



Lehrstuhl für Kommunikations- technik

Masterarbeit M 06-2015

Implementierung und Analyse eines PSSS-PHYs als Codemultiplexverfahren

von

Ramona Croonenbroeck

Abgabedatum: 17. Februar 2016

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Zielsetzung	1
1.3	Gliederung der Arbeit	2
2	Grundlagen	3
2.1	Industrielle Anforderungen	3
2.2	Frequenzspreizung	3
2.2.1	Frequency Hopping Spread Spectrum	4
2.2.2	Direct Sequence Spread Spectrum	5
2.2.3	Parallel Sequence Spread Spectrum	7
2.2.4	PSSS-ASK-PHY nach IEEE 802.15.4-2011	11
2.3	Code Division Multiple Access	13
2.4	Frequenz- und Phasenoffset	14
3	Konzeptionierung	15
3.1	PSSS als Codemultiplex	15
3.2	Herleitung der Bitfehlerrate für Sequenzen verschiedener Leistung	18
3.3	Dynamische Schwellwertanpassung im Decoder	22
3.4	Simulationsszenarien	25
4	Implementierung	27
4.1	Aufbau des PHY	27
4.2	Aufbau der Simulation	28
4.3	Validierung	31
5	Auswertung	32
5.1	AWGN-Kanal	32
5.2	Laufzeitverzögerung	36
5.3	Empfangsleistung	39
5.4	Frequenz- und Phasenoffset	44
5.5	Dynamische Schwellwertanpassung	49

6 Fazit	51
6.1 Zusammenfassung	51
6.2 Ausblick	52
A PSSS-Code-Tabelle	54
Abkürzungsverzeichnis	55
Symbolverzeichnis	57
Abbildungsverzeichnis	60
Tabellenverzeichnis	62
Literaturverzeichnis	63

1 Einleitung

1.1 Motivation

Im Bereich der Industrieautomatisierung nimmt die Bedeutung drahtloser Übertragung stetig zu, da im Vergleich zu kabelgebundenen Netzwerken hohe Mobilität und Flexibilität gegeben sind und ein geringerer Wartungsaufwand von Nöten ist. Allerdings bringt die kabellose Übertragung Herausforderungen wie zeitvariante Kanalverhältnisse mit sich, die durch Bewegung der Teilnehmer und Mehrwegeausbreitung aufgrund von Reflexionen hervorgerufen werden. Außerdem ist oft keine Line Of Sight (LOS)-Verbindung gegeben.

Für industrielle Umgebungen wird der PSSS-ASK-PHY im IEEE 802.15.4-2011 Standard beschrieben, bei dem das Parallel Sequence Spread Spectrum (PSSS)-Spreizverfahren genutzt wird, um zum einen die Störfestigkeit gegenüber schmalbandigem Rauschen und Mehrwegeempfang zu erhöhen und zum anderen mehrere Datenströme parallel senden zu können. Hierzu wird eine M-Sequenz, die sogenannte Basissequenz, um einen bestimmten Betrag zyklisch verschoben, sodass mehrere M-Sequenzen entstehen. Die Datenbits eines einzelnen Teilnehmers werden parallelisiert und jeweils mit einer der Sequenzen gespreizt. Danach wird die Summe der entstandenen PSSS-Sequenzen übertragen. Die Daten weiterer Teilnehmer werden im TDMA-Verfahren abgearbeitet [1].

Für industrielle Anwendungen ist es jedoch unter Umständen von Vorteil, wenn mehrere Teilnehmer gleichzeitig senden können. Durch die Spreizung mit unterschiedlichen Sequenzen kann aus PSSS auch ein Code Division Multiple Access (CDMA)-Verfahren realisiert werden, welches im Folgenden untersucht werden soll. Dieses Verfahren bietet gleich mehrere Vorteile: Zum einen ist aufgrund der parallelen Datenverarbeitung eine sehr geringe Latenz gegeben. Zum anderen ist bei Nutzung mehrerer Basissequenzen als CDMA-System eine Koexistenz mehrerer PSSS-Systeme auf der knappen Ressource Funk möglich.

1.2 Zielsetzung

Aufbauend auf einer vorangegangenen Bachelorarbeit wird im Rahmen dieser Arbeit die Implementierung des PSSS-Verfahrens so erweitert, dass mehrere Teilnehmer gleichzeitig Da-

ten senden können, indem sie jeweils eine oder mehrere der zyklisch verschobenen PSSS-Sequenzen erhalten. Die mit den M-Sequenzen gespreizten Bits werden übertragen und kommen als Summe am Empfänger an. Die Herausforderung hierbei ist, dass für alle Teilnehmer unterschiedliche Kanalverhältnisse existieren. Somit können sich verschiedene Laufzeiten und Empfangsleistungen aufgrund der Entfernungen zum Empfänger und eine Abweichung der Trägerfrequenzen ergeben. Anschließend sollen die von diesen Effekten beeinflussten Performanzgrenzen des Systems bestimmt werden.

1.3 Gliederung der Arbeit

Die Arbeit ist in sechs Kapitel unterteilt. Kapitel 1 enthält die Motivation und die Zielsetzung, wonach im zweiten Kapitel einige Grundlagen zum Verständnis der Arbeit erläutert werden. So werden die Frequenzspreizungsverfahren Frequency Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum (DSSS) und PSSS, letzteres sowohl allgemein als auch nach dem IEEE 802.15.4-2011 Standard, beschrieben. Weiterhin wird das CDMA Verfahren und der Einfluss eines Phasen- oder Frequenzoffsets auf das gesendete Signal erläutert.

Kapitel 3 dient der Konzeptionierung der Arbeit. Hier erfolgt die Beschreibung von PSSS als Codemultiplex. Daraus hervorgehend erfolgt auch eine neue Herleitung der Bit Error Rate (BER) für Teilnehmer mit unterschiedlichen Empfangsleistungen. Des Weiteren wird eine Erweiterung des Decoders beschrieben sowie eine Übersicht über die durchgeführten Simulationsszenarien gegeben.

Im vierten Kapitel wird die Implementierung der Simulationsumgebung und des darin enthaltenen PHY beschrieben. Außerdem erfolgt die Validierung der Simulation.

Das fünfte Kapitel dient der Auswertung der im dritten Kapitel entworfenen Simulationsszenarien. Mögliche Auswirkungen von Laufzeitverzögerungen, Unterschieden in der Empfangsleistung und Frequenzoffsets werden untersucht sowie die Erweiterung des Decoders durch dynamische Schwellenanpassung.

In Kapitel 6 werden die wesentlichen Ergebnisse zusammengefasst und es wird ein Ausblick auf mögliche Nachfolgearbeiten gegeben.

2 Grundlagen

In diesem Kapitel werden die zum Verständnis der Arbeit notwendigen Grundlagen behandelt. Zunächst wird auf industrielle Anforderungen eingegangen. Danach werden allgemein die Frequenzspreizung und im Detail die Verfahren FHSS, DSSS und PSSS erläutert. Anschließend wird das CDMA-Verfahren behandelt und die Auswirkungen eines Phasen- bzw. Frequenzoffsets bei der Demodulation erläutert.

2.1 Industrielle Anforderungen

In der Automatisierungstechnik herrschen hohe Anforderungen bezüglich Latenz, Datenrate und Zuverlässigkeit der Systeme. Die Funkkommunikation bietet hierbei ein hohes Maß an Mobilität und Flexibilität und außerdem den Vorteil der Verschleißfreiheit. Häufig wird eine Sterntopologie verwendet. Hier sind alle Teilnehmer des Netzwerks mit einem zentralen Knoten verbunden, mit welchem sie ausschließlich kommunizieren. Der Ausfall eines Teilnehmers hat somit keine Auswirkungen auf den Rest des Netzwerks und auch eine Erweiterung ist leicht möglich [19], [2]. Allerdings ist der freie Raum als Übertragungsmedium eine begrenzte Ressource, sodass ein koordinierter Umgang durch beispielsweise Multiplexverfahren nötig ist. Insbesondere im industriellen Umfeld ist durch reflektierende Objekte ein stark schwankender Übertragungskanal mit Mehrwegeausbreitung gegeben, bei dem unter Umständen keine LOS-Verbindung existiert. Somit ergeben sich im Gegensatz zu kabelgebundenen Übertragungssystemen höhere Paketfehlerraten, die durch geeignete Fehlerkorrekturverfahren ausgeglichen werden müssen. Diese verursachen einen zeitlichen Jitter und somit eine erhöhte Latenz. Durch eine Frequenzspreizung erhöht sich die Robustheit gegenüber vielen Störeinflüssen und somit verringern sich sowohl Latenz als auch die Paketfehlerrate [19].

2.2 Frequenzspreizung

Als Frequenzspreizung oder Spread Spectrum werden Verfahren bezeichnet, bei denen die Bandbreite eines schmalbandigen Signals vergrößert wird. Hierbei wird zwischen frequenzspringenden und Direct Sequence-Verfahren unterschieden [3].

Zur Spreizung werden Pseudorandom Noise (PN) Sequenzen verwendet. Diese bestehen aus einer Folge von Pseudozufallszahlen, das heißt sie erscheinen einem Betrachter als zufällig, folgen aber dem sich periodisch wiederholenden Schema eines linear rückgekoppelten Schieberegisters (LFSR) [4]. Die Zufallszahlen der PN-Sequenzen werden als Chips bezeichnet.

Aufgrund ihrer Abhörsicherheit und erhöhten Robustheit gegenüber schmalbandigen Störern wurden Frequenzspreizverfahren ursprünglich für militärische Zwecke verwendet. Heute werden sie aufgrund ihrer hohen Zuverlässigkeit auch in ziviler mobiler Kommunikation eingesetzt [7].

2.2.1 Frequency Hopping Spread Spectrum

Beim FHSS-Verfahren geschieht die Spreizung durch Aufteilung der verfügbaren Bandbreite in mehrere nicht überlappende Kanäle, über welche die Information übertragen wird. Dazu springt der Sender in jedem Zeitintervall zu einem anderen Kanal, wie in Abbildung 2.1 zu erkennen. Die Reihenfolge der Sprünge wird durch eine PN-Folge festgelegt. Mit einer M-ären PN-Sequenz können 2^M Kanäle unterschieden werden. Der Empfänger muss hierbei mit dem Sender synchronisiert sein [13].

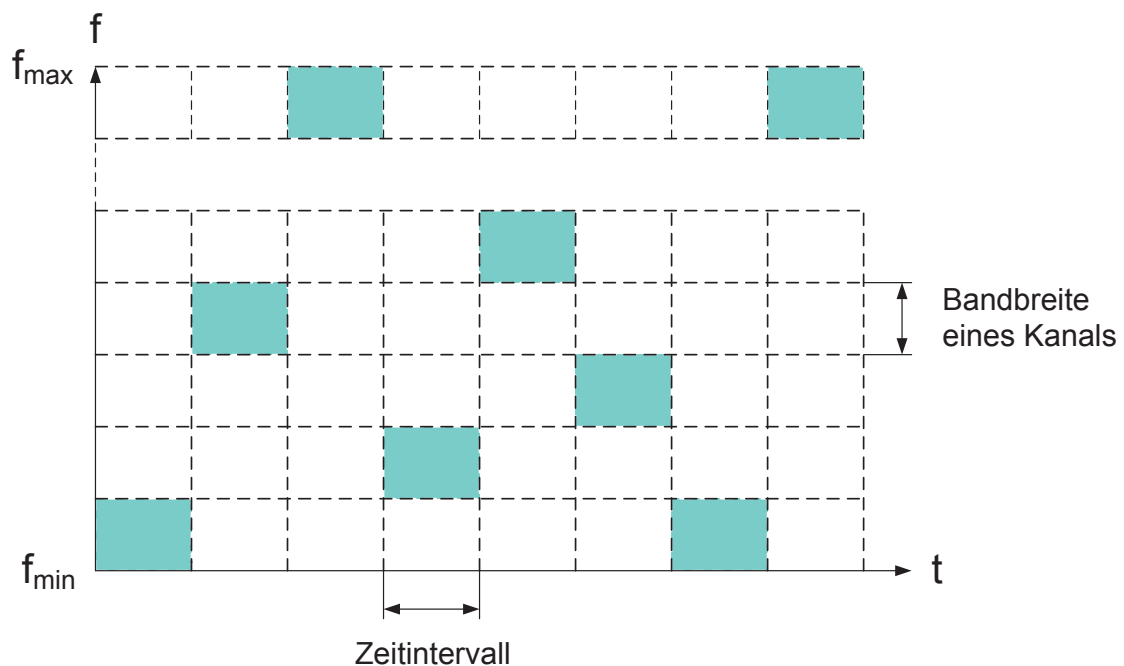


Abbildung 2.1: FHSS-Verfahren

2.2.2 Direct Sequence Spread Spectrum

Das DSSS-Verfahren ist aufgrund seiner Einfachheit die am weitesten verbreitete Spread Spectrum-Methode. Das Nutzsignal wird wie in Abbildung 2.2 bitweise mit einer PN-Sequenz multipliziert, welche eine Länge von M Chips hat. Die Chipfrequenz ist hierbei größer als die Bitfrequenz des Informationssignals.

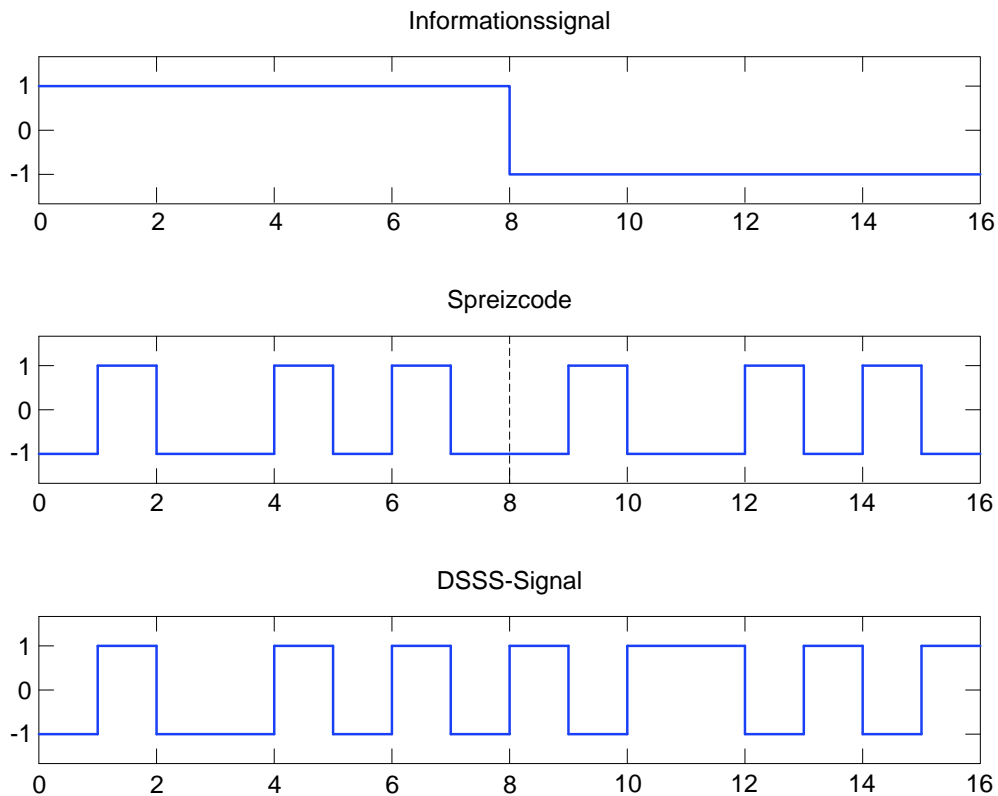


Abbildung 2.2: Erzeugung eines DSSS-Signals

Das entstandene Signal hat nun eine um den Spreizfaktor L_C größere Bandbreite. L_C kann als das Verhältnis aus Symboldauer T_t des Nutzsignals und Chipdauer T_{Chip} des Spreizsignals dargestellt werden, oder alternativ aus dem Verhältnis der Bandbreite W des Spreizsignals zur Bandbreite B des Nutzsignals [9].

$$L_C = \frac{T_n}{T_{\text{Chip}}} = \frac{W}{B} \quad (2.1)$$

Durch die Verbreiterung des Spektrums verkleinert sich die spektrale Leistungsdichte, sodass das Signal bei ausreichend großem Spreizfaktor vom Rauschen des Übertragungskanals verdeckt werden kann und somit nur schwer abhörbar ist. Mit dem Spreizfaktor wächst zudem die tolerable Störleistung von schmalbandigen Störern, da das Störsignal am Empfänger gespreizt, das Nutzsignal aber mit der gleichen PN-Sequenz wie am Sender wieder entspreizt

wird. Somit bleibt nach der anschließenden Bandpassfilterung nur noch ein kleiner Teil der Störleistung erhalten. Dies gilt allerdings nicht für Störsignale unendlicher Bandbreite wie z.B. AWGN. Da sich hier die Rauschleistung am Empfänger nach der Spreizung nicht verändert, ergibt sich folglich kein Spreizgewinn. Abbildung 2.3 zeigt die spektrale Leistungsdichte des ungespreizten Nutzsignals (links), des DSSS-Signals und eines schmalbandigen Störsignals (Mitte) und des entspreizten Nutzsignals mit dem nun gespreizten Störsignal (rechts) [4]. Hierbei ist f_1 die Frequenz des Nutzsignals im Basisband und f_c die Trägerfrequenz, auf der das DSSS-Signal übertragen wird.

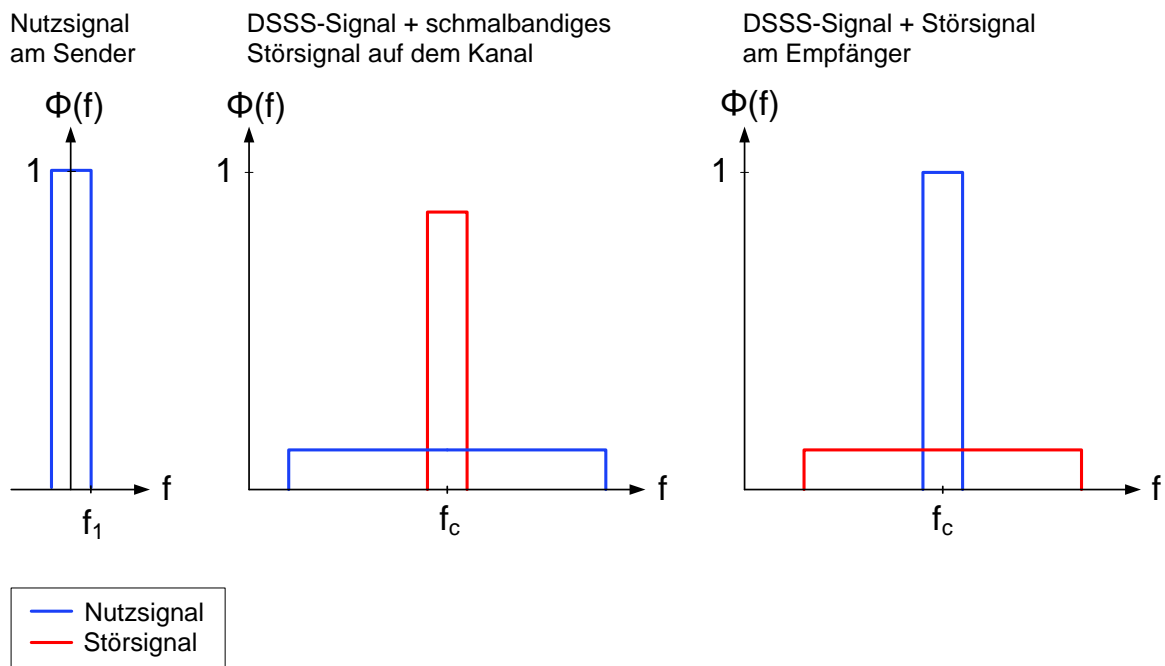


Abbildung 2.3: spektrale Leistungsdichte bei DSSS

Diese Verbesserung des Signal-Rauschverhältnisses um den Spreizfaktor wird als Spreizgewinn G_P bezeichnet [15]. Dieser lässt sich mit Hilfe der Signalleistung P_S und der Störleistung P_N berechnen [12].

$$\frac{P_N}{P_S} = \frac{G_P}{E_b/N_0} \quad (2.2)$$

Die Entspreizung am Empfänger erfolgt, wie in Abbildung 2.4 zu sehen, durch Multiplikation mit derselben Spreizsequenz wie am Sender. Hierzu müssen Sender und Empfänger synchronisiert sein.

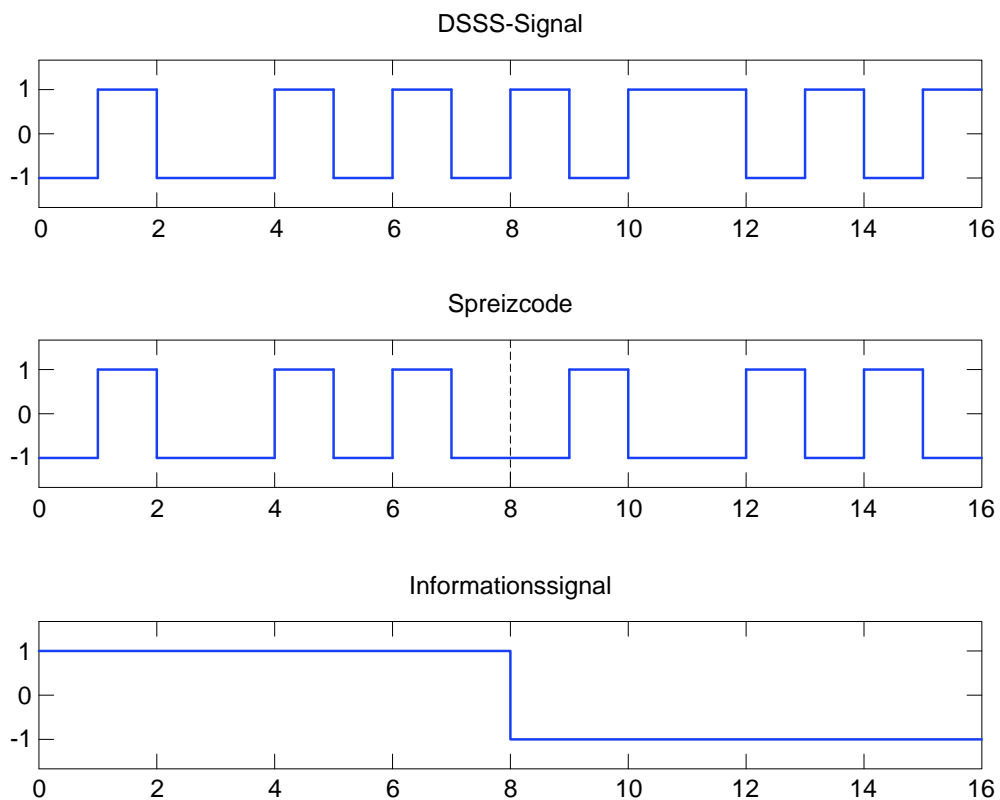


Abbildung 2.4: Rückgewinnung des Informationssignals

Das DSSS-Verfahren eignet sich gut zur Realisierung von Multiple Access-Verfahren wie CDMA, welches in Abschnitt 2.3 beschrieben wird. Weitere Anwendung findet das DSSS-Verfahren beispielsweise in der Entfernungsmessung beim Global Positioning System (GPS) und dem Satellitenkommunikationsnetz Globalstar [17].

2.2.3 Parallel Sequence Spread Spectrum

Das PSSS-Verfahren ist eine weitere Methode zur Frequenzspreizung, die sich durch ihre Stör-empfindlichkeit auszeichnet. Die Spreizung erfolgt per DSSS, allerdings werden hier aus der verwendeten M-Sequenz durch zyklische Verschiebung weitere, fast orthogonale Sequenzen generiert, sodass mehrere Sequenzen parallel übertragen werden können.

PSSS besitzt eine sternförmige Topologie. Im Uplink senden die Teilnehmer ihre PSSS-codierten Daten per Time Division Multiple Access (TDMA) zum Master, dieser wiederum sendet im Downlink nacheinander Informationen an die verschiedenen Teilnehmer.

Die für das PSSS-Verfahren genutzte M-Sequenz wird als Basissequenz bezeichnet. Ihre Länge n_R hängt von der Art des PSSS-Verfahrens ab. Es wird ein Shift festgelegt, um welchen die Basissequenz zyklisch verschoben wird, um die weiteren Sequenzen zu erzeugen.

Die maximal mögliche Anzahl der entstandenen Sequenzen hängt vom Shift und der Länge der Basissequenz ab. Es kann aber auch eine geringere Anzahl an Sequenzen genutzt werden. Aus den erzeugten Sequenzen wird eine PSSS-Code-Tabelle \underline{D} erstellt. Zusätzlich kann an jede entstandene Sequenz ein Guard Intervall in Form einer zyklischen Erweiterung zur Vermeidung von Inter Symbol Interference (ISI) angehängt werden [18], [14]. Die Informationsbits eines Senders werden in Blöcke von parallelen Bits aufgeteilt, deren Größe von der Anzahl der genutzten M-Sequenzen abhängig ist. Die entstandenen Symbole werden mit der PSSS-Code-Tabelle multipliziert, sodass jede M-Sequenz mit einem Bit gewichtet wird. Anschließend werden die entstandenen Sequenzen addiert, sodass eine multivalente PSSS-Sequenz entsteht [5]. Hierbei treten Amplituden bis zur Anzahl der verwendeten Sequenzen auf. Abbildung 2.5 zeigt das Codierungsverfahren von PSSS. Vor der Übertragung wird das PSSS-Signal auf einen Träger moduliert und einer Impulsformung unterzogen.

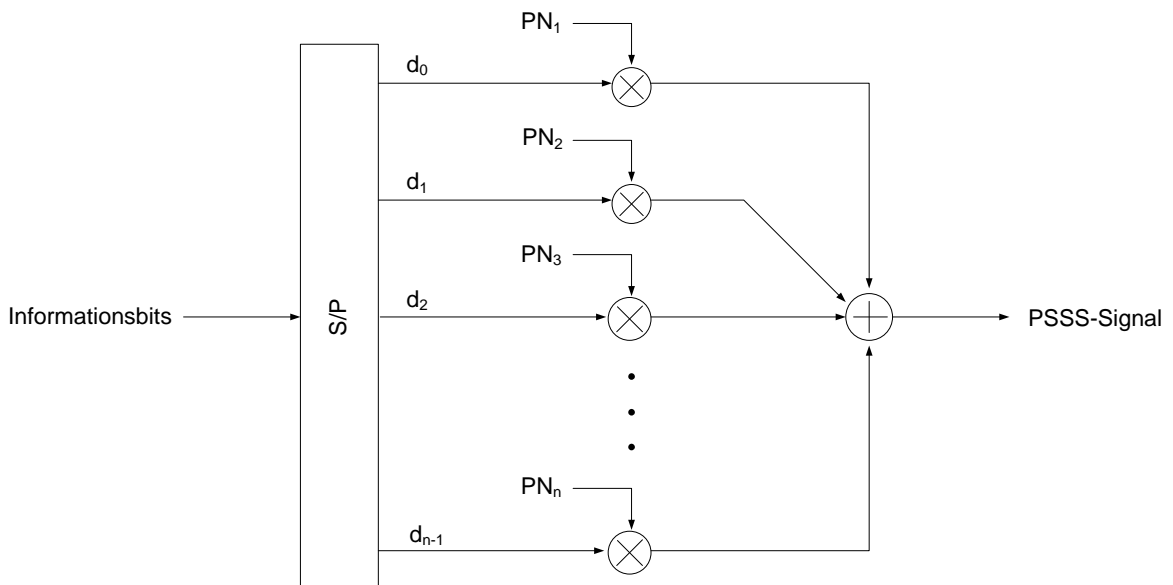


Abbildung 2.5: Encodierung des PSSS-Signals

Nach der Übertragung wird im Decoder eine zyklische Korrelation des Empfangssignals mit der Basissequenz durchgeführt. Hierbei wird das Signal mit jeder gültigen Verschiebung der Basissequenz multipliziert. Dank der Linearität dieser Operation können auch aus der Summe der Signale die Informationsbits zurückgewonnen werden. Abbildung 2.6 zeigt den Ablauf der Decodierung des PSSS-Signals.

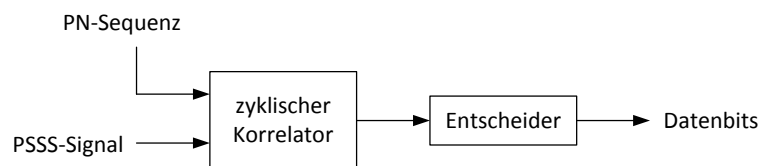


Abbildung 2.6: Decodierung des PSSS-Signals

Es soll nun die Korrelation genauer betrachtet werden. Die Autokorrelationsfunktion (AKF) einer bipolaren M-Sequenz lautet

$$\varphi(\tau) = \begin{cases} n_R & \text{wenn } \tau \bmod n_R = 0 \\ -1 & \text{sonst .} \end{cases} \quad (2.3)$$

Hierbei ist τ die Verschiebung der M-Sequenz in Chips. Es wird nun die Multiplikation des PSSS-Signals mit der unverschobenen Basissequenz betrachtet, also für $\tau = 0$. Hierdurch wird das erste Bit d_0 des übertragenen Symbols decodiert, da es bei der Codierung mit der Basissequenz gespreizt wurde. Für die AKF ergibt sich $\varphi(\tau = 0) = n_R$. Für alle anderen Bits ist $\tau \neq 0$, also ergibt sich $\varphi(\tau) = -1$. Der Wert der AKF wird mit dem Vorzeichen des jeweiligen Bits gewichtet. Die Korrelationsamplitude C_K ist die Summe aller Korrelationsergebnisse dieses Symbols. C_K ist also abhängig vom Wert von $n_0 = d_0 \cdot n_R$ und der Anzahl der 1en n_1 und -1en n_{-1} innerhalb des empfangenen Symbols. Der Betrag von C_K wird um die Anzahl der Bits mit gleichem Wert wie d_0 verringert und um die Anzahl der Bits mit entgegengesetztem Wert vergrößert, wobei der Einfluss aller Bits außer dem zu decodierenden Bit d_0 als Verschiebung n_s der Korrelationsamplitude bezeichnet wird.

$$\begin{aligned} C_K &= n_0 - n_1 + n_{-1} \\ &= n_0 + n_s \end{aligned} \quad (2.4)$$

Hieraus ist ersichtlich, dass der Signal-Rauschabstand abhängig von der Anordnung der Bits pro Symbol ist. C_K muss für jedes Bit des Symbols einzeln berechnet werden, da sich für jedes τ sowohl n_0 als auch n_s verändern kann. Abbildung 2.7 und 2.8 zeigen die Beeinflussung der Korrelationsamplitude durch ein Bit mit gleicher bzw. entgegengesetztem Wert, wenn insgesamt nur 2 Bits verwendet werden.

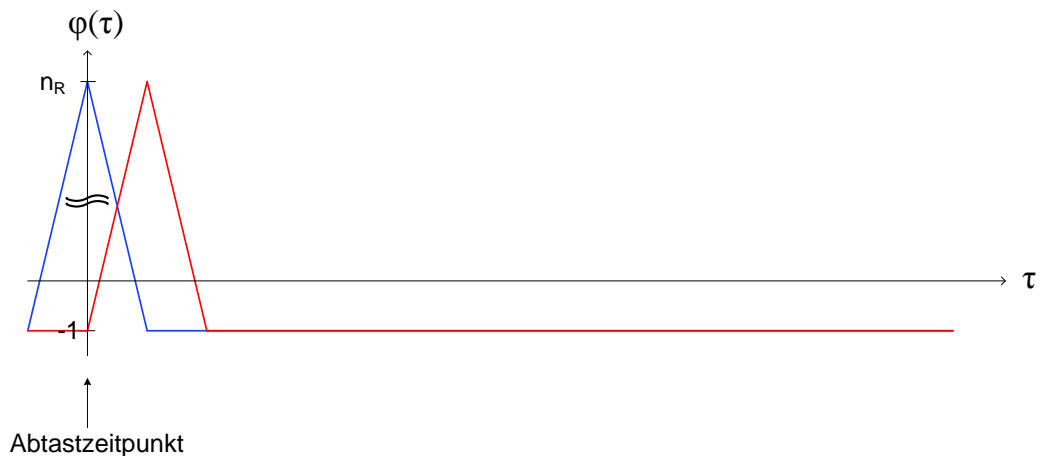


Abbildung 2.7: Korrelation des zu decodierenden Bits und eines Bits gleichen Werts

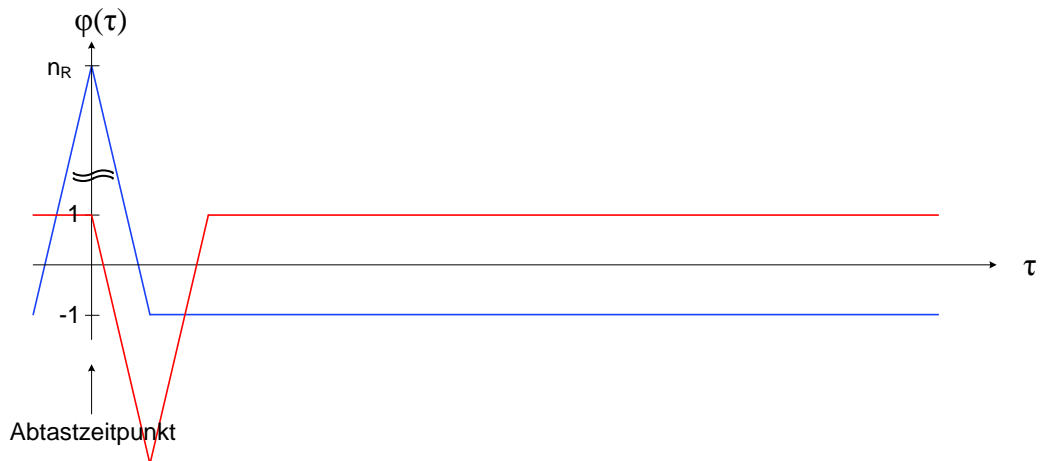


Abbildung 2.8: Korrelation des zu decodierenden Bits und eines Bits entgegengesetztem Wert

Die Korrelation einer ganzen PSSS-Sequenz ist in Abbildung 2.9 beispielhaft für eine 62 Half-Chips lange Sequenz nach PSSS IEEE 802.15.4-2011 gezeigt. Es ist zu sehen, dass die Korrelationsamplitude sich in jedem Chip abhängig von der Zusammensetzung der 1en und -1en verändert. Nach der Korrelation muss der Entscheider die Bits aus den Korrelationsamplituden C_K zurückgewinnen. Das Vorgehen hierfür ist nicht festgelegt.

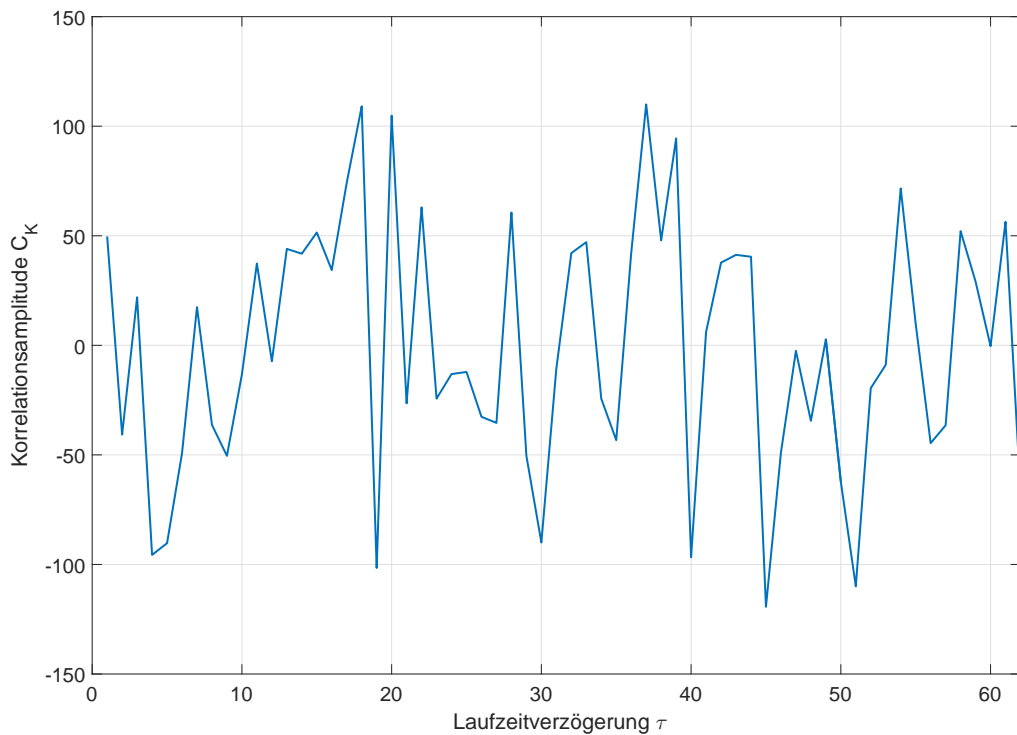


Abbildung 2.9: Korrelation einer PSSS-Sequenz nach IEEE 802.15.4-2011 bei 868 MHz

2.2.4 PSSS-ASK-PHY nach IEEE 802.15.4-2011

Der standardkonforme PSSS-ASK-PHY nach IEEE 802.15.4-2011 nutzt in Europa das 868 MHz Band für Short Range Devices (SRD) und in den USA das 915 MHz Industrial, Scientific and Medical (ISM) Band zur Übertragung. Es ist eine Datenrate von 250 kBit/s vorgesehen. Die Datenübertragung erfolgt paketweise in PHY Protocol Data Units (PPDUs), deren Aufbau in Abbildung 2.10 dargestellt ist.

Preamble	SFD	Frame Length	Reserved	PSDU
SHR		PHR		PHY Payload

Abbildung 2.10: PHY Protocol Data Unit nach IEEE 802.15.4-2011 [1]

Zunächst wird der Synchronisation Header (SHR) gesendet, welcher aus einer Präambel und einem Start-of-Frame Delimiter (SFD) besteht. Die Präambel dient der Synchronisation am Empfänger. Ihre Länge ist abhängig vom genutzten Frequenzband. Für 868 MHz wird zweimal Sequenz 0, also die Basissequenz, wiederholt, für 915 MHz sechsmal. Der SFD zeigt den Beginn des PHY Header (PHR) an und besteht aus der invertierten Sequenz 0. Die Sequenzen werden jeweils mitsamt der zyklischen Erweiterung gesendet. Sie sind in der PSSS-Code-Tabelle in Anhang A zu finden. Der PHR besteht aus einem 7 Bit langen Frame Length Feld, welches die Länge der PHY Payload (PSDU) angibt, und einem Reserved Bit. Anschließend folgt die PSDU, welche die Nutzdaten enthält. Der Ablauf der Codierung und Modulation der PDU ist in Abbildung 2.11 dargestellt.

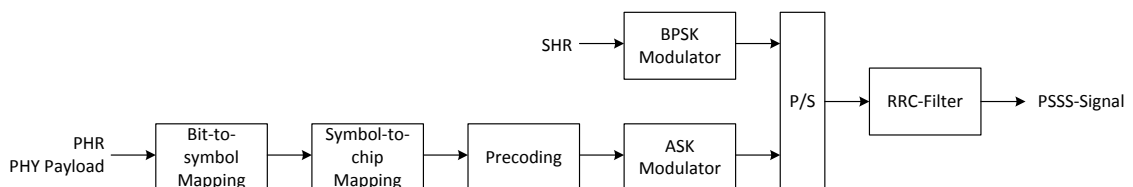


Abbildung 2.11: Codierung und Modulation der PDU nach IEEE 802.15.4-2011 [1]

Der SHR wird nicht wie der PHR und die PHY Payload PSSS-codiert, sondern lediglich mittels Binary Phase Shift Keying (BPSK) moduliert. Die PSSS-Codierung des PHR und der PHY Payload läuft nach folgendem Schema ab: Beim Bit-to-symbol Mapping werden für 868 MHz jeweils 20, für 915 MHz je 5 Bits zu einem Symbol zusammengefasst, um beim darauffolgenden Symbol-to-chip Mapping mit der zur jeweiligen Trägerfrequenz gehörenden PSSS-Code-Tabelle wie in Abbildung 2.5 multipliziert und anschließend aufaddiert zu werden.

Für 868 MHz ist die PSSS-Code-Tabelle eine 20×64 Matrix. Die Basissequenz ist eine bipolare M-Sequenz mit einer Länge von 31 Chips. Jeder Chip wird in zwei Half-Chips mit dem Wert

des jeweiligen Chips aufgeteilt, sodass eine 62 Half-Chips lange Sequenz entsteht. Durch eine zyklische Verschiebung der Basissequenz um drei Half-Chips werden die anderen 19 M-Sequenzen erzeugt. An jede Sequenz wird ein Guard Intervall angehängt, indem sie mit ihren ersten zwei Half-Chips zyklisch erweitert wird. Für 915 MHz ist die PSSS-Code-Tabelle eine 5×32 Matrix. Die Basissequenz wird so oft um sechs Chips verschoben, bis die Tabelle gefüllt ist. Anschließend wird jede Sequenz mit ihrem ersten Chip zyklisch erweitert.

Die durch das Symbol-to-Chip Mapping erzeugten PSSS-Sequenzen $p(m)$ haben Amplitudenwerte von bis zur Anzahl der verwendeten Sequenzen, welche im Folgenden als Symbollänge S bezeichnet wird. Deshalb werden die PSSS-Sequenzen zur leistungseffizienteren Übertragung zwei Precoding-Schritten unterzogen. Im ersten Schritt wird $p(m)$ so verschoben, dass die maximale Amplitude A_{\max} und die minimale Amplitude A_{\min} symmetrisch um Null liegen [16].

$$p'(m) = p(m) - \frac{A_{\max} - A_{\min}}{2} \quad (2.5)$$

Im zweiten Schritt wird die maximale Amplitude A'_{\max} von $p'(m)$ auf den Wert Eins normiert.

$$p''(m) = \frac{p'(m)}{A'_{\max}} \quad (2.6)$$

Nach dem Precoding wird das PSSS-Signal einer ASK Modulation unterzogen und anschließend zusammen mit dem SHR mit einem Root-Raised Cosine (RRC) Filter mit einem Roll-Off-Faktor von $r = 0.2$ gefiltert.

Nach der Übertragung wird das Signal im Empfänger verarbeitet, dessen schematischer Aufbau in Abbildung 2.12 gezeigt ist. Das Empfangsfilter ist als Matched-Filter zum Sendefilter entworfen. Danach wird das Signal demoduliert. Das Precoding kann nur teilweise rückgängig gemacht werden, da dem Empfänger die ursprüngliche Amplitude des Signals nicht bekannt ist. Im Decoder wird das Signal erneut mit der PSSS-Code-Tabelle multipliziert. Auf welche Art die Informationsbits aus den entstandenen Korrelationsergebnissen zurückgewonnen werden, ist im Standard nicht festgelegt [1].

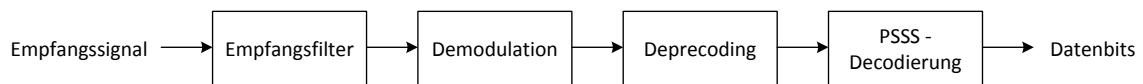


Abbildung 2.12: Schematischer Aufbau des Empfängers

2.3 Code Division Multiple Access

Um eine Übertragung mehrerer Teilnehmer zu einem Master zu ermöglichen, muss ein Multiplexverfahren verwendet werden. Hierbei wird zwischen Frequency Division Multiple Access (FDMA), TDMA und CDMA unterschieden. Das FDMA-Verfahren ordnet jedem Teilnehmer eine andere Trägerfrequenz zu, wodurch der Empfänger in der Lage ist, die Signale zu trennen. Beim TDMA-Verfahren übertragen die Teilnehmer ihre Daten in verschiedenen Zeitabschnitten. Beim synchronen TDMA identifiziert der Empfänger den Teilnehmer anhand seiner festen Position auf dem Übertragungskanal. Hierdurch bleiben Zeitabschnitte unter Umständen ungenutzt. Beim asynchronen TDMA hingegen werden die Zeitabschnitte nach Bedarf zugeordnet, dadurch müssen allerdings Kanalinformationen mitgesendet werden, sodass ein Overhead entsteht. Beim CDMA-Verfahren erhält jeder Teilnehmer eine PN-Sequenz, mit der seine Nutzdaten gespreizt werden. Die Datenströme aller Teilnehmer werden addiert und gemeinsam übertragen. Es wird zwischen synchronem und asynchronem CDMA unterschieden. Im synchronen Fall müssen alle Teilnehmer ihre Codes zeitlich exakt gleich ausrichten, damit der Empfänger durch Korrelation mit den PN-Sequenzen die ursprünglichen Nutzdaten wieder voneinander trennen kann. Hierbei werden streng orthogonale Sequenzen verwendet, wodurch gegenseitiges Übersprechen zwischen den Teilnehmern minimiert wird. Im asynchronen Fall werden nur fast orthogonale Sequenzen genutzt, um auf Kosten der Störfestigkeit eine höhere Teilnehmerzahl zu erlangen. Da die Sequenzen eine zeitliche Verschiebung aufweisen, wird im Empfänger die ungerade Kreuzkorrelation, die Verschiebungen zwischen den Sequenzen erlaubt, zur Rückgewinnung der Information genutzt. Abbildung 2.13 zeigt das Decodierverfahren bei CDMA. Anwendung findet das CDMA-Verfahren bei GPS und dem Mobilfunkstandard UMTS [8].

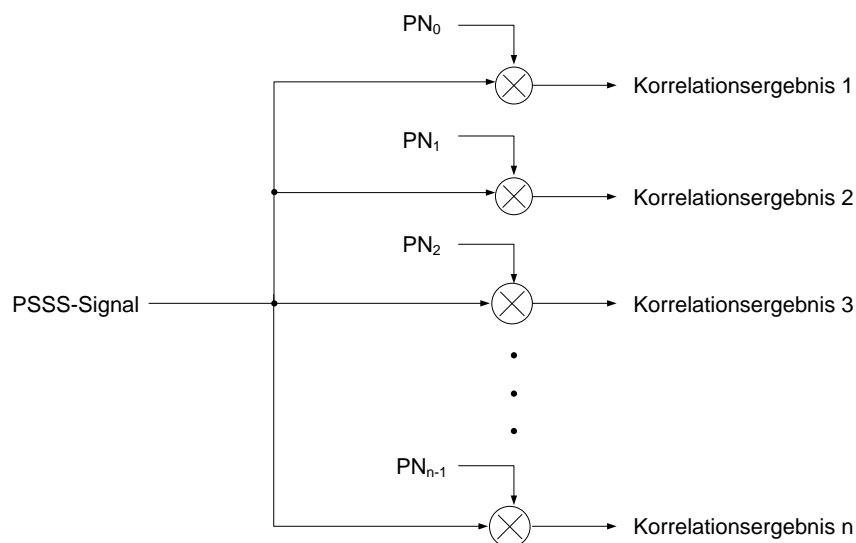


Abbildung 2.13: Decodierung eines CDMA-Signals

2.4 Frequenz- und Phasenoffset

Als Frequenzoffset Δf wird der Unterschied zwischen nominaler Trägerfrequenz f_0 und dem tatsächlichen Ausgangssignal f_c des Oszillators bezeichnet, der das Trägersignal erzeugt [10]. Im Basisband wird ein konstanter Frequenzoffset Δf durch

$$r(t) = s(t) \cdot \exp(j2\pi \cdot \Delta f \cdot t) \quad (2.7)$$

beschrieben. Die effektive Trägerfrequenz beträgt also $f_0 + \Delta f$. Ursachen für einen Frequenzoffset können zum Beispiel die Ungenauigkeit eines Schwingquarzes sein oder bei bewegten Sendern bzw. Empfängern die Dopplerverschiebung [6].

Bei PSSS nach IEEE 802.15.4-2011 beträgt die maximale Toleranz für die Trägerfrequenz 40 ppm. Für eine Trägerfrequenz $f_0 = 868$ MHz beträgt der Frequenzoffset dann

$$\begin{aligned} \Delta f_{\max,1} &= f_0 \cdot 40 \text{ ppm} \\ &= 34.72 \text{ kHz,} \end{aligned} \quad (2.8)$$

für $f_0 = 915$ MHz

$$\begin{aligned} \Delta f_{\max,2} &= f_0 \cdot 40 \text{ ppm} \\ &= 36.60 \text{ kHz.} \end{aligned} \quad (2.9)$$

Eine konstante Phasendrehung wird mit

$$r(t) = s(t) \cdot \exp(j \cdot \Delta \varphi). \quad (2.10)$$

modelliert. Diese kann auftreten, wenn die Schätzung der Trägerphase am Empfänger nicht funktioniert. Im Basisband entspricht diese Phasendrehung einer zeitlichen Verschiebung. Dies gilt allerdings nicht für trägerfrequente Signale wie PSSS-Signale.

3 Konzeptionierung

In diesem Kapitel werden Vorüberlegungen zur Umsetzung der Simulation von PSSS als CDMA-Verfahren getroffen. Zunächst wird das Verfahren erläutert. Aufgrund der unterschiedlichen Empfangsleistungen der verschiedenen Teilnehmer muss die Herleitung der BER entsprechend erweitert werden. Des Weiteren werden die Simulationsszenarien zur Untersuchung der Auswirkungen unterschiedlicher Laufzeiten, Empfangsleistungen und Sendefrequenzen aufgezählt.

Da für die Decodierung des PSSS-Signals keine Vorschrift existiert, wird aufbauend aus dem Decodierverfahren aus [11] ein weiterer Decodierschritt hinzugefügt, der zur Verbesserung der BER beitragen soll.

3.1 PSSS als Codemultiplex

Das PSSS-255-Verfahren ist eine erweiterte Form des PSSS IEEE 802.15.4-2011 mit sternförmiger Topologie, bei dem im Uplink mehrere Teilnehmer nicht mehr nacheinander per TDMA, sondern gleichzeitig per CDMA senden können, wie in Abbildung 3.1 zu sehen. Der Downlink gleicht dem PSSS IEEE 802.15.4-2011 Verfahren. Allerdings kann der Master hier in einem Symbol Daten für mehrere Teilnehmer gleichzeitig senden. Diese werten nur die für sie selbst bestimmten Informationen aus.

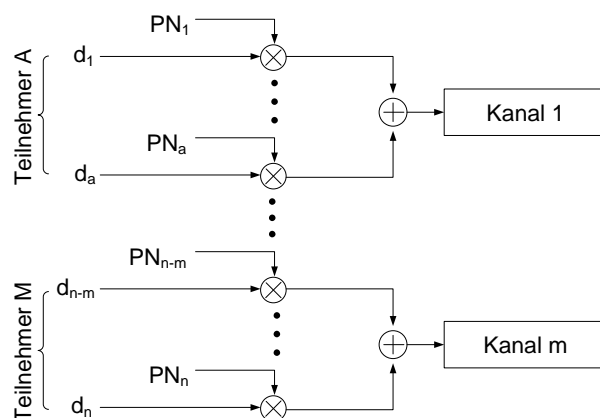


Abbildung 3.1: Symbol-to-chip Mapping im Uplink von PSSS-255

Die Nutzung von CDMA bietet den Vorteil einer hohen Flexibilität in Bezug auf die momentan zu sendende Datenmenge der einzelnen Teilnehmer und einer daraus resultierenden geringen Latenz, da keine ungenutzten Zeitabschnitte existieren.

Gleichzeitig ergeben sich, wie in Abbildung 3.2 dargestellt, neue Herausforderungen wie unterschiedliche Empfangsleistungen P_E und Laufzeitverzögerungen τ der Teilnehmer durch unterschiedliche Sender-Empfänger Distanzen sowie Abweichungen von der Sendefrequenz Δf bedingt durch Ungenauigkeiten der Oszillatoren im Sender.

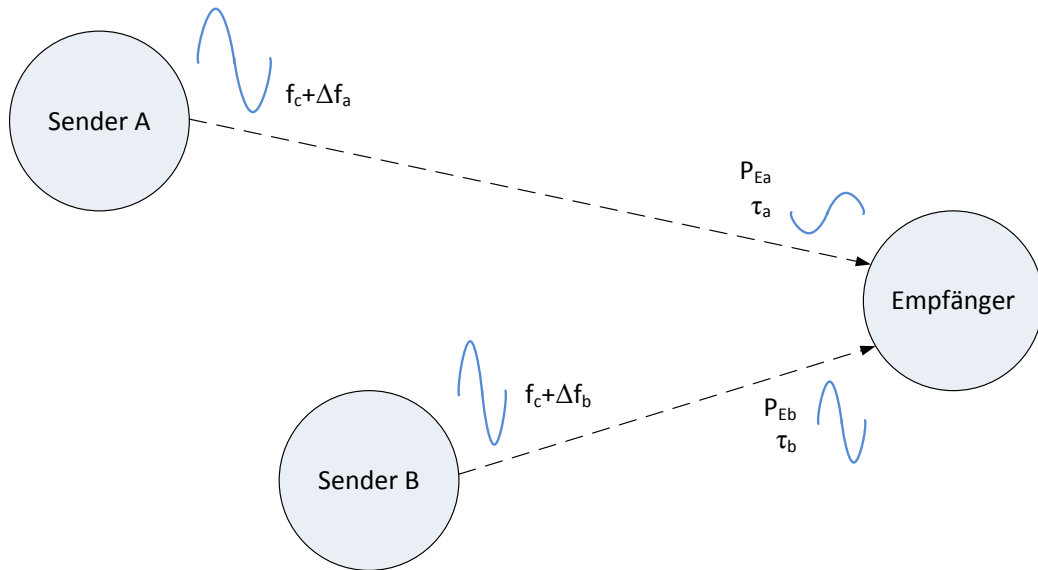


Abbildung 3.2: Herausforderungen bei PSSS als CDMA

Für PSSS-255 ist keine Trägerfrequenz vorgegeben. Das Verfahren wird für eine Trägerfrequenz von 5.8 GHz untersucht, da dieses Frequenzband exklusiv für den Industriefunk reserviert werden kann. Hierfür ist eine Chiprate von $f_{\text{Chip}} = 50 \text{ MChip/s}$ vorgegeben. Die Bitrate R ist somit abhängig von der Länge des Guard Intervalls und der Anzahl der verwendeten Sequenzen, welche auch hier als Symbollänge S bezeichnet wird.

$$R = f_{\text{Chip}} \cdot \frac{S}{n_R + GI} \quad (3.1)$$

Das Blockschaltbild der Codierung, welches in Abbildung 3.3 zu sehen ist, ähnelt dem in Abschnitt 2.2.3 beschriebenen Ablauf bei PSSS IEEE 802.15.4-2011, allerdings unterscheidet sich die Verarbeitung der Daten innerhalb der Blöcke bei beiden Verfahren stark.

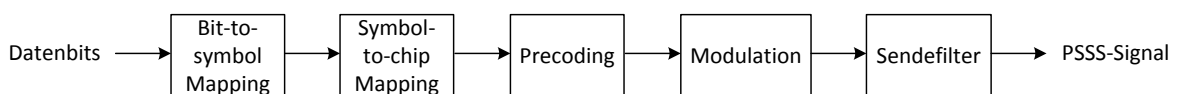


Abbildung 3.3: Codierung und Modulation der Nutzdaten bei PSSS-255

Die PSSS-Code-Tabelle \underline{D} wird aus einer 255 Chip langen Basissequenz gebildet. Die Weite der zyklischen Verschiebung zur Erzeugung der weiteren M-Sequenzen ist variabel. In dieser Arbeit wird jedoch meistens eine Verschiebung von einem Chip genutzt, da das Ziel die Nutzung der maximal möglichen Anzahl an Sequenzen ist. An die entstandenen Sequenzen wird ein Guard Intervall in Form einer zyklischen Erweiterung von 30 Chips angehängt, so dass eine 255×285 Matrix entsteht. Generell ist die Länge des Guard Intervalls variabel. Der genutzte Wert wurde aufgrund einer Abschätzung der maximalen Umweglänge in einer Industrieumgebung gewählt. Tabelle 3.1 zeigt die mit einem Guard Intervall von 30 Chips maximal korrigierbare Umweglängen für die vorgegebene Chiprate von $f_{\text{Chip}} = 50$ MChip/s und für $f_{\text{Chip}} = 20$ MChip/s, da unter Umständen lediglich diese Chiprate realisierbar ist.

Chiprate	maximale Umweglänge
50 MChip/s	180 m
20 MChip/s	450 m

Tabelle 3.1: maximale Umweglängen für ein 30 Chips langes Guard Intervall

Beim Bit-to-symbol Mapping werden die Sequenzen unter den Teilnehmern aufgeteilt, die einen Sendewunsch haben. Hierbei kann ein Teilnehmer mehrere PSSS-Code-Sequenzen gleichzeitig belegen. Die Informationsbits der Teilnehmer werden nun in Symbole aufgeteilt, deren Länge der Anzahl der dem Teilnehmer zugewiesenen Sequenzen entspricht. Es kann sein, dass beim Symbol-to-chip Mapping einige Codesequenzen ungenutzt bleiben. Dies sind hier immer die letzten Sequenzen in der Code-Tabelle. In Abbildung 3.1 ist das Symbol-to-chip Mapping dargestellt. Teilnehmer A bis M nutzen jeweils eine bestimmte Anzahl n_A bis n_m an Code-Sequenzen zum Übertragen ihrer Information. Die Bits werden mit den Codesequenzen gespreizt und je Teilnehmer alle entstandenen Sequenzen zu einem PSSS-Signal aufsummiert. Da alle Teilnehmer ihre PSSS-Signale über verschiedene Kanäle senden, ist kein gemeinsames Precoding möglich. Allerdings kann ein einzelnes Precoding für jeden Teilnehmer durchgeführt werden. Anschließend werden die PSSS-Sequenzen moduliert und gefiltert. Auch hierfür gibt es keine Vorgabe. In dieser Arbeit wird eine BPSK-Modulation vorgenommen. Die bipolaren Bits werden anschließend mit einem RRC-Filter mit einem Roll-Off-Faktor von $r = 1$ gefiltert.

Da jeder Teilnehmer einen bestimmten Abstand zum Empfänger hat, kommen dort alle PSSS-Signale mit unterschiedlichen Laufzeiten τ und Empfangsleistungen P_E an. Außerdem variiert die Sendefrequenz aufgrund von Toleranzen der Schwingquarze um bis zu 10 ppm. Das Empfangssignal ist die Summe der PSSS-Signale der Teilnehmer. Der Aufbau des Empfängers entspricht dem in Abbildung 2.12. Nach Filterung mit Matched-Filter und Demodulation wird das Signal erneut mit der PSSS-Code-Tabelle korreliert. Die Rückgewinnung der Informationsbits aus den Korrelationsergebnissen ist auch für PSSS-255 nicht festgelegt. Hier wird

die Decodierung in zwei Schritten durchgeführt. Zunächst wird eine einfache Schwellwertentscheidung vorgenommen. Oberhalb der Nullschwelle wird eine 1, unterhalb eine –1 decodiert. Im zweiten Schritt werden abhängig von der Anzahl der decodierten 1en und –1en pro Symbol neue Schwellen festgelegt. Das Vorgehen wird in Abschnitt 3.3 näher erläutert.

3.2 Herleitung der Bitfehlerrate für Sequenzen verschiedener Leistung

In der vorangegangenen Bachelorarbeit ([11]) wurde bereits die Bitfehlerrate (BER) für das PSSS Verfahren nach IEEE 802.15.4-2011 hergeleitet. Diese Herleitung gilt allerdings nur, wenn alle gesendeten Bits die gleiche Leistung haben. Werden mehrere Teilnehmer mit unterschiedlichen Leistungen betrachtet, so müssen grundsätzliche Erweiterungen vorgenommen werden. Deshalb erfolgt hier die Herleitung der Bitfehlerrate für mehrere Teilnehmer mit unterschiedlicher Empfangsleistung. Die BER wird unter Vernachlässigung der zyklischen Erweiterung und eventuellen Laufzeitverzögerungen τ oder Frequenzoffsets Δf einzelner Teilnehmer hergeleitet.

Die Abhängigkeit der Korrelationsamplitude C_K von der Anzahl an Bits mit gleichem Wert wie das zu decodierende Bit in einem Symbol wurde bereits in Abschnitt 2.2.3 erläutert. Die Wahrscheinlichkeit $P(n)$ für das Auftreten von n Bits mit gleichem Wert bei einer Symbollänge von S ist in Abbildung 3.4 dargestellt. Sie entspricht einer Binomialverteilung und wird wie folgt berechnet:

$$P(n) = \frac{1}{2^{S-1}} \cdot \binom{S-1}{n} \quad (3.2)$$

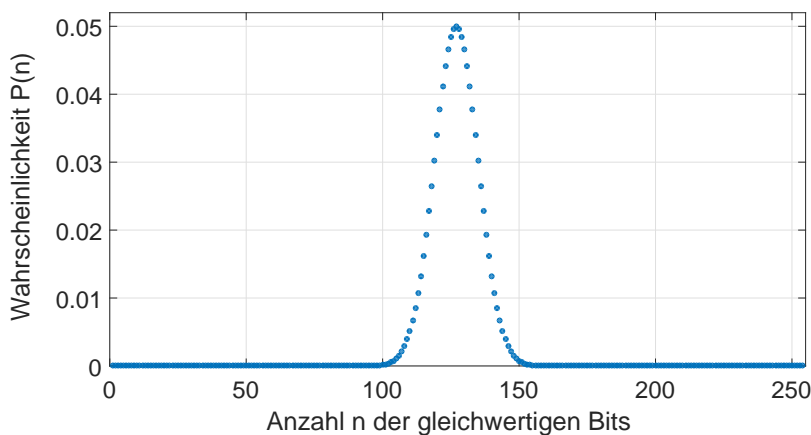


Abbildung 3.4: Wahrscheinlichkeiten für die Anzahl an Bits mit gleichem Wert für $S = 255$

Für den Fall, dass es mehrere Teilnehmer mit unterschiedlichen Empfangsleistungen gibt, muss $P(n)$ für die Sequenzen jedes Teilnehmers einzeln berechnet werden. Die Berechnung der BER erfolgt hier für zwei Teilnehmer A und B. Teilnehmer A wird hierbei als Träger (Carrier) des gewünschten, zu empfangenen Signals mit der Gesamtleistung P_C und Teilnehmer B als Störer (Interferer) mit der Gesamtleistung P_I betrachtet. Teilnehmer A nutzt n_A , Teilnehmer B n_B Sequenzen, wobei $n_A + n_B \leq n_R$ sein muss. P_A und P_B sind die Leistungen für jede Sequenz des entsprechenden Teilnehmers. Somit beträgt

$$P_C = P_A \cdot n_A \text{ und} \quad (3.3)$$

$$P_I = P_B \cdot n_B. \quad (3.4)$$

Eine Herleitung für mehr Teilnehmer kann analog erfolgen, verkompliziert aber unnötig die Kernaussage und ist deshalb nicht Bestandteil dieser Arbeit.

Das Verhältnis der Gesamtleistungen wird als Carrier to Interferer Ratio (CIR)

$$\begin{aligned} CIR &= \frac{P_A \cdot n_A}{P_B \cdot n_B} \\ &= \frac{P_C}{P_I} \end{aligned} \quad (3.5)$$

angegeben. Beispielhaft für die Decodierung eines Bits des gesendeten Symbols wird die BER des ersten Bits von Teilnehmer A berechnet. Die Berechnung für alle anderen Bits erfolgt analog. Für den Fall, dass Teilnehmer A eine 1 gesendet hat, lautet die Wahrscheinlichkeit, dass n_{A1} weitere Einsen von Teilnehmer A gesendet wurden

$$P(n_A, n_{A1}) = \frac{1}{2^{n_A-1}} \cdot \binom{n_A-1}{n_{A1}}. \quad (3.6)$$

Die Wahrscheinlichkeit, dass Teilnehmer B n_{B1} Einsen gesendet hat, lautet

$$P(n_B, n_{B1}) = \frac{1}{2^{n_B}} \cdot \binom{n_B}{n_{B1}}. \quad (3.7)$$

Es wird angenommen, dass die Leistung von Teilnehmer A zuvor auf 1 normiert wurde. Dann kann die Korrelationsamplitude von Teilnehmer A als die Summe der Korrelationen aller von ihm genutzten Sequenzen an der Stelle τ mit der betrachteten ersten Sequenz berechnet werden:

$$C_A(n_A, n_{A1}) = n_R + n_A - 1 - 2 \cdot n_{A1} \quad (3.8)$$

Damit E_b/N_0 im Mittel unverändert bleibt, muss der Erwartungswert der Leistung

$$E\{C_A^2(n_A, n_{A1})\} = 1 \quad (3.9)$$

betragen. Dazu muss $C_A(n_A, n_{A1})$ über $P(n_A, n_{A1})$ gemittelt werden. Die normierte Korrelationsamplitude beträgt somit

$$C_{AK}(n_A, n_{A1}) = \frac{C_A(n_A, n_{A1})}{\sqrt{C_A^2(n_A, n_{A1})}^{n_{A1}}} \quad (3.10)$$

$$= \frac{n_R + n_A - 1 - 2 \cdot n_{A1}}{\sqrt{\sum_{n_{A1}=0}^{n_A-1} \frac{1}{2^{n_A-1}} \cdot \binom{n_A-1}{n_{A1}} \cdot (n_R + n_A - 1 - 2 \cdot n_{A1})^2}} \quad (3.11)$$

Hierbei gilt $0 \leq n_{A1} < n_A \leq n_R$. Das Korrelationsergebnis von Teilnehmer B lautet analog

$$C_B(n_B, n_{B1}) = n_B - 2 \cdot n_{B1}. \quad (3.12)$$

Da hier nur verschobene Sequenzen miteinander korreliert werden, fehlt der Term n_R . Dieser muss allerdings zur einheitlichen Normierung im Nenner hinzugefügt werden.

$$C_{BK}(n_B, n_{B1}) = \frac{C_B(n_B, n_{B1})}{\sqrt{(C_B(n_B, n_{B1}) + n_R)^2}^{n_{B1}}} \quad (3.13)$$

$$= \frac{n_B - 2 \cdot n_{B1}}{\sqrt{\sum_{n_{B1}=0}^{n_B} \frac{1}{2^{n_B}} \cdot \binom{n_B}{n_{B1}} \cdot (n_R + n_B - 2 \cdot n_{B1})^2}} \quad (3.14)$$

Für das gesamte Korrelationsergebnis werden C_A und C_B unter Berücksichtigung der unterschiedlichen Leistungen von Teilnehmer A und B addiert:

$$C_K(n_A, n_B, n_{A1}, n_{B1}) = C_{AK}(n_A, n_{A1}) + \sqrt{\frac{P_B}{P_A}} \cdot C_{BK}(n_B, n_{B1}) \quad (3.15)$$

Dieser Ausdruck kann bei schlechtem CIR negativ werden. $C_K(n_A, n_B, n_{A1}, n_{B1})$ wird im Folgenden mit C_K abgekürzt. Für eine bestimmte Kombination $(n_A, n_B; n_{A1}, n_{B1})$ ergibt sich eine Bit Error Rate von

$$BER(n_A, n_B; n_{A1}, n_{B1}) = \begin{cases} \frac{1}{2} \cdot \operatorname{erfc}(C_K \cdot \sqrt{\frac{E_b}{N_0}}) & \text{wenn } C_K \geq 0 \\ 1 - \frac{1}{2} \cdot \operatorname{erfc}(|C_K| \cdot \sqrt{\frac{E_b}{N_0}}) & \text{sonst.} \end{cases} \quad (3.16)$$

Der zweite Fall berücksichtigt, dass bei schlechtem CIR die Korrelationsamplitude das Vorzeichen wechseln kann. Allerdings besteht bei schlechtem E_b/N_0 wiederum die Möglichkeit, dass

durch ausreichend hohe Rauschleistung das falsche Vorzeichen zurück ins richtige wechselt.

Zur Berechnung der Gesamtfehlerrate müssen die $BER(n_A, n_B; n_{A1}, n_{B1})$ aller Kombinationen $(n_{A1}, n_{B1}) \in \{1, 2, \dots, n_R\} \times \{0, 1, \dots, n_R - 1\}$ mit den Wahrscheinlichkeiten $P(n_A, n_{A1})$ und $P(n_B, n_{B1})$ gewichtet werden.

n_A und n_B sind für eine bestimmte Kombination konstant. Da n_{A1} und n_{B1} statistisch unabhängig sind, kann die Gesamtwahrscheinlichkeit als das Produkt der Wahrscheinlichkeiten angegeben werden:

$$P(n_A, n_{A1}; n_B, n_{B1}) = P(n_A, n_{A1}) \cdot P(n_B, n_{B1}) \quad (3.17)$$

Somit lautet die Gesamtbitfehlerrate:

$$BER(n_A, n_B) = \sum_{i=0}^{n_A-1} \sum_{j=0}^{n_B} \frac{\binom{n_A-1}{i} \cdot \binom{n_B}{j}}{2^{n_A-1+n_B}} \cdot BER(n_A, n_B; n_{A1}, n_{B1}) \quad (3.18)$$

Für PSSS-255 wurde mit der Formel die BER-Kurve für $n_R = 255$, $CIR = 1$, $n_A = 128$ und $n_B = 127$ berechnet. Diese ist in Abbildung 3.5 zu sehen. Zur besseren Einordnung ist außerdem die BER der bipolaren BPSK dargestellt.

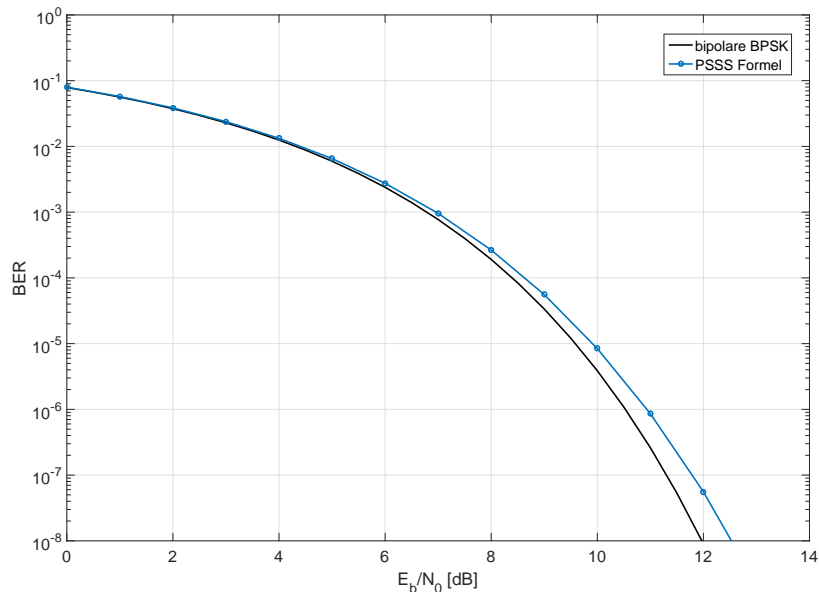


Abbildung 3.5: BER-Kurve für eine Sequenz von Teilnehmer A, $n_A = 128$ und $n_B = 127$, $CIR = 1$.

In Abbildung 3.6 ist ein Spezialfall dargestellt: Zwei Teilnehmer nutzen je eine Sequenz zur Übertragung. Bei einem sehr geringen CIR führt ein Bit des Störers mit gleichem Wert, wie in Abschnitt 2.2.3 beschrieben, dazu, dass das Decodierergebnis C_K des anderen Teilnehmers

in jedem Fall verfälscht wird. Liegt C_K nah an der Entscheidungsschwelle, so kann das AWGN-Rauschen des Übertragungskanals dazu führen, dass das Ergebnis wieder korrigiert wird. Die Wahrscheinlichkeit dafür steigt, je geringer das Signal-Rausch-Verhältnis ist. Somit steigt hier die Bitfehlerrate mit dem SNR. Dieses scheinbar paradoxe Verhalten existiert nicht für eine Übertragung mit vielen Sequenzen. Da die Anzahl der gleichwertigen Bits binomial verteilt ist, ist die Wahrscheinlichkeit dafür, dass alle Bits den gleichen Wert haben, nahezu Null.

Eine dreidimensionale Darstellung der BER über E_b/N_0 und dem CIR erfolgt in Abschnitt 5.3 zum Vergleich mit der simulierten Kurve.

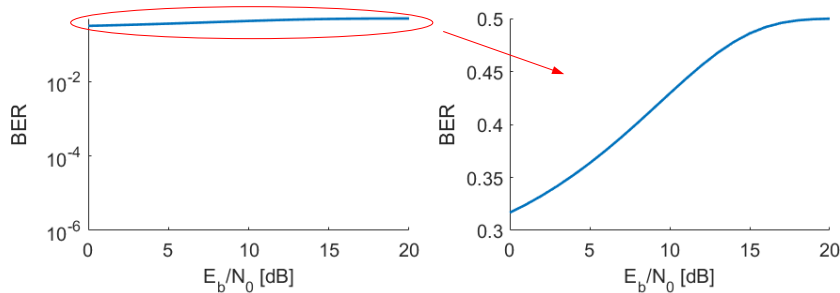


Abbildung 3.6: BER von Teilnehmer A, $n_A = 1$, $n_B = 1$, CIR = -50 dB

3.3 Dynamische Schwellwertanpassung im Decoder

Die Bits eines PSSS-Symbols beeinflussen sich gegenseitig linear mit einem Faktor von $1/n_R$, da eine nicht orthogonale Signalisierung vorliegt. Dadurch wird die Korrelationsamplitude C_K und einen Wert n_s verschoben, wie in Abschnitt 2.2.3 gezeigt. Dieser Effekt ähnelt einem echobehafteten Kanal mit ISI und wird hier Inter Code Interference (ICI) genannt. Um eine Verbesserung der Bitfehlerrate zu erreichen, soll im Empfänger eine nachträgliche Orthogonalisierung durchgeführt werden, durch die die ICI entfernt wird.

Eine Möglichkeit dazu bietet eine Entzerrung mit einer Art Zero Forcing. Hierbei handelt es sich um eine Filterung mit der inversen Übertragungsfunktion, um die Echos der Impulsantwort zu eliminieren. In dieser Arbeit soll erreicht werden, dass die Verschiebung n_s der Korrelationsamplitude C_K zu Null gesetzt wird.

Abbildung 3.7 zeigt den Ablauf des PSSS-Verfahrens mit Entzerrung. Das Signal wird zunächst mit der PSSS-Code-Tabelle \underline{D} multipliziert und über einen AWGN-Kanal übertragen. In diesem Abschnitt wird angenommen, dass \underline{D} derart normiert ist, dass sie für die redundanzfreien Daten b im Mittel leistungsinvariant sind. Dies wird erreicht, indem sich in der Matrix nicht nur die zeilenweise zyklisch verschobenen M-Sequenzen mit den Werten ± 1 befinden sondern die Werte $\pm 1/\sqrt{n_R}$. Damit gilt $\underline{D} \cdot b_2 = b'_2$. Damit ist die Signalleistung nach der

Codespreizung am Ausgang des Senders exakt die selbe wie ohne Codespreizung und es ergibt sich ein fairer Vergleich mit der BPSK.

Die Korrelation am Empfänger entspricht einer Multiplikation mit \underline{D}^T . Diese ist mit dem gleichen Faktor normiert wie \underline{D} . Um eine Orthogonalisierung zu erreichen, muss das Signal nun mit einer Entzerrungsmatrix \underline{C} multipliziert werden, welche die Inverse von $(\underline{D} \cdot \underline{D}^T)$ sein muss. Somit ist das Ergebnis die Einheitsmatrix I

$$(\underline{D} \cdot \underline{D}^T) \cdot \underline{C} \stackrel{!}{=} I, \quad (3.19)$$

womit auch die gesamte Strecke für die Daten leistungsnormiert ist.

Für eine ideale rauschfreie Übertragung ist das empfangene Signal nach der Entzerrung gleich dem gesendeten Signal, da die Inter Code Interference komplett eliminiert wird.

Das Rauschen des AWGN-Kanals durchläuft nur \underline{D}^T und \underline{C} . Somit ist

$$E\{(\underline{D} \cdot n_{\text{AWGN}})^2\} = 2 \cdot E\{n_{\text{AWGN}}^2\} \quad (3.20)$$

Damit ergibt sich zwar ein ICI-freies Symbol, aber das Rauschen ist um den Faktor 2 erhöht. Da an der BER-Kurve des PSSS-Systems bei BER-Werten von 10^{-8} nur maximal eine Verschlechterung um 1 dB im Gegensatz zur BPSK auftritt, würde sich durch die lineare ICI-Entzerrung eine Verschlechterung um 2 dB ergeben. Wie zu vermuten, hat sich das Zero-Forcing als unterlegen herausgestellt.

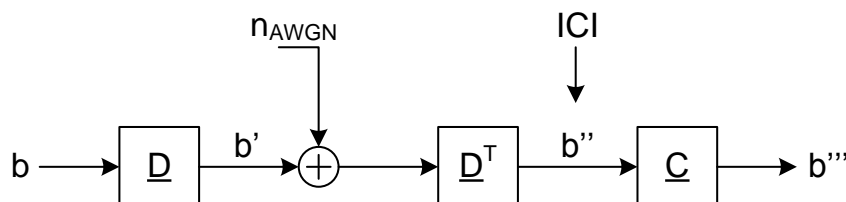


Abbildung 3.7: Entzerrung mit Zero Forcing

Eine Alternative bietet das Verfahren der Maximum Likelihood Sequence Estimation (MLSE). Hierbei wird der Kanal vom Empfänger nachgebildet und aus allen möglichen gesendeten Signalen dasjenige ausgewählt, für welches sich die geringste Abweichung zum empfangenen Signal ergibt. Da hier alle S Bits des Symbols gemeinsam decodiert werden, ist diese Methode mit 2^S möglichen Symbolen zu aufwändig.

Aus diesem Grund wird die Orthogonalisierung nach dem Ansatz eines Decision Feedback Equalizers (DFE) durchgeführt. Der DFE bestimmt den Fehler des decodierten Symbols und

subtrahiert diesen vom Symbol. Dies entspricht einer Verschiebung der Korrelationsamplituden C_K um den Wert n_s . Anschaulicher ist allerdings eine Verschiebung der Entscheidungsschwellen um denselben Wert. Dieses Verfahren ist in Abbildung 3.8 dargestellt [12].

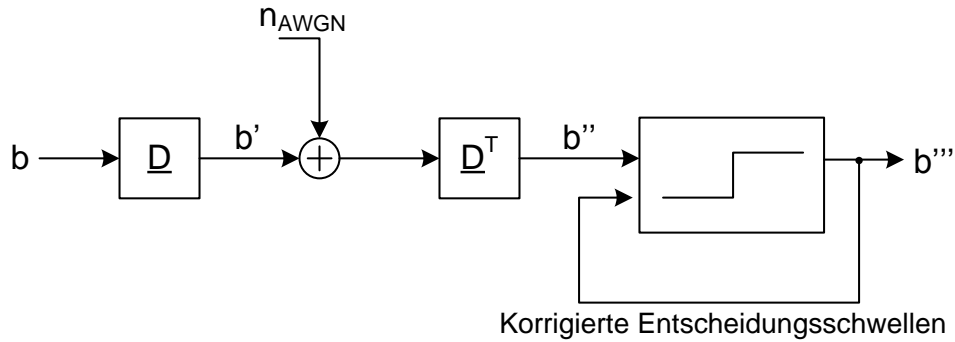


Abbildung 3.8: Entzerrung mit Decision Feedback

Für jedes Bit des Symbols wird die Anzahl der 1en n'_1 und 1en n'_{-1} des im ersten Decodierschritt bestimmten Symbols b'' berechnet. Das jeweils betrachtete Bit wird dabei nicht mitgezählt, da dessen Wert im zweiten Decodierschritt erst bestimmt werden soll. Die resultierende Verschiebung beträgt $n_s' = -n'_1 + n'_{-1}$. Im ersten Decodierschritt wurde die Nullschwelle zur Entscheidung herangezogen. Diese wird nun um n_s verschoben und eine erneute Decodierung des Signals b'' durchgeführt. Dieser Schritt kann mehrfach hintereinander durchgeführt werden. Für eine geringe Fehlerzahl wird dadurch eine Orthogonalisierung des Signals erreicht.

In der Praxis hat sich gezeigt, dass diese Rückkopplung und mehrfache Dekodierung im Sinne eines Decision Feedback gar nicht notwendig ist. Eine Feed-Forward Decision, wie in Abbildung 3.9 zu sehen, zeigt bereits im Bereich relevanter Werte für die BER kaum noch Abweichungen von der Performanz der BPSK:

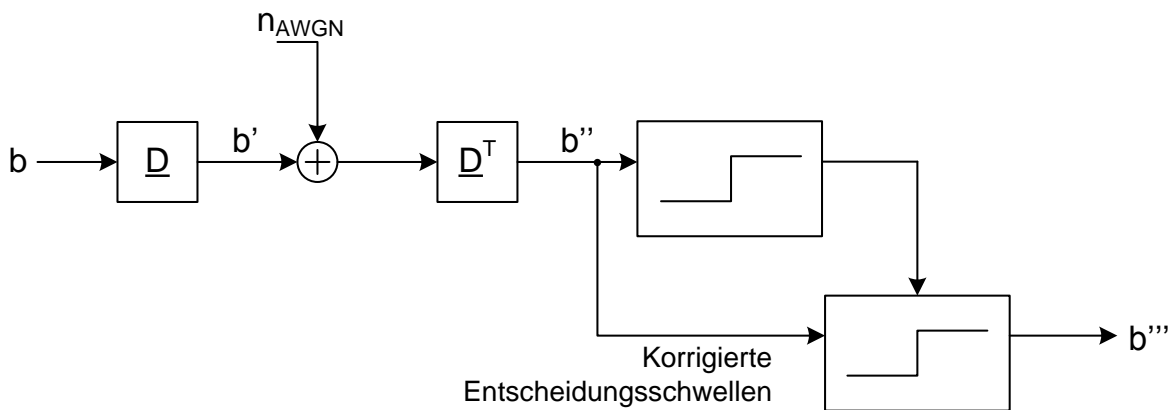


Abbildung 3.9: Entzerrung mit Feed Forward

Der Grund für die gute Performanz der Feed-Forward Decision ist, dass bei Vernachlässigung von Fehlern des ersten Entscheiders tatsächlich eine Orthogonalisierung entsteht und im Bereich der relevanten Bitfehlerwahrscheinlichkeiten von $BER < 10^{-4}$ die Anzahl der im ersten Schritt falsch decodierten Bits pro Symbol im Mittel weniger als 1 beträgt. Damit wird die Schwelle aber auch nur um den Betrag eines Bits falsch korrigiert. In Summe wird die Schwelle von den anderen Bits aber immer noch in die richtige Richtung korrigiert, sodass die Entscheidung im zweiten Schritt sehr oft das endgültig richtige Ergebnis liefert.

3.4 Simulationsszenarien

In der Simulation werden die Bitfehlerkurven für die Übertragung über einen Additive White Gaussian Noise (AWGN)-Kanal mit dem PSSS-255-Physical Layer (PHY) für Signal-Rausch-Abstände zwischen 0 und 20 dB simuliert. Zunächst erfolgt die Simulation des PSSS-255-Verfahrens unter idealen Bedingungen, also ohne Laufzeitunterschiede $\Delta\tau$, Frequenzoffset Δf und für $CIR = 0$. Hierbei wird der Einfluss der Anzahl der genutzten M-Sequenzen auf die Bitfehlerrate einbezogen. Anhand der in Abschnitt 3.2 berechneten theoretischen Bitfehlerkurven wird das Verfahren validiert.

Als nächstes werden die Auswirkungen unterschiedlicher Empfangsleistungen zwischen den Teilnehmern untersucht. Hierzu senden zur besseren Erkennbarkeit der Auswirkungen zunächst zwei Teilnehmer A und B jeweils ein Bit. Hierbei wird die von Teilnehmer A gesendete Sequenz als gewünschte, zu empfangene Sequenz (Carrier) und die von Teilnehmer B gesendete Sequenz als Störer (Interferer) angesehen. Das Verhältnis der Leistungen wird als CIR angegeben. Anhand der Herleitung aus Abschnitt 3.2 lässt sich auch diese Simulation validieren. Anschließend wird der Einfluss der Anzahl der Störer auf die zu empfangene Sequenz bei konstantem CIR , also konstanter Gesamtstörleistung, betrachtet. Dazu wird P_B gleichmäßig unter den Störern aufgeteilt. Es werden vom CIR abhängige, dreidimensionale Bitfehlerkurven erstellt.

In der nächsten Simulation wird der Einfluss von Laufzeitverzögerungen untersucht. Auch hier werden zunächst nur zwei bzw. drei Sequenzen betrachtet, welche um einen Chip verschoben sind. Es werden Verzögerungen zwischen 0 und 1 Chip in 1/10 Chip-Schritten betrachtet. Im Fall von zwei genutzten Sequenzen wird nur die erste Sequenz verschoben, im Fall von drei Sequenzen werden die erste und die dritte Sequenz jeweils in Richtung der zweiten, in der Mitte liegenden Sequenz verschoben. Auch hier werden dreidimensionale Bitfehlerkurven erstellt.

Weiterhin wird ein Phasen- bzw. Frequenzoffset eines Teilnehmers simuliert. Es wird deren Einfluss sowohl auf die Sequenzen des betroffenen, als auch auf die der nicht betroffenen Teilnehmer betrachtet.

Tabelle 3.2 zeigt eine Übersicht der betrachteten Simulationsszenarien. Die verschiedenen Szenarien wurden jeweils einzeln getestet, da die Grenzen der jeweiligen Störeinflüsse von Interesse sind. Durch eine kombinierte Betrachtung wird keine Aussage über die Systemgrenzen gewonnen. Allerdings wurden einige Szenarien für verschiedene Teilnehmerzahlen bzw. genutzte Sequenzen betrachtet.

	AWGN	CIR	τ	Δf
Theorie / Simulation	✓	(✓)		
CIR	✓	✓		
τ	✓		✓	
Δf	✓			✓
dyn. Schwellwertanpassung	✓			

Tabelle 3.2: Simulationsszenarien

4 Implementierung

In diesem Kapitel wird die Implementierung der in Kapitel 3 entwickelten Konzepte vorgestellt. Zunächst wird der Aufbau der physikalischen Schicht beschrieben. Anschließend wird der Aufbau der Simulation geschildert. Zum Schluss wird die Validierung der Simulation erläutert.

4.1 Aufbau des PHY

Die Implementierung der physikalischen Schicht erfolgt nach dem in Abbildung 4.1 gezeigten Blockschaltbild. Es handelt sich hierbei um den Uplink von PSSS-255. Der Downlink wird in der Implementierung nicht betrachtet.

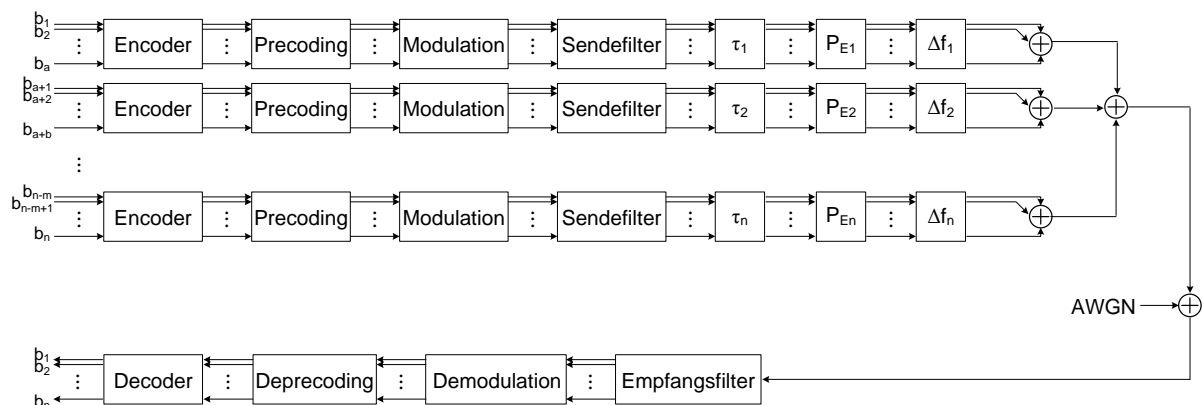


Abbildung 4.1: Aufbau der implementierten physikalischen Schicht

Die nach dem in Abschnitt 3.1 beschriebenen Bit-to-Symbol-Mapping angeordneten Datenbits der verschiedenen Teilnehmer werden im Encoder im Gegensatz zum PSSS-ASK-PHY nach IEEE 802.15.4-2011 nur mit den M-Sequenzen der PSSS-Code-Tabelle multipliziert, die entstandenen Sequenzen aber nur für jeden Teilnehmer einzeln aufsummiert. Ein Precoding kann demnach nur für jeden Teilnehmer gesondert stattfinden und wird in dieser Arbeit vernachlässigt. Die PSSS-Signale durchlaufen nun ein Sendefilter, welches als RRC-Filter mit einem Roll-Off-Faktor von $r = 1$ implementiert wird. Für die Impulsformung wird das Signal zunächst mit dem Faktor $M_{OS} = 10$ überabgetastet. Anschließend erhält jeder Teilnehmer eine individuelle zeitliche Verzögerung, die dessen Abstand zum Empfänger abbildet. Diese kann durch

die Überabtastung in $\text{Chip}/M_{\text{OS}}$ -Schritten erfolgen. Des Weiteren wird die Sendeleistung des ersten Teilnehmers auf einen bestimmten Wert eingestellt. Die Leistung der übrigen Teilnehmer wird durch den CIR bestimmt, welcher vorgegeben wird. Anschließend wird die Summe der Sequenzen berechnet und es wird AWGN-Rauschen aufaddiert. Die Rauschleistung wird entsprechend einer Vorgabe für E_b/N_0 abhängig von der Leistung des ersten Teilnehmers eingestellt, der als Carrier definiert ist. Nach dem Durchlaufen des Empfangsfilters, welches als Matched-Filter implementiert wurde, und des Decoders liegen wieder Datenbits für die einzelnen Teilnehmer vor. Es wird keine Synchronisierung oder Leistungsanpassung implementiert, da dies für die Ermittlung der Performanzgrenzen nicht benötigt wird.

4.2 Aufbau der Simulation

Die in [11] implementierte Simulation des PSSS-ASK-PHY nach IEEE 802.15.4-2011 wird um den PSSS-255-PHY erweitert. Die Implementierung erfolgt mit der Software Matlab von TheMathWorks in der Version R2015a. Mit Hilfe der Simulation werden durch Variation des Rauschabstandes Bitfehlerkurven für die in Kapitel 3.2 beschriebenen Simulationsszenarien berechnet.

Die Simulation setzt sich aus mehreren Bereichen zusammen: der Hauptumgebung, der Simulationsschleife, der darin enthaltenen PHY-Simulation und der Ausgabe und Speicherung der Daten. Die PHY-Simulation wiederum besteht aus mehreren Modulen, von denen sich einige zu- oder abschalten lassen und alle parametrierbar sind, wie zum Beispiel die Impulsformung oder die Variation der Empfangsleistung. Tabelle 4.1 enthält eine Aufzählung der bereits in Abbildung 4.1 gezeigten Module und deren Parametrierungsmöglichkeiten. Des Weiteren sind zwingend zu verwendende Module wie der Encoder mit (\checkmark), an- und abschaltbare Module wie die Modulation mit (\circ) gekennzeichnet.

In der Hauptumgebung werden die Parameter für PSSS-PHY, Module und Simulation festgelegt. Die Wahl des PHY geschieht durch die Auswahl des Frequenzbandes. Hiervon hängt ab, welche Basissequenz im Encoder genutzt wird. Die zyklische Verschiebung für PSSS nach IEEE 802.15.4-2011 beträgt drei Half-Chips bei 868 MHz und sechs Chips bei 915 MHz, bei PSSS-255 hingegen ist sie variabel. Auch die Parameter der Module hängen teilweise von der Wahl des PHY ab. So können für PSSS IEEE 802.15.4-2011 sowohl BPSK als auch M-ASK und M-QAM als Modulationsart gewählt werden, bei PSSS-255 hingegen wird nur eine BPSK verwendet. Außerdem existieren die Module zur Einstellung der Laufzeitverzögerung, der Empfangsleistung und des Frequenzoffsets ausschließlich für PSSS-255. Die Parameter für die Simulationsschleife sind unabhängig vom gewählten PSSS-PHY. So kann z.B. der zu durchlaufende E_b/N_0 -Bereich und dessen Schrittweite variiert werden. Die maximale Anzahl an Simulationsdurchläufen und die minimale Anzahl an Bitfehlern dienen als Abbruchkriterien

und können je nach gewünschter Genauigkeit bzw. Simulationsdauer angegeben werden. Die Simulation wird außerdem nach einem Durchlauf ohne Bitfehler beendet.

Modul	(o) / (✓)	Parametrierungsmöglichkeiten
Encoder	✓	PSSS-Code-Tabelle abh. vom Frequenzband Symbollänge
Precoding & Deprecoding	o	Precoding 1/2 einzeln an-/abschaltbar Deprecoding an-/abschaltbar
Modulation & Demodulation	o	BPSK M-ASK M-QAM
Sende- & Empfangsfilter	o	idealer Tiefpass RRC-Filter(r, M_{OS})
τ	o	in M_{OS} Schritten
CIR	o	P_A P_B über CIR abh. von P_A
Δf	o	Abweichung in ppm
AWGN	o	an/aus
Decoder	✓	PSSS-Code-Tabelle abh. vom Frequenzband dynamische Schwellenanpassung an/aus

Tabelle 4.1: Parametrierung der Module

Die Simulationsschleife besteht aus einer inneren und einer äußeren Schleife. In der äußeren Schleife wird der angegebene E_b/N_0 -Bereich durchlaufen. In der inneren Schleife wird die eigentliche PHY-Simulation aufgerufen. Hier werden die Funktionen der jeweils zugeschalteten Module durchlaufen und das Signal anschließend wieder an die PHY-Simulation zurückgegeben. Diese wird für das jeweilige E_b/N_0 so oft ausgeführt, bis eine der Abbruchbedingungen erfüllt ist.

Aus den in jedem Durchlauf der Simulation berechneten Bitfehlern wird eine Gesamtbitfehlerrate berechnet, welche als BER-Kurve geplottet und zusammen mit anderen wichtigen Simulationsparametern wie dem gewählten Frequenzoffset oder dem CIR gespeichert wird. Der gesamte Aufbau der Simulationsumgebung ist in Abbildung 4.2 gezeigt.

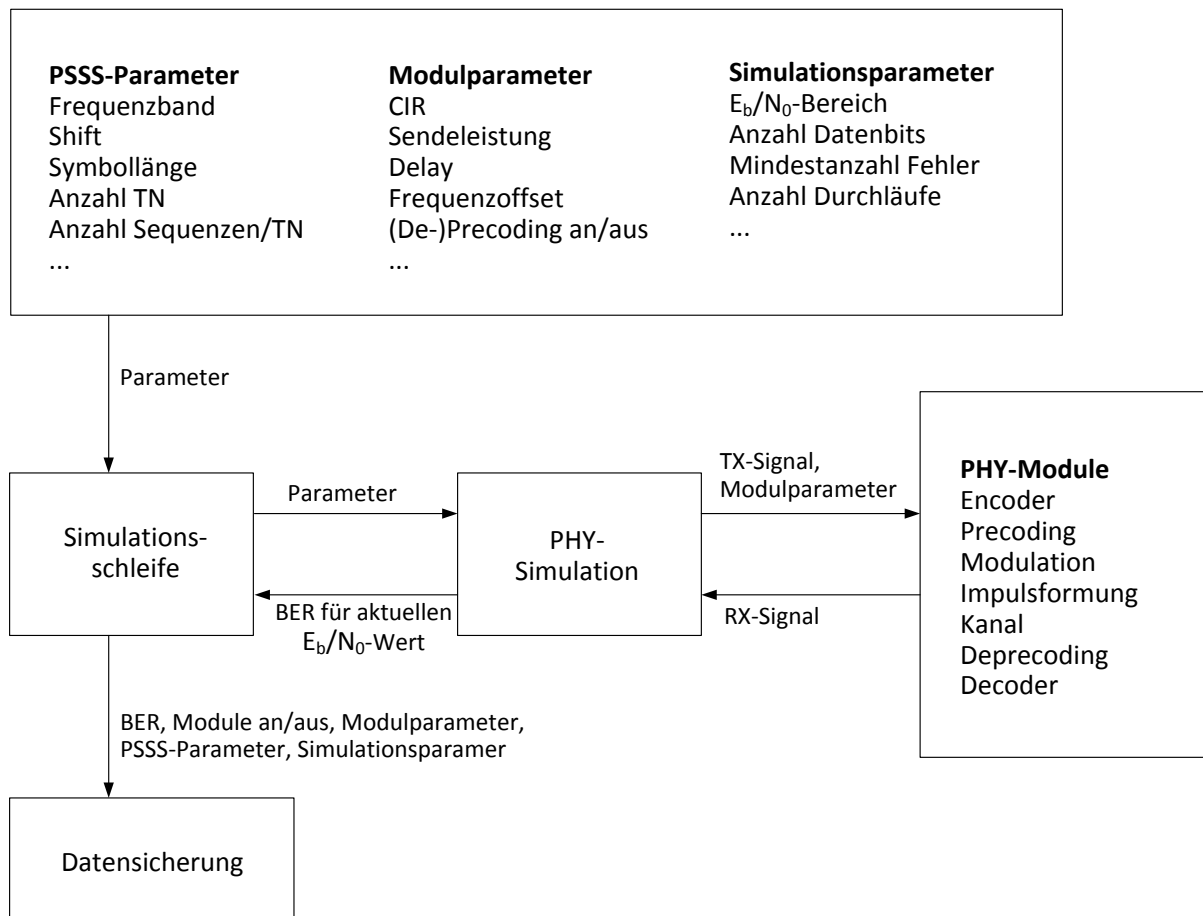


Abbildung 4.2: schematischer Aufbau der Simulationsumgebung

4.3 Validierung

Die für PSSS IEEE 802.15.4-2011 ausgelegten Module der Simulation aus [11] wurden zunächst so erweitert, dass auch PSSS-255 unterstützt wird. Dies betraf insbesondere den Encoder, das Sende- und Empfangsfilter und den AWGN-Kanal. Außerdem musste der Ablauf der PHY-Simulation leicht verändert werden. Im ersten Schritt wurde die Funktion des Encoders und des Kanals auf Richtigkeit überprüft, indem die simulierten mit den theoretisch berechneten und simulierten BER-Kurven aus [11] verglichen wurden.

Anschließend wurden die für PSSS-255 spezifischen Module wie die Zuweisung der Sequenzen an die Teilnehmer, die veränderbare Empfangsleistung, Laufzeitverzögerung und Frequenzoffset implementiert und einzeln validiert. Die BER-Kurven bei veränderter Empfangsleistung bei Nutzung der vorhandenen Sequenzen von zwei Teilnehmern wurde mit den Ergebnissen der Formel aus Abschnitt 3.2 verglichen. Die Bitfehlerraten für einen Phasenoffset wurden mit der theoretischen Berechnung der BER-Kurve bei entsprechender Leistungsverringerung überprüft.

Zuletzt wurde der Decoder durch eine dynamische Schwellwertanpassung erweitert, dessen Ergebnisse anhand theoretischer und simulierter BER-Kurven validiert wurden.

5 Auswertung

In diesem Kapitel werden die in Abschnitt 3.4 vorgestellten Simulationsszenarien ausgewertet. Als Kriterium dient hierbei die BER bei Übertragung über einen AWGN-Kanal. In jedem Simulationsdurchlauf werden die falsch decodierten Bits gezählt und zur Gesamtfehlerzahl addiert. Diese wird am Schluss durch die Gesamtzahl der übertragenen Bits geteilt, um die BER zu erhalten. Zunächst erfolgt die Validierung des PSSS-255 Verfahrens, indem die Simulation mit idealer Impulsformung und ohne Störeinflüsse durchgeführt wird und die Ergebnisse mit der in Abschnitt 3.2 hergeleiteten theoretischen Bitfehlerkurve für $CIR = 0$ dB verglichen werden. Außerdem wird die Abhängigkeit der BER von der Länge der für PSSS genutzten M-Sequenz und der Symbollänge untersucht. Anschließend wird der Einfluss einer Laufzeitverzögerung $\Delta\tau$, eines Unterschieds in der Empfangsleistung ($CIR \neq 0$ dB) und eines Frequenzoffsets Δf untersucht. Am Schluss wird die Beeinflussung der BER durch eine Erweiterung des Decoders durch eine dynamische Anpassung der Entscheidungsschwellen betrachtet.

5.1 AWGN-Kanal

Die Simulation wird zunächst zu Validierungszwecken als ideales PSSS-255-Verfahren durchgeführt, d.h. unter der Annahme, dass es keine Leistungs-, Laufzeit, oder Frequenzunterschiede gibt und alle Sequenzen über denselben AWGN-Kanal gesendet werden. Außerdem wird zunächst eine Impulsformung mit einer idealen Rechteckfunktion vorgenommen. Die simulierten Bitfehlerkurven werden mit den in Abschnitt 3.2 berechneten Kurven verglichen.

Zunächst werden die theoretischen Kurven genauer betrachtet. In Abbildung 5.1 sind die theoretischen BER-Kurven von PSSS mit M-Sequenzen unterschiedlicher Länge M und verschiedenen Symbolängen S abgebildet. Es fällt auf, dass die BER mit steigender Sequenzlänge besser wird. Außerdem verringert sich die Bitfehlerrate mit sinkender Anzahl genutzter M-Sequenzen.

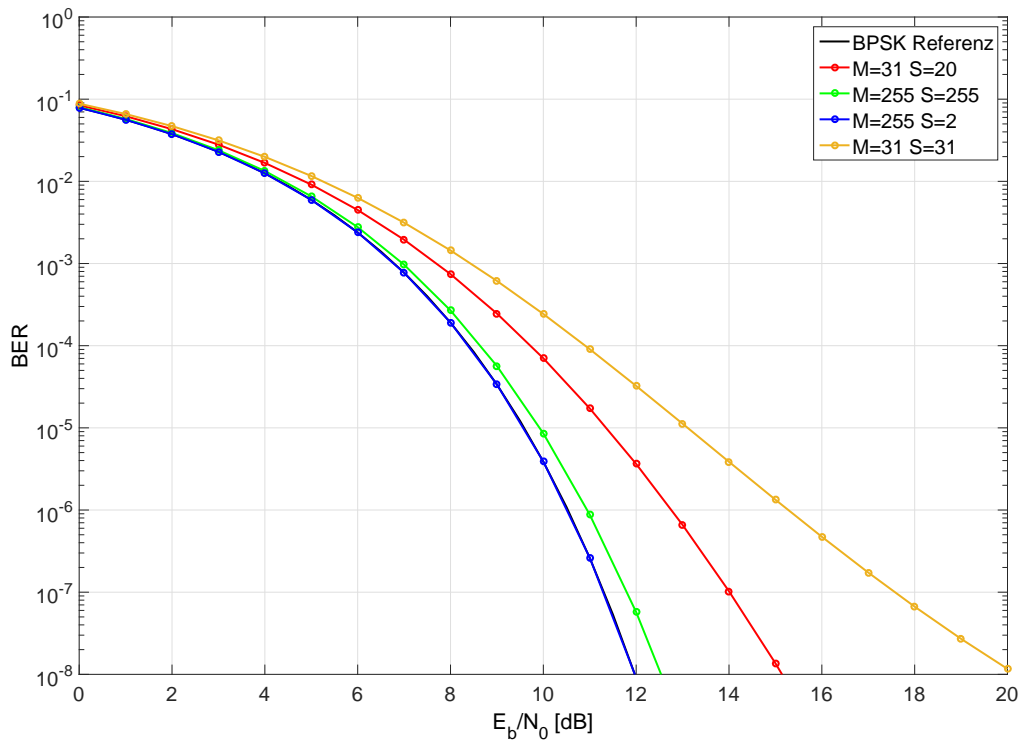


Abbildung 5.1: BER-Kurven von PSSS mit M-Sequenzen unterschiedlicher Länge M und variierenden Symbolängen S

Um die Verbesserung der Bitfehlerrate mit der Sequenzlänge zu erklären, werden die bei PSSS genutzten M-Sequenzen genauer betrachtet. Für die bei PSSS IEEE 802.15.4-2011 genutzte Basissequenz der Länge $n_R = 31$ Chips wird zunächst angenommen, dass bei einem Shift von einem Chip alle zyklisch verschobenen Sequenzen genutzt werden. Somit kann das zu decodierende Bit bis zu 30 konkurrierende Bits mit gleichem Wert haben. Bei einer Basissequenz der Länge $n_R = 255$ Chips können entsprechend bis zu 254 Bits des gleichen Werts wie das zu decodierende Bit auftreten.

Dadurch wird, wie in Abschnitt 2.2.3 gezeigt, die Korrelationsamplitude C_K kleiner und somit das Signal-Rauschverhältnis schlechter. Die Binomialverteilung der Konkurrenzsequenzen für eine M-Sequenz mit 31 Chips ist in Abbildung 5.2 dargestellt. Abbildung 5.3 zeigt die Verteilung für die 255 Chips lange M-Sequenz. Es ist zu erkennen, dass die Kurve der längeren Sequenz deutlich schmaler verläuft. Die Wahrscheinlichkeiten für eine große Anzahl an Konkurrenzsequenzen sind sehr klein, sodass die Fälle, in denen die C_K besonders klein ist, nur sehr selten vorkommen. Im Vergleich dazu deckt die Kurve für eine kürzere Sequenz einen deutlich größeren Teil der möglichen Verteilungen ab. Aus diesem Grund erzeugt die PSSS-Codierung einer längeren M-Sequenz eine niedrigere Bitfehlerrate.

Des Weiteren ist zu erkennen, dass bei einer Nutzung von nur 20 Sequenzen wie bei PSSS IEEE 802.15.4-2011 bzw. zwei Sequenzen, wie für PSSS-255 zu Testzwecken verwendet, die BER geringer ist als bei Nutzung aller Sequenzen bei gleicher Sequenzlänge. Dies liegt durch die vorherige Betrachtung auf der Hand, da bei einer geringeren Anzahl an Konkurrenzsequenzen die Korrelationsamplitude weniger verringert werden kann.

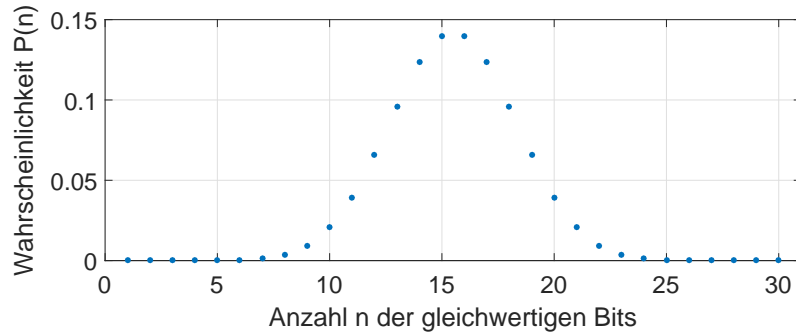


Abbildung 5.2: Wahrscheinlichkeitsverteilung der Anzahl der konkurrierenden Bits für eine Codelänge von 31 Chips

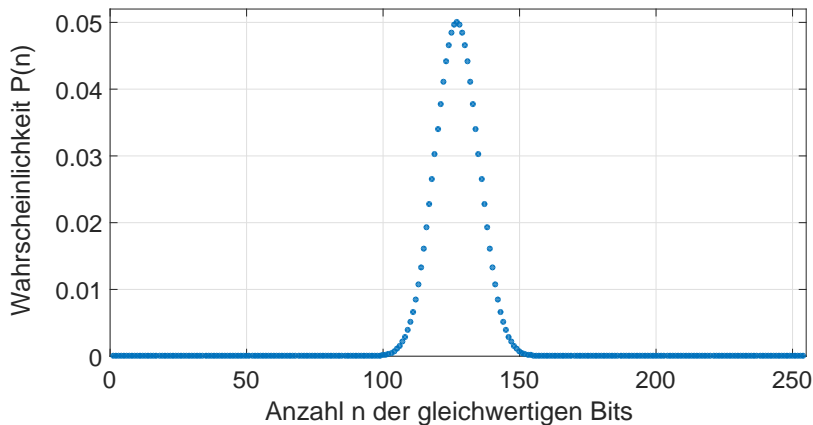


Abbildung 5.3: Wahrscheinlichkeitsverteilung der Anzahl der konkurrierenden Bits für eine Codelänge von 255 Chips

Mit diesem Wissen kann nun die Theorie mit den Simulationsergebnissen verglichen werden. In Abbildung 5.4 sind die theoretische und in der Simulation berechnete BER-Kurve für PSSS-255 mit 255 verwendeten Sequenzen dargestellt. Zur besseren Einordnung ist zusätzlich die BER-Kurve einer bipolaren BPSK dargestellt. Es ist zu erkennen, dass die simulierte Kurve auf der theoretischen PSSS-Kurve liegt. Somit ist die Korrektheit der Simulation validiert. Des Weiteren ist zu sehen, dass die Bitfehlerrate durch die PSSS-Codierung etwas schlechter ist als bei reiner Frequenzspreizung (BPSK).

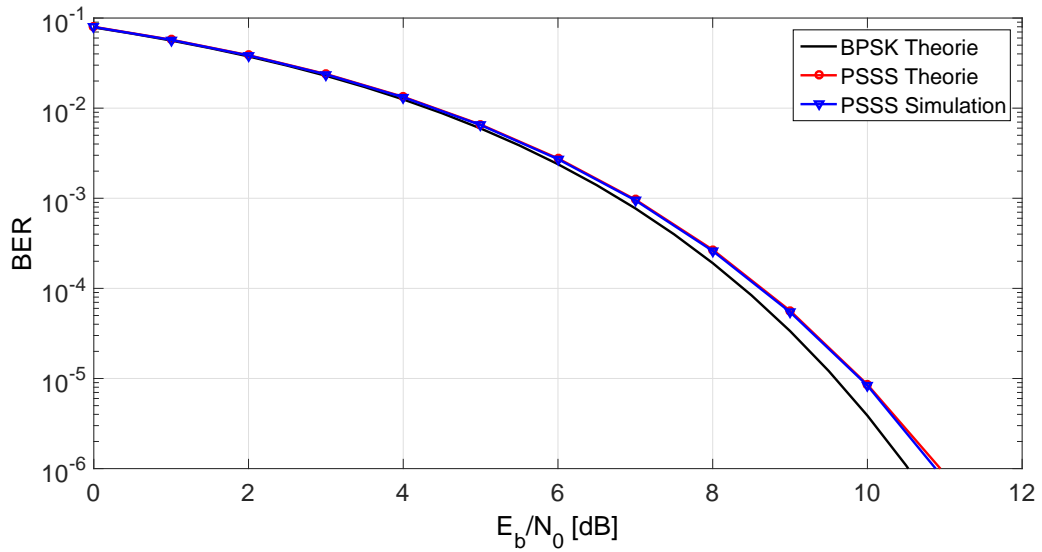


Abbildung 5.4: BER-Kurve von PSSS-255 mit einer Symbollänge von 255 Bits

In Abbildung 5.5 ist die Anordnung der Bitfehlerkurven für nur zwei verwendete Sequenzen dargestellt. Auch hier stimmt die theoretische mit der simulierten Kurve überein. Außerdem ist zu erkennen, dass die PSSS-Kurve nahezu auf der BPSK-Kurve liegt. Dies ist offensichtlich, da sich die PSSS-Kurve mit sinkender Anzahl genutzter Sequenzen immer weiter der BPSK-Kurve annähert und eine PSSS mit nur einer Sequenz einer Übertragung mit BPSK gleicht.

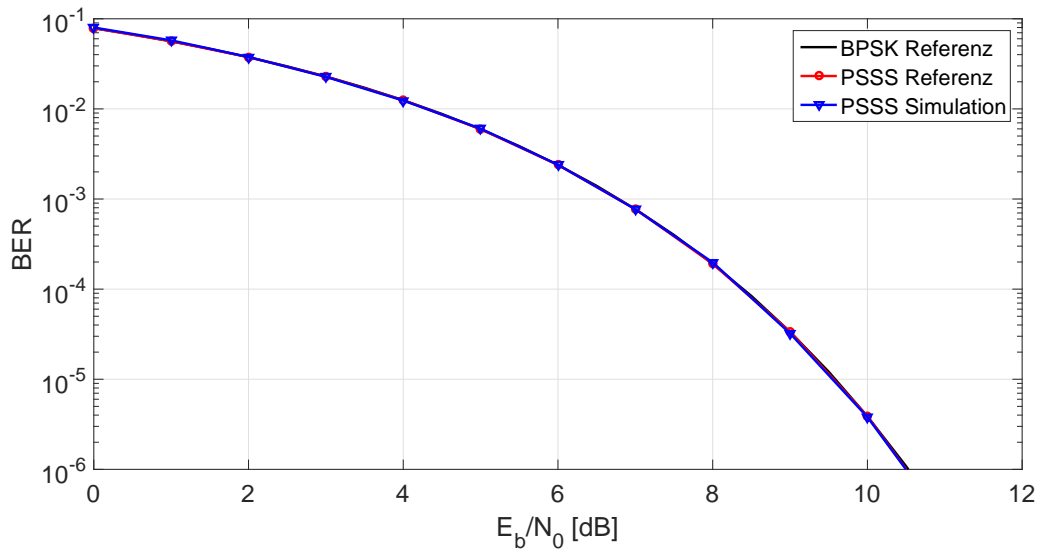


Abbildung 5.5: BER-Kurve von PSSS-255 mit einer Symbollänge von 2 Bits

5.2 Laufzeitverzögerung

PSSS-255 wird von mehreren gleichzeitig sendenden Teilnehmern genutzt. Durch unterschiedliche Entfernungen der Sender zum Empfänger kommt es zu einer zeitlichen Verschiebung zwischen den empfangenen PSSS-Sequenzen. Da die zyklische Verschiebung von einem Chip zwischen den genutzten M-Sequenzen ebenfalls einer zeitlichen Verzögerung entspricht, kommt es bei ausreichender Laufzeitverzögerung zu Fehlern bei der Decodierung von benachbarten Sequenzen zweier Teilnehmer. Eine Verzögerung von einem Chip führt somit zu Fehlern bei der Decodierung beider Sequenzen, da diese sich exakt überlagern.

Es wird nun angenommen, zwei Teilnehmer A und B nutzen jeweils eine PSSS-Sequenz. Diese sind um einen Shift von einem Chip zyklisch verschoben. Abbildung 5.6 zeigt die BER-Kurven der Sequenzen A und B in Abhängigkeit der Laufzeitverzögerung τ von Sequenz A, welche in Zehntel-Chip Schritten bis zu einem Chip ansteigt. Mit größer werdender Laufzeitverzögerung steigt die BER-Kurve von Teilnehmer A bis auf 0,5. Bei einer maximalen Verschiebung von $\tau = 1$ Chip werden nur noch Nullen empfangen, da nun effektiv keine Daten mehr mit Sequenz A codiert sind. Somit wird ein zufälliges Ergebnis erzielt, wobei die Wahrscheinlichkeit 0,5 für die Decodierung einer 1 bzw. -1 beträgt. Die Bitfehlerrate von Sequenz B, mit der nun zwei Bits gleichzeitig codiert werden, steigt bis auf 0,25. Hier müssen zwei Fälle betrachtet werden: Haben die Bits von Teilnehmer A und B den gleichen Wert, so wird korrekt decodiert. Die Wahrscheinlichkeit dafür ist 0,5. Haben die Bits entgegengesetzte Werte, so addieren sich beide Sequenzen zu Null und das Decodierergebnis ist, analog zu oben, zufällig. Die Gesamtfehlerwahrscheinlichkeit für Sequenz B für eine Laufzeitverzögerung von einem Chip beträgt somit $BER = 0,5 \cdot 0,5 = 0,25$.

Als nächstes werden drei benachbarte Sequenzen A, B und C betrachtet. Die beiden außen liegenden Sequenzen A und C werden um den gleichen Betrag in Richtung der mittleren Sequenz B verschoben. Dies kann der Fall sein, wenn die Sequenz A eine positive, Sequenz C eine negative zeitliche Verschiebung hat. In Abbildung 5.7 ist zu sehen, dass die BER-Kurve von Sequenz B zunächst steiler ansteigt als in Abbildung 5.6 und am Ende flach verläuft. Die zwei störenden Bits werden hier in der Korrelationsamplitude aufaddiert, sodass C_K im ungünstigen Fall mit der doppelten Amplitude wie im obigen Beispiel verringert wird und somit schon bei geringeren Laufzeitverschiebungen zu Null wird. Das Maximum der BER von Teilnehmer B beträgt auch hier 0,25, da auch bei drei überlagerten Bits in einem Viertel der Fälle falsch decodiert wird.

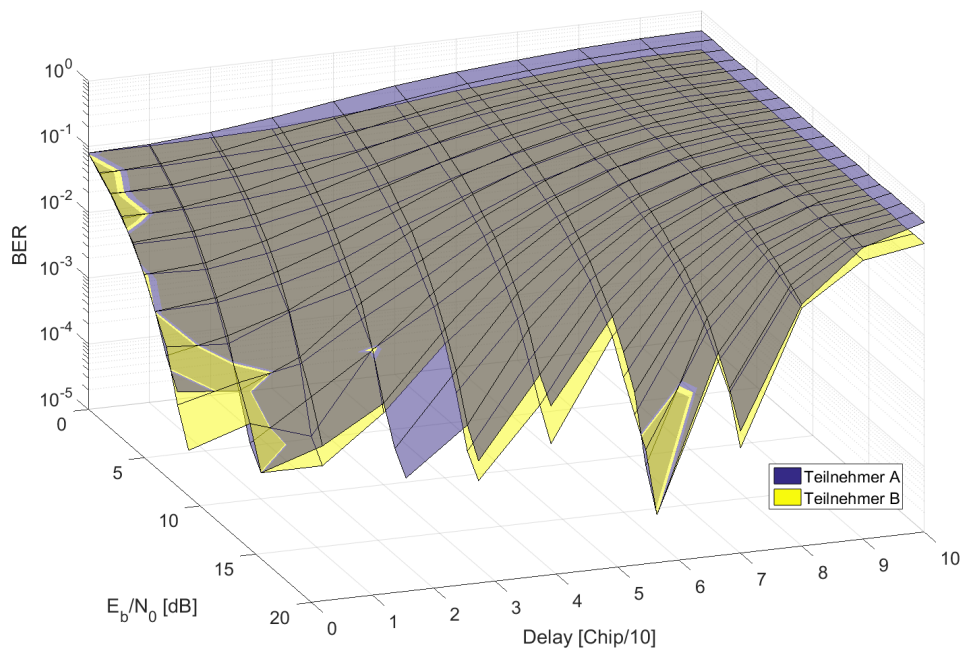


Abbildung 5.6: BER-Kurven von Teilnehmer A und B abhängig von der Laufzeitverzögerung, $n_A = 1$, $n_B = 1$

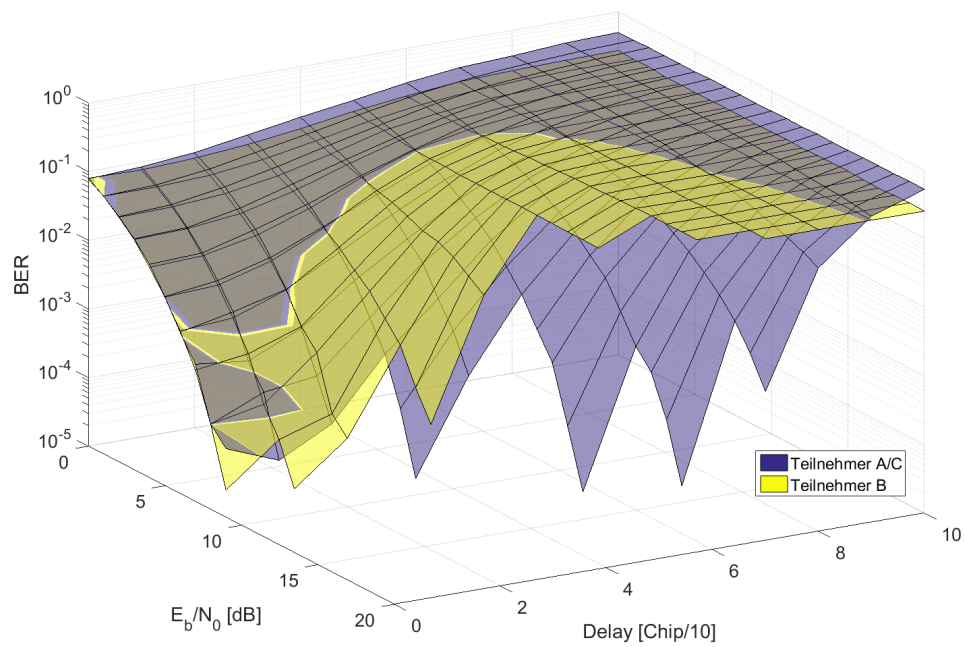


Abbildung 5.7: BER Kurven von Teilnehmer A, B und C abhängig von der Laufzeitverzögerung, $n_A = 1$, $n_B = 1$, $n_C = 1$

Mit Hilfe der Laufzeitverzögerung kann auch das Verhalten bei einer versehentlichen Doppelbelegung von Sequenzen simuliert werden. Hierzu werden die ersten 20 Sequenzen um je 20 Chips verschoben, sodass sie sich mit den Sequenzen 21–40 überlagern. In Abbildung 5.8 ist die Gesamtfehlerzahl aller Sequenzen nach der Simulation dargestellt. Abbildung 5.9 zeigt die gemittelten Bitfehlerkurven von Sequenz 1–20, 21–40 und 41–255. Es ist zu erkennen, dass die übrigen Sequenzen nicht durch den Delay beeinflusst werden, da die Fehlerzahl entsprechend der Bitfehlerrate aus Kapitel 5.1 unverändert bleibt.

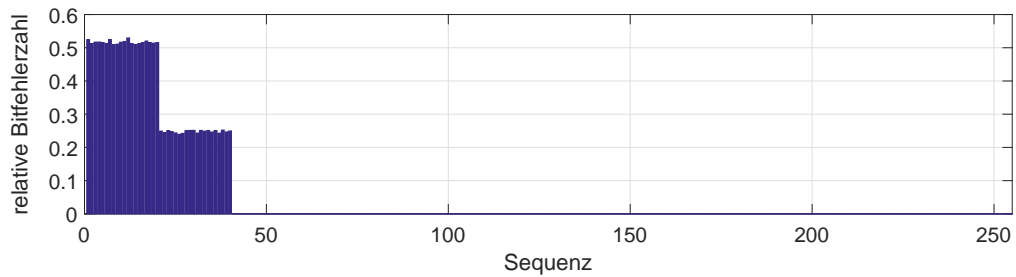


Abbildung 5.8: Fehlerzahl aller 255 Sequenzen nach Simulation, wenn die ersten 20 Sequenzen um 20 Chips verschoben sind

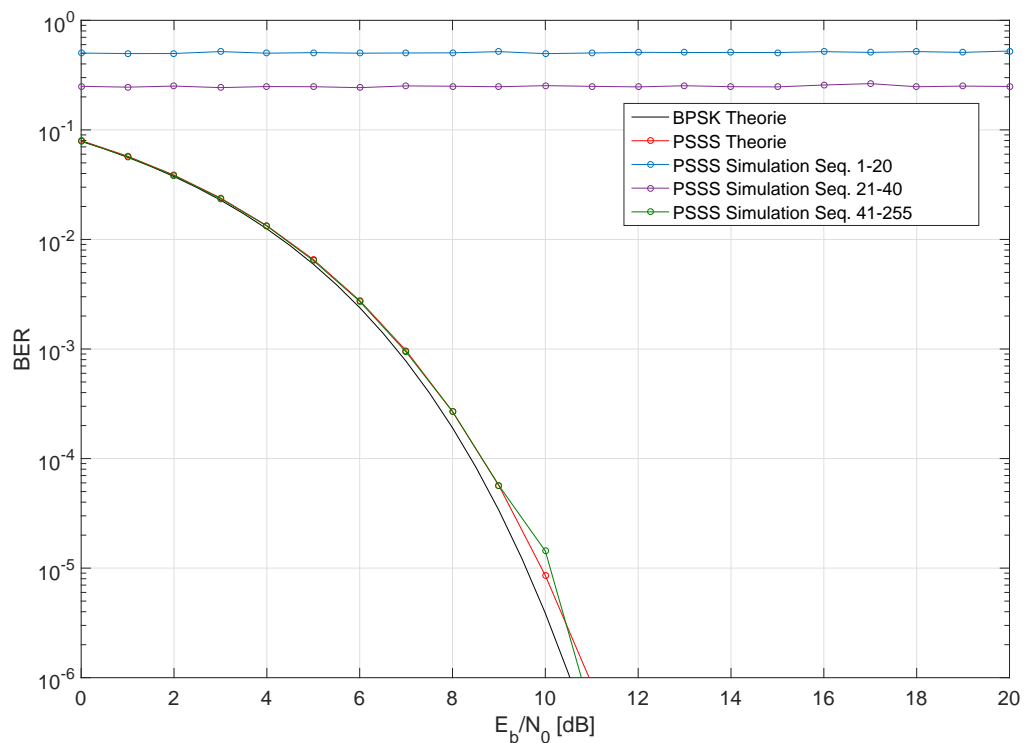


Abbildung 5.9: BER-Kurven aller 255 Sequenzen, wenn die ersten 20 um 20 Chips verschoben sind

Aus dieser Erkenntnis wird klar, dass bei einer aufgrund von Time Advance-Verfahren maximalen Laufzeitverschiebung von einem Chip jeweils nur die erste Sequenz des benachbarten Teilnehmers beeinflusst wird. Somit können nicht benutzte Sequenzen zwischen den verschiedenen Teilnehmern platziert werden, um den Einfluss einer Laufzeitverschiebung auf die BER zu minimieren.

5.3 Empfangsleistung

Aufgrund von verschiedenen Sender-Empfänger-Distanzen bzw. Sendeleistungen der Teilnehmer kann es bei PSSS-255 zu Unterschieden in der Empfangsleistung kommen. Wird eins der gesendeten Bits als das gewünschte, zu empfangene Signal (Carrier) und die anderen als Störer (Interferer) betrachtet, so lässt sich das Verhältnis aus deren Sendeleistungen als CIR angeben. Dessen Berechnungsvorschrift ist in Formel 3.5 angegeben. Die theoretische Herleitung und Berechnung der Bitfehlerraten für verschiedene CIR erfolgt in Abschnitt 3.2.

Im Folgenden werden zwei Szenarien verglichen:

Szenario 1

Zwei Teilnehmer A und B nutzen jeweils eine PSSS-Sequenz

$$P_C = P_A$$

$$P_I = P_B$$

Szenario 2

Teilnehmer A nutzt eine, Teilnehmer B 254 Sequenzen

$$P_C = P_A$$

$$P_I = 254 \cdot P_B$$

Die Leistung von Teilnehmer A wird auf $P_A = 1$ festgelegt. In beiden Szenarien wird ein CIR-Bereich zwischen 0 und -60 dB durchlaufen. Abbildung 5.10 zeigt die BER-Kurve von Teilnehmer A für Szenario 1. Hier bricht die BER bei einem CIR von -48 dB steil ein bis zu einer Fehlerwahrscheinlichkeit von 0,5. In Abbildung 5.11 ist die BER von Teilnehmer A für Szenario 2 dargestellt. Hier verläuft die Verschlechterung der BER mit sinkendem CIR wesentlich flacher. Dies lässt sich anhand der in Abschnitt 2.2.3 hergeleiteten Korrelationsamplitude C_K erklären: Um ein Abknicken der BER-Kurve bis zum vollständigen Versagen von PSSS bei $BER = 0,5$ zu erreichen, muss die Korrelationsamplitude $C_K = 0$ werden, in die Teilnehmer A mit $n_0 = n_R \cdot d_0$ eingeht. Also muss $n_s = -n_0$ betragen.

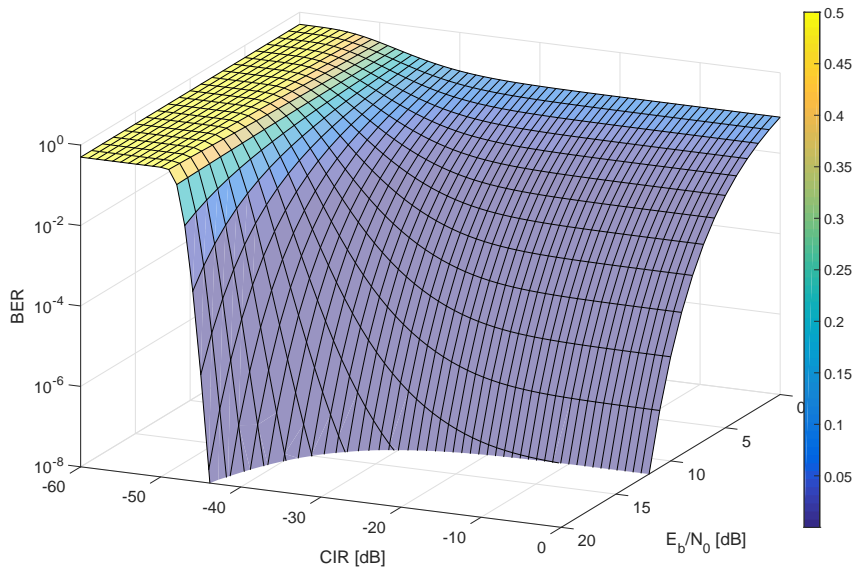


Abbildung 5.10: Theoretische BER-Kurve von Teilnehmer A für Szenario 1

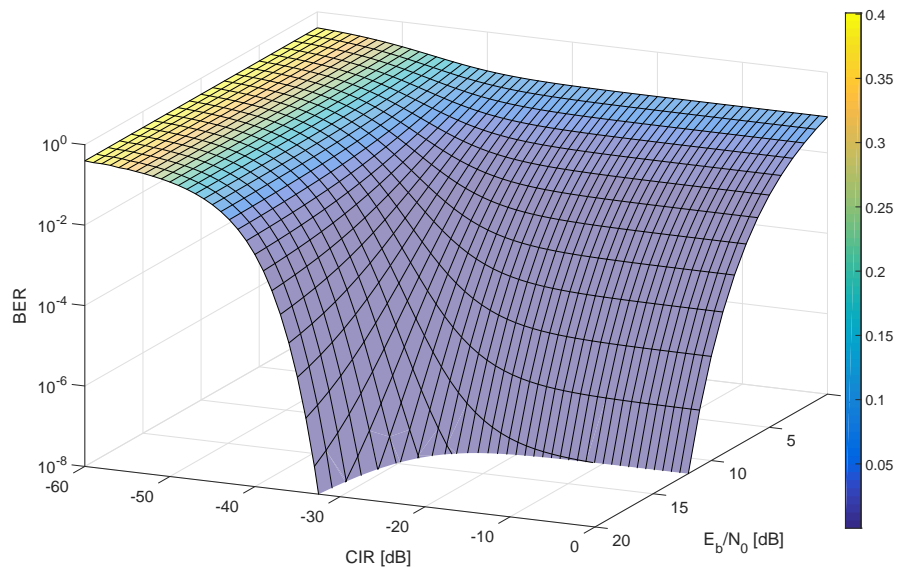


Abbildung 5.11: Theoretische BER-Kurve von Teilnehmer A für Szenario 2

In Szenario 1 wirkt sich P_I entweder positiv oder negativ mit einer Wahrscheinlichkeit von jeweils 0,5 auf C_K aus. Beträgt $P_I = (n_R)^2 = 255^2 = 48$ dBm oder mehr, so führt dies zu einer fehlerhaften Detektion von d_0 , da $C_K \leq 0$ wird. Für schlechtere E_b/N_0 wird der Knick in der BER Kurve immer flacher, da additives Rauschen bereits eher zu einer fehlerhaften Decodierung führen kann.

In Szenario 2 ist die Anzahl der Störer mit gleichem Wert binomial verteilt. Für $P_A = P_B = 1$ beträgt n_s bereits $254 \cdot 1$, sofern alle Bits von Teilnehmer B den gleichen Wert haben. C_K ist dann gerade noch > 0 . Dieser Fall ist aufgrund der Binomialverteilung allerdings sehr unwahrscheinlich. Der zugehörige CIR für diesen Fall beträgt $1/(1^2 \cdot 254) = -24$ dB. Die Wahrscheinlichkeit für weniger Bits mit gleichem Wert ist höher, allerdings muss die Störampplitude entsprechend größer werden, um $C_K = 0$ zu erreichen. Die Verschlechterung der BER-Kurve verläuft also im rauschfreien Fall, entsprechend der in Abbildung 5.12 gezeigten Wahrscheinlichkeitsverteilung für die Anzahl Bits gleichen Werts, bis hin zu einem Wert von 0,5. Auch hier wird der Verlauf mit schlechterem E_b/N_0 flacher. Für Szenario 1 springt die Fehlerwahrscheinlichkeit bei einem CIR von -48 dB von 0 auf 0,5.

Die Betrachtung macht deutlich, dass sich PSSS-255 bei Nutzung aller vorhandenen Sequenzen mit der gleichen Leistung bereits an der Grenze zur einer sich verschlechternden Bitfehlerrate befindet. Sobald Rauschen hinzukommt, kommt es zu fehlerhafter Decodierung einzelner Bits. Somit ist eine gute Leistungsregelung von Nöten, um die Bitfehlerzahl möglichst gering zu halten.

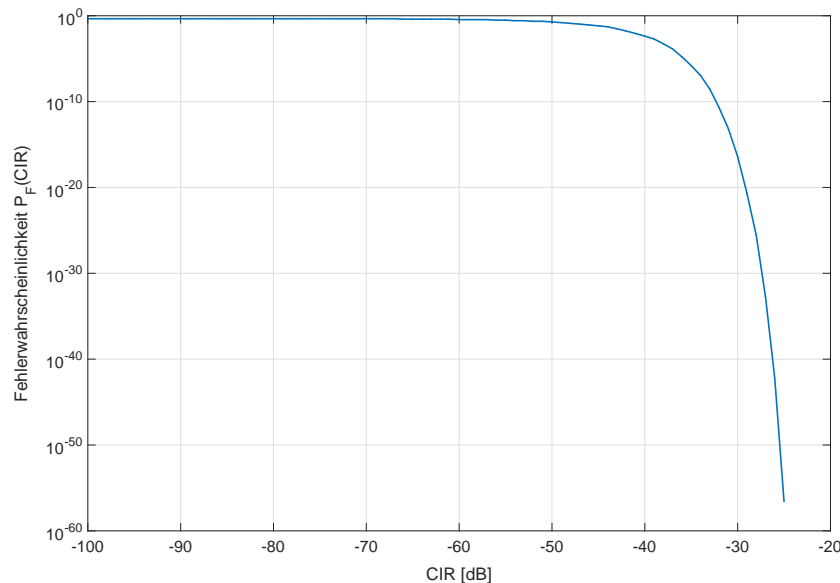


Abbildung 5.12: Fehlerwahrscheinlichkeit in Abhängigkeit vom CIR für Szenario 2

In den Abbildungen 5.13 und 5.14 sind die simulierten BER-Kurven von Teilnehmer A und B für Szenario 1 und 2 dargestellt. Die Übereinstimmung mit den theoretischen Kurven von Teilnehmer A ist anhand der Werte in Tabelle 5.1 für Szenario 2 und in Tabelle 5.2 zu sehen.

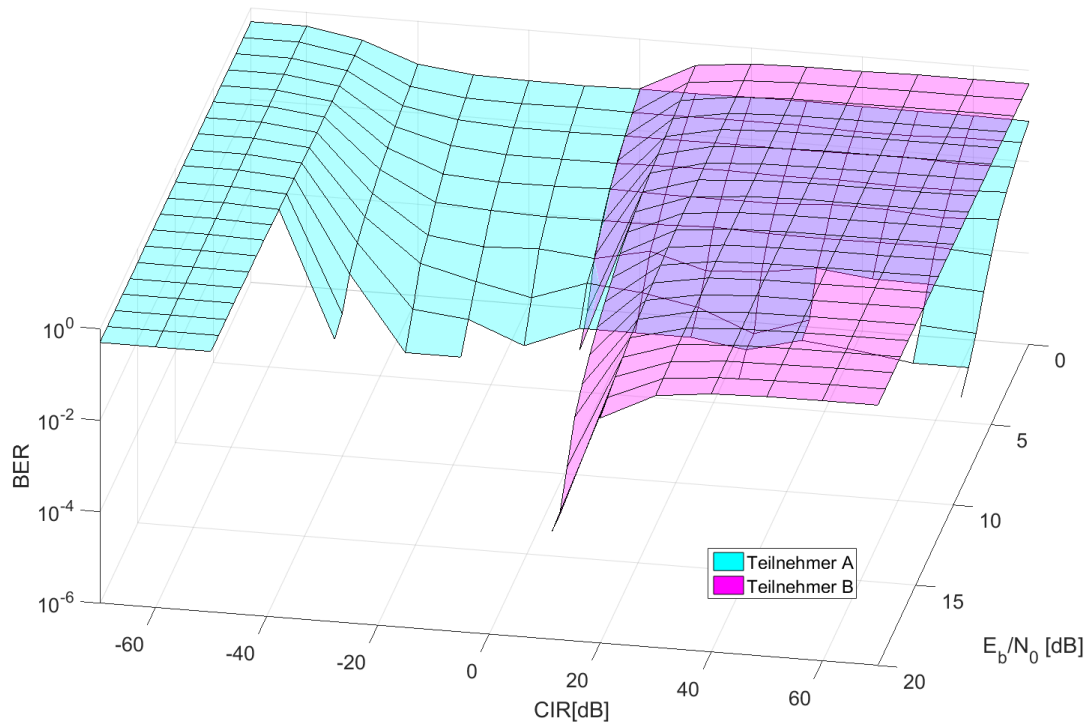


Abbildung 5.13: Simulierte BER-Kurve von Teilnehmer A und B für Szenario 1

CIR [dB]	E_b/N_0 [dB]	Simulation	Theorie
-30	0	0,08	0,08
-50	0	0,36	0,32
-60	0	0,49	0,5
-40	10	0,0013	0,0016
-50	10	0,43	0,34
-60	10	0,5	0,5

Tabelle 5.1: Vergleich der theoretischen und simulierten BER-Kurven von Szenario 1

Die Asymmetrie zwischen den BER-Kurven von Teilnehmer A und B ist der Normierung des E_b/N_0 auf Teilnehmer A zuzuschreiben: Mit kleiner werdendem CIR verschiebt sich die BER-Kurve von Teilnehmer A nach rechts, da er selbst mit sehr kleiner, der Störer aber mit großer Leistung sendet. Das Signal-Rausch-Verhältnis ist hierbei für Teilnehmer A über alle CIR-Werte konstant. Im Gegensatz dazu verschlechtert sich das Signal-Rausch-Verhältnis mit steigendem CIR für Teilnehmer B, sodass sich dessen BER-Kurve stärker nach rechts verschiebt, da sich die Effekte von CIR und E_b/N_0 überlagern.

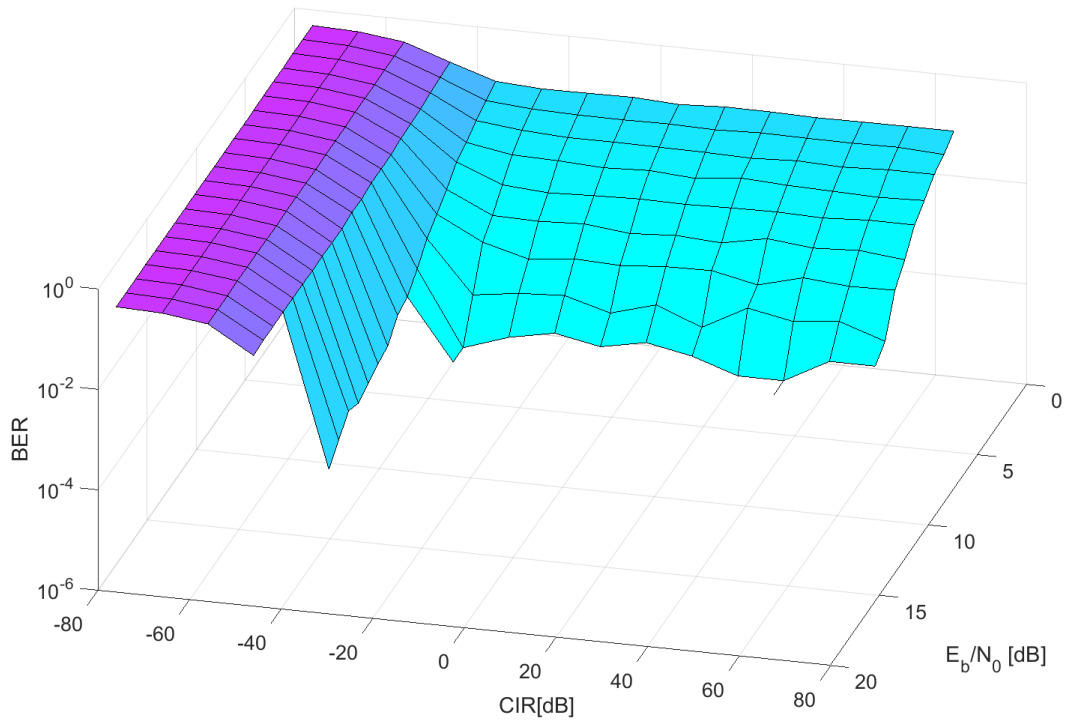


Abbildung 5.14: Simulierte BER-Kurve von Teilnehmer A und B für Szenario 2

CIR [dB]	E_b/N_0 [dB]	Simulation	Theorie
-26	0	0,078	0,079
-46	0	0,172	0,171
-56	0	0,348	0,349
-36	10	0,0012	0,0013
-46	10	0,11	0,109
-56	10	0,338	0,343

Tabelle 5.2: Vergleich der theoretischen und simulierten BER-Kurven von Szenario 1

Abbildung 5.15 zeigt die simulierte und die in Abschnitt 3.2 theoretisch berechnete Bitfehlerkurve von Teilnehmer A für Szenario 1 und einen CIR von -50 dB. Es fällt auf, dass die Bitfehlerrate mit wachsendem E_b/N_0 steigt. Dies ist ein Spezialfall, der nur auftritt, wenn beide Teilnehmer nur eine Sequenz nutzen, da so die gesamte Leistung des Störers die Korrelationsamplitude C_K entweder erhöht oder verringert. Bei einem schlechten CIR wird das Bit durch den Störer verfälscht, kann aber bei einem geringen E_b/N_0 durch das Rauschen wieder korrigiert werden. Aufgrund der allerdings sehr geringen Auswirkungen ist dieser Effekt in

Abbildung 5.13 nicht zu erkennen. Werden viele Sequenzen genutzt, geht die Wahrscheinlichkeit für den obigen Fall aufgrund der Binomialverteilung der Störbits gegen Null.

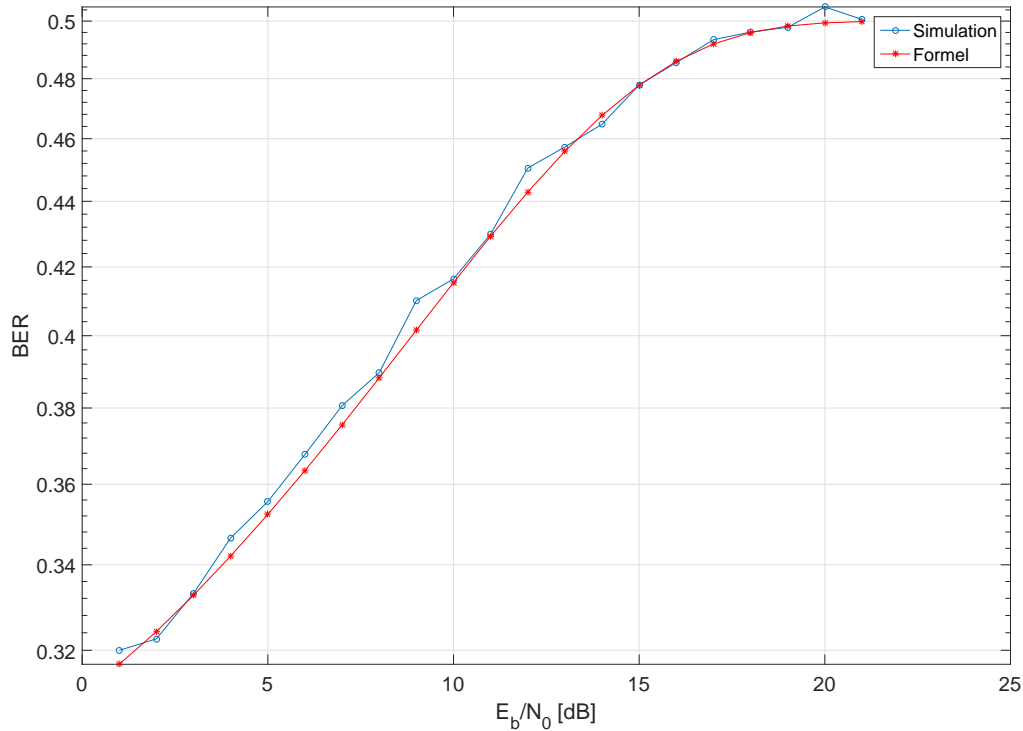


Abbildung 5.15: BER-Kurve von Teilnehmer A, simuliert (blau) und berechnet (rot), $n_A = 1$, $n_B = 1$

5.4 Frequenz- und Phasenoffset

Übliche Schwingquarze haben eine Frequenztoleranz von 10 ppm. Die Teilnehmer des PSSS-Systems senden also nicht exakt mit der gleichen Frequenz. Außerdem kann es zu einem Phasenoffset $\Delta\varphi$ zwischen Teilnehmern kommen.

Zunächst wird der Einfluss eines Phasenoffsets auf die Bit Error Rate betrachtet. Es senden zwei Teilnehmer mit jeweils einer Sequenz. In Abbildung 5.16 sind die BER-Kurven für verschiedene $\Delta\varphi$ von Teilnehmer B dargestellt. Für Teilnehmer A ist immer $\Delta\varphi = 0$. Es ist zu erkennen, dass mit $\Delta\varphi$ die BER von Teilnehmer B bis auf 0,5 für $\Delta\varphi = \pi/2$ steigt. Die BER-Kurve von Teilnehmer A liegt auf der theoretischen Kurve. Durch die Phasendrehung des Signals verringert sich die Leistung, sodass eine Phasendrehung mit $\Delta\varphi$ auch als Multiplikation der Sendeleistung mit $\cos^2(\Delta\varphi)$ dargestellt werden kann. In Abbildung 5.17 sind zusätzlich die BER-Kurven für eine um die entsprechende Leistung veränderte BPSK und

PSSS. Diese wurden theoretisch berechnet. Die PSSS-Kurven liegen exakt aufeinander und die BPSK-Kurven haben denselben minimalen Offset zur PSSS wie für $\Delta\varphi = 0$.

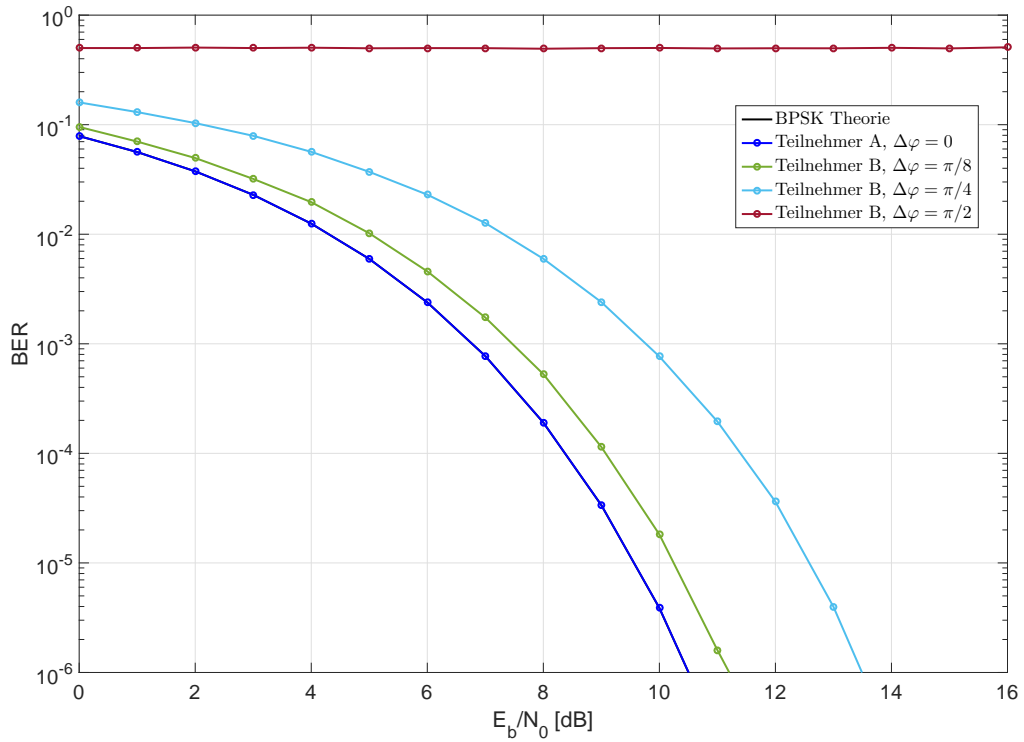


Abbildung 5.16: BER-Kurven von Teilnehmer A und B für verschiedene Phasenoffsets von Teilnehmer B

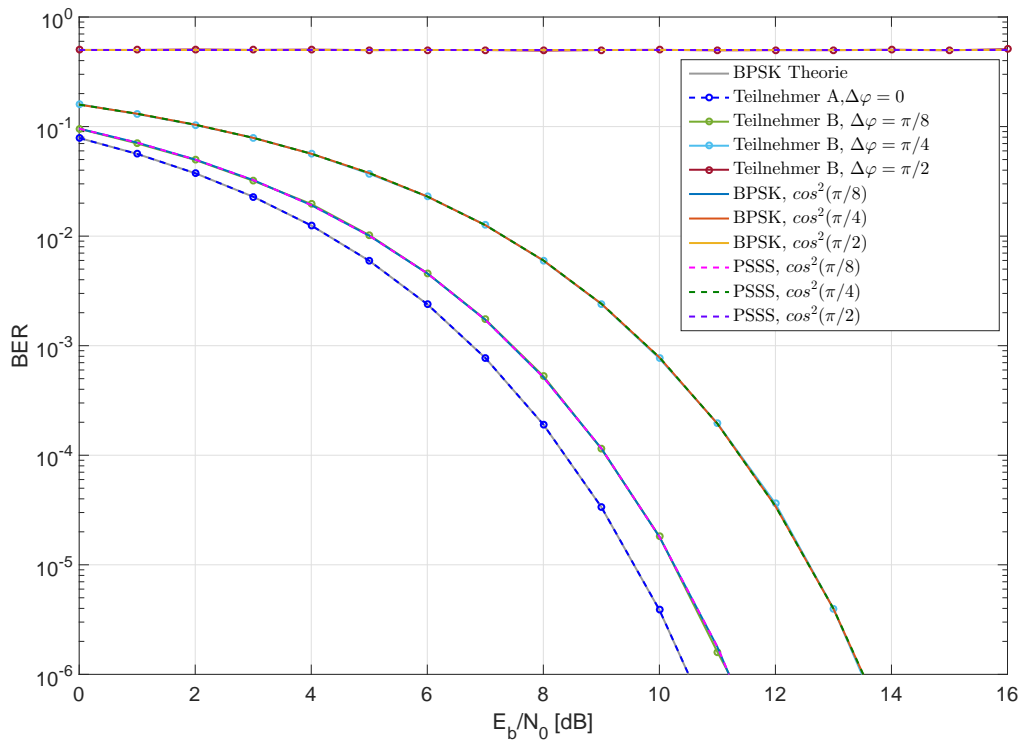


Abbildung 5.17: BER-Kurven für Phasenoffset und Leistungsveränderung im Vergleich

Ist eine Sequenz nicht in der Phase, sondern in der Frequenz verschoben, so steigt der Phasenoffset bis zum Ende der Sequenz immer weiter. Bei einer nominalen Trägerfrequenz von

$$f_0 = 5,8 \text{ GHz} \quad (5.1)$$

ergibt sich mit 10 ppm Frequenztoleranz eine maximale Frequenzvariation von

$$\begin{aligned} \Delta f &= f_0 \cdot 10 \cdot 10^{-6} \\ &= 58 \text{ kHz} \end{aligned} \quad (5.2)$$

Zur Bestimmung der Bitfehlerrate wird die Sequenz eines Teilnehmers in mehreren Simulationsdurchläufen jeweils um einen zufällig gewählten Wert zwischen 0 und Δf in der Frequenz verschoben. Für Δf lässt sich eine maximale Phasendrehung berechnen: Die Chiprate von $f_{\text{Chip}} = 50 \text{ MChip/s}$ ist vorgegeben. Somit beträgt die Chipdauer $T_{\text{Chip}} = 20 \text{ ns}$. Die Zeit, die für

die Übertragung eines Bits benötigt wird, beträgt somit

$$\begin{aligned}
 t_{\max} &= n_{\text{Chips}} \cdot T_{\text{Chip}}, & (5.3) \\
 &= 285 \cdot 20 \text{ ns} \\
 &= 5,7 \mu\text{s}
 \end{aligned}$$

wobei n_{Chips} die Länge der PSSS-Sequenz ist. Die maximale Phasenverschiebung am Ende der Sequenz beträgt somit

$$\begin{aligned}
 \varphi_{\max} &= e^{(j2\pi \cdot t_{\max} \cdot \Delta f)} & (5.4) \\
 &= e^{(j2\pi \cdot 0,3306)}.
 \end{aligned}$$

Dies entspricht einer Phasenverschiebung von $\frac{2}{3}\pi$. Da bereits bei einem Phasenoffset von $\frac{\pi}{8}$ Fehler entstehen, ist es nicht verwunderlich, dass die in Abbildung 5.18 dargestellte Bitfehlerrate sehr schlecht ist. Werden zwei Symbole hintereinander übertragen, so erhöht sich die Phasenverschiebung mit jedem Chip weiter, bis 2π erreicht sind. Aus diesem Grund ist die BER-Kurve für zwei übertragene Symbole in Abbildung 5.19 noch schlechter als für ein Symbol.

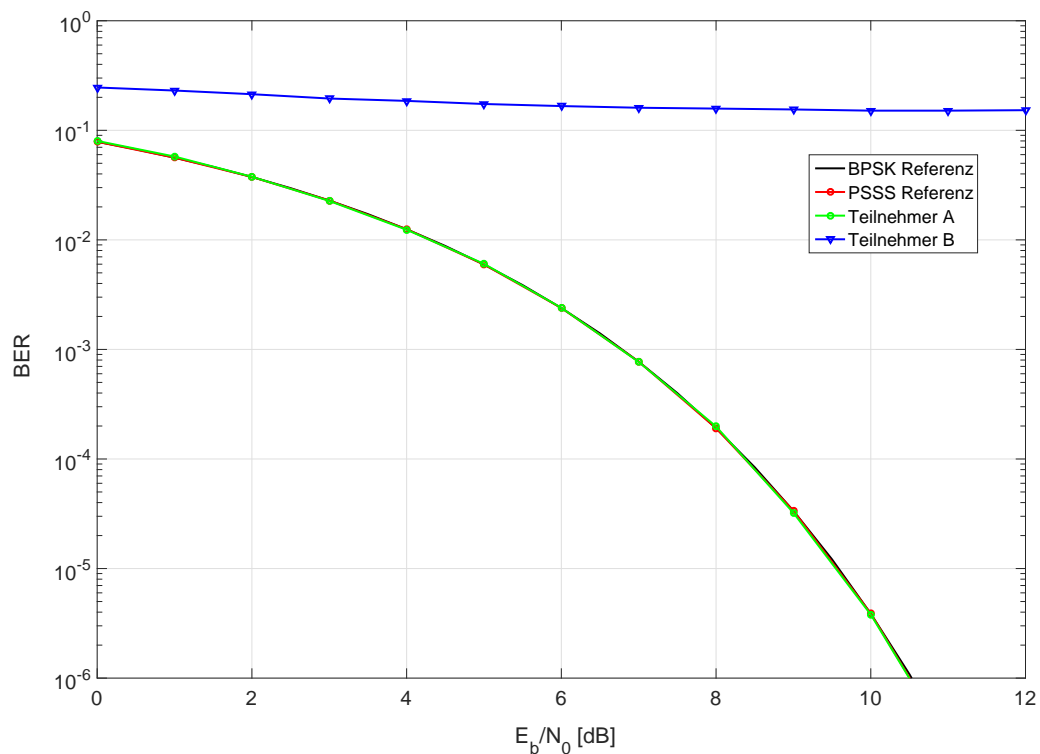


Abbildung 5.18: BER-Kurven von Teilnehmer A und B für einen Frequenzoffset von 10 ppm von Teilnehmer, 1 Symbol übertragen

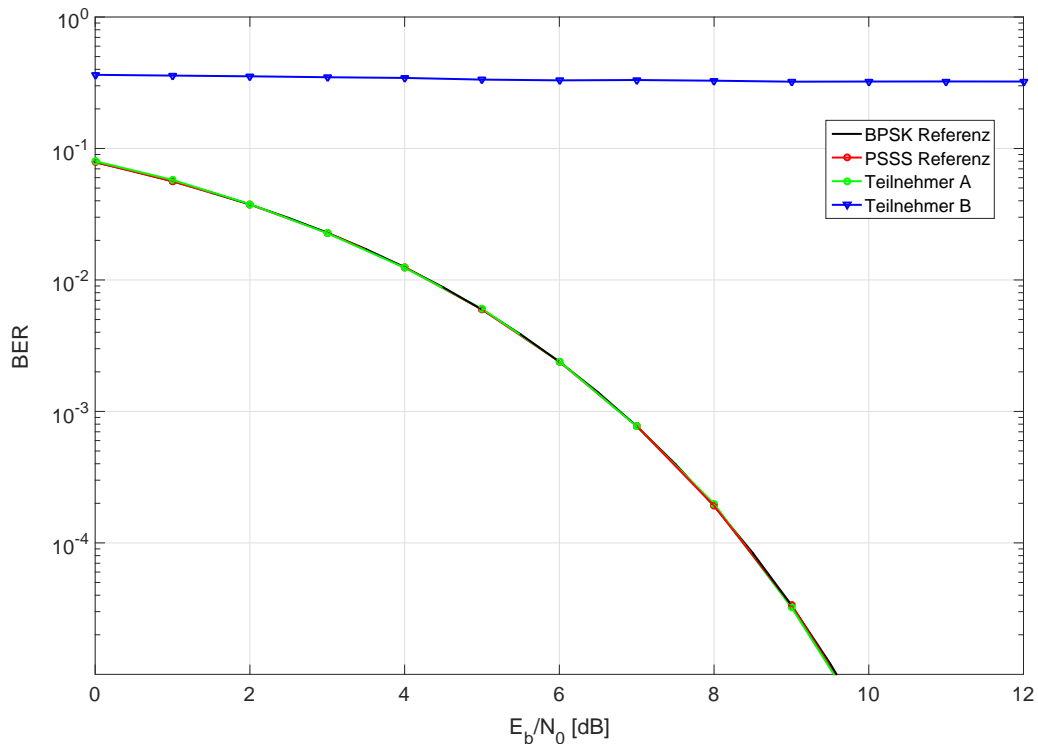


Abbildung 5.19: BER-Kurven von Teilnehmer A und B für einen Frequenzoffset von 10 ppm von Teilnehmer, 2 Symbole übertragen

Bisher wurde nur die Bitfehlerrate der frequenzverschobenen Sequenz betrachtet. Die BER der unverschobenen Sequenz liegt in Abbildung 5.18 auf der theoretischen Kurve, jedoch wurde bisher nicht in Betracht gezogen, dass auch die BER dieser Sequenz von einem Offset der anderen Sequenz beeinflusst wird: Die Kreuzkorrelationsfunktion (KKF) zweier M-Sequenzen hat einen im Gegensatz zur AKF sehr kleinen Wert. Durch die Frequenzverschiebung werden effektiv aber keine zwei M-Sequenzen mehr korreliert, wodurch die KKF einen höheren Wert hat und das decodierte Ergebnis verfälscht wird. Dieser Effekt ist gering, wird aber deutlich, wenn viele Sequenzen benutzt werden, deren KKF die Korrelationsamplitude beeinflussen, wie in Abschnitt 3.2 zu sehen ist. Zur Anschauung nutzen zwei Teilnehmer jeweils 128 bzw. 127 Sequenzen und der Frequenzoffset beeinflusst alle Sequenzen eines Teilnehmers. Abbildung 5.20 zeigt die BER-Kurven der ersten Sequenz, die Teilnehmer A nutzt, und Sequenz Nr. 129, die von Teilnehmer B genutzt wird und frequenzverschoben ist. Durch die Überlagerung mehrerer verschobener Sequenzen, die nun effektiv nicht mehr als M-Sequenzen angesehen werden können, ist eine deutliche Verschlechterung der BER der unverschobenen Sequenz sichtbar.

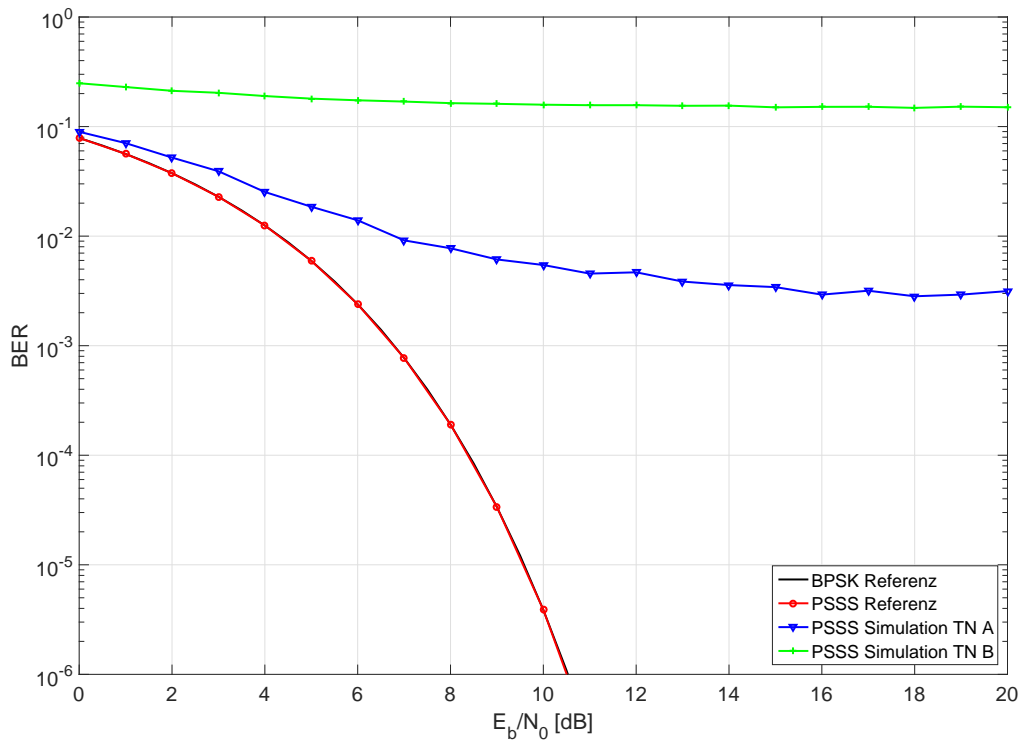


Abbildung 5.20: BER Kurven von Teilnehmer A / Sequenz 1 und Teilnehmer B / Sequenz 129; Teilnehmer A belegt die ersten 128, Teilnehmer B die übrigen Sequenzen; Teilnehmer B ist frequenzverschoben

5.5 Dynamische Schwellwertanpassung

Bisher wurde bei der Decodierung die Nullschwelle als Entscheidungsschwelle genutzt. Da diese aber eigentlich von der Zusammensetzung des Informationssymbols abhängig ist, wird ein zweiter Decodierschritt mit dynamisch angepassten Schwellen durchgeführt, welcher in Abschnitt 3.3 beschrieben ist. Abbildung 5.21 zeigt die BER-Kurve von PSSS-255 mit und ohne dynamische Schwellwertanpassung. Diese wurde sowohl mit idealen Schwellen, die aus dem Sendesymbol berechnet wurden, als auch mit aus dem Empfangssignal berechneten realen Schwellen durchgeführt. Es ist zu erkennen, dass die BER-Kurve durch den zweiten Decodierschritt fast auf der BPSK-Kurve liegt. Dies war zu erwarten, da die dynamische Anpassung der Entscheidungsschwellen, wie in Abschnitt 3.3 beschrieben, einer Orthogonalisierung des Signals entspricht, sofern nur wenige Fehler bei der vorherigen Decodierung gemacht wurden.

In Abbildung 5.22 sind die entsprechenden BER-Kurven für PSSS IEEE 802.15.4-2011 mit 868 MHz aufgetragen. Hier hat die sich ergebende BER-Kurve einen deutlichen Abstand zur

BPSK-Kurve. Aufgrund der kürzeren Codelänge fallen einzelne Decodierfehler hier stärker ins Gewicht, sodass keine optimale Orthogonalisierung stattfindet. Es ist möglich, den zweiten Decodierschritt mehrfach hintereinander durchzuführen und die Schwellen jeweils neu anzupassen, um die BER noch weiter zu verbessern. Bei PSSS-255 ist dies jedoch nicht notwendig und nicht sinnvoll, da die BER ohnehin fast auf der BPSK-Kurve liegt.

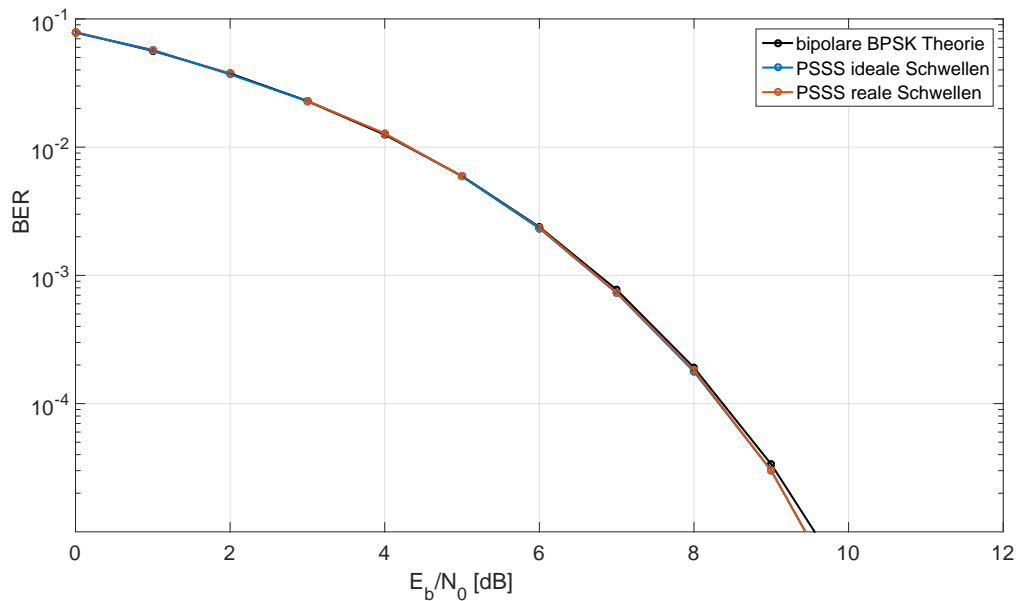


Abbildung 5.21: BER bei dynamischer Schwellenanpassung, $n_R = 255$

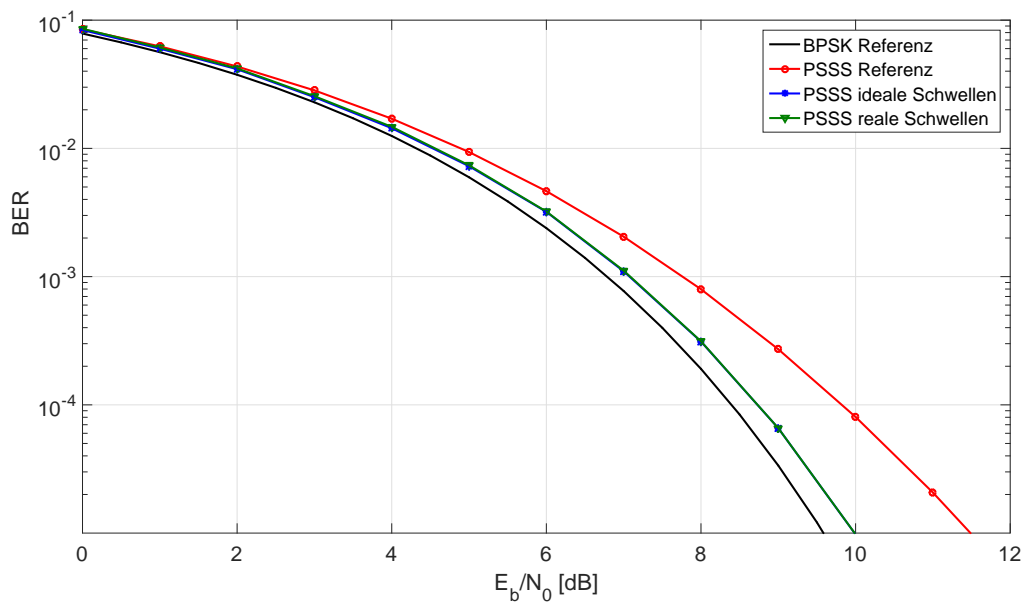


Abbildung 5.22: BER bei dynamischer Schwellenanpassung, $n_R = 31$

6 Fazit

In diesem Kapitel werden die Ergebnisse dieser Arbeit zusammengefasst und abschließend bewertet. Außerdem wird ein Ausblick auf mögliche Erweiterungen der Arbeit gegeben.

6.1 Zusammenfassung

Im Rahmen dieser Arbeit wurde ein PSSS-255-PHY implementiert, der auf dem in [11] implementierten PSSS-PHY nach IEEE 802.15.4-2011 aufbaut. Es wurde gezeigt, dass die Bitfehlerrate abhängig von der Länge der Basissequenz und des Sendesymbols ist. Die Implementierung des AWGN-Kanals wurde durch Simulation einer bipolaren BPSK mit Hilfe der entsprechenden theoretischen Kurve validiert. Die Validierung der Simulation mit PSSS-Codierung erfolgte anhand der in Abschnitt 3.2 hergeleiteten Formel.

Es wurde außerdem ein RRC-Filter entworfen, das eine zehnfache Überabtastung und eine Impulsformung mit Roll-Off-Faktor $r = 1$ durchführt. Aufgrund der Nutzung eines AWGN-Kanals führt dies im Gegensatz zu idealer Impulsformung nicht zu einer Verschlechterung der Bitfehlerrate. Allerdings hat die cos-förmige Filterung Auswirkungen auf die BER bei einer Laufzeitverzögerung, da bei einer Abtastung zum falschen Zeitpunkt ein Wert entsprechend der Stelle der gefilterten Flanke gemessen wird. Eventuelle weitere Auswirkungen der RRC-Filterung werden sich beim Einsatz anderer Kanalmodelle zeigen.

Ein Laufzeitunterschied zwischen PSSS-Sequenzen verschiedener Teilnehmer entspricht einer zyklischen Verschiebung und führt somit zu einer Verschlechterung der Bitfehlerrate der betroffenen Sequenzen proportional zur Verschiebung. Wird eine Sequenz von zwei verzögerten Sequenzen beeinflusst, so steigt die BER mit der Verschiebung schneller an. Bei einer angenommenen maximalen Laufzeitverzögerung von einem Chip wird jeweils nur die erste Sequenz benachbarter Teilnehmer beeinflusst. Nicht genutzte Sequenzen können daher zwischen die Teilnehmern gelegt werden.

Auch ein Unterschied in der Empfangsleistung der Teilnehmer beeinflusst die BER des Teilnehmers mit der geringeren Leistung. Je kleiner der CIR wird, desto schlechter wird die BER, allerdings besteht hier eine zusätzliche Abhängigkeit von der Anzahl der genutzten Sequenzen. Die Verschlechterung der BER-Kurve entlang der CIR-Achse verläuft entsprechend der

Wahrscheinlichkeitsverteilung der Anzahl der Störsequenzen. Für die Nutzung aller vorhandenen M-Sequenzen beginnt dies schon bei einem CIR von 0 dB.

Erhält eine Sequenz einen Phasen- oder Frequenzoffset, so steigt ihre Bitfehlerrate, da eine Phasenverschiebung einer Verringerung der Leistung entspricht und somit das Signal-Rausch-Verhältnis verschlechtert wird. Bereits für übliche Frequenzabweichungen von 10 ppm befindet sich die BER konstant im Bereich von 0,5. Die Veränderung der Frequenz bewirkt außerdem, dass am Empfänger effektiv keine M-Sequenzen mehr korreliert werden, sodass die Korrelationsamplitude von störenden Bits stärker beeinflusst wird und somit auch die Bitfehlerrate der anderen Sequenzen steigt. Dies geschieht allerdings erst sichtbar, wenn eine große Anzahl der Sequenzen frequenzverschoben ist.

Um die durch die Nutzung von PSSS als CDMA-Verfahren entstandenen Verschlechterungen der Bitfehlerrate zu kompensieren, wurde eine Erweiterung des Empfängers implementiert, bei der eine dynamische Schwellwertanpassung durchgeführt wird. Durch diese wird für kleine Fehlerzahlen eine Orthogonalisierung des Signals erreicht, sodass die Korrelationsamplitude nicht mehr von Störbits beeinflusst wird. Somit kann eine Bitfehlerrate erreicht werden, die auf der BPSK-Kurve liegt. Die Schwellwertanpassung funktioniert allerdings nur mit senderseitiger Leistungsanpassung, da für einen $CIR \neq 0$ dB die Schwellen nicht korrekt bestimmt werden können.

Die PSSS-Codierung an sich führt auf einem AWGN-Kanal zu einer Verschlechterung der Bitfehlerrate im Gegensatz zur bipolaren BPSK. Die durch die Nutzung als CDMA-Verfahren zusätzlichen Faktoren haben ebenfalls einen negativen Einfluss auf die Bitfehlerrate, sodass sich eine schlechte Leistungsfähigkeit des Verfahrens ergibt. Für eine Abschätzung der Gesamtleistungsperformanz von PSSS-255 sind weitere Untersuchungen zu den oben genannten Einflüssen sowie mit für Industrieumgebungen üblichen Kanalmodellen von Nöten.

6.2 Ausblick

Das Potential des PSSS-255 Verfahrens lässt sich nicht allein anhand von Simulationen mit einem AWGN-Kanal abschätzen. Um Angaben zur Störfestigkeit beispielsweise bei Mehrwegeausbreitung machen zu können, sind weitere Untersuchungen mit speziell für Industrieumgebungen ausgelegten Kanalmodellen erforderlich. Auch der dann auftretende Einfluss der Impulsformung muss untersucht werden.

Um genauere Aussagen über den Einfluss von Laufzeitverzögerungen sowie Frequenz- und Phasenoffsets treffen zu können, kann eine theoretische Herleitung für die Berechnung der BER unter diesen Einflüssen erfolgen. Die Herleitung der Bitfehlerrate für einen $CIR \neq 0$ dB kann zudem für eine beliebige Teilnehmerzahl erfolgen.

Das Senden eines Symbols, das kein Precoding durchlaufen hat, würde zu einer sehr schlechten Ausnutzung des Dynamikbereichs des Senders führen. Das Precoding muss bei PSSS-255 allerdings für jeden Teilnehmer gesondert durchgeführt werden. Die Auswirkungen auf die Bitfehlerrate müssen untersucht werden.

Um eine Verminderung des Einflusses von Laufzeitverzögerungen zu erreichen, kann die Anzahl nicht genutzter Sequenzen so aufgeteilt werden, dass zwischen zwei Teilnehmern jeweils eine größere zyklische Verschiebung entsteht. Die Verringerung des Einflusses ist abhängig von der Anzahl der freien Sequenzen, wird aber in jedem Fall zu einer Verbesserung der Bitfehlerrate führen.

A PSSS-Code-Tabelle

Table 81—PSSS code table used in symbol-to-chip mapping for 868 MHz

Sequence number	Chip number	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
2	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
3	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
4	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
5	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
6	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
7	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
8	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
9	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
10	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
11	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
12	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
13	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
14	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
15	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
16	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
17	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
18	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
19	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

Table 82—PSSS code table used in symbol-to-chip mapping for 915 MHz

Sequence number	Chip number	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
2	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
3	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
4	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

Abbildung A.1: PSSS-Code-Tabelle aus [1]

Abkürzungsverzeichnis

AKF	Autokorrelationsfunktion
ASK	Amplitude Shift Keying
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CDMA	Code Division Multiple Access
CIR	Carrier to Interferer Ratio
DFE	Decision Feedback Equalizer
DSSS	Direct Sequence Spread Spectrum
FDMA	Frequency Division Multiple Access
FHSS	Frequency Hopping Spread Spectrum
GI	Guard Intervall
GPS	Global Positioning System
ICI	Inter Code Interference
IEEE	Institute of Electrical and Electronics Engineers
ISI	Inter Symbol Interference
ISM	Industrial, Scientific and Medical
KKF	Kreuzkorrelationsfunktion
LED	Light Emitting Diode
LFSR	Linear Feedback Shift Register
LOS	Line Of Sight
PHR	PHY Header
PHY	Physical Layer
PN	Pseudorandom Noise
PPDU	PHY Protocol Data Unit
ppm	parts per million
PSDU	PHY Payload

PSSS	Parallel Sequence Spread Spectrum
RRC	Root-Raised Cosine
SFD	Start-of-Frame Delimiter
SHR	Synchronisation Header
SRD	Short Range Devices
TDMA	Time Division Multiple Access
UMTS	Universal Mobile Telecommunications System

Symbolverzeichnis

A_{\max}	maximale Amplitude
A'_{\max}	maximale Amplitude nach Precoding 1
A_{\min}	minimale Amplitude nach Precoding 1
A'_{\min}	minimale Amplitude
B	Bandbreite des Nutzsymbols
b	zu sendendes Symbol
b'	PSSS-Symbol nach der Codierung
b''	Symbol nach der Korrelation im Decoder
b'''	detektiertes Symbol nach dem Entscheider
BER	Bandbreite des Nutzsymbols
\underline{C}	Entzerrungsmatrix
C_A	Korrelationsamplitude von Teilnehmer A
C_{AK}	normierte Korrelationsamplitude von Teilnehmer A
C_B	Korrelationsamplitude von Teilnehmer B
C_{BK}	normierte Korrelationsamplitude von Teilnehmer B
C_K	Korrelationsamplitude
\underline{D}	Anteil der Korrelationsamplitude mit $\tau \neq 0$
d_0	erstes Bit im PSSS-Symbol
Δf	maximaler Frequenzoffset
\underline{D}^T	PSSS-Decodiermatrix
E_b	Bitenergie
E_b/N_0	Signal-Rausch-Verhältnis
E	Erwartungswert der Leistung von Teilnehmer A
f_0	nominale Trägerfrequenz
f_c	Trägerfrequenz
f_{Chip}	Chipfrequenz
f_t	Frequenz des Nutzsymbols

G_P	Spreizgewinn
I	Einheitsmatrix
L_C	Spreizfaktor
M	Länge einer PN-Sequenz
M_{OS}	Überabtastungsfaktor
n	Anzahl der störenden Bits
N_0	Rauschenergie
n_0	Anteil der Korrelationsamplitude mit $\tau = 0$
n_1	Anzahl der 1en im PSSS-Symbol
n'_1	Anzahl der 1en im PSSS-Symbol nach Decodierung
n_A	Anzahl der von Teilnehmer A genutzten Sequenzen
n_{A1}	Anzahl der störenden Bits von Teilnehmer A
n_{AWGN}	AWGN-Rauschen
n_B	Anzahl der von Teilnehmer B genutzten Sequenzen
n_{B1}	Anzahl der störenden Bits von Teilnehmer B
n_{Chips}	Anzahl der Chips des PSSS-Signals
n_m	Anzahl der von Teilnehmer M genutzten Sequenzen
n_{-1}	Anzahl der -1en im PSSS-Symbol
n'_{-1}	Anzahl der -1en im PSSS-Symbol nach Decodierung
n_R	Länge einer M-Sequenz in Chips
n_s	Anteil der Korrelationsamplitude mit $\tau \neq 0$
P_A	Empfangsleistung von Teilnehmer A
P_B	Empfangsleistung von Teilnehmer B
P_C	Gesamtleistung des Carriers
P_E	Empfangsleistung
φ	Autokorrelationsfunktion
φ_{max}	maximaler Phasenoffset eines PSSS-Signals
$\varphi(\tau)$	Autokorrelationsfunktion
P_I	Gesamtleistung des Interferers
$\rho(m)$	PSSS-Sequenz
P_N	Leistung des schmalbandigen Störers
$P(n)$	Wahrscheinlichkeit für das Vorhandensein von n störenden Bits
P_S	Leistung des Nutzsignals

R	Nutzdatenrate
r	Roll-Off-Faktor
$r(t)$	Ausgangssignal
S	Symbollänge
CIR	Träger-zu-Interferenz-Verhältnis (CIR)
$s(t)$	Eingangssignal
τ	Verschiebung einer PN-Sequenz in Chips
T_{Chip}	Chipdauer der Spreizsequenz
t_{max}	Dauer einer PSSS-Sequenz
T_n	Symboldauer des Nutzsignals
T_t	Symboldauer des PSSS-Signals
W	Bandbreite der Spreizsequenz

Abbildungsverzeichnis

2.1	FHSS-Verfahren	4
2.2	Erzeugung eines DSSS-Signals	5
2.3	spektrale Leistungsdichte bei DSSS	6
2.4	Rückgewinnung des Informationssignals	7
2.5	Encodierung des PSSS-Signals	8
2.6	Decodierung des PSSS-Signals	8
2.7	Korrelation des zu decodierenden Bits und eines Bits gleichen Werts	9
2.8	Korrelation des zu decodierenden Bits und eines Bits entgegengesetztem Wert	10
2.9	Korrelation einer PSSS-Sequenz nach IEEE 802.15.4-2011 bei 868 MHz	10
2.10	PHY Protocol Data Unit nach IEEE 802.15.4-2011 [1]	11
2.11	Codierung und Modulation der PPDU nach IEEE 802.15.4-2011 [1]	11
2.12	Schematischer Aufbau des Empfängers	12
2.13	Decodierung eines CDMA-Signals	13
3.1	Symbol-to-chip Mapping im Uplink von PSSS-255	15
3.2	Herausforderungen bei PSSS als CDMA	16
3.3	Codierung und Modulation der Nutzdaten bei PSSS-255	16
3.4	Wahrscheinlichkeiten für die Anzahl an Bits mit gleichem Wert für $S = 255$	18
3.5	BER-Kurve für eine Sequenz von Teilnehmer A, $n_A = 128$ und $n_B = 127$, $CIR = 1$	21
3.6	BER von Teilnehmer A, $n_A = 1$, $n_B = 1$, $CIR = -50$ dB	22
3.7	Entzerrung mit Zero Forcing	23
3.8	Entzerrung mit Decision Feedback	24
3.9	Entzerrung mit Feed Forward	24
4.1	Aufbau der implementierten physikalischen Schicht	27
4.2	schematischer Aufbau der Simulationsumgebung	30
5.1	BER-Kurven von PSSS mit M-Sequenzen unterschiedlicher Länge M und variierenden Symbolängen S	33
5.2	Wahrscheinlichkeitsverteilung der Anzahl der konkurrierenden Bits für eine Codelänge von 31 Chips	34
5.3	Wahrscheinlichkeitsverteilung der Anzahl der konkurrierenden Bits für eine Codelänge von 255 Chips	34

5.4	BER-Kurve von PSSS-255 mit einer Symbollänge von 255 Bits	35
5.5	BER-Kurve von PSSS-255 mit einer Symbollänge von 2 Bits	35
5.6	BER-Kurven von Teilnehmer A und B abhängig von der Laufzeitverzögerung, $n_A = 1, n_B = 1$	37
5.7	BER Kurven von Teilnehmer A, B und C abhängig von der Laufzeitverzögerung, $n_A = 1, n_B = 1, n_C = 1$	37
5.8	Fehlerzahl aller 255 Sequenzen nach Simulation, wenn die ersten 20 Sequen- zen um 20 Chips verschoben sind	38
5.9	BER-Kurven aller 255 Sequenzen, wenn die ersten 20 um 20 Chips verschoben sind	38
5.10	Theoretische BER-Kurve von Teilnehmer A für Szenario 1	40
5.11	Theoretische BER-Kurve von Teilnehmer A für Szenario 2	40
5.12	Fehlerwahrscheinlichkeit in Abhängigkeit vom CIR für Szenario 2	41
5.13	Simulierte BER-Kurve von Teilnehmer A und B für Szenario 1	42
5.14	Simulierte BER-Kurve von Teilnehmer A und B für Szenario 2	43
5.15	BER-Kurve von Teilnehmer A, simuliert(blau) und berechnet(rot), $n_A = 1, n_B = 1$	44
5.16	BER-Kurven von Teilnehmer A und B für verschiedene Phasenoffsets von Teil- nehmer B	45
5.17	BER-Kurven für Phasenoffset und Leistungsveränderung im Vergleich	46
5.18	BER-Kurven von Teilnehmer A und B für einen Frequenzoffset von 10 ppm von Teilnehmer, 1 Symbol übertragen	47
5.19	BER-Kurven von Teilnehmer A und B für einen Frequenzoffset von 10 ppm von Teilnehmer, 2 Symbole übertragen	48
5.20	BER Kurven von Teilnehmer A / Sequenz 1 und Teilnehmer B / Sequenz 129; Teilnehmer A belegt die ersten 128, Teilnehmer B die übrigen Sequenzen; Teil- nehmer B ist frequenzverschoben	49
5.21	BER bei dynamischer Schwellenanpassung, $n_R = 255$	50
5.22	BER bei dynamischer Schwellenanpassung, $n_R = 31$	50
A.1	PSSS-Code-Tabelle aus [1]	54

Tabellenverzeichnis

3.1	maximale Umweglängen für ein 30 Chips langes Guard Intervall	17
3.2	Simulationsszenarien	26
4.1	Parametrierung der Module	29
5.1	Vergleich der theoretischen und simulierten BER-Kurven von Szenario 1	42
5.2	Vergleich der theoretischen und simulierten BER-Kurven von Szenario 1	43

Literaturverzeichnis

- [1] Standard IEEE 802.15.4-2011.
- [2] BAUN, C. *Computernetze kompakt*. Springer Verlag, 2013, ch. 3.
- [3] DIXON, R. C. *Spread Spectrum Systems*, 2 ed. John Wiley & Sons, 1984.
- [4] G. R. COOPER, C. D. M. *Modern communications and spread spectrum*. McGraw-Hill, Inc.
- [5] H. SCHWETLICK, A. W. PSSS - Parallel Sequence Spread Spectrum. A Physical Layer for RF Communication. Tech. rep.
- [6] HÖHER, P. A. *Grundlagen der digitalen Informationsübertragung*, 2. auflage ed. Springer Vieweg, 2013.
- [7] J. G. PROAKIS, M. S. *Grundlagen der Kommunikationstechnik*. Pearson Studium, 2004.
- [8] KAMMEYER, K.-D. *Nachrichtenübertragung*. 2008.
- [9] M. K. SIMON, J. K. OMURA, R. B. *Spread Spectrum Communications Handbook*. McGraw-Hill, Inc., 1994.
- [10] NIST – NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. Time and Frequency from A to Z, <http://www.nist.gov/pml/div688/grp40/enc-f.cfm>. 12 2015.
- [11] PETER, E. Implementierung und Analyse eines PSSS-QAM-PHYs für den Einsatz in funkbasierter Industriekommunikation. Master's thesis, Technische Universität Dortmund, 2015.
- [12] PROAKIS, J. G. *Digital Communications*. McGraw-Hill, Inc., 1989.
- [13] R. L. PETERSON, R. E. ZIEMER, D. *Introduction to Spread Spectrum Communications*. Prentice Hall, 1995.
- [14] SCHWETLICK, H. PSSS - Parallel Sequence Spread Spectrum. A Potential Physical Layer for OBAN?
- [15] TORRIERI, D. *Principles of Spread-Spectrum Communication Systems*. Springer, 2015.

- [16] URRIZA, P. Parallel Sequence Spread Spectrum - Orthogonal Frequency Division Multiplexing (PSSS-OFDM) Scheme - a Novel Physical Layer for Robust Wireless Communication Systems. Master's thesis, College of Engineering University of the Philippines, 2009.
- [17] WETZKER, G. *Maximum-Likelihood Akquisition von Direct Sequence Spread-Spectrum Signalen*. PhD thesis, Universität Karlsruhe (T.H.), 1998.
- [18] WOLF, A. *Patent DE 103 01 250 A1*. 2004.
- [19] ZVEI – ZENTRALVERBAND ELEKTROTECHNIK- UND ELEKTRONIKINDUSTRIE E.V. *Koexistenz von Funksystemen in der Automatisierungstechnik - Erläuterungen zum zuverlässigen Parallelbetrieb von Funklösungen*.

Eidesstattliche Versicherung

Croonenbroeck, Ramona
Name, Vorname

128633
Matr.-Nr.

Ich versichere hiermit an Eides statt, dass ich die vorliegende Masterarbeit mit dem Titel
Implementierung und Analyse eines PSSS-PHYs als Codemultiplexverfahren

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Dortmund, 17. Februar 2016
Ort, Datum

Unterschrift

Belehrung:

Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50.000,00 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden. (§ 63 Abs. 5 Hochschulgesetz - HG -)

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird ggf. elektronische Vergleichswerkzeuge (wie z.B. die Software „turnitin“) zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.

Die oben stehende Belehrung habe ich zur Kenntnis genommen:

Dortmund, 17. Februar 2016
Ort, Datum

Unterschrift