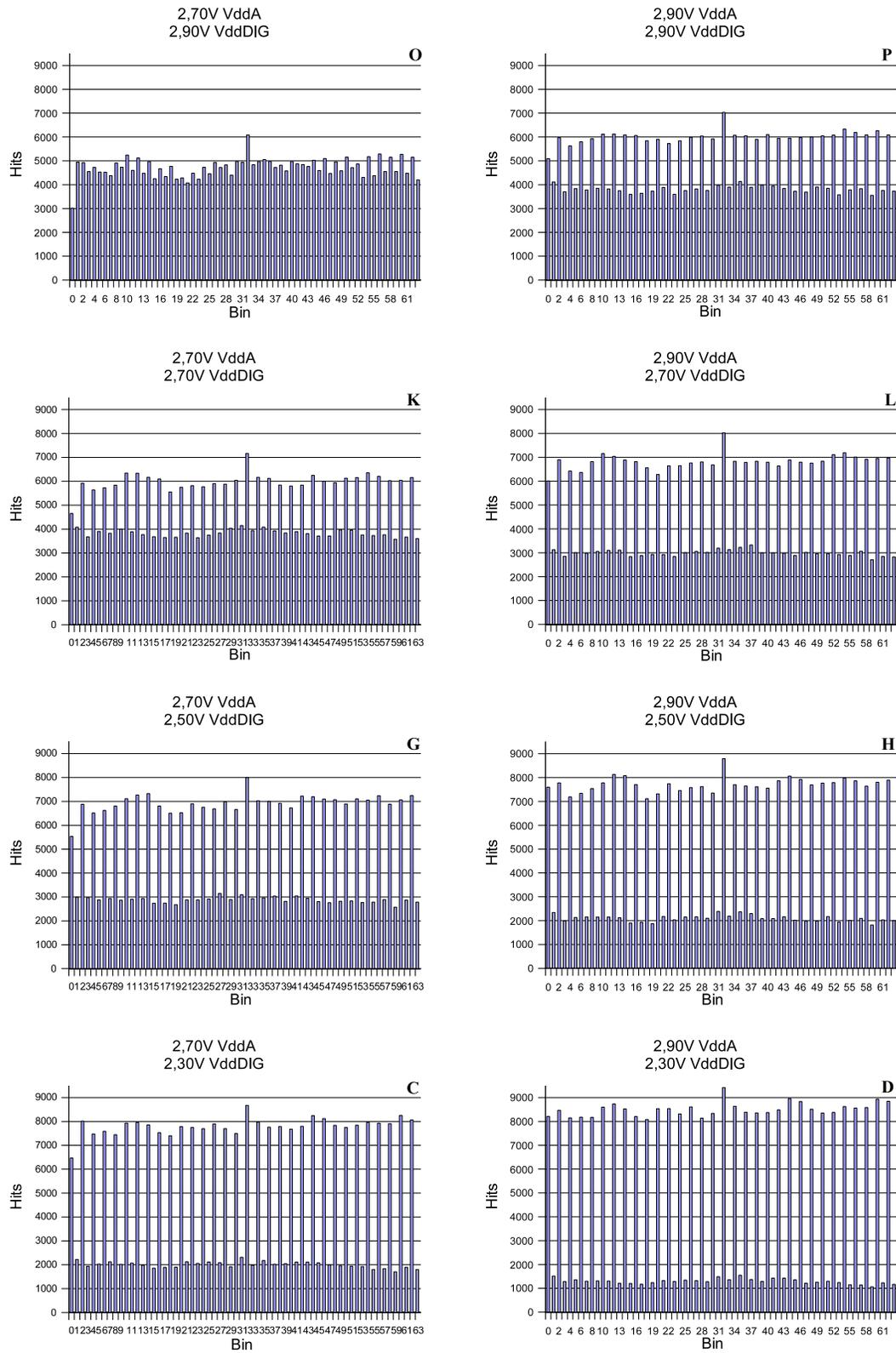


Abbildung 5.3: 2D-Spannungsscan am OTIS 1.2 (zweiter Teil)

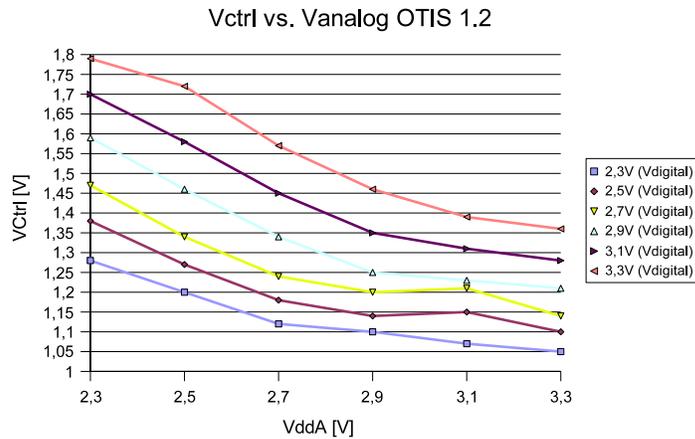


Abbildung 5.4: Verlauf von Vctrl

Der Verlauf ist stetig und läßt keine Sprünge der Kontrollspannung erkennen. Die Spannungen fallen monoton mit zunehmender Analogspannung. Das ist dadurch begründet, daß die DLL für hohe Versorgungsspannungen schneller als bei niedrigen arbeitet. Daher muß mit steigenden Werten für die Analogspannung die Gesamtverzögerung der DLL durch eine absteigende Kontrollspannungen ausgeglichen werden.

Der Verlauf der Asymmetrie in den Histogrammen

Die Histogramme F und O aus der Meßreihe wurden aufgenommen für die Betriebsspannungen $2.5\text{ V } V_{digital}$ und $2.5\text{ V } V_{analog}$ bzw. $2.9\text{ V } V_{digital}$ und $2.7\text{ V } V_{analog}$. Die erste Messung mit identischen Spannungen zeigt eine Asymmetrie zwischen den Einträgen in den geraden und den ungeraden Zeitbins. Die Einträge in den geraden Bins sind um ca. $\frac{1}{3}$ höher als diejenigen für die ungeraden. Dieses Verhalten des Chips ist bekannt und wurde bewußt in Kauf genommen. Die Auswirkungen auf die Driftzeitmessungen sind nur von geringer Bedeutung, da sich der Chip mit seinem Auflösungsvermögen weit unter dem geforderten Mindestwert von 1 ns bewegt.

In Bild 5.5 sind die DNL-Werte für jedes Zeitbin von Histogramm F aufgetragen. Die maximale negative Abweichung befindet sich zwischen Bin 31 und Bin 32 und beträgt -0.57 Bins. Die maximale positive Abweichung liegt bei 0.62 Bins und rührt von der Differenz der Einträge zwischen dem 32. und dem 33. Zeitbin her. Das ergibt eine DNL von 1.19 Bins bzw. 464 ps.

Das sich die Maximalwerte genau an diesen beiden Positionen anordnen, ist in der Konstruktionsart des Chips begründet. Für die ersten 16 Kanäle weicht das Routing in der DLL zwischen Inverter 31 und 32 leicht ab. Zwischen ihnen ist der Signallaufweg länger als zwischen den restlichen Invertern. Daher wird die DNL vom 32. Zeitbin bestimmt.

Histogramm O in Abbildung 5.3 zeigt, daß die Asymmetrie in den Einträgen bei geeigneter Wahl der Spannungen aufgehoben wird. Aus dem zugehörigen Graphen für die DNL pro Zeitbin (Abbildung 5.6) ist eine DNL von 0.67 Bins oder 261 ps ablesbar. Die

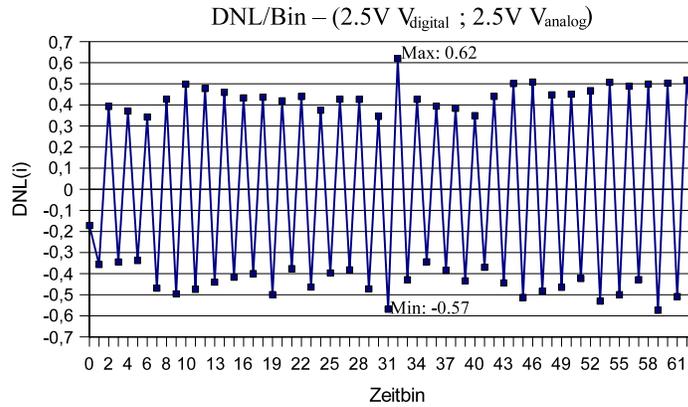


Abbildung 5.5: DNL pro Zeitbin für das linke Histogramm. Das 32. Zeitbin bestimmt den DNL-Wert. Er beträgt 1.19 Bins.

maximale positive Abweichung wird wiederum durch das 32. Zeitbin bestimmt, wobei die maximale negative Abweichung jetzt zwischen Bin 0 und Bin 1 auftritt.

Der Verlauf der Asymmetrie in den Driftzeitmessungen wird in Abbildung 5.7 verdeutlicht. Es ist eine gewisse Regelmäßigkeit zu erkennen, wenn man die Histogramme in den Bildern 5.2 und 5.3 von links nach rechts betrachtet. Mit steigender Analogspannung gewinnt auch die Asymmetrie ein stärkeres Gewicht. Für die oberste Reihe (Histogramm M bis P) nimmt die nach 5.1 berechnete Asymmetrie der Histogrammeinträge von ca. -17% auf 52% mit steigender Analogspannung und konstanter Digitalspannung von 2,9 Volt zu. Bei 2,7 Volt Analogspannung ist die Asymmetrie fast null.

Verfolgt man den Verlauf der Asymmetrie bei 2.7 V konstanter Analogspannung in der dritten Spalte in Abbildung 5.3 von oben nach unten (Histogramme O, K, G und C) stellt man ein entgegengesetztes Verhalten fest. Die Digitalspannung fällt in 200 mV-Schritten

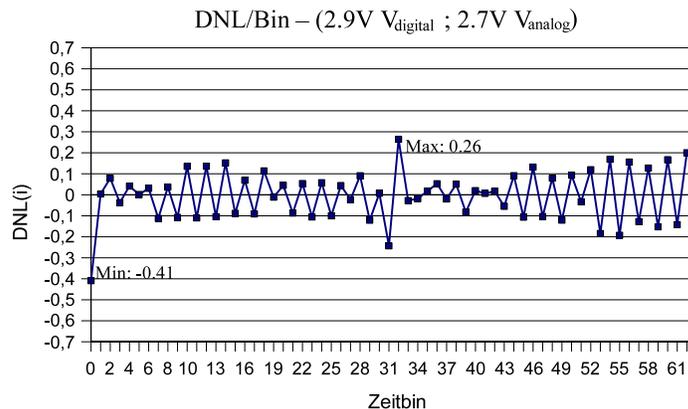


Abbildung 5.6: DNL pro Zeitbin bei 2.7 V Analog -bzw. 2.9 V Digitalspannung. Die DNL beträgt 0.67 Bins.

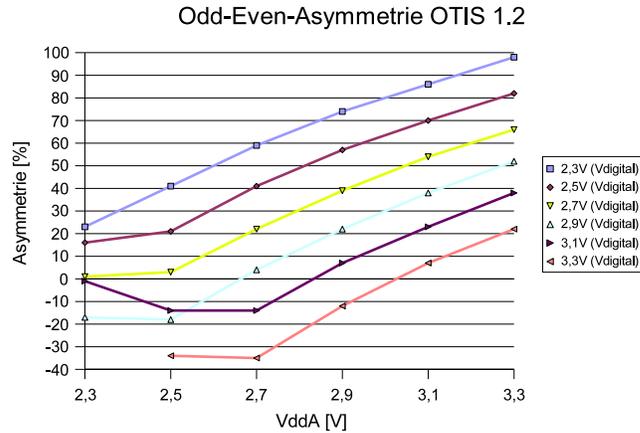


Abbildung 5.7: Odd-Even Asymmetrie des OTIS 1.2

ab, während die Asymmetrie gleichzeitig ansteigt. Bei 3.3 Volt Digitalspannung liegt zunächst eine Umkehr der Asymmetrie vor: die ungeraden Zeitbins sind um etwa 35% höher, als die geraden. Von hier an steigt die negative Asymmetrie an, bis sie bei 2,9 Volt $V_{digital}$ fast ganz verschwindet. Histogramm N in Abbildung 4.15 stellt ein Beispiel für eine negative Asymmetrie dar.

Die Asymmetrie in den Driftzeitbins wird mit dem Verhalten der Inverter bei unterschiedlichen Versorgungsspannungen begründet. Die Schwelle, ab der ein Inverter zwischen zwei Zuständen umschaltet varriert mit anliegender Analogspannung. Dies führt zu den unterschiedlichen Binlängen der DLL-Kette. Der Effekt wurde bei der Konstruktion des Chips berücksichtigt und wird an dieser Stelle nicht weiter erläutert.

$$\text{Asymmetrie} = \frac{\sum_{n=0}^{31} \text{Einträge in Bin}(2n) - \sum_{n=1}^{32} \text{Einträge in Bin}(2n-1)}{\sum_{n=0}^{63} \text{Einträge in Bin}(n)} \quad (5.1)$$

Die geringsten Asymmetrien sind im Graph 5.7 an den Stellen sichtbar, an denen die Digitalspannung 200 mV über der Analogspannung liegt. Dieses Kriterium ist wichtig für die Bestimmung eines optimalen Arbeitspunktes des OTIS. Wie sich gezeigt hat, wird bei diesen Arbeitspunkten die Asymmetrie zwischen geraden und ungeraden Zeitbins vollständig aufgehoben, so daß sie bei den Driftzeitmessungen nicht mehr ins Gewicht fällt.

Verlauf der Abweichung des nullten Zeitbins

In den Histogrammen ist eine Abweichung des nullten Zeitbins zu erkennen. Es besitzt im Vergleich weniger Einträge als alle anderen Bins. Dieses Problem ist in der OTIS-Version 1.3 behoben. Der Graph in Abbildung 5.8 bestätigt den Verlauf aus [15]. Er zeigt die Änderung der Größe des Bin 0 für die unterste Reihe in Abbildungen 5.2 und 5.3 (Histogramme A bis D) bei einer konstanten Digitalspannung. Die Abweichung reduziert sich bei 2,9 V Analogspannung fast zu null. Sie wurde folgendermaßen berechnet:

$$\text{Abweichung} = \frac{\text{Einträge in Bin}(0) - (\sum_{n=1}^{31} \text{Einträge in Bin}(2n))/31}{(\sum_{n=1}^{63} \text{Einträge in Bin}(n))/63} \quad (5.2)$$

Für eine niedrige Analogspannung von 2.3 Volt beläuft sich der Wert auf etwas mehr als -1 Bin. Bei höheren Analogspannungen wachsen die Einträge im nullten Bin an, bis sie bei 2.9 V gleichauf mit den übrigen geraden Zeitbins sind. Bei 3.3 Volt ist das nullte Bin 0.45 Bins länger als die durchschnittliche Binsgröße des Histogramms.

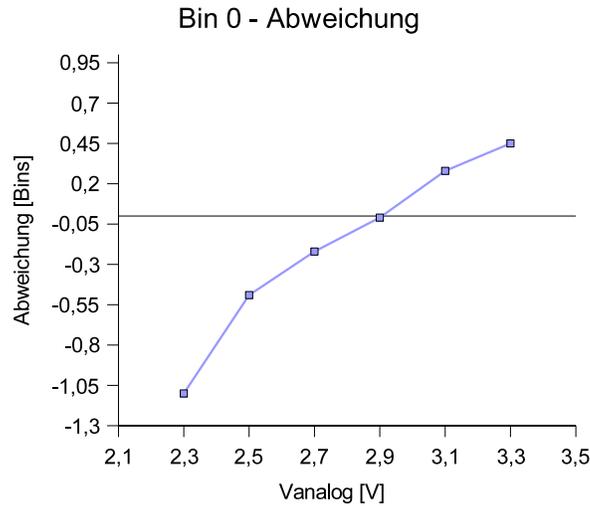


Abbildung 5.8: Bin 0-Abweichung bei konstanter Digitalspannung von 2,3 Volt.

Der Grund für dieses Verhalten liegt im Regelkreis der DLL des OTIS 1.2. Die DLL besteht aus spannungsabhängigen Verzögerungselementen, den Inverten. Innerhalb des Regelkreises der Inverterkette befindet sich ein Phasendetektor. Je nachdem ob die Phasendifferenz zwischen der DLL-Taktfrequenz und der von Aussen anliegenden Arbeitsfrequenz des OTIS positiv oder negativ ist, wird die Kontrollspannung (V_{ctrl}) an den Verzögerungselementen durch diesen Regelkreis erhöht oder erniedrigt. Dadurch verändert sich die Durchlaufzeit des Taktsignals durch die DLL-Kette, bis sie die optimalen Länge von 25 ns erreicht. Da der Phasendetektor als Proportionalregler agiert, ist die Regelspannung immer um einen Offsetwert verschoben. Die Größe der Abweichung variiert mit der anliegenden Analogspannung. Dies kann die Verzögerungszeit der DLL von 25 ns vergrößern oder verkleinern.

Für den Fall einer zu langen DLL überdeckt das 63. Zeitbin des Bunch-Crossings N Teile des nullten Zeitbins von Bunch-Crossing $N+1$ und macht es dadurch kleiner.

Im Extremfall kann der Überlapp so groß werden, daß das nullte Bin komplett und das erste Bin von Bunch-Crossing N teilweise überdeckt werden. Die der DLL nachgeschaltete Logik zur Speicherung einer Driftzeit wird in diesem speziellen Fall ein Ereignis im nullten Zeitbin vorgetäuscht, ein sog. „Ghost Hit“. Damit ist erklärbar, warum die Histogramme M, I und E in Bild 5.2 so hohe Zählwerte im nullten Bin haben. Diese

Histogramme liegen jedoch im Randbereich der Meßreihe, in dem der OTIS nicht betrieben wird. Daher hat dieser Effekt keine Auswirkungen auf sein Zeitmeßverhalten.

Mit dem OTIS 1.3 tritt dieses Effekt nicht mehr auf, da ein Phasendetektor mit integrierendem Regelverhalten implementiert wurde.

Verlauf der DNL

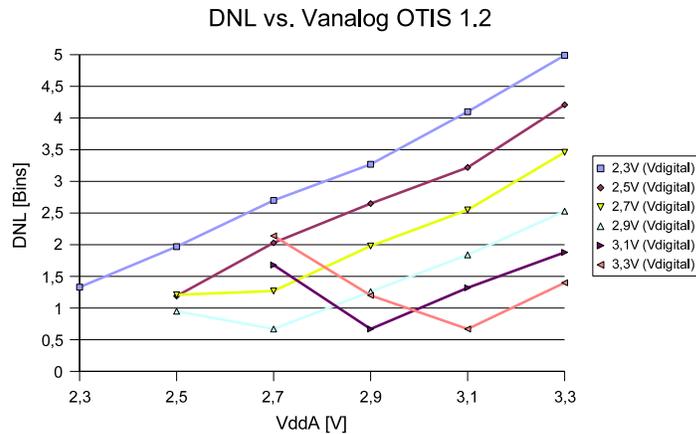


Abbildung 5.9: Verlauf der DNL

Die Graphen 5.9 bzw. 5.10 zeigen, welche Werte der Versorgungsspannungen eine optimale Zeitaufösung des OTIS-TDC garantieren. Generell liegen die Minima der DNL-Werte bei einem Abstand zwischen $V_{digital}$ und V_{analog} von 200 mV (siehe 5.10). Der Operationbereich von V_{analog} und $V_{digital}$ wird vom Herstellungsprozess bestimmt und liegt um

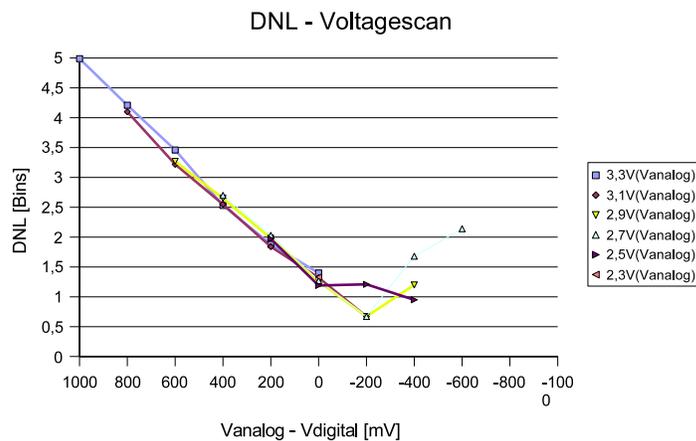


Abbildung 5.10: Abhängigkeit der DNL von der Abweichung zwischen $V_{digital}$ und V_{analog}

die 2.5 V. Dies ist allerdings nur eine Empfehlung, der Chip kann bei deutlich höheren Spannungen in Betrieb genommen werden. Messungen haben gezeigt, dass der Chip bis zu Spannungen von 3.5 V funktionierte.

Der optimale Bereich stellt demnach ein Wert um die 2,7 Volt für die Digitalspannung und 2,5 Volt für die Analogspannung dar. Hier nimmt die DNL einen Wert von 1,19 Bins an. Die besten Werte werden bei etwas höheren Spannungen erreicht. Beispielsweise liegt die DNL für Spannungen von 2,9 V ($V_{digital}$) und 2,7 Volt (V_{analog}) bei 0,67 Zeitbins oder 261 ps.

Graph 5.10 zeigt den Verlauf der DNL bei verschiedenen Abständen zwischen der Digital- und der Analogspannung. Die Minima liegen von allen Kurven bis auf einer bei -200 mV, wobei dieser Wert die Differenz zwischen V_{analog} und $V_{digital}$ angibt. Der Ausreißer für die Analogspannung von 2.5 V kann begründet werden, wenn man sich das zugehörige Histogramm (B) anschaut. Bei 2.5V V_{analog} und 2.3 V $V_{digital}$ ist die Asymmetrie noch so hoch, daß die DNL durch das 32. Zeitbin bestimmt wird. Das ist bei allen darüberliegenden Histogrammen nicht mehr der Fall.

5.2 Frequenzscan des OTIS

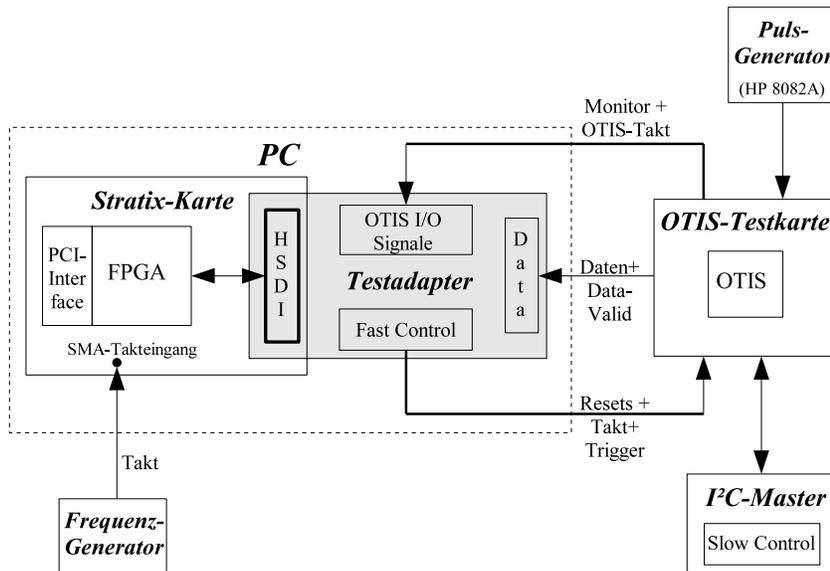


Abbildung 5.11: Aufbau der Testumgebung mit externem Frequenzgenerator

Hier wurde untersucht, in welchem Bereich der Taktfrequenz der OTIS noch korrekt arbeitet. Um diese Messung vornehmen zu können, wurde der Meßaufbau um einen externen Frequenzgenerator ergänzt. Mit seiner Hilfe war es möglich, die Arbeitsfrequenz des OTIS in kleinen Schritten zu variieren. Mit dem FPGA ist dies nicht möglich, da die auf ihm vorhandenen PLLs eine Variation der Taktfrequenz in Ein-Megahertz-Schritten nicht zulassen. In diesem Aufbau werden der Systemtakt des FPGA-Designs und des OTIS beide vom Frequenzgenerator getrieben. Das FPGA-Design ist bis ma-

ximal 65 MHz lauffähig, so daß man den OTIS bis an seine Leistungsgrenze von ca. 50 MHz hochtakten kann. Wie in dem obigen Aufbau erzeugte der Pulsgenerator zufälliger Treffersignale an einem Kanal des OTIS. Folgende Rückgabewerte des FPGA wurden anschließend betrachtet:

- Zahl der empfangenen Ereignisse
- Zahl der gültigen Treffer
- Zahl der Ereignisse, die keine Treffer mitlieferten
- Kontrollspannung der DLL
- Maximum und Minimum des Driftzeihistogramms
- Wert für die DNL

Die Digitalspannung und die Analogspannung des OTIS 1.2 waren zuerst beide auf 2,7 Volt und in einer zweiten Meßreihe auf 2,7 Volt für die Analogspannung und 2,5 Volt für die Digitalspannung eingestellt. Es wurden für jede Frequenz fünf Millionen Trigger geschickt. In Abbildung 5.12 ist die Frequenz gegen die Kontrollspannung aufgetragen. Korrekte Daten konnten für einen Bereich von 25 bis 44 MHz empfangen werden. Überhalb dieses Bereichs kamen nur noch Daten ohne Trefferereignisse zurück. In den Bereichen von 33 MHz bis 41 MHz wurden fünf Millionen Datensätze wieder vom FPGA empfangen. Ausserhalb dieses Bereichs wurde vom OTIS ein Großteil der Trigger abgelehnt. Der Bereich, in dem alle Zeitbins vom OTIS noch Treffer enthalten, ist etwas größer und reicht von 28 MHz bis 43 MHz. Zu erkennen ist dies an der Anzahl der Einträge für das Minimum des Histogramms, daß nicht null betragen darf. Im selben Bereich werden sinnvolle Werte für die DNL geliefert. Sie liegt je nach anliegender Frequenz zwischen 1,94 und 0,76 Bins. Der beste Wert für die DNL liegt bei einer Frequenz von 41 MHz. Bei der eigentlichen LHC-Frequenz ist der Wert von 0.9 Bins jedoch nur unwesentlich schlechter. Die Kontrollspannung stieg über den gesamten Frequenzbereich stetig und linear an.

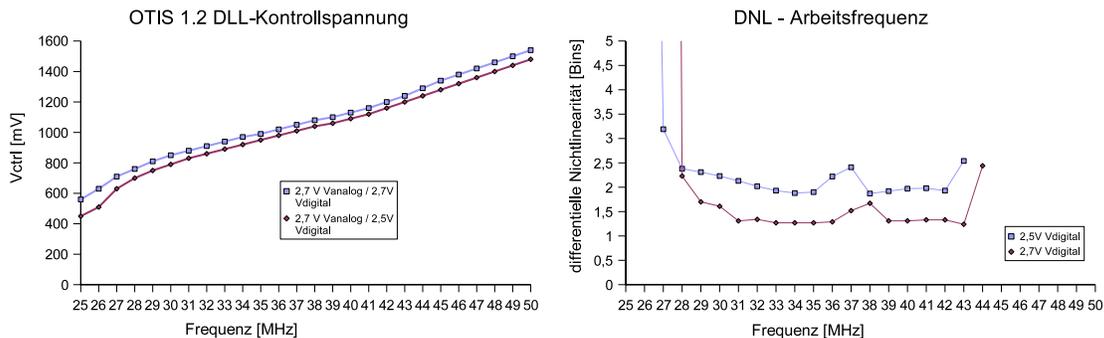


Abbildung 5.12: Verlauf von Vctrl und DNL bei Variation der Frequenz

5.3 Messungen am OTIS 1.3

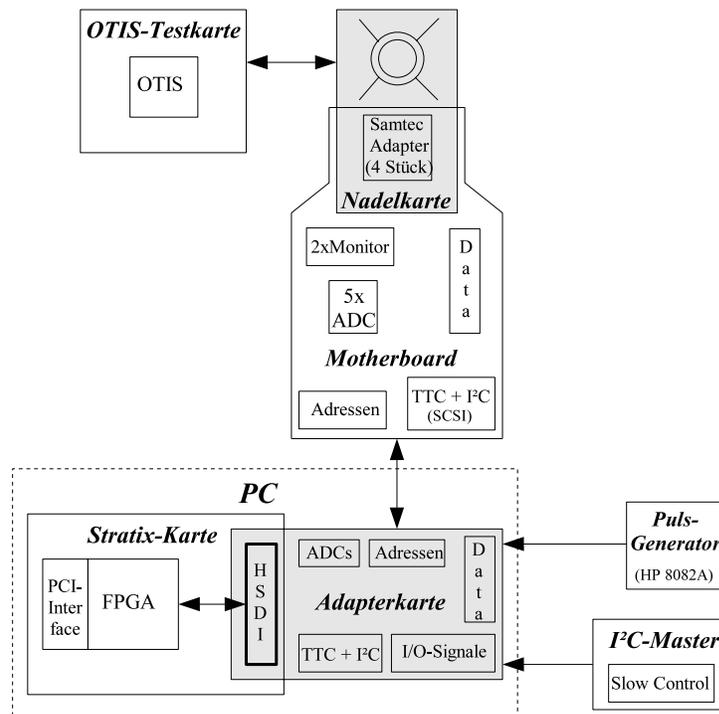


Abbildung 5.13: Aufbau der Testumgebung

Im gezeigten Testaufbau 5.13 ist erstmals die Nadelkarte integriert, mit der im Serientest die Chips auf dem Wafer kontaktiert werden. Die Verteilung der Signale von der Nadelkarte zu den übrigen Komponenten übernimmt das Motherboard. Mit diesem Testaufbau wird ermöglicht, alle geraden oder alle ungeraden Kanäle des OTIS gleichzeitig anzusprechen. Die Funktionalität des Channel-Map Histogrammers konnte unter anderem mit diesem Aufbau unter Beweis gestellt werden. Abbildung 5.14 zeigt eine erste Messung mit dem OTIS 1.3 der gegen Ende dieser Arbeit aus der Produktion zurückkam. Die Messung wurde bei 2.8 V Digitalspannung und 2.5 V Analogspannung für eine Million Trigger durchgeführt. Man erkennt deutlich, daß das Problem eines zu kleinen nullten Zeitbins nicht mehr existiert. Die DNL lag mit 2.98 Bins oder 116 ps auf einem vergleichsweise niedrigen Wert.

Abbildung 5.15 zeigt einen sog. Zebra-Plot, der mit diesem Aufbau durchgeführt werden konnte. Er gibt die Einträge aller Kanäle des OTIS in allen Zeitbins wieder. Aus der Breite der Balken läßt sich auf die Bin-Größen schließen. Diese bleiben für alle Kanäle nahezu konstant.

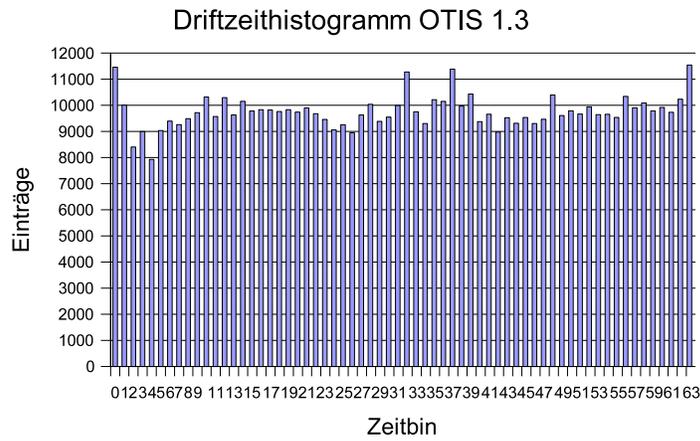


Abbildung 5.14: Histogramm des OTIS 1.3. Die DNL liegt bei 116 ps.

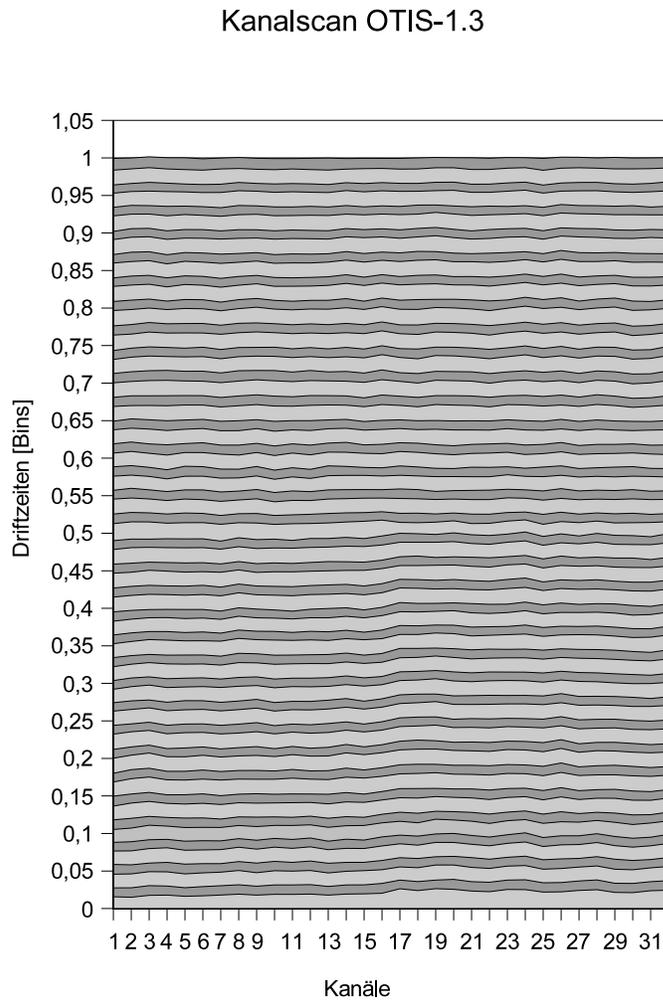


Abbildung 5.15: Größe der Zeitbins und Position bei Raumtemperatur

Kapitel 6

OTIS-Wafer-Test

6.1 Der Testaufbau im Reinraum

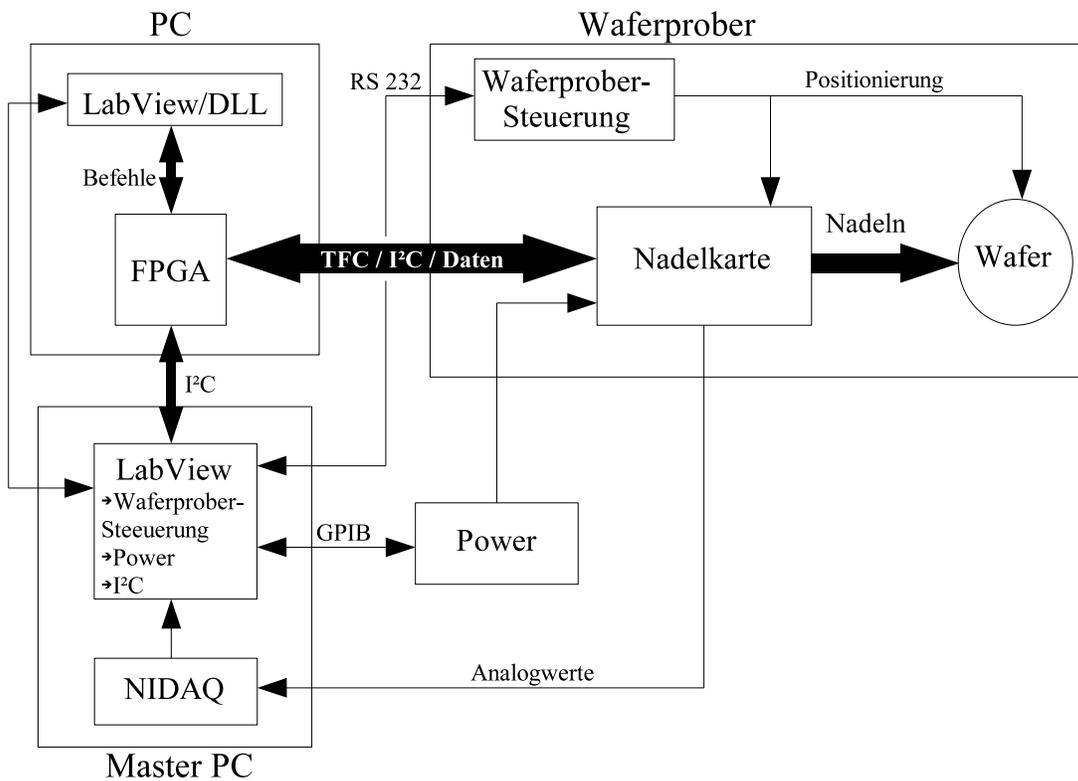


Abbildung 6.1: Schema vom Aufbau des Serientests im Reinraum

6.2 Überblick über den Testablauf

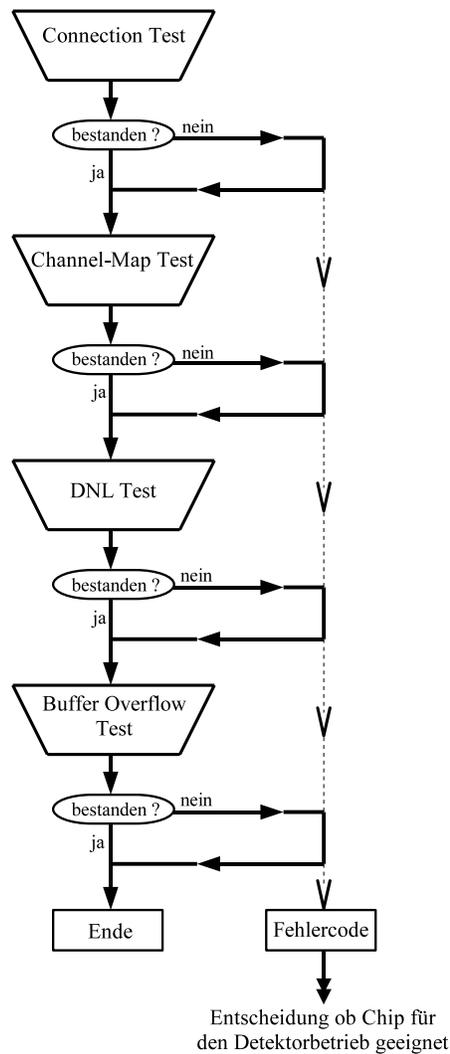


Abbildung 6.2: Ablauf der Serientestprozedur am Waferprober

Der Testablauf wird von einer LabView-Umgebung gesteuert. Labview nimmt die Programmierung der Slow-Control des OTIS und das Setzen der Konfigurationsregister des FPGA vor. Zusätzlich kann es Befehle aus dem Befehlsregister des FPGA aufrufen. Die Serientestprozedur besteht aus vier Stufen: dem General Connection Test, dem Channel-Map Test, dem DNL-Test und dem Buffer Overflow Test. Nach Absolvierung einer Stufe wird jedesmal ein Fehlercode erzeugt, der anzeigt, an welchen Stellen im Test Abweichungen vom erwarteten Verhalten des OTIS stattgefunden haben.

6.2.1 Der General Connection Test

In der ersten Stufe werden die digitalen Funktionen des OTIS-Chips getestet. Die Messung wird insgesamt zweimal mit verschiedenen Einstellungen wiederholt. Die beiden Wiederholungen unterscheiden sich darin, daß die 12 Bit Identifikationsnummer des OTIS variiert wird: im ersten Durchgang werden alle geraden Bits auf eins gesetzt, im zweiten alle ungeraden.

Der Ablauf der Prozedur ist in Tabelle 6.1 dargestellt. Zu Beginn werden die Konfigurationsregister des FPGA eingestellt. Hierzu zählen die Werte für die Triggerrate, die Länge der Resetsignale und die Zahl der zu generierenden Trigger. In diesem Fall wird mit 1 MHz getriggert, wobei die Zahl der Trigger 10.000 beträgt. Zusätzlich werden die Vergleichswerte für die zurückgelesenen Informationen aus dem OTIS-Datenstrom gesetzt. Dazu gehören alle Werte, die im Datenkopf des OTIS stehen, sowie die über die Monitor-Anschlüsse des OTIS gemessene Latenzzeit. Als letztes wird die erwartete Identifikationsnummer des OTIS eingestellt (vgl. Tabelle 6.2).

Der nächste Schritt im Testablauf beinhaltet das Zurücksetzen des OTIS mit dem PowerUp-Reset und dem Level0-Reset. Anschließend lassen sich die OTIS-Register per I²C programmieren. Die gewählten Einstellungen sind identisch mit den FPGA-Werten in Tabelle 6.2. Ob die Register wieder erfolgreich über I²C ausgelesen werden konnten, wird im nächsten Schritt überprüft. Nach Zurücksetzen des OTIS folgen nun die Befehle an die Fast-Control und an die Steuerung des Pulsers im FPGA. Im Triggermodus hält die Messung automatisch an, nachdem 10.000 Trigger gesendet wurden. Der Pulser bleibt ausgeschaltet, so daß keine Driftzeiten generiert werden.

Die Ergebnisse der beiden Testdurchläufe werden durch Fehlercodes in einer Datei abgespeichert. Folgende Bedingungen sind zum Bestehen der ersten Teststufe vorgegeben:

- alle Errorbit-Zähler stehen auf null. Diese Zähler registrieren das Fehlverhalten der OTIS-Funktionen, die in den Statusbits und über die Monitor-Signale wiedergegeben werden.
- die OTIS-Identifikationsnummer kann alle eingestellten Bitzustände richtig übernehmen.
- die Zahl der empfangenen Datensätze, die mit dem FPGA aufgenommen wird, entspricht der Zahl der gesendeten Trigger.
- die Driftzeiten aller Kanäle haben den Wert C0 angenommen, da der Pulser ausgeschaltet ist und folglich keine Treffer auf den OTIS gegeben werden.

6.2.2 Der Channel-Map Test

In diesem Teil wird die Channel-Map getestet. Es soll festgestellt werden, ob tatsächlich nur diejenigen Kanäle des OTIS Driftzeiten empfangen, die mit Pulsen versorgt werden. Auf dem FPGA ist für diesen Test der Channel-Map Histogrammer implementiert. Zunächst befinden sich alle Register auf dem FPGA und im OTIS-Chip in dem Zustand, der für die letzte Meßreihe des General Connection Test eingestellt wurde. Werte die geändert werden sollen werden im hiesigen Testablauf überschrieben. Das FPGA-Design

| | Aktion | Einstellungen 1.Messung | Einstellungen 2.Messung |
|-----|--|---|--|
| 1. | FPGA-Design Reset | | |
| 2. | Setze FPGA-Konfigurationsregister | siehe Tabelle 6.2 | |
| 3. | PowerUp Reset | Länge: 1 μ s | |
| 4. | Level0-Reset | Länge: 100 ns | |
| 5. | I ² C-Programmierung | Read Mode: Multi Hit Number of BX: 2 Truncation: ein Playback Mode: aus Latency: 675 ns | Read Mode: Single Hit Number of BX: 3 Truncation: aus Playback Mode: aus Latency: 675 ns |
| 6. | Zurücklesen der I ² C-Register des OTIS | | |
| 7. | Schreibe Fehlercode | | |
| 8. | Alle Resets | Level0-, Bx-Counter-, Event-Counter Reset jeweils 100 ns lang | |
| 9. | Pulserbetrieb | ausgeschaltet | |
| 10. | Setze Identifikationsnummer des OTIS | x"aaa" (alle geraden Bits an) | x"555" (alle ungeraden Bits an) |
| 11. | Fast-Control Betrieb | Triggermodus | |
| 12. | Starte Messung | | |
| 13. | Schreibe Fehlercode | | |

Tabelle 6.1: Ablauf des General Connection Tests

| FPGA-Konfigurationsregister | Einstellungen |
|---|--|
| Resetlänge | 100 ns |
| Triggerrate | 1 MHz |
| Zähler der Trigger | 10.000 |
| erwartetes Datenkopfmuster 1.Messung/2.Messung | Read Mode: Multi Hit / Single Hit Number of BX: 2 / 3 Truncation: ein / aus Playback Mode: aus / aus Buffer Overflow Bit: aus / aus Errorbit: aus / aus |
| erwartete Latency | 675 ns |
| erwartete Identifikationsnummer des OTIS | x"aaa" für Testlauf 1 x"555" für Testlauf 2 |

Tabelle 6.2: Setzten des Konfigurationsregisters des FPGA

Reset setzt lediglich alle Errorbit-Zähler zurück, die Einstellung in den Konfigurationsregistern bleiben jedoch bestehen.

Der Ablauf findet sich in Tabelle 6.3 wieder. Zwei Meßreihen werden aufgenommen, in denen zuerst die ungeraden Kanäle und beim zweiten Durchlauf die geraden Kanäle des OTIS Signale des Pulsers empfangen. Der Pulser arbeitet konstant mit einer Frequenz von ca. 8 MHz.

Zwei Bedingungen genügen zum bestehen des Channel-Map Tests:

- in den Channel-Map Histogrammen enthalten die nicht gepulsten Kanäle in beiden Fällen keine Einträge.
- die Zahl der Einträge in den getroffenen Kanälen liegt zwischen 60.000 und 65.000. Die Größenordnung, innerhalb der sich die Zahl der Einträge befinden muß folgt aus der Kombination der Frequenzen für die Triggerrate und die Pulserrate

| | Aktion | Einstellungen 1.Messung | Einstellungen 2.Messung |
|----|-----------------------------------|----------------------------|----------------------------|
| 1. | FPGA-Design Reset | | |
| 2. | Setze FPGA-Konfigurationsregister | Zahl der Trigger: 100.000 | |
| 3. | Level0-Reset | Länge: 100 ns | |
| 4. | Pulserbetrieb | eingeschaltet | |
| 5. | Pulsverteilung | ungerade Kanäle | gerade Kanäle |
| 6. | Starte Messung | | |
| 7. | Schreibe Fehlercode | | |

Tabelle 6.3: Ablauf des Channel-Map Tests

6.2.3 Der DNL Test

Im dritten Schritt der Testprozedur wird die Genauigkeit der Driftzeitmessung des OTIS analysiert. Die Histogramme werden auf dem FPGA gebildet. Die DNL-Berechnung findet in der Einheit zur Datenprozessierung statt. Ein C++ Programm übernimmt die Normierung der DNL auf die durchschnittliche Zahl der Einträge in jedem Zeitbin. Tabelle 6.4 zeigt den Ablauf. Insgesamt kommt es zu vier Wiederholungen der Messung. Dabei wird jedesmal ein anderer Kanal für die Histogrammierung eingestellt. In Messungen 1 und 2 werden die geraden Kanäle mit Pulsignalen versehen. Daher ist der Histogrammer-Block auf dem FPGA für die Entgegennahme von Driftzeiten in den Kanälen 0 und 16 eingestellt. In den letzten beiden Messungen werden die ungeraden Kanäle 15 und 31 ausgemessen. Damit werden im Serientest die Randkanäle des OTIS untersucht, bei denen am ehesten Fehler auftreten könnten. Ausserdem ist es naheliegend, die DNL-Werte der Kanäle 15 und 16 zu vergleichen, da sich zwischen diesen beiden Kanälen das Routing für die DLL-Kette auf dem OTIS ändert. Die Kanäle 0 bis 15 besitzen auf allen OTIS-Chips ein im Verhältnis zu den restlichen 63 Zeitbins geringfügig größeres 32. Zeitbin. Verantwortlich hierfür ist der verlängerte Signallaufweg zwischen dem 31. und dem 32. Inverter der DLL. Ab Kanal 16 bis 31 bestehen keine

Unterschiede mehr in den Signallaufwegen zwischen den Invertern.

Von der Fast Control auf dem FPGA werden 1.000.000 Trigger erzeugt. Es werden zwischen 600.000 und 650.000 Driftzeiteinträge in den Histogrammen erwartet, bzw. durchschnittlich etwa 10.000 Einträge in jedem Zeitbin.

Zum Bestehen des Test sind folgende Bedingungen zu erfüllen:

- das Minimum der Einträge eines Histogramms beträgt nicht null. Minima und Maxima werden aus den Driftzeithistogrammen von dem Banyan-Netzwerk auf dem FPGA herausgesucht.
- die Zahl der empfangenen Datensätze liegt zwischen 600.000 und 650.000.
- der Wert der DNL befindet sich unter einem definierten Wert, der zum jetzigen Zeitpunkt noch nicht festgelegt wurde.

| | Aktion | Einstellungen 1.Messung | Einstellungen 2.Messung | Einstellungen 3.Messung | Einstellungen 4.Messung |
|----|-----------------------------------|----------------------------|----------------------------|----------------------------|----------------------------|
| 1. | FPGA-Design Reset | | | | |
| 2. | Setze FPGA-Konfigurationsregister | Zahl der Trigger: 1000.000 | | | |
| | | Histogrammer Kanal 0 | Hitsogrammer Kanal 16 | Histogrammer Kanal 15 | Histogrammer Kanal 31 |
| 3. | Pulsverteilung | gerade Kanäle | | ungerade Kanäle | |
| 4. | Level0-Reset | Länge: 100 ns | | | |
| 5. | Starte Messung | | | | |
| 6. | Schreibe Fehlercode | | | | |

Tabelle 6.4: Ablauf des DNL Tests

6.2.4 Der Buffer Overflow Test

In der letzten Stufe des Serientest wird überprüft, wieviele Datensätze der Derandomizing Buffer des OTIS maximal aufnehmen kann. Sobald er eine Füllstufe erreicht hat, bei der nur noch ein freier Speicherplatz für einen Datensatz übrig ist (also 47 der 48 vorhandenen Speicherplätze belegt sind), wird das Buffer Overflow Bit im OTIS-Header aktiviert.

In der Testprozedur wird ein Buffer Overflow künstlich herbeigeführt. Dazu wird die Fast-Control auf dem FPGA im Konsekutiv-Modus betrieben (siehe Tabelle 6.5). Legt man ein Meßintervall von 75 ns zugrunde (bzw. den 3 Bunch-Crossing Auslesemodus), gelangen nach dem ersten Trigger drei Datensätze vom Pipeline-Register des OTIS in den Derandomizing Buffer. Bei allen nachfolgenden konsekutiven Triggern wird der Speicher jedesmal mit einem weiteren Datensatz aufgefüllt. Das heißt, daß bei einem konsekutiven Trigger mit einer Länge von 44 Taktzyklen der Speicher bis auf zwei Zeilen gefüllt ist. Das Buffer Overflow Bit sollte in diesem Fall nicht aktiviert werden.

Ist der Trigger jedoch um einen Taktzyklus länger, wird der Buffer Overflow in den Statusbits des OTIS angezeigt.

Beide Kombinationen werden in dieser Teststufe abgearbeitet.

Die zwei Bedingungen zum Bestehen des Test sind:

- nach der ersten Messung bleibt der Errorbit-Zähler für das Buffer Overflow Bit auf null. Der erwartete Wert für dieses Statusbit im OTIS-Header ist auf null gesetzt.
- nach Ablauf der zweiten Messung bleibt der gleiche Errorbit-Zähler ungleich null. Dies entspricht dem erwarteten Wert auf im FPGA-Konfigurationsregister.

| | Aktion | Einstellungen 1.Messung | Einstellungen 2.Messung |
|----|-----------------------------------|--|--|
| 1. | FPGA-Design Reset | | |
| 2. | Setze FPGA-Konfigurationsregister | Länge des konsekutiven Triggers: 44 Zyklen | Länge des konsekutiven Triggers: 45 Zyklen |
| | | erwarteter Status im OTIS-Header: | |
| | | Buffer Overflow Bit inaktiv | Buffer Overflow Bit aktiv |
| 4. | Fast-Control Betrieb | Konsekutiv-Modus | |
| 3. | Level0-Reset | Länge: 500 ns | |
| 5. | Starte Messung | | |
| 6. | Schreibe Fehlercode | | |

Tabelle 6.5: Ablauf des Buffer Overflow Tests

Kapitel 7

Zusammenfassung

Zur Auslese des LHCb Outer Trackers wurde in Heidelberg ein strahlenharter TDC-Chip entwickelt, der mittlerweile in hoher Stückzahl produziert wurde. Bevor die Chips aus den Wafern herausgeschnitten und weiterverarbeitet werden können, sollen sie einen Funktionstest durchlaufen. Im Rahmen dieser Diplomarbeit wurde dieser Funktionstest konzipiert und aufgebaut.

Es müssen 50 Wafer mit jeweils etwa 160 Chips den automatischen Funktionstest passieren. Hierbei werden zum einen die digitalen Funktionalitäten und zum anderen die Zeitmesseigenschaften untersucht.

Da zum Test für jeden TDC Chip eine sehr hohe Ereigniszahl aufgenommen werden soll, wurden die einzelnen Funktionstests, sowie die Ermittlung der Chipeigenschaft in Hardware realisiert. Statt einer großen Menge an TDC-Rohdaten werden nur die Ergebnisse der einzelnen Untertests an den über eine PCI-Schnittstelle angeschlossenen Mess-PC übergeben. Dies erlaubt Triggerraten von 1,1 MHz, bei denen der TDC Chip im LHCb-Experiment betrieben werden soll und damit auch eine enorme Beschleunigung des Testablaufs.

Der 8 Bit Datenstrom des TDC-Chips wird an einen FPGA-Baustein (Stratix EP1S25 von Altera) übergeben. Nach der Datenübernahme werden die im Daten-Header kodierten Chip-Funktionen überprüft. Das Design erlaubt zu dem die Erstellung eines Histogramms zur Bestimmung der Trefferhäufigkeit aller 32 Kanäle des TDC-Chips. Weiterhin besteht die Möglichkeit, für jeden beliebigen Eingangskanal das Zeitmessverhalten (differentielle Nichtlinearität) zu bestimmen. Hierzu wird ein Driftzeithistogramm für eine Million Zufallstreffer aufgezeichnet. Aus der Ungleichverteilung kann die differentielle Nichtlinearität ermittelt werden.

Bis auf die I²C-Slow Control Signale werden alle vom TDC Chip benötigten Steuersignale (Fast Control Signale) wie Resets, Trigger, Clock ebenfalls vom FPGA bereitgestellt, was den Vorteil hat, daß keine zusätzliche Steuerelektronik notwendig ist.

Durch ausgiebige Tests wurde die Funktionalität der implementierten Schaltungen verifiziert. Alle durchgeführten Messungen bestätigen die korrekte Funktion der Testalgorithmen.

Der vorläufige Testaufbau mit integrierter Nadelkarte wurde dazu genutzt, ausführliche Studien des Arbeitspunktes ($V_{digital}$ und V_{analog}) der Chip Version 1.2 durchzuführen. Es konnte gezeigt werden, daß diese Chipversion 1.2 bereits alle Anforderungen, die der LHCb-Einsatz stellt, erfüllt.

Anhang A

Anhang

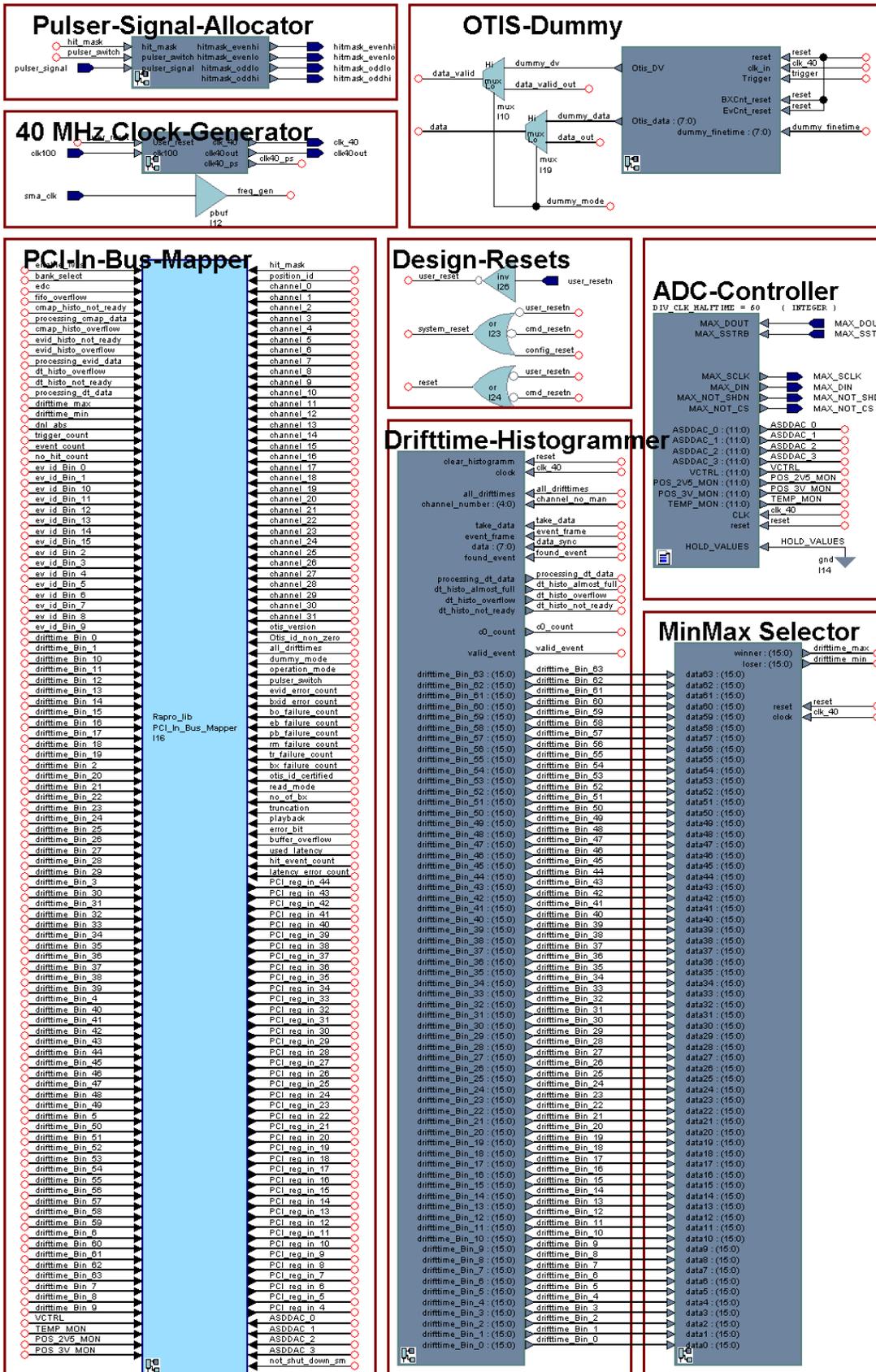


Abbildung A.2: Zweiter Teil des Toplevels

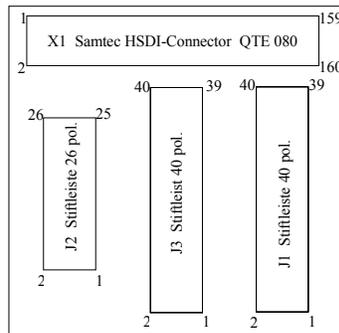
| Address | Bit | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|-------------------------------|----|----|----|----|----|----|----|----------------|----|----|----|----|----|----|----|-----------------------------------|----|----|----|----|----|---|---|---------------------------------|---|---|---|---|---|---|---|
| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 (x000) | Offset | | | | | | | | Latency | | | | | | | | Reset Value | | | | | | | | Otis_ID | | | | | | | |
| 1 (x004) | Header Settings | | | | | | | | Dummy Finetime | | | | | | | | Channel Number | | | | | | | | | | | | | | | |
| 2 (x008) | | | | | | | | | | | | | | | | | Trigger Rate | | | | | | | | | | | | | | | |
| 3 (x00C) | | | | | | | | | | | | | | | | | Number of Trigger | | | | | | | | | | | | | | | |
| 4 (x010) | | | | | | | | | | | | | | | | | Number of Events | | | | | | | | | | | | | | | |
| 5 (x014) | | | | | | | | | | | | | | | | | Run Number | | | | | | | | | | | | | | | |
| 6 (x018) | Number of consecutive Trigger | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 (x01C) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 (x020) | | | | | | | | | | | | | | | | | Number of BX Bits Failure Count | | | | | | | | | | | | | | | |
| 9 (x024) | | | | | | | | | | | | | | | | | Buffer Overflow Bit Failure Count | | | | | | | | | | | | | | | |
| 10 (x028) | | | | | | | | | | | | | | | | | Error Bit Failure Count | | | | | | | | | | | | | | | |
| 11 (x02C) | | | | | | | | | | | | | | | | | Playback Mode Bit Failure Count | | | | | | | | | | | | | | | |
| 12 (x030) | | | | | | | | | | | | | | | | | Truncation Mode Bit Failure Count | | | | | | | | | | | | | | | |
| 13 (x034) | | | | | | | | | | | | | | | | | Read Mode Bit Failure Count | | | | | | | | | | | | | | | |
| 14 (x038) | | | | | | | | | | | | | | | | | Number of send Trigger | | | | | | | | | | | | | | | |
| 15 (x03C) | | | | | | | | | | | | | | | | | Number of all received Events | | | | | | | | | | | | | | | |
| 16 (x040) | | | | | | | | | | | | | | | | | Event-ID Error Count | | | | | | | | | | | | | | | |
| 17 (x044) | | | | | | | | | | | | | | | | | Bunch-Crossing-ID Error Count | | | | | | | | | | | | | | | |
| 18 (x048) | Design Version | | | | | | | | | | | | | | | | Design Status Bits | | | | | | | | | | | | | | | |
| 19 (x04C) | DNL for selected Channel | | | | | | | | | | | | | | | | Number of Events in Fifo | | | | | | | | | | | | | | | |
| 20 (x050) | | | | | | | | | | | | | | | | | Number of COs | | | | | | | | | | | | | | | |
| 21 (x054) | Minimum of Histogramming | | | | | | | | | | | | | | | | Maximum of Histogramming | | | | | | | | | | | | | | | |
| 22 (x058) | | | | | | | | | | | | | | | | | Latency Error Count | | | | | | | | | | | | | | | |
| 23 (x05C) | | | | | | | | | | | | | | | | | Number of valid Hit Events | | | | | | | | | | | | | | | |
| 24 (x060) | OTIS Status Bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 25 (x064) | | | | | | | | | | | | | | | | | ASDDAC 1 | | | | | | | | ASDDAC 0 | | | | | | | |
| 26 (x068) | | | | | | | | | | | | | | | | | ASDDAC 3 | | | | | | | | ASDDAC 2 | | | | | | | |
| 27 (x06C) | | | | | | | | | | | | | | | | | Temperature Monitor | | | | | | | | Vctrl | | | | | | | |
| 28 (x070) | | | | | | | | | | | | | | | | | Positive 3V Monitor | | | | | | | | Positive 2.5V Monitor | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 34 (x088) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 1 | | | | | | | | Number of Drifttime Hits Bin 0 | | | | | | | |
| 35 (x08C) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 3 | | | | | | | | Number of Drifttime Hits Bin 2 | | | | | | | |
| 36 (x090) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 5 | | | | | | | | Number of Drifttime Hits Bin 4 | | | | | | | |
| 37 (x094) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 7 | | | | | | | | Number of Drifttime Hits Bin 6 | | | | | | | |
| 38 (x098) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 9 | | | | | | | | Number of Drifttime Hits Bin 8 | | | | | | | |
| 39 (x09C) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 11 | | | | | | | | Number of Drifttime Hits Bin 10 | | | | | | | |
| 40 (x0A0) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 13 | | | | | | | | Number of Drifttime Hits Bin 12 | | | | | | | |
| 41 (x0A4) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 15 | | | | | | | | Number of Drifttime Hits Bin 14 | | | | | | | |
| 42 (x0A8) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 17 | | | | | | | | Number of Drifttime Hits Bin 16 | | | | | | | |
| 43 (x0AC) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 19 | | | | | | | | Number of Drifttime Hits Bin 18 | | | | | | | |
| 44 (x0B0) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 21 | | | | | | | | Number of Drifttime Hits Bin 20 | | | | | | | |
| 45 (x0B4) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 23 | | | | | | | | Number of Drifttime Hits Bin 22 | | | | | | | |
| 46 (x0B8) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 25 | | | | | | | | Number of Drifttime Hits Bin 24 | | | | | | | |
| 47 (x0BC) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 27 | | | | | | | | Number of Drifttime Hits Bin 26 | | | | | | | |
| 48 (x0C0) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 29 | | | | | | | | Number of Drifttime Hits Bin 28 | | | | | | | |
| 49 (x0C4) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 31 | | | | | | | | Number of Drifttime Hits Bin 30 | | | | | | | |
| 50 (x0C8) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 33 | | | | | | | | Number of Drifttime Hits Bin 32 | | | | | | | |
| 51 (x0CC) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 35 | | | | | | | | Number of Drifttime Hits Bin 34 | | | | | | | |
| 52 (x0D0) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 37 | | | | | | | | Number of Drifttime Hits Bin 36 | | | | | | | |
| 53 (x0D4) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 39 | | | | | | | | Number of Drifttime Hits Bin 38 | | | | | | | |
| 54 (x0D8) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 41 | | | | | | | | Number of Drifttime Hits Bin 40 | | | | | | | |
| 55 (x0DC) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 43 | | | | | | | | Number of Drifttime Hits Bin 42 | | | | | | | |
| 56 (x0E0) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 45 | | | | | | | | Number of Drifttime Hits Bin 44 | | | | | | | |
| 57 (x0E4) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 47 | | | | | | | | Number of Drifttime Hits Bin 46 | | | | | | | |
| 58 (x0E8) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 49 | | | | | | | | Number of Drifttime Hits Bin 48 | | | | | | | |
| 59 (x0EC) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 51 | | | | | | | | Number of Drifttime Hits Bin 50 | | | | | | | |
| 60 (x0F0) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 53 | | | | | | | | Number of Drifttime Hits Bin 52 | | | | | | | |
| 61 (x0F4) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 55 | | | | | | | | Number of Drifttime Hits Bin 54 | | | | | | | |
| 62 (x0F8) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 57 | | | | | | | | Number of Drifttime Hits Bin 56 | | | | | | | |
| 63 (x0FC) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 59 | | | | | | | | Number of Drifttime Hits Bin 58 | | | | | | | |
| 64 (x100) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 61 | | | | | | | | Number of Drifttime Hits Bin 60 | | | | | | | |
| 65 (x104) | | | | | | | | | | | | | | | | | Number of Drifttime Hits Bin 63 | | | | | | | | Number of Drifttime Hits Bin 62 | | | | | | | |
| 66 (x108) | | | | | | | | | | | | | | | | | Number of Hits Event-ID 1 | | | | | | | | Number of Hits Event-ID 0 | | | | | | | |
| 67 (x10C) | | | | | | | | | | | | | | | | | Number of Hits Event-ID 3 | | | | | | | | Number of Hits Event-ID 2 | | | | | | | |
| 68 (x110) | | | | | | | | | | | | | | | | | Number of Hits Event-ID 5 | | | | | | | | Number of Hits Event-ID 4 | | | | | | | |
| 69 (x114) | | | | | | | | | | | | | | | | | Number of Hits Event-ID 7 | | | | | | | | Number of Hits Event-ID 6 | | | | | | | |
| 70 (x118) | | | | | | | | | | | | | | | | | Number of Hits Event-ID 9 | | | | | | | | Number of Hits Event-ID 8 | | | | | | | |
| 71 (x11C) | | | | | | | | | | | | | | | | | Number of Hits Event-ID 11 | | | | | | | | Number of Hits Event-ID 10 | | | | | | | |
| 72 (x120) | | | | | | | | | | | | | | | | | Number of Hits Event-ID 13 | | | | | | | | Number of Hits Event-ID 12 | | | | | | | |
| 73 (x124) | | | | | | | | | | | | | | | | | Number of Hits Event-ID 15 | | | | | | | | Number of Hits Event-ID 14 | | | | | | | |
| 74 (x128) | | | | | | | | | | | | | | | | | Number of Hits Channel 1 | | | | | | | | Number of Hits Channel 0 | | | | | | | |
| 75 (x12c) | | | | | | | | | | | | | | | | | Number of Hits Channel 3 | | | | | | | | Number of Hits Channel 2 | | | | | | | |
| 76 (x130) | | | | | | | | | | | | | | | | | Number of Hits Channel 5 | | | | | | | | Number of Hits Channel 4 | | | | | | | |
| 77 (x134) | | | | | | | | | | | | | | | | | Number of Hits Channel 7 | | | | | | | | Number of Hits Channel 6 | | | | | | | |
| 78 (x138) | | | | | | | | | | | | | | | | | Number of Hits Channel 9 | | | | | | | | Number of Hits Channel 8 | | | | | | | |
| 79 (x13C) | | | | | | | | | | | | | | | | | Number of Hits Channel 11 | | | | | | | | Number of Hits Channel 10 | | | | | | | |
| 80 (x140) | | | | | | | | | | | | | | | | | Number of Hits Channel 13 | | | | | | | | Number of Hits Channel 12 | | | | | | | |
| 81 (x144) | | | | | | | | | | | | | | | | | Number of Hits Channel 15 | | | | | | | | Number of Hits Channel 14 | | | | | | | |
| 82 (x148) | | | | | | | | | | | | | | | | | Number of Hits Channel 17 | | | | | | | | Number of Hits Channel 16 | | | | | | | |
| 83 (x14C) | | | | | | | | | | | | | | | | | Number of Hits Channel 19 | | | | | | | | Number of Hits Channel 18 | | | | | | | |
| 84 (x150) | | | | | | | | | | | | | | | | | Number of Hits Channel 21 | | | | | | | | Number of Hits Channel 20 | | | | | | | |
| 85 (x154) | | | | | | | | | | | | | | | | | Number of Hits Channel 23 | | | | | | | | Number of Hits Channel 22 | | | | | | | |
| 86 (x158) | | | | | | | | | | | | | | | | | Number of Hits Channel 25 | | | | | | | | Number of Hits Channel 24 | | | | | | | |
| 87 (x15C) | | | | | | | | | | | | | | | | | Number of Hits Channel 27 | | | | | | | | Number of Hits Channel 26 | | | | | | | |
| 88 (x160) | | | | | | | | | | | | | | | | | Number of Hits Channel 29 | | | | | | | | Number of Hits Channel 28 | | | | | | | |
| 89 (x164) | | | | | | | | | | | | | | | | | Number of Hits Channel 31 | | | | | | | | Number of Hits Channel 30 | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1023 (xFFC) | Command Register | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| OTIS Status Bits | | | | | | | | | |
|---------------------------|------------------------|-------------------------|-------------------------|----------------------|--------------------------|---------------------------|-----------------|--------------|--|
| 31...27 | 26...15 | 14 | 13...12 | 11 | 10 | 9 | 8 | 7...0 | |
| Reserved | OTIS-ID | Read Mode | No of BX | Truncation | Playback Mode | Error Bit | Buffer Overflow | Used Latency | |
| Design Status Bits | | | | | | | | | |
| 9 | 8 | 7 | 6...4 | | | 3 | 2 | 1...0 | |
| Bank-Select-Mode | Channel-Mode | Pulser enable | Trigger Operation Mode | | | Dummy-Mode | Non-Zero-Mode | OTIS Version | |
| Design Status Bits | | | | | | | | | |
| 16 | 15 | 14 | 13 | 12 | 11 | 10 | | | |
| Processing Drifttime Data | Hit Mask | Cmap Histo Overflow | Cmap Histo Not Ready | Processing Cmap Data | LVDS-Status | Sync Fifo Overflow | | | |
| Design Status Bits | | | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | | | |
| EDC-Mode | EventID Histo Overflow | EventID Histo Not Ready | Processing EventID Data | Position ID | Drifttime Histo Overflow | Drifttime Histo Not Ready | | | |

Farbcodierung: Read only Write only Read / Write Reserved Barred

Abbildung A.3: Belegung der PCI-Register

Otis-Stratix-Testadapter



| Stratix Pin | X1 QTE 080 | J2 Stiflleiste 26 pol. | Signal | Type |
|-------------|------------|------------------------|-------------|--------------------|
| V26 | 52 | 1 | LDO | LVTTL input |
| - | 24 | 2 | Ground | output |
| V27 | 60 | 3 | RP | LVTTL input |
| - | 30 | 4 | Ground | output |
| V28 | 58 | 5 | WP | LVTTLinput |
| - | 36 | 6 | Ground | output |
| AA24 | 40 | 7 | DLL_Reset | LVTTL output |
| - | 42 | 8 | Ground | output |
| AA25 | 38 | 9 | Bank_Select | LVTTL output |
| - | 48 | 10 | Ground | output |
| V25 | 54 | 11 | Enable_EDC | LVTTL output |
| - | 50 | 12 | Ground | output |
| V24 | 44 | 13 | Clock | LVDS output (pos.) |
| V23 | 46 | 14 | _Clock | LVDS output (neg.) |
| Y23 | 76 | 15 | Trigger | LVDS output (pos.) |
| Y24 | 78 | 16 | _Trigger | LVDS output (neg.) |
| W23 | 20 | 17 | L0_Reset | LVDS output (pos.) |
| W24 | 22 | 18 | _L0_Reset | LVDS output (neg.) |
| W25 | 26 | 19 | EV_Reset | LVDS output (pos.) |
| W26 | 28 | 20 | _EV_Reset | LVDS output (neg.) |
| W27 | 32 | 21 | BX_Reset | LVDS output (pos.) |
| W28 | 34 | 22 | _BX_Reset | LVDS output (neg.) |
| Y28 | 64 | 23 | Reserved | |
| Y27 | 66 | 24 | Reserved | |
| Y26 | 70 | 25 | Reserved | |
| Y25 | 72 | 26 | Reserved | |

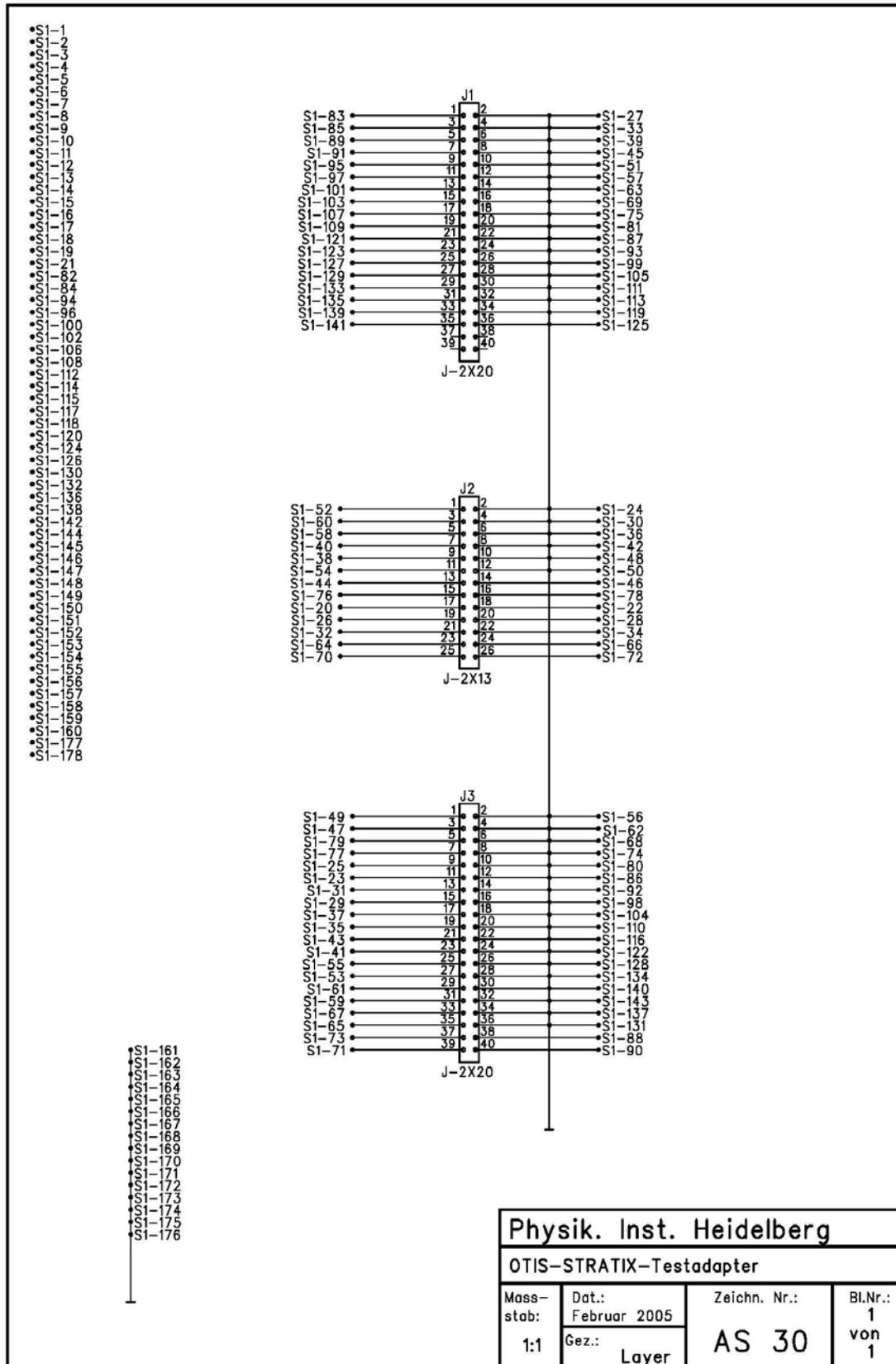
Abbildung A.4: Teil 1 der Testadapter - Pinbelegung

| Stratix Pin | X1 QTE 080 | J1 Stiflleiste 40 pol. | Signal | Type |
|-------------|------------|------------------------|------------|-------------------|
| V31 | 83 | 1 | _DataValid | LVDS input (neg.) |
| V32 | 85 | 3 | DataValid | LVDS input (pos.) |
| W31 | 89 | 5 | _Data0 | LVDS input (neg.) |
| W32 | 91 | 7 | Data | LVDS input (pos.) |
| Y31 | 95 | 9 | _Data1 | LVDS input (neg.) |
| Y32 | 97 | 11 | Data1 | LVDS input (pos.) |
| AA30 | 101 | 13 | _Data2 | LVDS input (neg.) |
| AA31 | 103 | 15 | Data2 | LVDS input (pos.) |
| AB30 | 107 | 17 | _Data3 | LVDS input (neg.) |
| AB31 | 109 | 19 | Data3 | LVDS input (pos.) |
| V29 | 121 | 21 | _Data4 | LVDS input (neg.) |
| V30 | 123 | 23 | Data4 | LVDS input (pos.) |
| W29 | 127 | 25 | _Data5 | LVDS input (neg.) |
| W30 | 129 | 27 | Data5 | LVDS input (pos.) |
| Y29 | 133 | 29 | _Data6 | LVDS input (neg.) |
| Y30 | 135 | 31 | Data6 | LVDS input (pos.) |
| AA29 | 139 | 33 | _Data7 | LVDS input (neg.) |
| AA28 | 141 | 35 | Data7 | LVDS input (pos.) |
| - | - | 37 | - | - |
| - | - | 39 | - | - |
| - | 27 | 2 | Ground | output |
| - | 33 | 4 | Ground | output |
| - | 39 | 6 | Ground | output |
| - | 45 | 8 | Ground | output |
| - | 51 | 10 | Ground | output |
| - | 57 | 12 | Ground | output |
| - | 63 | 14 | Ground | output |
| - | 69 | 16 | Ground | output |
| - | 75 | 18 | Ground | output |
| - | 81 | 20 | Ground | output |
| - | 87 | 22 | Ground | output |
| - | 93 | 24 | Ground | output |
| - | 99 | 26 | Ground | output |
| - | 105 | 28 | Ground | output |
| - | 111 | 30 | Ground | output |
| - | 113 | 32 | Ground | output |
| - | 119 | 34 | Ground | output |
| - | 125 | 36 | Ground | output |
| - | - | 38 | - | - |
| - | - | 40 | - | - |

Abbildung A.5: Teil 2 der Testadapter - Pinbelegung

| Stratix Pin | X1 QTE 080 | J3 Stiflleiste 40 pol. | Signal | Type |
|-------------|------------|------------------------|--------|---------------|
| AA26 | 49 | 1 | ID0 | LVTTTL output |
| AA27 | 47 | 3 | ID1 | LVTTTL output |
| AF28 | 79 | 5 | ID2 | LVTTTL output |
| AF27 | 77 | 7 | ID3 | LVTTTL output |
| AC26 | 25 | 9 | ID4 | LVTTTL output |
| AC25 | 23 | 11 | ID5 | LVTTTL output |
| AD25 | 31 | 13 | ID6 | LVTTTL output |
| AD26 | 29 | 15 | ID7 | LVTTTL output |
| AE26 | 37 | 17 | ID8 | LVTTTL output |
| AE25 | 35 | 19 | ID9 | LVTTTL output |
| AF25 | 43 | 21 | ID10 | LVTTTL output |
| AF26 | 41 | 23 | ID11 | LVTTTL output |
| AB26 | 55 | 25 | ID12 | LVTTTL output |
| AB27 | 53 | 27 | ID13 | LVTTTL output |
| AC28 | 61 | 29 | ID14 | LVTTTL output |
| AC27 | 59 | 31 | ID15 | LVTTTL output |
| AD27 | 67 | 33 | ID16 | LVTTTL output |
| AD28 | 65 | 35 | ID17 | LVTTTL output |
| AE27 | 73 | 37 | ID18 | LVTTTL output |
| AE28 | 71 | 39 | ID19 | LVTTTL output |
| - | 56 | 2 | Ground | output |
| - | 62 | 4 | Ground | output |
| - | 68 | 6 | Ground | output |
| - | 74 | 8 | Ground | output |
| - | 80 | 10 | Ground | output |
| - | 86 | 12 | Ground | output |
| - | 92 | 14 | Ground | output |
| - | 98 | 16 | Ground | output |
| - | 104 | 18 | Ground | output |
| - | 110 | 20 | Ground | output |
| - | 116 | 22 | Ground | output |
| - | 122 | 24 | Ground | output |
| - | 128 | 26 | Ground | output |
| - | 134 | 28 | Ground | output |
| - | 140 | 30 | Ground | output |
| - | 143 | 32 | Ground | output |
| - | 137 | 34 | Ground | output |
| - | 131 | 36 | Ground | output |
| AC30 | 88 | 38 | Ground | output |
| AC29 | 90 | 40 | Ground | output |

Abbildung A.6: Teil 3 der Testadapter - Pinbelegung



| | | | |
|---------------------------------|--------------|--------------|---------|
| Physik. Inst. Heidelberg | | | |
| OTIS-STRATIX-Testadapter | | | |
| Massstab: | Dat.: | Zeichn. Nr.: | Bl.Nr.: |
| 1:1 | Februar 2005 | AS 30 | 1 |
| | Gez.: | | von |
| | Layer | | 1 |

Abbildung A.7: Testadapter AS30

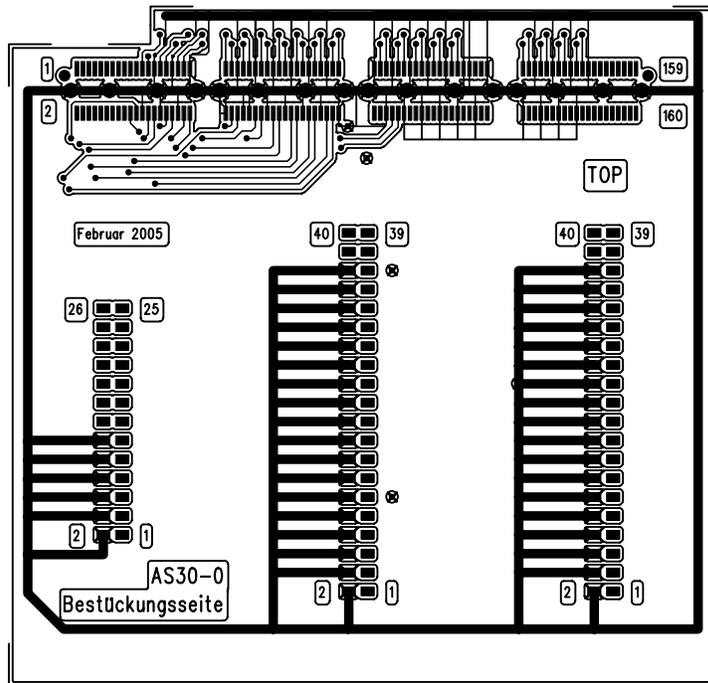


Abbildung A.8: Testadapter Bestückungsseite - Top

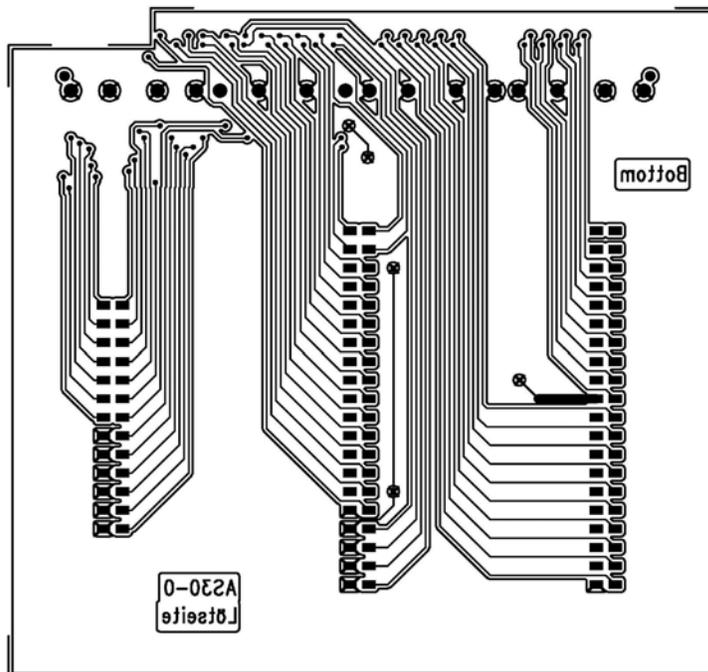


Abbildung A.9: Testadapter - Bottom

Literaturverzeichnis

- [1] CERN,
“*LHCb Technical Proposal*“ 20. Februar 1998

- [2] “*LHCb Technical Design Report*“
CERN, 9. September 2003

- [3] Dirk Wiedner,
„*Aufbau der Ausleseelektronik für das
äußere Spurkammersystem des LHCb-Detektors*“,
Physikalisches Institut der Universität Heidelberg 2004

- [4] T.Shuijk, U.Uwer
“*Specification of the OTIS-to-ASDBLR interface*”
NIKHEF, Physikalisches Institut der Uni Heidelberg, 1. September 2003.

- [5] N. Dressnandt et al.
“*Implementation of the ASDBLR Straw Tube Readout ASIC in DMILL Technology*”
IEEE (2000) Trans. on Nucl. Sci. V48 n4 p1239 R. Bevensee et al.,
“*An Amplifier-Shaper-Discriminator with Baseline Restoration
for the ATLAS Transition Radiation Tracker*”
IEEE (1996) Trans. on Nuc. Sci. V43 p1725
<http://www.hep.upenn.edu/atlas/asdblr>

- [6] Ad Berkien, Tom Shuijk, U.Uwer, D.Wiedner, Albert Zwart.
“*Specifications IF13-1 Prototype of the Auxiliary Board for the Outer Tracker*”,
Version 2.0, LHCb-2004-073 September 16, 2004
<http://cdsweb.cern.ch/search.py?recid=793180&ln=en>

- [7] P. Moreira, T. Toiff, A. Kluge, G. Cervelli, A. Marchioro, and J. Christiansen
“*GOL Reference Manual, Gigabit Optical Link Transmitter manual*”,
CERN - EP/MIC, Geneva Switzerland May 2002 Version 1.4.

- [8] Haefeli, G; Uwer, U; Vollhardt, A; Wiedner, D
“*Prototype IF14-1 for an Optical 12 input Receiver Card for the LHCb TELL1
Board*”
LHCb 2004-072, electronics, public; Geneva : CERN 6 Sep 2004
<http://cdsweb.cern.ch/search.py?recid=792529&ln=en>

- [9] Guido Haefeli, Aurelio Bay, Federica Legger, Laurent Locatelli, Jorgen Christiansen, Dirk Wiedner.
“*Specification for a common read out board for LHCb*”,
Version 3.0, LHCb 2003-007 IPHE 2003-02 September 2, 2003
- [10] “*TTCvx, Technical description and users manual.*
A VME-sized multiplexer, encoder and fiber-optics transmitter module for the Timing, Trigger and Control System of the LHC detectors.”
Per Gällnö CERN/EP/ATE/dq, per.gallno@cern.ch, May 21, 1999 Draft
- [11] “*TTC-VMEbus INTERFACE TTCvi-MkII,*
Module Identification: EP 680-1128-050-C”,
RD12 Project, Ph. Farthouat, P.Gällnö CERN EP-ATE, Rev1.6 May 2000
- [12] Jorgen Christiansen et al.,
“*TTCrx Reference Manual,*
A Timing, Trigger and Control Receiver ASIC for LHC Detectors”
- [13] Jan Knopf,
„*Aufbau eines Auslesesystems für die Äusseren Spurkammern des LHCb-Detektors*“
Physikalisches Institut der Universität Heidelberg 2004
- [14] Harald Deppe, Uwe Stange , Ulrich Trunk, Ulrich Uwer
Physikalisches Institut, Universität Heidelberg
* „*The OTIS Reference Manual*“,
Version 1.2 γ 1.1 β , 2005.
- [15] Uwe Stange,
“*Development and Characterization of a Radiation Hard Readout Chip for the LHCb Outer Tracker Detector*“
Physikalisches Institut der Universität Heidelberg 2005
- [16] Philips Semiconductors,
„*The I²C-bus specification*“,
version 2.1, January 2000
- [17] Mirco Nedos,
„*Entwicklung und Implementierung eines mit FPGAs realisierten Systems zur Auslese des Äußeren Spurkammersystems des LHCb-Detektors*“,
TU Dresden 2004
- [18] Altera Corporation.
“*Stratix PCI Development Board Data Sheet*”,
Version 2.0, September 2003.
- [19] Prof. Thomas Ludwig,
„*Vorlesung Betriebssysteme und Netzwerke*“
WS 2002/2003

- [20] Altera Corporation.
„*Stratix Device Handbook Volume 1*,
San Jose, September 2004.

Danksagung

An dieser Stelle möchte ich meinen herzlichen Dank an alle Personen aussprechen, die zum Gelingen dieser Arbeit beigetragen haben. Insbesondere bedanke ich mich bei:

Herrn Professor Ulrich Uwer für die freundliche Aufnahme in die LHCb-Gruppe und die intensive Betreuung. Während meiner Diplomarbeit habe ich bei ihm und seiner Arbeitsgruppe sehr viel gelernt.

Herrn Professor Lindenstruth danke ich für die Übernahme der Zweitkorrektur.

Ein ganz besonderer Dank geht an Dirk Wiedner für die umfassende Betreuung, ein immer offenes Ohr für alle meine Fragen, stets kompetente Hilfe, besonders bei Durchführung der Messungen und für das Durchsehen dieser Arbeit.

Ein sehr großer Dank geht Jan Knopf, der mir vor allem bei FPGA-Fragen stets hilfreich zur Seite stand, bei allen aufgetretenden Problemen nie die Geduld verloren hat und mir beim Erstellen dieser Arbeit viele nützliche Tipps geben konnte.

Uwe Stange für die vielen Gespräche über den OTIS und für die vielfältigen Anleitungen.

Ulrich Trunk für die Einführung ins ASIC Labor und die Gespräche über den OTIS.

Klaus Layer aus der Elektronikwerkstatt für die Fertigung der Testadapterkarte.

Sebastian Bachmann für die tatkräftige Unterstützung gegen Ende dieser Arbeit.

Ralph Achenbach für die Einführung in die Funktionsweise des automatischen Bonders.

Roger Wolf für die geleistete Hilfestellung am Ende dieser Arbeit.

Prof. Franz Eisele, Iuri Bagaturia, Tanja Haas und Michael Walther für eine stets angenehme Atmosphäre in der Gruppe.

Ich möchte meinen herzlichsten Dank gegenüber meinen Eltern Gudrun und Harald Muckerheide aussprechen, die mich zu jeder Zeit unterstützt und ermutigt haben.

Zu guter Letzt bedanke ich mich bei Christina Bäuerle dafür, daß sie immer für mich da war und für ihre Geduld über das vergangene Jahr hinweg.

Erklärung:

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

Heidelberg, den 16. November 2005