BOUNCE: On-Chip Signalleitungen als Basis digitaler Zeitmessung

Dissertation

zur

Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Informatik und Elektrotechnik

der Universität Rostock

Vorgelegt durch Dipl.-Ing. Ralf Joost, geboren am 21.06.1978 in Gardelegen, aus Kühlungsborn

URN: urn:nbn:de:gbv:28-diss2010-0074-7

Erstgutachter: Prof. Dr.-Ing. habil. Ralf Salomon,
Institut für Angewandte Mikroelektronik und Datentechnik
Fakultät für Informatik und Elektrotechnik
Universität Rostock

Zweitgutachter: Prof. Dr.-Ing. habil. Helmut Beikirch
Institut für Gerätesysteme und Schaltungstechnik
Fakultät für Informatik und Elektrotechnik
Universität Rostock

Drittgutachter: Prof. Dr. J. Leo van Hemmen
Lehrstuhl für theoretische Biophysik
Physik Department
Technische Universität München

Datum der Verteidigung: 12.04.2010

Danksagung

Besonderer Dank gebührt Herrn Prof. Dr.-Ing. Ralf Salomon. Seine fachliche und persönliche Unterstützung sowie die fortwährende Beharrlichkeit in puncto „praktische Untersuchungen“ haben die erfolgreiche Anfertigung meiner Dissertation überhaupt erst ermöglicht. Danke Ralf.

Herrn Diplom-Informatiker Stefan Goldmann möchte ich für das Zuhören und das Diskutieren danken. In vielen Fällen hat er mir bei größeren und kleineren Problemen geholfen und mich stets motiviert. Danke Stefan.

Weiterhin gilt mein Dank dem gesamten Team des Instituts für Angewandte Mikroelektronik und Datentechnik für die andauernde Unterstützung, die nette Atmosphäre und das angenehme Arbeitsklima.

Schließlich bedanke ich mich bei Julia und meinen Eltern, die stets an mich geglaubt haben und mir immer den Rücken gestärkt haben.

Inhaltsverzeichnis

1. Einleitung und Motivation 1

2. Lokalisierung 5

2.1. Trilateration 5

2.2. Triangulation 8

2.3. Kenngrößen von Lokalisierungsverfahren 10

2.4. Lokalisierungskonzepte 11

2.4.1. Laufzeitmessung von Schallwellen 11

2.4.2. Laufzeitmessung elektromagnetischer Wellen 12

2.4.3. Auswertung der Empfangsleistung von EM-Wellen 13

2.4.4. Phasen- und Frequenzsynchronisation 15

2.4.5. Optische Verfahren – Photogrammetrie 15

2.4.6. Vergleich der vorgestellten Lokalisierungskonzepte 16

3. Laufzeitmessung, Messen von Zeitintervallen 18

3.1. Basisgrößenbestimmung für Triangulation und Trilateration 18

3.2. Terminologie 20

3.3. Zähler als einfachste Methode der Zeitmessung 21

3.4. Aktuelle Verfahren zur präzisen Laufzeitmessung 25

3.4.1. Time Stretching 25

3.4.2. Time Stretching und Analog-Digital-Converter 26

3.4.3. Die Vernier-Methode 27

3.4.4. Das Konzept der Tapped Delay Lines 29

4. Field-Programmable Gate Arrays 31

4.1. Grundstruktur der Field-Programmable Gate Arrays 31

4.2. Aufbau und Funktionalität der Logikelemente 32

4.3. Logik Array Blocks und die Verschaltung der Logikelemente 36

4.4. Zeitverhalten des Logikelements 37

5. Stand der Forschung: Tapped Delay Lines 39

5.1. Grundstruktur der Tapped Delay Lines 39
5.2. Differentielle Tapped Delay Lines 41
5.3. Aktuelle Tapped Delay Line Implementationen 42
5.3.1. ASCP-basierte Tapped Delay Lines 43
5.3.2. FPGA-basierte Tapped Delay Lines 44
5.4. Schlussfolgerungen aus dem Stand der Technik 46
6. BOUNCE: Konzept 47
6.1. Zielstellung dieser Arbeit, Motivation 47
6.2. Entwicklung des technischen Konzepts 50
7. BOUNCE: Praktische Umsetzung 54
7.1. Signalleitungen 54
7.2. Eingangs-Pads 56
7.3. Orientierung der Grundelemente 57
7.4. Das idealisierte Grundelement 58
7.5. Übertragung des Konzepts in das FPGA 61
8. FPGA-Details 63
8.1. Synthese der Schaltung 63
8.1.1. Automatische Optimierung 63
8.1.2. Integration des Shift-Registers 64
8.2. Platzierung der BOUNCE-Architektur im FPGA 65
8.2.1. Einfluss der FPGA-Struktur 66
8.2.2. Automatisierte Positionierung der Grundelemente 68
9. BOUNCE: Vollständiges Messsystem 71
9.1. Steuerung der BOUNCE-Architektur mit dem NiosII 72
9.2. Generierung von Zeitintervallen 73
9.3. PC-basiertes Auswerteverfahren 75
10. Ergebnisse 77
10.1. Das reale Grundelement 77

10.2. BOUNCE: Verbund der Grundelemente 81
10.2.1. Bestimmung der zeitlichen Auflösung der BOUNCE-Architektur 84
10.2.2. Bestimmung der zeitlichen Genauigkeit der BOUNCE-Architektur 85
11. Diskussion 89
11.1. Vergleich mit den Tapped Delay Lines 89
11.2. Einfluss der Messschaltung auf die Genauigkeit 91
11.3. Die Zeitbasis im Messsystem 93
11.4. Verteilte Grundelemente und lineare Interpolation 94
11.5. Verbesserung von Auflösung und Genauigkeit 98
12. Zusammenfassung und Ausblick 102
Thesen 112
Eidesstattliche Erklärung 114
Lebenslauf 115
Liste der eigenen Veröffentlichungen 117
Kurzreferat 119
Abstract 120

1. Einleitung und Motivation

Das Messen von Zeitintervallen ist eine immer wiederkehrende Tätigkeit des alltäglichen Lebens. Das Konzept „Zeit“ hilft einem, den Tagesablauf zu organisieren, die Backzeit des Sonntagskuchens zu kontrollieren und Termine mit anderen zu koordinieren. Neben diesen direkten Einsatzmöglichkeiten gibt es auch viele Anwendungen, in denen die Zeitmessung eine indirekte, aber nicht minder bedeutende Rolle spielt. Beispielsweise ermittelt die Geschwindigkeitsanzeige eines Autos, wie weit sich das Rad während eines definierten Zeitintervalls gedreht hat. Für die Höhe der Zinsen auf dem Bankkonto ist entscheidend, wie lange das Geld angelegt war. In der Musik wird durch die Tonhöhe angegeben, wie oft der Schall in einem bestimmten Zeitintervall das Trommelfell im Ohr in Schwingung versetzt.

Die meisten Menschen verwenden verschiedene Formen von Uhren für das Messen der Zeit. Die zeitliche Genauigkeit der Messung entspricht der kleinsten auf der Uhr angezeigten Zeiteinheit, bei vielen Uhren zum Beispiel eine Sekunde. Aufgrund unserer Umgebung und der darin ablaufenden Prozesse reicht diese Genauigkeit für die alltäglichen Aufgaben völlig aus. Nur in kleinen Randgebieten, zum Beispiel im Sport, wird auf Zentel-, Hundertstel- oder Tausendstelsekunden genau gemessen.

Die Messung eines Zeitintervalls folgt dabei immer demselben Prinzip. Zwei Ereignisse definieren den Anfang und das Ende des Zeitintervalls und ein Messsystem ermittelt, wie viel Zeit zwischen diesen beiden Ereignissen verstrichen ist. Die Zeitmessung selbst kann auf unterschiedlichste Weisen erfolgen. Bei der Sanduhr basiert sie auf der Menge Sand, die durch eine kleine Öffnung läuft, bei der Kerzenuhr wird die verstrichene Zeit durch die Höhe einer abbrennenden Kerze bestimmt. Die meisten Zeitmesser zählen allerdings eine Vielzahl kleiner Zeitimpulse und errechnen die Länge des Intervalls aus der Anzahl der kleinen Zeitimpulse und der bekannten Dauer zwischen diesen Impulsen.

Ein großes Gebiet, in dem genaue Zeitmessungen eine entscheidende Rolle spielen, ist die Lokalisierung. Daher beschreibt Kapitel 2 zunächst die mathematischen Grundlagen der Lokalisierung am Beispiel der Trilateration und Triangulation. In den vergangenen Jahren hat insbesondere das Gebiet der Indoor-Lokalisierung an Bedeutung gewonnen, da es für viele aktuelle Forschungsschwerpunkte eine grundlegende Rolle einnimmt. Indoor-Lokalisierung beschreibt die Verwendung von Lokalisierungsverfahern, die nicht auf die Nutzung von Satelliten-Signalen angewiesen sind und somit auch innerhalb geschlossener Räume (engl. indoor = „innen, im Hause“) funktionieren. Hier sind insbesondere die Arbeiten auf den Gebieten „Ambient Intelligence“, „Ubiquitous Computing“ und „Aging Science“ zu nennen. Ziel in allen drei Bereichen ist die automatische Unterstützung eines Nutzers. Klassische Bespiele sind die interaktive Tourführung für Museumsbesucher oder das automatische

Konfigurieren des Besprechungsraums in Abhängigkeit der Verteilung und Anzahl der Gäste. Grundlage für diese Unterstützungssysteme ist immer eine genaue Kenntnis über Position und Orientierung der Anwesenden, möglichst im Millimeter- oder einstelligen Zentimeterbereich. Für das Gebiet der Indoor-Lokalisierung wurden daher verschiedene Techniken und Verfahren erforscht. Sie basieren auf unterschiedlichen Mechanismen und verwenden unterschiedliche Signale und physikalische Effekte. Aus diesem Grund gibt Kapitel 2 ebenfalls einen Überblick über aktuelle Konzepte, die in der Indoor-Lokalisierung Anwendung finden. Ein Vergleich dieser Konzepte zeigt: Systeme, die eine hohe Genauigkeit liefern, sind zu aufwendig und damit zu teuer, um einen breiten Markt zu erreichen. Systeme, die kostengünstig angeboten werden können, erreichen nicht die erforderlichen Genauigkeiten um eine exakte Positionsbestimmung zu ermöglichen.

Aus der Gruppe der vorgestellten Lokalisierungskonzepte ist insbesondere die Laufzeitmessung elektromagnetischer Signale aufgrund deren physikalischer Eigenschaften von technischem Interesse. Die Laufzeitmessung ermöglicht die Bestimmung von Weglängen oder Winkelgrößen zwischen dem zu lokalisierenden Objekt und bekannten Referenzpunkten. Die Länge einer Strecke lässt sich dadurch bestimmen, dass die Dauer (Laufzeit) gemessen wird, die ein Signal mit einer bekannten Ausbreitungsgeschwindigkeit zur Überwindung der Strecke benötigt. Aus dem allgemein bekannten Zusammenhang, dass der Weg dem Produkt aus Geschwindigkeit und benötigter Zeit entspricht, lässt sich dann die Weglänge zurückgewinnen. Mit den so ermittelten Größen lässt sich die Lokalisierung unter Anwendung der Trilateration bzw. Triangulation durchführen. Die Herausforderung besteht darin, trotz der hohen Ausbreitungsgeschwindigkeit elektromagnetischer Signale von 300.000 km/s präzise die von den Signalen zurückgelegten Strecken zu bestimmen. Dies erfordert hochgenaue Zeitmesser, entspricht doch eine zeitliche Auflösung des Messsystems von 1ns schon einer zurückgelegten Wegstrecke von 3m. Die Genauigkeit der errechneten Streckenlänge hängt somit direkt von der Genauigkeit des eingesetzten Zeitmesssystems ab. Dies stellt auch eins der großen Probleme im Forschungsfeld der Lokalisierung dar.

Kapitel 3 widmet sich daher verschiedenen technischen Umsetzungen von Laufzeitmess-systemen und stellt die erreichten Genauigkeiten in Bezug zum technischen Realisierungsaufwand dar. Wurden solche Systeme bisher explizit für diesen Zweck angefertigt, bietet die moderne Schaltungstechnik in Form von freiprogrammierbaren Schaltkreisen (field programmable gate arrays - FPGAs) günstige Plattformen für einen Teil dieser Systeme. Nach einer detaillierten Beschreibung der Struktur und Eigenschaften dieser Schaltkreise in Kapitel 4, stellt Kapitel 5 ausführlich das Konzept der Tapped Delay Line (TDL), einem FPGA-basierten Laufzeitmesssystem, vor. Diese stellen den fachlichen Ausgangspunkt dieser Arbeit dar, entsprechend werden aktuelle Varianten und

Implementationen der Tapped Delay Lines vorgestellt, verglichen und bewertet. Trotz der dort bisher erreichten zeitlichen Auflösungen von 50ps liegt der begrenzende Faktor für weitere Verbesserungen in den Struktureigenschaften der FPGAs. Ähnlich wie bei vielen der in Kapitel 3 vorgestellten Systemen ist auch bei den TDLs die Auflösung abhängig von den Schaltzeiten der integrierten Gatter, welche für die Zeitdiskriminierung, also der Zerlegung der Zeit in detektierbare Teilstücke, verantwortlich sind.

In der Biologie wurde ausgiebig untersucht, wie das Prinzip der Ortung von Signalquellen im Gehirn verschiedener Tierarten funktioniert. Am Beispiel der Schleiereule wurde dabei festgestellt, dass die als „Schaltelemente“ eingesetzten Neuronen des Gehirns eine Schaltzeit von ca. 5ms besitzen. Dennoch kann die Schleiereule die Quelle von Schallsignalen mit einer Genauigkeit von 2° orten, wofür prinzipiell Schaltzeiten von 1 ms erforderlich sind. Wie in Kapitel 6 dargestellt, wird dies dadurch ermöglicht, dass die Neuronen nicht als sequentielle Zeitdiskriminatoren eingesetzt werden, sondern parallel an einen Nervenstrang angeschlossen sind. Ebenfalls in Kapitel 6 wird die Übertragung der Erkenntnisse aus der Biologie in eine Hardware-Architektur zur Laufzeitmessung ausführlich beschrieben.

Als Kern dieser Arbeit wird ein technisches System entwickelt, welches auf dem gleichen Prinzip basiert und in der Lage ist, die gegenwärtig erreichten Auflösungen der Tapped Delay Lines unter Beibehaltung der günstigen FPGA Plattform um eine Größenordnung zu verbessern. Dabei handelt es sich um ein neuartiges Konzept zur hochpräzisen Messung von Zeitintervallen im Pikosekundenbereich. Dieses auf den Namen BOUNCE (Bunch Of UNconnected Chain Elements - BOUNCE) getaufte Verfahren ist inspiriert durch das biologische Ortungssystem der Schleiereule. Während die Schleiereule Signalverzögerungen auf den neuronalen Verbindungen im Gehirn zur Ortung einer Schallquelle benutzt, basiert BOUNCE auf der Ausnutzung von systeminherenten Laufzeiteigenschaften interner Signalleitungen integrierter Schaltkreise.

Da das BOUNCE-Verfahren für sich beansprucht, auf kostengünstigen Schaltkreis-plattformen wie FPGAs hochpräzise Laufzeitmessungen durchführen zu können, wird im Rahmen dieser Arbeit ein erster funktionsfähiger Prototyp der Schaltung mit dem dazugehörenden Messaufbau entwickelt. Die Beschreibung des Prototyps erstreckt sich über die Kapitel 7, 8 und 9. Kapitel 7 widmet sich der prinzipiellen Übertragung des technischen Konzepts in ein beliebiges FPGA. Dort ist beschrieben, in welcher Art und Weise die einzelnen Bestandteile der BOUNCE-Architektur auf die FPGA-internen Ressourcen abgebildet werden. Kapitel 8 geht detailliert auf FPGA-spezifische Besonderheiten ein, die für die im Rahmen dieser Arbeit verwendeten FPGAs des Herstellers Altera von Bedeutung sind. Es beschreibt die Vorkehrungen, die notwendig sind, um das auf den ersten Blick

ungewöhnliche Funktionsprinzip der BOUNCE-Architektur in einem Altera-FPGA zu realisieren. Abschließend beschreibt Kapitel 9 die Komponenten, die zusammen mit dem eigentlichen FPGA und der darin enthaltenen BOUNCE-Architektur letztendlich das vollständige Messsystem bilden.

Die mit diesem Messsystem durchgeführten Versuche und vor allem deren Ergebnisse werden in Kapitel 10 dargestellt. Dort wird ausführlich auf das Zeitverhalten der verwendeten Schaltungsbestandteile eingegangen sowie die für ein Zeitmesssystem wichtigsten Größen „Auflösung“ und „Genauigkeit“ bestimmt. Ebenfalls untersucht Kapitel 10 die Richtigkeit der bei der Entwicklung des Schaltungskonzepts in Kapitel 6 postulierten Eigenschaften. Eine kritische Diskussion der erreichten Ergebnisse schließt sich in Kapitel 11 an.

Kapitel 12 bietet eine Zusammenfassung dieser Arbeit und einen Ausblick auf die weiteren Forschungsarbeiten auf diesem Gebiet. Dort wird auch der mögliche Einsatz dieser Architektur in praktischen Lokalisierungsanwendungen und die daraus resultierenden Implikationen für das Forschungsgebiet der Lokalisierung diskutiert.

2. Lokalisierung

Als *Lokalisierung* oder Lokalisation werden die Vorgänge bezeichnet, welche die Ortsbestimmung (*Ortung*) eines Objektes relativ zu bekannten Positionen im Raum ermöglichen. Mit anderen Worten: die Lokalisierung ermittelt den Ort eines Objekts. Der Begriff „Ort“ bezeichnet dabei eine Position, die sich in einem durch bekannte Punkte gebildeten Koordinatensystem befindet.

Wesentlicher Bestandteil der Lokalisierung sind die Verfahren der *Triangulation* und der *Trilateration*. Grundlage beider Verfahren sind die Berechnungsvorschriften in Dreiecken. Während das erstgenannte Verfahren die Ausrichtung, also den Winkel des Objektes zu drei definierten Positionen verwendet, basiert die Trilateration auf der Verwendung der Entfernungen des Objektes zu drei bekannten Punkten im Raum. Die Koordinaten der drei Referenzpunkte und die gemessenen Entfernungen/Winkel des Objektes zu diesen ergeben ein Gleichungssystem, in denen die Raumkoordinaten (*x,y,z*) des Objektes als unbekannte Größen erscheinen. Die Lösung des Gleichungssystems liefert die Position des Objektes.

Dieses Kapitel widmet sich zunächst einer Zusammenstellung der mathematischen Grundlagen der Trilateration und der Triangulation. Es schließt sich eine Darstellung charakteristischer Merkmale von Lokalisierungsverfahren an. In Abschnitt 2.4 werden aktuelle technische Konzepte zur Lokalisierung anhand ihrer charakteristischen Merkmale miteinander verglichen. Dies erlaubt eine erste Abschätzung der Leistungsfähigkeit der technischen Konzepte in Bezug auf die vorgestellten charakteristischen Merkmale.

2.1. Trilateration

Die Trilateration bezeichnet die Ortsbestimmung eines Objektes anhand von Streckenlängen zu bekannten Punkten. Diese bekannten Punkte werden als Referenzpunkte bezeichnet. Bei der Trilateration ist die Länge der Strecke zwischen dem Objekt und jedem Referenzpunkt durch Messung bekannt. Ein gängiges Verfahren für die Bestimmung der Streckenlänge ist zum Beispiel die Laufzeitmessung. Bei bekannter Entfernung zum Referenzpunkt ist für das Lokalisierungsverfahren wichtig, dass sich das Objekt nur auf einer Kugeloberfläche mit dem Referenzpunkt im Mittelpunkt befinden kann. Der Radius der Kugeloberfläche entspricht gerade der gemessenen Streckenlänge zum Referenzpunkt.

Die Trilateration verwendet mehrere, mindestens aber drei, Referenzpunkte. Durch die Überlagerung von mindestens drei Kugeloberflächen ergibt sich ein eindeutiger Schnittpunkt, der die Position des Objektes markiert. Abbildung 1 stellt die Ausgangssituation der Trilateration grafisch dar. Das Objekt befindet sich am noch unbekannten Ort $P(x,y,z)$. R_1, R_2 und R_3 stellen die drei Referenzpunkt dar.

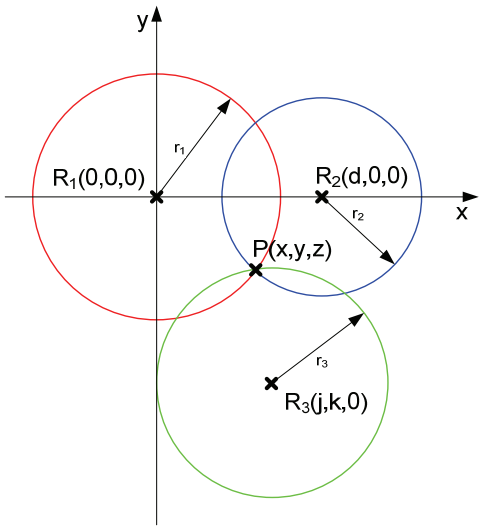


Abbildung 1: Schematische Darstellung der Ausgangssituation für die Trilateration; zur besseren Darstellung auf zwei Dimensionen vereinfacht.

Um eine einfache Berechnung der Koordinaten *x, y* und *z* des Objekts *P* zu ermöglichen werden drei Randbedingungen vorgegeben, die die Allgemeinheit dieses Ansatzes allerdings nicht einschränken, da sie durch eine Koordinatentransformation aus jeder beliebigen Konfiguration der drei Referenzpunkte erreicht werden können:

1. Alle Referenzpunkte liegen in der Ebene mit *z=0*
2. Der Referenzpunkt R_1 liegt im Koordinatenursprung
3. Der Referenzpunkt R_2 liegt auf der x-Achse

Mit diesen Randbedingungen ergeben sich für die drei Kugelradien:

$$r_1^2 = x^2 + y^2 + z^2 \tag{2.1}$$

$$r_2^2 = (x-d)^2 + y^2 + z^2 \tag{2.2}$$

$$r_3^2 = (x-j^2) + (y-k)^2 + z^2 \tag{2.3}$$

Durch Subtraktion der Gleichung (2.2) von Gleichung (2.1) ergibt sich als Lösung für die x-Koordinate des Objektes:

$$x = \frac{r_1^2 - r_2^2 - d^2}{2d} \tag{2.4}$$

Wird Gleichung (2.4) in Gleichung (2.1) rückübertragen ergibt sich:

$$y^2 + z^2 = r_1^2 - \frac{(r_1^2 - r_2^2 + d^2)^2}{4d^2} \tag{2.5}$$

Durch Gleichsetzen mit der Gleichung für die dritte Kugel erhält man als Lösung für die y-Koordinate:

$$y = \frac{r_1^2 - r_3^2 - x^2 + (x-j)^2 + k^2}{2k} = \frac{r_1^2 - r_3^2 + j^2 + k^2}{2k} - \frac{j}{k}x \tag{2.6}$$

Mit den beiden berechneten Koordinaten *x* und *y* wird durch Verwendung von Gleichung (2.1) die z-Koordinate berechnet:

$$z = \sqrt{r_1^2 - x^2 - y^2} \tag{2.7}$$

Damit ist der Punkt *P(x,y,z)* vollständig bestimmt. Durch den Wurzelausdruck in Gleichung (2.7) kann das Problem keine, eine oder zwei Lösungen aufweisen.

Die Lösung *z=0* ergibt sich, wenn sich die drei Kugeln in genau einem Punkt schneiden. Das Objekt befindet sich dann in der Ebene *z=0*, wo nach der Randbedingung 1 (siehe oben) auch alle Referenzpunkte liegen. Der Fall, dass alle vier Punkte in einer Ebene liegen ist der einzige, der zu einem einzigen Ergebniswert führt.

Liegen nicht alle Punkte in einer Ebene, entstehen auch immer zwei Schnittpunkte aller Kugeln. Zu erkennen ist dies zum einen daran, dass in den Gleichungen der einzelnen Kugelradien (2.1), (2.2) und (2.3) immer der Term z^2 auftaucht. Zum anderen ist auch an Abbildung 1 leicht nachzuvollziehen, dass zwei unterschiedliche Punkte $P_1(x,y,z)$ und $P_2(x,y,-z)$ zu allen Referenzpunkten die gleichen Entfernungen aufweisen. Im praktischen Einsatz der Trilateration wird meistens durch einfache Gültigkeitsprüfungen der richtige der beiden ermittelten Punkte bestimmt. Beim Global Positioning System (GPS) liegt einer der beiden Punkte auf der Erdoberfläche. Der andere hingegen würde sich „hinter“ den Satelliten im All befinden. Somit kann dieser schnell als gültige Lösung ausgeschlossen werden.

Nimmt der Ausdruck unter der Wurzel in Gleichung (2.7) einen negativen Wert an, entsteht keine gültige Lösung für das Problem. Ein negativer Wurzelausdruck entsteht, wenn zum Beispiel eine oder mehrere Entfernungen zu den Referenzpunkten falsch gemessen wurden und somit die Werte für die x-Koordinate und/oder y-Koordinate des Punktes *P* falsch sind. Dadurch wird deutlich, dass eine exakte Bestimmung der Weglängen zu den Referenzpunkten von entscheidender Wichtigkeit ist.

Bei der Verwendung verrauschter, fehlerhafter Messwerte für die Entfernungen zu den drei Referenzpunkten muss für die entsprechende Strecke statt eines einzelnen Wertes ein Bereich angegeben werden. Die drei Kugeloberflächen weiten sich zu Kugelschalen mit endlicher Dicke aus. Statt eines eindeutigen Schnittpunktes ergibt sich ein Schnittvolumen endlicher Größe [29] [49]. Der gesuchte Punkt liegt dann irgendwo in diesem Volumen. Auch hier ist erkennbar, dass eine genaue Messung der Entfernungen zu den Referenzpunkten

wichtig ist. Auf die physikalischen Effekte, die für die Messung dieser Entfernungen genutzt werden können, geht Abschnitt 2.4 ein. Abschnitt 3.1 erläutert detailliert den Messvorgang am Beispiel der Laufzeitmessung. Vorher stellt Abschnitt 2.2 noch das zweite Lokalisierungsverfahren, die Triangulation, vor.

2.2. Triangulation

Die Triangulation (auch: Triangulierung) verwendet, im Gegensatz zu Trilateration, Winkelangaben für die Positionsbestimmung eines Objektes. Die zwei gebräuchlichsten Verfahren in der Triangulation sind das Vorwärtseinschneiden und das Rückwärtseinschneiden. Beim Vorwärtseinschneiden werden in einem Dreieck aus zwei Referenzpunkten und dem gesuchten Punkt die Winkel an zwei Referenzpunkten bestimmt. Diese Konstellation ist schematisch in Abbildung 2 dargestellt.

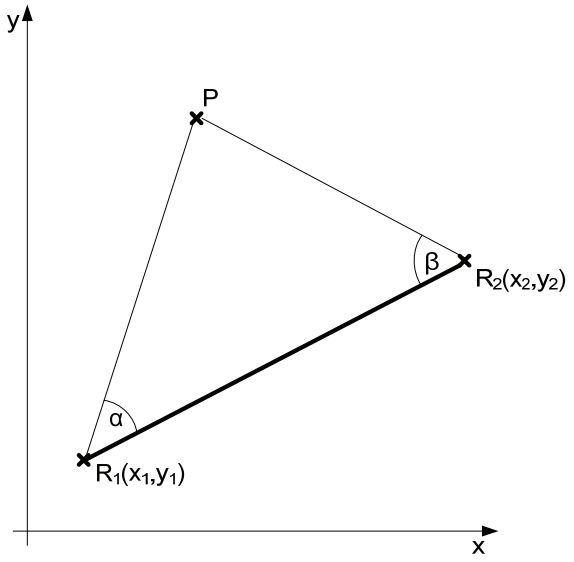


Abbildung 2: grafische Darstellung des Vorwärtseinschneidens (zweidimensional)

Durch die zwei Winkel *α* und *β* sowie der bekannten (oder zumindest bestimmbaren) Strecke zwischen beiden Referenzpunkten können alle übrigen Größen des Dreiecks berechnet werden. Allerdings erfordert diese Variante das Messen der Winkel an den Referenzpunkten. In praktischen Lokalisierungsanwendungen können die Berechnungen nur am Ort des Objektes, also im Punkt *P* durchgeführt werden. Durch alleiniges Messen des Winkels im Punkt *P* können die Koordinaten des Punktes nicht eindeutig bestimmt werden. Vielmehr ergeben sich unendlich viele Möglichkeiten zur Positionierung des Punktes *P* mit dem gemessenen Winkel R_1PR_2 auf dem Bogen eines Kreises, indem die Strecke R_1R_2 als Sekante auftritt. Abbildung 3 zeigt zwei dieser möglichen Punkte *P'* und *P''*.

Über den Peripheriewinkelsatz können die unbekannten Parameter des Kreises bestimmt werden.

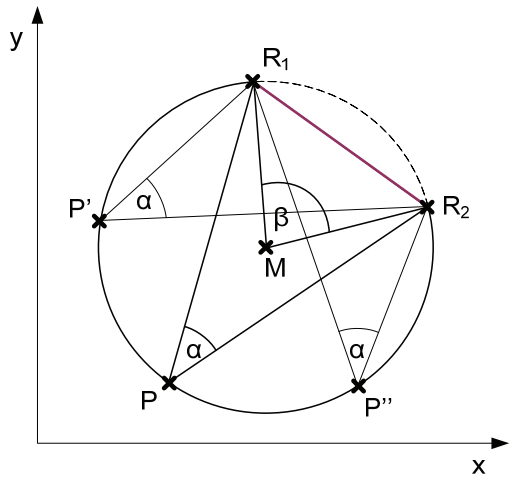


Abbildung 3: Darstellung der geometrischen Zusammenhänge zwischen Winkel und Strecke im Kreis und die Mehrdeutigkeit bei bekanntem Winkel und bekannter Streckenlänge.

Es gilt:

$$2\alpha = \beta \tag{2.8}$$

$$\sin\frac{\beta}{2} = \frac{\overline{R_1 R_2}}{2r} \tag{2.9}$$

Aus Gleichung (2.9) lässt sich der Radius *r* des Kreises bestimmen. Ferner kann nun der Mittelpunkt des Kreises berechnet werden. Um die Mehrdeutigkeit des Punktes *P* zu umgehen, wird ein weiterer Referenzpunkt R_3 hinzugefügt, sodass zwei sich überlagernde Kreise entstehen. Dieses Verfahren wird als Rückwärtseinschneiden bezeichnet und in folgender Abbildung skizziert.

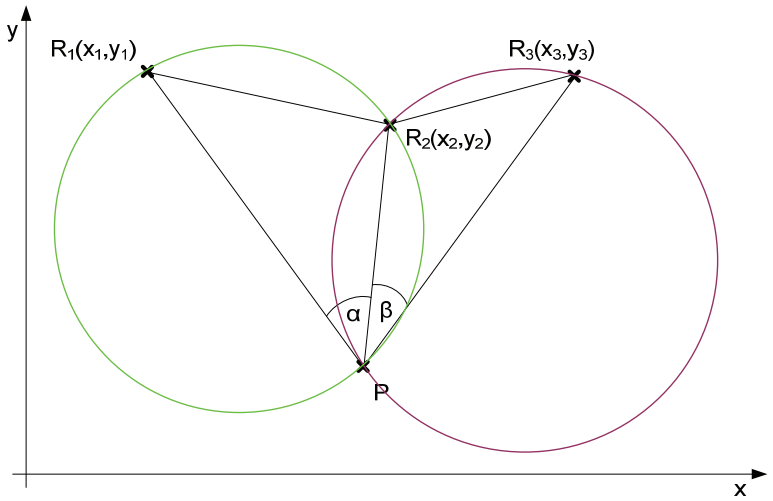


Abbildung 4: grafische Darstellung des Rückwärtseinschneiden mittels zweier Kreise zur Vermeidung der Mehrdeutigkeit des Punktes P

Sind die Parameter Radius und Mittelpunktkoordinaten beider Kreise bekannt, können die Schnittpunkte (*P* und R_2) bestimmt werden. Die Literatur [29] [10] bietet verschiedene

formale Vorschriften zur Lösung dieses Problems an, unter anderem das Rückwärtseinschneiden nach Snellius und das Rückwärtseinschneiden nach Cassini. Eine vollständige Darstellung dieser Verfahren ist für den Rahmen dieser Arbeit unangemessen, der geneigte Leser sei daher auf die angegebene Literatur verwiesen.

Die Triangulation ist ein Verfahren, welches insbesondere in der Geodäsie eine große Bedeutung für die Ermittlung der Koordinaten eines unbekannten Punktes *P* besitzt. Die Triangulation, insbesondere das Rückwärtseinschneiden, bietet für praktische Lokalisierungsaufgaben den Vorteil, dass alle zur Positionsbestimmung erforderlichen dynamischen Daten, wie die Winkel *α* und *β*, am Ort des zu lokalisierenden Objektes aufgenommen werden. Auf die Bestimmung der Winkelgrößen mit Hilfe der Laufzeitmessung geht Abschnitt 3.1 ein.

2.3. Kenngrößen von Lokalisierungsverfahren

Lokalisierung bezeichnet die (meist rechnergestützte) Ausführung der Triangulation bzw. der Trilateration zur Positionsbestimmung eines Objektes. Grundlegende Kenngrößen der verschiedenen Lokalisierungsverfahren sind *Auflösung* und *Genauigkeit*. Im Rahmen dieser Arbeit werden beide Größen wie folgt verwendet. Die *Auflösung* bezeichnet das (technisch/physikalische) Vermögen des Messsystems, zwischen zwei verschiedenen Positionen des Objektes einen reproduzierbaren Unterschied zu erkennen. Die Auflösung ist eng mit dem Begriff der Präzision verbunden, die angibt, wie exakt ein Messergebnis ist ohne Bezug auf den wahren Wert der Messgröße [65]. Der absolute Betrag der Messgröße ist dabei nicht von Bedeutung, da nur relative Änderungen des Messwertes betrachtet werden. In Lokalisierungsanwendungen wird die Auflösung im Allgemeinen in einem Längenmaß angegeben.

Die *Genauigkeit* hingegen stellt ein Maß für die Richtigkeit des Messergebnisses dar, sie bezieht sich also auf die Abweichung eines Messergebnisses vom wahren Wert [65]. Hier wird der gemessene, absolute Wert der Messgröße mit dem Wert eines quasi idealen Messgerätes verglichen. Auch sie wird in Lokalisierungsanwendungen in einem Längenmaß angegeben.

Weitere Kenngrößen unterschiedlicher Lokalisierungsverfahren sind *Messdauer, Reichweite* und *Messaufwand*. Die *Messdauer* gibt an, wie viel Zeit notwendig ist um die Lokalisierung durchzuführen. Sie hat entscheidenden Einfluss auf die Eignung des jeweiligen Lokalisierungsverfahrens für verschiedene Anwendungen. Für die kontinuierliche und genaue Ortung schnell bewegter Objekte ist eine kurze Messdauer von entscheidender Bedeutung. Anders bei Anwendungen, die höchste Genauigkeit bei wenigen Messungen, z.B. der Ortung quasi stationärer Objekte, erfordern. Hier kann zugunsten der Genauigkeit

eine lange Messdauer von Vorteil sein. Die *Reichweite* eines Lokalisierungsverfahrens gibt an, über welchen Entfernungsbereich zwischen Referenzpunkten und Objekt die angegebene Genauigkeit erzielt werden kann. Die Reichweite hängt im Wesentlichen vom im Messsystem ausgenutzten physikalischen Effekt zur Bestimmung der während der Lokalisierung verwendeten Größen ab. Der *Messaufwand* ist in erster Linie eine Kenngröße für ökonomische Vergleiche unterschiedlicher Lokalisierungssysteme. Im Messaufwand werden der Aufwand und die damit verbundenen Kosten für Realisierung und Betrieb des Messsystems zusammengefasst. Dazu zählen z.B. Fertigung, Eichung und Installation des Systems selbst sowie ggf. die Einstellung von Umgebungsparametern. Auch die für den laufenden Betrieb notwendigen Anstrengungen werden im Messaufwand berücksichtigt.

2.4. Lokalisierungskonzepte

Lokalisierungskonzepte basieren auf der Auswertung physikalischer beobachtbarer Effekte um im Rahmen einer anschließenden Nachverarbeitung eine Positionsbestimmung vornehmen zu können. In diesem Abschnitt werden die gebräuchlichsten Konzepte kurz dargestellt und verglichen.

2.4.1. Laufzeitmessung von Schallwellen

Eine technisch sehr einfache Methode zur Entfernungsbestimmung zwischen Objekten ist die Laufzeitmessung mit Hilfe von Schallwellen. Schallwellen sind vor allem durch eine langsame Ausbreitungsgeschwindigkeit von ca. 340 m/s in Luft gekennzeichnet. Bei den Schallwellen handelt es sich um Longitudinalwellen, bei denen die Ausbreitungsrichtung parallel zur Schwingungsrichtung orientiert ist. Ein mechanischer Schwinger (z.B. eine Lautsprechermembran) überträgt Energie in das Trägermedium, z.B. Luft. Die periodische Schwingung der Membran führt zu einer periodischen Druckänderung an der Membranfläche. Diese überträgt sich auf die Luftmoleküle in der Art, dass bei einer Druckerhöhung die Moleküle zusammengepresst und bei einer Druckverringerung entsprechend auseinandergezogen werden. Die Druckänderungen breiten sich in Form von Dichtewellen in Richtung ihre Erregung aus, die Luftteilchen übertragen die Energie, die ihnen durch die Membran zugeführt wurde.

Die Entfernung zwischen einem Sender und einem Empfänger errechnet sich aus dem Produkt der bekannten Ausbreitungsgeschwindigkeit der Schwingung und der Zeit, die vergeht, bis die die vom Sender ausgehende Schwingung den Empfänger erreicht. Dabei ist aber zu berücksichtigen, dass die von einem quasi punktförmigen Sender (idealisiert als Kugelstrahler nullter Ordnung) ausgehenden Druckwellen sich gleichmäßig in alle Raumrichtungen ausbreiten. Da die Wellenfronten in diesem Fall eine Kugeloberfläche beschreiben, nimmt die in einem beliebigen Punkt messbare Energie mit zunehmender

Entfernung zum Sender ab. Die messbare Energie ist dabei umgekehrt proportional zum Quadrat der Entfernung. Neben diesem allein auf der Entfernung basierenden Abfall der Schwingungsamplitude dämpft ein weiterer Effekt die Schwingung. Da es sich bei Schallwellen um einen materiegebundenen Effekt handelt, sorgt die Massenträgheit des Übertragungsmediums für eine weitere Dämpfung. Die Auslenkung der Luftteilchen benötigt Energie, welche wiederum in einer Verringerung der Schwingungsamplitude resultiert. Die Größe dieser Dämpfung ist in hohem Maße abhängig von den Parametern des Trägermediums. Hier ist vor allem Druck, Temperatur und Feuchtigkeit zu berücksichtigen. Aber auch die Eigenschaften der ausgesendeten Schallwelle, insbesondere ihre Frequenz, haben einen Einfluss auf die Dämpfung.

Alle diese Faktoren führen dazu, dass die Verwendung von Schallwellen zur Laufzeitmessung nur für kurze Distanzen geeignet ist. Um akzeptable Genauigkeiten zu garantieren, muss die Entfernung zwischen Signalquelle und Objekt auf Entfernungen kleiner als 50m begrenzt bleiben [66] [56] [54].

2.4.2. Laufzeitmessung elektromagnetischer Wellen

Die Laufzeitmessung elektromagnetischer Wellen bietet ebenfalls eine Möglichkeit zur Entfernungsbestimmung für die bekannten Verfahren der Lokalisierung. Elektromagnetische Wellen sind im Vergleich zu Schallwellen durch eine wesentlich höhere Ausbreitungsgeschwindigkeit von ca. 300.000 km/s gekennzeichnet. Eine elektromagnetische Welle entsteht durch Umladungsvorgänge am offenen Schwingkreis (auch Dipol genannt).

Bekannt ist, dass um einen stromdurchflossenen Leiter herum ein Magnetfeld entsteht. Ebenfalls ist bekannt, dass sich an einem Kondensator ein elektrisches Feld ausbildet. Der einfache Dipol besteht lediglich aus einem einfachen Stab. Die Stabenden stellen die rudimentären Reste eines Plattenkondensators dar, der Stab selbst fungiert als Spule. Speist man mittig in diesen Stab einen Wechselstrom ein, bewegen sich die Ladungsträger auf dem Stab je nach Richtung des Stroms in die eine, bzw. andere Richtung. Es bilden sich ein elektrisches und ein magnetisches Feld um den Leiter herum aus. Die Wirkung dieser Felder breitet sich mit Lichtgeschwindigkeit aus [18].

An der einfachen Spule und auch am Kondensator ist es nun so, dass beide Bauteile die ihnen zugeführte Energie im magnetischen Feld (Spule) und im elektrischen Feld (Kondensator) speichern. Bei Entfernen der Ursache für den Aufbau dieser Felder, wird die gespeicherte Energie wieder zurück in den Stromkreis überführt (Wärmeverluste und ähnliche Phänomene seien an dieser Stelle vernachlässigt).

Am Dipol ist es so, dass die Zeit für die Umpolung des Stabes durch den zugeführten Wechselstrom so gering ist, dass nicht die gesamte in beiden Feldern gespeicherte Energie

in den Stab überführt werden kann. Vielmehr entsteht bereits ein neues elektrisches und magnetisches Feld, was zu dem vorhergehenden Feld eine umgekehrte Polarität aufweist. Das vorhergehende Feld wird dadurch vom Stab abgeschnürt und durch das neue Feld mit Lichtgeschwindigkeit in den Raum „gedrückt“. Die elektromagnetische Welle breitet sich ausgehendend von der Sendeantenne in alle Raumrichtungen aus [9] [18]. Die Wellenlänge ist dabei abhängig von der Frequenz der Erregung in der Sendeantenne. Eine Informationsübertragung ist durch Modulation der Wellen möglich.

Bei elektromagnetischen Wellen handelt es sich um Transversalwellen, da Ausbreitungsrichtung, elektrische Feldstärke und magnetische Feldstärke jeweils orthogonal zueinander stehen. Die Ausbreitung der elektromagnetischen Welle ist dabei nicht an das Vorhandensein eines Trägermediums gebunden. Demzufolge wird sie auch nicht von den Eigenschaften eines Mediums beeinflusst, wie es bei den oben beschriebenen Schallwellen der Fall ist. Die elektromagnetischen Wellen weisen daher eine geringere Dämpfung auf und können somit auch für Übertragungen über große Entfernungen genutzt werden.

In der Empfängerantenne erzeugen die auftretenden elektromagnetischen Wellen wieder einen Wechselstrom. Dieser kann nach entsprechender Verstärkung in digitale Systeme überführt werden, wo auf das Trägersignal modulierte Informationen wieder extrahiert werden. Die Messung der Laufzeit elektromagnetischer Wellen ist somit an die Bestimmung des Auftretens der Wellen am Empfänger gebunden. Die hohe Ausbreitungsgeschwindigkeit stellt allerdings hohe Anforderungen an die Auflösung der Messsysteme. Dadurch ergibt sich ein hoher Messaufwand. Da sich die elektromagnetischen Wellen mit Lichtgeschwindigkeit ausbreiten, ist für eine räumliche Genauigkeit der Entfernungsbestimmung von einem Meter bereits eine zeitliche Genauigkeit des Messsystems von $1/3 \cdot 10^{-8}$ s (ca. 3,3 ns) erforderlich. Für Genauigkeiten von einem Zentimeter muss das Messsystem bereits eine zeitliche Auflösung von $1/3 \cdot 10^{-10}$ s oder 33 ps besitzen. In taktgebundenen digitalen Systemen entspräche dies einer notwendigen Taktrate von 30 GHz, was für Standardsysteme derzeit unmöglich ist.

Allerdings ist es auch möglich, die Entfernung zwischen Sender und Empfänger anhand eines anderen, entfernungsabhängigen Parameters der elektromagnetischen Wellen zu bestimmen. Dieses Verfahren basiert auf der Auswertung der Sende-/Empfangsleistung und wird im folgenden Abschnitt beschrieben.

2.4.3. Auswertung der Empfangsleistung von EM-Wellen

Der Betrag der elektrischen Feldstärke elektromagnetischer Wellen, die sich ausgehend von einer punktförmigen Quelle ausbreiten, folgt im Fernfeld:

$$E \sim \frac{1}{r} \tag{2.10}$$

Für den Betrag der magnetischen Feldstärke *H* gilt im Fernfeld ebenfalls:

$$H \sim \frac{1}{r} \tag{2.11}$$

Im Fernfeld befinden sich elektrisches und magnetisches Feld in Phase. Das Kreuzprodukt aus den Vektoren der elektrischen und der magnetischen Feldstärke ergeben den Poyntingvektor ***S*** [59] [9]. Sein Betrag ist ein Maß für die Flächendichte der Leistungsströmung (Leistungsdichte, Strahlungsdichte), gibt also an, welche Leistung auf einer definierten Fläche auftrifft. Dieser Betrag ist nach Gleichung (2.11) und Gleichung (2.10) vom Abstand von der Signalquelle abhängig und folgt:

$$S \sim \frac{1}{r^2} \tag{2.12}$$

Wobei *r* den Abstand zwischen der Quelle des Signals und dem Ort des Objektes angibt. Da ein Zusammenhang zwischen Empfangsleistung und Entfernung zwischen Objekt und Signalquelle besteht, wird aus dem gemessenen Wert der Leistung ein Längenmaß errechnet. Dieses wird unter Einbeziehung mehrerer, auf diese Weise bestimmte, Entfernungen zu bekannten Punkten in weiteren Verarbeitungsschritten (siehe 2.1) zu einer Positionsangabe umgerechnet. Die nichtlineare Abnahme der Leistungsdichte führt allerdings zu einer mit der Entfernung sinkenden Genauigkeit des Messergebnisses trotz gleich bleibender Auflösung. Dieser Effekt ist in Abbildung 5 illustriert.

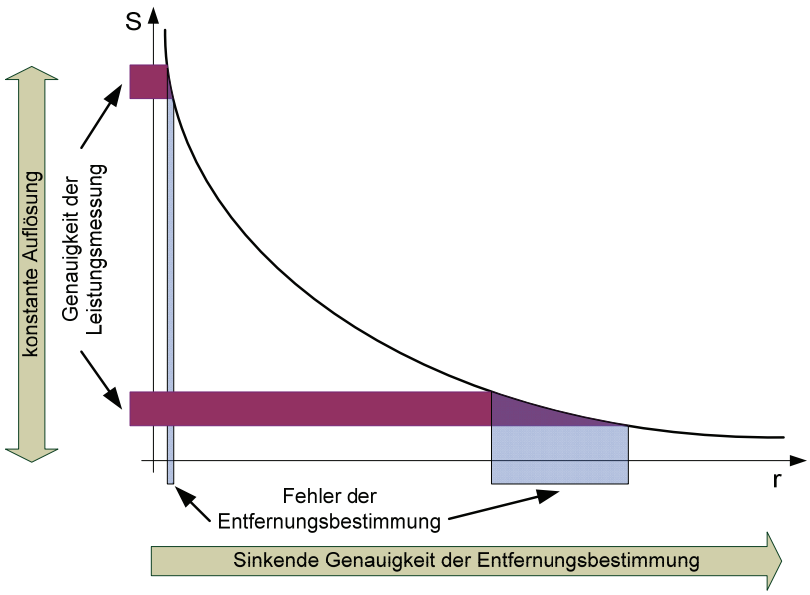


Abbildung 5: Genauigkeitsabnahme der Entfernungsbestimmung aufgrund des mathematischen Zusammenhangs zwischen Leistungsdichte und Entfernung

Aufgrund dieser Eigenschaften können mit diesem Verfahren nur kurze Entfernungen zwischen Signalquelle und Objekt genau bestimmt werden. Ist es erforderlich einen großen Bereich abzudecken, müssen weitere Signalquellen als Referenzpunkte hinzugefügt werden. Dadurch erhöht sich der Messaufwand signifikant. Dennoch findet dieses Verfahren vor allem für Lokalisierungsaufgaben in Sensornetzwerken Anwendung [8] [35].

2.4.4. Phasen- und Frequenzsynchronisation

Roehr et al. haben 2007 [53] einen Ansatz vorgestellt, in dem der Sender ein in der Zeit linear frequenzmoduliertes Signal aussendet, auf das sich der Empfänger in Phase und Frequenz aufsynchronisiert. Anschließend tauschen Sender und Empfänger ihre Rollen; der Empfänger sendet nun seinerseits in entsprechend frequenzmoduliertes Signal aus. Der ehemalige Sender empfängt dieses Signal und vergleicht es mit seinem weiterhin laufenden Oszillator. Aufgrund der Signallaufzeiten zwischen Sender und Empfänger entsteht ein Versatz zwischen empfangenen Signal und dem weiterhin frei laufenden Oszillator. Diese Zeitdifferenz schlägt sich in einem Frequenzunterschied nieder, der linear zum zurückgelegten Weg ist. Diese indirekte Form der Zeitmessung ist in der Lage, Objekte unter Laborbedingungen mit einer Genauigkeit von 20 cm zu lokalisieren. In realitätsnahen Lager- und Büroszenarien verschlechtert sich die Genauigkeit auf etwa 60 cm. Ein Hauptgrund für diese Verringerung der Genauigkeit ist in der Mehrwegeausbreitung zu suchen [53]. Die Phasendifferenz elektromagnetischer Signale wurde bereits im Ubisense-System [64] als eine Informationsquelle integriert. Trotz optimaler Kalibrierung treten in der Praxis Ungenauigkeiten in der Größenordnung von 50 cm auf. Ein Ubisense-System zur Lokalisierung von bis zu vier bewegten Objekten kostet etwa 10.000 €, sodass sich dessen Einsatz aus ökonomischer Sicht nur im Einzelfall rechtfertigen lässt.

2.4.5. Optische Verfahren – Photogrammetrie

Optische Verfahren werten Bildinformationen aus und schließen so auf die Position eines Objektes. Beispielsweise kann die Verschiebung einer Kamera anhand der aufgenommenen Bildinformationen festgestellt werden. Als zweite Möglichkeit kann durch eine stationäre Kamera die Verschiebung von Objekten im aufgenommenen Bild detektiert werden.

Grundsätzlich basiert die Photogrammetrie auf den Gesetzen der Zentralprojektion: Der Objektpunkt, das Projektionszentrum und der Bildpunkt auf dem Bild oder Lichtsensor der Kamera bilden eine Gerade. Ist die Position und Orientierung der Kamera bekannt, kann nach dem Strahlensatz die Position in einer Ebene bestimmt werden. Für eine dreidimensionale Bestimmung des Objektpunktes sind mindestens zwei Bilder notwendig, man spricht von der Mehrbildauswertung oder Mehrbild-Triangulation [49].

Verfahren dieser Art erreichen Auflösungen und Genauigkeiten im Millimeterbereich. Allerdings sind eine ausreichende Beleuchtung und Kontrast sowie freie Sicht zwischen Kamera und Objekt zu garantieren. Insbesondere bei sich verändernden dynamischen Umgebungen können diese Bedingungen nicht immer eingehalten werden. Ebenfalls sind die Verfahren für eine vollständige Bildauswertung nicht trivial sondern erfordern einen hohen Rechenaufwand, was in einer u.U. langen Messdauer resultiert. Für den praktischen Einsatz, auch in Hinblick auf einfache und überall verfügbare Lokalisierungssysteme, ist die Pflicht zur Einhaltung der oben genannten Randbedingungen als nachteilig anzusehen.

2.4.6. Vergleich der vorgestellten Lokalisierungskonzepte

Dieses Kapitel hat verschiedene Konzepte vorgestellt, die für Lokalisierungsverfahren eine Rolle spielen. Jedes Verfahren hat seine eigenen Stärken und Schwächen und wird daher in unterschiedlichen Anwendungen implementiert.

Die Photogrammetrie wird vorrangig im Vermessungswesen eingesetzt, aber auch autonome Roboter werten für die Aufgabe der Wegeplanung zunehmend optische Informationen aus. Dennoch handelt es sich aufgrund der hohen Kosten für die Kamera und die anschließende Auswertung um ein sehr aufwendiges Verfahren.

Die auf Phasen- und Frequenzsynchronisation basierenden Verfahren sind derzeit noch nicht im breiten Einsatz zu finden, stellen aber ein Gebiet dar, auf dem intensive Forschungsarbeiten erfolgen. Die Genauigkeiten, die aktuell im Bereich von einigen Dezimetern liegen, reichen allerdings für die in der Einleitung beschriebenen Szenarien von Lokalisierungsaufgaben nicht aus.

Gleiches gilt für die Verwendung von Schallsignalen. Schallsignale werden vor allem in Sonaranwendungen und im medizinischen Bereich der Ultraschall-Scanner eingesetzt. Bei Ultraschall-Scannern muss das Signal nur Strecken von wenigen Dezimetern zurücklegen auf denen die Signalqualität ausreichend gut erhalten bleibt. Bei Sonar und Tiefensonar ist Wasser der Träger der Schallwellen. Dieses Medium weist eine wesentliche geringere Dämpfungskonstante als Luft auf und lässt somit auch Tiefenmessungen von mehreren Hundert Metern zu. Die hier im Vordergrund stehenden Lokalisierungsaufgaben gehen allerdings von Luft als Umgebungsmaterie aus, wodurch die Reichweite von Schallsignalen stark begrenzt wird. Obwohl die notwendige Technik zur Laufzeitmessung von Schallsignalen aufgrund der im Vergleich zur Lichtgeschwindigkeit langsamen Ausbreitungsgeschwindigkeit ausgesprochen einfach ausfällt, werden bei der Verwendung von Schallsignalen nicht die erforderlichen Genauigkeiten für eine Indoor-Lokalisierung erzielt. Dadurch scheidet auch dieser Ansatz aus.

Die mangelnde Genauigkeit ist ebenfalls der Grund dafür, dass die Entfernungsbestimmung über die Leistungsmessung elektromagnetischer Wellen ungeeignet ist für Distanzen größer als einige Meter. Da diese aber durchaus üblich sind bei Aufgaben der Indoor-Lokalisierung, müssen flächen- bzw. raumdeckend Sender/Empfänger-Paare installiert werden, was zum einen nicht in jedem Raum umsetzbar ist und zum anderen hohe Kosten hervorruft.

Das letzte verbleibende Verfahren ist die Laufzeitmessung von elektromagnetischen Wellen. Abschnitt 2.4.2 hat bereits dargestellt, dass für anspruchsvolle Lokalisierungsaufgaben die Laufzeitmessung elektromagnetischer Wellen mit einer sehr hohen zeitlichen Auflösung erfolgen muss. Auch wurde bereits mehrfach erwähnt, dass der technische Aufwand für das Erreichen derartiger Auflösungen hoch ist, weshalb auch die Indoor-Lokalsierung auf Basis der Ausbreitung von EM-Wellen noch nicht in größerem Maße umgesetzt wird. Die derzeit existierenden Techniken, die in der Lage sind Laufzeitmessungen für EM-Wellen mit ausreichender Genauigkeit durchzuführen, werden im nächsten Abschnitt vorgestellt und verglichen. Da bisher allerdings nur der einfache Zusammenhang zwischen Lokalisierung, Entfernungs- und Laufzeitmessung dargestellt wurde, beginnt das folgende Kapitel zunächst mit einer grundlegenden Betrachtung der Laufzeitmessung und ihre Bedeutung für die in dieser Arbeit im Vordergrund stehenden Lokalisierungsverfahren.

3. Laufzeitmessung, Messen von Zeitintervallen

Laufzeitmessung ist ein Verfahren, welches in vielen technischen Bereichen Anwendung findet. Dazu zählen insbesondere die Physik und Technik [57], die Medizin [60] und die Lokalisierung [50] von Objekten. Die Laufzeitmessung ist in den meisten technischen Anwendungen nur ein Teil des Gesamtverfahrens. Die hier vorliegende Arbeit konzentriert sich auf die Anwendung der Laufzeitmessung im Rahmen von Lokalisierungsverfahren, da sie das größte kommerzielle Einsatzgebiet darstellen. Die in diesem Kapitel getroffenen Aussagen lassen sich aber ohne Probleme auf andere Anwendungsfelder übertragen.

Nachdem Kapitel 2 die Lokalisierungsverfahren Triangulation und Trilateration sowie deren mathematischen Grundlagen vorgestellt hat, wird im folgenden Abschnitt der Zusammenhang zwischen Laufzeitmessung und Lokalisierung verdeutlicht. Nach Zusammenstellung und Erläuterung der grundlegenden Terminologie wird am einfachen Beispiel eines digitalen Zählers die praktische Laufzeitmessung im Detail erläutert. Da die digitalen Zähler nur begrenzt für aktuelle Problemstellungen in der Lokalisierung einsetzbar sind, widmet sich Abschnitt 3.4 aktuellen Verfahren auf dem Gebiet der Laufzeitmessung.

3.1. Basisgrößenbestimmung für Triangulation und Trilateration

Die Basisgrößen beider Verfahren, Winkel für die Triangulation und Entfernungen für die Trilateriation, werden während des Vorgangs der Lokalisierung durch Messung bestimmt. Hierfür nimmt das Verfahren der Laufzeitmessung einen entscheidenden Platz ein. Für die Trilateration wird die Entfernung zwischen dem zu lokalisierenden Objekt und einer bekannten Position über die Zeitdauer *t* bestimmt, die ein Signal für die Überwindung der zwischen beiden liegenden Strecke *s* benötigt. Für die technisch einfachste Bestimmung dieser Laufzeit (time of flight - TOF) sendet das Objekt an der unbekannten Position ein Signal aus, welches von der Referenzposition zurückgesendet wird. Das verwendete Signal legt die Strecke zwischen Objekt und Referenzpunkt zweimal zurück. Nach Gleichung:

$$\Delta s = \frac{1}{2} \Delta t \cdot v \tag{3.1}$$

Ergibt sich die Strecke Δs aus dem Produkt der gemessenen Zeitdifferenz Δt und der Ausbreitungsgeschwindigkeit v des verwendeten Signals. Der Faktor ½ ist durch die Tatsache bedingt, dass das Signal die Strecke zweimal zurücklegt, vom Objekt zum Referenzpunkt und zurück. Praktische Verfahren, die nach diesem Konzept arbeiten, sind z.B. Radar und Sonar. Vorteilhaft ist die Einfachheit des Konzeptes, es werden keine am

Objekt nicht verfügbaren Informationen verwendet. Insbesondere bei den genannten Verfahren Radar und Sonar erfolgt die Rücksendung des Signals automatisch durch den physikalischen Effekt der Reflexion. In andern Anwendungen muss ggf. am Referenzpunkt eine Rückantwort generiert und ausgesendet werden. Die dafür notwendige Zeit ist in Gleichung (3.1) nicht explizit berücksichtigt, sondern in der gemessenen Zeitdauer Δt enthalten. Dadurch wird das Messergebnis verfälscht.

Umgangen wird dieser Effekt durch Systeme, die nur die Laufzeit für eine Wegstrecke zwischen Referenzpunkt und Objekt bestimmen. Dieses Verfahren (time of arrival – TOA) trägt in eine Nachricht die aktuelle Zeit an einem der beiden Orte, z.B. dem Referenzpunkt, ein und versendet die Nachricht an das zu lokalisierende Objekt. Erhält das Objekt die Nachricht, errechnet es aus der in der Nachricht eingetragenen Zeit und der eigenen aktuellen Zeit die Zeitdifferenz, die für die Übertragung der Nachricht vergangen ist. Dafür ist es allerdings notwendig, dass eine Zeitsynchronisation zwischen beiden Orten vorgenommen wird, was technisch aufwendig ist. Eine weitere Fehlerquelle ist die in der Nachricht vermerkte Zeit. Entspricht diese nicht dem Zeitpunkt des endgültigen Versands der Nachricht sondern erfolgen vorher noch weitere Verarbeitungsschritte, wird das Ergebnis der Wegbestimmung verfälscht.

Eine Verbesserung wird durch die gleichzeitige Verwendung zweier unterschiedlicher Signale erzielt. Sendet der Referenzpunkt gleichzeitig eine Nachricht mit zwei unterschiedlichen Ausbreitungsgeschwindigkeiten aus, misst das Objekt die Zeitdifferenz zwischen dem Eintreffen der ersten Nachricht und dem Eintreffen der zweiten Nachricht. Dieses Verfahren (time difference of arrival – TDOA) benötigt keine in die Nachrichten eingebetteten Zeitinformationen. Nachteilig wirkt sich aus, dass Signale mit unterschiedlichen Ausbreitungsgeschwindigkeiten unterschiedlichen physikalischen Eigenschaften, wie z.B. Dämpfung, Reichweite und Störanfälligkeit, unterliegen. Abbildung 6 stellt die drei angesprochenen Methoden TOF, TOA und TDOA gegenüber.

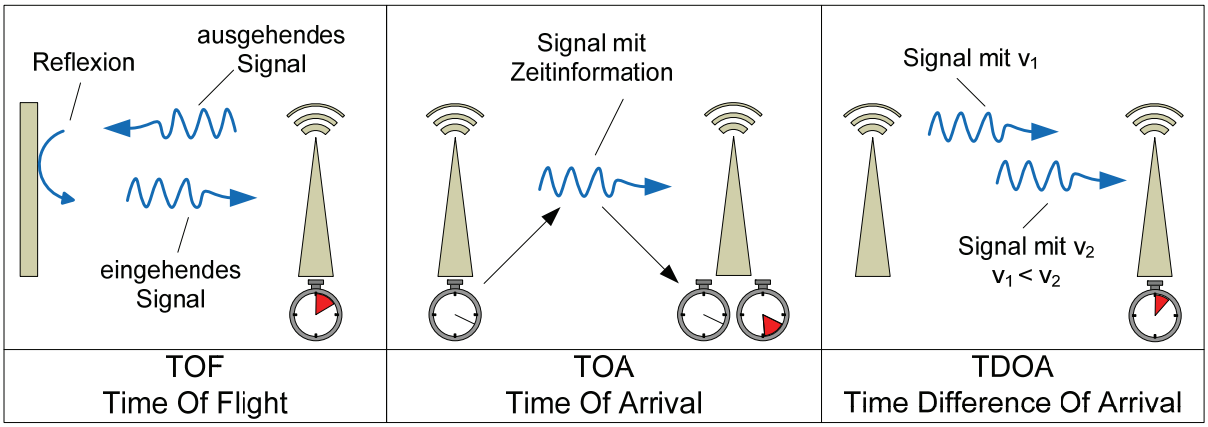


Abbildung 6: schematische Darstellung Entfernungsbestimmung über verschiedene Formen der Laufzeitmessung

Die Orientierung eines Objektes lässt sich mit Hilfe zweier am Objekt befestigter Empfänger Bestimmen. Die Bestimmung des Winkels ist insofern technisch weniger aufwändig als die Entfernungsbestimmung: Im Vergleich zur Triangulation sind keine Zeitinformationen zwischen den Referenzpunkten und dem Objekt im Punkt *P* auszutauschen. Somit entfällt der hohe technische Aufwand für eine Zeitsynchronisation zwischen den Referenzpunkten und dem Objekt. Alle zur Winkelbestimmung notwendigen Daten werden an einem Ort, mit Hilfe nur eines einzigen Signals, erhoben. Abbildung 7 zeigt eine prinzipielle Darstellung des Vorgangs.

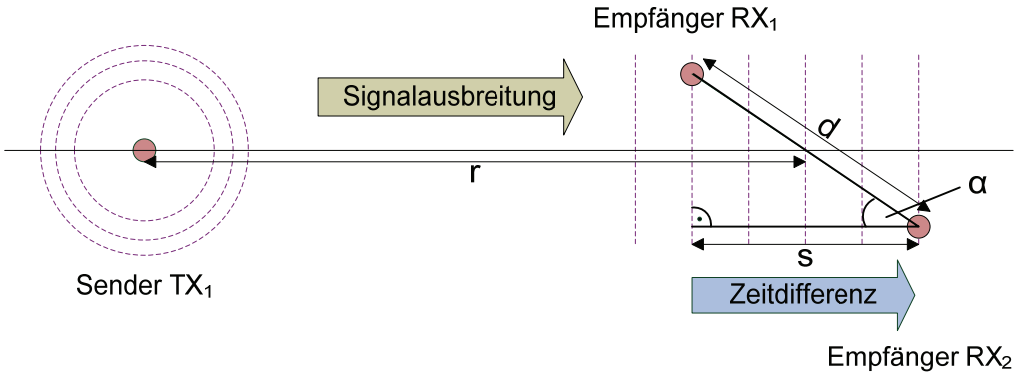


Abbildung 7: Winkelmessung über Laufzeitdifferenzen

Ein Sender, der sich am Referenzpunkt befindet, sendet ein Signal aus. Dieses Signal breitet sich kugelförmig aus. Am Ort des zu lokalisierenden Objekts, der zum Referenzpunkt die Entfernung *r* besitzt, befinden sich die zwei Empfänger RX_1 und RX_2. Für $r \gg d$ können am Ort der Empfänger ebene Signalfronten angenommen werden. Das Messsystem bestimmt die Zeitdifferenz, die das Signal benötigt, um den Weg *s* zwischen beiden Empfängern zu überwinden. Mit der gemessenen Zeitdifferenz Δt und dem bekannten Abstand *d* beider Empfänger kann der Winkel α zwischen der Verbindungslinie beider Empfänger und der Signaleinfallsrichtung bestimmt werden nach:

$$cos\,\alpha = \frac{s}{d} = \frac{v \cdot \Delta t}{d} \tag{3.2}$$

Zusammenfassend wird deutlich, dass in allen hier gezeigten Beispielen zur Entfernungs- bzw. Winkelbestimmung das Ermitteln einer Zeitdifferenz eine grundlegende Rolle spielt. Die folgenden Abschnitte befassen sich daher detailliert mit dem Thema der Laufzeitmessung.

3.2. Terminologie

Der Begriff *Laufzeitmessung* beschreibt das Messen eines Zeitintervalls, das ein beliebiges Signal zum Durchlaufen einer Messstrecke benötigt. Verallgemeinernd lässt sich die Laufzeitmessung als ein Vorgang betrachten, der die Zeitdauer zwischen zwei Ereignissen bestimmt. Als Ereignisdefinition eignen sich zum Beispiel der Eintritt des Signals in die

Messstrecke und der Austritt des Signals aus der Messstrecke. In technischen Systemen sind diese Ereignisse mit feststellbaren Änderungen physikalisch messbarer Größen verknüpft. Am häufigsten wird eine Spannungsänderung zur Ereignisdefinition genutzt. Da es sich bei der Zeitdifferenz zwischen den Ereignissen um eine analoge Größe handelt, erfolgt bei der technischen Laufzeitmessung eine Zeitquantisierung. Quantisierung bedeutet in diesem Fall, dass die Messgröße in eine Darstellung überführt wird, in der die analoge Größe *Zeit* in eine endliche Anzahl Teilbereiche aufgeteilt ist und der Zeitdifferenz ein Symbol aus dieser endlichen Menge zugeordnet wird. Als *Auflösung* wird die Größe eines Teilbereiches definiert. Der Begriff *Messbereich* beschreibt die maximale, durch das Messsystem darstellbare, Zeitdifferenz.

Ziel jeder Laufzeitmessung ist eine sichere und exakte Bestimmung der Zeitdifferenz zwischen den Ereignissen. Dies stellt die größte Herausforderung an Laufzeitmesssysteme dar, insbesondere wenn man berücksichtigt, dass aktuell realisierte Genauigkeiten bereits im Pikosekundenbereich liegen [17] [2] [13]. Digitale Zähler sind die am einfachsten aufgebauten Zeitmesssysteme. Obwohl sie keine Auflösungen im Pikosekundenbereich erlauben, sind sie für viele Anwendungen ausreichend genau und werden daher entsprechend häufig verwendet. Der folgende Abschnitt geht im Detail auf die Realisierung solcher Systeme ein und stellt ihre wichtigsten Eigenschaften dar.

3.3. Zähler als einfachste Methode der Zeitmessung

Die Verwendung eines integrierten digitalen Zählers ist das technisch einfachste und am günstigsten zu implementierende Verfahren zur Laufzeitmessung. Viele Mikrocontroller und Mikroprozessoren bieten bereits integrierte Zähler oder Timer (Rückwärtszähler) an. Der Zähler wird mit der dem Startereignis folgenden positiven Taktflanke des freilaufenden Systemtaktes *clk* gestartet. Anschließend wird mit jedem Taktimpuls des Systemtaktes *clk* der Statuswert des Zählers um den Wert 1 erhöht (bei Rückwärtszählern verringert). Der Zählvorgang endet mit der letzten positiven Taktflanke vor dem Auftreten des Stopsignals. Aus der Differenz der Zählerstände, also der Anzahl verstrichener Taktperioden, und der bekannten Periode des Taktes *clk* wird das zwischen beiden Ereignissen liegende Zeitintervall bestimmt. Abbildung 8 verdeutlicht dieses Prinzip.

Die Auflösung des Messsystems ist durch die Frequenz *f* des Systemtakts *clk* definiert. Das kleinste, sicher bestimmbare Zeitintervall ergibt sich durch:

$$\Delta t_{min} = T_0 = \frac{1}{f} \tag{3.3}$$

Mit anderen Worten: Start- und Stop-Ereignis müssen mindestens eine Taktperiode auseinander liegen.

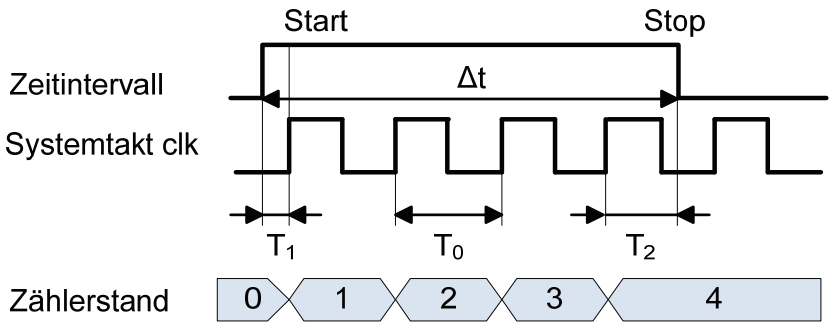


Abbildung 8: Prinzip des digitalen Zählers zur Messung der Zeit zwischen zwei Ereignissen

Die gemessene Zeitdauer *Δt* errechnet sich im Beispiel aus Abbildung 8 mit Hilfe des Zählerstandes *cv* und der Periode des Taktsignals T_0 zu:

$$\Delta t = (cv - 1) \cdot T_0 = 3 \cdot T_0 \tag{3.4}$$

Offensichtlich ist dieser Wert aber kleiner als die Zeitdifferenz *Δt*, weil die beiden in der Abbildung dargestellten Größen T_1 und T_2 durch den Zähler nicht berücksichtigt werden können. Die Genauigkeit des Ergebnisses ist somit von der zwischen dem Start-Ereignis und der nächsten positiven Taktflanke liegenden Zeitdauer T_1 und der zwischen dem Stop-Ereignis und der letzen positiven Taktflanke liegende Zeitdauer T_2 abhängig. Da beide maximal der Taktperiode T_0 entsprechen können, beträgt die Genauigkeit der Einzelmessung mit $2T_0$ bzw. $+/-T_0$. Die relative Genauigkeit verschlechtert sich mit kürzeren Zeitintervallen. Dies hat zur Folge, dass für eine hohe Genauigkeit und eine hohe Auflösung hohe Taktraten benötigt werden. Darin ist einer der wesentlichen Nachteile der Verwendung integrierter Zähler zur Laufzeitmessung zu sehen.

Berücksichtigt man allerdings das im Allgemeinen asynchrone Verhalten der Ereignisse *Start* und *Stop* zum Systemtakt *clk*, verringern mehrfache Messungen eines identischen Zeitintervalls und eine anschließende statistische Auswertungen den Fehler der Zeitmessung. Die Werte T_1 und T_2 ändern sich bei jeder Einzelmessung, weil sich das Zeitintervall relativ zum Systemtakt *clk* verschiebt. Misst man wiederholt das gleiche Zeitintervall *Δt*, werden während der Messungen zwei Ergebnisse $TI_1 < \Delta t$ und $TI_2 > \Delta t$ auftreten:

$$TI = \begin{cases} TI_1, TI_1 < \Delta t \\ TI_2 = TI_1 + T_0, TI_2 > \Delta t \end{cases} \tag{3.5}$$

Dieses Ergebnis ist dadurch bedingt, dass je nach Lage des Zeitintervalls zum Systemtakt der Zähler *n*-mal zählt oder *(n+1)*-mal. Die folgende Abbildung verdeutlicht diesen Sachverhalt.

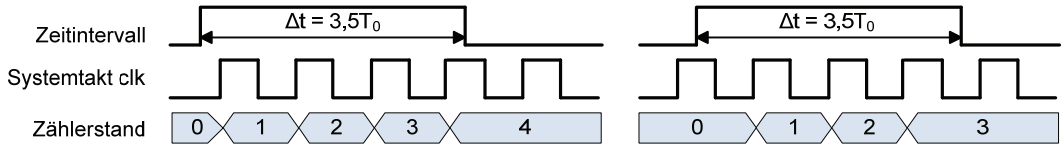


Abbildung 9: Darstellung der Entstehung unterschiedlicher Messergebnisse trotz identischen Zeitintervalls bei Verwendung digitaler Zähler

Die Häufigkeit des Auftretens beider Werte hängt vom gebrochenen Rest *R* des Verhältnisses $\Delta t/T_0$ ab. Die Auftretenswahrscheinlichkeit des Wertes TI_1 wird durch $p(TI_1)=R$ definiert, während der Wert TI_2 mit der Wahrscheinlichkeit $q(TI_2)=(1-R)$ auftritt.

Werden *n* Messungen durchgeführt, bei denen mit der Häufigkeit N_1 das Ergebnis TI_1 und mit der Häufigkeit N_2 das Ergebnis TI_2 auftritt, so ergibt sich für den Mittelwert T_n:

$$T_n = \frac{TI_1 \cdot N_1 + TI_2 \cdot N_2}{n} \tag{3.6}$$

Die Genauigkeit des gemessenen Wertes für das Zeitintervall Δt kann unter Verwendung der allgemeinen Vorschrift für die Standardabweichung

$$\sigma = \sqrt{\frac{\sum_{i=0}^{n} (x_i - \overline{x})^2}{n-1}} \tag{3.7}$$

berechnet werden. Berücksichtig man die Tatsache, dass für x_i lediglich zwei Werte (TI_1 und TI_2) entstehen können, wobei jeder mit der Häufigkeit N_1 bzw. N_2 auftritt, kann formuliert werden:

$$\sigma = \sqrt{\frac{N_1(TI_1 - T_n)^2 + N_2(TI_2 - T_n)^2}{n-1}} \tag{3.8}$$

Setzt man Gleichung (3.6) für den Term T_n ein und berücksichtigt Gleichung (3.5), ergibt sich:

$$\sigma = \sqrt{\frac{T_0{}^2 N_1 N_2}{(n-1)(N_1 + N_2)}} \tag{3.9}$$

Unter Verwendung $n = N_1 + N_2$, $N_1 = p \cdot n, N_2 = q \cdot n$ und der Nährung $n - 1 \approx n$ für große *n*, ergibt sich als Unsicherheit für die Einzelmessung:

$$\sigma = T_0 \sqrt{pq} \tag{3.10}$$

Wie in Gleichung (3.10) zu erkennen, hat diese ihren Maximalwert bei $p=q=0{,}5$. Die Unsicherheit des Mittelwertes T_n ist die Standardabweichung des Mittelwertes nach:

$$\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n}} = \frac{T_0\sqrt{pq}}{\sqrt{n}} \tag{3.11}$$

Damit ist garantiert, dass für große n der mittlere Quantisierungsfehler $T_n - \Delta t$ gegen null geht. Dadurch wird die Genauigkeit der Zählermethode deutlich verbessert, bei 100 Messungen ist sie zehnmal höher als bei einer Einzelmessung. Um dies zu erreichen, muss allerdings auch eine gegenüber der Einzelmessung um den Faktor 100 vergrößerte Messdauer akzeptiert werden.

Weiterhin kann in Lokalisierungsverfahren mit mobilen Objekten nicht garantiert werden, dass eine Mehrfachmessung der gleichen Zeitdifferenz möglich ist. Wird also ein solcher Zähler zur Laufzeitmessung im Rahmen von Lokalisierungsverfahren eingesetzt, eignet er sich vorrangig für Signalformen, die sich langsam ausbreiten, wie z.B. Schall und Ultraschall. Wie bereits in Abschnitt 3.1 beschrieben, verwendet die Trilateration Entfernungsangaben, die mit Hilfe der Laufzeitmessung errechnet werden. Durch die hohe Ausbreitungsgeschwindigkeit elektromagnetischer Wellen von ca. 300.000 km/s ist bereits eine Taktfrequenz von 3 GHz erforderlich, um eine räumliche Auflösung des Lokalisierungsverfahrens von 0,1 m zu erzielen. Derartig hohe Taktraten sind in mobilen Systemen unüblich. Die zählerbasierte Variante der Laufzeitmessung erfüllt daher nicht die Anforderungen, die ein hochpräzises Lokalisierungsverfahren an das Messsystem stellt.

Die Literatur, zum Beispiel [63], bietet unzählige Varianten digitaler Zähler. Der einfachste Zähler ist aus einer Reihe hintereinandergeschalteter FlipFlops, im allgemeinen Toggle-FlipFlops oder rückgekoppelte D-FlipFlops, aufgebaut. Jedes FlipFlop repräsentiert eine Ziffer im Zählerstand. Der Ausgang eines jeden FlipFlops ist mit dem Takteingang des nachfolgenden FlipFlops verbunden. Jedes FlipFlop besitzt somit sein eigenes Taktsignal, weshalb diese Form auch als asynchroner Zähler bezeichnet wird. Verwenden alle FlipFlops denselben Takt *clk*, spricht man von einem synchronen Zähler.

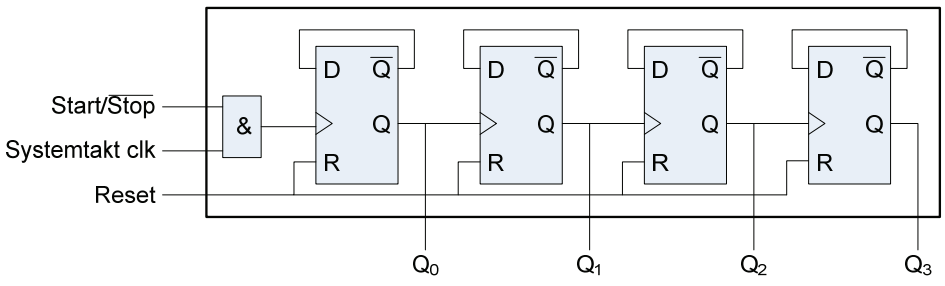


Abbildung 10: einfachste Grundschaltung eines binären, asynchronen Rückwärtszählers

Die Größe des Messbereiches wird durch die Bit-Breite des Zählers definiert. Mit einem *n* Bit breiten Zähler können 2^n verschiedene Zustände dargestellt werden, was in einem Messbereich von $2^n T_0$ resultiert. Vorteil dieser Darstellung ist die einfache Möglichkeit zur

Vergrößerung des Messbereiches. Um eine Verdoppelung des Messbereiches zu erzielen, muss lediglich ein weiteres FlipFlop hinzugefügt werden und so die Breite des Zählers von *n* auf *n+1* erhöht werden.

3.4. Aktuelle Verfahren zur präzisen Laufzeitmessung

Nachdem das technisch einfachste Laufzeitmessverfahren ausführlich vorgestellt wurde, bietet dieser Abschnitt eine Zusammenstellung über andere, praktische eingesetzte Verfahren. Diese bieten eine höhere Auflösung als das Zählerverfahren und stellen somit eine bessere Basis für Lokalisierungssysteme zur Verfügung.

3.4.1. Time Stretching

Wie bereits ausgeführt bietet das Zählerverfahren eine maximale Auflösung, die mit der Taktfrequenz *f* des Zählers korreliert und damit die minimale Größe des messbaren Zeitintervalls begrenzt. Ansatz des Time Stretching ist eine Konvertierung des zu bestimmenden Zeitintervalls nach

$$\Delta t \longrightarrow \Delta t' = k\Delta t \tag{3.12}$$

Dadurch wird das reale Zeitintervall *Δt* um den Faktor *k* (mit *k*>1) verlängert. Abbildung 11 zeigt den schematischen Aufbau dieses Konzeptes.

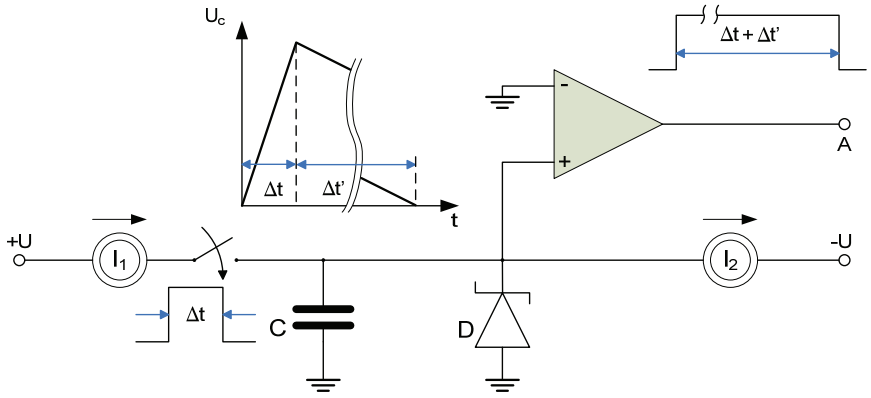


Abbildung 11: Grundschaltung der Laufzeitmessung nach der Time Streching - Methode

Zur Realisierung der Zeitkonvertierung nach Gleichung (3.12) wird eine kapazitive Komponente *C* in das System integriert. Diese wird während des Zeitintervalls *Δt* aufgeladen. Anschließend wird sie langsam in der Zeit *Δt'* entladen und die Summe der Zeitintervalle *Δt'* und *Δt* am Ausgang *A* in Abbildung 11 durch ein Zählersystem (nicht dargestellt) gemessen.

Die Kapazität *C* wird durch den konstanten Strom $I = (I_1 - I_2)$ aufgeladen. Nach dem Aufladen wird die Kapazität *C* über den Strom $I_2 \ll I_1$ entladen. Der Proportionalitätsfaktor *k* ist von den Parametern der beiden Stromquellen I_1 und I_2 abhängig. Da *k* das Verhältnis der Ladedauer

Δt und der Entladedauer *Δt'* angibt und beide Vorgänge mit dem Ladestrom (I_1-I_2) und dem Entladestrom I_2 erfolgen, ergibt sich *k* zu:

$$k = \frac{(I_1 - I_2)}{I_2} \tag{3.13}$$

Der Messbereich ist durch die Größe der Kapazität *C* begrenzt, da diese definiert, welche Ladungsmenge auf dem Kondensator gespeichert werden kann. Aufgrund der linearen Verhältnisse kann die maximal messbare Zeitdauer t_{max} angegeben werden mit:

$$t_{\max} = \frac{C \cdot U}{(I_1 - I_2)} \tag{3.14}$$

Praktische Umsetzungen der in Abbildung 11 dargestellten Architektur realisieren Faktoren *k* = 10.000, und erreichen so Auflösungen von 1 ps [31]. Allerdings können durch Jitter lediglich Genauigkeiten von 5 ps angegeben werden. Die einschlägige Literatur zeigt Anwendungen dieses Konzepts in verschiedenen Bereichen, unter anderem im Bereich Physik [68], Laser-Entfernungsmessung [41] [42] und Messung dynamischer Parameter integrierter Schaltkreise [32].

Vorteil dieses Ansatzes ist, dass das Zeitintervall *Δt'* wiederum mit dem schon dargestellten Zählersystem gemessen werden kann. Die zeitliche Auflösung des verwendeten Zählers beträgt zwar nach wie vor $T_0=1/f$, wobei *f* wieder die Taktfrequenz des Zählers angibt. Dadurch, dass das zu messende Zeitintervall *Δt* um den Faktor *k* verlängert wurde, verbessert sich die zeitliche Auflösung ebenfalls um den Faktor *k*. Gleiches gilt für die Genauigkeit der Einzelmessung. Die Messdauer erhöht sich zwar auch um den Faktor *k*, allerdings ist eine Bewertung des dadurch entstehenden Nachteils vom jeweiligen Einsatzgebiet und den daraus resultierenden Anforderungen an die Geschwindigkeit der Messung abhängig.

3.4.2. Time Stretching und Analog-Digital-Converter

Dieses Verfahren, im Englischen auch als Time-to-Digital Conversion (TDC) bezeichnet, basiert auf dem gleichen Ansatz wie das Time Stretching. Auch hier wird während des zu messenden Zeitintervalls *Δt* eine Kapazität *C* aufgeladen. Der durch die Aufladung der Kapazität *C* eingestellte Spannungswert U_c wird mit Hilfe eines angeschlossenen Analog-Digital-Converters (ADC) dann aber direkt in einen digitalen Wert überführt. Dieses digitale Datum wird anschließend in einen Zeitwert konvertiert. Abbildung 12 zeigt eine Darstellung dieses Prinzips.

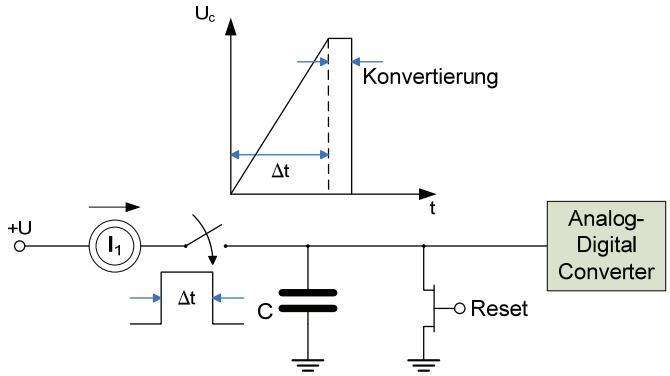


Abbildung 12: direkte Umwandlung eines zeitabhängigen Spannungswertes in ein digitales Datum – „Time-to-Digital Conversation“

Die Auflösung und Genauigkeit dieses Verfahrens werden vorrangig durch die Eigenschaften des eingesetzten ADC bestimmt. Auch die Größe des Stroms I_1 hat einen Einfluss auf die erzielbare Auflösung, da sie bestimmt, wie schnell die Spannung U_c am Kondensator C zunimmt. Je schneller die Kapazität aufgeladen wird, desto kleiner ist die Zeitdifferenz Δt, die notwendig ist, um am ADC unterscheidbare Eingangswerte zu erzeugen. Der Anstieg der Kennlinie $U_c(t) = f(I_1)$ begrenzt den maximalen Messbereich der Schaltung. Aus Gleichung (3.14) wird ersichtlich, dass für das maximal messbare Zeitintervall t_{max} gilt $t_{max} \sim 1/I_1$ sowie $t_{max} \sim C$.

Erreicht werden mit diesem Verfahren zeitliche Auflösung im Bereich zwischen 1 ps und 20 ps. Aufgrund dieser Werte wird das Verfahren, neben den im obigen Abschnitt angeführten Verfahren, auch in Hochpräzisionsmessgeräten [30] [61] eingesetzt. Allerdings gilt für dieses Verfahren, wie auch für das Time Stretching, dass es für eine vollständig integrierte Bauweise Nachteile besitzt. Die Erzeugung von kapazitiven Komponenten in integrierten Schaltkreisen ist schwierig, die diskrete Realisierung ist anfällig gegenüber Temperaturschwankungen und Störungen. Im Vergleich zur Time Stretching Methode liegt der Vorteil dieses Konzeptes darin, dass die Messdauer nicht signifikant erhöht wird, sondern nur um die für die Konvertierung notwendige Zeit wächst.

3.4.3. Die Vernier-Methode

Ein weiteres Verfahren, welches vorrangig zur Messung sehr kurzer Zeitintervalle eingesetzt wird, ist die Vernier-Methode [36]. Sie basiert auf dem Einsatz zweier startbarer Frequenzgeneratoren, die Rechteckimpulse mit geringfügig unterschiedlichen Frequenzen erzeugen. Der Start-Impuls, also der Beginn des zu messenden Zeitintervalls, startet den Generator mit der Frequenz f_1. Das durch den Stop-Impuls markierte Ende des Zeitintervalls Δt startet den Generator mit der Frequenz $f_2 < f_1$. Das Messsystem zählt die Anzahl der Rechteckimpulse beider Frequenzen bis zu dem Zeitpunkt, an dem beide Rechtecksignale

gleichzeitig eine positive Flanke aufweisen oder die Phasenverschiebung zwischen dem vorlaufenden Signal f_1 und dem nachlaufenden Signal f_2 negativ wird.

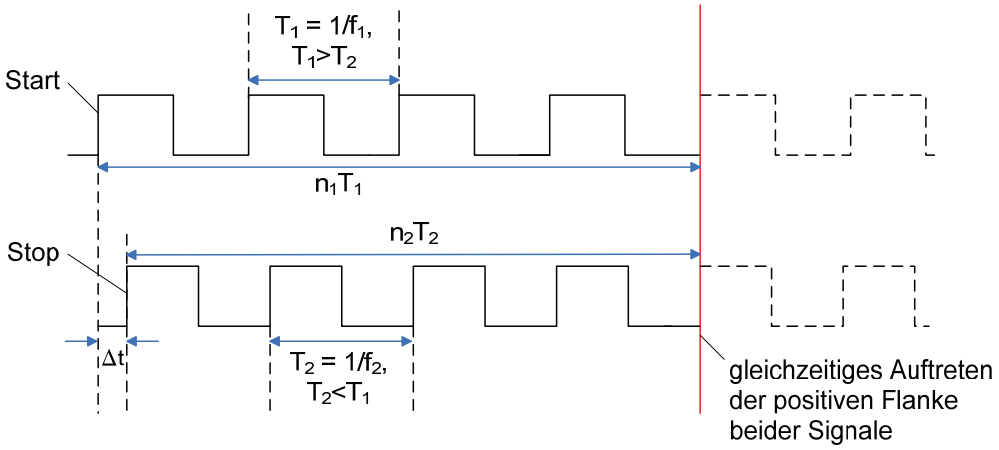


Abbildung 13: Darstellung des Konzepts der Vernier Methode, die Verwendung zweier digitaler Zähler mit unterschiedlichen Frequenzen

Das Zeitintervall Δt lässt sich berechnen über

$$\Delta t = n_1 T_1 - n_2 T_2 \tag{3.15}$$

Die Auflösung R wird durch die Differenz beider Signalperioden bestimmt. Es gilt:

$$R = T_1 - T_2 \tag{3.16}$$

Damit ist eine Unabhängigkeit von der absoluten Größe beider Taktperioden gegeben, worin einer der wesentlichen Vorteile gegenüber den klassischen Zählern besteht. Die Messdauer t_{conv} des Systems wird durch die Auflösung r, bzw. die beiden Taktperioden T_1 und T_2, bestimmt.

$$t_{conv} = \frac{T_1 T_2}{r} + \Delta t \tag{3.17}$$

Der Messbereich ist durch die Maximalwerte der verwendeten Zähler bestimmt, unterliegt aber keiner physikalischen Grenze, wie es zum Beispiel beim Time Stretching Verfahren der Fall ist. Allerdings ist es erforderlich, dass beide Signalgeneratoren eine hohe Stabilität und Präzision aufweisen, wodurch die Implementierung dieses Verfahrens erschwert wird. Aus diesem Grund wird das Verfahren vorrangig in Systeme zusammen mit einem klassischen Zähler integriert. Die Vernier-Methode wird dann nur für die Messung der kurzen Zeitintervalle zwischen dem Start-/Stop-Ereignis und der nächsten/vorhergehenden positiven Flanke des Taktsignals des klassischen Zählers verwendet (siehe T_1, T_2 in Abbildung 8). Der klassische Zähler misst den zwischen T_1 und T_2 liegenden Zeitabschnitt.

3.4.4. Das Konzept der Tapped Delay Lines

Die bereits vorgestellten Verfahren zur Laufzeitmessung haben zwar ihre Leistungsfähigkeit mit Auflösungen im unteren, teilweise einstelligen Pikosekundenbereich bereits unter Beweis gestellt, allerdings ist auch deutlich zu erkennen, dass sie für eine einfache Implementierung in Kleinstgeräten aufgrund ihrer Komplexität nicht geeignet sind oder nicht die erforderliche Genauigkeit liefern. Eine konzeptionell einfache Methode zur Laufzeitmessung stellen die so genannten Tapped Delay Lines (TDL) dar [30] [33] [34].

Zur Zeitmessung eingesetzte Tapped Delay Lines sind durch eine sequentielle Struktur von Verzögerungselementen gekennzeichnet, wie in Abbildung 14 dargestellt. Das Signal S_1, welches die Kette durchläuft, wird in jedem Element um das Delay τ verzögert. Idealerweise besitzen alle Elemente die gleiche Verzögerungszeit τ. Dadurch erfolgt eine Zeitdiskriminierung des Signals S_1 mit der Auflösung τ. Mit anderen Worten: Das Signal S_1 benötigt von der sicheren Erkennung in Kettenelement *n* bis zur sicheren Erkennung in Element *n+1* die Zeit τ. Als Moment der Erkennung in einem Element wird der Zeitpunkt definiert, zu dem das Signal am Eingang des FlipFlops innerhalb des Elements anliegt. Um die Dauer des Zeitintervalls Δt zu bestimmen, sichert das Stopsignal S_2 ideal gleichzeitig den Status der FlipFlops in allen Kettenelementen. Je nach Dauer des Zeitintervalls Δt hat das Signal S_1 bis zum Auftreten des Stopsignals eine bestimmte Anzahl Kettenelemente durchlaufen. In Abbildung 14 sind diese Elemente durch den Ausgangswert *Q=1* symbolisiert.

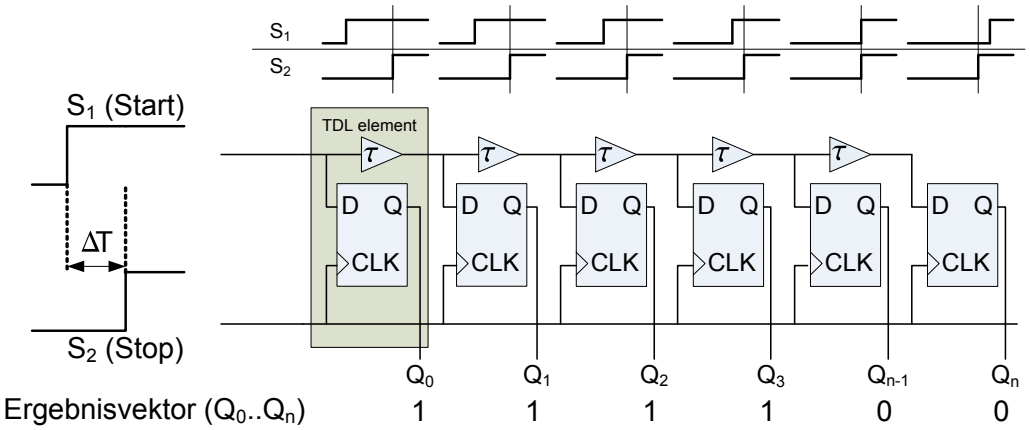


Abbildung 14: Grundstruktur der Tapped Delay Lines

Die Bestimmung des Zeitintervalls erfolgt einfach über das Produkt aus der Anzahl der durch S_1 durchlaufenen Kettenelemente und der Verzögerungszeit τ eines jeden Elementes.

$$\Delta t = n\tau \tag{3.18}$$

Die Genauigkeit des Ergebnisses hängt stark von der Genauigkeit der Delays τ_i ab. Geht man von ideal gleichen Verzögerungszeiten aus, hat sie ebenfalls den Wert τ.

Als vorteilhaft sind zunächst die Unabhängigkeit von einem Systemtakt *clk* und der Verzicht auf kapazitive Elemente zu sehen. Dadurch wird eine vollständige Intergrierbarkeit dieser Struktur in digitale Schaltkreise ermöglicht. Integrierte Schaltkreise, vor allem kundenspezifizierte maskenprogrammierte Aplication Specific Integrated Circuits (ASICs), können sehr kleine Verzögerungszeiten τ realisieren. Die einfachste Möglichkeit ist die Hintereinanderschaltung einer geraden Anzahl Inverter in jedem Element. Der Nachteil, der aus kleinen Verzögerungszeiten τ resultiert, ist der begrenzte Messbereich der Struktur, der mit $n\tau$ definiert ist. Im Vergleich zu zählerbasierten Systemen, wo nur ein weiteres FlipFlop zur Verdopplung des Messbereichs benötigt wird, sind in Tapped Delay Lines *n* weitere FlipFlops zu integrieren, um den Messbereich auf $2n\tau$ zu vergrößern. Ferner sind die hohen Entwicklungskosten für die oben genannten ASICs zu berücksichtigen, die sich erst bei großen Stückzahlen (>10000) rentieren.

Um die Verwendung speziell angefertigter Schaltkreise zu umgehen, werden Tapped Delay Lines vorzugsweise auf freiprogrammierbaren Schaltkreisen (Field-Programmable Gate Array – FPGA) implementiert. Um ein grundlegendes Verständnis des Aufbaus und der Funktion dieser Schaltkreise zu vermitteln, werden die FPGAs im nächsten Kapitel vorgestellt. Anschließend behandelt Kapitel 5 ausführlich FPGA-basierte Implementationen von Tapped Delay Lines, um den aktuellen Stand der Forschung aufzuzeigen.

4. Field-Programmable Gate Arrays

Field-Programmable Gate Arrays (FPGAs) sind integrierte Schaltkreise, welche frei konfiguriert werden können und somit in der Lage sind, beliebige Hardware-Strukturen nachzubilden. Sie werden effizient für die Entwicklung anwendungsspezifischer Hardware-Komponenten und die Fertigung von integrierten Schaltungen mit geringen Stückzahlen eingesetzt. Hier sind zum Beispiel die Bereiche der Netzwerktechnik [67] und der Industrieanlagensteuerung [24] zu nennen. Insbesondere für die Implementierung der Tapped Delay Lines bieten sie günstige strukturelle Eigenschaften, die in den folgenden Abschnitten dargestellt werden.

4.1. Grundstruktur der Field-Programmable Gate Arrays

Abbildung 15 gibt eine Übersicht über die Grundstruktur eines FPGAs. Das Grundelement eines FPGAs ist die Logikzelle (engl. logic cell). Die Logikzelle besteht im Wesentlichen aus einer konfigurierbaren Tabelle, der Look-Up Table (LUT), und einem Register. Unter den FPGA-Herstellern haben sich verschiedene Bezeichnungen für die Logikzelle etabliert. Der Hersteller Altera bezeichnet die Logikzelle als Logic Element (LE), die Firma Xilinx verwendet den Begriff Configurable Logic Block (CLB). Andere, häufig in der Literatur zu findende Termini sind „logic block“, „adaptive logic module“ und „slice“. Da im Rahmen dieser Arbeit auf FPGAs des Herstellers Altera zurückgegriffen wird, wird im den folgenden Ausführungen immer die Bezeichnung Logikelement (LE) verwendet. Unabhängig von der jeweiligen Bezeichnung dient das Logikelement dazu, einfachste und grundlegende Hardware-Strukturen abzubilden.

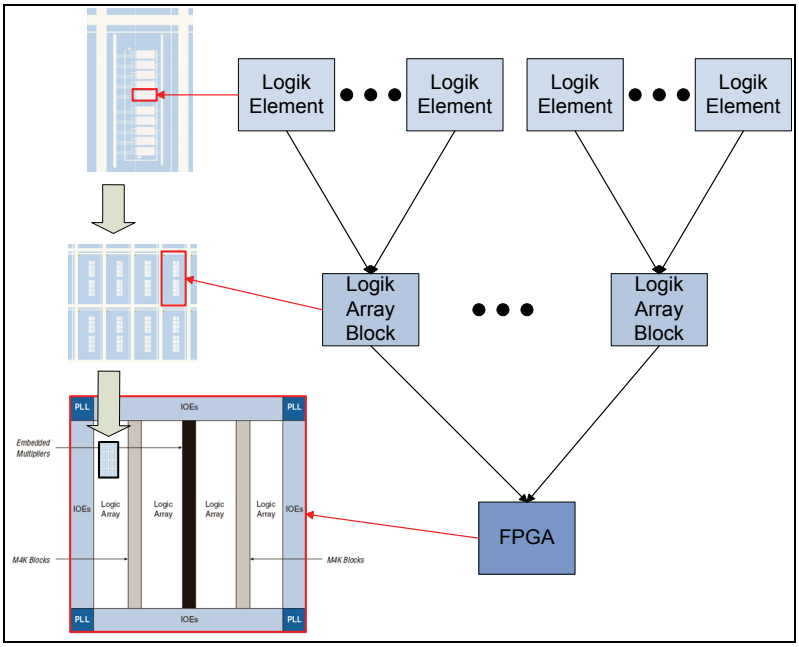


Abbildung 15: Grundstruktur eines FPGAs (Altera Cyclonell Architektur)

Um komplexere, aus einfachen Gattern zusammengesetzte, Hardware-Komponenten effizient zu synthetisieren, werden mehrere Logikelemente zu einem Logic Array Block (LAB) zusammengefasst. Der LAB zeichnet sich vor allem durch schnelle Signalleitungen zwischen den in ihm befindlichen Logikelementen aus. Bei Altera FPGAs sind meist 10 oder 16 Logikelemente zu einem LAB zusammengefasst.

Eine Vielzahl regelmäßig angeordneter LABs wiederum bildet den konfigurierbaren Bereich des FPGAs. Das FPGA wird komplettiert durch Sonderstrukturen, zu denen Elemente zur Taktgenerierung, spezielle Speicherkomponenten, eingebettete Hardware-Multiplizierer und Ein-/Ausgabemodule gehören. Da Aufbau und Funktion der Logikelemente sowohl für die Implementierung aktueller Laufzeitmesssysteme als auch für die Umsetzung der in dieser Arbeit vorgestellten Architektur von essentieller Bedeutung sind, gibt der folgende Abschnitt eine detaillierte Beschreibung.

Dem geneigten Leser, der umfangreichere Ausführungen zur FPGA-Technologie und -Struktur wünscht, sei [44] empfohlen.

4.2. Aufbau und Funktionalität der Logikelemente

Wie bereits in der Übersicht zur Grundstruktur der FPGAs dargestellt, ist das Logikelement die atomare Basiskomponente eines FPGAs. Abbildung 16 gibt einen schematischen Überblick über die interne Struktur eines Logikelementes.

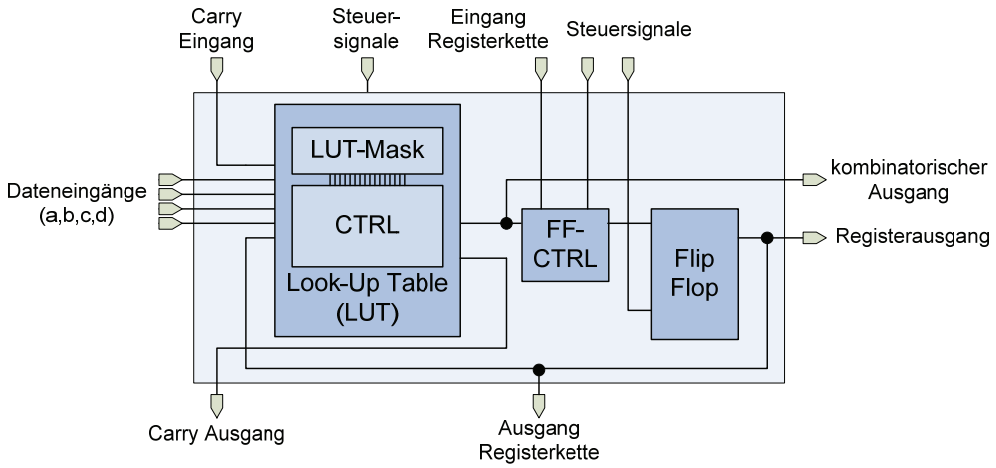


Abbildung 16: Struktur eines Logikelements (vereinfacht)

Kern eines Logikelements ist die Look-Up Table (LUT), eine SRAM-basierte Tabelle mit 16 Einträgen, wobei jeder Eintrag jeweils ein Bit speichern kann. Die 16 Einträge werden als Konfigurationsvektor der LUT (LUT-Mask) bezeichnet. Welcher der 16 vorhandenen Einträge ausgegeben wird, wird durch eine kaskadierte Multiplexer-Struktur (als CTRL-Block in Abbildung 16 dargestellt) gesteuert. Zur Steuerung der Multiplexer werden vier Eingangssignale verwendet. Durch diese Signale kann jedes einzelne Bit der LUT-Mask ausgelesen

werden. In Abbildung 17 wird deutlich, dass der Vektor *ABCD* = `0000´ das Bit 0 der LUT-Mask auswählt, wohingegen der Vektor *ABCD* = `1111´ Bit 15 auf den Ausgang schaltet.

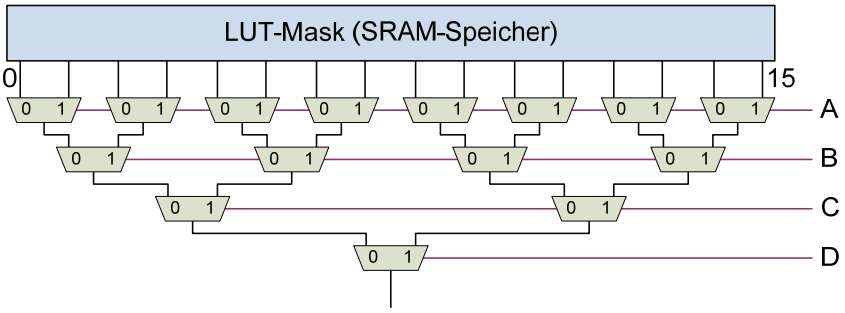


Abbildung 17: Schematische Darstellung der Werte-Adressierung im Konfigurationsvektor (LUT-Mask)

Für jede mögliche Kombination der vier Eingangssignale ABCD speichert die LUT einen Ausgabewert. Der Ausgabewert der LUT kann sowohl direkt aus dem Logikelement über einen kombinatorischen Ausgang herausgeführt, als auch in einem in das Logikelement integriertem FlipFlop gesichert und separat ausgegeben werden. Die LUT ist in der Lage, eine beliebige kombinatorische Verschaltung der vier Eingangssignale nachzubilden. Die Art der kombinatorischen Verschaltung spiegelt sich im Konfigurationsvektor wieder. Durch ihre 16 Speicherstellen kann die LUT 2^{16} verschiedene Funktionen darstellen. Die Erzeugung des der Funktion entsprechenden Konfigurationsvektors erfolgt meist durch ein Software-Tool. Abbildung 18 verdeutlicht dies an einem einfachen Beispiel. Ausgangspunkt ist in den meisten Fällen eine textuelle Beschreibung der Kombinatorik[1]. Für die Beschreibung werden sogenannte „hardware description languages“, also Hardware-Beschreibungssprachen verwendet, zu deren bekanntesten Vertretern VHDL und Verilog gehören. Die FPGA-Entwicklungsumgebung erstellt auf Basis dieser Beschreibung eine aus Grundelementen bestehende Schaltungsvariante und extrahiert den Konfigurationsvektor für die LUT. Dieser wird während des Konfigurationsprozesses des FPGAs an die entsprechende LUT übertragen. Die Entwicklungsumgebung übernimmt ebenfalls die Aufgaben der Aufteilung kombinatorischer Blöcke auf mehrere Logikelemente, die Auswahl geeigneter Logikelemente auf dem FPGA und die Verschaltung der Logikelemente untereinander. Für detailliertere Ausführungen zur Erstellung von Hardware-Komponenten für FPGAs mit Hilfe der Sprache VHDL sind [40] [52] geeignet.

[1] Moderne Tools der FPGA-Hersteller erlauben ebenfalls eine grafische Erstellung der Kombinatorik. Aufgrund des Aufwands für größere Hardware-Strukturen wird diese Möglichkeit allerdings selten verwendet.

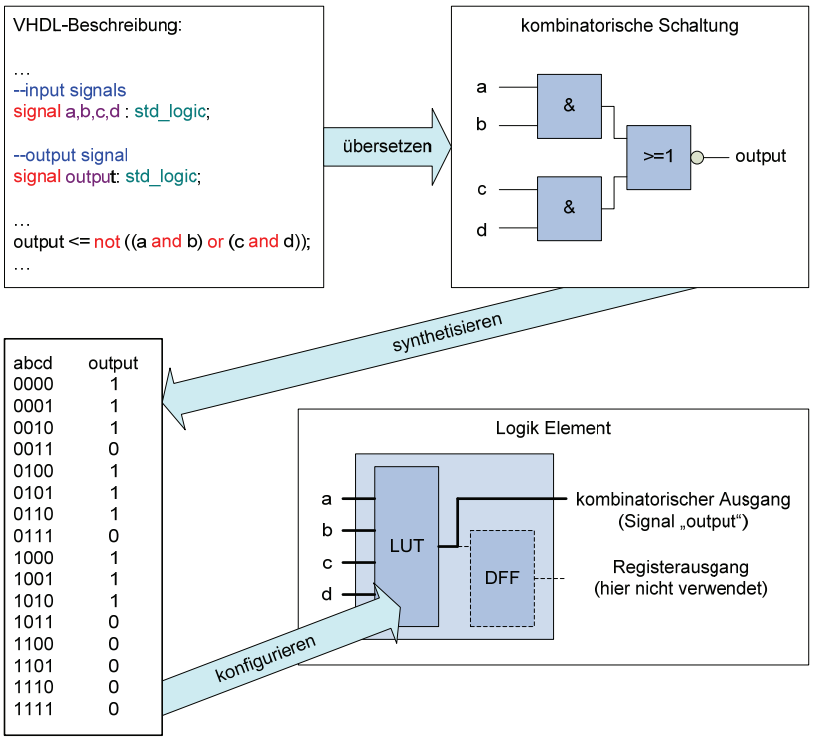


Abbildung 18: Konfiguration eines FPGAs ausgehend von einer VHDL-Beschreibung. Die Eingangssignale dienen zur Adressierung eines Tabelleneintrages. Die Ausgangswerte der Tabelle repräsentieren so die kombinatorische Schaltung, die durch VHDL-Beschreibung definiert wird. Die einzelnen Arbeitsschritte zur Konfiguration werden durch das verwendete Entwicklungstool automatisiert.

Wie bereits angegeben, verfügt die Multiplexer-Struktur über vier Steuereingänge. Jedes Logikelement besitzt ebenfalls vier allgemeine Dateneingänge (abcd in Abbildung 16), die direkt zur Adressierung innerhalb der LUT genutzt werden können. Da dies aber nicht immer sinnvoll ist, existiert eine Kontrollstruktur, die festlegt, welche Signale auf die Steuereingänge der Multiplexer geschaltet werden. Neben den allgemeinen Dateneingängen des Logikelements kommen dafür noch weitere Signal in Frage. Zur effizienten Realisierung von Hardware-Architekturen existiert so z.B. schon im Logikelement ein Signalpfad zur Rückführung des Ausgangssignals des FlipFlops auf einen Eingang in die Kontrollstruktur. Ferner wird auch ein besonderes Carry-Signal in die Kontrollstruktur geführt. Mit diesem Carry-Pfad können effizient Zähler, Akkumulatoren und vor allem Addierer realisiert werden. Für diesen Zweck kann die 4-Eingangs-LUT auch als zwei 3-Eingangs-LUTs oder gar vier 2-Eingangs-LUTs verwendet werden, wie die folgende Abbildung zeigt [4].

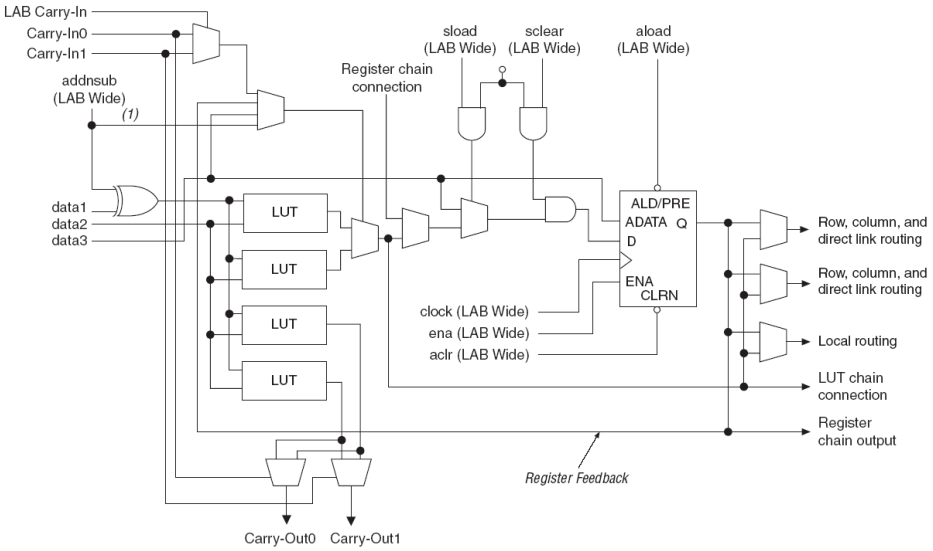


Abbildung 19: Segmentierung der LUT eines Logikelementes im „Dynamic Mode“

Am Beispiel eines Addierers wird die Wirkungswiese dieser Konfiguration deutlich. Ein Addierer besteht aus mehreren 1-Bit Addieren, deren Anzahl durch die Bit-Breite der Summanden festgelegt ist. Jeder 1-Bit Addierer verwendet 3 Eingangssignale (*Summand A, Summand B, Übertrag* aus vorheriger Stelle) um zwei Ausgangssignale (*Summe, Übertrag* in nächste Stelle) zu erzeugen. Dabei ist zu beachten, dass ggf. ein auftretender Übertrag in einer Stelle der Addition sich auf viele oder gar alle folgenden Stellen auswirkt. Das Übertragssignal durch alle Bit-Addierer stellt somit den zeitkritischen Pfad dar. Bei Betrieb der LUT als vier 2-Eingangs-LUTs, können mit aber schon mit den beiden Eingängen *Summand A* und *Summand B* vier Werte vorberechnet werden:

1. *Summe* und *Übertrag* für den Fall, dass der Übertrag aus der vorhergehenden Stelle logisch ‚1’ ist.
2. *Summe* und *Übertrag* für den Fall, dass der Übertrag aus der vorhergehenden Stelle logisch ‚0’ ist.

Von besonderer Bedeutung ist die Tatsache, dass eine Auswahl der richtigen Ergebnisse für *Summe* und *Übertrag* in Abhängigkeit des Übertrags aus der vorhergehenden Stelle getroffen wird. Technisch erfolgt dies über die Multiplexer, welche in das Steuerungsmodul integriert sind, an den entsprechenden Ausgängen der LUTs. In Abbildung 19 sind diese z.B. an den Ausgängen *Carry-Out0* und *Carry-Out1* sowie hinter den beiden oberen LUTs zu erkennen. Dadurch wird erreicht, dass bei stabilen Zuständen der Eingänge *Summand A* und *Summand B* eine besonders schnelle Änderung des Ergebnisses für *Summe* und *Übertrag* bei Zustandsänderung des Carry-Signals erfolgt.

4.3. Logik Array Blocks und die Verschaltung der Logikelemente

Um eine effiziente Verdrahtung der bereits besprochenen Logikelemente zu ermöglichen, werden diese in Gruppen, den logic array blocks (LABs) angeordnet. Eine Detailansicht eines solchen LABs ist in Abbildung 20 dargestellt. Beispielsweise bilden bei der Altera Cyclone Architektur [3] 10 Logikelemente einen Logic Array Block (LAB), welcher mit den globalen Routing-Kanälen verbunden ist. Zur schnellen Verbindung verschiedener Logikelemente innerhalb eines LABs verfügen diese über weitere, interne Routing Ressourcen, in Abbildung 20 als lokale Signalleitungen bezeichnet. Ferner werden unmittelbar benachbarte Logikelemente durch gesonderte Signalleitung miteinander verbunden, einem Carry-Signal[2] und einem register-chain Signal[3]. Diese Leitungsarten dienen dazu, kettenartige Strukturen mit besonders kurzen Signalwegen zu ermöglichen. Mit Hilfe dieser Signale können die im vorhergehenden Kapitel beschriebenen 1-Bit Addierer effizient verbunden werden. Das register-chain Signal ist für die Implementierung von Schieberegistern vorgesehen.

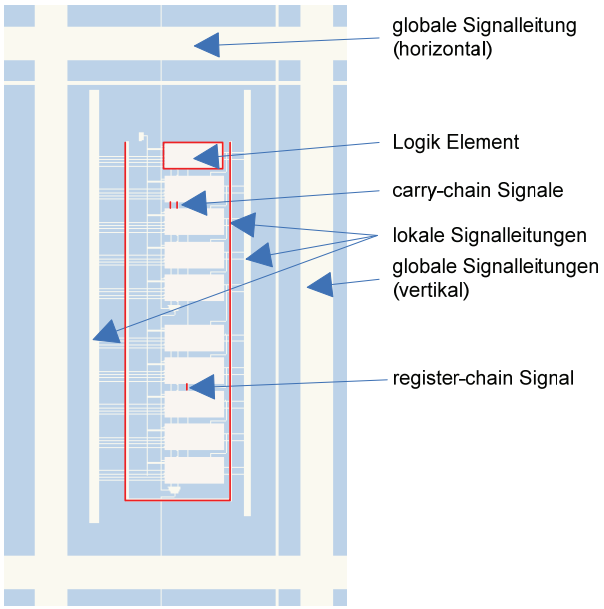


Abbildung 20: Darstellung eines Logic Array Blocks, bestehend aus 10 Logikelementen und Kennzeichnung der verfügbaren Signalleitungen (Altera Cyclonell Architektur), teilweise zur besseren Sichtbarkeit eingefärbt.

---

[2] Carry bezeichnet das Übertragssignal zwischen 1-Bit Addierern, wenngleich dieses Signal auch für andere Strukturen (z.B. Zähler) verwendet werden kann. Technisch werden zwei Signale verwendet, ein Carry und ein Not-Carry Signal

[3] register-chain bezeichnet eine kettenartige Verschaltung von FlipFlops (z.B. als Schieberegister)

Für eine genauere Ausführung zu den Verbindungsmöglichkeiten der Logik-Elemente wird auf die entsprechenden Dokumentationen der einzelnen FPGA-Familien hingewiesen [4] [5].

4.4. Zeitverhalten des Logikelements

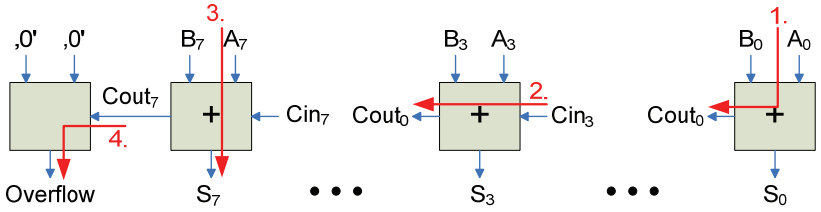
In der Einleitung zu diesem Kapitel wurde festgestellt, dass FPGAs zur prototypischen Realisierung anwendungsspezifischer Schaltkreise und für in geringen Stückzahlen gefertigte Schaltkreise eingesetzt werden. Die weiteren Ausführungen haben aber gezeigt, dass innerhalb der FPGAs prinzipiell gar keine Grundelemente elektronischer Schaltkreise, z.B NAND-, NOR-, XOR-Gatter, zur Verfügung stehen sondern vielmehr das Verhalten dieser Elemente mit speicherbasierten Tabellen nachgeahmt wird.

Die Frage, warum nicht gleich eine software-basierte Nachahmung der Schaltkreise auf einem leitungsstarken Prozessor emuliert wird, lässt sich mit dem Zeitverhalten der Logik-elemente beantworten. Eine software-basierte Beschreibung der Logik müsste zunächst eine Änderung der Eingangssignale erkennen und verarbeiten, um anschließend die entsprechende Software-Funktion auszuwerten und das Ergebnis zu sichern. Die Zeitdauer dafür ist nicht konstant, da unterschiedliche Software-Funktionen unterschiedlich lang sein können. Ferner kann auch die Ereignisdetektion unterschiedlich lange dauern, wenn z.B. gerade andere Programmabschnitte bearbeitet werden. Insgesamt sind eine Vielzahl Speicherzugriffe, Funktionsaufrufe und Funktionsbefehle zu verarbeiten, sodass schnell mehrere Hundert Takte für die vollständige Abarbeitung notwendig sind. Auch ist ein einfacher Prozessor nicht in der Lage, Änderungen an verschiedenen Eingangssignalen gleichzeitig zu verarbeiten.

Durch die Tatsache, dass im FPGA hingegen die LUT-Mask, und damit die Funktionalität des Logik-Elementes, während des Betriebes im SRAM-Speicher gesichert sind, sind die Eingänge der Multiplexer-Struktur (siehe Abbildung 17) in einem statischen Zustand. Eine Änderung der zu verarbeitenden Eingangssignale, welche als Steuersignale der Multiplexer agieren, führt lediglich zu einer Umkonfiguration der Multiplexer, um das entsprechende Bit der LUT-Mask auf den Ausgang zu transferieren. Ferner sind diese Eingangssignale direkt mit Eingabe-Ports des FPGAs oder den Ausgangssignalen anderer Logikelemente verbunden. Im FPGA führt somit eine Änderung am Eingang des Logik-Elements unmittelbar zu einem Umschalten der Multiplexer. Die dafür notwendige Zeit ist quasi konstant und liegt durch die technische Realisierung der Logikelemente im Bereich von unter 1 ns. Weiterhin arbeiten alle Logikelemente unabhängig voneinander, sodass gleichzeitige Änderungen verschiedener Eingangssignale des FPGAs auch gleichzeitig verarbeitet werden.

Die Geschwindigkeit, mit der sich eine Änderung der Eingangssignale auf den (kombinatorischen) Ausgang auswirkt ist von der Geschwindigkeitsklasse des eingesetzten

FPGAs abhängig. Die folgende Abbildung stellt Richtwerte für verschiedene FPGAs des Herstellers Altera dar. Weiterhin sind alle Zeitangaben Informationen, die aus dem Altera-Entwicklungstool QuartusII extrahiert wurden.



FPGA-Typ	Pfad 1. normaler Eingang, Carry-Ausgang	Pfad 2. Carry-Weiterleitung	Pfad 3. Normaler Eingang, kombinatorischer Ausgang des LEs	Pfad 4. Carry-Pfad auf kombinatorischen Ausgang
EP2C20 C6 (CycloneII)	ca. 410ps	71ps	ca. 430ps	410ps
EP2C20 C8 (CycloneII)	ca. 610ps	86ps	ca. 630ps	506ps
EP2S60 C3 (StratixII)	ca. 410ps	35ps	ca. 500ps	125ps

Abbildung 21: Zeitverhalten unterschiedlicher Signalpfade im Logikelement eines FPGAs

Der deutliche Geschwindigkeitsvorteil des Carry-Pfads wird durch die bereits vorberechneten Zwischenergebnisse (vgl. Abschnitt 4.2) und die dadurch reduzierte Anzahl der zu schaltenden Multiplexer (siehe Abbildung 17 und Abbildung 19) erreicht. Hinzu kommt, dass der direkte Signalpfad durch die Nähe zum nächsten Logikelement extrem kurz ist und somit einem schnellen Informationstransport zu Gute kommt. Der Carry-Pfad bietet diesen Geschwindigkeitsvorteil allerdings nur innerhalb eines LABs, müssen Daten in den nächsten Block übertragen werden, verschlechtert sich an dieser Stelle der Wert deutlich. Bei der in Abbildung 20 gezeigten Struktur eines logik array blocks eines CycloneII-FPGAs ist eine weitere Abweichung in der regelmäßigen Struktur in der Mitte des Blocks zu erkennen, an der sich die Verzögerungszeit im Carry-Pfad erhöht.

Alle in der obenstehenden Tabelle angegebenen Werte sind Richtwerte. Aufgrund von Fertigungseinflüssen können in verschiedenen Logikelementen andere Verzögerungszeiten entstehen. Weiterhin wurde hier davon ausgegangen, dass jeder der vier Standard-Eingänge eines Logik-Elements der gleichen Verzögerung unterliegt, bis seine Auswirkung am kombinatorischen Ausgang des Logikelementes sichtbar wird. Dennoch wird deutlich, dass FPGAs für die Realisierung einer einfachen Hardware-Funktion signifikante Geschwindigkeitsvorteile gegenüber der software-basierten Schaltkreis-Simulation auf einem Prozessor erzielen.

5. Stand der Forschung: Tapped Delay Lines

Verzögerungselemente (engl.: delay elements), wurden bereits in den 1940er Jahren in der Radartechnik zur Echounterdrückung eingesetzt. Später wurden sie zum Aufbau von Speicherstrukturen genutzt, indem sie ein Signal über einen bestimmten Zeitraum verzögerten und somit das Verhalten eines Speichers nachbildeten [19] [46]. Aufgrund des technischen Aufwandes konnte sich diese Anwendung allerdings nicht durchsetzen.

Seit den 1980er Jahren sind Verfahren entwickelt worden, die Verzögerungselemente zur Zeitmessung einsetzen. Nachdem bereits im Kapitel 3.4.4 das Grundkonzept der Tapped Delay Lines kurz vorgestellt wurde, dienen die Ausführungen der folgenden Abschnitte einer detaillierten Darstellung der aktuellen Entwicklungen auf dem Gebiet der Tapped Delay Lines im Rahmen von Laufzeitmesssystemen.

5.1. Grundstruktur der Tapped Delay Lines

Aus den Ausführungen in Abschnitt 3.4.4 wird deutlich, dass es sich bei den Tapped Delay Lines um eine sequentielle Struktur von Verzögerungselementen handelt. Im einfachsten Fall durchläuft das Signal S_1 die Kette aus Verzögerungselementen. In jedem Element wird es um die Zeit τ verzögert. Das Signal S_2, welches das Ende des Zeitintervalls kennzeichnet sichert ideal gleichzeitig den Status aller Verzögerungselemente, sodass das Zeitintervall über:

$$\Delta t = n\tau \tag{5.1}$$

bestimmbar ist, wobei n die Anzahl der durch S_1 durchlaufenen Kettenelemente angibt. Sowohl die Auflösung als auch die Genauigkeit haben im Idealfall den Wert τ. Der Messbereich M der Schaltung wird über die Anzahl N der verwendeten Delay Elemente bestimmt:

$$M = N\tau \tag{5.2}$$

Die technologisch einfachste Variante zur Implementierung einer Tapped Delay Line ist die Verwendung von pegelbasierten Latches in den einzelnen Verzögerungselementen. Somit wird gleichzeitig die Speicherung des Zustandswertes und die Verzögerung des Signals S_1 realisiert.

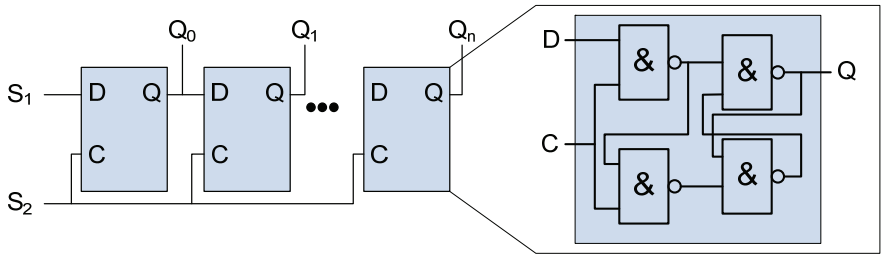


Abbildung 22: Eine Kette aus einfachen Latches bildet eine Tapped Delay Line

Abbildung 22 zeigt neben der Struktur der Tapped Delay Line ebenfalls den internen Aufbau eines Latches. So lange S_2 den über den Eingang *C* jedes Latch freischaltet, wird das eingehende Signal S_1 durch die Kette geleitet. Da von Element zu Element jeweils eine Umschaltung des Ausgangs *Q* notwendig ist, wird das Signal verzögert. Die Größe der Verzögerung ist durch die technologieabhängige Geschwindigkeit des Umschaltvorgangs definiert. Die Weiterleitung des Signals S_1 endet mit der Zustandsänderung von S_2, die das Ende des Zeitintervalls markiert und gleichzeitig alle Latches sperrt. Die Latch-basierte Struktur ermöglicht, bedingt durch den internen Aufbau der Latches, nur relativ große Verzögerungszeiten τ. Seferiadis et al. geben für eine aus Latches aufgebaute Delay Line in einem Spartan II FPGA des Herstellers XILINX eine Verzögerungszeit von 1 ns pro Latch an [57]. Altera gibt Werte für FPGA-basierte Latches (im LUT integriert) mit ca. 250 ps an. In dieser Region liegen auch Angaben des Herstellers XILINX. Die Verzögerungszeit von CMOS-Latches hängt von Aufbau und Technologie ab, liegt aber in Größenordnung von ca. 100 – 200ps.

Aus diesem Grund gibt es in der Literatur eine Vielzahl von Implementationen, die auf eine separate Ausführung des Delays, losgelöst von der Statusspeicherung setzen. Abbildung 23 stellt noch einmal die Grundstruktur einer Tapped Delay Line mit flankengesteuerten D-FlipFlops und separaten Verzögerungselementen τ dar.

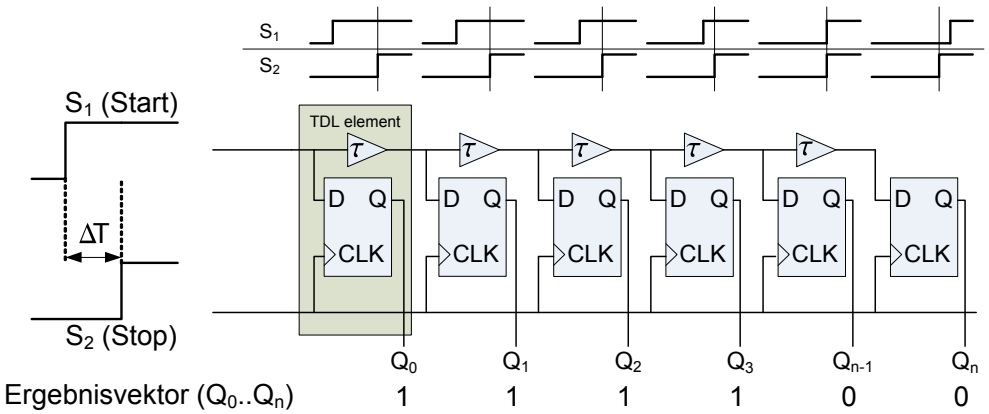


Abbildung 23: Grundstruktur der Tapped Delay Lines mit expliziten Verzögerungselementen

Für die Verzögerung wird in einem CMOS-Design am einfachsten eine gerade Anzahl Inverter verwendet. Aber auch andere Strukturen, z.B. NAND-Gatter eigenen sich zur Signalverzögerung [71].

Um tatsächlich eine möglichst gleichzeitige Sicherung der Statuswerte zu erreichen wird das Signal S_2 nicht linear an die Kettenelemente herangeführt. Vielmehr wird auf Baumstrukturen zurückgegriffen, die z.B. in FPGAs bereits für die Verteilung des Taktsignals auf dem Chip vorhanden sind [58].

5.2. Differentielle Tapped Delay Lines

Bisher wurde bei den gezeigten Delay Lines jeweils ein Signal verzögert, wohingegen das zweite Signal für die ideal gleichzeitige Sicherung aller Kettenelemente genutzt wurde. Die Auflösung der Zeitmessung wird, wie bereits mehrfach erwähnt, über die Verzögerungszeit τ der einzelnen Elemente bestimmt. Eine signifikante Erhöhung der Auflösung *R* von Tapped Delay Lines wird durch die gleichzeitige Verzögerung beider Signale mit unterschiedlichen Verzögerungszeiten τ_a und τ_b erreicht. Es gilt für die Auflösung *R*:

$$R = \left| \tau_a - \tau_b \right| \tag{5.3}$$

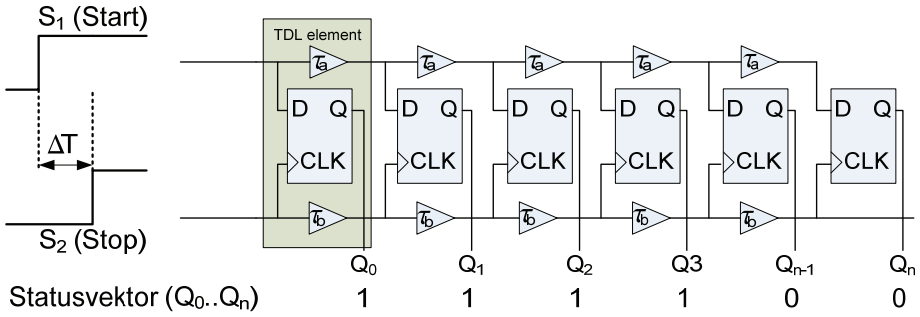


Abbildung 24: Differentielle Tapped Delay Line mit zwei Verzögerungselementen pro Stufe

In der Literatur wird diese Struktur als differentielle Tapped Delay Line oder auch Vernier Delay Line bezeichnet [15] [2] [57]. Die Auflösung ist über die Differenz beider Verzögerungswerte τ_a und τ_b in jedem Element definiert. Als Voraussetzung dafür muss die Technologie-Plattform allerdings in der Lage sein, mindestens zwei stabile, unterschiedliche Delays der gleichen Größenordnung zu erzeugen, worauf noch einmal in den Ausführungen zu den Technologie-Plattformen maskenprogrammierbarer ASICs und FPGAs eingegangen wird.

Aufgrund der Tatsache, dass die resultierende Verzögerung eines Elementes durch den Betrag der Differenz der Einzelverzögerungen in beiden Signalpfaden definiert ist, ist der Messbereich bei den differentiellen Tapped Delay Lines deutlich kleiner als bei der oben dargestellten Variante, da gilt:

$$M = N \cdot \left| \tau_a - \tau_b \right| \tag{5.4}$$

Erfordert die Anwendung einen großen Messbereich *M*, muss die Tapped Delay Line eine Vielzahl an Einzelelementen enthalten. Um dies zu umgehen werden die Tapped Delay Lines häufig mit einem zählerbasierten Verfahren gekoppelt. Diese Methode ist in der Literatur unter anderem von Nutt bereits 1968 vorgestellt wurden [51]. Abbildung 25 zeigt noch einmal das bereits in Abschnitt 3.3 dargestellte Prinzip der zählerbasierten Zeitmessung.

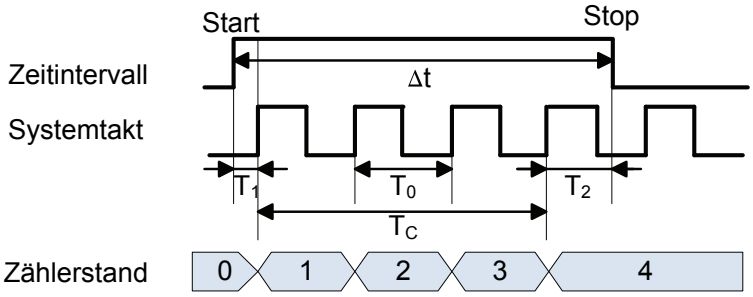


Abbildung 25: schematische Darstellung der Zeitmessung mittels digitaler Zähler

Es wurde bereits diskutiert, dass die dargestellten Zeiten T_1 und T_2 nicht durch den Zähler berücksichtigt werden können und somit das Messergebnis verfälscht ist. Um diesen Mangel zu beheben, werden Tapped Delay Lines in das System integriert. Sie messen hochpräzise die beiden Zeiten T_1 und T_2. Somit kann aus den mittels Tapped Delay Lines bestimmten Zeiten T_1 und T_2 sowie der aus dem Zählerstand bestimmbaren Zeitdauer T_c die Gesamtdauer des Zeitintervalls Δt berechnet werden.

5.3. Aktuelle Tapped Delay Line Implementationen

Aktuelle Tapped Delay Line Implementationen unterscheiden sich hauptsächlich in der Realisierung der Verzögerungszeiten τ und der verwendeten Technologie-Plattform. Die beiden wesentlichen Plattformen zur Realisierung von Tapped Delay Lines sind die maskenprogrammierbaren anwendungsspezifischen Schaltkreise und die anwender-programmierbaren Schaltkreise (z.B. FPGAs). Zwar zählen nach [20] beide Varianten zur Gruppe der Anwendungsspezifischen Schaltkreise (application specific integrated circuit – ASIC), sie unterscheiden sich aber signifikant in ihrer Handhabung.

Die maskenprogrammierbaren anwendungsspezifischen Schaltkreise (engl. application specific custom product – ASCP) sind Schaltkreise, für die der Entwickler die interne Schaltung des Mikrochips auf Basis von vorgegebenen Standardzellen entwirft. Im Vergleich zu Standardschaltkreisen (z.B. Speicher und Prozessoren) muss er sich dabei allerdings an Vorgaben betreffend der Geometrie, Platzierung und Verdrahtung der Standardzellen halten,

die durch den Hersteller des Chips definiert werden. Auch kann der Entwickler in den meisten Fällen keine neuen Zellen aus einzelnen Transistoren aufbauen.

ASCPs unterschieden sich von den FPGAs dahingehend, dass ASCPs explizit für einen bestimmten Zweck, eine bestimmte Aufgabe, entworfen werden, wohingegen FPGAs bereits fertige Mikrochips sind, die eine beliebige Hardware nachbilden können, wie ausführlich in Abschnitt 4 dargestellt. Die Bezeichnung „anwenderprogrammierbarer Schaltkreis“ (user-programmable integrated circuit – UPIC) ist dabei zwar allgemein anerkannt, prinzipiell aber falsch, da keine Programmierung im eigentlichen Sinne erfolgt. Der Schaltkreis führt kein Programm aus, wie es von Prozessoren und Mikrocontrollern bekannt ist. Vielmehr werden die Logikzellen des FPGAs so konfiguriert, dass sie eine bestimmte Hardware nachbilden und so eine durch den Anwender vorgegebene Funktionalität bieten.

Für weiterführende Informationen zum Thema des Schaltkreisentwurfs und der beschriebenen Klassifikation sei der geneigte Leser auf [20] verwiesen. Dieser Abschnitt bietet eine Aufstellung über aktuelle Tapped Delay Line Implementationen beider Technologien, die jeweils eingesetzten Varianten und erzielten Genauigkeiten.

5.3.1. ASCP-basierte Tapped Delay Lines

Aufgrund der Tatsache, dass ein auf Standardzellen basierendes Schaltungs-Design vielfältige Möglichkeiten zur Gestaltung der integrierten Strukturen bietet, verwenden ASCP-basierte Tapped Delay Lines vorrangig differentielle Tapped Delay Lines. Die unterschiedlichen Verzögerungszeiten τ_a und τ_b werden vielfach durch Einstellung der Lastkapazitäten der in den verwendeten Grundgattern integrierten Transistoren erzielt. Die Lastkapazität eines Transistors wird durch seine geometrischen Parameter definiert. Je größer die während des Schaltvorgangs aufzuladende, bzw. zu entladene Kapazität ist, desto größer ist die Verzögerungszeit zwischen dem Eingang und dem Ausgang des aus den Transistoren aufgebauten Gatters. Abas et al. haben 2004 eine auf einem 0,18 µm CMOS-Prozess basierende Messschaltung vorgestellt, die mit diesem Ansatz eine Auflösung von unter 10 ps erreicht [2]. Als Verzögerungselemente wurden in jedem Signalpfad pro Element zwei Inverter verwendet, wobei sich die Inverter in der Größe um 25% unterschieden. Diesem Verfahren sehr ähnlich ist der 2005 von Jannson et al. vorgestellte Ansatz, in dem unterschiedliche Delays durch eine Vervielfachung einer definierten Lastkapazität erzeugt werden. Im Ergebnis erreicht die in einem 0,35 µm CMOS Prozess gefertigte, 2,5 mm x 3,0 mm große, Messschaltung eine zeitliche Auflösung für eine Einzelmessung von 13 ps [22]. Detaillierte Ausführungen zur Anpassung von Lastkapazitäten sind ebenfalls in der von Mäntyniemi 2004 publizierten Arbeit enthalten, wo

eine in einem 0,6 µm CMOS-Prozess gefertigte Delay Line eine zeitliche Auflösung von 30 ps erzielte [43].

Andere Verfahren zur Einstellung von unterschiedlichen Delays sind unter anderem die Verwendung von integrierten Schmidt-Triggern mit einstellbaren Schwellwerten [47], der Einsatz spezieller, passiver RC-Komponenten mit einstellbarer Lastkapazität [48] und die Steuerung des Stromflusses in Invertern via einzeln zuschaltbarer Transistoren im pull-up und pull-down Netzwerk von CMOS-Strukturen [16].

Der Vollständigkeit halber sei erwähnt, dass einige dieser Verfahren zur Delay-Erzeugung ursprünglich nicht aus dem Bereich der Zeitmessung stammen, sondern in Strukturen zur Generierung stabiler Takt-Signale (so genannte delay locked loops – DLLs) eingesetzt werden. Dennoch sind alle genannten Methoden potentiell zur Zeitmessung geeignet.

Diese Form der Herstellung kunden- bzw. anwendungsspezifischer, integrierter Schaltkreise ist allerdings ein aufwendiger und kostenintensiver Prozess. Der Aufwand ist zum einen dadurch begründet, dass der Entwurf eines anwendungsspezifischen Schaltkreises nicht nur die Entwicklung der Schaltungsstruktur sondern auch deren Platzierung auf dem Chip, die Verdrahtung mit den Ein- und Ausgangspins, die Integration eines Taktsystems und Anbindung der Spannungsversorgung notwendig ist. Hinzu kommt, dass vor der Erstellung des ersten Exemplars alle eventuellen Fehlerquellen ausgeschlossen werden müssen und die funktionale Richtigkeit des Gesamtsystems nachgewiesen werden muss. Ist dies erfolgt, werden die für die Fertigung notwendigen Photomasken erstellt. Zwei Aspekte sind dabei zu berücksichtigen: Zum einen ist diese Maskenerstellung der kostenintensivste Teil bei der Entwicklung eines integrierten Schaltkreises. Zum anderen handelt es sich dabei um einen einmaligen Vorgang. Nachträgliche Änderungen am Design erfordern eine vollständige Neuerstellung dieser Masken. Die Kosten für die anschließende Massenproduktion des Schaltkreises sind im Vergleich dazu verschwindend gering. Der hohe Aufwand für diesen Entwicklungsprozess hat in den vergangenen Jahren dazu geführt, dass anwender-programmierbare Schaltkreise im Bereich der Hardware-Entwicklung, nicht nur auf dem Gebiet der Zeitmesssysteme, immer mehr an Bedeutung gewannen.

5.3.2. FPGA-basierte Tapped Delay Lines

FPGA-basierte Tapped Delay Lines nutzen die interne FPGA Struktur zur Realisierung der Grundelemente, ihrer internen Verzögerungszeiten τ und zur ketteförmigen Hintereinander-schaltung der Elemente. Ferner werden vielfach Hardware-Strukturen zur Weiterverarbeitung des Messergebnisses ebenfalls in das FPGA integriert, um eine ganzheitliche Lösung für ein Zeitmesssystem zu ermöglichen.

Die ersten Varianten FPGA-basierter Tapped Delay Lines wurden bereits in den 1990er Jahren vorgestellt. Unter anderem haben Kalisz et al. 1997 eine Lösung vorgestellt, in der ein 100 MHz Zähler in Verbindung mit zwei Tapped Delay Lines eine zeitliche Auflösung von 200 ps erreicht [33]. Das System wurde in einem FPGA der QuickLogic-Reihe implementiert. Die Tapped Delay Lines wurden mit jeweils 63 Delay-Elementen entworfen, von denen 50 Elemente für die Zeitmessung relevant sind. Zum einen wird so sichergestellt, dass eine komplette Taktperiode (100 MHz = 10 ns) abgedeckt werden kann. Zum anderen soll durch die Überlänge (63·200 ps = 12,6 ns) dem Einfluss von Temperatur- und Spannungsänderungen entgegengewirkt werden, indem die im jeweiligen Moment günstigen 50 Element aus allen 63 Elementen ausgewählt werden können. Das Delay τ wurde durch die FPGA-internen Look-Up Tables erzeugt. Das Start-Signal adressiert im ersten Verzögerungselement eine Speicherstelle, die einen Umschaltvorgang am Ausgang des Elements zur Folge hat. Der kombinatorische Ausgang wird als Eingang des nächsten Kettenelementes genutzt. Die Zeit von der Adressierung der Speicherstelle bis zum stabilen Zustand am kombinatorischen Ausgang des Logikelements stellt die Verzögerungszeit τ dar, in diesem Fall 200 ps. Durch Verwenden einer zweidimensionalen Tapped Delay Line haben Andaloussi et al. Im Jahre 2002 eine Verzögerungszeit von 150 ps erreicht [7]. Durch weitere Schaltungsmodifikationen haben im Jahre 2005 Xie et al. [69] sowie Szymanowski und Kalisz [62] Auflösungen von 140 ps bzw. 200 ps erreicht. Schon wenige Jahre später konnte dieses Ergebnis auf $\tau = 100\ ps$ verbessert werden. Kalizs et al. stellten eine verbesserte Variante vor, die auf differentiellen Tapped Delay Lines basierte. Als Verzögerungselemente wurden hier jeweils zwei nicht-invertierende Buffer verwendet. Die 2006 von Seferiadis et al. vorgestellte Implementation einer Delay Line verwendet klassische, durch Look-Up Tables implementierte Latches als Verzögerungselemente. Durch eine differentielle Ausführung konnten trotz der komplexeren Struktur der Verzögerungselemente (Latches im Vergleich zu nicht-invertierenden Buffern) ebenfalls 100 ps als zeitliche Auflösung erreicht werden. Erfolgreich eingesetzt wurde dieses System in einem orbitalen Partikeldetektor.

Alle bisher vorgestellten Verfahren haben gemein, dass sie die Logikelemente eines FPGAs zur Erzeugung der Verzögerungszeiten nutzen. Genauer nutzen sie den im Abschnitt 4.4 beschriebenen Effekt der Signalverzögerung vom Standardeingang eines Logikelements zu dessen kombinatorischen Ausgang.

Jian et al. Stellten 2005 eine erste FPGA-basierte Implementation einer Tapped Delay Line vor, welche die FPGA-internen Carry-Pfade (siehe Abschnitt 4.4) zur Generierung der Verzögerungszeiten τ ausnutzte. Durch diesen schaltungstechnischen Trick konnten Verzögerungszeiten von 60 ps erzielt werden. Bei dieser Schaltungsvariante ist es nicht mehr möglich, differentielle Delay Lines zu verwenden. Dies ist dadurch begründet, dass jedes

Logikelemente lediglich ein Carry-Pfad enthält und somit die geringe Verzögerungszeit von lediglich 60 ps auch nur für ein Signal appliziert werden kann. Eine logische Verknüpfung von zwei, in verschiedenen Logikelementen integrierten, Verzögerungselementen würde die Verwendung eines dritten Logikelementes erzwingen, welches durch sein eigenes Zeitverhalten die hohe Auflösung zu Nichte machen würde. Daher bleibt hier nur die Variante der einfachen Verzögerung mit gleichzeitiger Statussicherung aller Elemente. Ebenfalls in der Größenordnung von 60 ps liegen die von Zhang et al. 2006 und von Cicalese et al. 2007 publizierten Ergebnisse, die ebenfalls auf der Ausnutzung der FPGA-internen Carry-Pfade beruhen [70] [12].

5.4. Schlussfolgerungen aus dem Stand der Technik

Tapped Delay Lines basieren auf der Zerlegung eines zu messenden Zeitintervalls in viele, kleine Segmente, was einer Zeitquantisierung entspricht. Die Länge eines Segments und damit die Auflösung des Messsystems wird durch die Verzögerungszeit τ eines Delay Elementes bestimmt. Durch die kettenförmige Struktur der Delay Lines wird diese Signalverzögerung wiederholt auf eines der beiden Signale oder beide Signale angewendet. Im Ergebnis gibt der Status der Delay Line Aufschluss darüber, wie oft das Signal während der Messung verzögert wurde, bzw. wie viele Kettenelemente während des Zeitintervalls durchlaufen wurden. Aus diesem Grund werden in aktuellen Delay Line Implementationen immer identische Verzögerungszeiten in jedem Kettenelement angestrebt, um eine einfache Auswertung des Ergebnisses zu erlauben. Das Hauptaugenmerk der Forschungsarbeiten auf diesem Gebiet lag auf der stetigen Verringerung der Verzögerungszeiten τ. Insbesondere ist auf dem Gebiet der FPGAs der Übergang zu der Carry-Pfad-basierten Signalverzögerungen zu nennen, der die Auflösungen um den Faktor zwei verbesserte. Allerdings ist damit auch eine technische Grenze erreicht, die bisher nicht überschritten werden konnte. Zukünftige, auf diesem Verfahren basierende Implementationen werden nur noch geringe Verbesserungen der zeitlichen Auflösung im Maße der technologischen Fortschritte der FPGAs erreichen. Die bisher veröffentlichten Ergebnisse basieren auf Xilinx‘ Virtex-4 und Virtex-5 FPGAs. Die aktuellen StratixII FPGAs von Altera stellen Carry-Pfade in einem Delay von 35 ps in Aussicht. Auf maskenprogrammierbaren ASICs basierende Lösungen haben zwar eine um eine Größenordnung bessere zeitliche Auflösung, dennoch verhinderten bisher die hohen Entwicklungskosten einen Einsatz dieser Art der Zeitmessung in einem breiten Anwendungsspektrum.

6. BOUNCE: Konzept

6.1. Zielstellung dieser Arbeit, Motivation

Die bisherigen Ausführungen zu den Gebieten Lokalisierung, Laufzeitmessung, Tapped Delay Lines und FPGAs dienten dazu, einen Kontext für die im Rahmen dieser Arbeit erfolgten Forschungsarbeiten zu liefern. Im Mittelpunkt steht die Frage nach einer signifikanten Verbesserung der Laufzeitmessung mit einfachsten, digitalen Schaltkreisen (FPGAs) um der Lokalisierung und hier insbesondere der Indoor-Lokalisierung neue Möglichkeiten zu eröffnen.

Die Lokalisierung, also dass Bestimmen der Positionen eines Objektes, ist bereits bei Lebensformen unterschiedlichster Art zu finden. Jeder Mensch kann zum Beispiel eine Schallquelle „orten“. Werden wir in einer Menschenmenge unvermittelt durch eine fremde Person angesprochen, wenden wir uns ihr automatisch zu. Unser Gehirn gibt dabei auf Basis der empfangenen Schallsignale eine Richtung für das Drehen des Kopfes vor. In der Natur ist die Schleiereule ein Tier mit einem ausgeprägten Lokalisierungsvermögen. Auf Basis der Zeitpunkte, an denen ein Schallsignal einer einzelnen Quelle beide Ohren der Schleiereule erreicht, kann sie ihre Beute mit Richtungsauflösung von 2° orten.

Abbildung 26: Schleiereule (tyto alba)

Das verblüffende an diesem Ergebnis ist die Tatsache, dass die Neuronen, die für die Ortung im Gehirn der Eule verwendet werden, eine Schaltzeit von 0,1 ms (Membrankonstantzeit) aufweisen. Geht man von einer Schallgeschwindigkeit von 330 m/s aus, erreichen seriell verschaltete Neuronen nur eine räumliche Genauigkeit von 3,3 cm. Da die Ohren der Eule allerdings nur 5cm auseinander liegen, entspräche dies einer Ortungsgenauigkeit von 33°.

Ein Modell, welches die Erklärung für die hohe Genauigkeit von 2° trotz der langsamen Neuronen erklären kann, wurde bereits 1948 durch Jeffress vorgestellt [23].

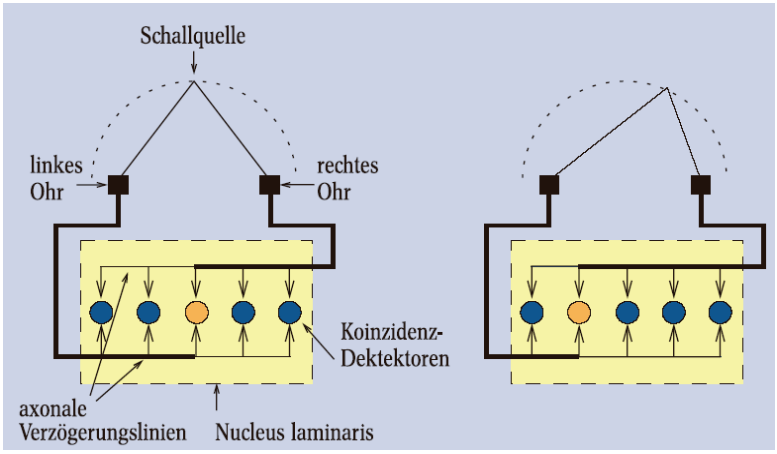


Abbildung 27: Vereinfachtes Jeffress-Modell zur Schallquellenortung der Schleiereule

Kern ist ein System aus axonalen Verzögerungsleitungen, die mit Hilfe von parallel verschalteten Konizidenzdetektoren (Neuronen) aus dem zeitlichen Unterschied des Auftretens des akustischen Signals eine Ortsinformation erstellen. Das Neuron, welches gleichzeitig durch die Information von beiden Ohren erreicht wird (im Bild orange dargestellt), erzeugt die stärkste Reaktion, während andere Neuronen (blau) kaum oder gar nicht reagieren. Die Richtigkeit des Modells wurde durch Konishi et al. 1993 nachgewiesen [39]. Zu ähnlichen Ergebnissen kommt auch Carr, ebenfalls 1993 [11]. Auch die Arbeiten von Kempter et al. aus dem Jahre 1996 stützen dieses Model [37]. Kempter et al. veröffentlichten 2001 ferner ein Modell, welches die Entstehung dieser neuronalen Strukturen im Gehirn der Eule plausibel erklärt [38]. Neben der eigentlichen Struktur ist heute bekannt, dass während der Wachstumsphase eine selbstorganisierte Optimierung der Struktur erfolgt, um mit möglichst wenig Neuronen eine hoch-präzise Ortung zu ermöglichen.

Diese Arbeit behandelt die Übertragung der aus der Biologie bekannten Struktur in ein technisches System. Hierbei konzentriert sich die Arbeit zunächst auf die reine Laufzeitmessung. Die Verwendung für Lokalisierungsaufgaben ist naheliegend, weshalb die erzielten Resultate am Beispiel der Indoor-Lokalisierung bewertet werden. Kern dieser Arbeit ist ein Konzept, welches auf den Namen BOUNCE – „Bunch Of UNconnected Chain Elements“ getauft wurde. Die grundlegende Eigenschaft des Systems besteht darin, dass die von den TDLs bekannte Kettenstruktur aus Verzögerungselementen vermieden wird (vgl. Abschnitt 5). Stattdessen wird eine, dem Ortungssystem der Eule nachempfundene, parallele Struktur ohne explizite Verzögerungselemente entwickelt. Hier steht die Hypothese, dass die in Kapitel 4 beschriebene FPGA-Technologie zum einen die Möglichkeit bietet, die Struktur aus parallelen Detektoren nachzubilden. Zum anderen wird davon ausgegangen, dass die FPGA-internen Signalleitungen in der Lage sind, die axonalen Verzögerungs-

leitungen zu bilden. Der aus technischer Sicht entscheidende Aspekt ist die Behauptung, dass die Nutzung der Signalleitungen zur Verzögerung der Messsignale eine wesentlich höhere zeitliche Auflösung erreicht, als es die FPGA-basierten Tapped Delay Lines bisher erlauben. Konkret wird eine Verbesserung der zeitlichen Auflösung FPGA-basierter Laufzeitmesssysteme um eine Größenordnung erwartet. Gleichzeitig soll aber der Aufwand für die praktische Realisierung des BOUNCE-Konzeptes möglichst gering ausfallen. All diese Aspekte und ein qualitativer Vergleich mit den bisher vorgestellten Laufzeitmessverfahren befindet sich in Abbildung 28.

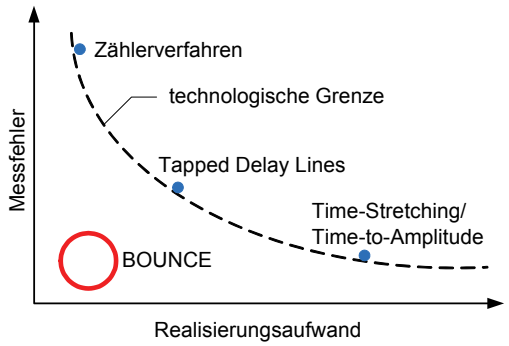


Abbildung 28: Qualitative Darstellung aktueller Laufzeitmesssysteme in Bezug auf die Größen „Realisierungsaufwand“ und „Messfehler“

Die folgenden Abschnitte dieses Kapitels dienen dazu, zunächst eine technische Beschreibung des Konzepts zu bieten, bevor die praktisch erreichten Ergebnisse unter Zuhilfenahme technischer Prototypen diskutiert werden. Konkret verfolgt diese Arbeit nachstehende Ziele:

1. Untersuchung der grundsätzlichen Eignung, Einsatzmöglichkeiten und Grenzen der Verwendung FPGA-interner Signalleitungen zur hochpräzisen Messung von Signalverzögerungszeiten im einstelligen Pikosekundenbereich,
2. Überführung des vom biologischen Vorbild bekannten Laufzeitmessstruktur in eine technisch umsetzbare Form,
3. Erreichen einer zeitlichen Auflösung von unter 10ps mittels einer FPGA-basierten Realisierung der BOUNCE-Architektur.

Diese Arbeit stellt also in Aussicht, mit *einfachster Technik hochpräzise zeitliche Auflösungen* in Laufzeitmesssystemen zu erreichen, die auch für andere Forschungsgebiete von großem Interesse sind. Hier sind explizit *Ubiquitous Computing* und *Ambient Intelligence* zu nennen, die für ihre internen Selbstorganisationsprozesse Positionsangaben verschiedenster mobiler Objekte mit einer Genauigkeit im Zentimeterbereich benötigen. Im Folgenden werden dieses Konzept und die daraus resultierenden Ergebnisse vorgestellt.

6.2. Entwicklung des technischen Konzepts

Alle bisher vorgestellten Laufzeitmesssysteme benutzen explizit angelegte Schaltungen, bzw. Schaltungselemente, um eine Zeitdiskriminierung zu erzielen. Das in dieser Arbeit vorgestellte Konzept verzichtet auf diese Strukturen. Vielmehr wird die zur Messung eines Zeitintervalls benötigte Zeitdiskriminierung durch interne Signalleitungen eines Mikrochips erzeugt. Eine Signalleitung endlicher Länge wird durch einen Leitungswiderstand und eine Leitungskapazität charakterisiert. Entsprechend des Betrages beider Größen entsteht so ein Tiefpass erster Ordnung mit einer daraus resultierenden Signalverzögerung zwischen Eingang und Ausgang der Leitung. Die angesprochene Signalverzögerung ist dabei proportional zur Länge der Signalleitung. Werden also zwei Signalquellen, die Begin und Ende eines Zeitintervalls markieren, an einen beliebigen Ort auf einem Mikrochip geführt, entstehen allein durch die Signalleitungen zwischen dem Ort der Signaleinspeisung und des Ziels zwei charakteristische Verzögerungen T_1 für S_1 und T_2 für S_2 die sich durch die verwendete elektrische Leitung ergeben. Abbildung 29 stellt diesen Effekt dar.

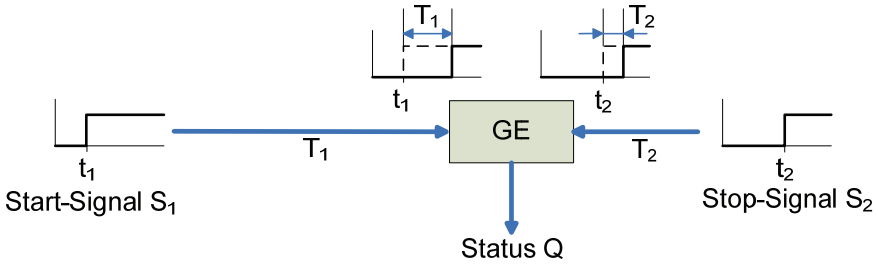


Abbildung 29: Darstellung eines Grundelements und die zeitlichen Zusammenhänge, die die Basis für das Konzept der BOUNCE-Architektur bilden

Für eine Laufzeitmessung wird dieser Effekt dahingehend ausgenutzt, dass ein am Endpunkt beider Signalleitungen befindliches Grundelement (GE) auf einfache Art und Weise eine Aussage zum zeitlichen Zusammenhang zwischen den Ereigniszeitpunkten t_1 und t_2 unter Berücksichtigung der Signalverzögerungen T_1 und T_2 trifft. Der Status-Ausgang des Grundelementes ist in der Lage, zwei Werte darzustellen. In Abhängigkeit des Eintreffens beider Signale gilt:

$$\begin{aligned} t_1 + T_1 > t_2 + T_2 &\longrightarrow Status = 1 \\ t_1 + T_1 < t_2 + T_2 &\longrightarrow Status = 0 \end{aligned} \tag{6.1}$$

Da die beiden Zeiten T_1 und T_2 hierbei systembedingt sind und als konstant betrachtet werden, wird eine für das Grundelement charakteristische Schaltgrenze oder auch charakteristische Differenz ε definiert:

$$\varepsilon = T_1 - T_2 \tag{6.2}$$

Somit ergibt sich aus den Gleichungen (6.1) und (6.2) mit der Vereinfachung $\Delta t = t_2 - t_1$:

$$\begin{aligned} \Delta t > \varepsilon &\longrightarrow Status = 0 \\ \Delta t < \varepsilon &\longrightarrow Status = 1 \end{aligned} \tag{6.3}$$

Dadurch wird die Zeitachse in zwei Bereiche geteilt, was für eine präzise Zeitmessung unzureichend ist. Werden allerdings auf einem Mikrochip an verschiedenen Orten Grundelemente platziert, entstehen aufgrund der unterschiedlichen Zuleitungswege verschiedene Schaltgrenzen ε_i. Die Orte der Grundelemente werden so bestimmt, dass jedes Element eine systemweit einzigartige Schaltgrenze besitzt. Abbildung 30 verdeutlicht dies am Beispiel von fünf parallelen Grundelementen. Die Zuleitungsverzögerungen T_{1i} und T_{2i} sind durch die unterschiedlich langen Signalpfeile, die zu jedem Grundelement führen, symbolisiert. Je länger die Zuleitung ist, desto größer ist die Verschiebung des Signals auf der Zeitachse, was in den farbigen Kästchen in der Nähe der Grundelemente illustriert ist. Die Farbe des Kästchen selbst gibt an, welches Signal als erstes auftritt (grün) und welches als zweites (rot).

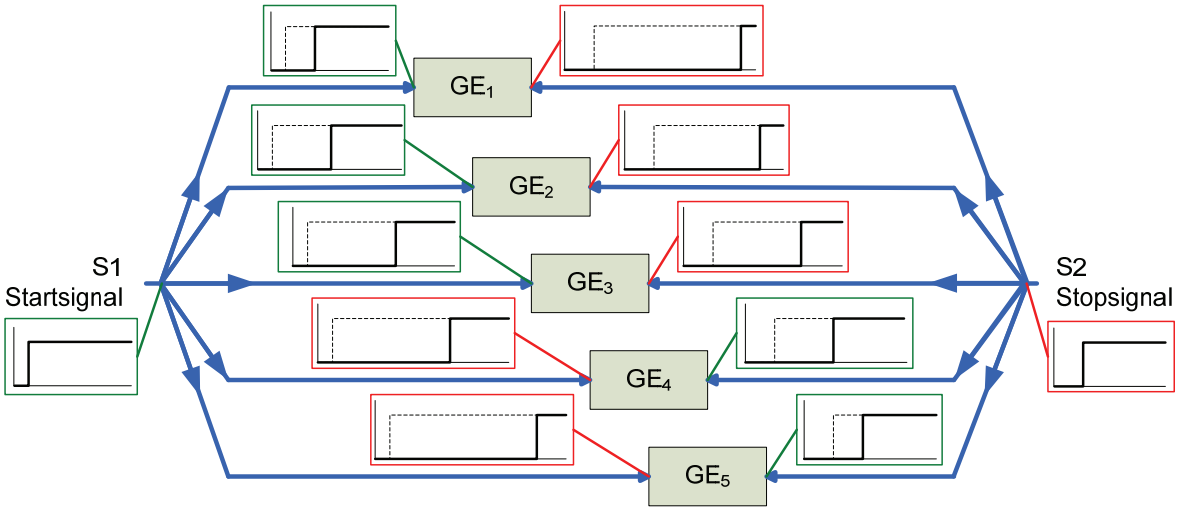


Abbildung 30: parallele Grundelemente mit unterschiedlich langen Signalpfaden bilden den Kern der BOUNCE-Architektur

Durch diese Struktur entstehen bereits fünf Schaltgrenzen ε_1 bis ε_5. Diese ergeben sich jeweils aus der Differenz der Verzögerung auf den Leitungen zum Grundelement, wie bereits in Gleichung (6.2) formuliert und in Abbildung 29 dargestellt. In diesem Beispiel soll gelten:

$$\varepsilon_i < \varepsilon_{i+1} \tag{6.4}$$

Damit steht bereits ein rudimentäres Zeitmesssystem zur Verfügung, welches einen Messbereich *M* besitzt, der durch die kleinste und größte Schaltgrenze, im dargestellten Fall ε_1 und ε_5, definiert ist. Allgemein gilt für *M*:

$$M : \varepsilon_{\min} < \Delta t < \varepsilon_{\max} \tag{6.5}$$

Die absolute Größe des Messbereiches ist von den geometrischen Abmessungen des Mikrochips abhängig, weil dadurch die maximalen Signalpfadlängen bestimmt werden und diese wiederum die charakteristischen Schaltgrenzen ε_i definieren. Werden die Grundelemente so auf dem Mikrochip platziert, dass eine homogene Verteilung der Schaltgrenzen ε_i erreicht wird, kann die Auflösung *R* definiert werden als:

$$R = \frac{M}{n-1} \tag{6.6}$$

wobei *n* die Anzahl der Grundelemente, bzw. der unterschiedlichen Schaltgrenzen ε_i, angibt. Kann keine homogene Verteilung der Schaltgrenzen garantiert werden, soll die mittlere Differenz aufeinanderfolgender Schaltgrenzen als Auflösung gelten. Im dargestellten Fall mit fünf Grundelementen ergeben sich somit vier Zeitabschnitte im Messbereich. Vom Konzept her liegt es also nahe, auf dem räumlich begrenzten Mikrochip möglichst viele Grundelemente möglichst dicht zu platzieren, um eine hohe Auflösung der Zeitmessung zu erreichen.

Als Ergebnis der Zeitmessung wird der Vektor betrachtet, der sich aus den entsprechend ihrer Platzierung oder ihrer Zeitgrenze sortierten Statuswerten der Grundelemente ergibt. Alle Elemente, die zuerst durch das Start-Signal erreicht werden, besitzen den Status logisch-0. Alle Elemente, bei denen das Stop-Signal zuerst eintrifft, besitzen den Status logisch-1. Verdeutlicht wird dies an einem einfachen Beispiel unter Zuhilfenahme der Abbildung 30: Geht man von einem unbekannten Zeitintervall Δt aus, für das gelten soll: $\varepsilon_3 < \Delta t < \varepsilon_4$, so werden die Grundelemente 1 bis 3 nach Gleichung (6.3) den Status logisch-0 einnehmen, wohingegen die Grundelemente 4 und 5 in den Status logisch-1 wechseln. Der Vektor der Statuswerte Q(1..5) ergibt sich also zu Q = 00011. Es ist am Vektor der Statuswerte ersichtlich, dass die Grenze der beiden möglichen Statuswerte zwischen Grundelement 3 und Grundelement 4 liegt. Als Rückschluss kann nun formuliert werden, dass das Zeitintervall Δt ebenfalls zwischen den charakteristischen Schaltgrenzen ε_3 und ε_4 liegen musste, um diesen Vektor zu erzeugen. Technisch verständlich ist auch die Erklärung, dass die Verbindungen vom Eingang des Start-Signals zu den Grundelementen 4 und 5 so lang waren, dass das Stop-Signal mit den kürzeren Verbindungen trotz des späteren Auftretens am Eingang vor dem Start-Signal an beiden Grundelementen eintraf.

Diese Form des Ergebnisses ergibt sich immer, unabhängig von der Anzahl der eingesetzten Grundelemente oder ihrer Platzierung auf dem Mikrochip. Das Ergebnis entspricht damit einer Thermometer-Kodierung: je mehr Grundelemente den Wert logisch-0 einnehmen, desto länger war das Zeitintervall, je mehr Grundelemente den Wert logisch-1 einnehmen, desto kürzer war es. Ein wichtiger Vorteil der BOUNCE-Architektur gegenüber den Tapped Delay Lines ist die Tatsache, dass auch „negative Zeitintervalle“ gemessen werden können,

also wenn das Stop-Signal vor dem Start-Signal auftritt. Dies ist besonders dann von Bedeutung, wenn die Reihenfolge der beiden Ereignisse, die das zu messenden Zeitintervall definieren, nicht bekannt ist[4].

[4] Die Ausdrücke „Start“ und „Stop“ bezeichnen dann lediglich den einen bzw. anderen Eingang der Schaltung, haben aber nicht mehr die ursprüngliche Bedeutung für das Zeitintervall.

7. BOUNCE: Praktische Umsetzung

Nachdem das vorhergehende Kapitel die theoretischen Grundlagen der BOUNCE-Architektur beschrieben hat, widmet sich dieses Kapitel der praktischen Umsetzung der BOUNCE-Architektur auf einem modernen FPGA. Als Trägerplattform dient ein FPGA, das in diesem Fall Bestandteil eines kompletten Entwicklungsboards des FPGA-Herstellers Altera ist. Das FPGA nimmt die aus dem Konzept bekannten Bestandteile der BOUNCE-Architektur auf. Daher wird in diesem Kapitel auf folgende, grundlegende Aspekte der FPGA-Variante der BOUNCE-Architektur eingegangen:

1. Die Signalleitungen, welche die Eingangssignale an die Grundelemente weiterleiten.
2. Die Eingangspads, welche die Ereignisse *Start* und *Stop* auf die Signalleitungen übertragen.
3. Die Grundelemente, die als Entscheidungselemente arbeiten und kennzeichnen, welches der beiden Signale das Grundelement zuerst erreicht hat.
4. Den Einfluss der geometrischen Verteilung der Grundelemente in Bezug auf die Größen Messbereich und Genauigkeit der BOUNCE-Architektur.

7.1. Signalleitungen

Kern der BOUNCE-Architektur ist die Aussage, dass Signalleitungen aufgrund ihrer physikalischen Eigenschaften eine Verzögerung des auf ihnen übertragenen Signals verursachen. Die Größe der Verzögerung hängt dabei von der Länge der Signalleitung ab. BOUNCE basiert darauf, zu allen Grundelementen verschiedene Leitungslängen zu realisieren, damit vom Eingangs-Pin, bzw. Eingangs-Pad, des jeweiligen Signals ebenfalls unterschiedliche Laufzeiten zu den Grundelementen entstehen. Diese unterschiedlichen Laufzeiten sind die technische Grundlage für die Übertragung der Zeitdifferenz zwischen den beiden Signalen *Start* und *Stop* in eine ortsabhängige Darstellung in den Grundelementen. Abbildung 31 verdeutlicht dieses Verhalten vereinfacht anhand zweier Grundelemente.

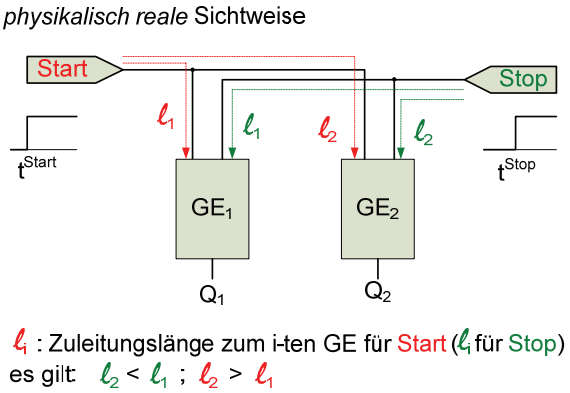


Abbildung 31: die Verteilung der Grundelemente (GE) auf dem FPGA führt zu unterschiedlich langen Signalpfaden

Um die Erläuterungen zu vereinfachen, verwendet die Konzeptdarstellung (Abbildung 30) separate Signalleitungen unterschiedlicher Länge für jedes Grundelement. Aus technischer Sicht bestehen bei dieser Art der Verschaltung deutliche Nachteile bei der Übertragung des Konzepts in das FPGA. Zum einen benötigen *n* Grundelemente *2n* Signalleitungen um die Grundelemente mit beiden Eingangs-Pads für *Start* und *Stop* zu verbinden. Abschnitt 4.3 hat bereits die im FPGA befindlichen Strukturen zur Signalübertragung dargestellt. Diese sind bei der Verwendung von zwei dedizierten Signalleitungen pro Grundelement schnell erschöpft. Zum anderen ist zu berücksichtigen, dass die Summe aller Leitungslängen pro Eingang sehr hoch ist und damit eine hohe kapazitive Belastung der Eingänge einhergeht. Dies wiederum führt zu einer Abflachung und damit Verschlechterung übertragener Signalflanken. Umgangen werden diese Nachteile bei der Übertragung der BOUNCE-Architektur in das FPGA dadurch, dass die Grundelemente an einen Signalbus angeschlossen werden, wie in der folgenden Abbildung dargestellt.

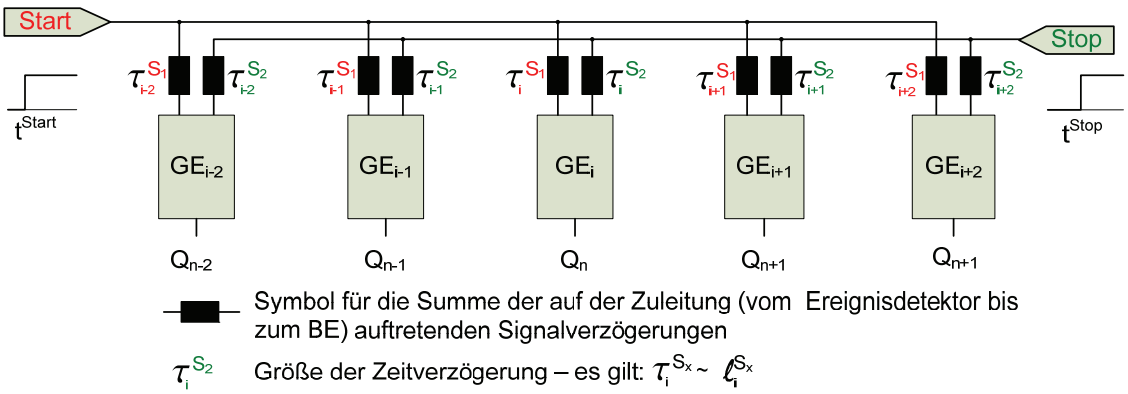


Abbildung 32: schematische Darstellung der Entstehung unterschiedlicher Verzögerungszeiten auf den Signalpfaden zu den einzelnen Grundelementen in einer busartigen Struktur

Die Tatsache, dass die Grundelemente parallel an den aus den beiden Signalen *Start* und *Stop* gebildeten Signalbus angeschlossen sind, führt dazu, dass lediglich zwei

Signalleitungen unter Verwendung der globalen Routing-Kanäle über das FPGA geführt werden. Dadurch ist auch eine deutlich reduzierte kapazitive Belastung an den Eingängen erreicht. Dennoch ist die wesentliche Forderung der BOUNCE-Architektur nach unterschiedlichen Laufzeitverzögerungen der Signale erfüllt, da jedes Element eine andere Entfernung zum Eingang eines Signals besitzt. Diese unterschiedlichen Laufzeiten und die damit verbundene Zeitverzögerung der Signale sind in der Abbildung durch das Symbol τ gekennzeichnet. Wie bereits ausgeführt gilt, dass τ_i proportional zu l_i ist, wobei l_i jeweils die Signalpfadlänge vom Eingang zum Grundelement beschreibt.

7.2. Eingangs-Pads

Bereits in der Abbildung ist zu erkennen, dass die Eingänge für *Start* und *Stop* an gegenüberliegende Seiten platziert wurden. Dies ist nicht zufällig geschehen sondern stellt eine besonders geeignete Form für die BOUNCE-Architektur dar. Je größer der räumliche Abstand der Eingangspads ist, desto größer ist die Diversität der unter Berücksichtigung der dargestellten Verteilung der Grundelemente erzeugten Laufzeitdifferenzen ε_i. Weiterhin ist der Messbereich signifikant vom räumlichen Abstand der beiden Eingänge abhängig. Die folgende Abbildung illustriert diese Zusammenhänge an zwei Beispielen. Die in der Abbildung angegebenen Zeitwerte sind exemplarischer Natur.

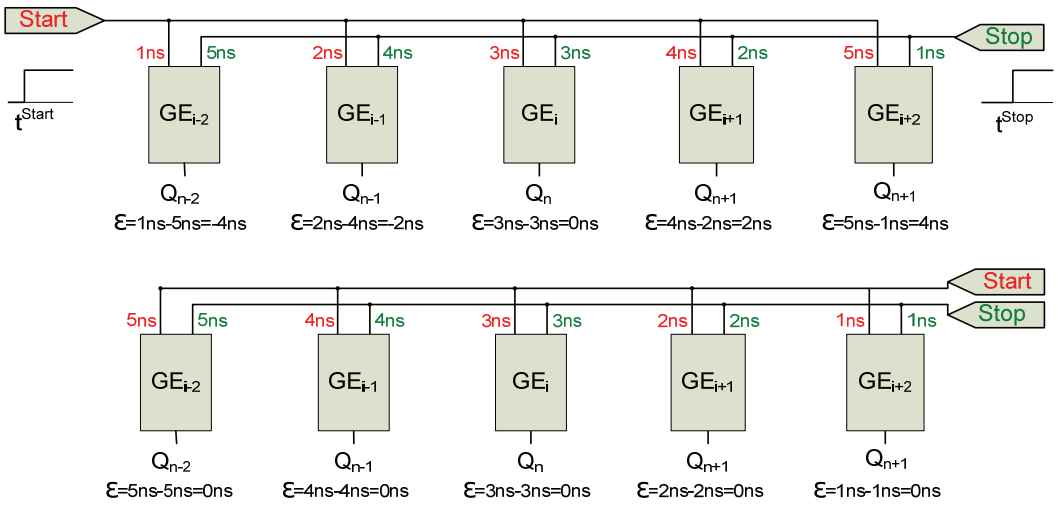


Abbildung 33: Einfluss der Positionen der Signaleingänge auf die Bildung der charakteristischen Differenzen

Die Abbildung stellt zwei Extremfälle der Platzierung der Eingangspads dar. Im oberen Teilbild liegen diese gegenüber, was dazu führt, dass sich im dargestellten Verbund der 5 Grundelemente ebenfalls 5 charakteristische Zeitdifferenzen ε_i ergeben. Im Beispiel sind diese ε_i = -4ns, -2ns, 0ns, 2ns, 4ns. Das System besitzt also einen Messbereich von 8 ns bei einer Auflösung von 2 ns. Im unteren Teilbild befindet sich das gleiche System aus fünf

Grundelementen, nur befinden sich dieses Mal die Eingangs-Pads direkt nebeneinander. Das hat zur Folge, dass die Laufzeiten zu den Grundelementen für beide Signale (idealisiert) identisch sind und sich nur eine einzige charakteristische Differenz von 0 ns ergibt. Das System kann in dieser Konfiguration keine sinnvollen Messungen durchführen.

7.3. Orientierung der Grundelemente

Die vorhergehenden Abschnitte haben bereits dargestellt, dass die Größe des Messbereichs von der Länge der Signalstrecke zwischen den beiden Eingangs-Pads abhängig ist. Dabei ist allerdings ein weiterer Punkt zu berücksichtigen, der in Abschnitt 7.2 stillschweigend vorausgesetzt wurde, um die Aussagen auf die Eingangs-Pads zu konzentrieren. Nicht nur der Abstand der Pads ist von Bedeutung für den Messbereich sondern ebenfalls die Art und Weise, wie die Grundelemente zwischen den Pads platziert werden. Die Länge der Signalleitungen zwischen den Eingangs-Pads definiert die maximal möglichen Verzögerungszeiten für jeweils ein Signal. Die maximalen Verzögerungszeiten D beider Signale bestimmen den theoretischen Messbereich M_t. Explizit kann er durch die folgende Formel beschrieben werden:

$$|M_t| = |D_{start} + D_{stop}|$$

Der praktisch nutzbare Messbereich hängt nun allerdings von der Orientierung der Grundelemente ab, da so festgelegt wird, wie groß die Abdeckung des theoretischen Messbereichs ist. Die folgende Abbildung stellt zwei Extrema in Bezug auf die Orientierung der Grundelemente gegenüber.

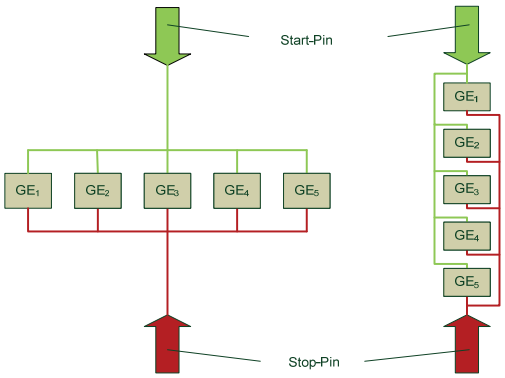


Abbildung 34: Vergleich von horizontaler und vertikaler Platzierung und daraus resultierender Signalpfadlängen zu den einzelnen Grundelementen

Im linken Teilbild, mit orthogonal zur Verbindungslinie der Eingangspins angeordneten Grundelementen, ist die Laufzeitdifferenz beider Signale zu den Grundelementen nahezu gleich. Das führt zu dem Ergebnis, dass alle fünf Grundelemente die nahezu gleiche charakteristische Zeitdifferenz ε_i aufweisen. Fertigungs- und Strukturvariationen sind verantwortlich dafür, dass nicht überall identische Laufzeitdifferenzen entstehen. Die Folge ist,

dass der theoretische Messbereich nicht vollständig ausgenutzt wird. Im rechten Teilbild mit den parallel zur Verbindungslinie der Pins angeordneten Grundelementen entstehen deutlich verschiedene charakteristische Zeitdifferenzen ε_i. Obwohl der Abstand der Pins nicht verändert wurde, und somit der theoretische Messbereich konstant bleibt, ist der praktische Messbereich wesentlich größer als im linken Teilbild. Der Effekt ähnelt der bereits im vorhergehenden Abschnitt geführten Diskussion (vgl. Abbildung 33), nur das die Ursache für das geschilderte Verhalten diesmal nicht mit der Position der Eingangs-Pads sondern mit der Orientierung der Grundelemente begründbar ist.

Somit ist ein großer Messbereich davon abhängig, dass sowohl die Pads eine räumliche Distanz aufweisen, als auch die Grundelemente parallel zu den möglichst entgegengesetzt verlaufenden Signalen orientiert sind. Aber auch die Variante der orthogonal angeordneten Grundelemente hat ihre Daseinsberechtigung. Sie weist zwar einen extrem kleinen Messbereich auf, hat aber einen positiven Einfluss auf die Auflösung der Zeitmessung. Leicht nachvollzogen werden kann das ebenfalls an Abbildung 34. Während im rechten Teil der Darstellung die fünf Grundelemente auf den nahezu maximal ausgenutzten theoretischen Messbereich verteilt sind, ist der Messbereich im linken Bild zwar dramatisch kleiner, aber ebenfalls mit fünf Grundelementen besetzt. Somit gilt: Während die orthogonale Verteilung einen sehr kleinen Messbereich mit einer hohen Auflösung bietet, entsteht bei der parallelen Verteilung ein großer Messbereich mit einer schlechteren Auflösung. Die Verbindung beider Verteilungsvarianten führt in der Summe dann aber zu einem System, in dem gleichzeitig beides erreicht wird: ein möglichst großer Messbereich bei einer hohen Auflösung.

7.4. Das idealisierte Grundelement

Das Grundelement hat zur Aufgabe, die Reihenfolge des Eintreffens beider Signale, *Start* und *Stop*, am Grundelement zu bestimmen. Im Ruhezustand vor der Messung besitzen beide Signaleingänge des Grundelements den Zustand logisch-0. Während der Messung wechseln beide Signale den Zustand auf logisch-1. Für den Messvorgang ist die Reihenfolge der Signaländerungen am Grundelement entscheidend, da alle Grundelemente im Verbund die zeitliche Abhängigkeit der Signale in eine Ortsinformation umwandeln. Es ergibt sich eine „Abbildung“ in Abhängigkeit von der Zeitdifferenz und des Ortes eines jeden Grundelementes. In dieser „Abbildung“ ist dargestellt, welches Signal welche Grundelemente zuerst erreicht hat. Aus dieser Ortsinformation wird die Zeitinformation durch Vergleich mit bekannten „Abbildungen“ für bekannte Zeitdifferenzen zurückgewonnen.

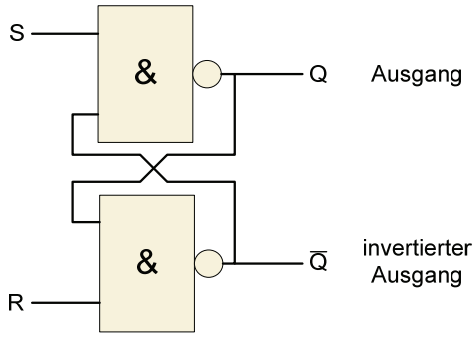
Im Ruhezustand, wenn beide Signaleingänge logisch-0 anzeigen, ist der Ausgang des Grundelements auf logisch-1 gesetzt. Erreicht das Start-Signal den Zustand logisch-1 vor

einer Änderung des Stop-Einganges, wird der Ausgang auf logisch-0 geschaltet. Ändert sich der Zustand des Stop-Einganges auf logisch-1 bevor eine Änderung des Start-Einganges erfolgt, wird eine Zustandsänderung am Ausgang blockiert, sodass eine spätere Änderung des Start-Einganges keine Auswirkung auf den Ausgang mehr hat. Zusammenfassend ergibt sich somit die Wahrheitstabelle des Grundelementes:

Signal 1 (*Start*)	Signal 2 (*Stop*)	Ausgang des Grundelements	Bemerkungen
Logisch-0	Logisch-0	Z_1 (logisch-1)	Ruhezustand vor der Messung
Logisch-0	Logisch-1	Z_1 (logisch-1)	Signal 2 erreicht das GE zuerst (A)
Logisch-1	Logisch-0	Z_2 (logisch-0)	Signal 1 erreicht das GE zuerst (B)
Logisch-1	Logisch-1	Z_1 oder Z_2	Messung beendet, Zustand abhängig von (A) oder (B)

Bei den in der Wahrheitstabelle festgehaltenen Informationen ist zu berücksichtigen, dass im Rahmen dieser Arbeit die Ereignisse *Start* und *Stop* als Übergang des Signalpegels von logisch-0 zu logisch-1 definiert wurden. Diese Festlegung erfolgte willkürlich, genauso wie die Festlegung der Ausgangszustände des Grundelementes. Insgesamt stellt dies aber keine Einschränkung dar, da diese Definitionen beliebig geändert werden können ohne das Funktionsprinzip der BOUNCE-Architektur zu beeinflussen. Wichtig ist bei der Festlegung lediglich, dass das Grundelement zwei Zustände unterscheidet: Z_1 für eine Signaländerung *Start* vor *Stop* und Z_2 für den umgekehrten Fall.

Aus technischer Sicht bietet sich das RS-FlipFlop für die Realisierung der oben gezeigten Wahrheitstabelle an. Das RS-FlipFlop kann in einer Variante aus NAND-Gattern oder in einer Variante mit NOR-Gattern realisiert werden. Für die Verwendung in der BOUNCE-Architektur ist entscheidend, dass das RS-FlipFlop in beiden Varianten vier funktionale Zustände unterscheidet: Setzen, Rücksetzen, Speichern und einen irregulären Zustand. Die weiteren Ausführungen beziehen sich auf das aus NAND-Gattern aufgebaute RS-FlipFlop, weil dieses in der praktischen Umsetzung der BOUNCE-Architektur verwendet wird. In der folgenden Abbildung sind die Struktur und die Wahrheitstabelle des verwendeten RS-FlipFlops abgebildet. Es sei darauf hingewiesen, dass zur Verwendung des RS-FlipFlops mit der gebräuchlichen Semantik *Setzen* (*S=1/R=0*), *Rücksetzen* (*S=0/R=1*) und *Speichern* (*S=0/R=0*) eine Invertierung beider Eingänge erforderlich ist.



S	R	Q	$\overline{Q}$
0	0	1	1
0	1	1	0
1	0	0	1
1	1	Q'	$\overline{Q}$'

Abbildung 35: technische Realisierung eines NAND-basierten RS-FlipFlops mit dazugehörender Wahrheitstabelle

Der irreguläre Zustand, gekennzeichnet durch *S=0* und *R=0*, ist deshalb nicht zulässig, weil die beiden komplementären Ausgänge des FlipFlops den gleichen Zustand logisch-1 einnehmen. Dabei handelt es sich nicht um einen technischen Fehler sondern vielmehr um ein logisches Problem, welches zu Fehlverhalten in nachgeschalteten Hardware-Komponenten führt, wenn beide Ausgänge des FlipFlops ausgewertet werden. Innerhalb der BOUNCE-Architektur stellt dies allerdings kein Problem dar, da lediglich der Q-Ausgang verwendet wird. Der Zustand *S=0* und *R=0* stellt den Default- oder Ruhezustand vor Messung dar. Die beiden Zustände Z_1 und Z_2 aus der Wahrheitstabelle des Grundelements korrespondieren mit den Zuständen *S=1/R=0* bzw. *S=0/R=1*. Erreichen beide Signale den Zustand logisch-1, speichert das RS-FF den vorhergehenden Zustand, also Z_1 oder Z_2. Noch während beide Signale den Zustand logisch-1 aufweisen und das RS-FlipFlop als Speicher fungiert, wird der aktuelle Zustand des Grundelements durch das BOUNCE-System ausgewertet.

Kritischer Punkt des RS-FlipFlops ist der Eintritt in den Zustand *S=1 und R=1*. Erfolgen beide Signaländerungen quasi gleichzeitig, ist unbestimmt, welcher Zustand gespeichert wird. Aus logischer Sicht kommt es zu einer Wettlaufsituation zwischen den beiden Zuständen Z_1 / Z_2. Aus technischer Sicht kann diese Wettlaufsituation zu einem Schwingverhalten des verwendeten Ausgangs *Q* des RS-FlipFlops führen. In der Praxis konnte ein Schwingverhalten allerdings nicht beobachtet werden. Gesondert wird dieser Aspekt noch mal im Abschnitt zur Stabilität des BOUNCE-Grundelements aufgegriffen.

Im Hinblick auf die FPGA-Umsetzung ist festzuhalten, dass das beschriebene RS-FlipFlop nicht aus dedizierten NAND-Gattern aufgebaut wird sondern die Funktionalität des FlipFlops in einer Logiktabelle abgebildet wird. Der Umstand, dass das RS-FlipFlop lediglich zwei Dateneingänge und eine Signal-Rückkopplung benötigt erlaubt die Platzierung des FlipFlops in einer einzigen 4-Eingangs-Logiktabelle. Somit ist wiederum ein einziges Logikelement des FPGAs für die Aufnahme des RS-FlipFlops ausreichend. Daraus folgt, dass auch das Grundelement der BOUNCE-Architektur innerhalb eines einzigen Logikelements des FPGAs

abgelegt werden kann. Dieser Punkt spielt eine wichtige Rolle bei der Platzierung der BOUNCE-Architektur auf dem FPGA und wird im Abschnitt zur Synthese der Architektur noch einmal betrachtet.

7.5. Übertragung des Konzepts in das FPGA

Nachdem die beiden vorhergehenden Abschnitte die wesentlichen Aspekte der Übertragung der BOUNCE-Architektur in ein FPGA dargestellt haben, zeigt dieser Abschnitt einen ersten Prototyp der Architektur auf einem StratixII-FPGA. Die folgende Abbildung ist eine Darstellung der verwendeten Logikressourcen des FPGAs.

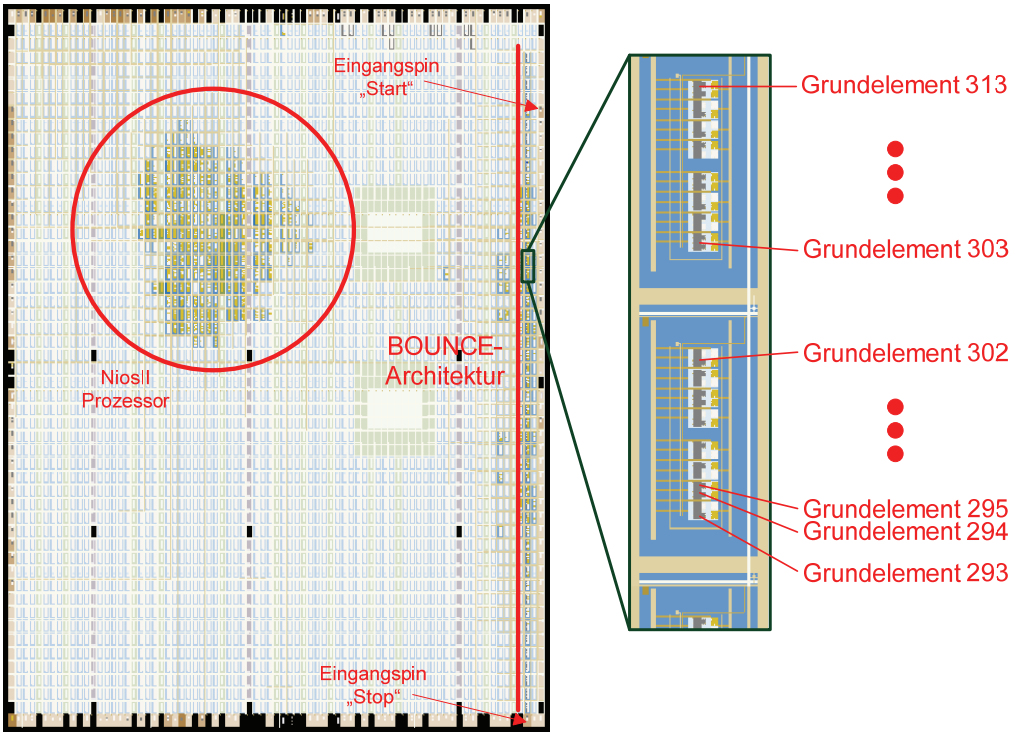


Abbildung 36: Vorstellung der ersten praktischen Umsetzung mit 512 Grundelementen in einer Reihe aus Logikelementen eines StratixII-FPGAs

Der Prototyp besteht aus 512 der angesprochenen Grundelemente, die allesamt innerhalb einer Spalte im FPGA platziert sind. Diese Architektur stellt eine der Referenz-implementationen der BOUNCE-Architektur dar, ein Teil der in Abschnitt 10 aufgeführten Ergebnisse beruht auf dieser Variante.

Die Eingänge der Signale befinden sich an der rechten Kante des FPGAs. Sie sind so gewählt, dass ein möglichst großer Abstand zwischen ihnen liegt, um einen großen Messbereich zu ermöglichen. Ein größerer Abstand zwischen den Eingangs-Pads kann nicht erreicht werden, da die äußere Beschaltung des FPGAs auf dem Entwicklungsboard dies verhindert. Die Grundelemente befinden sich allesamt in einer Spalte des FPGAs um eine effiziente Verschaltung mit den Signaleingängen zu garantieren. Im Hinblick auf die in

Abschnitt 7.3 gemachten Ausführungen wurde der theoretische Messbereich möglichst komplett ausgenutzt ohne durch weitere orthogonale Grundelemente eine Auflösungserhöhung anzustreben. Die Verbindung der Grundelemente mit den Eingangs-Pads erfolgt über einen Singalbus (vgl. Abschnitt 7.1), der entlang der rot eingezeichneten Line innerhalb der BOUNCE-Struktur verläuft. Der dargestellte NiosII-Prozessor dient der Steuerung und Auswertung der BOUNCE-Architektur.

Dieses Kapitel hat die wesentlichen theoretischen Aspekte der Überführung der BOUNCE-Architektur in ein handelsübliches FPGA beschrieben. Dabei wurde bisher nicht auf spezifische Details der Umsetzung eingegangen um das Verständnis dieser Überlegungen nicht zu erschweren. Das nächste Kapitel widmet sich daher den Anstrengungen, die notwendig sind, um mit Hilfe der FPGA-abhängigen Software-Tools eine in der Praxis funktionierende Schaltung zu erreichen.

8. FPGA-Details

Nachdem der vorhergehende Abschnitt die grundlegenden Zusammenhänge zwischen FPGA-Implementation und theoretischem Konzept der BOUNCE-Architektur dargestellt hat, bietet dieser Abschnitt ausführliche Informationen zur Synthese der Schaltung. Insbesondere stehen FPGA-spezifische Details, Eigenschaften der verwendeten Synthesetools und die Problematik der automatisierten Platzierung der Grundelemente im Fokus.

8.1. Synthese der Schaltung

Der Begriff „Synthese“ beschreibt den automatisierten Vorgang der Erstellung einer Konfigurationsdatei für das FPGA, wie bereits in Abschnitt 4.2 dargestellt. Dabei wertet das Synthesetool die grafische oder textuelle Beschreibung des Systems aus und erstellt die Konfigurationsvektoren für die Logikelemente. Im Rahmen dieser Arbeit ist vorrangig die Synthese der Grundelemente und ihrer Verschaltung zur BOUNCE-Architektur von Interesse.

8.1.1. Automatische Optimierung

Die Beschreibung des Grundelementes liegt in der Hardware-Beschreibungssprache VHDL vor. Funktional wird daraus die in Abbildung 35 dargestellte NAND-Struktur erzeugt, die wiederum in einen Konfigurationsvektor für ein Logikelement überführt wird. Da aber alle Grundelemente am selben Signalbus angeschlossen sind, meldet das Synthesetool eine funktionale Redundanz. Das liegt daran, dass das Synthesetool die Absicht dieser Verschaltung, nämlich die Ausnutzung der unterschiedlichen Laufzeiten zu den Grundelementen, nicht erkennt. Ohne Berücksichtigung der Laufzeiten ist an allen Grundelementen das gleiche Ergebnis zu erwarten, da ja alle Grundelemente dieselben Eingangssignale (*Start* und *Stop*) besitzt. Die folgende Abbildung illustriert diesen Aspekt.

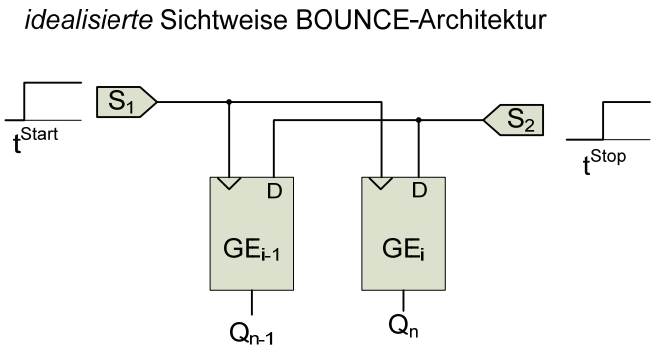


Abbildung 37: idealisierte Sichtweise der BOUNCE-Architektur im Synthesetool

Eine durch das Synthesetool vorgenommene Hardware-„Optimierung“ entfernt daraufhin alle Grundelemente bis auf eins. Das verbleibende Grundelement dient dann als „Stellvertreter“ für alle nicht mehr implementierten Grundelemente, da es ja dieselbe kombinatorische Funktion aufweist. Mit anderen Worten: eine aus mehreren Grundelementen gebildete BOUNCE-Architektur wird nur dann richtig synthetisiert, wenn entweder die automatische Hardware-Optimierung vollständig deaktiviert wird oder wenn eine Ergänzung der Schaltung diese Optimierung verbietet. Im Rahmen dieser Arbeit wird die zweite Möglichkeit genutzt.

8.1.2. Integration des Shift-Registers

Eine Option zur Unterdrückung der Optimierung ist die Integration eines Shift-Registers an den Ausgängen der Grundelemente. Ein Shift-Register ist eine Kette aus Registern, wobei der Ausgang eines Registers immer mit dem Eingang des Folgeregisters verbunden ist. Am Eingang eines Registers kann wahlweise der Ausgangswert des angeschlossenen Grundelementes übernommen werden, oder der Ausgang des vorhergehenden Registers in der Kette weitergeleitet werden. Abbildung 38 verdeutlicht die Integration des Shift-Registers.

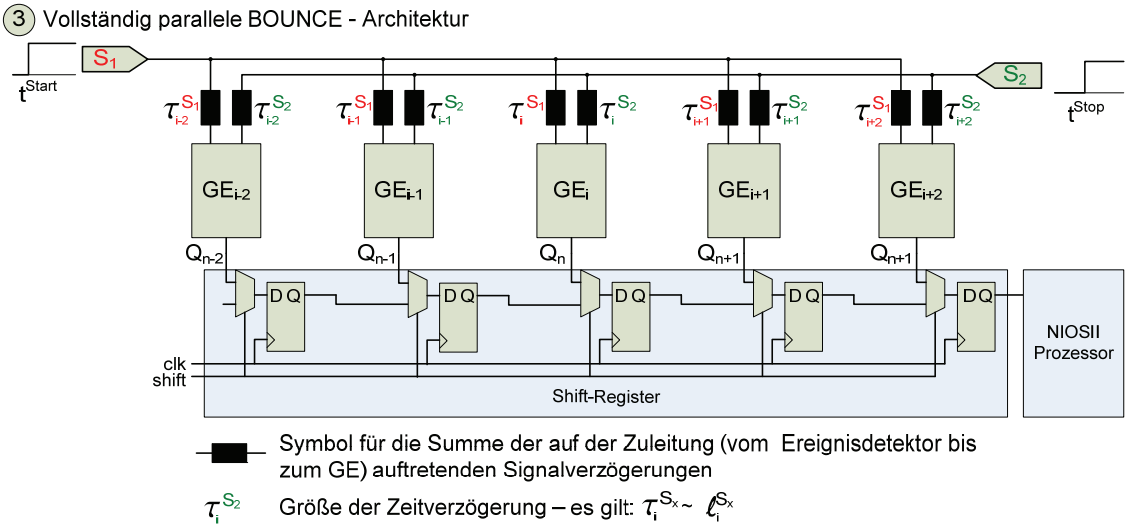


Abbildung 38: BOUNCE Architektur, mit Shift-Register erweitert

Durch dieses Shift-Register kann das Synthesetool nicht mehr von einer Redundanz der Grundelemente ausgehen, da der Inhalt des Shift-Registers nicht vorhergesagt werden kann. Dieser ist vielmehr abhängig von der zum Synthese-Zeitpunkt unbekannten Verwendung der Schaltung im Betrieb. Das Shift-Register erfüllt neben der Unterdrückung der Optimierung gleich noch einen zweiten Zweck: Da die Auswertung aller Grundelemente im bereits mehrfach angesprochenen und im FPGA implementierten NiosII-Prozessor erfolgt, muss eine Anpassung an die Schnittstelle zum Prozessor vorgenommen werden. Das Shift-Register übernimmt nach einer Messung gleichzeitig die Ergebniswerte aller Grundelemente. Der NiosII-Prozessor kann aber maximal 32 Datenbits gleichzeitig aufnehmen. Daher liest der Prozessor in einem Schritt lediglich die letzten 32 Bit des Schieberegisters aus und

verschiebt danach den Inhalt des Registers um 32 Stellen, sodass sich die nächsten 32 Ergebnisbits an der Leseposition befinden. Insofern ist die BOUNCE-Schaltung in Bezug auf die Anzahl der Grundelemente nicht durch den Prozessor limitiert. Derzeit ist bei der Erzeugung von Varianten der BOUNCE-Architektur lediglich darauf zu achten, dass die Anzahl der Grundelemente Vielfache von 32 sind, um eine richtige Funktion des Schieberegisters zu garantieren.

Ein weiterer Vorteil des Shift-Registers ist die Tatsache, dass dadurch die Größe der Schaltung auf dem FPGA kaum beeinflusst wird. Allen im Rahmen dieser Arbeit verwendeten FPGAs ist gemein, dass sie innerhalb eines Logik-Elementes neben der LUT ein separates Speicher-FlipFlop zur Sicherung des Ausgangswertes der Logik-Tabelle bereithalten. Da ja das Grundelement selbst in die Logik-Tabelle überführt wird, steht das Speicher-FlipFlop für das Shift-Register zur Verfügung. Lediglich die Steuerung des Shift-Registers erfordert weitere Logikelemente.

8.2. Platzierung der BOUNCE-Architektur im FPGA

Die synthetisierten Komponenten werden durch das Synthesetool auf Logikelemente des FPGAs abgebildet. Alle bekannten Synthesetools nehmen standardmäßig eine automatische Platzierung der erzeugten Elemente vor. Das bedeutet, dass die Funktionalität automatisch einem Logikelement des FPGAs zugeordnet wird, welches unter den Aspekten Geschwindigkeit und Ressourcenverbrauch als optimal angesehen wird. Im Regelfall führt dies z.B. zu einer Konzentration aller Logikelemente in einem bestimmten Bereich (z.B. in Nähe der Ein-/Ausgangspads des Systems) auf dem FPGA.

Das Konzept der BOUNCE-Architektur geht davon aus, dass möglichst viele Grundelemente verwendet werden. Diese Aussage muss bei praktischen Umsetzungen eingeschränkt werden. Ein modernes FPGA bietet mehrere 10.000 Logikelemente. Würde die BOUNCE-Architektur alle zur Verfügung stehenden Logikelemente nutzen, wäre die kapazitive Belastung der beiden Signaleingänge trotz der bereits angesprochenen Busstruktur derart hoch, dass kein vernünftiges Messsignal mehr garantiert werden kann. Infolgedessen muss die Anzahl der Grundelemente begrenzt werden. Durch diese Begrenzung kann allerdings nicht mehr die im Konzept besprochene, wahlfreie oder zufällige Platzierung der Elemente eingesetzt werden. Vielmehr ist es notwendig eine in Bezug auf Messbereich und Genauigkeit optimale Verteilung der Grundelemente im FPGA zu erreichen. Auf diese Problematik gehen die nächsten beiden Abschnitte ein.

8.2.1. Einfluss der FPGA-Struktur

Die Platzierung der Eingangspins und ihr Einfluss auf die Größe des Messbereichs wurden bereits in Abschnitt 7.1 besprochen. Je größer der geometrische Abstand der Signaleingänge, desto größer ist der Messbereich. In Anbetracht der rechteckigen Grundfläche des FPGAs entsteht der größtmögliche Abstand bei der Platzierung der Eingangspads für die Signale *Start* und *Stop* an diagonal gegenüberliegenden Ecken des FPGAs. Für die Platzierung der Grundelemente bietet sich dadurch ebenfalls die Diagonale an. Aufgrund der Routing-Eigenschaften innerhalb des FPGAs ist diese Wahl allerdings ungünstig. Für ein Verständnis dieser Aussage ist eine Betrachtung der Struktur der auf dem FPGA befindlichen Signalleitungen erforderlich.

Aus Kapitel 4 ist bekannt, dass ein FPGA eine Vielzahl von konfigurierbaren Logikelementen bereithält. Für die Verbindung dieser Logikelemente stehen in das FPGA integrierte Signalleitungen zur Verfügung, die durch konfigurierbare Schaltelemente mit den Logikelementen verbunden werden. Somit lassen sich Signalpfade zwischen beliebigen Logikelementen herstellen. Die Struktur der zur Verfügung stehenden Signalleitungen wurde ebenfalls in Abschnitt 4.3 bereits kurz angesprochen. Sie ist hierarchisch gegliedert. Die oberste Ebene der Signalleitungen bilden die globalen Signale. Diese liegen in horizontalen und vertikalen Routing-Kanälen und verlaufen zwischen den Logik-Array-Blocks. An den Kreuzungspunkten von vertikalen und horizontalen Kanälen ist durch konfigurierbare Schalter eine Verbindung zwischen den beiden Richtungen möglich. Ein vertikales bzw. horizontales globales Signal verläuft dabei nicht über die komplette Länge, bzw. Breite, des FPGAs. Vielmehr überspannt es nur einen kurzen Abschnitt von Logik-Array-Blocks. Bei der StratixII Architektur sind pro Richtung zwei Signaltypen für diese Aufgabe vorhanden. Sie übertragen ein Signal beispielsweise 4 (C4-Interconnect) oder 16 (C16-Interconnect) Logik-Array-Blocks in vertikaler bzw. 4 (R4-Interconnect) oder 24 (R24-Interconnect) Logik-Array-Blocks in horizontaler Richtung [5]. Muss ein Signal größere Strecken überbrücken, werden mehrere Abschnitte der globalen Signalleitungen zusammengeschaltet. Die beiden Eingangssignale *Start* und *Stop* verwenden diese globalen Signalleitungen zur Verbindung der Eingangs-Pads mit den Logik-Array-Blocks, in denen sich die durch Logikelemente gebildeten Grundelemente der BOUNCE-Architektur befinden. Eine vorrangig diagonale, kettenförmige Ausrichtung der Grundelemente würde folglich eine Vielzahl an Umschaltpunkten erfordern, da ständig zwischen vertikaler und horizontaler Ausbreitung gewechselt werden muss. Behält man stattdessen vorrangig eine Richtung (horizontal oder vertikal) bei, entstehen wesentlich weniger Umschaltpunkte, mit entsprechend verringerten kapazitiven Lasten und somit verbessertem Signalverhalten.

Von besonderer Bedeutung für die BOUNCE-Architektur ist die Verbindung eines Grundelementes zu den globalen Signalleitungen. Kein Logikelement ist direkt an die globalen Signalleitungen gekoppelt. Das Logikelement als kleinste Einheit ist nur an die Signalleitungen angeschlossen, die innerhalb eines Logik-Array-Blocks vorhanden sind. Diese bilden die zweite Hierarchieebene der Routing-Ressourcen im FPGA. Alle Signale eines Logik-Array-Blocks sind an einem Punkt über Schalter mit den globalen Routing-Kanälen zu verbinden. Somit ergibt sich die im Folgenden dargestellte Verschaltung der Grundelemente.

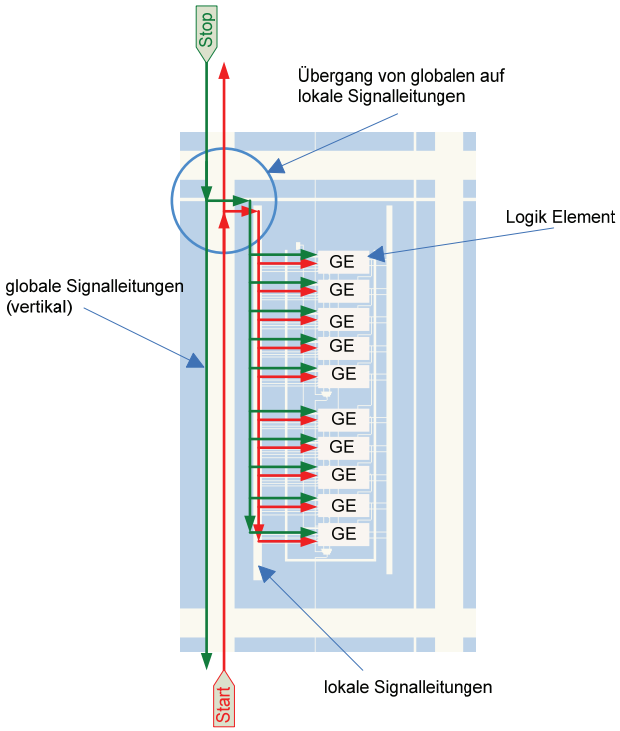


Abbildung 39: Reale Verschaltung der Grundelemente am Beispiel eines Cyclonell Logik-Array-Blocks

Von besonderer Bedeutung in Abbildung 39 ist der Übergang zwischen den globalen und lokalen Signalleitungen. Abbildung 33 hat bereits dargestellt, dass es innerhalb der BOUNCE-Architektur für die Erzeugung unterschiedlicher charakteristischer Differenzen ε_i günstig ist, wenn die Signale entgegengesetzt zu einander verlaufen. Auch in Abbildung 39 ist das so, allerdings laufen die Signale nicht bis zu jedem Grundelement in entgegengesetzte Richtungen sondern lediglich bis zum Übergang der Signale in den Logik-Array-Block. Somit entsteht die charakteristische Zeitdifferenz bereits dort. Ab dem Übergang in den Logik-Array-Block laufen die Signale parallel bis zu den Grundelementen. Im Vergleich mit Abbildung 33 entspräche dies dem Entstehen von lediglich einer charakteristischen Differenz unter allen Grundelementen eines Logik-Array-Blocks. Die praktischen Ergebnisse widersprechen diesen Überlegungen. Die Begründung dafür liegt in der fehlerhaften Annahme identischer Laufzeiten der Signale auf parallelen Pfaden. Es zeigt

sich, dass ebenfalls innerhalb eines Logik-Array-Block unterschiedliche charakteristische Zeitdifferenzen entstehen. Eine genaue, quantitative Beschreibung der zeitlichen Ordnung der Grundelemente befindet sich im Abschnitt zu den Ergebnissen.

Zusammenfassend sind bei der Verteilung der Grundelemente folgende Aspekte zu berücksichtigen:

1. Die Grundelemente werden vorrangig entlang einer Linie angeordnet, welche parallel zu einem globalen Routing-Kanal verläuft.
2. Innerhalb eines Logik-Array-Blocks entstehen durch Fertigungs- und Struktur-toleranzen unterschiedliche charakteristische Zeitdifferenzen ε_i an den einzelnen Grundelementen.

8.2.2. Automatisierte Positionierung der Grundelemente

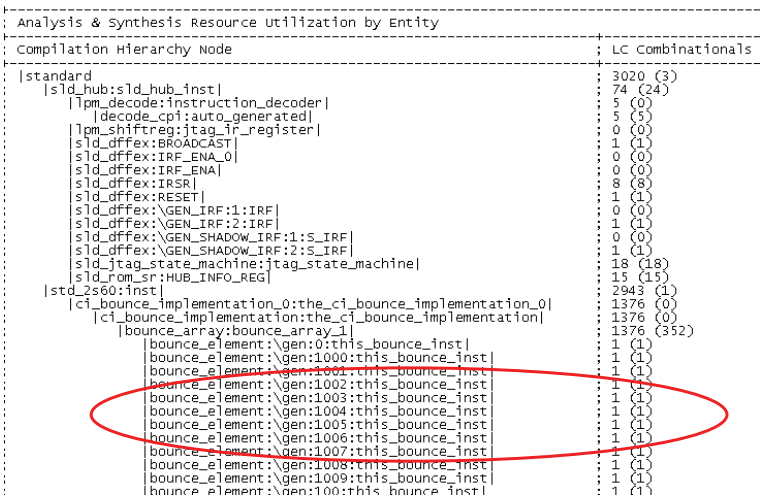
Die vorhergehenden Abschnitte haben bereits diskutiert, dass für ein reales Messsystem, im Gegensatz zum theoretischen Konzept, eine begrenzte Anzahl an Grundelementen vorrangig entlang vertikaler Ketten angeordnet werden sollte. Dafür ist die automatische Platzierung der Schaltungselemente innerhalb der Synthese-Tools zu umgehen. Die eingesetzten Synthesetools erlauben zwar die manuelle Zuweisung bestimmter Positionen an Logikelemente, allerdings ist dies in Anbetracht von mehreren Hundert Grundelementen, die die BOUNCE-Architektur bilden, wenig zweckmäßig. Aus diesem Grund wurde im Rahmen dieser Arbeit ein eigenes Software-Tool entwickelt, welches die Platzierung der Grundelemente auf dem FPGA automatisiert. Dieses Tool ist so gestaltet, dass es die Parameter des FPGAs und die Parameter der BOUNCE-Architektur entsprechend berücksichtigt. Zu den wesentlichen Parametern des FPGAs gehören:

- Anzahl der Spalten,
- Anzahl der Reihen,
- Anzahl der Logikelemente pro Spalte
- Gesperrte Sonderbereiche (Speicherblöcke, PLLs, etc.)

Die Parameter der BOUNCE-Architektur beschränken sich auf die Anzahl der Grundelemente und die gewünschte Geometrie.

Für die automatische Platzierung wertet das Platzierungstool die während eines Synthese-Vorgangs durch das Synthesetool erstellte Report-Datei in Bezug auf die erzeugten Grundelemente aus. Die Tatsache, dass ein Grundelement auf ein einziges Logikelement abgebildet wird, ist hier ein entscheidender Vorteil. In der Report-Datei ist aufgelistet, welche

Entitäten, also Schaltungskomponenten auf wie viele Logikelemente abgebildet werden. Zum besseren Verständnis zeigt Abbildung 40 einen Ausschnitt aus der angesprochenen Datei.



Analysis & Synthesis Resource Utilization by Entity

Compilation Hierarchy Node	LC Combinationals
|standard	3020 (3)
|sld_hub:sld_hub_inst|	74 (24)
|lpm_decode:instruction_decoder|	5 (0)
|decode_cpi:auto_generated|	5 (5)
|lpm_shiftreg:jtag_ir_register|	0 (0)
|sld_dffex:BROADCAST|	1 (1)
|sld_dffex:IRF_ENA_0|	0 (0)
|sld_dffex:IRF_ENA|	0 (0)
|sld_dffex:IRSR|	8 (8)
|sld_dffex:RESET|	1 (1)
|sld_dffex:\GEN_IRF:1:IRF|	0 (0)
|sld_dffex:\GEN_IRF:2:IRF|	1 (1)
|sld_dffex:\GEN_SHADOW_IRF:1:S_IRF|	0 (0)
|sld_dffex:\GEN_SHADOW_IRF:2:S_IRF|	1 (1)
|sld_jtag_state_machine:jtag_state_machine|	18 (18)
|sld_rom_sr:HUB_INFO_REG|	15 (15)
|std_2s60:inst|	2943 (1)
|ci_bounce_implementation_0:the_ci_bounce_implementation_0|	1376 (0)
|ci_bounce_implementation:the_ci_bounce_implementation|	1376 (0)
|bounce_array:bounce_array_1|	1376 (352)
|bounce_element:\gen:0:this_bounce_inst|	1 (1)
|bounce_element:\gen:1000:this_bounce_inst|	1 (1)
|bounce_element:\gen:1001:this_bounce_inst|	1 (1)
|bounce_element:\gen:1002:this_bounce_inst|	1 (1)
|bounce_element:\gen:1003:this_bounce_inst|	1 (1)
|bounce_element:\gen:1004:this_bounce_inst|	1 (1)
|bounce_element:\gen:1005:this_bounce_inst|	1 (1)
|bounce_element:\gen:1006:this_bounce_inst|	1 (1)
|bounce_element:\gen:1007:this_bounce_inst|	1 (1)
|bounce_element:\gen:1008:this_bounce_inst|	1 (1)
|bounce_element:\gen:1009:this_bounce_inst|	1 (1)
|bounce_element:\gen:100:this_bounce_inst|	1 (1)

Abbildung 40: Synthetisierte Grundelemente – Auszug aus einer vom Synthesetool erzeugten Report-Datei

Das selbstgeschriebene Platzierungstool nutzt diese Information, um zu bestimmen, wie viele Grundelemente in der Schaltung eingesetzt sind. Ist die Anzahl der Grundelemente erkannt, ordnet das Programm ihnen Positionen auf dem FPGA zu. Diese Positionen werden in Abhängigkeit eines gewählten Musters errechnet. Dabei unterstützt das Programm verschiedenste Muster, von der einfachen Linie über parallele Linien bis hin zu rechteckigen Strukturen oder diagonalen Linien. Die Positionsinformation besteht immer aus drei Angaben: in welcher Spalte des FPGAs sich der gewählte Logik-Array-Block mit dem zu verwendenden Logikelement befindet, in welcher Zeile der Logik-Array-Block liegt, und welche Ordnungsnummer das gewählte Logikelement innerhalb des Logik-Array-Blocks einnimmt. Diese Werte werden nach einer vorgegebenen Schreibweise zusammen mit der Bezeichnung des Logikelements in einen String umgewandelt und in eine separate Datei zurückgeschrieben. Dieser String hat folgenden Aufbau:

```
set_location_assignment LCCOMB_X63_Y26_N18 -to
"std_2s60:inst|ci_bounce_implementation_0:the_ci_bounce_implementatio
n_0|ci_bounce_implementation:the_ci_bounce_implementation|bounce_arra
y:bounce_array_1|bounce_element:gen:1005:this_bounce_inst|reg_in~0"
```

LCCOMB gibt an, dass es sich um die Angabe einer kombinatorischen Struktur handelt. *X63_Y26_N18* bezeichnen das Logikelement. Es ist das Logikelement mit der Nummer 18 im Logik-Array-Block, der in der 63. Spalte und der 26. Zeile des FPGAs liegt. Der Name in den Anführungszeichen benennt dass Element. Bei Vergleich mit Abbildung 40 fällt auf, dass es sich um den vollständigen Hierarchie-Namen des Grundelementes, ergänzt um die Nennung des kombinatorischen Ausgangssignals, handelt. Ein solcher String wird für alle

Grundelemente erstellt und gespeichert. Das Synthesetool liest diese sogenannte Assignment-Datei während einer weiteren Synthese der BOUNCE-Architektur aus und verarbeitet diese Informationen bei der eigentlichen Platzierung.

Durch das Platzierungstool entfällt einerseits die Notwendigkeit zur manuellen Platzierung mehrerer Hundert oder gar Tausend Grundelemente. Andererseits ist das Tool in der Lage, verschiedenste geometrische Verteilungen umzusetzen (z.B. Linie, Gitter, Rechteck, parallele Linien). Die Anzahl der verwendeten Grundelemente wird dabei automatische berücksichtigt, ebenfalls können verschiedene FPGA-Typen berücksichtigt werden. Dadurch wird eine flexible BOUNCE-Architektur mit einer frei wählbaren Anzahl Grundelementen unterstützt. Die folgende Abbildung zeigt exemplarisch zwei Varianten der BOUNCE-Architektur, deren Platzierung mit der erstellten Software erfolgt ist: im linken Teilbild eine quadratische Architektur bestehend aus 512 Grundelementen (noch ohne NiosII-Prozessor), rechts eine diagonale Linienstruktur mit NiosII-Prozessor, beide auf einem FPGA der Altera CycloneII Reihe.

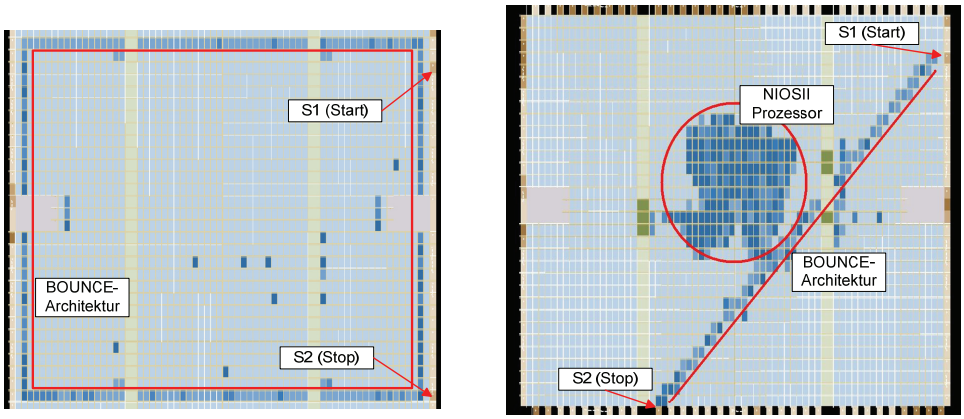


Abbildung 41: Mit Hilfe der automatischen Platzierung erzeugbare Varianten der BOUNCE-Architektur (Beispiele)

Die Abbildungen stammen aus einem in das Synthesetool integrierten Ansichtsprogramm, welches die tatsächliche Platzierung der Hardware auf dem FPGA illustriert. Jedes der blauen Quadrate im Bild entspricht einem Logik-Array-Block (vgl. Abbildung 20). Je dunkler die Blaufärbung, desto mehr der (hier 10) Logikelemente des Blocks sind belegt. Auffallend in der rechten Teilabbildung ist die kreisförmige Konzentration der zum NiosII-Prozessor gehörenden Logikelemente, die automatisch platziert wurden. Wie bereits in Abschnitt 8.2 beschrieben, erfolgt diese Form der Platzierung, da das Synthese-Tool besonders geringe Latenzen innerhalb des Prozessors erreicht, wenn alle dafür benutzten Logikelemente dicht beieinander liegen.

9. BOUNCE: Vollständiges Messsystem

Da die BOUNCE-Architektur lediglich der Grundbaustein eines Zeitmesssystems ist, erläutert dieses Kapitel die Komponenten des vollständigen Messsystems, welches für die Evaluierung und Bewertung der BOUNCE-Architektur im Rahmen dieser Arbeit genutzt wird.

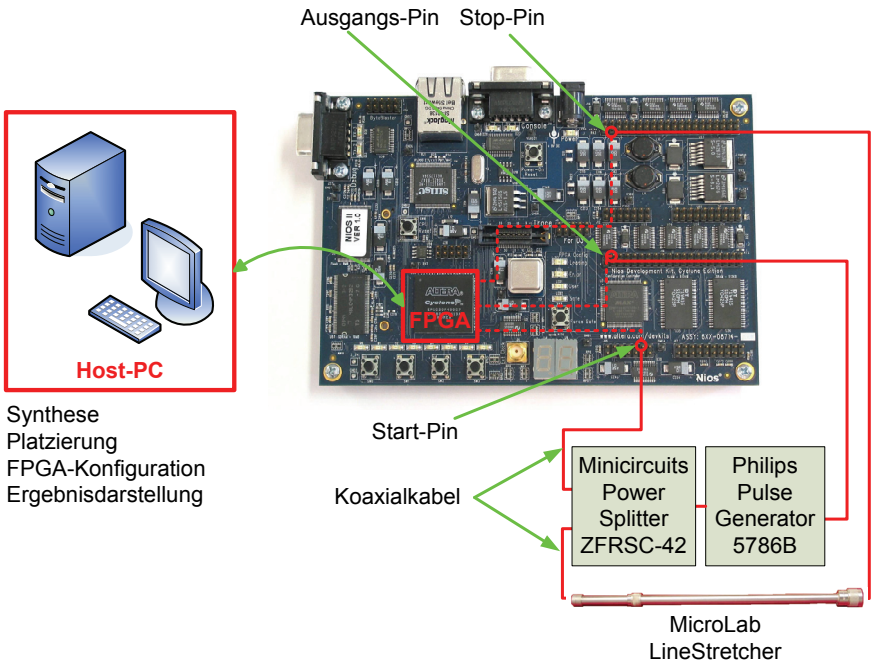


Abbildung 42: Bestandteile des Messsystems

Das vollständige Messsystem ist in Abbildung 42 dargestellt und wird durch folgende Komponenten gebildet:

1. Das Entwicklungsboard mit FPGA, Infrastruktur für die Kommunikation mit dem PC und Stiftleisten für den Anschluss der Messsignale.
2. Koaxialkabel (rot dargestellt) für die Verbindung der Einzelkomponenten
3. Philips Pulse Generator zur Erzeugung des Messsignals
4. Minicircuits Power Splitter zur Einspeisung des Messsignals in beide Signalpfade
5. MicroLab LineStretcher, ein ausziebares Koaxialrohr, zur Erzeugung unterschiedlicher Signallaufzeiten
6. Der Host-PC für Synthese, Download des Designs in das FPGA und Auswertung der Messergebnisse

Die BOUNCE-Architektur ist im FPGA implementiert, welches sich auf dem Entwicklungsboard befindet. Sie wurde in den vorhergehenden Kapiteln bereits ausführlich beschrieben. Zur Verwaltung und Steuerung der BOUNCE-Architektur ist ein NiosII soft-core Prozessor ebenfalls in das FPGA integriert. Da die Auswertung der Ergebnisse der BOUNCE-

Architektur aktuell auf dem PC erfolgt, kommuniziert dieser über die serielle Schnittstelle mit dem FPGA-internen NiosII-Prozessor. Der PC veranlasst über ein Kommando den NiosII-Prozessor, die Generierung eines Messsignals zu starten. Der NiosII-Prozessor wiederum legt das Signal zum Start einer Messung an einen Ausgang des FPGAs. Das Ausgangssignal ist mit einem frei konfigurierbaren Ausgangs-Pin einer Stiftleiste des Entwicklungsboards verbunden. Dieser Pin ist mit dem Trigger-Eingang eines Phillips Pulse Generators verbunden. Der Pulse Generator erzeugt das eigentliche Messsignal, welches durch den Minicircuits Power Splitter in zwei getrennte Signalpfade eingespeist wird. Einer der beiden Singalpfade ist über ein Koaxialkabel direkt mit einem Pin auf dem Entwicklungsboard verbunden und erreicht im weiteren Verlauf den Start-Eingang der BOUNCE-Architektur. Der zweite Signalpfad ist mit dem Stop-Eingang der BOUNCE-Architektur verbunden. Als Besonderheit ist allerdings in diesem Pfad noch ein ausziehbares Koaxialrohr, der MicroLab LineStretcher, integriert. Durch diesen ist es möglich, die Länge des Signalpfades um bis zu 26cm zu variieren. Damit ist das Messsystem in der Lage, unterschiedlich lange Signallaufzeiten auf diesem Signalpfad zu erzeugen, die sich wiederum in unterschiedlichen Zeitdifferenzen beider Signale bei der Ankunft im FPGA niederschlagen. Somit kann die BOUNCE-Architektur mit unterschiedlichen Zeitdifferenzen getestet werden. Die Ergebnisse der Messungen werden über die serielle Schnittstelle an den PC transferiert, wo derzeit noch die Auswertung und Visualisierung der Daten erfolgt.

Nach dieser einleitenden Vorstellung der Struktur und der Funktionsweise des prototypischen Messsystems widmen sich die folgenden Abschnitte einer detaillierten Darstellung und Erläuterung der einzelnen Bestandteile des Messsystems.

9.1. Steuerung der BOUNCE-Architektur mit dem NiosII

Da im Verlauf einer Messung mehrere hundert oder tausend Messdaten in Form von Statuswerten der Grundelemente der BOUNCE-Architektur entstehen, ist ein NiosII-Prozessor für ein sicheres Auslesen der Daten und zur Steuerung des Messvorgangs in das FPGA integriert. Insbesondere im Entwicklungsstadium wird der NiosII-Prozessor auch zur Kommunikation mit einem Host-PC genutzt, welcher wiederum eine adäquate Visualisierung und umfangreiche Auswertung der Daten ermöglicht. Der NiosII-Prozessor ist ein 32bit RISC-Prozessor, der von Altera für die Verwendung auf den FPGAs des Herstellers vertrieben wird [14]. Der Prozessor liegt in einer VHDL-Beschreibung vor, und kann mit Hilfe grafischer Softwaretools modifiziert und für verschiedene Verwendungszwecke konfiguriert werden. Aus diesem Grund wird er als „soft core processor“ bezeichnet. Die von Altera mitgelieferte Software-Entwicklungsumgebung erlaubt die einfache und schnelle Erstellung und Übersetzung von C-Programmen für den Prozessor. Sofern in dem FPGA-System neben dem Prozessor auch Speicherkomponenten integriert sind, wird die Software mithilfe

der Entwicklungsumgebung in den Speicher heruntergeladen und kann so im FPGA auf dem NiosII-Prozessor ausgeführt werden.

Eine Besonderheit dieses soft-core Prozessors ist die Möglichkeit, beliebige Hardware-Komponenten direkt in die CPU integrieren zu können [28] [25]. Altera hat diese Möglichkeit der Hardware-Integration entwickelt, um den Prozessor in seiner Funktionalität an problem-spezifische Anforderungen anpassen zu können. Beispielsweise besitzt der NiosII-Prozessor in der Standard-Version keine Multiplizier-Hardware. Multiplikationen werden während der Übersetzung der auszuführenden Software in entsprechende Additionen überführt. Ergänzt man die CPU mit entsprechender Multiplizierer-Hardware, erzeugt das Synthesetool automatisch Informationen über diese Hardware, die von der Software-Entwicklungs-umgebung ausgewertet werden. Insbesondere stellt diese dann entsprechende Assembler-Befehle und C-Makros zur direkten Nutzung der Hardware bereit. Funktional wird in diesem System auch die BOUNCE-Architektur als Bestandteil der CPU angesehen. Altera hat für diese Form der CPU-Ergänzung den Term „custom instruction“ gefunden, der verdeutlichen soll, dass es sich um eine funktionale, hardware-basierte Ergänzung der CPU handelt. Die Kommunikation zwischen dem NiosII-Prozessor und der BOUNCE-Architektur erfolgt über ein festgelegtes Standard-Interface für custom instructions, welches in [14] [6] ausführlich beschrieben ist.

Neben dem Auslesen des Shift-Registers der BOUNCE-Architektur übernimmt der NiosII-Prozessor auch die Kontrolle des Messvorgangs. Er erzeugt nach einer Aufforderung des Host-PCs ein Ausgangssignal, welches den im Prototyp eingesetzten Philips Pulse Generator veranlasst, das eigentliche Messsignal auszugeben, aus dem ein definiertes Zeitintervall für die Messung durch die BOUNCE-Architektur generiert wird. Nach einer Messung übernimmt der NiosII-Prozessor den Statusvektor der Grundelemente und versetzt die BOUNCE-Architektur wieder in den Grundzustand. Die gelesenen Statuswerte werden zur weiteren Verarbeitung an den PC transferiert.

9.2. Generierung von Zeitintervallen

Dadurch, dass mit der BOUNCE-Architektur extrem kurze Zeitintervalle im Bereich weniger Pikosekunden gemessen werden sollen, wurden mehrere Möglichkeiten zur Erzeugung von adäquaten Eingangssignalen getestet. Für erste Tests in Bezug auf die grundlegende Funktion der Architektur und Stabilität der Grundelemente wurde auf innerhalb des FPGAs generierte Signale zurückgegriffen. Im einfachsten Fall wurden beide Eingänge der BOUNCE-Architektur durch den Ausgang eines FPGA-internen FlipFlops beschaltet. Da vom FlipFlop ausgehend unterschiedlich lange Signalpfade zu beiden Eingängen bestanden, stellt sich so eine feste Zeitdifferenz Δt ein. Eine Neupositionierung des FlipFlops auf dem

FPGA führt zu veränderten Signalpfadlängen zu den beiden Eingängen der BOUNCE-Architektur und damit zu einer anderen Zeitdifferenz. Allerdings ist keine definierte Modifikation des Zeitverhaltens zwischen den beiden Signalen möglich, ohne die Position zu verändern. Vorteil der FPGA-internen Erzeugung des Messsignals ist die räumliche Nähe zur BOUNCE-Architektur, was äußere Störungen der Messsignale minimiert. Nachteilig ist, dass jede innerhalb des FPGAs erzeugte Zeitdifferenz in ihrer absoluten Größe nicht (oder zumindest nicht mit vertretbarem Aufwand) bestimmbar ist. Daher ist einer außerhalb des FPGAs liegenden Einrichtung der Vorzug zu geben.

Bei der Verwendung externer Geräte zur Erzeugung der Zeitdifferenzen ist insbesondere auf die zeitliche Stabilität der erzeugten Signale zu achten. Die Latenz der Signalgenierung ist dabei prinzipiell zu vernachlässigen. Der Abstand zwischen dem Zeitpunkt, an dem der NiosII-Prozessor die Messung startet bis zu dem Moment, an dem die Signale durch den Signalerzeuger in die Signalleitungen zum FPGA eingespeist werden, kann beliebig groß sein. Einzige Randbedingung ist die Forderung, dass die Latenz auf beiden Kanälen ungefähr gleich groß ist, damit die Zeitdifferenz zwischen beiden Signalen nicht außerhalb des Messbereiches liegt. Von essentieller Bedeutung ist allerdings die Stabilität der Latenz. Da die BOUNCE-Architektur zeitliche Auflösungen im Bereich von 10 ps und besser erzielen soll, darf die Latenz der Signalgenerierung auf beiden Kanälen nur um wenige Pikosekunden schwanken. Bereits Ungenauigkeiten von 2 ps auf beiden Kanälen lassen die zu messende Zeitdifferenz um vier Pikosekunden schwanken. Da Geräte zur Generierung von Signalen auf zwei Ausgangskanälen mit einem stabilen zeitlichen Versatz im Bereich weniger Pikosekunden Preise im Bereich von 10.000 € und mehr aufweisen, wurde auf ein simples Verfahren zurückgegriffen, welches nur ein Signal erzeugt. Aus diesem Signal werden durch ausschließlich passive Komponenten zwei Signale mit einem einstellbaren zeitlichen Versatz erzeugt. Die Beschränkung auf passive Komponenten führt zu einer besseren Stabilität als sie unter Verwendung aktiver Komponenten erreicht wird.

Da die Geschwindigkeit der Signalausbreitung in Koaxialkabeln mit 2/3 c bekannt ist [1], werden die für die Messungen benötigten Zeitdifferenzen mit verschieden langen Koaxialkabeln generiert. Im Messaufbau wird über das FPGA ein Philips Pulse Generator veranlasst, ein Rechtecksignal zu generieren. Das Ausgangssignal des Pulse Generators wird über einen Power Splitter ZFRSC-42-S+ des Herstellers Minicircuits in die zwei Signalstrecken eingespeist, die zu den Eingangspins auf dem FPGA-Entwicklungsboard führen. Um eine schnelle, einfache und sichere Umkonfiguration der Kabel zu ermöglichen, ist in eine der Signalleitungen eine in der Länge variierbare Koaxialstrecke eingesetzt. Genauer handelt es sich um den Coaxial Line Stretcher SR-05N des Herstellers microLab [45]. Dieser kann um 26 cm in seiner Länge verstellt werden. Vom Power Splitter zum Line

Stretcher und von diesem zum Eingangs-Pin des FPGA-Entwicklungsboards werden Koaxialkabel vom Typ RG174/U verwendet. Die zweite notwendige Signalstrecke wird komplett durch ein vorkonfektioniertes Koaxialkabel vom Typ RG174/U gebildet. Die Gesamtlängen beider Strecken sind mit ca. 2 m ähnlich groß. In Anbetracht der Ausbreitungsgeschwindigkeit elektrischer Signale in Koaxialkabeln von 2/3 c ist durch die Verstellmöglichkeit des Line Stretchers um 26 cm eine zusätzliche Signalverzögerung von maximal 26 cm * 50 ps/cm = 1300 ps möglich. Im Gegensatz zur FPGA-internen Erzeugung des Messsignals ist zwar mit einer verringerten Signalqualität zu rechnen, allerdings besteht der Vorteil dieser Variante darin dass tatsächlich definierte Zeitdifferenzen über den Zusammenhang Δt ~ Δs mit dem Line Stretcher erzeugt werden können.

9.3. PC-basiertes Auswerteverfahren

Die BOUNCE-Architektur verfolgt zwar das Ziel, eine vollständig FPGA-basierte Zeitmessung mit entsprechender on-chip Auswertung zu erreichen, allerdings wurde für die Entwicklungs- und Testphase ein Software-Tool für den PC erstellt, da dieser eine wesentlich komfortablere Auswertung und Visualisierung der Messergebnisse erlaubt. Die Software ist in C++ geschrieben und verwendet zur grafischen Darstellung die Grafik-Bibliothek Qt 3.3.0. Abbildung 43 zeigt einen Screenshot der Auswerte-Software während einer Messung.

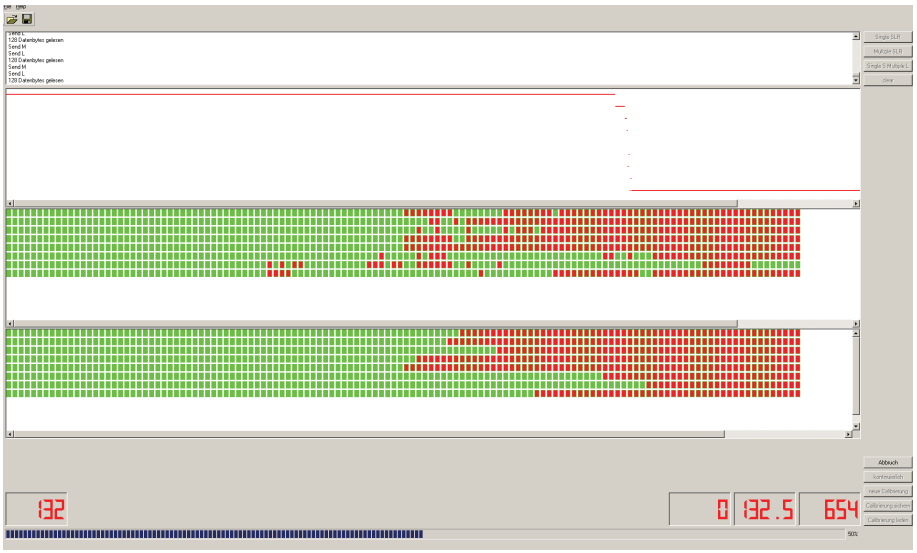


Abbildung 43: Auswertesoftware während einer Messung: dargestellt werden der Übergangsbereich (rote Line), die Grundelemente mit dem Zustand logisch-1 (mittig entsprechend der Verteilung auf dem FPGA, unten entsprechend ihrer charakteristischen Differenz), die errechnete Position auf dem Line Stretcher und die absolute Anzahl der Grundelemente mit dem Zustand logisch-1

Die Software kommuniziert über die serielle Schnittstelle des PCs mit dem im FPGA implementierten NiosII-Prozessor. Über einfache Byte-Kommandos veranlasst die Software den NiosII-Prozessor, die gewünschte Operation mit der BOUNCE-Architektur auszuführen. Die wesentlichen Funktionen sind dabei das Rücksetzen der BOUNCE-Architektur, das Veranlassen einer Messung und das Auslesen des Ergebnisvektors. Der Ergebnisvektor wird Byte-weise gelesen und intern in ein Array abgespeichert. Dadurch steht er auf dem PC für die Auswertung zur Verfügung. Die angesprochenen Einzelaktionen erlauben in ihrer entsprechenden Kombination dem Nutzer die einfache Durchführung unterschiedlicher Aktionen. Dazu zählen vorrangig die Kalibrierung der BOUNCE-Architektur, die Durchführung von Einzelmessungen und die Durchführung von Mehrfachmessungen. Ebenfalls in die Software integriert sind verschiedene Auswertevorgänge, wie z.B. Mittelwertbildung bei Mehrfachmessung, Durchführung der Laufzeitberechnung auf Basis der gelesenen Ergebniswerte der BOUNCE-Architektur und eine Reihe von Darstellungs-möglichkeiten. Jede dieser Funktionen wird in den folgenden Kapiteln an geeigneter Stelle noch einmal im Detail beschrieben. Zu guter Letzt ist eine Sicherung der gewonnenen Daten in einfache Textdateien implementiert, damit aufwendigere Auswertungen und adäquate Grafiken mit externen Kalkulationsprogrammen erstellt werden können.

10. Ergebnisse

Nachdem das vorhergehende Kapitel die theoretischen und technischen Aspekte der Laufzeitmessung mit der BOUNCE-Architektur behandelte, sind die praktischen Ergebnisse der BOUNCE-Architektur Kern dieses Abschnittes. Im ersten Schritt widmet sich dieses Kapitel der Betrachtung des realen Verhaltens des einzelnen Grundelementes der BOUNCE-Architektur. Dabei steht neben dem eigentlichen Schaltverhalten auch die Bestimmung zeitlicher Größen im Mittelpunkt, welche das Grundelement charakterisieren.

Der Verbund der einzelnen Grundelemente, welcher in seiner Gesamtheit den wichtigsten Teil der BOUNCE-Architektur darstellt, ist Gegenstand des zweiten Abschnitts. Dieser Abschnitt diskutiert, in welcher Art und Weise die Messergebnisse der BOUNCE-Architektur unter Berücksichtigung der Eigenschaften der einzelnen Grundelemente entstehen.

Den Abschluss dieses Kapitels bildet die Darstellung der erreichten zeitlichen Auflösung sowie der erzielten Messgenauigkeit für verschiedene Messverfahren (Einzelmessung, Mehrfachmessung).

Es wurde bereits mehrfach erwähnt, dass BOUNCE eine äußerst flexible Architektur besitzt, die auf verschiedensten FPGAs eingesetzt werden kann. Alle hier vorgestellten Ergebnisse basieren, soweit nicht anders angegeben, auf einer BOUNCE-Variante mit 1024 Grundelmenten in 8 parallelen Reihen auf einem StratixII-FPGA (EP2S60).

10.1. Das reale Grundelement

Die Gleichungen 2.33 und 2.34 in Kapitel 6.2 beschreiben das Zeitverhalten des idealen Grundelements. Ist das zu messende Zeitintervall größer als die charakteristische Differenz des Grundelements, wechselt der Status des Elements auf logisch-0, ist das Zeitintervall kleiner, bleibt er auf logisch-1. Unterschieden werden diese beiden Möglichkeiten durch die Reihenfolge der Signaländerungen auf den Leitungen *Start* und *Stop* am Eingang des Grundelements, wie in Abschnitt 7.4 dargestellt.

Unter realen technischen Bedingungen kann ein solches sprungartiges Verhalten zwischen den beiden möglichen Ausgangszuständen nicht erreicht werden. Die endliche Steilheit der Signalflanken und die notwendige Zeit für die Umladevorgänge am Ausgang des FlipFlops sind dafür verantwortlich, dass sich vielmehr ein Übergangsbereich zwischen den Zuständen einstellt, in dem jeweils beide Zustände mit unterschiedlicher Wahrscheinlichkeit auftreten. Dieser Übergangsbereich des einzelnen Grundelements ist in der folgenden Abbildung als kritischer Bereich bezeichnet.

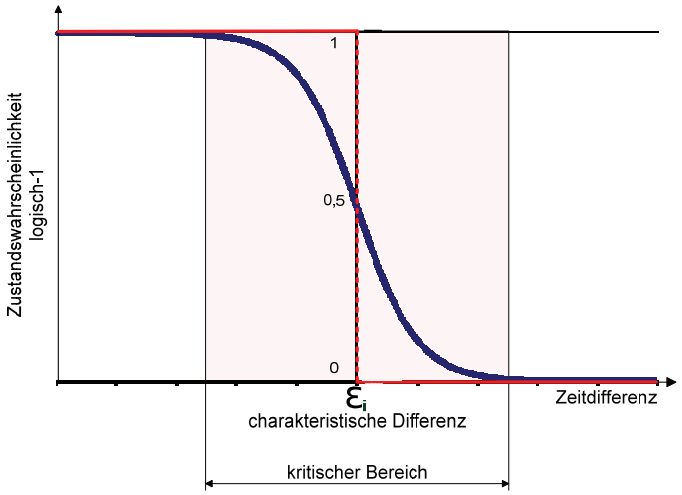


Abbildung 44: Zeitverhalten des Grundelements

Abbildung 44 verdeutlicht diesen Sachverhalt. Die rote Linie stellt die theoretische Wahrscheinlichkeit für den Zustand logisch-1 am Grundelement in Abhängigkeit von der Größe des gemessenen Zeitintervalls dar. In Blau ist das stochastische Verhalten des realen Grundelementes innerhalb des kritischen Bereiches dargestellt.

Außerhalb des kritischen Bereiches liegt zwischen den Signaländerungen an beiden Eingängen des FlipFlops innerhalb des Grundelements ein genügend großer Zeitabstand, sodass Setup- und Hold-Zeiten des FlipFlops vollständig eingehalten werden. Je dichter beide Signaländerungen zusammenliegen, desto häufiger und gravierender werden die Setup- und Hold-Zeiten verletzt. Dadurch entsteht im Zusammenspiel mit zufälligen Ungenauigkeiten der Effekt, dass ein fehlerhafter Zustand eingenommen wird. Ändern sich beide Signale am Eingang des FlipFlops gleichzeitig, ist die Wahrscheinlichkeit für beide möglichen Ausgangszustände gleich groß. Per Definition entspricht die charakteristische Differenz ε_i des Grundelements somit der Zeitdifferenz Δt, deren wiederholte Messung eine 50%-ige Auftretenswahrscheinlichkeit für beide möglichen Ausgangszustände aufweist.

Die Auswertung von Messergebnissen erlaubt die genauere quantitative Beschreibung des kritischen Bereichs am Grundelement. Die folgende Abbildung zeigt die Wahrscheinlichkeit für den Eintritt eines Grundelementes in den Zustand logisch-1 in Abhängigkeit von der gemessenen Zeitdifferenz. Dafür wurde ein reales Grundelement in einer BOUNCE-Variante ausgewählt, dessen kritischer Bereich während einer Messreihe mit unterschiedlichen Zeitdifferenzen vollständig durchlaufen wurde.

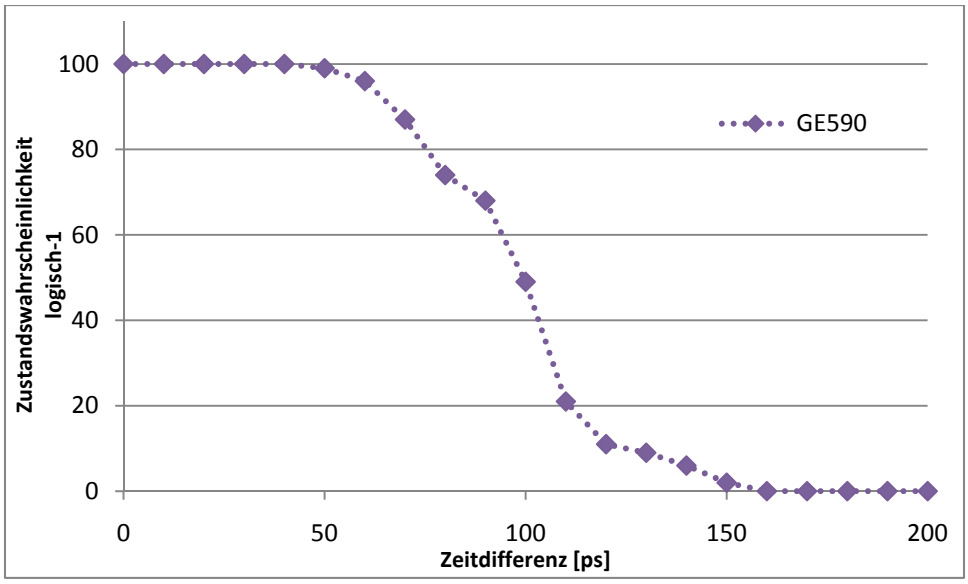


Abbildung 45: Zeitverhalten am einzelnen Grundelement (StratixII-FPGA)

Deutlich zu erkennen ist, dass die Wahrscheinlichkeit für den Zustand logisch-1 ab einer Zeitdifferenz von 40 ps stetig abnimmt, bevor sie ab einer Zeitdifferenz von 160 ps dauerhaft den Wert logisch-0 einnimmt. Die Breite des kritischen Bereiches umfasst somit 120 ps. Die Messung der Zeitdifferenz 100 ps zeigt eine Wahrscheinlichkeit für den Zustand logisch-1 von 50%. Nach den obigen Ausführungen entsprechen 100 ps somit der charakteristischen Differenz des hier dargestellten Grundelementes. Neben seinen Index, in diesem Fall 590, beschreiben die charakteristische Differenz ε_{590}=100 ps und die Breite des kritischen Bereiches von 120 ps die Eigenschaften dieses Grundelements vollständig.

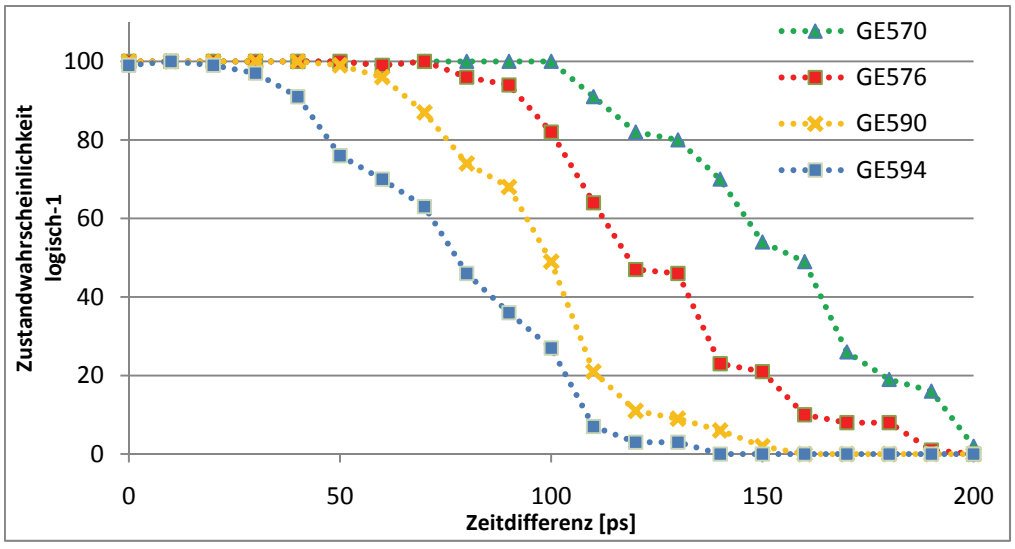


Abbildung 46: Wahrscheinlichkeit für das Auftreten des Zustandes logisch-1 in Abhängigkeit von der Zeitdifferenz für vier ausgewählte Grundelemente

Abbildung 46 ergänzt die vorhergehende Darstellung um die Messergebnisse für drei weitere Grundelemente. Hier handelt es sich um vier Vertreter aus einer Gruppe von 24

Grundelementen, die alle eine charakteristische Differenz zwischen ca. 75 ps und 160 ps aufweisen. Zur besseren Darstellung sind lediglich vier Vertreter dieser Gruppe eingezeichnet. Allerdings sind bereits diese vier Vertreter ausreichend, um die Ähnlichkeiten des Wahrscheinlichkeitsverlaufs unterschiedlicher Grundelemente zu erkennen. Aufgrund dieser Ähnlichkeit der zeitlichen Eigenschaften aller Grundelemente lässt sich der gemittelte Verlauf der Wahrscheinlichkeit für den Zustand logisch-1 in Abhängigkeit des Abstandes der gemessenen Zeitdifferenz von der charakteristischen Differenz ε_i eines Grundelementes berechnen. Der resultierende Verlauf ist in Abbildung 47 dargestellt. Diese Normkurve wurde anhand der gemessenen Eigenschaften von 18 Grundelementen berechnet. Die Fehlerindikatoren an den einzelnen Messpunkten stellen den jeweils größten und kleinsten Wert dar, der in die Mittelwertberechnung mit eingeflossen ist.

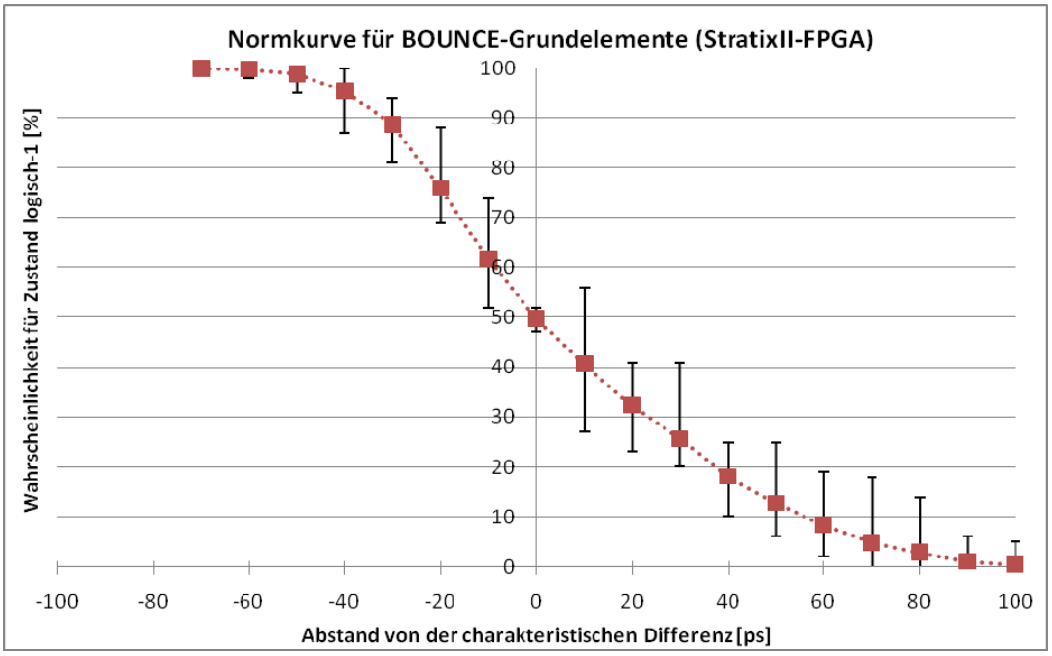


Abbildung 47: errechnete Normkurve des Zeitverhaltens eines Grundelementes (anhand der StratixII-Variante mit 1024 Grundelementen)

Diese Normkurve stellt zugleich eine Möglichkeit für die Bestimmung der Größe einer gemessenen, unbekannten Zeitdifferenz dar. Ist aus einer Mehrfachmessung die Häufigkeit des Zustandes logisch-1 bekannt, lässt sich der Abstand der gemessenen Zeitdifferenz zur charakteristischen Differenz bestimmen. Bedingung dafür ist, dass die charakteristischen Differenzen aller Grundelemente bekannt sind. In die Auswertung fließen alle Messpunkte ein, für die ein eindeutiger Wert aus der Normkurve entnommen werden kann. Das sind die Elemente, die während der Messungen eine Wahrscheinlichkeit 95%>p>10% für den Zustand logisch-1 aufweisen. Detailliert wird diese Möglichkeit der Auswertung noch einmal im Abschnitt zur Rekonstruktion der Zeitdifferenz diskutiert.

10.2. BOUNCE: Verbund der Grundelemente

Der Verbund aller Grundelemente bildet die BOUNCE-Architektur. Das zeitabhängige Verhalten des einzelnen Grundelementes wurde bereits im vorhergehenden Abschnitt dargestellt. Die Grundelemente teilen sich bei einer praktischen Messung auf drei Gruppen auf. Eine Gruppe beinhaltet alle Elemente mit dem stabilen Statuswert logisch-1, alle Grundelemente mit dem stabilen Statuswert logisch-0 bilden die zweite Gruppe. In der dritten Gruppe liegen alle Elemente, die sich innerhalb ihres kritischen Bereiches befinden. Insbesondere in dem Fall, dass eine einzige Zeitdifferenz mehrfach gemessen wird, lassen sich diese Bereiche in den Messdaten deutlich identifizieren. Abbildung 48 zeigt einen vollständigen Datensatz, der aus 100 Einzelmessungen der BOUNCE-Architektur mit 1024 Grundelementen in 8 Reihen eines StratixII-FPGA gewonnen wurde. Alle folgenden Ausführungen beziehen sich auf diese Umsetzung der BOUNCE-Architektur.

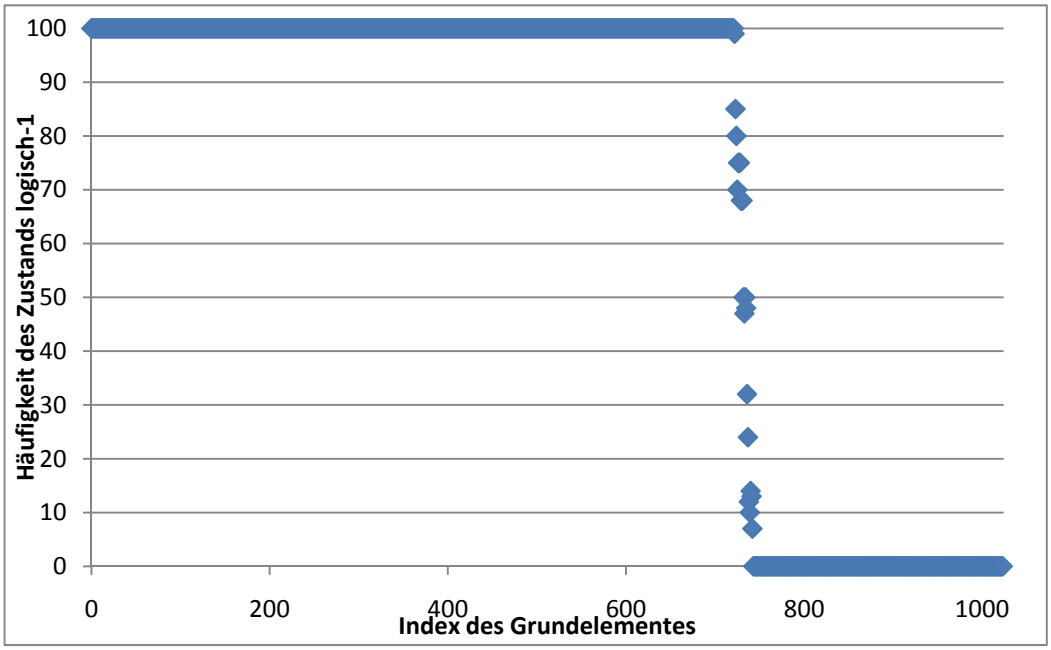


Abbildung 48: vollständiger Datensatz für die BOUNCE-Variante mit 1024 Grundelementen, aufgenommen aus 100 Einzelmessungen einer konstanten Zeitdifferenz

Zur deutlicheren Darstellung zeigt Abbildung 49 den Ausschnitt aus Abbildung 48, der die Grundelemente enthält, die sich bei den Messungen innerhalb ihres jeweiligen kritischen Bereichs befanden. Gleichzeitig sind die drei angesprochenen Bereiche eingezeichnet. Die Grenzen zwischen diesen Bereichen verschieben sich in Abhängigkeit von der Größe der gemessenen Zeitdifferenz. Dies lässt sich am einfachsten mit einem exemplarisch gewähltem Grundelement erklären, welches in Abbildung 49 rot eingekreist ist.

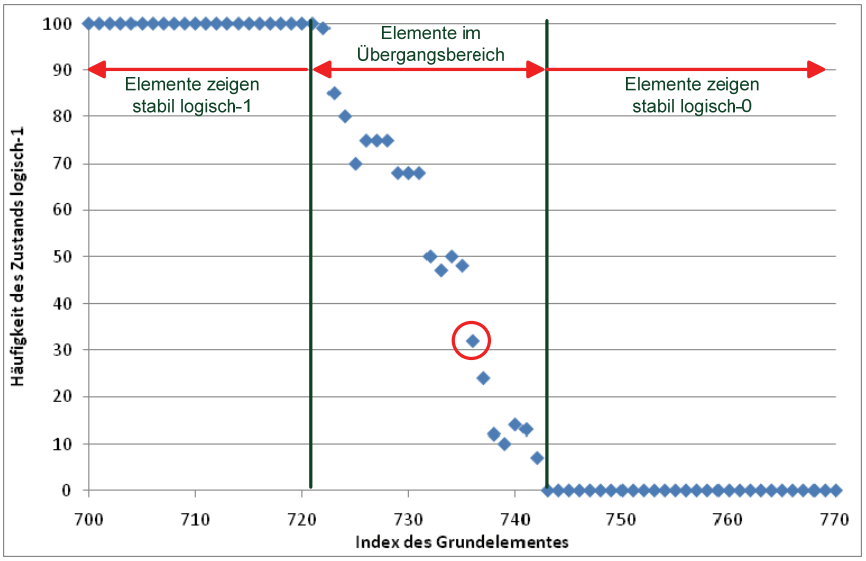


Abbildung 49: Detaillierte Ansicht des Übergangsbereichs, jeder Datenpunkt repräsentiert ein Grundelement

Im dargestellten Datensatz weist das markierte Grundelement offensichtlich eine Wahrscheinlichkeit für den Zustand logisch-1 von ungefähr 30% auf. Im Vergleich mit der Normkurve der Grundelemente in Abbildung 47 ergibt sich, dass die gemessene Zeitdifferenz ungefähr 20 ps größer als die charakteristische Differenz des Grundelementes sein muss. Würde in anderen Messungen eine größere Zeitdifferenz eingestellt werden, würde das Grundelement eine geringere Häufigkeit für den Zustand logisch-1 aufweisen. Im umgekehrten Fall, also einer kleineren Zeitdifferenz, würde die Häufigkeit für den Zustand logisch-1 steigen. Überträgt man diese Überlegungen auf alle Grundelemente, zeigt sich, dass eine Vergrößerung der gemessenen Zeitdifferenz eine Verschiebung des gesamten Übergangsbereichs nach links bewirkt. Eine Verkleinerung der gemessenen Zeitdifferenz verschiebt diesen Übergangsbereich nach rechts. Abbildung 50 verdeutlicht diese Aussagen am Beispiel von drei gemessenen Zeitdifferenzen.

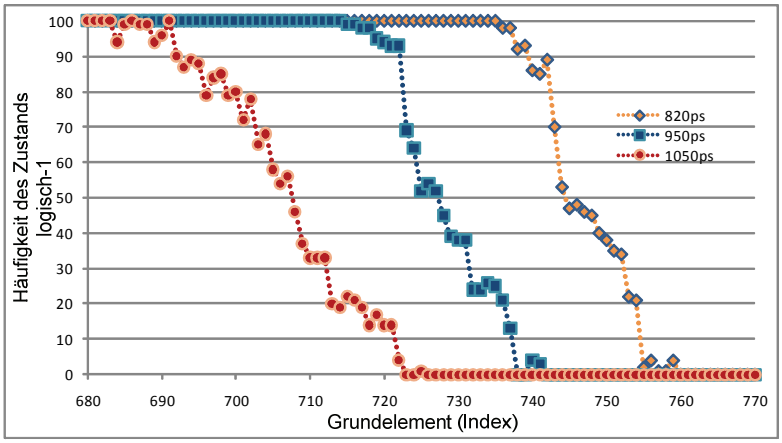


Abbildung 50: unterschiedliche Zeitdifferenzen verschieben den Umschaltbereich

Der Übergangsbereich ist somit für die Bestimmung der gemessenen Zeitdifferenz von essentieller Bedeutung. Der einfachste Ansatz für eine Auswertung ist es, die Anzahl der in einem Datensatz enthaltenen Zustände logisch-1 zu zählen. Da sich der Übergangsbereich in Abhängigkeit von der gemessenen Zeitdifferenz verschiebt, ändert sich auch die Anzahl der insgesamt im Datensatz befindlichen Zustände logisch-1. Während einer Kalibrierungsphase wird einmal festgestellt, welche Zeitdifferenz welche Anzahl der Zustände logisch-1 im Datensatz zur Folge hat. Vorzugsweise wird dazu jeder Kalibrierungspunkt mehrfach gemessen. Die Kalibrierungspunkte sind dabei gleichmäßig über den Messbereich verteilt.

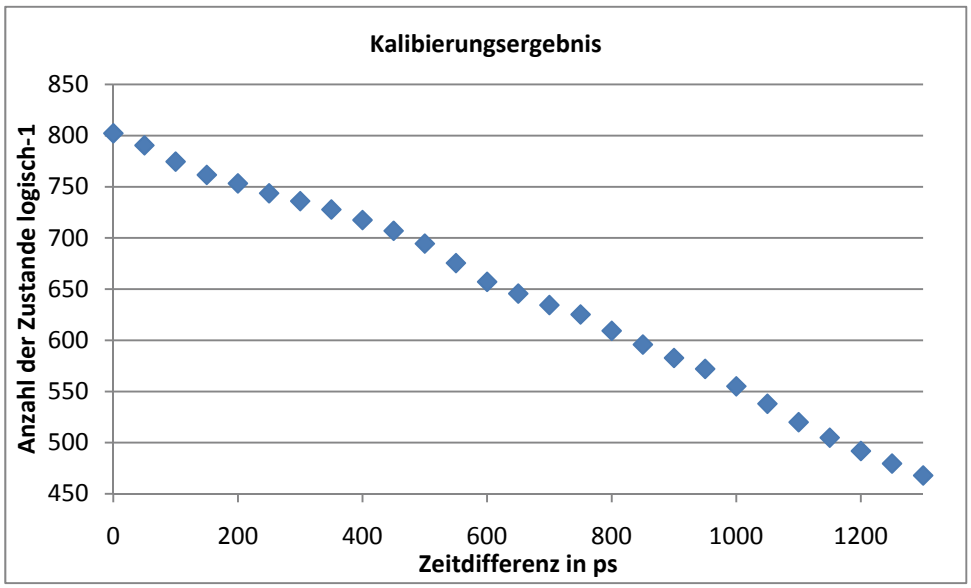


Abbildung 51: Kalibrierungsergebnis der BOUNCE-Variante mit 1024 Grundelementen in 8 parallelen Reihen; je nach Größe der gemessenen Zeitdifferenz zeigen unterschiedlich viele Grundelemente den Zustand logisch-1

Abbildung 51 zeigt das Ergebnis eines Kalibrierungsvorgangs. An jedem Kalibrierungspunkt wurden 100 Messungen durchgeführt. Der Abstand der Kalibrierungspunkte wurde auf 50 ps (entspricht einer Längenänderung von 1 cm am Line Stretcher) festgelegt. Deutlich ist zu erkennen, dass die Anzahl der in einem Datensatz befindlichen Zustände logisch-1 abhängig von der eingestellten Zeitdifferenz sind. Zufällig ergibt sich bei der verwendeten Variante der BOUNCE-Architektur ein nahezu linearer Zusammenhang zwischen der Häufigkeit des Zustands logisch-1 und der Größe der Zeitdifferenz.

Die in der Grafik dargestellten Ergebnisse werden in der Auswertesoftware gesichert. Während der Messung unbekannter Zeitdifferenzen wird das jeweilige Ergebnis mit diesen Punkten verglichen. Stimmt die ermittelte Anzahl der Zustände logisch-1 im Datensatz nicht exakt mit einem Kalibrierungspunkt überein, wird der Raum zwischen zwei Kalibrierungs-

punkten linear interpoliert und so die gemessene Zeitdifferenz bestimmt. Diese Näherung wird in Kapitel 11.4 noch einmal gesondert diskutiert.

10.2.1. Bestimmung der zeitlichen Auflösung der BOUNCE-Architektur

Aus dem Kalibrierungsdatensatz (siehe Abbildung 51) lässt sich die zeitliche Auflösung der BOUNCE-Architektur bestimmen. Dabei gilt diese Auflösung, genau wie der Kalibrierungsdatensatz, nur für die hier verwendete Variante der BOUNCE-Architektur und für den eingesetzten Versuchsaufbau.

Die Auflösung eines Messsystems gibt an, wie groß der minimale Abstand unterschiedlicher Messpunkte einer kontinuierlichen Größe ist, damit diese beiden Messwerte noch als unterschiedlich erkannt werden. Bei der BOUNCE-Architektur basiert die Rücküberführung des Messergebnisses in einen Zeitwert bekanntlich auf der im Datensatz enthaltenen Anzahl der Zustände logisch-1. Der kleinste mögliche Unterschied im Ergebnisvektor zweier Messpunkte beträgt genau 1. Dies ist dann der Fall, wenn die Anzahl der Zustände logisch-1 in zwei Datensätzen sich genau um diesen Wert 1 unterscheidet. Daher ist es sinnvoll, den zeitlichen Abstand zweier Kalibrierungspunkte auf die Differenz ihrer Ergebniswerte zu beziehen und diesen Quotienten als Auflösung zu betrachten. Beispielsweise zeigt der Kalibrierungsdatensatz, dass für die Zeitdifferenz 0 ps im Durchschnitt 802-mal der Zustand logisch-1 im Ergebnis auftritt. Für die Zeitdifferenz 50 ps verringert sich diese Zahl auf den Wert 790. Folglich beträgt die Differenz der Ergebniswerte 12. Die BOUNCE-Architektur kann also auf 50 ps zwischen zwölf verschiedenen Zuständen unterscheiden. Einen vereinfachten linearen Zusammenhang zwischen der Anzahl der Zustände logisch-1 vorausgesetzt, liegt der zeitliche Abstand zwischen zwei unterscheidbaren Zeitdifferenzen demzufolge bei 50 ps / 12 = 4,25 ps.

Mit den im Kalibrierungsdatensatz enthaltenen Informationen wird die Auflösung im Folgenden für jeden zwischen benachbarten Kalibrierungspunkten liegenden Bereich separat berechnet. Das Ergebnis für die Auflösung über den gesamten Messbereich ist in Abbildung 52 dargestellt.

Rechnerisch ergibt sich eine mittlere zeitliche Auflösung dieser Variante der BOUNCE-Architektur von 4,2 ps. Diese Auflösung wird allerdings nicht über den gesamten Messbereich erzielt. Dennoch sind alle Auflösungswerte besser als 7 ps. An dieser Stelle wird nochmals darauf hingewiesen, dass die Zeitwerte auf der mit 2/3 c angenommenen Ausbreitungsgeschwindigkeit der Signale im verwendeten, ausziehbaren Koaxialrohr basieren.

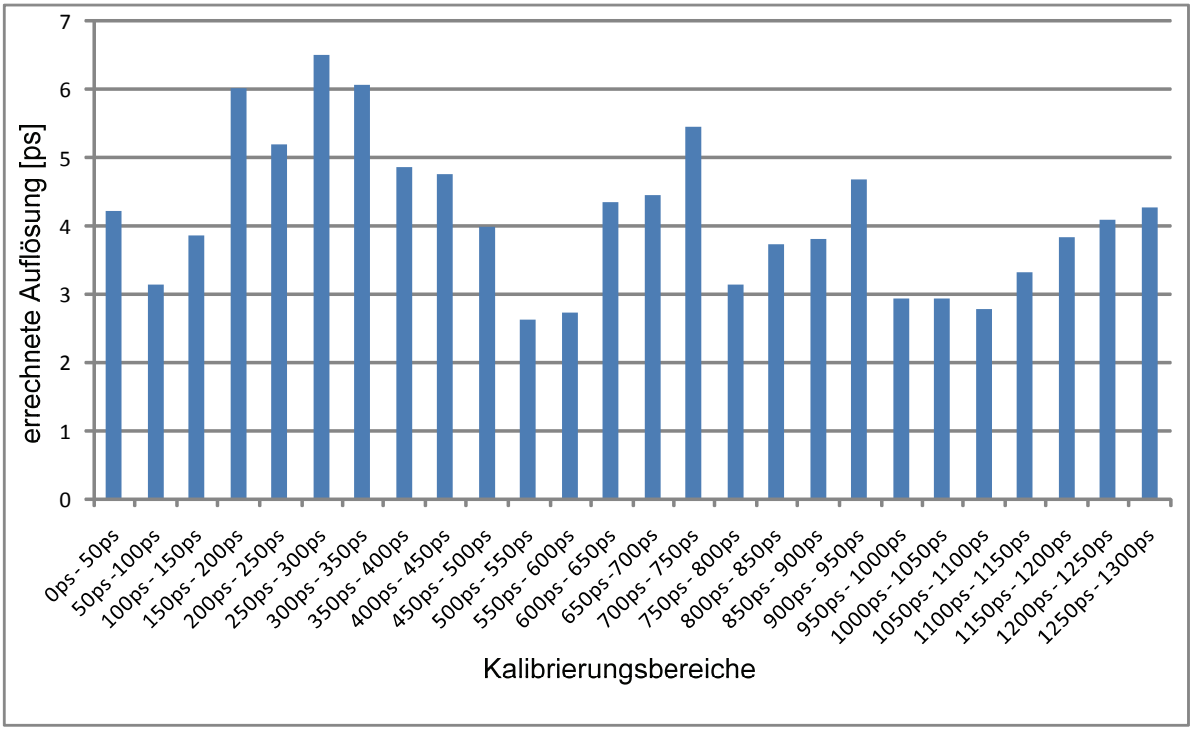


Abbildung 52: errechnete Auflösung für die BOUNCE-Variante mit 1024 Grundelementen in den einzelnen Bereichen zwischen den Kalibrierungspunkten

10.2.2. Bestimmung der zeitlichen Genauigkeit der BOUNCE-Architektur

Die Genauigkeit eines Messsystems gibt an, mit welcher Zuverlässigkeit die Messergebnisse bestimmt werden können. Dieser Wert ist unabhängig von der Größe der Auflösung, da die Auflösung nur die technischen Möglichkeiten des Messsystems berücksichtigt.

Die Angabe der Genauigkeit hängt signifikant vom verwendeten Messverfahren ab. Basiert die Bestimmung des Ergebnisses auf nur einer einzigen Messung, sind geringe Genauigkeiten zu erwarten. Dies ist dadurch begründet, dass die Grundelemente in ihrem kritischen Bereich ein stochastisches Verhalten aufweisen, wie bereits ausführlich in Abschnitt 10.1 diskutiert wurde. Wie in Abbildung 50 dargestellt, befinden sich an einem Messpunkt immer ca. 20 bis 30 Grundelemente in ihrem kritischen Bereich. Das wiederum bedeutet, dass in Extremfällen sich die Anzahl der im Datensatz enthaltenen Zustände logisch-1 zwischen zwei Messungen an ein und demselben Punkt um 30 unterscheidet. Mit der im vorhergehenden Abschnitt angegebenen mittleren Auflösung von ca. 5 ps errechnet sich die erreichbare mittlere Genauigkeit der Einzelmessung zu +/-75 ps. Durchgeführte Messungen bestätigen diese Größenordnung, wie Abbildung 53 verdeutlicht.

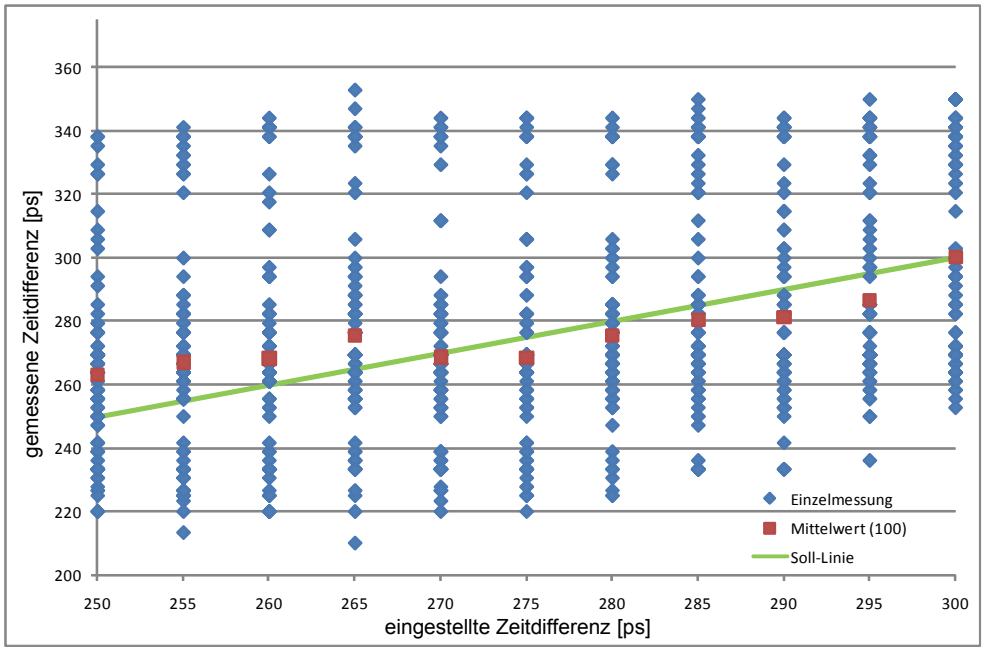


Abbildung 53: Verteilung der berechneten Zeitdifferenzen an verschiedenen Messpunkten (grün-Ideallinie, blau-Einzelmessung, rot-Mittelwert aus 100 Messungen)

Die Messpunkte, an denen jeweils 100 Einzelmessungen durchgeführt wurden, haben einen zeitlichen Abstand von 5 ps (entspricht einer Längenänderung von 1 mm am Line Stretcher). Die grüne Linie stellt die Ideallinie dar, bei der der Sollwert mit dem Istwert identisch ist. Die blauen Punkte markieren die aus jeweils einer Einzelmessung errechneten Zeitwerte. Dabei ist zu beachten, dass mehrfach errechnete Ergebniswerte übereinanderliegen. Auffallend ist, dass die Abweichungen hin zu größeren Zeitwerten deutlicher ausfallen als die Abweichungen hin zu geringeren Zeitwerten. Dies deckt sich mit der Darstellung der Zustandswahrscheinlichkeit eines Grundelementes in Abbildung 47. Auch diese ist nicht exakt symmetrisch zur charakteristischen Differenz des Grundelementes. Vielmehr ist die Wahrscheinlichkeit einen „fehlerhaften“ Zustand einzunehmen für Zeitdifferenzen oberhalb der charakteristischen Differenz größer als für Zeitdifferenzen unterhalb der charakteristischen Differenz. Erkennbar ist dies am flacheren Verlauf der Kurve im Bereich für Zeitdifferenzen oberhalb der charakteristischen Differenz in der angesprochenen Abbildung.

Die rot markierten Punkte in Abbildung 53 stellen den Mittelwert aus allen 100 zum jeweiligen Messpunkt gehörenden Ergebniswerten dar. Deutlich zu erkennen ist, dass diese wesentlich dichter an der Idealkurve liegen als viele Ergebnisse der Einzelmessungen. Die Mittelwertbildung gleicht die stochastischen Eigenschaften der Grundelemente aus. Aus diesem Grund wurden weitere Messreihen aufgenommen, die sich über den gesamten Messbereich verteilten. Der zeitliche Abstand zwischen zwei Messpunkten betrug dabei 25 ps.

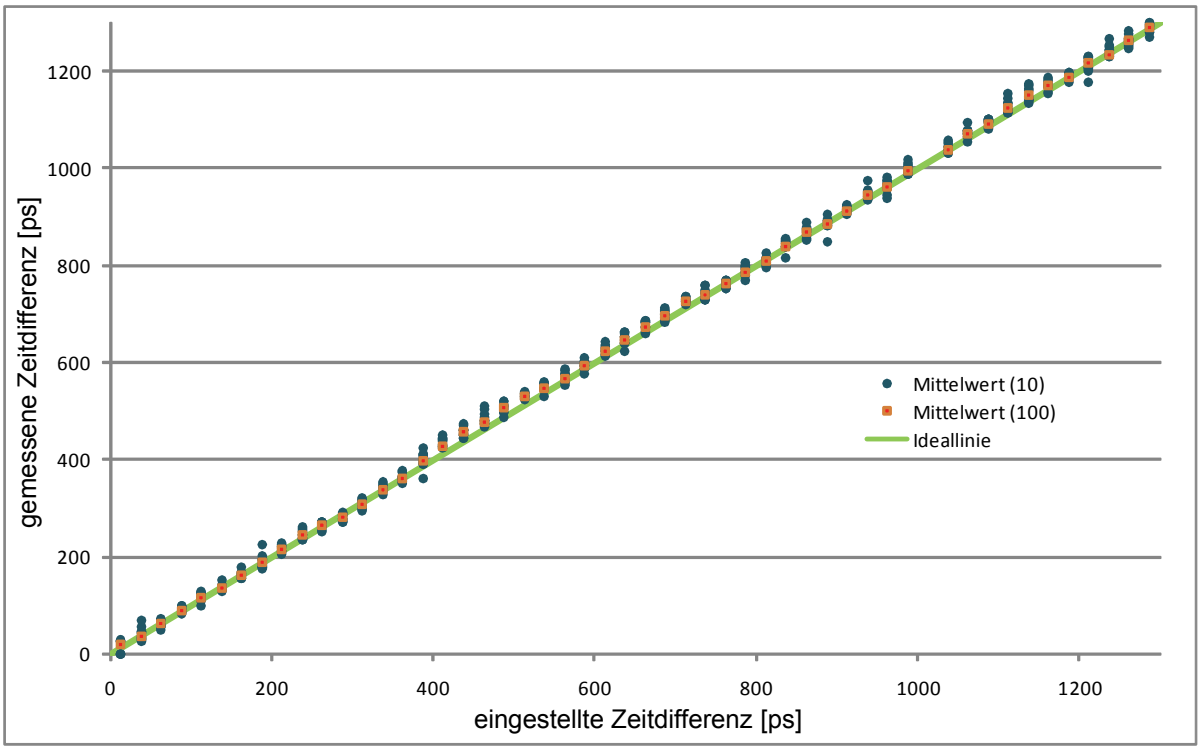


Abbildung 54: Genauigkeit über den gesamten Messbereich

Für alle Messungen gilt, dass sie in der Nähe der Ideallinie liegen. Auch zeigt sich, dass jeder Messpunkt bei der Mittelwertbildung aus 100 Einzelmessungen zu einem eindeutigen Ergebniswert führt. Um die Fehlerwerte an den einzelnen Messpunkten besser erkennen zu können, sind diese in Abbildung 55 separat dargestellt. Der Fehler für berechnete Zeitdifferenzen, basierend auf dem Mittelwert von 10 Einzelmessungen (Mw10 in Abbildung 55), liegt im Regelfall zwischen -20 ps und +40 ps. Ausnahmen, die außerhalb dieses Bereiches liegen, treten nur vereinzelt auf. Basiert die Bestimmung der gemessenen Zeitdifferenz auf 100 Messungen, erhöht sich die Genauigkeit signifikant. Die Mehrzahl dieser Messergebnisse zeigt eine Genauigkeit von +/-10 ps. Ungefähr 20 Prozent der Messergebnisse erreichen nur eine Genauigkeit zwischen +10 ps und +20 ps.

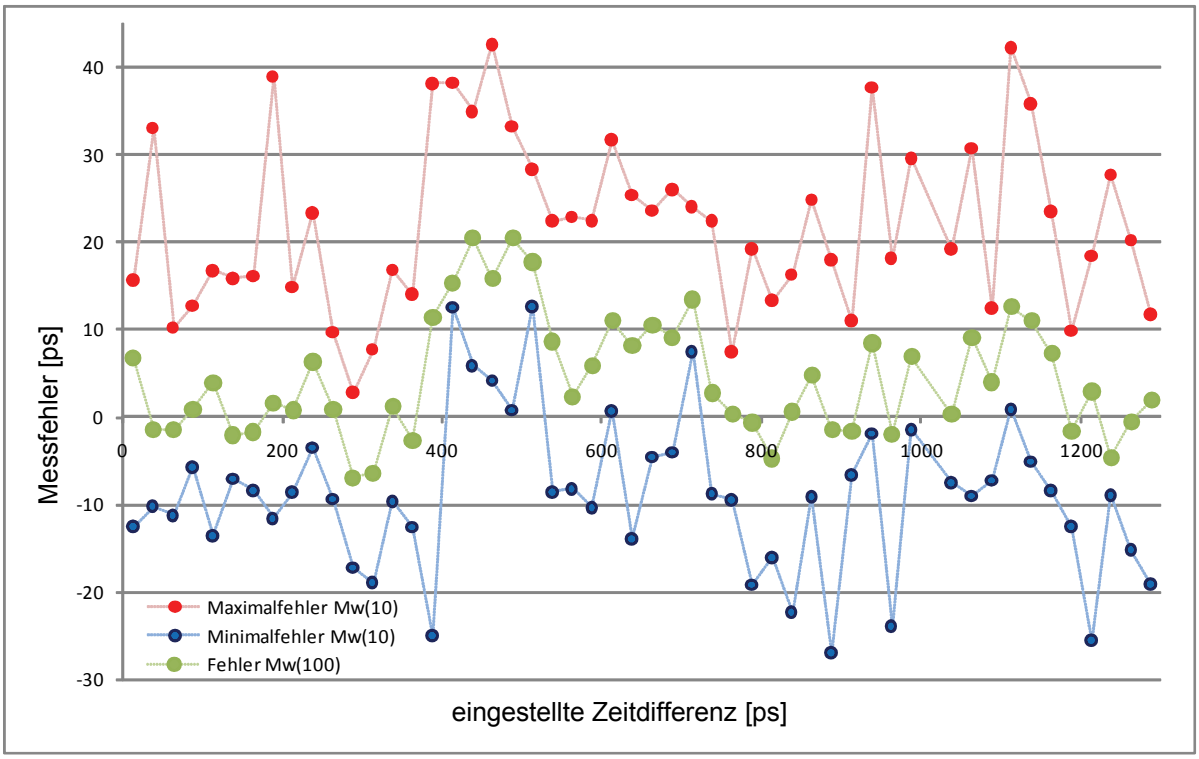


Abbildung 55: Fehlerwerte der Mehrfachmessungen

Zusammenfassend zeigt die hier verwendete Variante der BOUNCE-Architektur mit 1024 gleichmäßig auf 8 parallele Reihen verteilten Grundelementen eine zeitliche Auflösung besser als 7 ps bei einer Genauigkeit von +/-20 ps bei der Berechnung des Ergebniswertes aus 100 Einzelmessungen. Die Genauigkeit verringert sich, wenn weniger Messdaten zur Ergebnisberechnung genutzt werden. Die Einzelmessung erreicht nur noch eine Genauigkeit von +/-75 ps. Das folgende Kapitel diskutiert die erreichten Ergebnisse und präsentiert Ansätze zur weiteren Verbesserung der bisher dargestellten Resultate.

11. Diskussion

Die Diskussion der erreichten Ergebnisse konzentriert sich auf drei Aspekte. Als erstes werden die erreichten Ergebnisse mit den im Abschnitt zum Stand der Forschung (Kapitel 5) erwähnten Arbeiten verglichen. Zweitens wird der Einfluss der genutzten BOUNCE-Architektur und der verwendeten Messschaltung auf die im vorhergehenden Kapitel dargestellten Ergebnisse betrachtet. Der dritte Teil der Ausführungen behandelt Möglichkeiten und Grenzen der weiteren Erhöhung der zeitlichen Auflösung des Messsystems.

11.1. Vergleich mit den Tapped Delay Lines

Die vorliegenden Ergebnisse der BOUNCE-Architektur leisten einen signifikanten Beitrag zur FPGA-basierten Zeitmessung. Das technologisch am engsten verwandte Fachgebiet ist daher die Zeitmessung mittels FPGA-basierter Tapped Delay Lines. Das grundlegende Konzept dieser Schaltungsart wurde bereits in Kapitel 3.4.4 dargestellt. Eine ausführliche Beschreibung der technischen Realisierung der Tapped Delay Lines befindet sich in Abschnitt 5 dieser Arbeit.

Ziel aller Zeitmesssysteme ist das Erreichen einer hohen zeitlichen Auflösung unter der Vorgabe, gleichzeitig eine angemessene Genauigkeit zu erzielen. Der Terminus „angemessen“ bezieht sich dabei auf die im jeweiligen Einsatzgebiet gestellten Anforderungen an das System. Dabei ist zu unterscheiden, ob die zu bestimmende Zeitdifferenz mehrfach detektiert werden kann, oder es sich um ein einmaliges Ereigniss handelt. Die Einzelmessung stellt die höchsten Anforderungen an das Messsystem, da hier keine statistische Verfahren, wie z.B. Mittelwertbildung aus mehreren Messergebnissen, eingesetzt werden können.

Die Realisierung der Tapped Delay Lines mit der aktuell höchsten dokumentierten Auflösung basiert auf der einfachen Tapped Delay Line unter Verwendung der FPGA-internen Carry-Pfade (siehe Abschnitt 4.3 ff.). Die erreichbare Auflösung FPGA-basierter Tapped Delay Lines liegt gegenwärtig bei 50 ps. Somit stellt die BOUNCE-Architektur mit ihrer mittleren Auflösung von unter 5 ps eine Verbesserung der zeitlichen Auflösung um nahezu eine Größenordnung dar. Gleichzeitig basiert aber die BOUNCE-Architektur auf derselben, günstigen FPGA-Technologie wie die Tapped Delay Lines. Die Auflösungsverbesserung um eine Größenordnung wird ausschließlich durch die neuartige Verwendung der FPGA-internen Signalpfade als Verzögerungselemente erreicht. Somit hängt die zeitliche Auflösung auch nicht mehr von der technologieabhängigen Realisierung der FPGA-internen Logikelemente ab.

Ein weiterer, entscheidender Vorteil der BOUNCE-Architektur, welcher bereits in Kapitel 6.2 kurz angesprochen wurde, ist die Tatsache, dass BOUNCE keine definierte Reihenfolge des Auftretens beider Ereignissignale voraussetzt. Die einfache Tapped Delay Line aus Abschnitt 5.3.2 kann aufgrund der Struktur der FPGA-Logik-Elemente nur ein Signal einer Verzögerung unterziehen, während das zweite Signal den aktuellen Status aller Verzögerungselemente gleichzeitig sichert. Das zweite Signal muss somit zwangsweise nach dem ersten Signal auftreten. Betrachtet man das Beispiel der laufzeitbasierten Winkelmessung mit zwei Empfängern RX_1 und RX_2 in Abschnitt 3.1, Abbildung 7, ist offensichtlich, dass die Reihenfolge des Auftretens des Empfangssignals an beiden Empfängern nicht feststeht. Die Reihenfolge ist vielmehr vom eingestellten Winkel abhängig. Die Messung des Laufzeitunterschieds in diesem Szenario ist mit der einfachen FPGA-implementierten Tapped Delay Line somit nicht möglich. Die BOUNCE-Architektur hingegen funktioniert unabhängig von der Signalreihenfolge. Der Grund dafür liegt in der Ähnlichkeit zu den gegenläufigen, differentiellen Tapped Delay Lines. Diese lassen sich in FPGAs allerdings nur schwierig und unter erhöhtem Ressourcenaufwand realisieren (siehe Abschnitt 5.2). Die BOUNCE-Architektur zeichnet sich dadurch aus, dass ihre atomares Modul, das Grundelement, lediglich ein Logikelement des FPGAs belegt und damit keine Erhöhung der benötigten Ressourcen im Vergleich zu einfachen Delay Line erfolgt. Die BOUNCE-Architektur vereint also die schaltungstechnische Schlichtheit der einfachen Tapped Delay Lines mit der erweiterten Funktionalität der gegenläufigen, differentiellen Tapped Delay Lines.

Ein ganz besonders hervorzuhebender Beitrag dieser Arbeit zum Stand der Forschung ist der Nachweis, dass sich passive, interne Signalleitungen als Verzögerungselemente eignen. Bis zum heutigen Zeitpunkt ist kein anderes System bekannt, was diese Eigenschaften der Signalleitungen zur Zeitmessung einsetzt. Vielmehr ist ein Bestreben in der Forschung zu erkennen, die Auflösung der Tapped Delay Lines durch immer neue Arten dedizierter Verzögerungselemente mit kleineren Verzögerungszeiten zu erhöhen. BOUNCE stellt somit einen revolutionären Ansatz zur Zeitmessung auf Basis von Signalverzögerungen dar.

Bisher wurden lediglich die FPGA-basierten Tapped Delay Line Implementationen zur Einordnung der erreichten Ergebnisse herangezogen. Doch auch im Vergleich mit den ASIC-basierten Tapped Delay Lines erreicht die BOUNCE-Architektur sehr gute Werte. Die in Abschnitt 5.3.1 vorgestellten Laufzeitmessysteme erreichen zeitliche Auflösungen zwischen 10ps und 30ps [21] [22] [43]. Selbst im Hinblick auf diese Ergebnisse bedeutet die BOUNCE-Architektur eine Verbesserung der zeitlichen Auflösung um den Faktor zwei. Die Bedeutung dieses Ergebnis liegt in der Tatsache, dass die BOUNCE-Architektur dabei auf Standard-FPGA-Technologie aufbaut damit und die kosten- und arbeitsintensiven Prozesse der

kundenwunschbasierten Schaltkreisentwicklung umgeht. In ihrer Konsequenz erreicht die BOUCNE-Architektur damit Auflösungsbereiche, die bisher den Zeitmesssystemen mit integrierten Analog-Komponenten vorbehalten waren. Diese Systeme wurden in Abschnitt 3.4 vorgestellt, wo ebenfalls auf die technologischen Schwierigkeiten dieser Systeme eingegangen wurde, die bei der BOUNCE-Architektur gar nicht erst auftreten.

Um einen Vergleich der durch die BOUNCE-Architektur erreichten Genauigkeiten mit denen anderer Zeitmesssysteme zu ermöglichen, sind detaillierte Betrachtungen der Umstände notwendig, die zu diesen Ergebnissen geführt haben. Aus diesem Grund wird der Aspekt der Genauigkeit im folgenden Abschnitt separat behandelt.

11.2. Einfluss der Messschaltung auf die Genauigkeit

Die in Kapitel 10 präsentierten Ergebnisse der BOUNCE-Architektur lassen erkennen, dass trotz der hervorragenden Ergebnisse in Bezug auf die zeitliche Auflösung die erzielte Genauigkeit wesentlich schlechter ist. Insbesondere die Angabe der besonders aussagekräftigen Genauigkeit der Einzelmessung ist kritisch zu betrachten. Ihr Wert ist um mehr als eine Größenordnung schlechter als die Angabe zur zeitlichen Auflösung des Systems. Unter diesen Umständen sind die guten Auflösungseigenschaften in der Praxis nicht nutzbar. Allein die statistische Auswertung von mehreren Messungen ein und derselben Zeitdifferenz lassen die Genauigkeitswerte akzeptable Regionen erreichen. Die wesentliche Frage ist, ob es sich bei diesem Verhalten um ein systeminternes Problem der BOUNCE-Architektur handelt, oder äußere Umstände verantwortlich sind.

Im Kapitel 10 zu den Ergebnissen dieser Arbeit wurde eine Darstellung gezeigt, die das stochastische Verhalten der Grundelemente beinhaltet (Abbildung 47). Dort ist zu erkennen, dass z.B. eine Zeitdifferenz Δt, welche um 70 ps größer als die charakteristische Differenz ε_i des Grundelementes ist, eine fünfprozentige Wahrscheinlichkeit aufweist, den Zustand logisch-1 einzunehmen. Nach den in Abschnitt 6.2 postulierten Gleichungen 2.33 und 2.34 müsste diese Wahrscheinlichkeit aber für alle Zeitdifferenzen größer als ε_i identisch null sein. Mit anderen Worten, die Kurve in Abbildung 47 müsste wesentlich schmaler ausfallen, idealerweise unendlich schmal.

Das die Grundelemente eine abweichendes Verhalten zeigen, liegt mit hoher Wahrscheinlichkeit an der externen Signalgenerierung, aus der beide im Messsystem verwendeten Signale hervorgehen. Wie bereits in Abschnitt 9.2 dargestellt, erzeugt ein Philips Pulse Generator PM5786B das Grundsignal, welches anschließend über einen Mini-circuits Power Splitter in zwei verschiedene Signalpfade eingespeist wird. Ein Signalpfad ist durch das in der Länge variierbare Koaxialrohr gekennzeichnet, mit dessen Hilfe unterschiedliche Zeitdifferenzen erzeugt werden. Der entscheidende Punkt ist, dass der

Pulsgenerator nur eine äußerst begrenzte Flankensteilheit besitzt. Diese beträgt, mit einem Philips PM3392A Combiscope gemessen, 20 ns am Eingang des FPGAs. Dass bedeutet, dass jedes der beiden Eingangssignale 20 ns benötigt, um vom 0 V-Pegel auf 2,8 V anzusteigen. Die Eingangstreiber des FPGAs erkennen ab einem Spannungspegel von 2,8 V den Zustand logisch-1. Die folgende Abbildung zeigt das beschriebene Verhalten am Stop-Eingang des FPGAs.

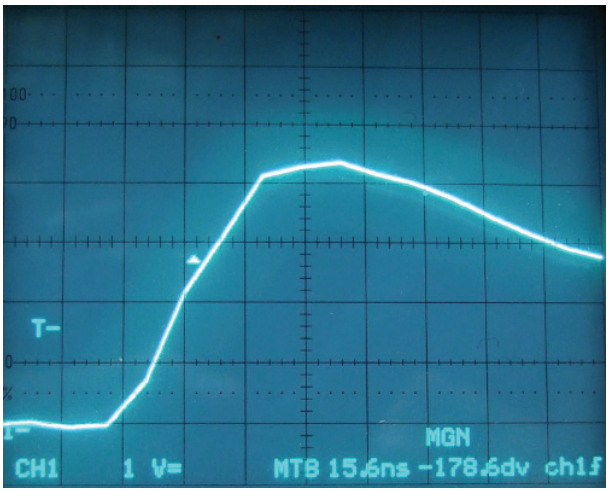


Abbildung 56: Signalflanke am Stop-Eingang des FPGAs

Rechnerisch ergibt sich also ein Anstieg der Spannung von 140 mV/ns. Das bedeutet, wenn die Eingangstreiber den Zustand logisch-1 mit einer Genauigkeit 2,8 V +/- 15 mV detektieren können, führt allein diese Ungenauigkeit schon zu einer zeitlichen Ungenauigkeit von ungefähr +/-100 ps. In der Konsequenz muss also der verwendete Philips Pulse Generator als ungeeignet betrachtet werden. Leider sind bessere Signalquellen nur schwer erhältlich und an der Universität Rostock nicht verfügbar. Ein Picosecond Pulse Labs Pulsgenerator mit einer Flankensteilheit von mehr als 5 V/ns ist zwar am Markt erhältlich, mit einem (erfragten) Preis von ca. 15.000 US$ außerhalb der umsetzbaren Möglichkeiten. Andere Versuche, z.B. durch Einsatz schneller Low-Voltage Differential Signaling (LVDS) Schaltkreise, konnten bis zum Abschluss dieser Arbeit noch nicht erfolgreich durchgeführt werden. Für zukünftige Forschungsarbeiten auf diesem Gebiet sollten die Suche und der Einsatz besser geeigneter Signalgeneratoren allerdings höchste Priorität haben. Eine signifikante Verbesserung der Flankensteilheit der Eingangssignale führt mit hoher Wahrscheinlichkeit zu einem deutlich schmaleren kritischen Bereich der Grundelemente. Die Folge davon wäre eine wesentlich höhere Genauigkeit, da die Wahrscheinlichkeit von falschen Statuswerten am Grundelement schneller vernachlässigbar kleine Werte annimmt.

Nachdem mit dem verwendeten Pulsgenerator eine der naheliegenden Ursachen für die auftretenden Ungenauigkeiten identifiziert ist, sollen im Folgenden weitere Störquellen nur kurz betrachtet werden. Während der durchgeführten Messreihen wurden ständig weitere

Einflüsse auf den Messaufbau untersucht. Einen deutlichen Einfluss hat die mechanische Stabilität des Versuchsaufbaues. Bereits kleinste Änderungen am Aufbau, z.B. veränderte Masseleitungen, Veränderung der Steckverbindungen zwischen Board und Koaxialkabel, führten zu detektierbaren Messabweichungen. Die Begründung dafür liegt in der Eigenschaft der BOUNCE-Architektur, dass die FPGA-interne Schaltung alle äußeren Einflüsse auf das Signalzeitverhalten mit berücksichtigt. Aus diesem Grund ist auch eine Kalibrierung des Messsystems unerlässlich, um quasi virtuelle Fixpunkte zu definieren. Ändern sich die äußeren Eigenschaften, verliert die Kalibrierung somit sofort ihre Gültigkeit, bzw. Messungen mit der ungültigen Kalibrierung führen fehlerhaften Ergebnissen.

Gesondert wurde ebenfalls der Einfluss von Temperaturänderungen auf die BOUNCE-Architektur untersucht. Dafür wurde die Temperatur des FPGAs zwischen 10 °C und 65 °C variiert[5]. Die BOUNCE-Architektur hat in diesem Temperaturbereich keine nennenswerten Abweichungen gezeigt.

11.3. Die Zeitbasis im Messsystem

Alle Angaben von Zeitwerten in den Abschnitten zu den Ergebnissen der BOUNCE-Architektur basieren auf der Annahme einer Signalausbreitungsgeschwindigkeit im koaxialen Line Stretcher von 2/3 c oder $2 \cdot 10^8$ m/s. Somit wird ausgehend von einer Längenänderung des Line Stretcher die Änderung der Signallaufzeit bestimmt. Wie einfach nachzuvollziehen ist, ergibt sich bei einer Längenänderung um einen Zentimeter eine Laufzeitänderung um 50 ps. Die Annahme von 2/3 c ist dadurch begründet, dass die einschlägige Literatur diesen Wert für Koaxialkabel angibt. Gleichzeitig kann dieser Wert mit einfachen Versuchen nachgemessen werden [1].

Ein definitiver Wert für die Ausbreitungsgeschwindigkeit der Signale im Line Stretcher könnte mit hochauflösenden Oszilloskopen ermittelt werden. Die Versuchsdurchführung ist trivial. Ein in den Line Stretcher einlaufendes Signal wird mit dem Oszilloskop aufgenommen. Das Signal wird am gegenüberliegenden, offenen Ende reflektiert und durchläuft den Line Stretcher ein weiteres Mal. Beim Austritt aus dem Line Stretcher wird das Signal mit dem Oszilloskop erneut detektiert. Aus dem zeitlichen Abstand der im Oszilloskop aufgenommenen Signale und der bekannten Wegstrecke durch den Line Stretcher würde sich die Ausbreitungsgeschwindigkeit exakt bestimmen lassen. Da der Line Stretcher allerdings eine maximale Länge von einem Meter besitzt, benötigt man ein Oszilloskop mit einer Auflösung von mindestens $2/3 \cdot 10^{-8}$ s, um überhaupt einen Zeitversatz zwischen Eintritt

[5] Die Temperatur wurde am Gehäuse des FPGAs mit einem handelsüblichen Temperatursensor bestimmt.

und Wiederaustritt des Signals aus dem Line Stretcher zu erkennen. Dies entspricht einer notwendigen Taktfrequenz des Oszilloskops von mindestens $1{,}5 \cdot 10^8$ Hz = 150 MHz. Soll die Ausbreitungsgeschwindigkeit im Line Stretcher mit einer Genauigkeit von 1/10 c bestimmt werden, erhöht sich die notwendige Taktfrequenz bereits auf 1,5 GHz. Ein derartiges Gerät stand für die Versuche im Rahmen dieser Arbeit nicht zur Verfügung.

Eine weitere, einfache Möglichkeit, die Ausbreitungsgeschwindigkeit im Line Stretcher zu bestimmen, oder zumindest abzuschätzen, ist ein Vergleich mit Standard-Koaxialkabel. Die BOUNCE-Architektur ist ein wichtiges Hilfsmittel in diesem Versuch. Im ersten Schritt wird die Signalstrecke mit dem auf minimale Länge eingestellten Line Stretcher verwendet. Die BOUNCE-Architektur bestimmt die Anzahl der Zustände logisch-1 für diese Konfiguration. Anschließend bestimmt BOUNCE die Anzahl der Zustände logisch-1 für den um 25 cm verlängerten Line Stretcher. Die Differenz beider Ergebnisse ist proportional zur Zeitdauer, die das Signal für das Durchlaufen der 25 cm im Line Stretcher benötigt. Der gleiche Vorgang wird anschließend wiederholt, mit dem Unterschied, dass dieses Mal die Strecke einmal mit und einmal ohne vorkonfektionierten Koaxialkabel (Typ RG174/U, SMA-SMA) der Länge 25cm gemessen wird.

Im Ergebnis zeigt sich, dass 25 cm Koaxialkabel eine Verschiebung des durch die BOUNCE-Architektur bestimmten Umschaltpunktes um 435 Grundelemente bewirkt. Die gleiche, durch den Line Stretcher verursachte Längenänderung, führt zu einer Verschiebung des Umschaltpunktes um lediglich 341 Grundelemente.

Aufgrund des nahezu linearen Zusammenhangs zwischen Zeitdifferenz und Verschiebung des Umschaltpunktes steht fest, dass die Signallaufzeit in 25 cm Line Stretcher deutlich kleiner ist als im gleichlangen Koaxialkabel. Folglich ist auch die Ausbreitungsgeschwindigkeit im Line Stretcher höher als die angenommenen 2/3 c. Unter der Voraussetzung, dass zumindest im Kabel die Ausbreitungsgeschwindigkeit 2/3 c Gültigkeit besitzt, berechnet sich die Ausbreitungsgeschwindigkeit im Koaxialrohr zu 0,833 c oder $2{,}5 \cdot 10^8$ m/s.

Im Hinblick auf die vorgestellten Ergebnisse der BOUNCE-Architektur bedeutet dies, dass alle Zeitangaben um ca. 20% zu groß sind. Mit anderen Worten: es handelt sich bei den Ergebnissen um sehr konservativ bestimmte Werte.

11.4. Verteilte Grundelemente und lineare Interpolation

Mehrfach wurde im Ergebniskapitel darauf hingewiesen, dass die Kalibrierungspunkte der mittleren, im Messdatensatz am Kalibrierungspunkt zu findenden Anzahl der Zustände logisch-1 zugeordnet werden. Liegt die Anzahl der Zustände logisch-1 im Messdatensatz einer unbekannten Zeitdifferenz zwischen zwei Kalibrierungspunkten, wird der Bereich zwischen den Kalibrierungspunkten linear interpoliert. Die Differenz der Anzahl der Zustände

logisch-1 zwischen zwei Kalibrierungspunkten bestimmt die zeitliche Auflösung in diesem Bereich. Auch hier gilt die Annahme, dass sich die Grundelemente mit ihren jeweiligen charakteristischen Differenzen ε_i äquidistant auf den Messbereich verteilen. Da die ermittelten Messergebnisse trotz dieser Näherungen ausreichend genau sind, wurde auf diese Annahmen bisher nicht weiter eingegangen. Der Vollständigkeit halber folgt an dieser Stelle eine detaillierte Betrachtung der tatsächlichen Gegebenheiten im Verbund der Grundelemente. Dies ermöglicht ebenfalls eine Erklärung für einige Auffälligkeiten bei den Messergebnissen, die nicht ausschließlich mit den bereits dargestellten Eigenschaften der Signalgenerierung erklärt werden können.

Die Messergebnisse lassen erkennen, dass die charakteristischen Differenzen der Grundelemente nicht gleichmäßig über den Messbereich verteilt sind. Vielmehr existieren (Zeit-)Bereiche, in denen die Grundelemente dichter beieinanderliegen. Genauso gibt es Bereiche, in denen der Abstand der charakteristischen Differenz zweier (zeitlich) benachbarter Grundelemente größer ist. Ersichtlich ist dies auch in Abbildung 52, die die zeitliche Auflösung der BOUNCE-Architektur zwischen den einzelnen Kalibrierungspunkten darstellt. So ist die Auflösung im Bereich 200 ps - 300 ps mit 6,8 ps nur halb so groß wie im Bereich zwischen 500 ps - 550 ps mit 2,8 ps. Das bedeutet, dass die Differenz der Anzahl der Zustände logisch-1 zwischen 200 ps und 300 ps ebenfalls nur halb so groß ist wie zwischen 500 ps und 550 ps. Mit anderen Worten: Für Zeitdifferenzen zwischen 200 ps und 300 ps sind weniger Grundelemente in ihrem kritischen Bereich als zwischen 500 ps und 550 ps. Folglich muss der Abstand der charakteristischen Differenzen entsprechend unterschiedlich sein. Der Grund, weshalb die lineare Interpolation dennoch genügend genaue Resultate liefert, liegt darin, dass jeweils zwischen zwei Kalibrierungspunkten eine separate Interpolation erstellt wird. Dies macht erneut deutlich, dass die Kalibrierungspunkte ausreichend dicht, z.B. im Abstand von 50 ps, liegen sollten. Dadurch wird erreicht, dass die Interpolation immer nur einen sehr begrenzten Bereich des gesamten Messbereiches umfasst und so kleinere Fehler erzeugt, als für den Fall, dass der gesamte Messbereich nur mit einer einzigen Interpolationsgerade beschrieben wird.

Ein weiterer Aspekt der oben beschriebenen Tatsache, dass die charakteristischen Differenzen der Grundelemente mal mehr, mal weniger dicht zusammenliegen, ist in der folgenden Abbildung dargestellt. Sie zeigt, wie oft bei 100 Messungen ein und derselben Zeitdifferenz welcher Ergebniswert ermittelt wurde. Eine zweidimensionale Darstellung dieser Messergebnisse befindet sich bereits in Abbildung 53. Dort fehlen allerdings die Angaben zur Häufigkeit der einzelnen Ergebniswerte, auf die hier explizit eingegangen wird.

Betrachtet man die Messreihe, die zur eingestellten Zeitdifferenz 300 ps gehört, fällt auf, dass die Ergebnisberechnung nur zweimal Werte zwischen 300 ps und 305 ps ausgewiesen

hat. Wesentlich häufiger, nämlich 20-mal, wurden Werte zwischen 265 ps und 270 ps errechnet. Die Tatsache, dass im Mittel über 100 Messungen dennoch nahezu der exakte Wert von 300 ps bestimmt wurde (vgl. Abbildung 53), liegt darin begründet, dass in den Einzelmessungen ebenfalls häufig Werte zwischen 325 ps und 340 ps ermittelt wurden. Gleichzeitig ist bemerkenswert, dass die Messungen über den gesamten Bereich von 250 ps bis 300 ps eine deutliche Häufung der Ergebniswerte um 250 ps aufweisen. Dass der Mittelwert dennoch mit Fehlern kleiner als 10 ps folgt, liegt daran, dass sich Nebenmaxima ausbilden (ab 260 ps im Bereich 320 ps bis 345 ps), oder verschwinden (z.B. bis 290 ps im Bereich um 220 ps).

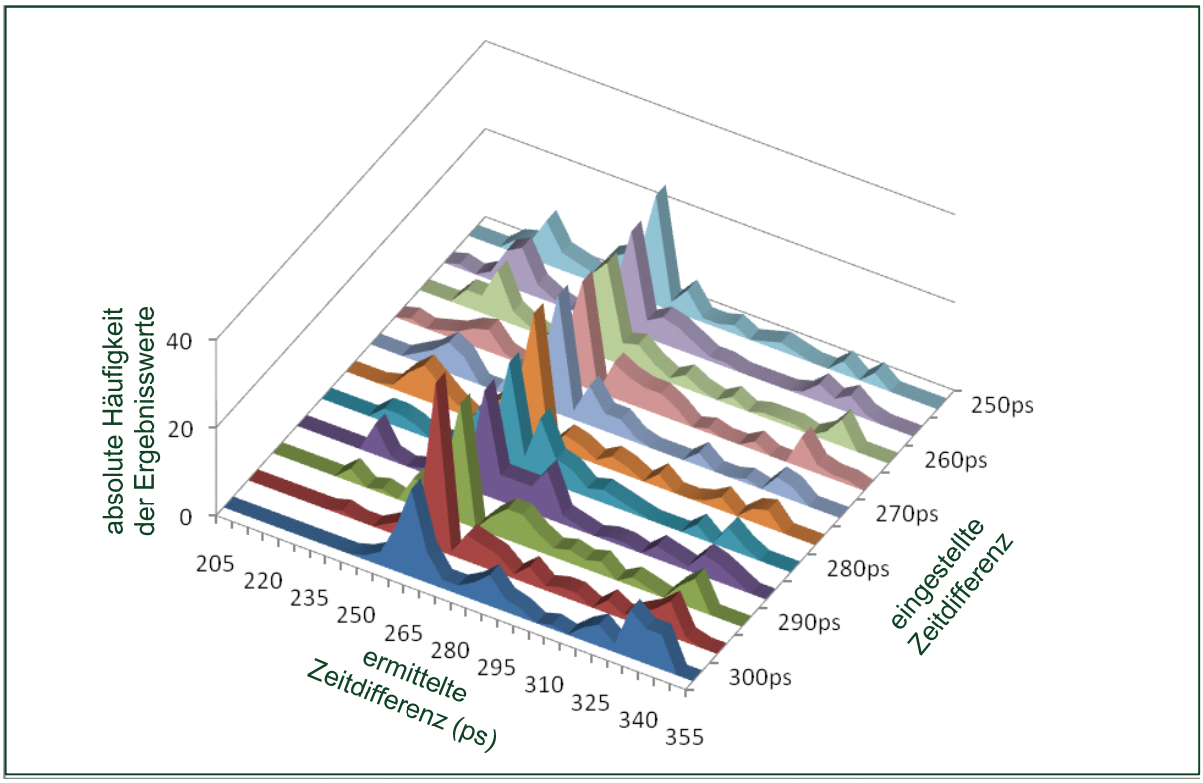


Abbildung 57: Gruppenbildung der Messergebnisse an einzelnen Zeitwerten - Indikator für ungleichmäßige Verteilung der charakteristischen Differenzen

Dieses Verhalten hat somit keinen negativen Einfluss auf Mehrfachmessungen, führt doch die Mittelwertbildung zu ausreichend genauen Ergebnissen. Anders ist die Lage in Bezug auf Einzelmessungen. Dort ist ein relativ großer Fehler zu erwarten, weil die Auswertung einige Zeitdifferenzen scheinbar nicht direkt darstellen kann.

Die nächste Abbildung erläutert diesen Umstand genauer. Dargestellt ist das Zeitverhalten verschiedener, benachbarter Grundelemente. Dieses ergibt sich, wie bereits in Abschnitt 10.1 erläutert und in Abbildung 45 dargestellt, aus der Zustandswahrscheinlichkeit logisch-1 in Abhängigkeit von der eingestellten Zeitdifferenz Δt. Eine zweidimensionale Darstellung dieser Eigenschaften für mehrere Grundelemente befindet sich ebenfalls in Abbildung 46.

Ging es dort vorrangig um den Verlauf dieser Kurve am einzelnen Grundelement, steht hier die Frage nach dem Gesamtbild unter Berücksichtigung vieler Grundelemente im Vordergrund. Die Auswertung einer Messung geht davon aus, dass die charakteristischen Differenzen ε_i der Grundelemente gleichmäßig weit auseinanderliegen. In Abbildung 58 zeigt sich ein anderes Bild: Es entstehen Gruppen von Grundelementen mit nahezu gleichen oder nur leicht veränderten charakteristischen Differenzen ε_i. Diese Gruppen sind zur besseren Erkennbarkeit mit unterschiedlichen Farben markiert. Zwischen diesen Gruppen aber treten deutliche Sprünge auf. Faktisch bedeutet dies, dass zwischen zwei benachbarten Grundelementen mit unterschiedlicher Farbmarkierung ein wesentlich größerer Abstand der beiden charakteristischen Differenzen liegt, als das zum Vergleich innerhalb einer Gruppe der Fall ist.

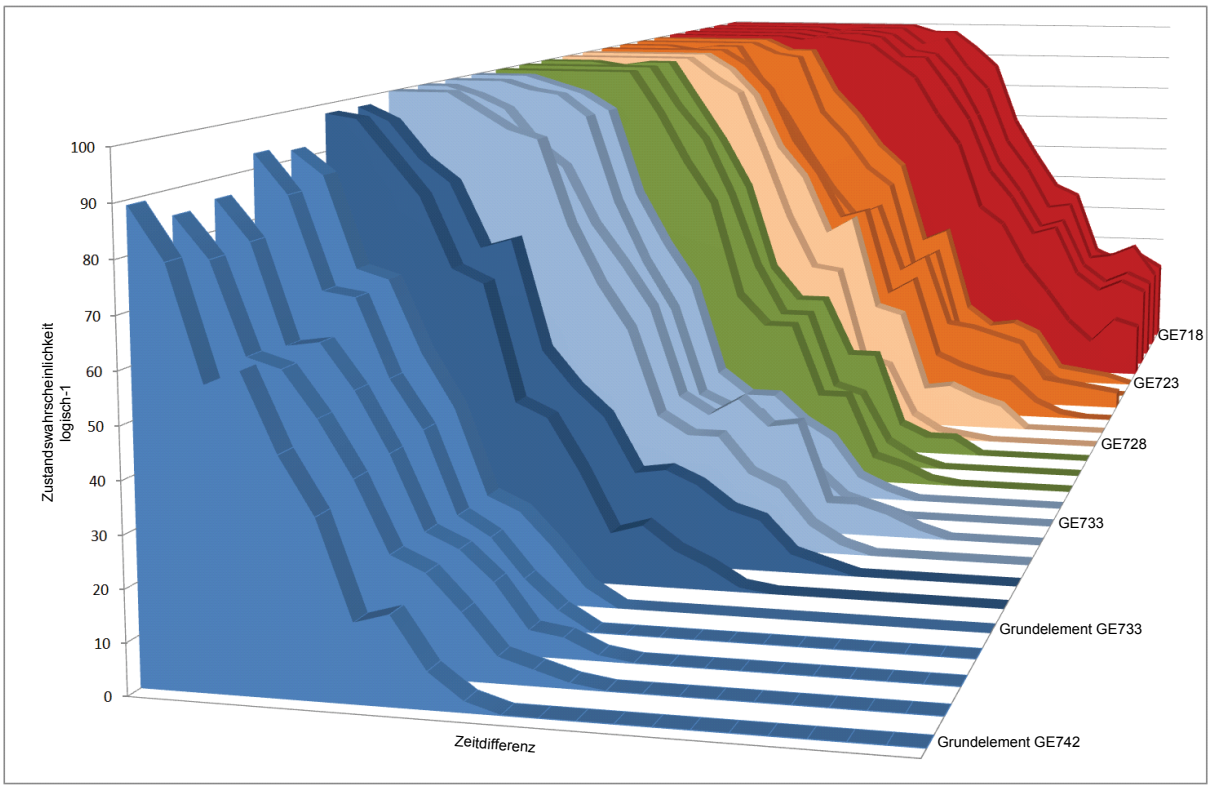


Abbildung 58: Zeitverhalten benachbarter Grundelemente, die Farben kennzeichnen Gruppen mit geringen Abständen zwischen ihren charakteristischen Differenzen, zwischen diesen Gruppen entsteht ein deutlicher Sprung der charakteristischen Differenz

Die Ergebnisse deuten somit darauf hin, dass es Zeitbereiche gibt, in denen wenige oder gar keine Grundelemente ihre charakteristische Differenz haben. Die Folge ist, dass gemessene Zeiten in diesen Bereichen immer mit einem größeren Fehler rekonstruiert werden, als innerhalb der dicht besetzten Gruppen einer Farbe. Gleichzeitig unterstreicht dies die Darstellung in Abbildung 57. Liegt ein Messergebnis mitten in einer der farblich gekennzeichneten Gruppen, führt eine leichte Ungenauigkeit im Messvorgang dazu, dass die Anzahl der Zustände logisch-1 sich deutlich im Datensatz ändert, da ja alle benachbarten

Grundelemente eine ähnliche charakteristische Differenz aufweisen. Die Rekonstruktion kennt allerdings diese Eigenschaft genauso wenig, wie die Sprünge der charakteristischen Differenz ε_i. Die Interpolation führt deshalb zu einem erhöhtem Fehler im ermittelten Zeitwert. Gut erkennbar ist dies auch in Abbildung 57. Die Messung von 300 ps führt vorrangig zu den Ergebniswerten 265 ps und 340 ps.

Eine Möglichkeit diese, insbesondere für die Einzelmessung, negativen Eigenschaften der BOUNCE-Architektur zu umgehen, ist die Ermittlung der tatsächlichen Eigenschaften aller Grundelemente. Dadurch würde die Notwendigkeit zur Interpolation zwischen den bekannten Kalibrierungspunkten entfallen. Da die charakteristischen Differenzen aber nicht bekannt sind, müsste die Kalibrierung mit minimal verschiedenen Zeitdifferenzen durchgeführt werden. Mit „minimal verschiedenen Zeitdifferenzen“ ist hier gemeint, dass die Kalibrierungspunkte nur noch ca. 1 ps bis 2 ps auseinanderliegen. Die dadurch entstehende Menge der Kalibrierungspunkte und die Forderung nach hochstabilen Zeitdifferenzen führen dazu, dass dieser Ansatz unpraktisch ist.

Eine Verringerung der Abstände zwischen den einzelnen Kalibrierungspunkten auf Werte von z.B. 25 ps führt aber immerhin zu einer Abmilderung der Interpolationsfehler, da der zu interpolierende Bereich zwischen zwei Kalibrierungspunkten immer kleiner wird. Hier liegt es im Ermessen des Entwicklers, einen angemessenen Kompromiss zwischen Genauigkeit und Kalibrierungsaufwand zu bestimmen. Soll die Messschaltung vorrangig für Einzelmessungen eingesetzt werden, sind dicht beieinanderliegende Kalibrierungspunkte zu bevorzugen.

Eine weitere Möglichkeit der Erhöhung der Genauigkeit besteht darin, bereits bei der Erstellung und Platzierung der BOUNCE-Architektur möglichst gleichmäßige Abstände der charakteristischen Differenzen anzustreben. Dieser Aspekt wird im nächsten Abschnitt behandelt.

11.5. Verbesserung von Auflösung und Genauigkeit

Die vorhergehende Diskussion hat, trotz der überzeugenden Messergebnisse der BOUNCE-Architektur, einige Diskussionspunkte aufgezeigt, die im Folgenden Gegenstand der Betrachtung sind. Insbesondere den nicht äquidistanten charakteristischen Differenzen ε_i wurde große Bedeutung zugemessen, da dieser Punkt Ursache systembedingt falscher Messergebnisse ist. Diese können zwar durch Mittelwertbildung abgemildert werden, stellen aber ein grundlegendes Problem dar. Dieser Abschnitt beschreibt Möglichkeiten, dieses Verhalten der BOUNCE-Architektur zu umgehen.

Das Problem hat seinen Ursprung in der Platzierung der Grundelemente innerhalb des FPGAs. Die Platzierung der Grundelemente erfolgte unter Berücksichtigung verschiedener

technischer Aspekte und einer durch den Computer umsetzbaren Platzierungsvorschrift. Die genaue Betrachtung des Platzierungsvorgangs befindet sich in Abschnitt 8.2.2. Eine Darstellung platzierter Grundelemente innerhalb des FPGAs ist in Abbildung 36 zu sehen. Was dort ausdrücklich keine Rolle spielte, sind die zeitlichen Eigenschaften der verwendeten Logikelemente. Diese sind ohnehin nicht bekannt und auch nicht im Vorfeld der Platzierung bestimmbar. Folglich kann im Vorfeld auch keine Platzierungsvorschrift bestimmt werden, die im Ergebnis eine besonders gleichmäßige Verteilung der charakteristischen Differenzen ε_i bietet.

Dennoch besteht eine einfachere Möglichkeit zur gleichmäßigeren Verteilung der charakteristischen Differenzen ε_i darin, die Anzahl der verwendeten Grundelemente zu erhöhen. Dabei ist darauf zu achten, dass die Grundelemente nicht so platziert werden, dass sie den Messbereich vergrößern sondern die Zahl der im konstanten Messbereich verwendeten Grundelemente erhöht wird. Natürlich sind bei dieser Vorgehensweise die in Kapitel 7 angesprochenen Aspekte für die praktische Umsetzung der BOUNCE-Architektur zu berücksichtigen. Dennoch lässt sich der zu erwartende Einfluss der Erhöhung der Zahl der Grundelemente abschätzen. Bekannt ist, dass sich an jedem Grundelement eine platzierungsabhängige charakteristische Differenz ε_i einstellt, deren genaue Größe aufgrund der Vielzahl zu berücksichtigender Parameter quasi zufällig ist. Diese Zufälligkeit führt ebenfalls zur im vorhergehenden Abschnitt besprochenen Lückenbildung zwischen einzelnen charakteristischen Differenzen. Wird die Anzahl der Grundelemente erhöht ohne den Messbereich zu vergrößern, sinkt einfach die Wahrscheinlichkeit, dass signifikante Lücken in der Abdeckung des Messbereiches mit charakteristischen Differenzen entstehen. Mit anderen Worten: Wird die Anzahl der im (konstanten) Messbereich befindlichen Grundelemente erhöht, erhöht sich auch die Wahrscheinlichkeit dafür, dass die charakteristischen Differenzen dicht und ohne große Abstände zusammenliegen. Die Ergebnisse der BOUNCE-Architektur in verschiedenen Entwicklungsstadien belegen diese These.

Im Zuge der Entwicklung des Prototyps, auf dessen Ergebnisse sich diese Arbeit bezieht, wurden verschiedene geometrische Formen der BOUNCE-Architektur getestet. Insbesondere zwei nahezu baugleiche Varianten sind für die Betrachtungen dieses Abschnittes relevant. Zum einen ist die Variante mit 1024 Grundelementen in 8 parallel angeordneten Linien zu erwähnen. Mit dieser Form der BOUNCE-Architektur wurden alle in dieser Arbeit dargestellten Ergebnisse ermittelt. Eine Vorstufe dieser Variante besteht aus nur 512 Elementen, verteilt auf lediglich 4 parallele Reihen. Für Testzwecke wurde ebenfalls eine Variante der BOUNCE-Architektur mit 2048 Grundelementen in 16 Reihen hinzugezogen.

Alle drei Vertreter der BOUNCE-Architektur wurden wie bereits beschrieben kalibriert, die Variante mit 2048 nur im Bereich zwischen 550 ps und 750 ps. Die Kalibrierungspunkte lagen dabei 1cm auf dem Line Stretcher auseinander, was nach den konservativen Annahmen für die Ausbreitungsgeschwindigkeit im Koaxialkabel einer Änderung der Signallaufzeit um 50 ps entspricht. Die folgende Abbildung stellt für diese Varianten die Anzahl der zwischen den Kalibrierungspunkten liegenden charakteristischen Differenzen innerhalb der jeweiligen Struktur dar.

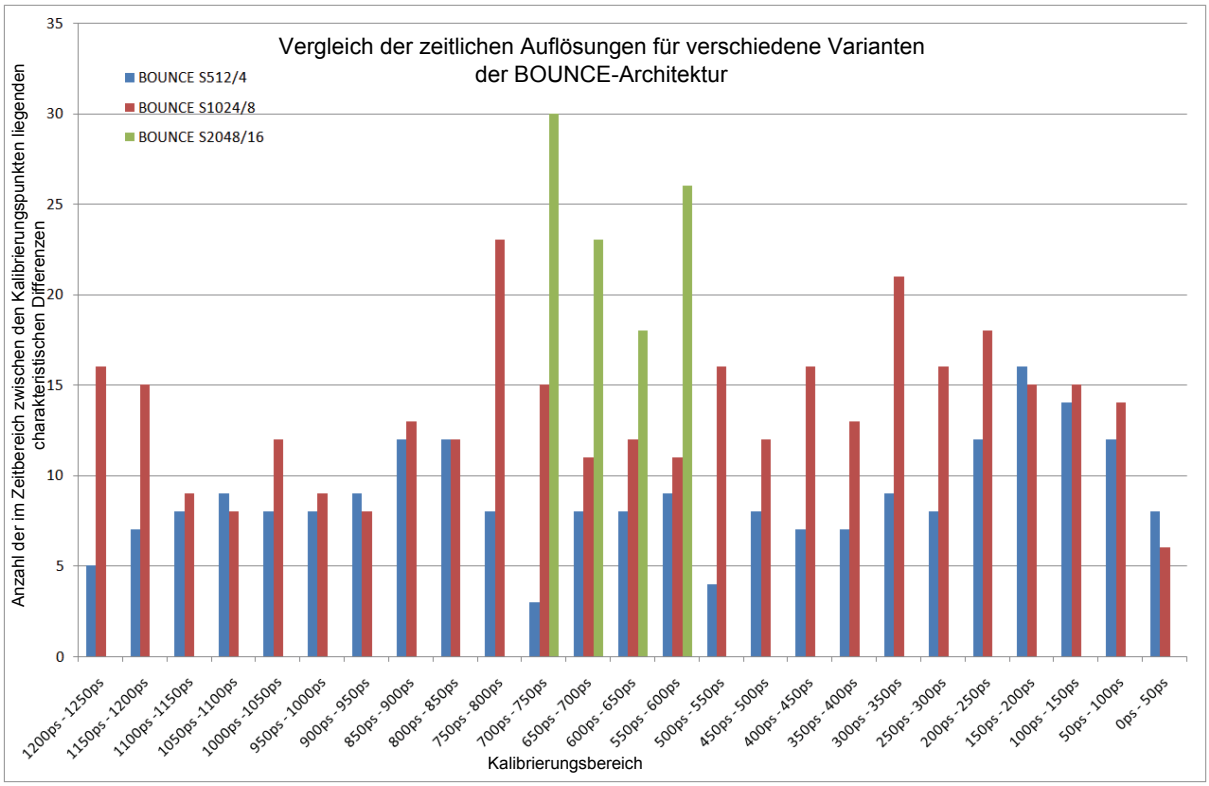


Abbildung 59: Vergleich der rechnerischen Auflösung zwischen den Kalibrierungspunkten für verschiedene Varianten der BOUNCE-Architektur

Deutlich zu erkennen ist, dass für die Variante mit 1024 Grundelementen die Werte für nahezu alle Zeitbereiche deutlich höher liegen als für die BOUNCE-Variante mit nur 512. Für die Variante mit 2048 Grundelementen liegen nahezu dreimal so viele charakteristische Differenzen zwischen den Kalibrierungspunkten, wie bei der Variante mit 512 Grundelementen. Da die genauen Werte dieser charakteristischen Differenzen zufälliger Natur sind, kann glaubwürdig argumentiert werden, dass die charakteristischen Differenzen bei mehr Grundelementen somit auch dichter zusammen liegen müssen. Dadurch wird ebenfalls erreicht, dass die Abweichungen durch die lineare Interpolation kleiner werden und somit der systematische Fehler sinkt.

Wie bereits in Abschnitt 10.2.1 dargestellt, errechnet sich aus der Anzahl der zwischen den Kalibrierungspunkten liegenden charakteristischen Differenzen die zeitliche Auflösung des

Messsystems. Die zeitliche Auflösung ist der Quotient aus der zwischen den Kalibrierungspunkten liegenden Zeitdifferenz und der Anzahl der in dieser Zeitdifferenz liegenden charakteristischen Differenzen. Dabei wird davon ausgegangen, dass diese charakteristischen Differenzen gleichmäßig zwischen den Kalibrierungspunkten verteilt sind, was allerdings nur bedingt stimmt, wie Abschnitt 11.4 diskutiert. Aus Abbildung 59 wird dennoch ersichtlich, dass nach diesem Ansatz die zeitliche Auflösung mit zunehmender Anzahl der Grundelemente steigt. Betrachtet man exemplarisch den Bereich zwischen 650ps und 700ps, so erzielen die drei Varianten jeweils eine zeitliche Auflösung von:

BOUNCE S512/4 → 6,25 ps

BOUNCE S1024/8 → 4,17 ps

BOUNCE S2048/16 → 2,17 ps

Die BOUNCE-Variante mit 2048 zeigt somit die höchste zeitliche Auflösung in diesem Vergleich. Im Bereich zwischen 700 ps und 750 ps erreicht die zeitliche Auflösung unter konservativen Schätzungen sogar einen Wert von 1,67 ps. Berücksichtigt man die Ausführungen aus Abschnitt 11.3, die eine im Vergleich zur konservativen Annahme höhere Geschwindigkeit der Signalausbreitung stützen, verringert sich dieser Wert weiter zu 1,3 ps.

Für praktische Versuche wurde diese Variante der BOUNCE-Architektur allerdings nicht eingesetzt, da sich diese Angaben mit dem bestehenden Versuchsaufbau nur bedingt verifizieren lassen. Die ursächlich durch den Versuchsaufbau bedingten Ungenauigkeiten sind um ein Vielfaches größer als die angegebene Auflösung, womit derartige Angaben unseriös wären.

Dennoch bleibt festzuhalten, dass die Auflösung der BOUNCE-Architektur prinzipiell beliebig erhöht werden kann. Auch die Angabe der Auflösung der Variante mit 2048 Grundelementen mit 1,3 ps stellt keineswegs eine feststehende Grenze dar. Da es sich bei der BOUNCE-Architektur um ein FPGA-basiertes Zeitmessssystem handelt, erlaubt diese auch die Anpassung der Struktur auf beliebige andere Auflösungen. Ob Auflösungen von unter 2 ps in Anbetracht der notwendigen äußeren Beschaltung des Messsystems (Verstärker, Detektoren, usw.) sinnvoll einzusetzen sind, liegt schlussendlich im Ermessensbereich der verantwortlichen Ingenieure. Aus technischer Sicht stellt die BOUNCE-Architektur hier keinen limitierenden Faktor dar.

12. Zusammenfassung und Ausblick

Die vorliegende Dissertation hat die Entwicklung eines Zeitmesssystems dargestellt, welches auf den Namen BOUNCE getauft wurde [26] [27]. Die Arbeit hat sich dabei auf zwei Ansichten der BOUNCE-Architektur konzentriert. Zum einen wurde das neuartige Grundkonzept für die Zeitmessung entwickelt, zum anderen wurde eine praktische Umsetzung des Konzepts vorgestellt. BOUNCE macht erstmalig Gebrauch von rein passiven Komponenten integrierter Schaltkreise um eine Zeitdiskriminierung und Signalverzögerung zu erzielen. Bei diesen passiven Komponenten handelt es sich um die internen Signalleitungen integrierter Schaltkreise. Die zwei Signale, die das zu messenden Zeitintervall definieren, nutzen jeweils eine Signalleitung, um den Schaltkreis zu durchlaufen. Zwischen beiden Signalleitungen befinden sich räumlich über den Schaltkreis verteilt einfache Entscheidungskomponenten, die erkennen, welches der beiden Signale sie jeweils zuerst erreicht hat. Diese Grundelemente stellen die beiden möglichen Ergebnisse durch unterschiedliche Ausgabebits dar. Aus der zeitlichen Abhängigkeit der beiden Signale wird somit ein räumliches „Bild“ erzeugt. Die Ausführungen zum Konzept der BOUNCE-Architektur haben dargestellt, dass in Abhängigkeit von der Größe des Zeitintervalls sich dieses „Bild“ signifikant ändert und somit zur Bestimmung der ursprünglichen Zeitdifferenz nutzbar ist.

Um eine Charakterisierung der Eigenschaften der Grundelemente zu ermöglichen, wurde der Begriff der charakteristischen Differenz ε_i eingeführt. Ist das zu messende Zeitintervall größer als die charakteristische Differenz des Grundelementes, zeigt dieses den Ausgabewert logisch-0, im anderen Fall logisch-1. Die charakteristische Differenz selbst ist von der geometrischen Position des Grundelementes auf dem Schaltkreis abhängig. Kapitel 6.2 hat dargestellt, dass bei der Verwendung vieler Grundelemente auf einem begrenzten Raum viele dicht beieinanderliegende charakteristische Differenzen erzeugt werden. Für die Zeitmessung führt dieses Verhalten zu einer besonders hohen zeitlichen Auflösung, da auf diesem Wege kleinste unterscheidbare „Zeitsegmente“ erzeugt werden. Während in anderen Technologien wie den Tapped Delay Lines die Laufzeit durch ein Logikgatter die zeitliche Auflösung limitiert, verwendet BOUNCE den Laufzeitversatz auf kürzesten Stücken der Signalleitungen integrierter Schaltkreise. Eine untere Grenze ist lediglich durch die maximale Platzierungsdichte der Grundelemente gegeben.

Da die Darstellung des Konzepts eine rein theoretische Sichtweise auf diese neue Möglichkeit der Zeitmessung ist, wurde in dieser Arbeit ein funktionierender Prototyp auf Basis eines Altera StratixII Entwicklungsboard vorgestellt. Dieser Prototyp ist in der Lage, unterschiedlichste Varianten der BOUNCE-Architektur im StratixII FPGA aufzunehmen. Die Unterschiede beziehen sich dabei vor allem auf die Geometrie und die Anzahl der

verwendeten Grundelemente. Eine externe Einrichtung zur Erzeugung unterschiedlich großer Zeitdifferenzen ermöglicht das Testen der FPGA-internen BOUNCE-Architektur. Die Auswertung der Messergebnisse erfolgt bei diesem Prototyp auf dem PC durch ein, im Rahmen dieser Arbeit erstellten, Programm.

Die verschiedenen Tests der BOUNCE-Architektur dienten im Wesentlichen der Feststellung der zeitlichen Auflösung und der erreichbaren Genauigkeit. Die Ergebnisse liegen für ein Stratix II FPGA unter Verwendung von 1024 Logikelementen bei 7 ps für die zeitliche Auflösung und einer Genauigkeit von +/-10 ps für Mehrfachmessungen [55]. Für Einzelmessungen erzielt die angesprochene Variante der BOUNCE-Architektur allerdings nur eine Genauigkeit von +/-100 ps. Die Gründe hierfür, die ausführlich in Abschnitt 11.2 und Abschnitt 11.4 diskutiert wurden, liegen zum einen in der externen Beschaltung der Messvorrichtung und zum anderen an den zeitlichen Eigenschaften der verwendeten Grundelemente. Für beide Aspekte wurden Ansätze für eine Verbesserung der Genauigkeit dargestellt.

Die Größe des Messbereiches liegt bei den dargestellten Varianten der BOUNCE-Architektur bei ca. 2 ns. Dieser Wert ist auf die Komponenten der externen Zeitgenerierung abgestimmt, mit denen nur Zeitdifferenzen zwischen 0 ns und 1,3 ns erzeugt werden können. Bei Beibehaltung der einfachen Reihengeometrie kann der Messbereich bis auf ca. 8 ns erweitert werden, wenn eine Spalte von Logikelementen des FPGAs vollständig ausgenutzt wird. Der maximale Messbereich kann noch größer ausgelegt werden, wenn auf die einfache Reihengeometrie verzichtet wird. Genaue Untersuchungen zum maximalen Messbereich sind im Rahmen dieser Arbeit nicht erfolgt, da dafür im jetzigen Versuchsaufbau keine Zeitintervalle generiert werden können.

Mit einer erreichten Auflösung von unter 10 ps und der Möglichkeit, diese Werte weiter zu verbessern, stellt BOUNCE gegenüber den Tapped Delay Lines eine Verbesserung um mindestens den Faktor fünf dar.

Damit ist es möglich, besonders kostengünstige Laufzeitmesssysteme zu gestalten. Alle verwendeten Komponenten des Messsystems (siehe Abschnitt 9) sind Standardprodukte der Elektronik. Ein FPGA mit ca. 10.000 Logikelementen ist bereits zu Preisen von unter 20 € zu erhalten. Die BOUNCE-Architektur verwendet pro Grundelement ein Logikelement, sodass die vorgestellten Varianten mit 1024 oder 2048 Grundelementen ohne Probleme in solche Schaltkreise implementiert werden können.

Für den Bereich der Laufzeitmesssysteme eröffnet BOUNCE damit neue Gestaltungsmöglichkeiten. Systeme zur Laufzeitmessung im praktischen Gebrauch sind zum Beispiel Abstandswarnsysteme in PKWs. Diese basieren derzeit auf der Radar-Technologie und sind aufgrund ihrer Komplexität bisher Fahrzeugen aus den oberen Preissegmenten vorbehalten.

Mit BOUNCE ist ein möglicher Ansatz gefunden, diese Systeme auch für den Massenmarkt wirtschaftlich sinnvoll umzusetzen. Die erreichten Auflösungen von 10 ps und weniger erlauben die Abstandsbestimmung zum vorausfahrenden Fahrzeug mit Genauigkeiten im Zentimeterbereich. Ebenfalls können Abstandsänderungen genauer detektiert werden und so eventuell integrierte Reaktionssystem, z.B. Bremsassistenten, verbessert werden.

Auch für den Bereich der Positionsbestimmung, der ausführlich in Kapitel 2 beschrieben wurde, hat die Entwicklung von BOUNCE eine große Bedeutung. Das in Abschnitt 2.2 vorgestellte Verfahren der Triangulation zum Beispiel verwendet Winkelangaben zu bekannten Punkten im Raum, um die Position, an denen die Winkel bestimmt wurden, zu ermitteln. Die Winkelmessung selbst basiert auf der Bestimmung der Laufzeitdifferenz zwischen zwei Empfängern, wie in Abbildung 7 dargestellt. Die Laufzeitdifferenz gibt an, in welchem Winkel die Verbindungslinie der beiden Empfänger zum Sender ausgerichtet ist. Um einen mobilen Einsatz der Positionsbestimmung zu unterstützen, wird an dieser Stelle mit einem Abstand der beiden Empfänger von 10 cm ausgegangen. Mit der durch BOUNCE realisierten Auflösung von deutlich unter 10 ps können in diesem Fall Winkelgenauigkeiten besser als 2° erzielt werden. Dabei ist zu beachten, dass die erreichbare Genauigkeit von der Ausrichtung des Senders zu den Empfängern abhängig ist. Die Angabe von 2° bezieht sich auf eine nahezu orthogonale Orientierung der Verbindungslinie beider Empfänger zum Sender. Ändert sich diese Orientierung, nimmt die Genauigkeit ab und erreicht ihr Minimum, wenn die Verbindungslinie beider Empfänger direkt auf den Sender zeigt.

Um ein vollständiges Lokalisierungssystem zu erhalten, sind daher noch einige Folgeuntersuchungen durchzuführen. Dazu gehört zunächst die Realisierung geeigneter Sender/Empfänger-Kombinationen, welche die Messsignale für die BOUNCE-Architektur liefern. Insbesondere ist darauf zu achten, dass die Funkstrecke sich durch konstante Latenzzeiten auszeichnet. Die Zeit vom Auftreten des physikalischen Effekts, z.B. einer Frequenzänderung am Empfänger, bis zu dessen Detektion und der damit verbundenen Aktivierung der BOUNCE-Architektur muss unter allen Umständen eine zeitliche Konstanz mit Abweichungen kleiner als 10 ps garantiert werden. Weiterhin muss eine geeignete Anordnung von Empfängern gefunden werden, um die bereits angesprochenen Schwankungen der Genauigkeit der Winkeldetektion zu verringern. Vorstellbar sind hier Systeme mit mehr als zwei Empfängern, um die ungünstige Orientierung von nur zwei Empfängern zu einem Sender zu umgehen.

Sind diese Probleme gelöst, wäre ein einfaches und vor allem mobiles Lokalisierungssystem verfügbar, welches im Vergleich zu gegenwärtig erhältlichen Systemen wesentlich kostenfreundlicher ist.

Derzeit basieren Lokalisierungssysteme, z.B. Ubisense [64], auf einem, im Vergleich zum BOUNCE-Ansatz, umgekehrten Funktionsprinzip. Die mobile Komponente, deren Position bestimmt werden soll, stellt den Sender dar. An bekannten Positionen im Raum befinden sich die Empfänger. Für eine Laufzeitmessung, also die Bestimmung, welcher Empfänger wann erreicht wurde, ist eine aufwendige Synchronisation der Empfänger untereinander nötig. In einem quadratischen Raum mit 10 m Kantenlänge sind so mehr als 40 m Netzwerkkabel für die Verbindung der Empfänger untereinander notwendig. Ferner muss eine separate Einheit aus den Zeitdaten die Position des mobilen Senders errechnen und diese dann wieder an die mobile Komponente übertragen. All diese Schritte wären mit BOUNCE und dem dargestellten Winkelmessverfahren überflüssig. Die Position wird direkt dort bestimmt, wo sie für weitere Verwendung benötigt wird. Da BOUNCE vollständig in einem Schaltkreisbaustein implementiert ist, ließe sich das Verfahren auch ohne großen Aufwand in bereits bestehende Systeme integrieren. Das größte Potenzial dürft hier von Mobiltelefonen, Navigationssystemen und ähnlichen Geräten ausgehen. Insgesamt würde so die flächendeckende Ausstattung von Räumen und Gebäuden mit Systemen zur Positionsbestimmung deutlich attraktiver werden. Dies wiederum kann die Basis für neuartige Assistenzsysteme bilden, zum Beispiel für Passagierleitsysteme auf Flughäfen oder in öffentlichen Gebäuden.

Literaturverzeichnis

[1] Bestimmung der Ausbreitungsgeschwindigkeit von Spannungsimpulsen auf Koaxialkabeln. In: *LEYBOLD Handblätter Physik* , S. P5.6.2.2

[2] Abas, M.A. ; Russell, G. ; Kinniment, D.J.: Design of Sub-10-Picoseconds On-Chip Time Measurement Circuit. In: *DATE '04: Proceedings of the conference on Design, automation and test in Europe*. Washington, DC, USA : IEEE Computer Society, 2004. – ISBN 0–7695–2085–5–2, S. 804–809

[3] Altera Corporation: *Cyclone Device Handbook*. 2008

[4] Altera Corporation: *Cyclone II Device Handbook, Volume1*. 2008

[5] Altera Corporation: *Stratix II Device Handbook*. 2008

[6] Altera Corporation: *NiosII Software Developer's Handbook*. 2009

[7] Andaloussie, M. S. ; Boukadoum, M. ; Aboulhamid, E.-M.: A novel time-to-digital converter with 150ps time resolution and 2.5ns pulse-pair resolution. In: *14th Int. Conf. Microelectron.*, 2002, S. 123–126

[8] Blumenthal, J. ; Reichenbach, F. ; Timmermann, D.: Minimal Transmission Power vs. Signal Strength as Distance Estimation for Localization in Wireless Sensor Networks. In: *3rd IEEE International Conference in Sensor and Ad Hoc Communications and Networks, (SECON 2006), Reston, USA*, 2006, S. 761–766

[9] Bürkle, H. (Hrsg.): *Grundlagen der Funktechnik*. R.v. Decker's Verlag, Heidelberg, 1989

[10] Bronstein, I.N. ; Semendjajew, K.A. ; Musiol, G. ; Mühlig, H.: *Taschenbuch der Mathematik*. Verlag Harri Deutsch, 1999

[11] Carr, Catherine E.: Delay Line Models of Sound Localization in the Barn Owl. In: *Amer. Zool.* 33 (1993), S. 79–85

[12] Cicalese, R. ; Aloisio, A. ; Branchini, P. ; Giordano, R. ; Izzo, V. ; Loffredo, S.: Implementation of High-Resolution Time-to-Digital Converters on two different FPGA devices. In: *10th Conference on Astroparticle, Particle and Space Physics, Detectors and Medical Physics Applications*, 2008

[13] Collins, M. ; Al-Hashimi, B.M.: On-Chip Time Measurement Architecture with Femtosecond Timing Resolution. In: *11th IEEE European Test Symposium (ETS06)*, 2006

[14] Corporation, Altera: *NiosII Processor Reference Handbook*. Altera Corporation, 2009

[15] Dudek, P. ; Szczepanski, S. ; Hatfield, J.V.: A high-resolution CMOS time-to-digital converter utilizing a Vernier delay line. In: *Solid-State Circuits, IEEE Journal of* 35 (2000), Feb, Nr. 2, S. 240–247. – ISSN 0018–9200

[16] Dunning, J. ; Garcia, G. ; Lundberg, J. ; Nuckolls, E.: An all-digital phase-locked loop with 50-cycle lock time suitable for high-performance microprocessors. In: *Solid-State Circuits, IEEE Journal of* 30 (1995), Apr, Nr. 4, S. 412–422. – ISSN 0018–9200

[17] Favi, C. ; Charbon, E.: A 17ps time-to-digital converter implemented in 65nm FPGA technology. In: *FPGA '09: Proceeding of the ACM/SIGDA international symposium on Field programmable gate arrays*. New York, NY, USA : ACM, 2009. – ISBN 978–1–60558–410–2, S. 113–120

[18] Grehn, J. (Hrsg.): *Metzler Physik*. Metzler Verlag, 1992

[19] Haley (Hrsg.) ; Scott (Hrsg.): *Analogue an Digital Computers*. Newness, 1960

[20] Herrmann, G. ; Müller, D.: *ASIC - Entwurf und Test*. Fachbuchverlag Leipzig, 2004

[21] Jansson, J.-P. ; Mantyniemi, A. ; Kostamovaara, J.: A CMOS time-to-digital converter with better than 10 ps single-shot precision. In: *Solid-State Circuits, IEEE Journal of* 41 (2006), June, Nr. 6, S. 1286–1296. – ISSN 0018–9200

[22] Jansson, Jussi-Pekka ; Mäntyniemi, Antti ; Kostamovaara, Juha: A delay line based CMOS time digitizer IC with 13 ps single-shot precision. In: *ISCAS (5)*, IEEE, 2005, S. 4269–4272

[23] Jeffress, Lloyd A.: A place theory of sound localization. In: *Journal of Comparative and Physiological Psychology* 41 (1948), February, Nr. 1, S. 35–39

[24] Joost, R. ; Salomon, R.: Advantages of FPGA-Based Multiprocessor Systems in Industrial Applications. In: *31st Annual Conference of the IEEE Industrial Electronics Society (IECON 2005)*. Raleigh, NC, USA : IEEE, 2005

[25] Joost, R. ; Salomon, R.: Hardware-Software Co-Design in Practice: A Case Study in Image Processing. In: *32nd Annual Conference of the IEEE Industrial Electronics Society (IECON 2006)*. Paris, France, November 2006, S. 3674–3679

[26] Joost, R. ; Salomon, R.: BOUNCE, a Concept ti Measure Picosecond Time Intervals with Standard Hardware. In: *13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA2008)*, 2008, S. 1010–1015

[27] Joost, R. ; Salomon, R.: BOUNCE, a new Approach to Measure Sub-Nanosecond Time Intervals. In: *International Conference on Field Programmable Logic and Applications (FPL 2008)*, 2008, S. 511–514

[28] Joost, R. ; Salomon, R. ; Schneider, M.: FPGAs and Soft-Core Processors: Understanding Computer Architecture and Processing Principles. In: *European Workshop on Microelectronics Education (EWME) 2006*. Stockholm, Sweden, June 2006, S. 79–82

[29] Kahmen, H.: *Angewandte Geodäsie: Vermessungskunde*. Berlin : Walter de Gruyter, 2005

[30] Kalisz, J.: Review of methods for time interval measurements with picosecond resolution. In: *Metrologia* 41 (2004), Nr. 1, S. 17–32

[31] Kalisz, J. ; Pawlowski, M. ; Pelka, R.: Error analysis and design of the Nutt time-intervall digitiser with picosecond resolution. In: *J. Phys. E: Sci. Instrum.* 20 (1987), S. 1330–1341

[32] Kalisz, J. ; Pawlowski, M. ; Pelka, R.: Präzisions-Zeitintervall-Messsystem. In: *Elektronik* 14 (1988), S. 65–68

[33] Kalisz, J. ; Szplet, R. ; Pasierbinski, J. ; Poniecki, A.: Field-programmable-gate-array-based time-to-digital converter with 200-ps resolution. In: *IEEE Transactions on Instrumentation and Measurement* 46 (1997), February, Nr. 1, S. 51–55

[34] Kalisz, J. ; Szplet, R. ; Pelka, R. ; Poniecki, A.: Single-chip low-cost time counter for distance measurements with 3 cm resolution. In: *Journal of Optics* 29 (1998), Juni, S. 199–205

[35] Karalar, T. C.: *Implementation of a Localization System for Wireless Sensor Networks*, University of California, Berkeley, Diss., 2006

[36] Karpov, N. R.: Vernier Method of Measuring Time Intervals. In: *Measuring Techniques* 23 (9) (1980), S. 44–46

[37] Kempter, R. ; Gerstner, W. ; Hemmen, J. L. ; Wagner, H.: Temporal coding in the sub-millisecond range: Model of barn owl auditory pathway. In: *Advances in Neural Information Processing Systems 8*, 1996, S. 124–130

[38] Kempter, R. ; Leibold, C. ; Wagner, H. ; Hemmen, J. L.: Formation of temporal-feature maps by axonal propagation of synaptic learning. In: *Proc Natl Acad Sci U S A* 98 (2001), March, Nr. 7, S. 4166–4171. – ISSN 0027–8424

[39] Konishi, M.: Neuroethology of sound localization in the owl. In: *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology* 173 (1) (1993), S. 3–7

[40] Lehmann, G. ; Wunder, B. ; Selz, M.: *Schaltungsdesign mit VHDL*. Franzis Verlag, 1998

[41] Leskovar, B. ; Turko, B.: Optical Timing Receiver for the NASA Laser Ranging System. In: *Lawrence Berkeley Laboratory Report LBL* 6133 (1977)

[42] Leskovar, B. ; Turko, B.: Optical Timing Receiver for the NASA Spaceborn Ranging System. In: *Lawrence Berkeley Laboratory Report LBL* 8129 (1978)

[43] Mäntyniemi, A.: *An integrated CMOS high precision time-to-digital converter based on stabilised three-stage delay line interpolation*. Oulu, Finnland, Diss., 2004

[44] Maxfield, C.: *The Design Warrior's Guide to FPGAs*. Orlando, FL, USA : Academic Press, Inc., 2004. – ISBN 0750676043

[45] Microlab: *Line Stretchers, SR-Series*. 2008

[46] Middelhoek ; George ; Dekker: *Physics of Computer Memory Devices*. Academic Press, Inc., 1976

[47] Minami, K. ; Mizuno, M. ; Yamaguchi, H. ; Nakano, T. ; Matsushima, Y. ; Sumi, Y. ; Sato, T. ; Yamashida, H. ; Yamashina, M.: A 1 GHz portable digital delay-locked loop with infinite phase capture ranges, 2000, S. 350–351, 469

[48] Mota, M. ; Christiansen, J. ; Debieux, S. ; Ryjov, V. ; Moreira, P. ; Marchioro, A.: A flexible multi-channel high-resolution time-to-digital converter ASIC. In: *Nuclear Science Symp. Conf. Rec.* 2 (2000), S. 155–159

[49] Möser, M. ; Müller, G. ; Schlemmer, H.: *Handbuch Ingenieursgeodäsie*. H.Werner, 2000

[50] Määttä, K. ; Kostamovaara, J.: High-Precision Time-to-Digital Converter for pulsed Time-of-Flight Laser Radar Applications. In: *IEEE Trans. Instrum. Meas.* 47 (1998), S. 521–536

[51] Nutt, Ronald: Digital Time Intervalometer. In: *Review of Scientific Instruments* 39 (1968), Nr. 9, S. 1342–1345

[52] Reichardt, J. ; Schwarz, B.: *VHDL-Synthese - Entwurf digitaler Schaltungen und Systeme*. Oldenbourg Verlag, 2007

[53] Roehr, S. ; Vossiek, M. ; Gulden, P.: Method for High Precision Radar Distance Measurement and Synchronization of Wireless Units. In: *Microwave Symposium, 2007, IEEE/MTT-S International*, 2007, S. 1315–1318

[54] Salomon, R. ; Schneider, M. ; Wehden, D.: Low-Cost Optical Indoor Localization System for Mobile Objects without Image Processing. In: *11th IEEE International Conference on Emerging Technologies and Factory Automation*, 2006, S. 629–632

[55] Salomon, Ralf ; Joost, Ralf: BOUNCE, a New High-Resolution Time Interval Measurement Architecture. In: *IEEE Embedded Systems Letters* 1 (2) (2009)

[56] Schneider, M. ; Wehden, D. ; Salomon, R.: Akustische Ortung mobiler Objekte in großen Räumen. In: *32. Deutsche Jahrestagung für Akustik (DAGA'06)*. Braunschweig, Germany, March 2006, S. 405–406

[57] Seferiadis, G. ; Pouchet, M. ; Gough, M.P.: FPGA implemntation of a delay-line readout system for a particle detector. In: *Measurement* 39 (1) (2006), S. 90–99

[58] Song., J. ; An, Q. ; Liu, S.: A High-Resolution Time-to-Digital Converter Implemented in Field-Programmable-Gate-Arrays. In: *IEEE Transactions on Nuclear Science* 53 (2006), Februar, S. 236–241

[59] Strassacker, G.: *Rotation, Divergenz und das Drumherum*. B.G.Teubner, Stuttgart, 1999

[60] Swan, B. K.: A 100-ps Time-Resolution CMOS Time-to-Digital Converter for Positron Emission Tomography Imaging Applications. In: *IEEE Journal of Solid State Circuits* 39 (2004), Nr. 11, S. 1839–1852

[61] Systems, Stanford R.: *Frequency Counters SRS620 - Universal Time Interval Counter and Frequency Counter*. Stanford Research Systems, 2006

[62] Szymanowski, R. ; Kalisz, J.: Field programmable gate array time counter with two-stage interpolation. In: *Rev. Sci. Instrum.* 76 (2005), S. 045 104

[63] Tietze, U. ; Schenk, Ch.: *Halbleiter-Schaltungstechnik*. Springer Verlag, 2002

[64] Ubisense Ltd.: *Ubisense Systemübersicht*. 2007

[65] Walter, G. ; Herms, G.: *Einführung in die Behandlung von Meßfehlern*. Rostock : Universität Rostock, 1998

[66] Wehden, D. ; Salomon, R. ; Schneider, M.: Low-Cost Sonic-Based Indoor Localization for Mobile Robots. In: *3rd Workshop on Positioning, Navigation and Communication (WPNC'06)*. Hannover, Germany, March 2006, S. 53–58

[67] Widiger, H. ; Salomon, R. ; Timmermann, D.: Packet Classification with Evolvable Hardware Hash Functions. In: *2nd International Workshop, BioADIT 2006*. Osaka Japan, January 2006, S. 64–79

[68] Wiedwald, J.D.: A CAMAC High Resolution Time Interval Meter. In: *IEEE Transactions on Nuclear Science* 20 (1973), S. 242–245

[69] Xie, D. K. ; Zhang, Q. C. ; Qi, G. S. ; Xu, D. Y.: Cascading delay line time-to-digital converter with 75ps resolution and a reduced number of delay cells. In: *Rev. Sci. Instrum.* 76 (2005), S. 014 701

[70] Zhang, Y. ; Huang, P. ; Zhu, R.: Upgrading of Integration of Time to Digit Converter on a Single FPGA. In: *15th International Laser Ranging Workshop*. Canberra, Australia, October 2006

[71] Zhou, Wei ; Xuan, Zongqiang ; Yu, Jianguo: Some new methods for precision time interval measurement. In: *IEEE International Frequency Control Symposium*, 1997, S. 418–421

Thesen

1. Field-Programmable Gate Arrays (FPGAs) eignen sich aufgrund ihrer regelmäßigen Struktur zur Implementierung von digitalen Zeitmesssystemen, welche aus kettenförmigen Strukturen aufgebaut sind.
2. Digitale, FPGA-implementierte Zeitmesssysteme basieren häufig auf der kettenförmigen Aneinanderreihung einzelner Verzögerungselemente und Entscheidungsbausteine. Die Verzögerungselemente werden dabei explizit aufgebaut, die Entscheidungsbausteine ermitteln die zeitliche Abhängigkeit zweier Signale.
3. Ist die Größe der zeitlichen Verzögerung im Verzögerungselement bekannt, wird aus der in den Entscheidungselementen gespeicherten zeitlichen Abhängigkeit zweier Signale eine Zeitdifferenz errechenbar.
4. Die im FPGA erreichbare zeitliche Auflösung aktueller Messsysteme ist dabei durch die minimal erreichbare Signalverzögerung in den Verzögerungselementen begrenzt. Bei aktuellen FPGAs beträgt dieser Wert etwa 50 ps.
5. Auf den expliziten Aufbau von Verzögerungselementen in solchen Zeitmesssystemen kann verzichtet werden, da die Signalverzögerung auf den schaltkreisinternen Signalleitungen bereits die notwendige Zeitquantisierung erlaubt. Diese Annahme wurde mit der Realisierung der BOUNCE-Architektur belegt.
6. Mit den globalen Routing-Kanälen bieten FPGAs Leitungen, welche die gesamte Chip-Fläche parallel zu dessen Kanten durchlaufen. Entlang dieser Routing-Kanäle sind eine Vielzahl Logikzellen an die Signalleitungen angeschlossen. Jede Logikzelle nimmt einen Entscheidungsbaustein auf. Aufgrund ihrer räumlichen Nähe liegen zwischen zwei benachbarten Elementen nur sehr kurze Leitungsabschnitte. Demzufolge ist die Signalverzögerung zwischen zwei Elementen sehr klein, was zu einer hohen zeitlichen Auflösung führt. Eine weitere Erhöhung der zeitlichen Auflösung ist durch Ausbildung paralleler Signalpfade möglich.
7. Als Entscheidungsbaustein für die BOUNCE-Architektur eignet sich das pegel-gesteuerte RS-FlipFlop. Vor der Messung befindet es sich in seinem „irregulären“ Zustand, während der Messung wechselt es in den Zustand speichern. Der gespeicherte Zustand ist abhängig von Reihenfolge der Änderung beider Eingangssignale. Die theoretische Möglichkeit der Oszillation des Ausgangs bei gleichzeitiger Änderung beider Eingangssignale konnte in der Praxis nicht nachgewiesen werden. Vielmehr zeigt sich die Implementierung des RS-FlipFlops als bistabiles Element mit einem zeitabhängigen Verlauf der Zustandswahrscheinlichkeit.

8. Die der BOUNCE-Architektur von außen zugeführten Messsignale, die die zu bestimmende Zeitdifferenz definieren, durchlaufen den Chip in entgegengesetzter Richtung. Jedes erreichte Entscheidungselement hält fest, welches der beiden Signale das Element als erstes erreicht hat. Bei Betrachtung aller Entscheidungselemente zeigt sich eine zeitabhängige, räumliche Verteilung zwischen Elementen, die jeweils durch eins der beiden Messsignale als erstes erreicht wurden. Daraus lässt sich auf die Größe der gemessenen Zeitdifferenz schließen.
9. Da die exakte Größe der Signalverzögerung zwischen einzelnen Elementen nicht bekannt ist, ist eine Kalibrierung der BOUNCE-Architektur mit bekannten Zeitdifferenzen erforderlich. Die daraus resultierenden Verteilungen der Zustandswerte der Entscheidungselemente werden bei Messung unbekannter Zeitdifferenzen zum Vergleich herangezogen. Dieser Vergleich, in Kombination mit Interpolationsverfahren, erlaubt die exakte Bestimmung der gemessen Zeitdifferenz.
10. Die implementierte BOUNCE-Architektur mit 1024 Entscheidungselementen erreicht eine zeitliche Auflösung von 4,2 ps und stellt damit eine neue Qualität im Bereich der FPGA-basierten Zeitmesssysteme dar, da BOUNCE die von bestehenden Implementierungen der technologisch verwandten Tapped Delay Lines erreichten Auflösungswerte um den Faktor zehn verbessert.
11. Die Genauigkeit der BOUNCE-Architektur ist wesentlich von der Qualität der Ereignisdetektion außerhalb des Chips abhängig. Die BOUNCE-Architektur erfordert eine hochgenaue Signaldetektion mit konstanter Detektionszeit. Die eigentliche Latenz der Ereigniserkennung ist dabei unerheblich, sofern sie konstant ist. Praktische Versuche mit einem Puls-Generator als Signalquelle haben für Mehrfachmessungen eine mittlere Genauigkeit von +/-10 ps ergeben.
12. Die maximal erreichbare Auflösung der BOUNCE-Architektur ist vom Konzept her nicht begrenzt. Die praktisch erreichbare Auflösung wird dadurch begrenzt, wie klein die Zeitverzögerung zwischen den einzelnen Entscheidungselementen gestaltet werden kann.
13. Bereits eine zeitliche Auflösung von 10 ps ermöglicht eine Längenmessung mit einer Auflösung von 3 mm bis 4 mm, was bei entsprechender Empfangssensorik eine neue Qualität für auf Trilateration basierende Lokalisierungsverfahren ermöglicht.
14. Die BOUNCE-Architektur mit ihrer zeitlichen Auflösung von unter 10 ps ermöglicht bei der Verwendung von zwei Empfängern eine Winkelmessung zu einem Sender mit einer Auflösung von 2° bei einem Antennenabstand von nur 10 cm. Damit lassen sich Lokalisierungsverfahren erreichen, die a) in Kleinstgeräte (z.B. Handheld) integrierbar sind und b) keine bidirektionale Kommunikation mit einer Infrastruktur erfordern, was insbesondere im Indoor-Bereich eine Neuerung darstellt.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe. Die Arbeit wurde bisher in gleicher oder ähnlicher Form weder veröffentlicht noch einer anderen Prüfungsbehörde vorgelegt.

Rostock, den 01.11.2009 Ralf Joost

Lebenslauf

Persönliche Daten

Name:	Ralf Joost
Adresse:	Fritz-Reuter-Str. 6, 18225 Kühlungsborn
Tel.-Nr.:	0179 1376414
E-Mail:	ralf.joost@uni-rostock.de
Geburtsdatum:	21.06.1978
Geburtsort:	Gardelegen (Sachsen-Anhalt)
Familienstand:	ledig

Schulausbildung

09/1985 – 07/1992	Karl-Friedrich-Wilhelm Wander Schule, Gardelegen
09/1992 – 07/1997	Geschwister Scholl Gymnasium, Gardelegen, Abschlussnote: sehr gut

Wehrdienst

09/1997 – 06/1998	Wehrdienst beim Artillerie-Regiment 1 in Nienburg

Hochschulausbildung

10/1998	Beginn des Studiums an der Universität Rostock, Fachrichtung Elektrotechnik
02/2000	Aufnahme in die Studienstiftung des Deutschen Volkes
04/2001	Abschluss des Grundstudiums, Abschlussnote: Gut
05/2001	Beginn des Hauptstudiums, Schwerpunkt: Angewandte Mikroelektronik und Datenverarbeitung
09/2004	Abgabe und Verteidigung der Diplomarbeit, Thema: „Implementierung eines abstrakten Verhaltensmodells eines Ethernet-Aggregators zur Produkt-Exploration und Realisierung

	konkreter Teillösungen“, Ergebnis: 1,3
10/2004	Abschluss des Studiums als Diplomingenieur für Elektrotechnik Gesamtnote: Gut

Berufserfahrung

10/2004 – 12/2004	Wissenschaftlicher Mitarbeiter am Institut für Allgemeine Elektrotechnik der Universität Rostock
ab 12/2004	Wissenschaftlicher Mitarbeiter am Institut für Angewandte Mikroelektronik und Datentechnik der Universität Rostock

Liste der eigenen Veröffentlichungen

- Ralf Salomon, Ralf Joost: **BOUNCE, a New High-Resolution Time Interval Measurement Architecture** in IEEE Embedded System Letters, Vol. 1, No. 2, August 2009, DOI: 10.1109/LES.2009.2034711
- Enrico Heinrich, Marian Lüder, Ralf Joost, Ralf Salomon: **FPGA-based Implementation Alternatives for Keyed-Hash Message Authentication Code in Networked Embedded Systems** IEEE 14th International Conference on Emerging Technologies and Factory Automation (ETFA 2009), La Palma, Spanien, September 2009
- Ralf Joost, Ralf Salomon: **BOUNCE, a Concept to Measure Picosecond Time Intervals with Standard Hardware** 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA2008), pp. 1010-1015, ISBN: 1-4244-1506-3, Hamburg, Deutschland, September 2008
- Ralf Joost, Ralf Salomon: **BOUNCE, a new Approach to Measure Sub-Nanosecond Time Intervals** International Conference on Field Programmable Logic and Applications (FPL 2008), pp. 511-514, ISBN: 978-1-4244-1961-6, Heidelberg, Deutschland, September 2008
- Enrico Heinrich, Sebastian Staamann, Ralf Joost, Ralf Salomon: **Comparison of FPGA-based Implementation Alternatives for Complex Algorithms in Networked Embedded Systems - the Encryption Example** 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008), pp. 1449-1456, ISBN: 1-4244-1506-3, Hamburg, Deutschland, September 2008
- Ralf Joost, Ralf Salomon: **High Quality Offset Printing - An Evolutionary Approach** in Proceedings of the ACM Genetic and Evolutionary Computation Conference GECCO 2007, pp.2053-2058, ISBN: 987-1-59593-697-4, London, England, Juli 2007
- Ralf Joost, Ralf Salomon: **Hardware-Software Co-Design in Practice: A Case Study in Image Processing** 32nd Annual Conference of the IEEE Industrial Electronics Society (IECON 2006), ISBN: 1-4244-0136-4, pp. 3674-3679, Paris, France, November 2006
- Ralf Joost, Frank Golatowski, Dirk Timmermann: **Power Saving Mechanisms in Wireless Sensor Networks** 4th International Forum Life Science Automation, Rostock, Deutschland, September 2006
- Ralf Joost, Frank Golatowski, Dirk Timmermann: **Sensor Networks in Laboratory Automation** 4th International Forum Life Science Automation, Rostock, Deutschland, September 2006

- Ralf Joost, Ralf Salomon, Matthias Schneider: **FPGAs and Soft-Core Processors: Understanding Computer Architecture and Processing Principles** European Workshop on Microelectronics Education (EWME) 2006, pp. 79-82, ISBN: 91-7178-402-0, Stockholm, Schweden, Juni 2006
- Ralf Joost, Ralf Salomon: **Advantages of FPGA-Based Multiprocessor Systems in Industrial Applications** 31st Annual Conference of the IEEE Industrial Electronics Society (IECON 2005), ISBN: 0-7803-9253-1, Raleigh, North-Carol, USA, November 2005

Kurzreferat

Das Messen von Zeitintervallen spielt in vielen technischen Anwendungen eine große Rolle. Ein Zeitintervall wird dabei durch ein Start- und ein Stop-Ereignis definiert. Von besonderer Bedeutung ist die präzise Zeitmessung z.B. im Bereich der Medizin, der Physik und der Lokalisierungssysteme.

Durch Messen der Dauer, die ein Signal zur Überwindung einer Strecke benötigt, lässt sich die Länge des zurückgelegten Weges errechnen. Die Anforderungen an die Auflösung der Zeitmessung hängen von der Art des verwendeten Signals ab. Aufgrund ihrer physikalischen Eigenschaften werden häufig elektromagnetische Signale genutzt, die mit etwa 300.000 km/s allerdings die höchste Ausbreitungsgeschwindigkeit besitzen. Soll die für ein Lokalisierungsverfahren notwendige Streckenmessung mit einer räumlichen Auflösung besser als 1 cm erfolgen, ist eine zeitliche Auflösung besser als 30 ps erforderlich, die von keinem der bekannten Verfahren zur Laufzeitmessung mit vertretbarem Aufwand erreicht werden.

Kern dieser Arbeit ist die Entwicklung eines technischen Systems, das unter Verwendung von kostengünstiger Standard-Hardware zeitliche Auflösungen von 30 ps erreicht. Das System basiert dabei auf einem neuartigen Konzept, welches den Effekt der Signalverzögerung auf internen Leitungen integrierter Schaltkreise ausnutzt. Dieses Konzept trägt den Namen BOUNCE.

Im Rahmen dieser Arbeit wurde das entwickelte Konzept mittels eines Entwicklungsboards auf einem StratixII-FPGA prototypisch umgesetzt. Die durchgeführten Experimente zeigen, dass diese Variante des BOUNCE-Konzepts eine mittlere zeitliche Auflösung von 4,2 ps bei einer Genauigkeit von +/-10 ps für die Mehrfachmessung erreicht. Weitere Untersuchungen zeigen, dass die bisher erreichte Auflösung keine Limitierung darstellt, da weder Gatterlaufzeiten noch die Platzierungsdichte Einfluss auf die Auflösung hat. Optimierungsmöglichkeiten in Bezug auf die Genauigkeit der BOUNCE-Architektur identifiziert diese Arbeit vor allem im Bereich der im externen Messaufbau verwendeten Komponenten.

Abstract

Measuring the duration of a time interval is of particular importance in many technical (real-world) applications. In this context, a time interval is defined by two events, called start and stop. In application areas, such as medicine, physics, and localization, the focus is primarily on precision and resolution.

In localization, for example, the time-of-flight of a signal is used to calculate the distance between sender and receiver. Then, the required resolution depends on the type signal that is used. Many applications try to utilize electromagnetic signals. These signals have several, quite advantageous properties, but have with about 300,000 km/s the fastest traveling speed. Consequently, a resolution of 1 cm would then require a time resolution of at least 30 ps, which is not provided by any of the existing, cost-effective systems.

This thesis discusses a new time measurement system. This system is called BOUNCE, and is based on the delay of electromagnetic signals on the internal wirings of integrated circuits. A significant feature of BOUNCE is that it does not require specialized circuits but can be implemented on cheap, of-the-shelf standard hardware.

The developed concept was prototypically implemented on a StratixII field-programmable gate array. In the experiments, this prototype was already able to yield a time resolution of 4.2 ps at a precision of +/- 10 ps. Further considerations indicate that the obtained resolution of 4.2 ps is not a limit, since the resolution does not depend on gate delays and/or the on-chip package density of active elements. Furthermore, the discussion identifies several optimization options in terms of the external apparatus that has been used in all the practical experiments.