

Diplomarbeit

Malte Borchers

Vergleichende Untersuchungen von Messverfahren
an Detektoren für Neutronen- und Röntgenstrahlung

*Fakultät Technik und Informatik
Department Informations- und
Elektrotechnik*

*Faculty of Engineering and Computer Science
Department of Information and
Electrical Engineering*

Malte Borchers

Vergleichende Untersuchungen von Messverfahren
an Detektoren für Neutronen- und Röntgenstrahlung

Diplomarbeit eingereicht im Rahmen der Diplomprüfung
im Studiengang Informations- und Elektrotechnik
Studienrichtung Informationstechnik
am Department Informations- und Elektrotechnik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr.-Ing. Franz Schubert
Zweitgutachter : Prof. Dr.-Ing. Jürgen Missun

Abgegeben am 21. April 2008

Malte Borchers

Thema der Diplomarbeit

Vergleichende Untersuchungen von Messverfahren an Detektoren für Neutronen- und Röntgenstrahlung

Stichworte

Verzögerungsleitung, CSRD, Anodenbewertung, Impulsbreitenverhältnis, Frequenztoleranz

Kurzzusammenfassung

Die hier vorliegende Diplomarbeit beschäftigt sich mit einem neuen Verfahren, das bisher nur als Idee vorliegt und beim Einsatz von ortsauflösenden Detektoren für Neutronen- und Röntgenstrahlung eingesetzt werden könnte.

Es wird ein Modell erstellt, in dem das Funktionieren dieses Verfahrens im niederfrequenten Bereich nachgewiesen wird. Die Optimierung der Arbeitspunkte ist ebenfalls in diesem Modell möglich.

Die Funktionsweisen aller gegenwärtig verwendeten Messmethoden werden hardwaretechnisch auf das Niveau dieses Modells gebracht. So sind Vergleiche aller Messmethoden unter gleichen Bedingungen möglich.

In einer abschließenden Versuchsreihe werden die gewaltigen Vorzüge der innovativen Messmethode praktisch belegt.

Malte Borchers

Title of the paper

Comparative studies of measuring methods against detectors for Neutron- and X-rays

Keywords

Delayline, CSRD, influence of anode, proportion of pulse width, tolerance of frequency

Abstract

This following thesis deals with a new method, only what is known as an idea. This innovative method could be used in high-precision detectors for Neutron- and X-rays. A model is created, in which the functioning of this method is demonstrated, especially in the low-frequency range. In this model, the optimization of the working points is also possible.

The operating modes of all current measurement methods are technically reconstructed and upgraded to the level of this model. Thus it is possible for all measurement methods to work under the same test conditions.

In a final test series, the huge advantages of the innovative measuring method are practically demonstrated.

Inhaltsverzeichnis

Einleitung	1
1 Grundlagen	3
1.1 Arbeitsprinzip eines ortsauflösenden Detektors	3
1.2 Aufbau und Funktionsweise einer Delayline.....	5
1.3 Gegenwärtig übliche TDC – Messverfahren	6
1.3.1 TDC1- Verfahren.....	7
1.3.2 TDC2- Verfahren.....	8
1.4 Datenerfassung	9
2 Problembeschreibung	10
2.1 Beschreibung des Totzeiteffektes	10
2.2 Aufgabenstellung.....	12
3 Lösung	13
3.1 Grundgedanke.....	13
3.2 Weiterführender Gedanke.....	14
3.3 Entstehung eines neuen Messverfahrens (CSRSD)	15
3.4 Realisierung der Begleitumstände	16
3.5 Realisierung der Vergleichsmöglichkeit.....	17
4 Hardwareumsetzung	18
4.1 Detektorsignalsimulator.....	18
4.2 Entwicklungsumgebung	19
4.3 Interface	21
4.4 Aufbau CSRSD- Verfahren	22
4.5 Aufbau der TDC- Verfahren.....	24
4.6 Handshake FPGA/MCA	24
5 Software	26
5.1 CSRSD- Verfahren	26
5.1.1 Encoder.vhd	28
5.1.2 Simulationsergebnis des Encoders	34

5.1.3 Delay_anode.vhd	35
5.1.4 Koinzidenz.vhd	36
5.1.5 Handshake.vhd.....	37
5.1.6 Toplevel.vhd	37
5.2 TDC1- Verfahren.....	37
5.2.1 Counter.vhd	39
5.2.1 Xy.vhd	39
6 Experimentell gewonnene Ergebnisse.....	40
6.1 Messaufbau	40
6.2 Vergleichende Untersuchungen im Frequenzband	42
6.3 Untersuchung der Anodenbewertung (CSRD)	45
6.4 Untersuchung des Arbeitspunktes (CSRD)	47
7 Zusammenfassung und Ausblick.....	51
8 Literaturverzeichnis	53
9 Bildverzeichnis	55
10 Anhang.....	57
A Tabellen (vergleichende Untersuchungen)	57
A.1 Messergebnisse TDC1- Verfahren	57
A.2 Messergebnisse TDC2- Verfahren	58
A.3 Messergebnisse CSRD- Verfahren	59
B Tabellen (Arbeitspunkt Betrachtung)	60
B.1 Messergebnisse CSRD (ohne Anodenbewertung).....	60
B.2 Messergebnisse CSRD (Impulsbreitenoptimierung)	61
B.3 Messergebnisse CSRD (Frequenztoleranz)	62
C Inhalt der beigelegten CD	63
Versicherung über die Selbständigkeit.....	64

Einleitung

Die Materialforschung ist ein umfangreicher und bedeutender Zweig der Wissenschaft. Die Forschungsergebnisse können in unterschiedlichsten Anwendungsbereichen (z.B. in der Medizin) verwendet werden. Modernste Materialforschung wird heutzutage unter anderem mit hochempfindlichen, ortsauflösenden 2D- Neutronen- oder Röntgendetektoren betrieben. Die Detektoren werden eingesetzt, um den Strahlendurchsatz durch beliebiges Material zu messen. Unter Berücksichtigung der Zeit und der Strahlendosis kann somit ein Eindruck über die innere Materialbeschaffenheit gewonnen werden.

Diese Diplomarbeit wurde in dem Unternehmen DENEX- GmbH (**D**etektoren für **N**eutronen- und **X**-Strahlung) geschrieben. Die DENEX entwickelt und vertreibt die eben beschriebenen Detektoren für Neutronen- und Röntgenstrahlung. Der Firmensitz ist in Lüneburg.

Unter Strahleneinwirkung erzeugen diese Detektoren elektrische Impulse. Diese werden durch eine geeignete Elektronik interpretiert und hinterlassen auf dem Bildschirm eines Computers das Schattenbild des Neutronen- bzw. Röntgenstrahls. Es gibt unterschiedliche Messverfahren, die Information dieser Impulse aufzubereiten. Zwei davon sind praxisüblich, und eines ist nur vom Ansatz her als Idee bekannt.

Ziel dieser Arbeit ist es, diese Idee aufzuarbeiten, um sie dann mit den beiden bekannten Messverfahren¹ zu vergleichen. Dieses innovative Verfahren gilt als revolutionär, da es einem ganz neuen Grundkonzept unterliegt. Es sollen seine Vorzüge nicht nur theoretisch, sondern auch praktisch gezeigt werden.

¹ Im Folgenden werden die (Mess)verfahren optional auch als (Mess)methoden bezeichnet.

In einer abschließenden Betrachtung dieser Ausarbeitung ist dann anhand der gewonnenen Ergebnisse der weitere Projektverlauf dieser innovativen Messmethode zu diskutieren. - Die Firmenleitung der DENEX- GmbH erwägt vom Ergebnis dieser Arbeit abhängig, einen patentrechtlichen Schutz für dieses neue Verfahren zu beantragen.

Der Hauptteil dieser Diplomarbeit umfasst sechs Kapitel.

Das **erste** Kapitel erläutert die prinzipiellen Grundlagen eines Detektors. Sie sind für das weitere Verständnis dieser Arbeit erforderlich.

Im **zweiten** Kapitel wird die konkrete Aufgabenstellung formuliert.

Das **dritte** Kapitel beschäftigt sich mit der Lösung bzw. mit der theoretischen Umsetzbarkeit des Messverfahrens.

Im **vierten** Kapitel wird auf die Entwicklungsumgebung und auf die Hardwarerealisierung eingegangen.

Das **fünfte** Kapitel bezieht sich rein auf die VHDL programmiertechnische Umsetzung.

Im **letzten** Kapitel (Kapitel 6) des Hauptteils erfolgt dann die messtechnische Gegenüberstellung aller Methoden und eine gezielte Arbeitspunkt Betrachtung.

1 Grundlagen

In diesem Kapitel geht es darum, die prinzipielle Funktionsweise eines ortsauflösenden Neutronen- bzw. Röntgenstrahldetektors zu erklären.

1.1 Arbeitsprinzip eines ortsauflösenden Detektors

Ein Neutron, welches z.B. zuvor beim Durchdringen des Atomgitters eines zu untersuchenden Messobjektes gestreut, abgelenkt bzw. absorbiert wurde, besitzt einen hohen Informationsgehalt über die Materialbeschaffenheit des Messobjektes. Sein Einschlagsort gilt es zu bestimmen. [1]

Detektoren bestehen im wesentlichen aus drei Rahmen, die mit feinen Drähten im Abstand von ca. 2 mm bespannt sind (siehe Bild 1). Diese Rahmen sind in einem gasdichten Gehäuse montiert, welches mit einem Spezialgasgemisch² befüllt ist. Sie sind die Elektroden, zwischen denen ein elektrisches Feld herrscht.

Die in der Mitte liegende Anode ist mit ca. + 2000 V vorgespannt, während die sich außen befindenden Kathoden auf Erdpotential liegen. Die Drähte der beiden Kathodenrahmen sind quer zueinander angeordnet. So teilt jeder Rahmen eine der X- od. Y-Achsen in einzelne Abschnitte auf.

² unter anderem bestehend aus Argon (Zählgas) und Methan (Löschgas) [2, Seite3, Abschnitt 1.4]

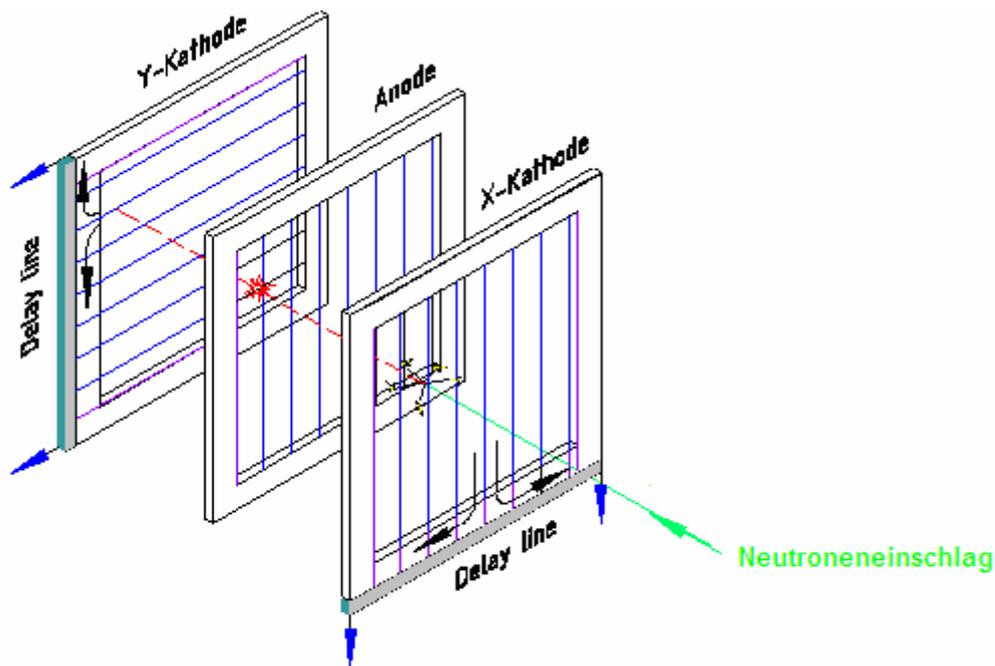


Bild 1: Die Detektionsfläche wird durch X/Y-Kathode + Anode gebildet

An einer Stirnseite der Kathodenrahmen ist eine elektrische Verzögerungsleitung³ mit $n-2$ Anzapfungen angebracht (n = Anzahl der Drähte). Die Drahtenden sind der Reihe nach mit einem Anschlusspunkt der Delayline verbunden.

Dringt ein Teilchen in den Detektor ein, so wird das Gasgemisch an der Eintrittsstelle ionisiert. Zwischen der Anode und den beiden Kathoden entsteht ein Ladungsblitz. Dieser bewirkt, dass an den sich in der Nähe befindlichen Elektroden-Drähten (Anode u. Kathoden) eine elektrische Ladungverschiebung auftritt. Diese wird durch die Kathodendrähte als Stromimpuls den Anzapfungen der Delayline zugeführt. Dort fließen sie jeweils mit halber Energie in beide Richtungen ab. [2, Seite 8, Abschnitt 2.1]

³ im Folgenden mit engl. Fachausdruck „Delayline“ bezeichnet

Am Austritt der Delayline stehen die Signalpaare⁴ dann zur Weiterverarbeitung zur Verfügung. Ihr zeitlicher Bezug zueinander gibt Aufschluss über die Einschlagsposition des auslösenden Teilchens. Der Anodenimpuls wird direkt und ohne Zeitverzögerung herausgeführt. Er signalisiert, dass ein Ereignis stattgefunden hat.

Wenn ein Teilchen zwischen zwei benachbarten Kathodendrähten einschlägt, dann erzeugt es auf beiden Drähten im umgekehrten Verhältnis seines jeweiligen Abstandes einen mehr oder weniger starken Stromimpuls. In dem dazugehörigen Teilstück der Delayline fließen sie dann wieder zusammen und addieren sich zu einem einzigen Impuls. Auch zeitlich stellt sich am Ausgang der Delayline ein Mittelwert ein. – **Somit ist die zeitliche Auflösung und damit auch die Ortsauflösung des Detektors höher, als durch das Drahrastermaß vorgegeben!**

1.2 Aufbau und Funktionsweise einer Delayline

Die horizontal und vertikal gespannten Kathodendrähte (z.B. 150*150) werden auf eine Delayline, bestehend aus Kapazitäten und Induktivitäten geführt, (siehe Bild 2). Die gesamte Delayline besitzt eine Gesamtverzögerungszeit von ca. 400 ns bei einer Impedanz von 50 Ohm. Durch exakt gleiche Bauteilverwendung und Anordnung lässt sich eine proportionale Schrittweite von

$$t_{Step} = \frac{t_{Delayline}}{n-1} \quad (1)$$

herstellen (n = Anzahl der Kathodendrähte). Die Schrittweite gibt die Signallaufzeit zwischen zwei Kathodendrähte (ca. 2,7 ns) an .

⁴ im Folgenden mit X0,X1 und Y0, Y1 bezeichnet

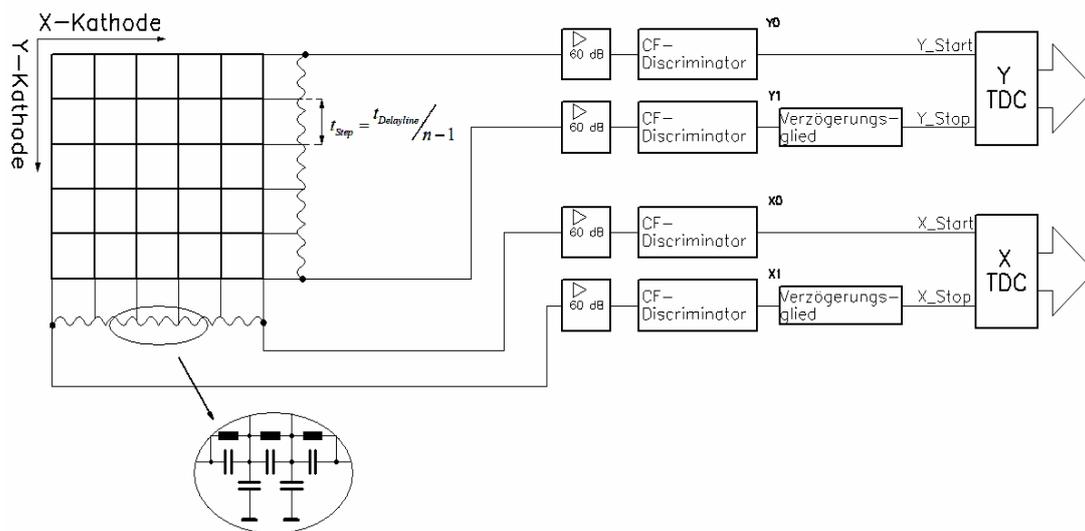


Bild 2: Prinzipieller Aufbau des Delayline- Readouts

An den beiden Ausgängen einer Delayline ist jeweils ein linearer, rauscharmer Verstärker (60 dB) und ein *Constant Fraction Discriminator*⁵ zur Signalaufbereitung nachgeschaltet. Dieses ist notwendig, da die Analogsignale stark gedämpft wurden und nun digitaltechnisch weiterverarbeitet werden sollen.

1.3 Gegenwärtig übliche TDC – Messverfahren

An den Ausgängen der Constant Fraction Discriminatoren stehen die zu messenden Signale zur Verfügung. Bei den gegenwärtig verwendeten **Time-to-Digital-Conversion-** Methoden (TDC- Methoden) wird der zeitliche Bezug der Signale untereinander festgestellt und interpretiert. Das Ergebnis liegt dann als Digitalwert vor.

Ein TDC ist nichts anderes als eine elektronische Stoppuhr, die mit einem Start- und einem Stoppeingang versehen ist. Der Vorteil dieser Messmethoden ist der geringe

⁵ fein einstellbare Unterscheidungsschwelle, ob es sich um ein Nutzsignal oder molekulares Rauschen handelt

Elektronikaufwand. Durch die Signalamplitudenunabhängigkeit sind sie auch sehr robust gegen äußeren Störeinflüssen.

Zwei TDC- Messmethoden haben sich in der Praxis durchgesetzt. Sie liefern bei verhältnismäßig geringer Strahlungsleistung gute Ergebnisse und werden im Folgenden mit TDC1- und TDC2 - Methode vorgestellt. [2, Seite 6, Abschnitt 1.8]

1.3.1 TDC1- Verfahren

Die Signale X0 und Y0 bestimmen immer die Startzeitpunkte der TDC's. Damit die Stoppzeitpunkte niemals vor den Startzeitpunkten auftreten können, werden die Stoppimpulse zusätzlich um die Signallaufzeit einer Delayline verzögert [2, Seite 7, Abb. 11].

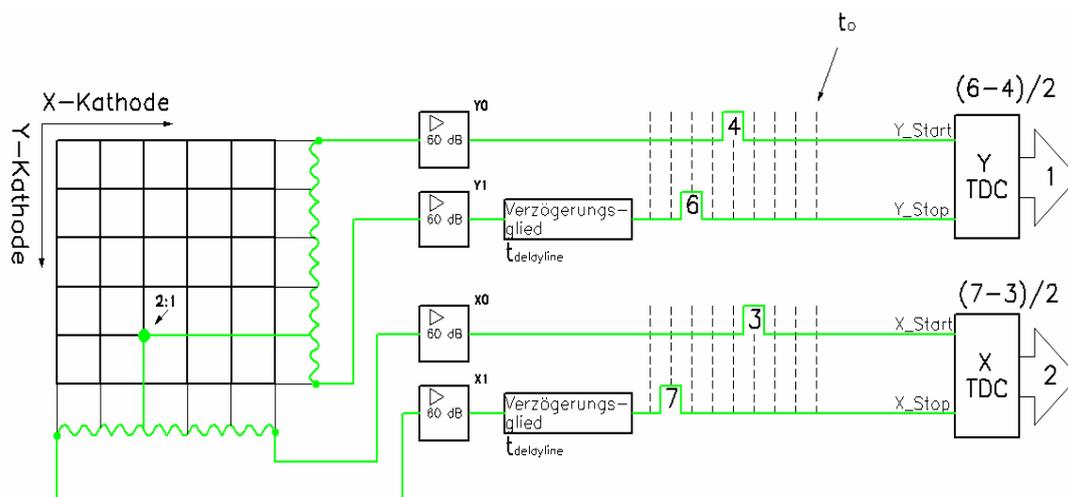


Bild 3: Signalfluss + Auswertung bei einfachem Neutroneneinschlag
(TDC1- Methode)

Der Zeitabstand zwischen Start- u. Stoppimpuls ist immer proportional dem zu messenden Weg. Von Vorteil ist, dass dieser Zeitabstand doppelt so hoch ist, wie es der Laufzeit auf der Delayline entsprechen würde. Der TDC kann also mit halber Zählfrequenz arbeiten, was bei den hohen Frequenzen (GHz- Bereich) eine

Erleichterung bedeutet. Ein weiterer Vorteil ist, dass sich die Position des Ergebnisbildes durch Verändern der Verzögerungszeit am Verzögerungsglied auf einfache Weise justieren lässt (siehe hierzu Abschnitt 1.4 Datenerfassung).

1.3.2 TDC2- Verfahren

Sowohl der X- als auch der Y-TDC werden mit dem gleichen Anodenimpuls gestartet [2, Seite 7, Abb. 12]. Gestoppt werden die TDC's mit den entsprechenden Signalen X1 u. Y1. Auch hier ist der Zeitabstand zwischen den Start- u. Stoppsignalen dem Weg proportional, allerdings nur halb so lang wie bei der TDC1- Methode (siehe Bild 3).

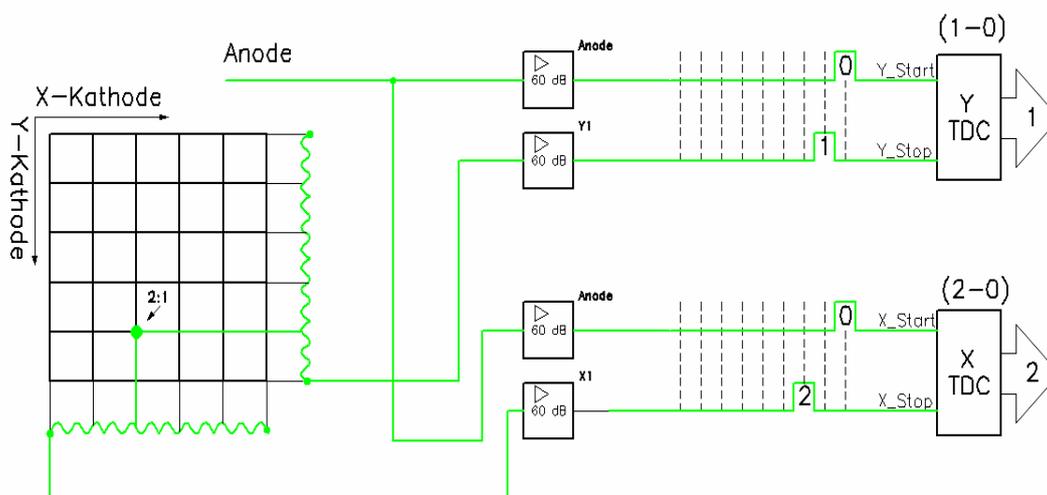


Bild 4: Signalfloss + Auswertung bei einfachem Neutroneneinschlag
(TDC2- Methode)

Diese Methode hat den Vorteil, dass die mittlere Totzeit nur 1/3 der mittleren Totzeit der TDC1- Methode ist. Es ist somit eine höhere Strahlungsleistung zulässig, um auf gleiche Messwertqualität zu kommen (siehe hierzu Abschnitt 2.1 Beschreibung des Totzeiteffektes)

1.4 Datenerfassung

Das durch die TDC's erzeugte digitale X/Y-Zahlenwertepaar beschreibt die Position des Neutroneneinschlages. Es wird an ein Datenerfassungssystem MCA (Multi Chanel Analysator) gesendet. Der MCA ordnet jeder möglichen Position eine Adresse eines 32-Bit-Speichers zu. Jedes Mal, wenn er einen Datensatz empfängt, incrementiert er den Wert des betreffenden Speicherwortes um 1. Hierdurch sammelt sich für jede Position ein Wertevorrat an. Graphisch dargestellt ergibt sich somit eine 3- dimensionale Struktur des Messobjektes. Die Werteübergabe der TDC's an den MCA erfolgt nach einem Handshake-Verfahren, welches unter Abschnitt 4.6 näher erläutert wird. Der MCA kommuniziert seinerseits zur Grafikdarstellung mit einem handelsüblichen PC. [3]

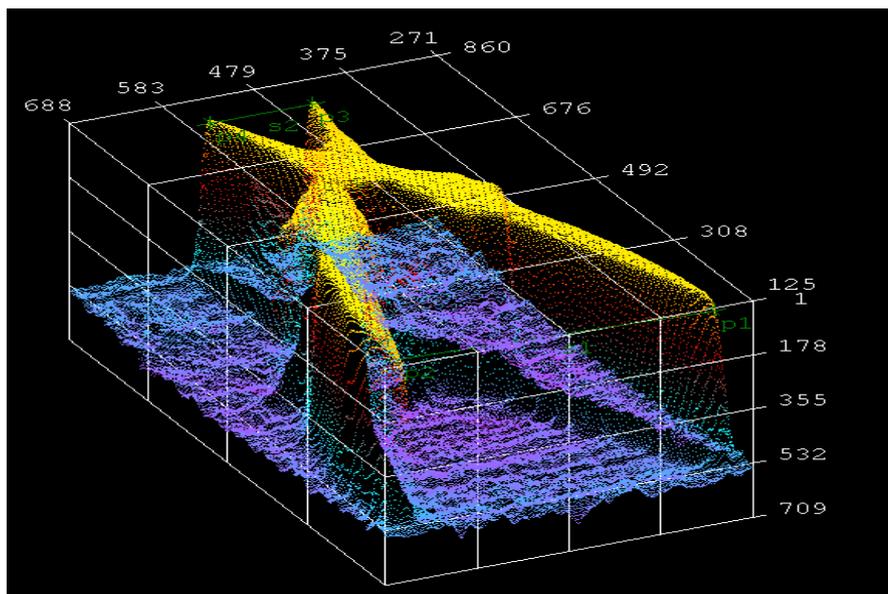


Bild 5: 3D- Darstellung einer strahlendurchsetzten Spitzzange [4]

2 Problembeschreibung

Im 2. Kapitel soll mit den bereits gegebenen Grundlagenkenntnissen über die TDC- Messmethoden die Aufgabenstellung dieser Ausarbeitung erläutert werden. Hierzu muss der Totzeiteffekt detailliert beschrieben werden. Der Totzeiteffekt tritt bei beiden TDC- Messmethoden auf, wird aber im folgenden Abschnitt 2.1 nur für die TDC1- Methode beschrieben. Dieses ist für das Verständnis völlig ausreichend.

2.1 Beschreibung des Totzeiteffektes

In der Praxis beträgt die zu messende maximale **statistische** Ereignisrate ca. 10^6 Cycle/ s [2, Seite 2, Abschnitt 1.3]. Es können mehrere Ereignisse gleichzeitig auftreten. Dieser Fall soll im folgenden Bild anhand eines Doppelereignisses dargestellt werden.

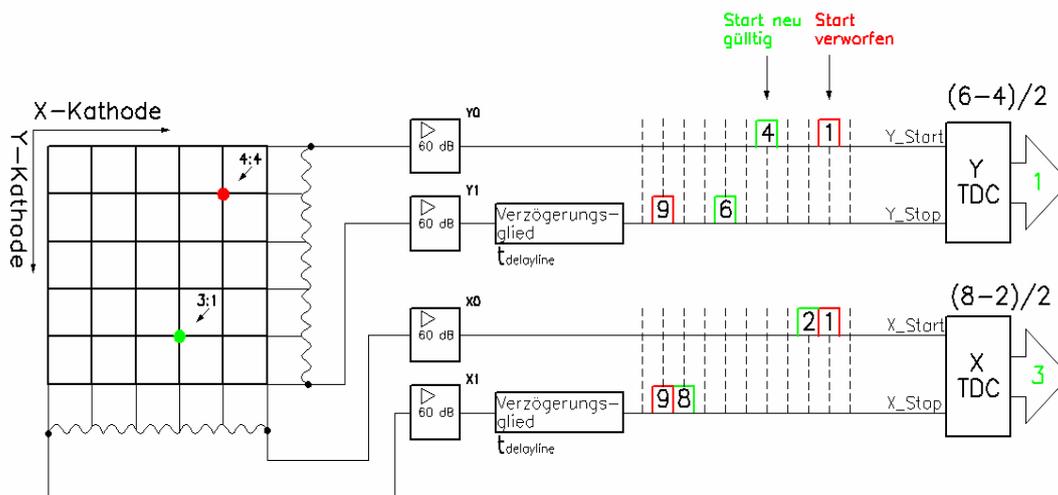


Bild 6: Signalfluss + Auswertung bei Doppelereignis (TDC1- Methode, grün gültig)

Trotz eines tatsächlichen Doppelereignisses kann nur ein Ereignis gezählt werden. Dieses liegt daran, dass der TDC (Bezug nur auf eine Achse) doppelt gestartet wird.

Die durch das erste Signal (**rot**) gestartete Zeitmessung wird komplett verworfen und durch ein später eintreffendes zweites Signal (**grün**) erneut gestartet. Das erste eintreffende Stoppsignal (**grün**) ist dem Startsignal (**grün**) zugehörig, und es wird ein gültiges Ergebnis umgesetzt. Das zweite eintreffende Stoppsignal (**rot**) ist nun bedeutungslos.

Treffen beispielsweise in zeitnahen Abständen weitere Ereignisse ein, so würde sich der eben beschriebene Effekt entsprechend fortsetzen. Bei ungünstigen Zeit- und Ereigniskonstellationen kann es zu Fehlmessung kommen. Dieses liegt daran, dass die Umsetzung durch zwei nicht zueinander gehörenden Signalgruppen veranlasst wurde.

Zusammenfassend ist festzuhalten, dass bei beiden TDC- Methoden des Delayline-Readouts eine Zeit vergeht, bis ein Ereignis verarbeitet ist. In dieser Zeit kann **nur das eine** Ereignis verarbeitet werden. Sie soll im Folgenden als Totzeit bezeichnet werden. Sie ist je nach Ereigniskonstellation unterschiedlich. Die mittlere Totzeit beträgt bei beiden Methoden unterschiedlich:

$$t_{Tot(TDC1)} = 1,5 * t_{Delayline} \quad (2)$$

und

$$t_{Tot(TDC2)} = 0,5 * t_{Delayline} \quad (3)$$

Grundsätzlich bedeutet eine hohe Totzeit eine geringe Ausbeute. Der zu untersuchende Gegenstand muss also über einen längeren Zeitraum der Neutronenquelle ausgesetzt sein.

In der Praxis laufen die Messungen oft über mehrere Tage bzw. Wochen, um aussagefähige Daten zu erhalten. Dieses ist mit entsprechenden Unkosten und Aufwand von Personal und Laboreinrichtungen verbunden. Könnte man die Strahlendosis erhöhen und damit die Messzeit verkürzen, wäre das ein kolossaler Gewinn.

2.2 Aufgabenstellung

An diesem Punkt setzt nun das übergeordnete Thema dieser Ausarbeitung ein. **Es soll eine elektronische Messmethode für das „Delayline Readout Prinzip“ entwickelt und getestet werden, welche eine geringere Totzeit als die herkömmlichen TDC- Messmethoden besitzt.** Im Schlusskapitel soll eine vergleichende Gegenüberstellung aller Messmethoden unter exakt gleichen Versuchsbedingungen durchgeführt werden.

3 Lösung

Im dritten Kapitel wird die theoretische Lösung eines leistungsfähigeren Verfahrens systematisch hergeleitet.

3.1 Grundgedanke

Das Prinzip der Zeitdifferenzmessung mittels eines TDC's soll komplett verworfen werden. Es muss zukünftig möglich sein, dicht nacheinander folgende Signale aufzufangen, um sie dann für eine Ortsbestimmung zu verwenden. Die laufende Vorfolgemessung darf hierbei nicht beendet werden. Es müssen alle Ereignisse zu jedem Zeitpunkt parallel weiterverarbeitet werden können.

Durch diesen theoretischen Ansatz ist die Idee gekommen, die Signalpaare auf gleichem Wege wieder über externe Delaylines umgekehrt zusammenzuführen. An dem Punkt, an welchem sich die Signale in der externen Delayline treffen, addieren sich die beiden Signalamplituden. Durch Festsetzen einer Schaltschwelle können nun die Punkte der Signalkoinzidenz⁶ logisch erfasst werden. Die nachfolgend dargestellten Bilder beziehen sich aus Gründen der Übersichtlichkeit nur noch auf eine Achse. Das Messprinzip ist für beide Achsen aber das gleiche.

⁶ Koinzidenz ist eine messtechnische Übereinstimmung

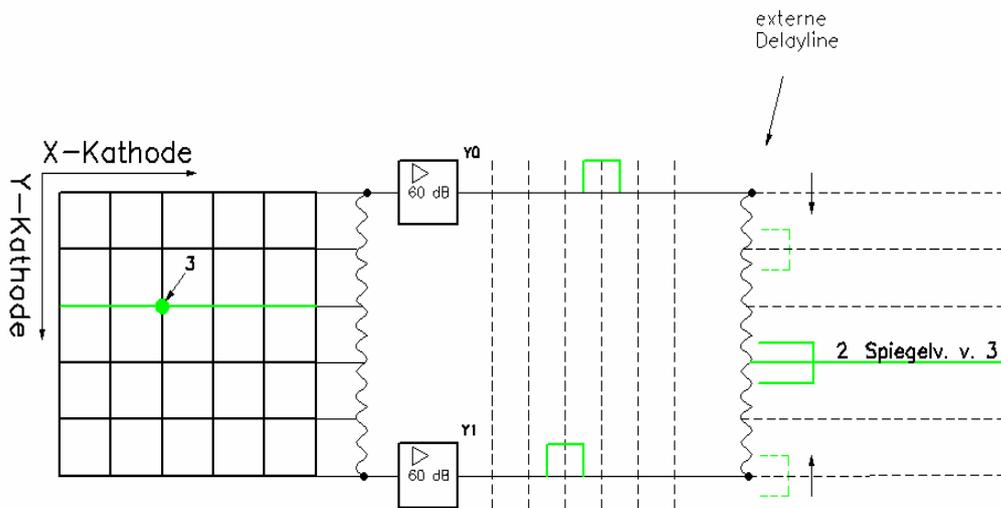


Bild 7: Spiegelverkehrte Signalkoinzidenz, durch Addition der Signalamplituden

Nachteil dieses Verfahrens ist der enorme Arbeits- und Schaltungsplatzaufwand, der nötig wäre. Es müssten pro Detektor zwei weitere handgefertigte Delaylines aufgewendet werden. Die einzelnen Schritintervalle der Delayline wären parallel mit einer Elektronik zu verdrahten. Der ganze Aufbau muss hierbei den Hochfrequenzanforderungen des Systems entsprechen. Er würde zuviel Platz in Anspruch nehmen. Anders als beim TDC ließe sich dieser Aufbau nicht mehr in einem NIM- Einschub⁷ unterbringen. Diese beiden Faktoren rechtfertigen eine derartige Umsetzung nicht, aber sie liefern einen neuen Denkansatz!

3.2 Weiterführender Gedanke

Die externen analogen Delaylines sind jeweils durch 2 synchrone und gegensinnig laufende digitale Schieberegister zu ersetzen. Die Anzahl der Registerzellen sollte mindestens gleich der Anzahl der Kathodendrähte sein. Beide Schieberegister

⁷ Einschubmaß in der Nuklearelektronik

werden von entgegen gesetzter Seite mit den zu messenden Signalen einer Gruppe gespeist. Der Arbeitstakt der Schieberegister muss **exakt** auf die analoge Signallaufzeit eines Schritintervalls der internen Delayline abgestimmt sein. Durch eine paarweise Verundung (Konjunktion) aller zueinander gehörender Registerzellen können nun Koinzidenzen logisch erfasst werden.

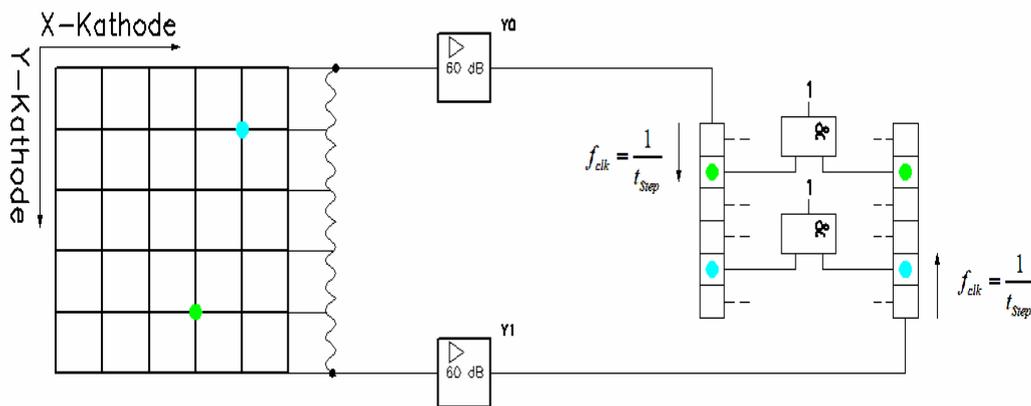


Bild 8: Signalkoinzidenz durch synchrone Schieberegister + Registerzellverundung

3.3 Entstehung eines neuen Messverfahrens (CSR)

Anhand der vorhergegangenen Überlegungen soll im nächsten Kapitel eine Hardware-schaltung entwickelt werden, welche diesen bislang theoretischen Gedanken des Koinzidenzverfahrens umsetzt. Im Folgenden wird diese innovative Messmethode als **Coincidence- Shift- Register- Detection- Method**⁸ bezeichnet.

Diese Methode soll auf Grund des hohen Hardwareaufwandes in einem FPGA⁹ implementiert werden. Der Hardwareaufwand kommt Großteils durch die Länge der Schieberegister zustande. So müssen für eine 10 Bit- Ortsauflösung die Register eine Länge von 1024 Zellen aufweisen. Als ganz große Schwachstelle erweisen sich die Taktanforderungen an das System. So müsste der Arbeitstakt der Schieberegister

⁸ CSR- Methode ist ein Eigenname und in der Literatur nicht bekannt

⁹ „Vor Ort modifizierbarer“ Logikbaustein der modernen Digitaltechnik

ca.300 MHz betragen, um eine 8 Bit- Ortsauflösung zu bekommen. Voraussetzung hierfür wäre bereits, die Signallaufzeit der gesamten Delayline von 400 auf 800 ns zu verdoppeln.

Durch eine theoretische Erhöhung der Signallaufzeit der Delayline in den Millisekundenbereich könnten scheinbar alle Probleme der Taktanforderung gelöst sein. Eine Lösung dieser Überlegung funktioniert jedoch nicht, da die Signale durch zu starke Dämpfung auf der Delayline an Bandbreite verlieren. Eine maximale Erhöhung der Signallaufzeit auf 800 ns ist tatsächlich möglich.

Durch diese Begleitumstände der Taktanforderung ist eine FPGA- Entwicklung im Low Cost Bereich¹⁰ nicht erfolgsversprechend. Wäre es jedoch möglich, die Funktionalität und die Vorteile des CSR-D-Verfahrens eindeutig zu beweisen, so könnten weitere Entwicklungsschritte (nicht Bestandteil dieser Arbeit) unternommen werden.

3.4 Realisierung der Begleitumstände

Aus den eben genannten Beweisgründen ist ein Detektorsignalsimulator durch Mitarbeiter der Denex-GmbH gebaut worden (siehe [Anhang C](#)). Er erzeugt künstliche Impulsgruppen, wie sie ein Detektor bei entsprechender Bestrahlung erzeugen würde.- Nur viel langsamer!

Dieser Simulator ermöglicht es, die Signallaufzeit der Pseudodelayline auf 2,5 bzw. 10 ms einzustellen. Entsprechend einer am Simulator einstellbarer parametrierbarer Ortsauflösung, erzeugt der Simulator die gewünschten Messsignale. Um nun bspw.

¹⁰ Unter welchen Bedingungen dieses Verfahren entwickelt werden soll

eine 8-Bit Ortsauflösung darstellen zu können, müssten die Schieberegister mit 100 kHz getaktet werden. Bei der max. gewünschten Ortsauflösung von 10 Bit wären es 400 kHz. Diese Taktanforderungen können nun bedenkenlos mit dem FPGA eingehalten werden. Natürlich könnte der FPGA auch höher getaktet arbeiten, nur liegen die Grenzen diesmal aufseiten des Simulators.

Ein weiterer wesentlicher Vorteil des Simulators liegt darin, dass verschiedene Parametrierungen vorgenommen werden können, welches unter echten Bedingungen nicht möglich ist. Es kann bspw. wahlweise zwischen **statistischer** oder **periodischer** Impulsfolge des Simulators entschieden werden. Diese Parametrierbarkeit ist Mittel zum Zweck, um im Kapitel 6 (Experimentell gewonnene Ergebnisse) fundierte und umfangreichere Aussagen machen zu können.

3.5 Realisierung der Vergleichsmöglichkeit

Da im Schlusskapitel die Vorzüge der CSR-D- Methode experimentell dargestellt werden sollen, bietet es sich an, vergleichende Untersuchungen mit den TDC- Methoden durchzuführen. Um exakt gleiche Versuchbedingungen einzuhalten, müssen die bereits bekannten TDC- Methoden auf den Signalsimulator angewendet werden. Hierfür müssen eigene Hardwareschaltungen beider TDC Verfahren nachempfunden werden. Die Implementierung erfolgt entsprechend in einem FPGA.

4 Hardwareumsetzung

Dieses Kapitel erläutert die genaueren hardwaretechnischen Zusammenhänge. Im ersten Abschnitt werden der Aufbau und die Funktionsweise des Detektorsimulators besprochen. Im zweiten Abschnitt wird die Entwicklungsumgebung vorgestellt, mit welcher die 3 Messverfahren realisiert wurden. Entsprechend wird auch auf das hierfür benötigte Interface und den konkreten Schaltungsaufbau aller Verfahren eingegangen. Abschließend wird das Kommunikationsprotokoll (siehe Abschnitt 4.6) zwischen dem FPGA und dem MCA erläutert.

4.1 Detektorsignalsimulator

Wie bereits im vorherigen Kapitel beschrieben, wurde ein Detektorsimulator entwickelt. Dieser Simulator besteht im wesentlichen aus drei PIC- Prozessoren vom Typ 16F876 aus dem Hause Microchip [5]. Die Bedienungsanleitung des Simulators ist dem Anhang C zu entnehmen.

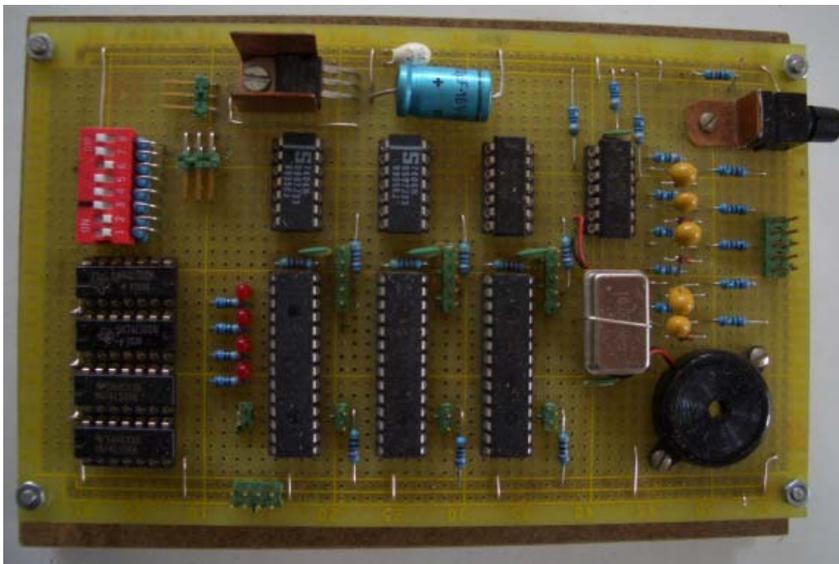


Bild 9: 3 PIC- Prozessoren bilden das Herzstück des Simulators

Die Besonderheit des Simulators liegt darin, dass alle 3 Prozessoren unterschiedlich parametrierbar sind. Jeder der Prozessoren kann wahlweise das Impulsmuster (Anode, X0, X1, Y0, Y1) einer von 7 geometrischen Figuren liefern. Die Impulsbreite und der Impulsabstand sind individuell über DIL- Schalter einzustellen. Des weiteren lässt sich zwischen periodischer und statistischer Ereignisfolge wählen. Durch eine logische Veroderung (Disjunktion) gleichnamiger Prozessorausgänge werden 5 zentrale Ausgabepunkte geschaffen. Bei gleichzeitigem jedoch unabhängigen Prozessorlauf entsteht eine annähernd realistische Impulsfolge an den Ausgabepunkten. Bei statistischer Ereignisrate können nun bis zu drei zeitgleichen Mehrfachereignissen auftreten.

Eine Ausgangsstufe sorgt dafür, dass sich an den zentralen Ausgängen Impulse von -1V an 50 Ohm ausbilden. Das entspricht den Anschlussdaten der Detektoren.

4.2 Entwicklungsumgebung

Für die praktische Umsetzung der drei Detektormessverfahren bietet sich nach heutigem Stand der Technik eine FPGA- Lösung an. Die Hardwarebeschreibungssprache hierfür ist unter anderem VHDL¹¹, mit welcher die Designs erstellt wurden. Als Entwicklungsboard dient das Starterkit *Spartan 3E* der Firma Xilinx. [7]

¹¹ Very High Speed Integrated Circuit **H**ardware **D**escription **L**anguage

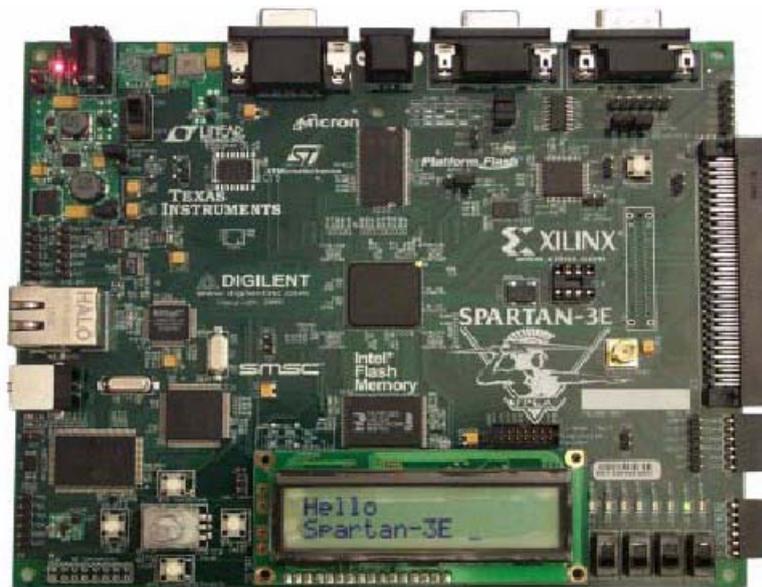


Bild 10: FPGA- Entwicklungsboard *Starterkit Spartan 3E*

Neben dem FPGA¹² selbst verfügt das Board über zahlreiche Features, welche die Entwicklungsarbeit sinnvoll unterstützen:

- 50 MHz On-Board Oscillator
- PROM (XCF04S)
- 4-Kanal Digital/Analog Umsetzer (DAC)
- 512 Mbit DDR SDRAM
- I/O Pins (Tri-state- fähig)
- CoolRunner II CPLD
- RS 232- Schnittstelle
- Character LCD- Screen
- LED's , uvm.

¹² Typ XC3 S500e [8]

Die Software für das Starter Kit kann als Industrieversion des ISE-Webpacks 8.2i, unter dem im Literaturverzeichnis angegebenen Link [9] kostenlos herunter geladen werden. Das Webpack 8.2i unterstützt u.a. die Designsimulation (Modelsim), VHDL Synthese, Implementierung und Programmierung über die JTAG- Schnittstelle¹³ .

4.3 Interface

Um die Kommunikation zwischen Signalsimulator, FPGA und MCA zu gewährleisten, ist ein I/O-Interface gebaut worden. Dieses Interface besitzt einen 100- poligen Hirose- FX2 Stecker¹⁴. Dieser kann in die FX2- Buchse des FPGA-Boards eingesteckt werden. Hierdurch ist der Zugriff auf alle Inputs/Outputs des FPGA's über die Interfaceplatine gegeben.

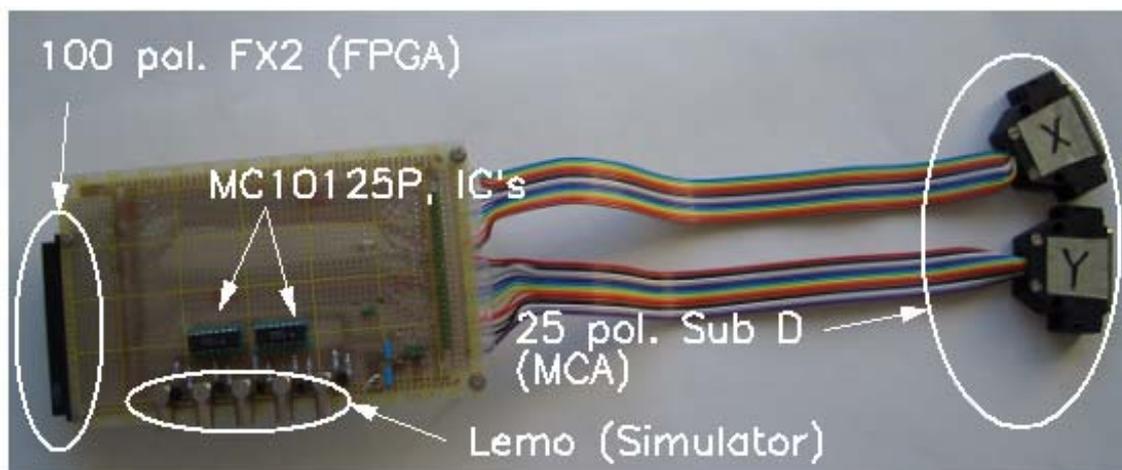


Bild 11: Interface zwischen Simulator, FPGA und MCA

¹³ das JTAG- Protokoll ermöglicht das Debuggen und Programmieren von FPGA's und Prozessoren direkt in der Schaltung

¹⁴ sehr hochfrequenztauglicher Stecker von der Firma HIROSE

Die Simulatorsignale werden über Lemo- Steckverbindungen auf die Interfaceplatine geführt. Die Wandlung der $-1V$ Pegel, in die für das FPGA verständliche TTL- Pegel, erfolgt mit einer Transistorschaltung und IC's aus dem Hause Motorola¹⁵. Der genaue Beschaltungsplan ist dem Anhang C zu entnehmen. Die ausgangsseitige Verbindung zwischen FPGA und MCA wird über zwei 25- pol. Sub D- Steckverbindungen¹⁶ realisiert. Hier ist keine weitere Pegelwandlung nötig. Die Verbindung kann direkt erfolgen.

4.4 Aufbau CSRD- Verfahren

Um das unter Abschnitt 3.3 vorgestellte CSRD- Verfahren hardwaretechnisch umzusetzen, muss die genaue Funktionsweise der Messmethode erklärt werden:

Es werden insgesamt 5 parametrierbare Schieberegister benötigt. 4 der Register sind für die Signalkoinzidenz beider Kathoden zuständig (jeweils 2 Register pro Kathode). Da es innerhalb der Register zu einem hohen Signalaufgebot kommt, kann es auch hier zu Fehlmessungen kommen. Es können Koinzidenzen durch nicht zueinandergehörender Signale stattfinden. Um den Fehler möglichst gering zu halten, müssen zusätzliche logische Bedingungen mit in die Bewertung einfließen.

- Ein Ergebnis kann nur dann gültig sein, wenn **kathodengleichzeitig** Signalkoinzidenzen festgestellt werden. Dieses liegt daran, dass die Gesamtlaufzeit beider Signalgruppen vom Entstehungszeitpunkt bis zur Koinzidenz genau einer Gesamtlaufzeit der Delayline betragen muss.
- Das Ereignis kann nur innerhalb dieses kurzen Zeitfensters möglich sein. Dieses Zeitfenster wird durch den Anodenimpuls bestimmt, da er überhaupt Ursprung der Ereignisentstehung ist. Hierfür wird das 5. Schieberegister verwendet.

¹⁵ Typ: MC10125P [6]

¹⁶ Weit verbreitete Bauform eines Steckersystems für Datenverbindungen

Der Anodenimpuls wird eingelesen und bis an das Registerende durchgeschoben. Die letzte Registerzelle fließt durch eine zusätzliche Verundung mit in das Bewertungskriterium ein.

- Mehrfachkoinzidenzen müssen verworfen werden. Es lässt sich nicht feststellen, ob die Mehrfachkoinzidenz ein tatsächliches Mehrfachereignis oder nur ein tatsächliches Einfachereignis und zufällige Signalkoinzidenzen repräsentiert.

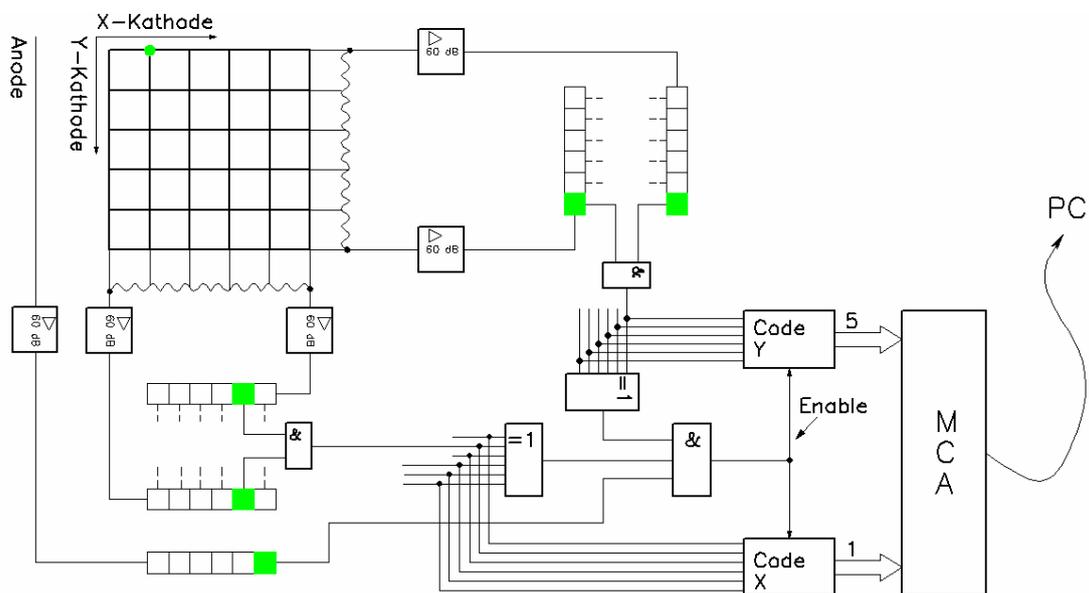


Bild 12: Freigabe der gültigen Signalkoinzidenz durch Anodenfenster

Kommt es unter Einhaltung der 3 logischen Bedingungen zu einer Koinzidenz, so wird ein *Enable-Signal* erzeugt, und eine 10-Bit Binärcodierung beider gesetzter Registerzellenindizes veranlasst.

Ein nachgeschaltetes Verfahren führt den Handshake mit dem MCA aus.

4.5 Aufbau der TDC- Verfahren

Die TDC- Verfahren zählen die Takte zwischen Start- und (intern verzögerten) Stoppsignalen . Das fertige Ergebnis liegt somit schon in der gewünschten Form vor. Anders als bei dem CSRD- Verfahren entfällt ein aufwendiger Kodierungsprozess. Lediglich bei der TDC1- Methode muss das Ergebnis durch eine einfache Shift rechts \gg Operation durch 2 dividiert werden. Liegen die Ergebnisse beider Kathoden vor, so wird der Handshake ausgeführt. Ebenfalls werden bei beiden Methoden die Anodenimpulse bzw. Ereignisse gezählt.

4.6 Handshake FPGA/MCA

Für die Ausführung des Handshakes zwischen dem FPGA¹⁷ und dem MCA¹⁸ ist kein zusätzliches Interface aufzuwenden. Die beiden logischen Schaltschwellen liegen in Spannungsbereichen , welche für beide Pegel eindeutig zu interpretieren sind. Für einen sicheren Datentransfer muss der MCA nach einem bestimmten Kommunikationsprotokoll angesprochen werden. Der MCA betrachtet das Empfangen der Datensätze (X und Y) als zwei eigenständige Prozesse. Wichtig zur eindeutigen späteren Zuordnung ist nur, dass beide Datensätze innerhalb eines bestimmten Zeitfensters empfangen werden. Dieses Zeitfenster wird auch als Koinzidenzfenster bezeichnet und ist gleich der Länge der Signallaufzeit einer Delayline.

¹⁷ CMOS- Level: LOW = 0.0 – 1.3V, HIGH = 3.7 – 5.0V (Eingang)

¹⁸ TTL- Level : LOW = 0.0 – 0.8V, HIGH = 1.5 – 5.0V (Eingang)

Ein eintreffendes *Deadtime* Signal gibt das Ende einer herkömmlichen TDC-Umsetzung bekannt. Der MCA ist nun auf den Empfang eines Datensatzes sensibilisiert.

Ein *DataReady* Signal meldet dem MCA, das ein Datensatz abholbereit ist. Nach erfolgreichem Einlesen der Daten quittiert der MCA mit einem *DataAccepted* Signal. Obwohl die Notwendigkeit des *Deadtime* Signals nicht zwingend ersichtlich ist, muss es auch für die innovative CSRD- Methode realisiert werden

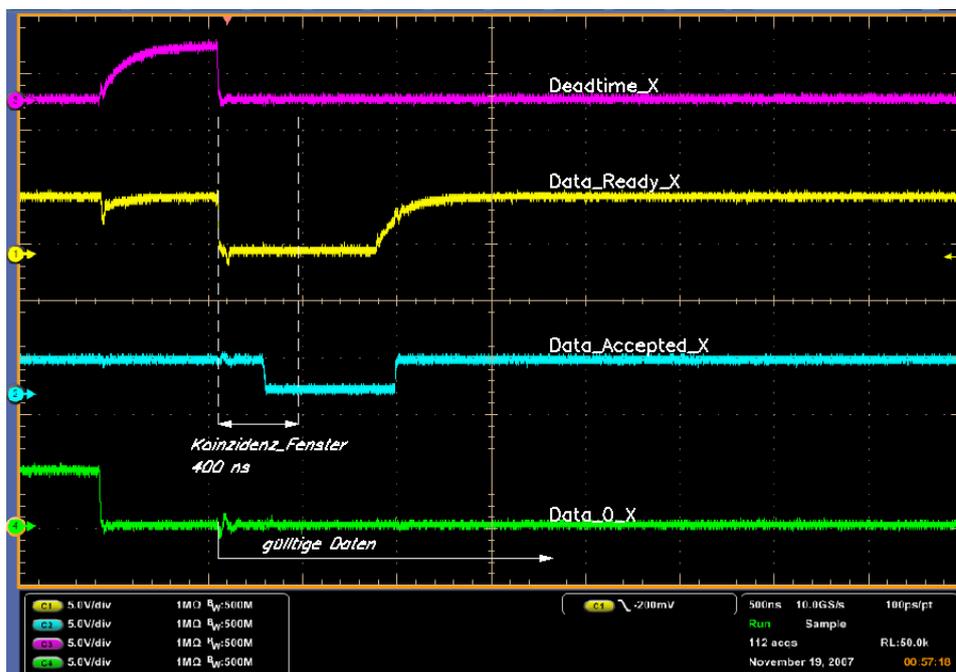


Bild 13: Oszillogramm des Handshakes für einen Datensatz, nach *Deadtime*-Signal freigabe

Das Oszillogramm zeigt den Handshake zwischen FPGA und MCA für einen Datensatz. Exemplarisch wird hier das LSB¹⁹ übertragen, welches entsprechend für den ganzen Datensatz steht.

¹⁹ hier Data_0_X

5 Software

Die gesamte Software für die drei Messmethoden ist in VHDL geschrieben worden. Als geeignetes Nachschlagewerk hat sich das Lehrbuch [10] erwiesen.

Für ein strukturiertes und fehlerfreies Arbeiten muss das Projekt systematisch und modularartig aufgebaut werden. Die Module werden als Komponenten bezeichnet und sollten einzeln auf Funktionalität getestet werden. Um eine Komponente zu erstellen, muss eine vhd- Datei erzeugt werden. Diese vhd- Datei kann nun über entsprechendes Portmapping beliebig häufig verwendet werden.

Mittels hierarchischer Instanziierung können beliebig hohe Verschachtelungstiefen realisiert werden. Der **Vorteil** dieser Herangehensweise liegt in der Wiederverwendbarkeit einzelner Komponenten und in der systematischen Fehlereingrenzbarkeit. Der **Nachteil** liegt in dem „Verdrahtungsaufwand“, der durch das Portmapping der vielen Komponenten entsteht.

Die Schnittstelle zum XILINX FPGA ist die oberste hierarchische Komponente. Sie wird als Toplevel bezeichnet. Da sich die beiden TDC Methoden vom Aufbau und der Umsetzung sehr stark ähneln, wird im Abschnitt 5.2 lediglich die schaltungstechnisch kompliziertere TDC- 1 Methode besprochen.

5.1 CSRD- Verfahren

Das Toplevel- Design bindet alle für die Umsetzung der Messmethode nötigen Komponenten ein. Im Folgenden soll der komplette Aufbau der CSRD- Methode mit seinen Besonderheiten dargestellt und erklärt werden. Die Erklärung erfolgt im direkten Bezug zu den Quelldateien.

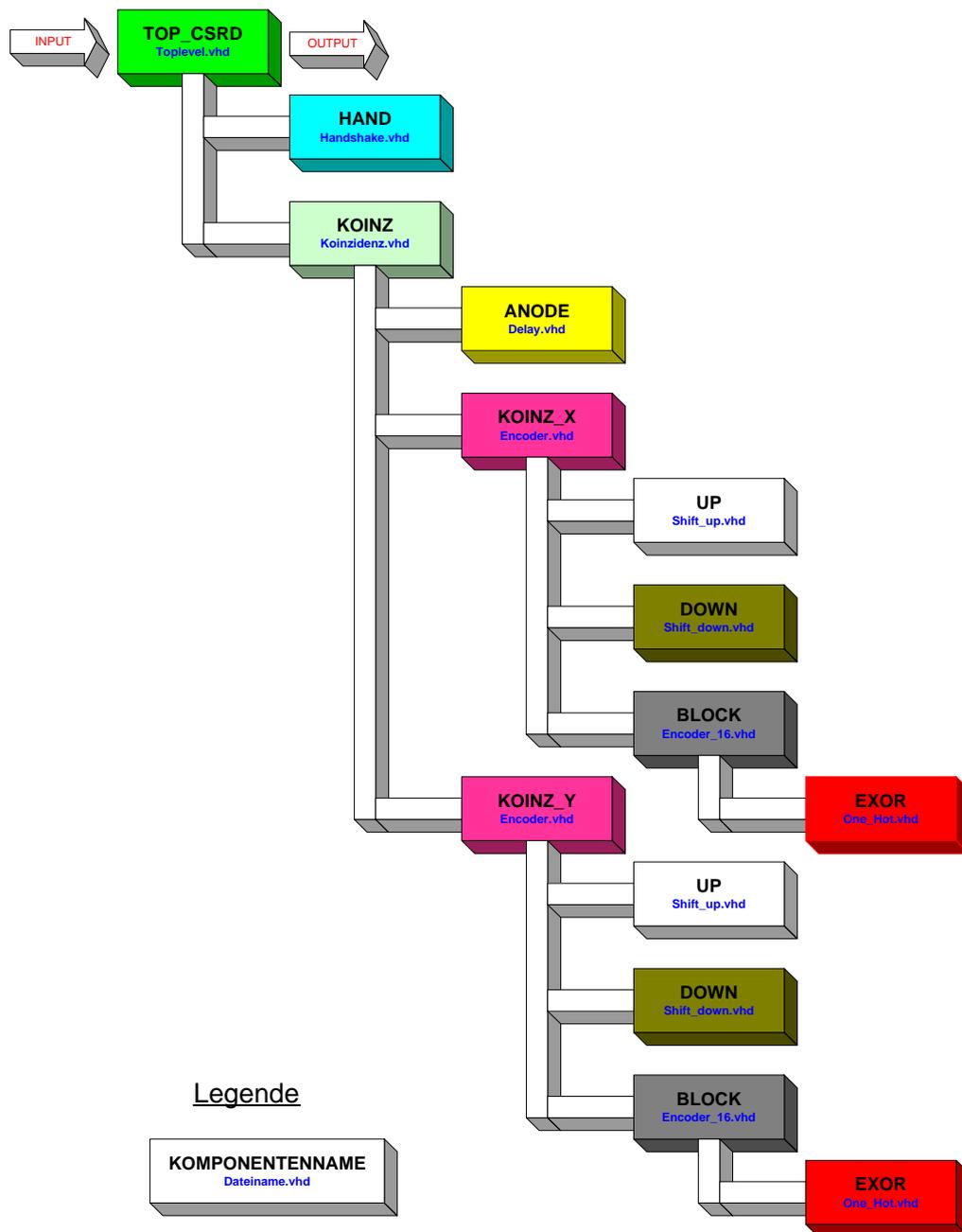


Bild 14: Hierarchischer Aufbau der CSRD- Methode (Blockbild)

Die farbliche Abtrennung im Blockbild soll die Wiederverwendbarkeit einzelner Komponenten hervorheben.

Es wird deutlich erkennbar, dass **KOINZ_X** und **KOINZ_Y** über die exakt gleiche Quelldatei (*Encoder.vhd*) eingebunden wurden.

5.1.1 Encoder.vhd

Wird in den Schieberegistern (*Shift_up.vhd* und *Shift_down.vhd*) eine Signalkoinzidenz festgestellt, so muss eine Kodierung des Registerzellenindex veranlasst werden. Diese Aufgabe übernimmt der allgemeingültig programmierte Encoder. Je nach eingestellter Registerlänge führt er eine entsprechende Binärcodierung durch. Die vorgeschaltete XOR- Stufe (*One_hot.vhd*) untersucht die Registerzellen auf gültige Einfachkoinzidenzen.

Bild 15 zeigt einen **ersten Denkansatz**, wie dieser eben beschriebene Encoder im Idealfall aussehen müsste. Die n Eingangssignale²⁰ werden ihrem Wert entsprechend durch ODER- Gatter binärcodiert. Nachgeschaltete UND-Verknüpfungen und die XOR- Stufe verhindern die Datenausgabe bei Mehrfachkoinzidenzen.

²⁰ entsprechend der Anzahl der Registerzellen (Auflösung)

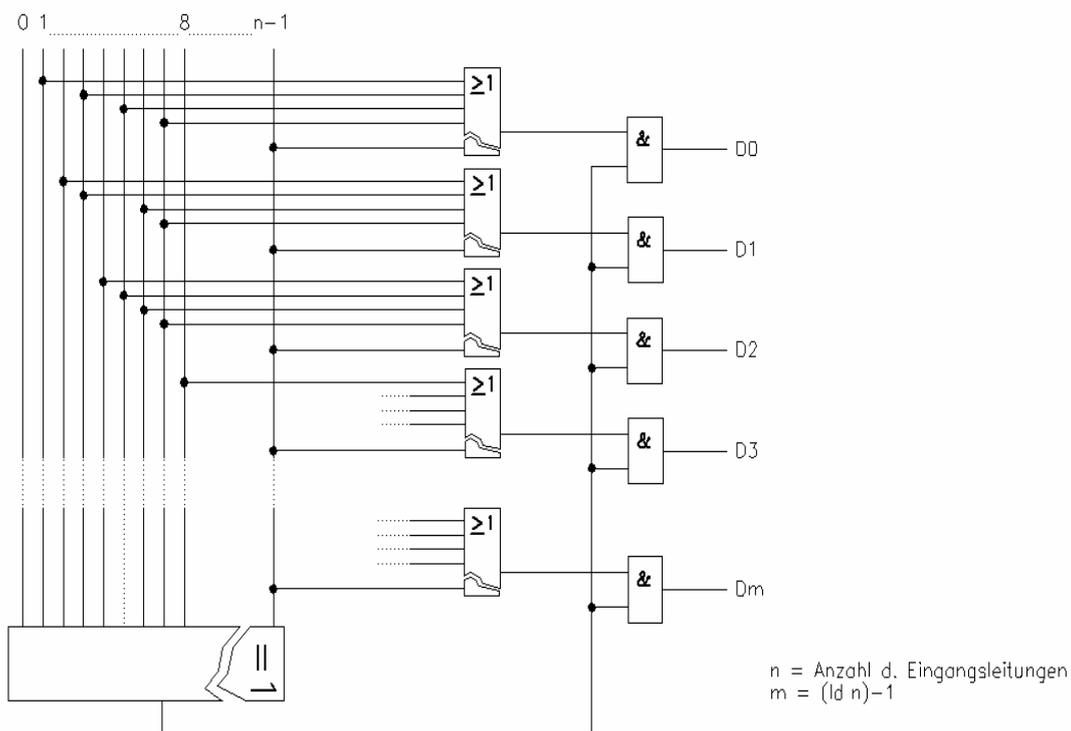


Bild 15: Aufbau eines idealen Encoders, jedoch nicht praxistauglich

Der **erste Ansatz**, diesen Encoder zu realisieren, erwies sich als **nicht praxistauglich**. Um bspw. 1024 Registerzellen kodieren zu können, wäre ein Hardwareaufwand von 10 ODERverknüpfungen mit jeweils 512 Eingängen notwendig.

Da solche riesigen ODERverknüpfungen im Spartan 3E nicht existieren, sondern nur logische Baugruppen mit bis zu 16- Dateneingängen verfügbar sind, müsste über eine Kaskadierung der einzelnen Unterbaugruppen die Gesamtfunktion realisiert werden. Durch die Kaskadierungen summieren sich aber die Gatterlaufzeiten. Hinzu kommen die Zeitverzögerungen durch den verhältnismäßig höheren chipinternen Verdrahtungsaufwand.

Letztendlich wird die Signallaufzeit so groß, dass die Taktanforderungen an die Gesamtschaltung nicht mehr eingehalten werden können.

Aus diesem Grund musste der Encoder neu durchdacht werden. Es ist ein **zweiter praxistauglicher Ansatz** entstanden:

- Der Encoder ist blockweise organisiert, wobei ein Block sich auf die Kodierung von max. 16 Registerzellen beschränkt (siehe Bild 17). Alle Blöcke werden parallel abgearbeitet, es treten keine Kaskadierungen mehr auf. Alle 16er Blöcke arbeiten nach dem gleichen Kodierungsverfahren. Sie unterscheiden sich nur darin, dass sie mit unterschiedlichen Basisadressen voreingestellt sind.

Jede Basisadresse ist immer ein ganzzahliges Vielfaches von 16. Je nach gesetzter Adresse kann so jeder beliebige Registerindex kodiert werden. Alle blockweise kodierten Ergebnisse werden auf einen gemeinsamen Datenbus geführt, welcher über Tri-State Treiber H bzw. L ²¹ getrieben wird. Der Bus wird immer nur dann niederohmig geschaltet, wenn nur die Kodierung eines 16er Blockes veranlasst wurde. Hierdurch wird sichergestellt, dass immer nur eine Einfachkoinzidenz auf den Bus geschaltet wird. Aus diesem Grund muss das *bus_enable* Signal des 16_er Blockes nach außen geführt werden, um durch die nachgeschaltete generisch programmierte XOR- Komponente auf Einfachübereinstimmung überprüft zu werden. Das Ergebnis ist immer dann gültig, wenn der Bus niederohmig ist.

²¹ H = High = logisch 1; L = Low = logisch 0

Durch die in Bild 16 dargestellten Tri- State- Anbindungen an den Bus werden die notwendige Oderverknüpfung aller n Eingangssignale vervollständigt, ohne dass wesentliche Signalverzögerungszeiten zu beklagen sind.

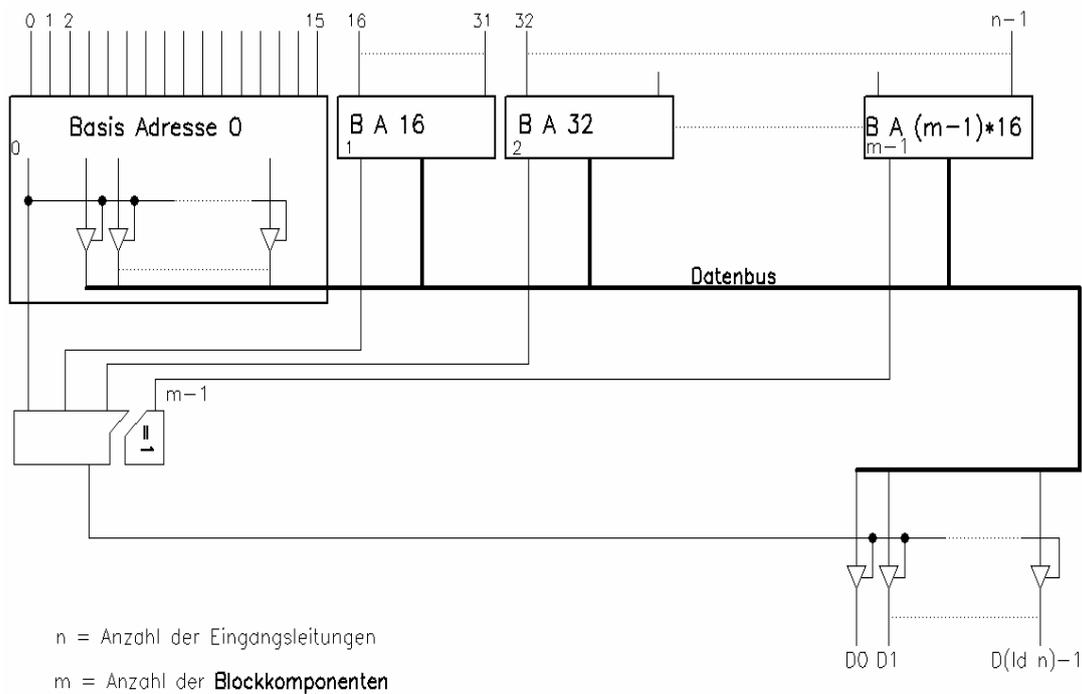


Bild 16: Gesamtdarstellung der blockweise organisierten Codiereinheit.

Im nächsten Bild wird der Aufbau der einzelnen Blockkomponente dargestellt, welcher in *Encoder_16.vhd* implementiert worden ist.

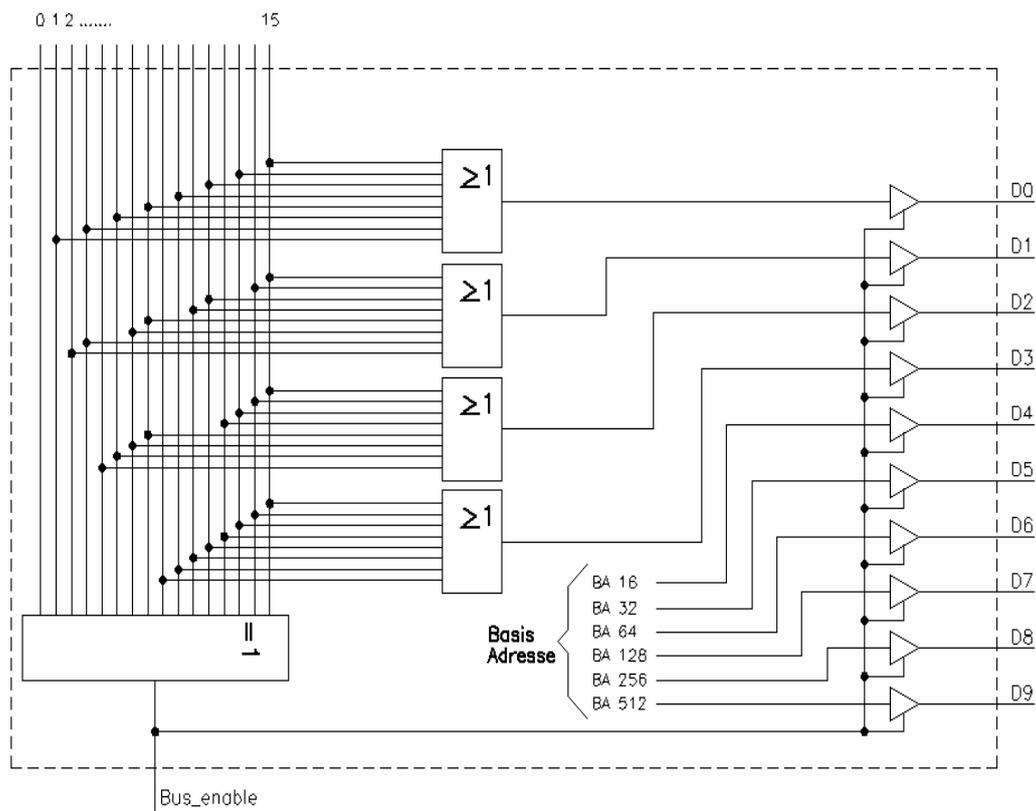


Bild 17: Darstellung einer einzelnen Blockkomponenten

Die einzelne Blockkomponente kodiert immer nur 16 Registerzellen. Für die Gesamtfunktion muss weiterer Schaltungsaufwand gemäß Bild 16 betrieben werden.

Vorteil dieses Kodierungsverfahren ist eine Taktoptimierung, da durch den modularen blockartigen Schaltungsaufbau die Ressourcen des FPGA's deutlich besser genutzt werden können. Durch die überschaubare Größe der einzelnen Komponenten werden keine unnötigen Kaskadierungen und Schaltungsveränderungen, bedingt durch die Architektur des FPGA's, bei der Synthese und beim Place and Route des Designs vollzogen. **Nachteil** dieses Verfahrens ist die relativ hohe Komplexität für ein eigentlich einfaches Problem. So müssen die einzelnen Blockkomponenten iterativ erzeugt und instanziiert werden.

Listing 1: Auszug aus Encoder.vhd (iterative Komponenteninstanziierung)

```

-- iterarive Komponenteninstanziierung:
-- Schleife wird so oft durchlaufen, wie
-- es 16-fache Registerzellenanteile gibt.
-- Der Rest wird gesondert instanziiert.
block_array:for I in 1 to (REG_LENGTH/16) generate
signal basic_adress_konv: std_logic_vector(9-1 downto 0);

begin
-- Konvertierung des Laufparameters I in eine
-- fest eingestellte Basisadresse vom Typ std_logic_vector
basic_adress_konv <= conv_std_logic_vector((I-1),9);
block_a: sechszehn_er_block
-- hier erfolgt die gezielte Kodierung der 16_er Blöcke
generic map ( LEN => 16)
port map
(
    data_a      => data_a_s ((I*16)-1 downto (I-1)*16),
    data_b      => data_b_s ((I*16)-1 downto (I-1)*16),
    bus_enable  => bus_enable_s(I-1),
    basic_address => basic_adress_konv,
    encode_out  => data
);
end generate block_array;

-- hier die Restinstanziierung der übrig gebliebenen
-- Registerzellen, welcher < 16 ist
-- Schleife wird nur einmal durchlaufen!!!
rest:for I in 0 to 0 generate
signal basic_adress_konv2: std_logic_vector(9-1 downto 0);
begin
basic_adress_konv2 <= conv_std_logic_vector((REG_LENGTH/16),9);
block_b : sechszehn_er_block
generic map(LEN => REG_LENGTH mod 16)

Port Map
(
data_a => data_a_s ((REG_LENGTH-1)downto((REG_LENGTH)-(REG_LENGTH mod 16))),
data_b => data_b_s ((REG_LENGTH-1)downto((REG_LENGTH)-(REG_LENGTH mod 16))),
bus_enable  => bus_enable_s(REG_LENGTH/16),
basic_address  => basic_adress_konv2,
encode_out  => data
);
end generate rest;

```

Da die tatsächliche Registerlänge mit $n = \text{REG_LENGTH}$ sich nicht vollständig ganzzahlig durch 16 teilen lässt, muss die Blockkomponentengenerierung in zwei unterschiedlichen Prozeduren erfolgen. In der ersten Prozedur werden in einer for-Schleife gemäß des ganzzahligen Divisionsanteils entsprechend viele 16_er_Blockkomponenten erzeugt. Die Kodierungslänge kann somit also mit ($\text{LEN} \Rightarrow 16$) fest eingestellt werden. Durch Inkrementierung des Laufindex I werden die Schieberegisterdaten (*data_a*, *data_b*) immer um 16_er-Schritte erhöht. Im Signalvektor *bus_enable_s* wird die Information gespeichert, ob **eine** Blockkomponente **ein** gültiges Ergebnis liefert.

Durch die Konversionsfunktion *conv_std_logic_vector* ((I-1),9) wird entsprechend des Indexwertes die Basisadresse eingestellt. Das Ergebnis wird durch *encode_out* auf den Datenbus (*data*) gelegt.

Die zweite Generierungsprozedur wird nur einmalig abgearbeitet. Sie verarbeitet nun die restlichen Registerzellen, welche einen Wert < 16 annehmen. Beide Prozeduren zusammen ermöglichen somit, jede beliebige Registerzellenanzahl kodieren zu können.

5.1.2 Simulationsergebnis des Encoders

Zur graphischen Veranschaulichung des beschriebenen Encoders dient ein Auszug aus dem Simulationsprogramm *Modelsim*²². Eine kostenlose aktuelle Version kann unter dem im Literaturverzeichnis angegebenen Link heruntergeladen werden. [11]

²² Sehr durchdachte Simulationsumgebung (spez. für Timing, Taktsynchronismus, usw.), welche in das ISE WebPack eingebunden werden kann

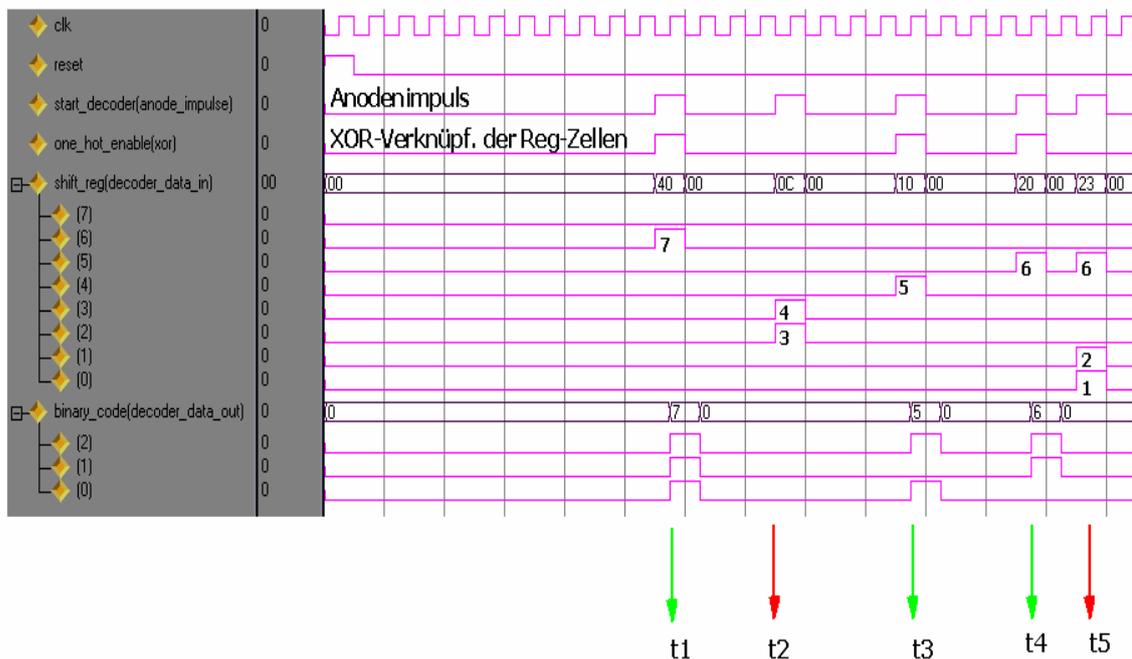


Bild 18: Simulationsergebnis des Encoders. Mehrfachübereinstimmungen werden verworfen (rot)

Die Zeitpunkte t1, t3 und t4 liefern gültige Ergebnisse, die gesetzte Registerzelle wird entsprechend kodiert. Zu den Zeitpunkten t2 und t5 treten Mehrfachkoinzidenzen auf, das Flag *one_hot_enable* wird nicht gesetzt, die Ausgänge bleiben auf NULL.

5.1.3 Delay_anode.vhd

Diese Komponente realisiert die Verzögerung des Anodenimpulses um die Länge der Signallaufzeit der Delayline. Diese Verzögerung wird mit einem Schieberegister realisiert. Der verzögerte Anodenimpuls fließt mit in die Koinzidenzbewertung ein. Er öffnet ein Zeitfenster, in welchem überhaupt mögliche Koinzidenzen stattfinden dürfen. Das Zeitfenster kann durch entsprechende Signalveroderungen (Disjunktionen) fein abgestimmt werden.

Listing 2: Auszug aus Dealy_anode.vhd (Realisierung des Schieberegisters)

```

if reset = '1' then
    anode_s      <= (others => '0');
    -- abarbeitung bei steigender Taktflanke
    elsif rising_edge(clk) then
        -- Registerwert einlesen
        if anode_impulse = '1' then
            anode_s(REG_LENGTH-1) <= '1';
        else
            anode_s(REG_LENGTH-1) <= '0';
        end if;
        -- Schieben
        for I in REG_LENGTH-1 downto 1 loop
            anode_s(I-1) <= anode_s(I);
        end loop;
    end if;
end process;

-- Hier wird die Größe des Anodenfensters bestimmt.
-- Aktuell ist das Anodenfenster zwei Takte breit.
-- Durch zusätzliche Zellveroderungen, kann das
-- Anodenfenster beliebig verändert werden.

anode_delay <= anode_s(0) or anode_s(1); -- Feinabstimmung des Koizidenzfensters

```

Das verzögerte Anodensignal wird durch folgende Signalzuweisung $anode_delay \leq anode_s(0) \text{ or } anode_s(1)$ an den Ausgang der Schnittstelle geführt.

5.1.4 Koizidenz.vhd

Diese Komponente leitet den Handshake mit dem MCA ein. Voraussetzung hierfür ist ein zeitnahes/gleiches Eintreffen gültig umgesetzter X/Y- Werte und ein zeitnahes/gleiches Eintreffen des verzögerten Anodensignals $anode_delay$. Die Daten werden dann einen Takt lang an den Ausgang der Schnittstelle geführt, und es wird ein Ausgangsflag $enable_x_y$ gesetzt.

5.1.5 Handshake.vhd

Erkennt die Komponente das gesetzte *enable_x_y* Flag, so werden die nur kurz anliegenden Daten in einen Buffer eingelesen und der Handshake ausgeführt. Die Ausführung erfolgt mittels eines Moore- Automaten, welcher über 7 unterschiedliche Zustände verfügt²³. Für die Einhaltung eines strikten Zeitverhaltens müssen 2 zusätzliche Zähler verwendet werden.

5.1.6 Toplevel.vhd

Das Toplevel- Design bindet alle zuvor beschriebenen Komponenten ein und liefert die Schnittstelle nach außen. Des weiteren wird im Toplevel ein verlangsamter Arbeitstakt bereitgestellt, welcher individuell auf die Signallaufzeit der Delayline des Detektorsimulators abgestimmt sein muss. Der Handshake läuft einzig mit dem internen Boardtakt²⁴ ab.

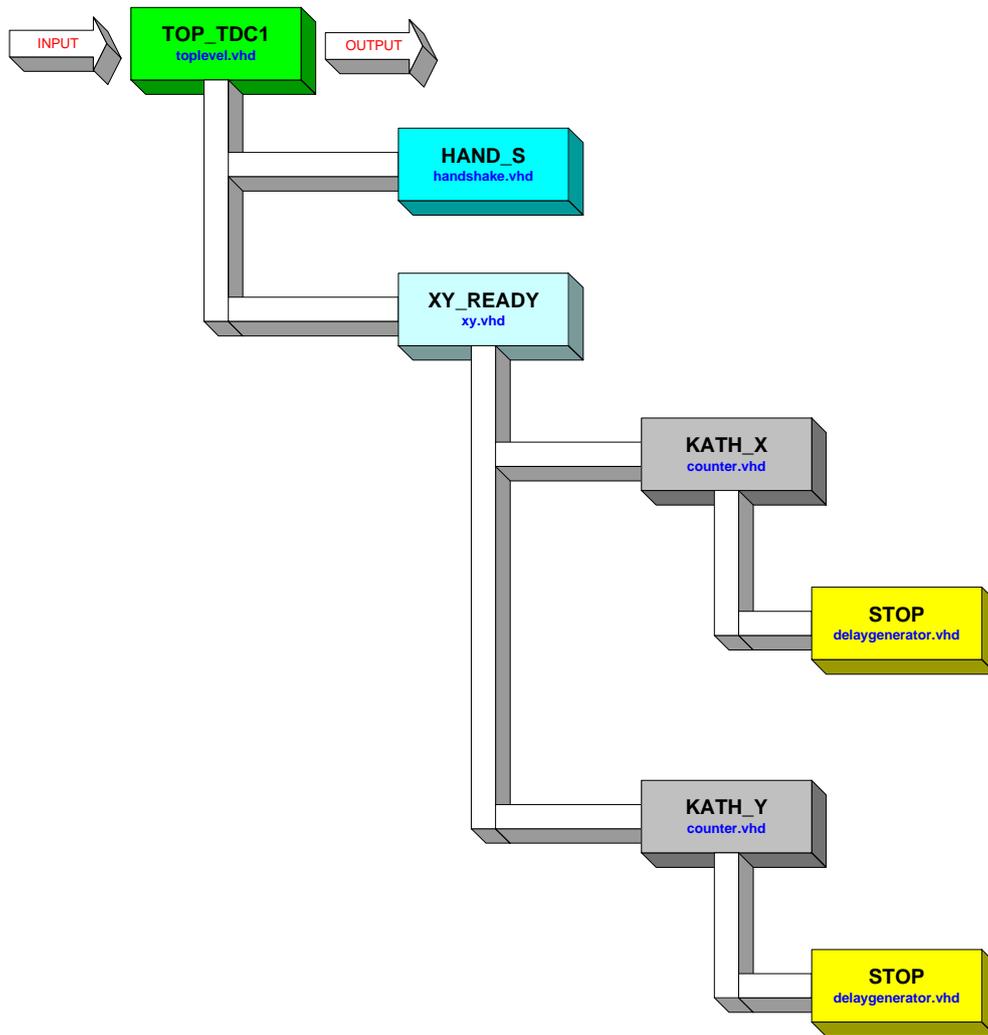
Für die prozentuale Auswertung der erfassten Ereignisse werden die Anodenimpulse in einem gesonderten Prozess mit einem 24- Bit Zähler gezählt. Mittels selektiver Signalzuweisung kann der Zählerstand jederzeit kaskadiert über LED's zur Anzeige gebracht werden. Die Umschaltung erfolgt über boardinterne Schalter.

5.2 TDC1- Verfahren

Im folgenden Abschnitt soll nun der Aufbau des TDC1- Verfahrens besprochen werden. Es gibt gewisse Parallelen zum Aufbau der CSR- Methode. So können die Handshake- und Toplevelkomponente bspw. direkt von der CSR- Methode übertragen werden.

²³ Beim Erreichen eines Zustandes erfolgt eine entsprechende Reaktion am Ausgangsschaltnetz. Die Konstellation der Eingangssignale ist hierbei bedeutungslos (anders beim Mealy- Automaten)

²⁴ 50 MHz Quarzoszillator



Legende



Bild 19: Hierarchischer Aufbau des TDC1- Verfahrens (Blockbild)

Die beiden Komponenten **KATH_X** und **KATH_Y** entspringen ein- und derselben Quellcodedatei (*counter.vhd*). Die darin eingebundene **STOP**- Komponente, welche für die Verzögerung des Stopimpulses zuständig ist, entspricht dem Aufbau gemäß Abschnitt 5.1.3. Sie trägt lediglich einen andere Namen und wird von einem anderen Signal gespeist.

5.2.1 Counter.vhd

Diese Komponente besteht im wesentlichen aus einem Moore- Automaten und einem Zähler. Der Zähler zählt die Takte zwischen einem Start- und einem verzögerten Stoppsignal (*Delaygenerator.vhd*). Der Automat ist für die korrekte Einhaltung zuständig. Ähnlich wie bei einer Impulsfolgeerkennung [12], achtet er auf nacheinander folgendes Eintreten von Anoden-, Start- und Stoppsignal. Wird diese Reihenfolge nicht eingehalten, so kann der Automat entsprechend reagieren²⁵. Das fertige Ergebnis ist der Zählerstand selbst und wird durch Setzen eines *enable* Flags angezeigt. Es entfällt ein aufwendiger Kodierungsprozess.

5.2.1 Xy.vhd

Um den Handshake gezielt einleiten zu können, müssen die Zählergebnisse beider Kathoden vorliegen. Diese zuletzt vorgestellte Komponente wartet auf das Eintreffen beider *enable* Flags und schreibt die Ergebnisse in einen Buffer. Erst dann wird einen Takt lang das *enable_x_y* Flag gesetzt, welches den Handshake veranlasst.

²⁵ Häufig eintretender Fall bei statistischer Ereignisrate

6 Experimentell gewonnene Ergebnisse

Im letzten Kapitel sollen alle 3 Messmethoden miteinander verglichen werden, um die Stärken der CSR-D- Methode zu zeigen. Um eine aussagefähige Messreihe zu bekommen, wird mit dem Detektorsimulator die ganze Bandbreite der statistischen, theoretischen Ereignisraten durchfahren. Alle 3 Methoden werden über den gleichen Zeitraum zu gleichen Bedingungen an dem Simulator erprobt.

Durch eine absolute Fehlerzählung und Zählung der tatsächlich auftretenden und erfassten Ereignisse lässt sich eine prozentuale Aussage über den Fehler und die Ausbeute treffen. Des weiteren soll die Wichtigkeit der Anodenbewertung bei der CSR-D- Methode bewiesen werden. Abschließend soll mit der gewonnenen Messreihe der Arbeitspunkt der CSR-D- Methode näher untersucht werden.

6.1 Messaufbau

Um den absoluten Fehler messtechnisch erfassen zu können, muss ein weiterer PIC-Controller zum Einsatz kommen. Dieser Controller ist parallel zu den Datenleitungen zwischen FPGA und MCA geschaltet. Er greift nicht in den bestehenden Datenverkehr ein, sondern vergleicht die zu übertragenden Daten mit eigens vorgegebenen Daten.

Bei Übereinstimmung wird ein Parameter *treffer* inkrementiert. Bei Nichtübereinstimmung ein Parameter *fehler*. Beide Parameter werden zur Anzeige gebracht. Um eine leichte Handhabung der Fehlerüberwachung zu gewährleisten, senden alle drei PIC- Controller des Simulators die gleiche geometrische Figur (Rechteck).

Der Vorteil besteht darin, dass die Überwachungseinheit lediglich über die 4 Eckpunkte des Rechteckes verfügen muss. Alle anderen Datenpunkte liegen auf den äußeren 4 Geraden zwischen den Eckpunkten. Sie können somit rechnerisch schnell ermittelt werden (Quellcode siehe Anhang C).

Um einen gewissen Anzeigekomfort bei den tatsächlichen Ereignissen zu besitzen, werden die Anodenimpulse durch einen handelsüblichen Zähler aus dem Hause Hawlet Packard gezählt²⁶.

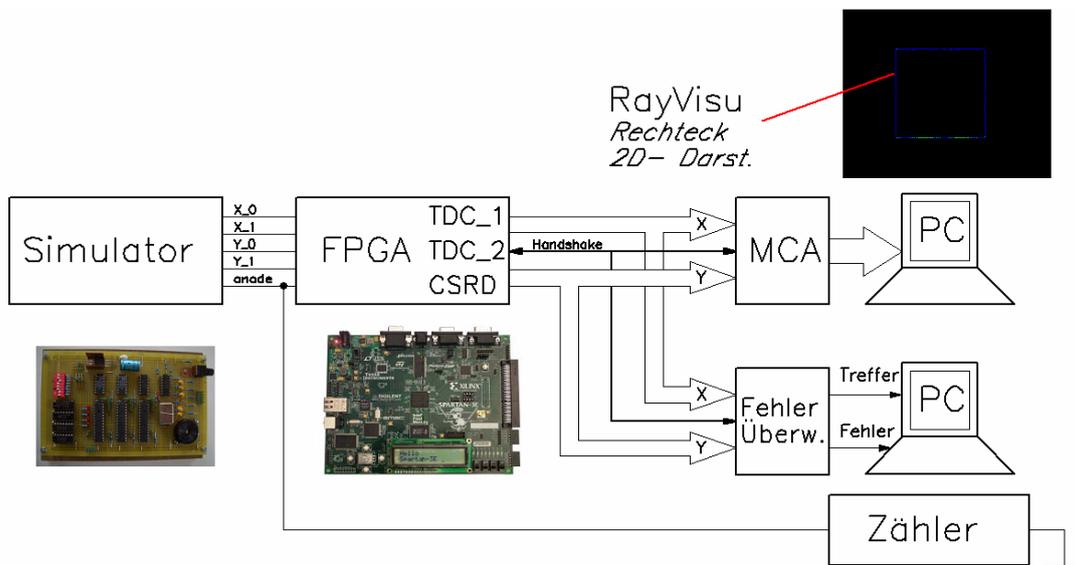


Bild 20: Blockbildartiger Messaufbau mit Fehlerüberwachung

Um eine Vorstellung vom praktischen Messaufbau zu bekommen, stellt Bild 21 den tatsächlichen Aufbau dar.

²⁶ Eigentlich nicht notwendig, da FPGA über entsprechenden Zähler + Anzeigemechanismus verfügt

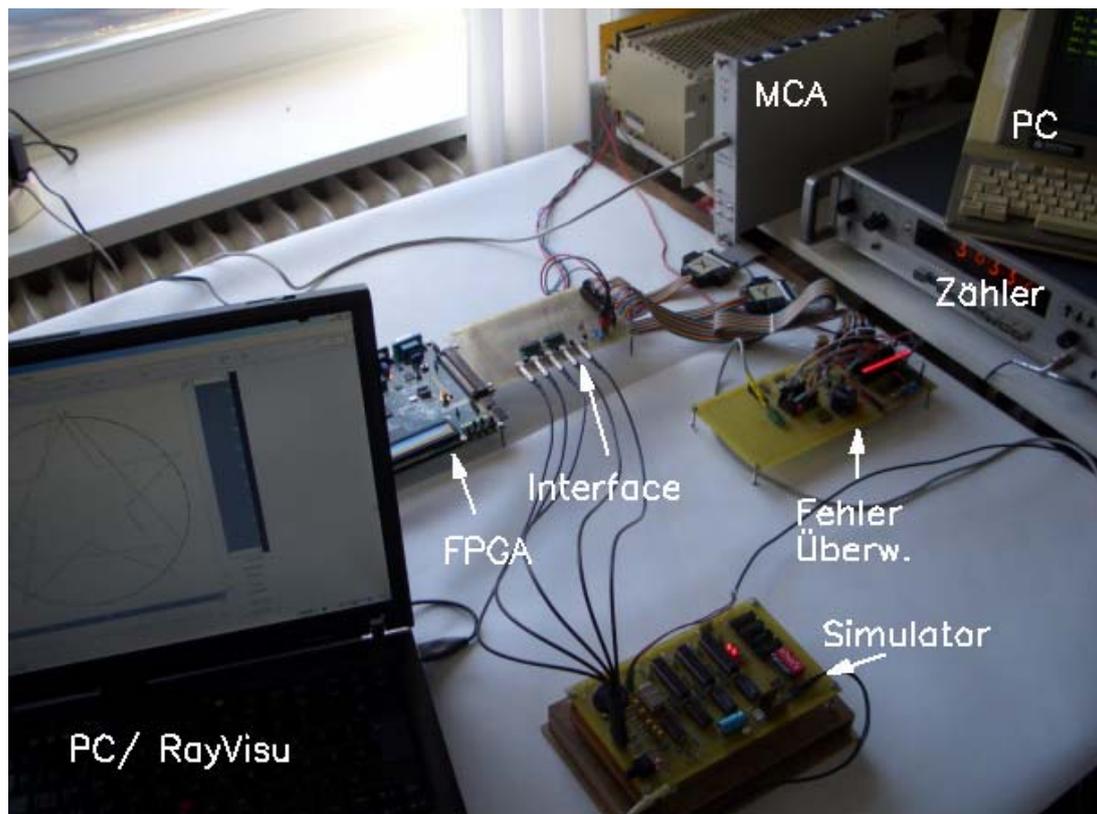


Bild 21: Tatsächlicher Messaufbau mit Fehlerüberwachung

Anders als zuvor beschrieben, senden die 3 Controller des Detektorsimulators unterschiedliche geometrische Figuren (Stern, Kreis, Dreieck). Das 2D- Bild ist durch den PC (RayVisu) dargestellt.

6.2 Vergleichende Untersuchungen im Frequenzband

In vorangegangenen unprotokollierten Messungen stellte sich heraus, dass durch Einstellung des Impulsbreitenverhältnisses ($X1/X0 = Y1/Y0 = 2$) die Ausbeute bei der CSR- Methode deutlich am höchsten ist.

Aus diesem Grund ist die folgende Messreihe aller 3 Methoden mit dieser Einstellung durchgeführt worden. Die theoretische integrale Ereignisrate am Detektor errechnet sich durch :

$$\text{theor. Ereignisrate} = \frac{\text{Ereignisse}}{\text{Messzeit}} * \frac{t_{\text{Pseudodelayline}}}{t_{\text{Detektordelayline}}} \quad (4)$$

Diese Angabe ist wichtig, um den Bezug zur Wirklichkeit herzustellen.

Der prozentuale Fehler errechnet sich durch:

$$\text{Fehler} = \frac{\text{Fehler}}{\text{Ereignisse}_{\text{MCA}_{\text{erfasst}}}} * 100 \quad / \% \quad (5)$$

Die prozentuale Ausbeute errechnet sich durch:

$$\text{Ausbeute} = \frac{\text{Ereignisse}_{\text{MCA}_{\text{erfasst}}}}{\text{Ereignisse}} * 100 \quad / \% \quad (6)$$

Mit diesen Erkenntnissen lässt sich die Messreihe aufnehmen und die Ergebnisse tabellarisch (siehe Anhang A) und graphisch darstellen. Alle nachfolgend dargestellten Messreihen sind bei 10.000 Ereignissen aufgenommen worden.

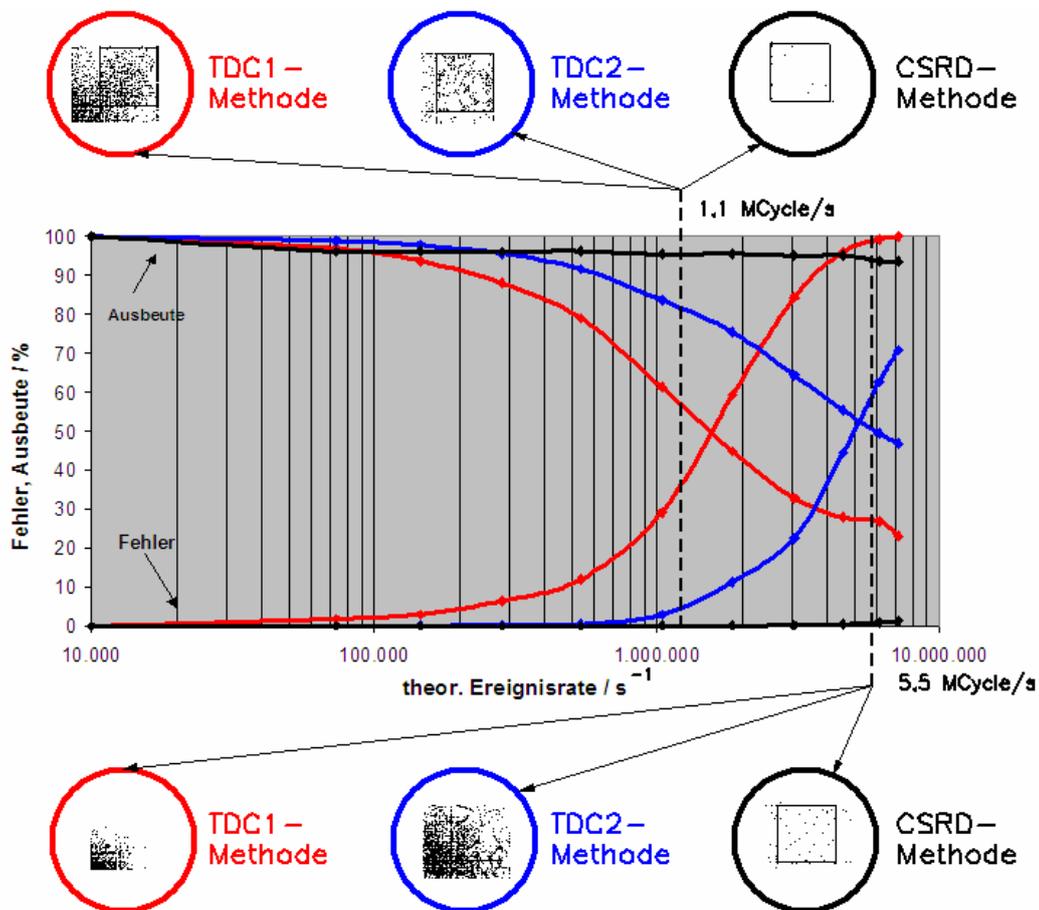


Bild 22: Vergleichende Darstellung der 3Methoden bei unterschiedlichen Ereignisraten

In den 6 Kreisen werden die 2D- Koordinaten der durch die Messmethoden unterschiedlich umgesetzten Signale dargestellt. Im Idealfall sollten die umgesetzten Daten wieder die Umrissform eines Rechteckes ergeben, welches im RayVisu- Bild des Messaufbaus (Bild 20) bereits dargestellt ist. Es fällt beim ersten Betrachten der Grafik sofort auf, dass die CSR- Methode mit Abstand die fehlerfreiesten Ergebnisse liefert.

Die TDC1- Methode, welche die höchste Totzeit besitzt, liefert erwartungsgemäß die schlechtesten Ergebnisse. Bereits bei einer theoretischen Ereignisrate von ca. $1,5 \cdot 10^6 \text{ Cycle/s}$ ist die Ausbeute auf 50% gesunken und der Fehler auf 50% angestiegen.

Es ließen sich bei dieser verhältnismäßig geringen Rate keine verwertbaren Daten mehr sammeln. Folglich ist diese Methode nur im niederenergetischen Bereich für praktische Messanwendungen geeignet.

Die blau gekennzeichnete TDC2- Methode besitzt ein Drittel der Totzeit der TDC1- Methode. Sie liefert vergleichsweise schon wesentlich bessere Ergebnisse. Hier liegt der 50%- Schnittpunkt der Fehler- und Ausbeutekurve bei einer Rate von ca. $5 \cdot 10^6 \text{ Cycle/s}$. Bei einer Rate von ca. $2,5 \cdot 10^6 \text{ Cycle/s}$ können noch verwertbare Daten gesammelt werden, welches bei der TDC1- Methode absolut unmöglich ist.

Die „fast“ totzeitlose CSRD- Methode schneidet mit Abstand am besten ab. Sie besitzt einen ganz anderen charakteristischen Kurvenverlauf. Anders als bei beiden TDC Methoden führt es hier nicht zum Schnittpunkt zwischen Fehler- und Ausbeutekurve. Vielmehr kommt es bei steigender Ereignisrate zu einem fast achsenparallelen Kurvenverlauf, wobei die Ausbeutekurve annähernd auf 95% verweilt. Der Fehler bleibt mit 0-1 % vernachlässigbar klein.

6.3 Untersuchung der Anodenbewertung (CSRD)

Um die Wichtigkeit der Anodenbewertung in der Ereigniszuordnung zu beweisen, ist eine Einzelmessreihe der CSRD- Methode durchgeführt worden. Die Messbedingungen entsprachen hierbei denselben wie unter Abschnitt 6.2 beschrieben. Die CSRD Methode ist dahingehend verändert worden, dass das enge

Zeitfenster des Anodenimpulses nun nicht mehr relevant für eine gültige Signalkoinzidenz ist.

Gültige Koinzidenzen können nun zu jedem beliebigen Zeitpunkt auftreten. Zu erwarten ist hierdurch eine sehr hohe Fehlerquote, bezogen auf die absoluten Ereignisse (über 100%).

Die Ergebnisse sind in tabellarischer Form dem Anhang B zu entnehmen und stellen sich in graphischer Form wie folgt dar:

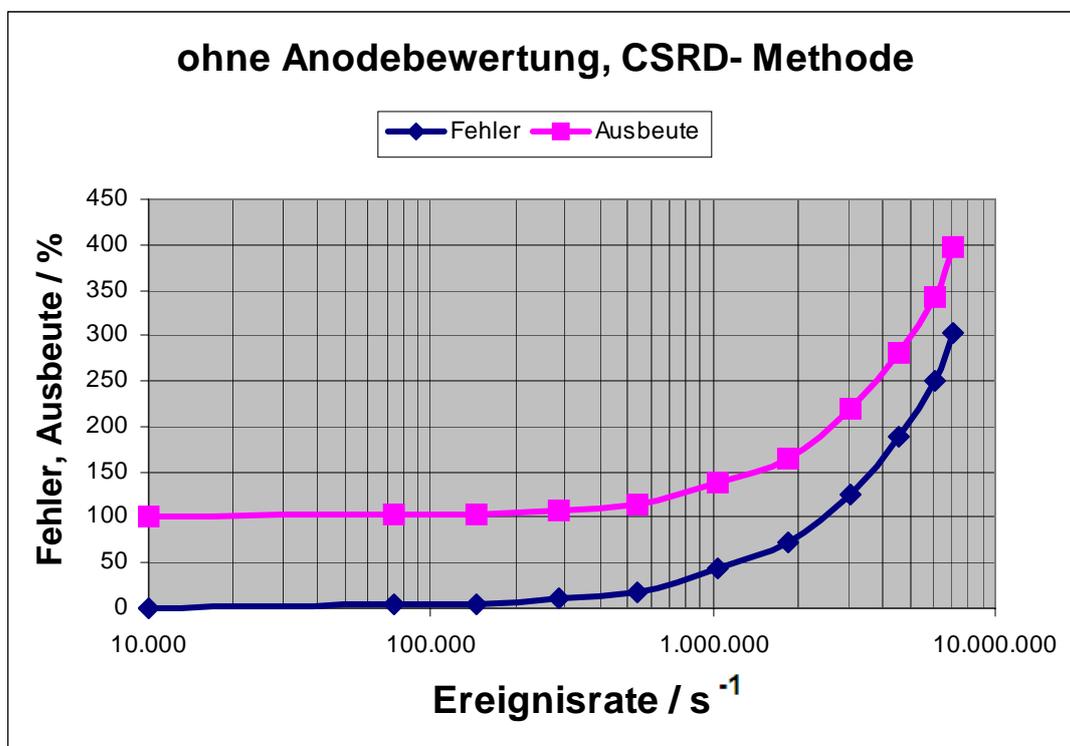


Bild 23: Einzelmessung der CSRD- Methode (ohne Anodebewertung) bei unterschiedlichen Ereignisraten

Anders als in Bild 22 dargestellt errechnet sich die Fehlerquote über die absoluten Ereignisse und nicht über die erfassten Ereignisse.

$$Fehler = \frac{Fehler}{Ereignisse} * 100 \quad \text{/\%} \quad (7)$$

Hierdurch soll gezeigt werden, dass die scheinbar hohe Ausbeutequote stark durch den Fehler beeinflusst wird. Beide Kurven besitzen im logarithmischen Zeitmaßstab einen exponentiellen Verlauf, die Fehlerquote erreicht bereits bei einer Rate von $2,5 * 10^6 \text{ Cycle/s}$ die 100% Grenze. Insgesamt wird eine Fehlerquote von 300 % erreicht.

Dieses Ergebnis beweist, dass das Anodenfenster eine enorm wichtige Funktion bei der CSRD- Methode hat. Ein richtig gewähltes Fenster lässt den Fehler nahezu bei 0-1% bleiben (siehe Bild 19). Bei Weglassen des Anodenfenster schnellte der Fehler exponentiell in die Höhe.

6.4 Untersuchung des Arbeitspunktes (CSRD)

Um ideale Messergebnisse mit der CSRD Methode zu erzielen, muss der Arbeitspunkt optimal eingestellt werden. Der Arbeitspunkt bezieht sich auf das Impulsbreitenverhältnis und die Frequenztoleranz. Mit Frequenztoleranz ist die prozentuale Abweichung des digitalen Systemtaktes von der Frequenz, die durch die tatsächliche Laufzeit der Delayline begründet wäre. An dieser Stelle soll noch einmal ausdrücklich erwähnt werden, dass die bereits getätigten Messreihen alle unter Einhaltung des optimalen Arbeitspunktes durchgeführt wurden. Der Leser soll nur gezielt auf die bestehende Problematik hingewiesen werden.

Anders als bei den vorangegangenen Messungen wird bei der Arbeitspunktbestimmung nicht die ganze Bandbreite aller Ereignisraten

durchfahren. Stattdessen wird eine feste Ereignisrate von $1,85 \cdot 10^6 \text{ Cycle/s}$ eingestellt. Durch schrittweise Veränderungen der Impulsbreitenverhältnisse lässt sich die Messreihe tabellarisch (siehe Anhang B) und graphisch aufnehmen.

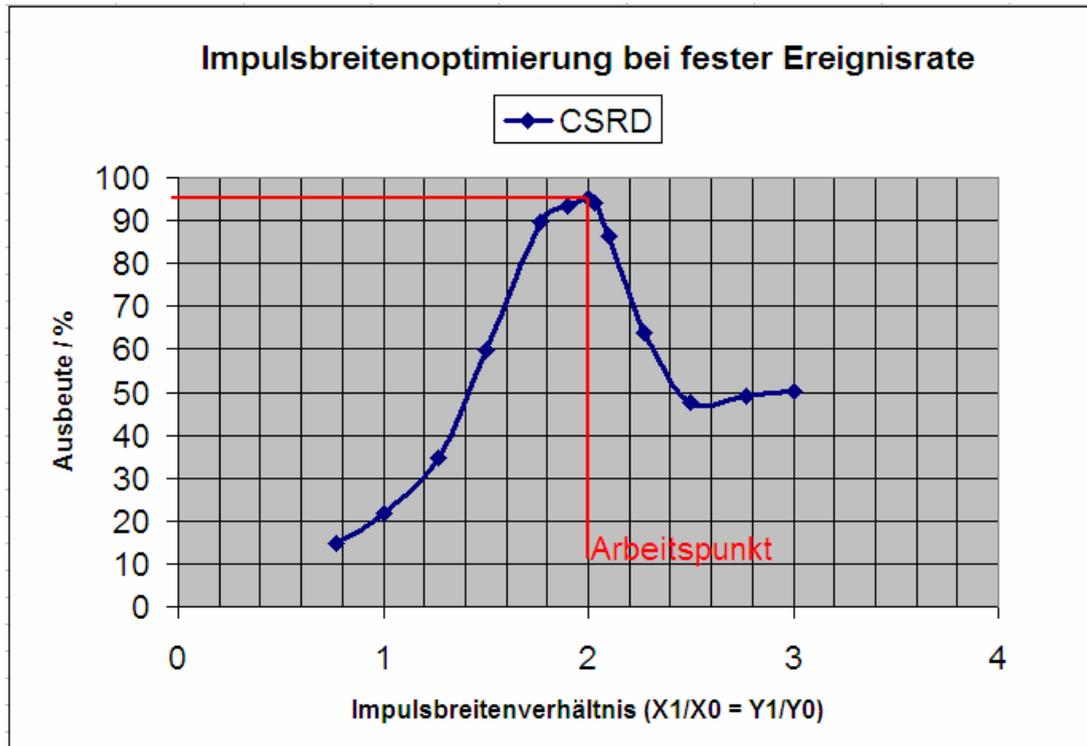


Bild 24: Optimierung der Ausbeute bei gleich bleibender Ereignisrate und veränderlichem Impulsbreitenverhältnis (CSRD)

Eine maximale Ausbeute von 95 % wird bei einem Impulsbreitenverhältnis von 2 erzielt. Dieses ist die Erkenntnis, welche für diese Messreihe wichtig ist. Dass die Ausbeute bei einem Verhältnis > 2 wieder sinkt, ist nicht zwingend einzusehen. Dieses liegt aber daran, dass nun des öfteren Mehrfachkoinzidenzen auftreten. Mehrfachkoinzidenzen können fehlerbehaftet sein und werden von der CSRD-Methode verworfen. Für niedrigere Verhältnisse < 2 gilt genau der umgekehrte Fall.

Sind die Impulse zu schmal, können mögliche Koinzidenzen verpasst werden. Hierdurch sinkt ebenfalls die Ausbeute.

Abschließend soll nun in einer letzten Messreihe die Frequenztoleranz bei fester Ereignisrate untersucht werden. An dieser Stelle muss nun erwähnt werden, dass der Detektorsimulator über eine veränderbar einstellbare Verzögerungszeit der Pseudodelayline verfügt. Sie kann in sehr feinen Schritten verändert werden. Diese Möglichkeit gäbe es bei einer echten Detektordelayline nicht.

Es ist für die folgende Messreihe bedeutungslos, ob die Taktfrequenz bei gleich bleibender Delayline verändert wird oder ob die Delayline bei gleich bleibender Taktfrequenz verändert wird. Aus Praxisgründen wurde bei der Frequenztoleranzmessung aber die Pseudodelayline in feinen Schritten verändert. Die Ergebnisse sind in tabellarischer Form dem Anhang B zu entnehmen. Die graphische Ergebnisdarstellung sieht wie folgt aus:

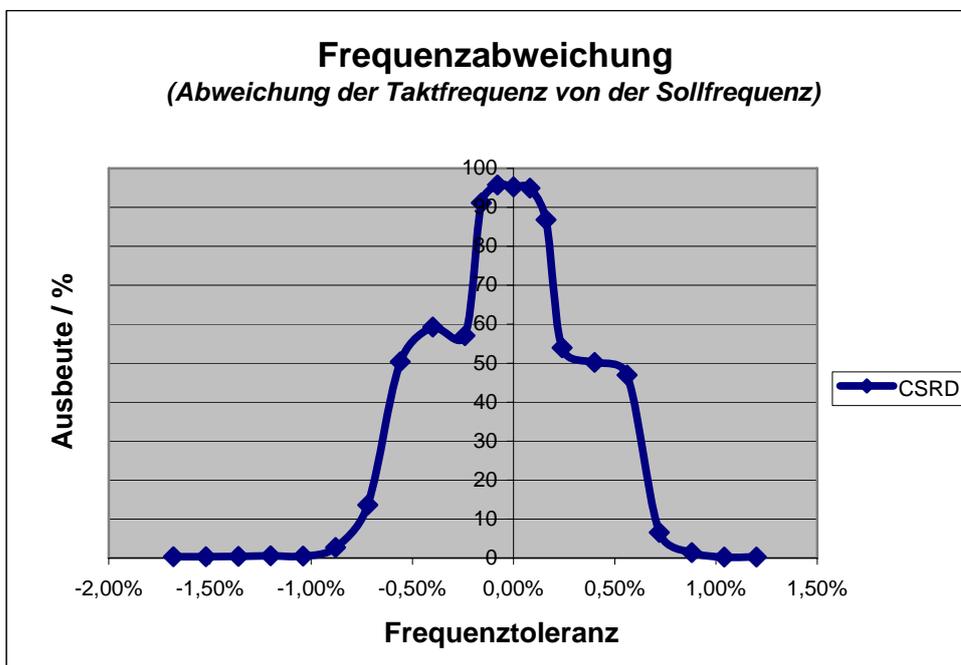


Bild 25: Takteingrenzung zum Erhalt der Maximalausbeute

Die CSRD Methode ist eine sehr takt sensible Messmethode. Bereits bei kleinen Abweichungen der Taktfrequenz zur Sollfrequenz ($\pm 0,1\%$), nimmt die prozentuale Ausbeute rapide ab und verliert ihr Maximum. Ab einer Fehlertoleranz $\langle \rangle \pm 1\%$ strebt die Ausbeutekurve gegen 0%. Dieser Kurvenverlauf bezieht sich auf eine Ereignisauflösung von 500*500 Kanälen. Bei einer höheren Auflösung würde das Plateau des Maximums wahrscheinlich insgesamt schmaler werden und bei einer niedrigeren Auflösung insgesamt breiter. Dieses sind allerdings nur Mutmaßungen die messtechnisch nicht bewiesen wurden.

Fazit dieser Messreihe wäre dann, je höher die gewünschte Auflösung der CSRD Methode ist, desto genauer muss die Taktfrequenz an die Sollfrequenz angepasst sein.

7 Zusammenfassung und Ausblick

Diese Arbeit soll Aufschluss über die tatsächliche Machbarkeit eines innovativen Detektormessverfahrens (CSR-D) liefern. Hierfür wurde das CSR-D- Verfahren eigens hardwaretechnisch entwickelt. Es wurde entsprechend auf alle zu untersuchenden Gesichtspunkte für eine optimale Ausnutzung des Verfahrens eingegangen.

Da die Taktanforderungen der zu verarbeitenden Detektorsignale sehr hoch sind, konnten die abschließenden Messreihen nur mit Hilfe eines niederfrequenteren Detektorsimulators durchgeführt werden. Um vergleichende Untersuchungen mit gegenwärtig üblichen Messmethoden durchführen zu können, mussten diese entsprechend hardwaretechnisch nachgebaut und an den Simulator angepasst werden.

Alles in allem hat die CSR-D- Methode höchst überzeugende Ergebnisse geliefert, wie dem letzten Kapitel dieser Ausarbeitung zu entnehmen ist. Gerade bei höheren Ereignisraten, bei welchen die herkömmlichen Messverfahren komplett versagen, zeigt sich die volle Stärke der CSR-D- Methode. Diese vorhergegangenen Theorie ist mit dieser Diplomarbeit nun experimentell bewiesen worden.

Die in der Einleitung aufgeworfene Frage nach einer lohnenden Weiterverfolgung des Projektes kann folgendermaßen beantwortet werden:

- Um in der Zukunft eine realistische Auflösung von 1024×1024 Kanälen mit der CSR-D- Methode an einem echten Detektor zu erzielen, müsste der Arbeitstakt der Messschaltung ca. 2GHz betragen. Ob diese Taktanforderungen mit einem FPGA realisiert werden können, möchte ich zum heutigen Stand der Technik stark in Frage stellen.

Da während des gesamten Verlaufs dieser Diplomarbeit ein ständiger theoretischer Weiterentwicklungsprozess an die Takt- und Auflösungsanforderungen stattfand, könnte dieses eben beschriebene Problem wie folgt gelöst werden:

- Die 4 Kathodensignale und das Anodensignal sind über externe Minimaldelaylines²⁷ seriell in ein n-Bit²⁸ breites Wort einzulesen. Anstatt wie bei der CSRD- Methode jeweils nur ein Informationsbit in den gegenläufigen Registern zu schieben, wird nun das ganze n-Bit breite Wort parallel und gegensinnig geschoben. Hierdurch können gemäß der einfachen Methode ebenfalls Koinzidenzen festgestellt werden. In diesem neuen Fall wären die Koinzidenzen bitmusterabhängig, und müssten durch einen entsprechenden mathematischen Algorithmus bewertet werden.

Vorteil dieses neuen Parallelverfahrens wäre, dass die Auflösung proportional zur Breite des n- Bit Wortes zunimmt, ohne dass die Taktfrequenz erhöht werden müsste.

Nachteil dieses neuen Verfahrens wäre der zusätzliche Schaltungs- und Logikaufwand, der gravierend zunehmen würde.

Abschließend sei festzuhalten, dass das eben beschriebene Parallelverfahren aus heutiger Sicht die einzige realistische Möglichkeit ist, diese CSRD- Messmethode im vollen Umfang umsetzen zu können. Es bietet sich an, eine weitere Diplomarbeit über die hardwaretechnische Umsetzung dieser modifizierten CSRD- Methode anfertigen zu lassen. Die Rahmenbedingungen sollten dabei nicht verändert werden.

²⁷ verfügen über bspw. 7 statt 199 Steps, je nach gewünschter Auflösungshöhe

²⁸ Multiplikator der Drahtauflösung

8 Literaturverzeichnis

[1] Forschungszentrum GKSS (03.01.2008),

http://www.gkss.de/pages.php?page=f_neutronenforschung.html&language=d&version=g

[2] DENEX- GmbH (03.01.2008),

<http://www.denex-gmbh.de/Arbeitsprinzip%20eines%20DENEX-Detektors.pdf>

CD:/Anhang/Literatur

[3] FAST ComTec (07.01.2008),

<http://www.fastcomtec.com/fwww/datasheet/mcd/mca-3.pdf>

CD:/Anhang/Datenblätter

[4] DENEX- GmbH (10.02.2008),

<http://www.denex-gmbh.de/Software.html>

[5] Microchip (07.01.2008),

CD:/Anhang/Datenblätter

[6] Semiconductor (07.01.2008),

http://komponenten.es.aau.dk/fileadmin/komponenten/Data_Sheet/ec1/MC10125.pdf

CD:/Anhang/Datenblätter

[7] XILINX (20.02.2008),

<http://www.xilinx.com/products/devkits/HW-SPAR3E-SK-US-G.htm>

[8] XILINX (20.02.2008),

http://www.xilinx.com/support/documentation/data_sheets/ds312.pdf

CD:/Anhang/Datenblätter

[9] XILINX (20.02.2008),

http://www.xilinx.com/ise/logic_design_prod/webpack.htm

[10] J. Reichardt/B. Schwarz: VHDL-Synthese, Entwurf digitaler Schaltungen und Systeme; 2.Auflage, Oldenburg, 2001

[11] XILINX (20.02.2008),

<http://www.xilinx.com/ise/mxe3/download.htm>

[12] TFH-Berlin (03.02.2008),

<http://www.tfh-berlin.de/~haas/lehre/eda1/WS02/stud8.pdf>

CD:/Anhang/Literatur

9 Bildverzeichnis

Bild 1:	Die Detektionsfläche wird durch X/Y-Kathode + Anode gebildet	4
Bild 2:	Prinzipieller Aufbau des Delayline- Readouts	6
Bild 3:	Signalfluss + Auswertung bei einfachem Neutroneneinschlag (TDC1- Methode)	7
Bild 4:	Signalfluss + Auswertung bei einfachem Neutroneneinschlag (TDC2- Methode)	8
Bild 5:	3D- Darstellung einer strahlendurchsetzten Spitzzange [4]	9
Bild 6:	Signalfluss + Auswertung bei Doppelereignis (TDC1- Methode, grün gültig).....	10
Bild 7:	Spiegelverkehrte Signalkoinzidenz, durch Addition der Signalamplituden	14
Bild 8:	Signalkoinzidenz durch synchrone Schieberegister + Registerzellverundung	15
Bild 9:	3 PIC- Prozessoren bilden das Herzstück des Simulators	18
Bild 10:	FPGA- Entwicklungsboard <i>Starterkit Spartan 3E</i>	20
Bild 11:	Interface zwischen Simulator, FPGA und MCA	21
Bild 12:	Freigabe der gültigen Signalkoinzidenz durch Anodenfenster.....	23
Bild 13:	Oszillogramm des Handshakes für einen Datensatz, nach <i>Deadtime</i> - Signalfreigabe	25
Bild 14:	Hierarchischer Aufbau der CSRD- Methode (Blockbild)	27
Bild 15:	Aufbau eines idealen Encoders, jedoch nicht praxistauglich	29
Bild 16:	Gesamtdarstellung der blockweise organisierten Codiereinheit.....	31
Bild 17:	Darstellung einer einzelnen Blockkomponenten	32
Bild 18:	Simulationsergebnis des Encoders. Mehrfachübereinstimmungen werden verworfen (rot)	35
Bild 19:	Hierarchischer Aufbau des TDC1- Verfahrens (Blockbild).....	38

Bild 20:	Blockbildartiger Messaufbau mit Fehlerüberwachung.....	41
Bild 21:	Tatsächlicher Messaufbau mit Fehlerüberwachung	42
Bild 22:	Vergleichende Darstellung der 3Methoden bei unterschiedlichen Ereignisraten	44
Bild 23:	Einzelmessung der CSR- Methode (ohne Anodenbewertung) bei unterschiedlichen Ereignisraten.....	46
Bild 24:	Optimierung der Ausbeute bei gleich bleibender Ereignisrate und veränderlichem Impulsbreitenverhältnis (CSR)	48
Bild 25:	Takteingrenzung zum Erhalt der Maximalausbeute	49

10 Anhang

A Tabellen (vergleichende Untersuchungen)

A.1 Messergebnisse TDC1- Verfahren

Msg. Nr.	Rate(/s)	theor. Rate(/s)	MCA erfasst (absolut)	Treffer (absolut)	Fehler (absolut)	Fehler(%)	Ausbeute(%)
1 *	0,4	10,00 k	10000	10000	0	0,00	100
2	2,94	73,50 k	9685	9505	179	1,85	96,85
3	5,80	145,00 k	9366	9086	278	2,97	93,66
4	11,33	283,25 k	8795	8229	563	6,40	87,95
5	21,51	537,75 k	7900	6952	948	12,00	79,00
6	41,49	1,04 M	6130	4346	1783	29,09	61,30
7	74,07	1,85 M	4483	1824	2659	59,31	44,83
8	121,95	3,05 M	3291	517	2774	84,29	32,91
9	181,82	4,55 M	2791	113	2678	95,95	27,91
10	243,90	6,10 M	2693	18	2674	99,29	26,93
11	285,71	7,14 M	2309	0	2309	100	23,09

Tabelle A.1: TDC1-Verfahren, 10.000 Ereignisse, Impulsbreitenverhältnis
($X1/X0 = Y1/Y0 = 2$)

* Angenommene Werte, da Messung zu viel Zeit in Anspruch nehmen würde

A.2 Messergebnisse TDC2- Verfahren

Msg. Nr.	Rate(/s)	theor. Rate(/s)	MCA erfasst (absolut)	Treffer (absolut)	Fehler (absolut)	Fehler(%)	Ausbeute(%)
1 *	0,4	10,00 k	10000	10000	0	0,00	100
2	2,94	73,50 k	9886	9886	0	0,00	98,86
3	5,80	145,00 k	9772	9769	3	0,03	97,72
4	11,33	283,25 k	9558	9526	31	0,32	95,58
5	21,51	537,75 k	9166	9122	44	0,48	91,66
6	41,49	1,04 M	8368	8127	241	2,88	83,68
7	74,07	1,85 M	7540	6682	858	11,38	75,40
8	121,95	3,05 M	6436	4987	1448	22,50	64,36
9	181,82	4,55 M	5537	3072	2465	44,52	55,37
10	243,90	6,10 M	4948	1850	3098	62,61	49,48
11	285,71	7,14 M	4684	1366	3318	70,84	46,84

Tabelle A.2: TDC2-Verfahren, 10.000 Ereignisse, Impulsbreitenverhältnis
($X1/X0 = Y1/Y0 = 2$)

* Angenommene Werte, da Messung zu viel Zeit in Anspruch nehmen würde

A.3 Messergebnisse CSRD- Verfahren

Msg. Nr.	Rate(/s)	theor. Rate(/s)	MCA erfasst (absolut)	Treffer (absolut)	Fehler (absolut)	Fehler(%)	Ausbeute(%)
1 *	0,4	10,00 k	10000	10000	0	0,00	100
2	2,94	73,5 k	9607	9606	1	0,01	96,07
3	5,80	145,00 k	9618	9618	0	0,00	96,18
4	11,33	283,25 k	9605	9605	0	0,00	96,05
5	21,51	537,75 k	9617	9617	0	0,00	96,17
6	41,49	1,04 M	9537	9533	4	0,04	95,37
7	74,07	1,85 M	9560	9551	9	0,09	95,60
8	121,95	3,05 M	9499	9478	21	0,22	94,99
9	181,82	4,55 M	9510	9466	44	0,46	95,10
10	243,90	6,10 M	9367	9282	85	0,91	93,67
11	285,71	7,14 M	9361	9240	120	1,28	93,61

Tabelle A.3: CSRD- Verfahren, 10.000 Ereignisse, Impulsbreitenverhältnis

$$(X1/X0 = Y1/Y0 = 2)$$

* Angenommene Werte, da Messung zu viel Zeit in Anspruch nehmen würde

B Tabellen (Arbeitspunkt Betrachtung)

B.1 Messergebnisse CSRD (ohne Anodenbewertung)

Msg. Nr.	Rate(/s)	theor. Rate(/s)	MCA erfasst (absolut)	Treffer (absolut)	Fehler (absolut)	Fehler(%)	Ausbeute(%)
1 *	0,4	10,00 k	10000	10000	0	0,00	100
2	2,94	73,5 k	10.223	9.866	357	3,57	102,23
3	5,80	145,00 k	10.381	9.888	493	4,93	103,81
4	11,33	283,25 k	10.805	9.807	998	9,98	108,05
5	21,51	537,75 k	11.518	9.716	1.802	18,02	115,18
6	41,49	1,04 M	13.797	9.375	4.422	44,22	137,97
7	74,07	1,85 M	16.552	9.417	7.135	71,35	165,52
8	121,95	3,05 M	21.855	9.234	12.621	126,21	218,55
9	181,82	4,55 M	28.034	9.260	18.773	187,73	280,34
10	243,90	6,10 M	34.351	9.405	24.946	249,46	343,51
11	285,71	7,14 M	39.825	9.635	30.190	301,90	398,25

Tabelle B.1: CSRD- Verfahren (ohne Anodenbewertung), 10.000 Ereignisse, Impulsbreitenverhältnis = 2

* Angenommene Werte, da Messung zu viel Zeit in Anspruch nehmen würde

B.2 Messergebnisse CSRD (Impulsbreitenoptimierung)

Nr.	Verhältnis X1/X0 = Y1/Y0	MCA erfasst (absolut)	Treffer (absolut)	Fehler (absolut)	Fehler (%)	Ausbeute (%)
1	0,77	1.462	1.452	10	0,68	14,62
2	1,00	2.166	2.158	8	0,37	21,66
3	1,27	3.462	3.455	7	0,20	34,62
4	1,50	5.977	5.973	4	0,07	59,77
5	1,77	8.959	8.945	14	0,16	89,59
6	1,90	9.349	9.343	6	0,06	93,49
7	2,00	9.521	9.515	6	0,06	95,21
8	2,03	9.392	9.380	12	0,13	93,92
9	2,10	8.627	8.620	7	0,08	86,27
10	2,27	6.389	6.378	10	0,16	63,89
11	2,50	4.760	4.752	8	0,17	47,60
12	2,77	4.896	4.884	12	0,25	48,96
13	3,00	5.015	4.997	18	0,36	50,15

Tabelle B.2: CSRD- Verfahren (Impulsbreitenoptimierung), 10.000 Ereignisse,

Rate: $1,85 \cdot 10^6$ Cycle/ s

B.3 Messergebnisse CSRD (Frequenztoleranz)

Msg. Nr.	Frequenz-Toleranz (%)	MCA erfasst (absolut)	Treffer (absolut)	Fehler (absolut)	Fehler (%)	Ausbeute (%)
1	-1,68	32	26	6	18,75	0,32
2	-1,52	35	30	5	14,29	0,35
3	-1,36	45	40	5	11,11	0,45
4	-1,20	54	48	6	11,11	0,54
5	-1,04	48	47	1	2,08	0,48
6	-0,88	272	268	4	1,47	2,72
7	-0,72	1361	1351	10	0,73	13,61
8	-0,56	5034	5029	5	0,10	50,34
9	-0,40	5931	5924	7	0,12	59,31
10	-0,24	5706	5700	6	0,11	57,06
11	-0,16	9119	9112	7	0,08	91,19
12	-0,08	9572	9568	4	0,04	95,72
13	0,00	9522	9515	7	0,07	95,22
14	0,08	9493	9491	2	0,02	94,93
15	0,16	8684	8681	3	0,03	86,84
16	0,24	5401	5394	7	0,13	54,01
17	0,40	5020	5010	10	0,20	50,20
18	0,56	4691	4683	7	0,15	46,91
19	0,72	656	650	6	0,91	6,56
20	0,88	142	141	1	0,70	1,42
21	1,04	28	23	5	17,85	0,28

Tabelle B.3: CSRD- Verfahren (Frequenztoleranz), 10.000 Ereignisse,
Rate: $1,85 \cdot 10^6$ Cycle/ s, Impulsbreitenverhältnis = 2

C Inhalt der beigelegten CD

- Diplomarbeit (PDF-Format)
CD:/
 - VHDL-Quellcode
CD:/Anhang/VHDL_Quellcode/
 - Schaltpläne Interface
CD:/Anhang/Interface/
 - Schaltpläne/Quellcode Simulator
CD:/Anhang/Simulator/
 - Quellcode Ueberwachungseinheit
CD:/Anhang/ Ueberwachungseinheit /
 - Literatur
CD:/Anhang/Literatur/
 - Datenblätter
CD:/Anhang /Datenblätter /
-
- Impressionen/Beispielmessungen
CD:/Anhang /Impressionen /

Versicherung über die Selbständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit im Sinne der Prüfungsordnung nach §25(4) ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtliche oder dem Sinn nach aus anderen Werken entnommene Stellen habe ich unter Angabe der Quellen kenntlich gemacht.

Hamburg, den 21. April 2008

Malte Borchers