

SubSea-Modul zur berührungslosen Daten- und Energieübertragung (cMODEM)

**Fachhochschule Kiel
Fachbereich Informatik und Elektrotechnik**

Prof. Dr. S. Badri-Höher und Sebastian Noack
E-Mail: sabah.badri-hoeher@fh-kiel.de

Telefon: (0431)-210 4243

Endbericht
18. Januar 2018

Inhaltsverzeichnis

1	Kurzfassung	4
2	Einleitung	7
2.1	Aufgabenstellung	7
2.1.1	Beschreibung des Vorhabens	7
2.1.2	Arbeitsaufteilung	8
2.2	Aufbau des Berichts	8
3	Grundlagen	9
3.1	Übertragungskanal	9
3.1.1	Übertragungsfrequenz	9
3.1.2	Arbeitsumgebung	11
3.1.3	Übertragungsprinzip	13
3.1.4	Elektromagnetische Induktion	14
3.2	Signalverarbeitung	14
3.2.1	Kanalkapazität	15
3.2.2	Multiplexing	15
3.2.3	OFDM	16
3.3	Das RedPitaya-System	17
3.3.1	Interfaces	18
3.3.2	AD-/DA-Wandlung	19
3.3.3	Software	21
3.4	Verwendete Libraries	25
3.4.1	Qt	25
3.4.2	Liquid-DSP	26
3.4.3	Red Pitaya Notes	26
3.4.4	Der Qi-Standard	26
4	Entwurf	28
4.1	Antennensystem	29
4.1.1	Spulenentwurf	30
4.2	Hardware	33
4.2.1	Sendeverstärker	33
4.2.2	Empfangsverstärker	35
4.2.3	Sende- und Empfangsumschaltung	36
4.2.4	Energieversorgung	42
4.2.5	Leistungsübertragung	43
4.3	Field Programmable Gate Array (FPGA)-Logik	43
4.3.1	Quadrature Amplitude Modulation (QAM)	43
4.4	Advanced RISC Machines (ARM)-Prozessor	45
4.4.1	CModem-Programm	45
4.4.2	DMA-Treiber	49
4.5	Qi-Umsetzung zur Energieübertragung	50
5	Implementierung und Messung der Komponenten	53
5.1	Hardware	54
5.1.1	Übertragungsspulen	54

5.1.2	TransmissionBoard	56
5.2	FPGA-Logik	60
5.2.1	Empfänger	61
5.2.2	Sender	63
5.3	CModem-Software	63
5.4	Übertragung der Frames	64
6	Zusammenfassung und Ausblick	67
6.1	Zusammenfassung	67
6.2	Ausblick	67
	Literaturverzeichnis	70
	Abbildungsverzeichnis	72

1 Kurzfassung

In diesem Projekt wurde ein System zur gemeinsamen Daten- und Energieübertragung entwickelt und getestet. Für die Datenübertragung wurde aufgrund der hohen Dämpfung elektromagnetischer Wellen bei hohen Frequenzen ein Frequenzbereich von 1 – 10 MHz gewählt. Da im Rahmen dieser Arbeit verschiedene Lösungsansätze entwickelt und getestet werden sollen, wurde ein Ansatz größtmöglicher Flexibilität gesucht. Diese Fähigkeiten bietet ein möglichst weitgehender softwarebasierter Ansatz. Ein Software Defined Radio (SDR) setzt ein flexibles System mit schnellen Analog Digital (AD)- und Digital Analog (DA)- Wandlern und eine hohe Verarbeitungsgeschwindigkeit voraus. Gewählt wurde ein kombinierter ARM/FPGA-Prozessor (Field Programmable Gate Array). Das bereits auf dem Markt verfügbare System *RedPitaya* übertraf die geforderten Eigenschaften.

Auf Basis der RedPitaya-Plattform wurde zunächst ein Software-Defined-Radio-System entworfen. Das zusätzlich entworfene und gebaute analoge Frontend wurde speziell für induktive Übertragung mit Spulen niedriger Impedanz ausgelegt. Auf diesem Gesamtsystem wurden dann, unter Verwendung moderner Bibliotheken der Signalverarbeitung und Berücksichtigung der Eigenschaften des Übertragungskanal, gängige Modulationsverfahren wie z.B. Quadraturamplitudenmodulation QAM und Orthogonal Frequency Division Multiplexing (OFDM) implementiert. Bei der Übertragung kann hierbei zwischen Ein- und Mehrträgerverfahren ausgewählt werden.

Die Kommunikation wurde anschließend mit Daten einer Messung in der Schwentine verifiziert. Dieser Demonstrator bildet den ersten Schritt eines Modems für die Fahrzeug-Basisstation und Fahrzeug-Fahrzeug Kommunikation autonomer Unterwasserfahrzeuge über kurze Entfernungen.

Die Energieübertragung unter Wasser stellt ebenso wie die Datenübertragung eine große Herausforderung dar. Wasserdichte Steckverbindungen sind schwierig auf Dauer dicht zu halten, drahtlose Übertragungsverfahren sind bisher nicht entwickelt oder gar standardisiert. Der Schwerpunkt der Arbeit lag auf der Entwicklung einer neuen induktiven Datenübertragung und darauf aufbauend die Übertragung von Energie. Für die Energieübertragung wurde der Qi-Standard für Überwasser-Energieübertragung gewählt. Dieser Standard wird von vielen Komponentenherstellern unterstützt.

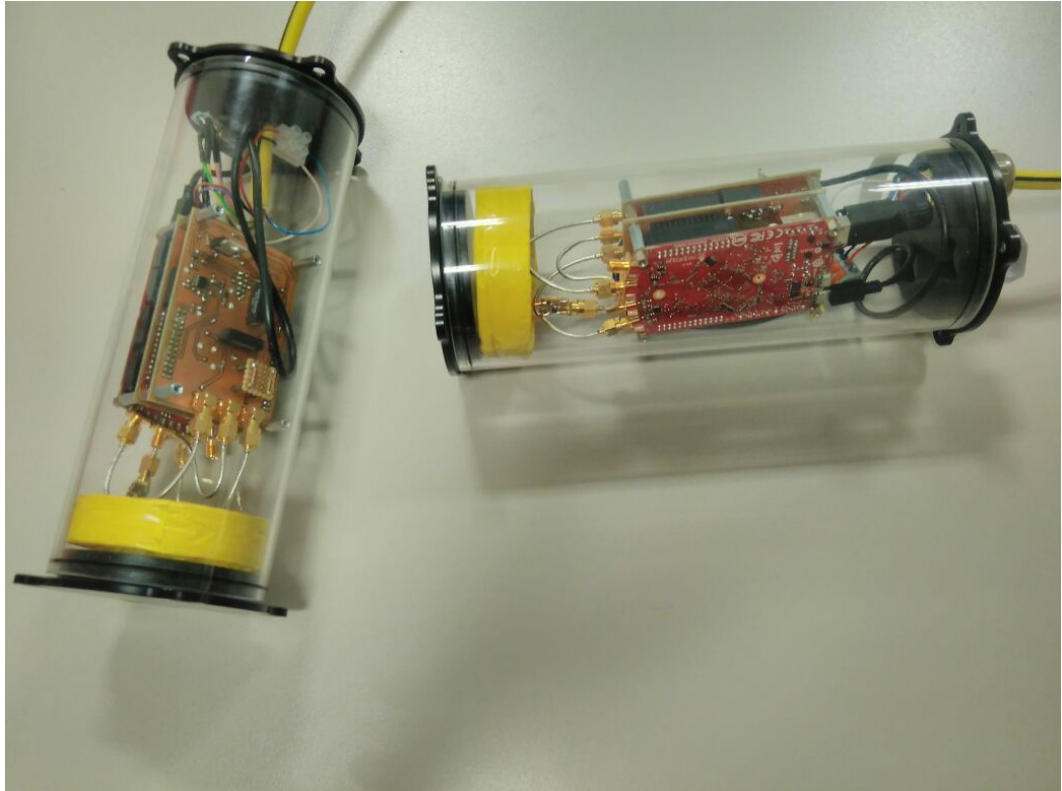


Abbildung 1.1: cModem: Ein System zur gemeinsamen Daten- und Energieübertragung

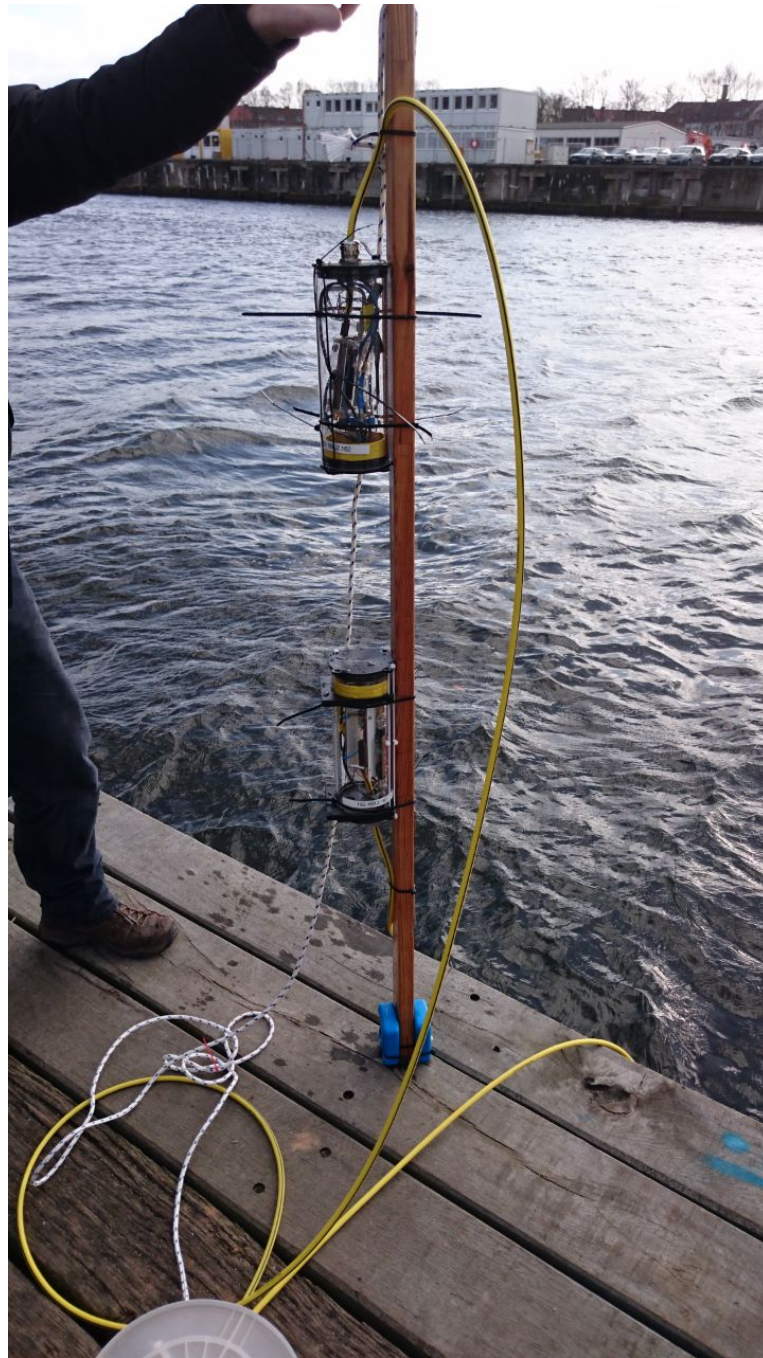


Abbildung 1.2: cModem: Messungen mit dem cMODEM-Demonstrator an der Kieler Förde

2 Einleitung

In der Unterwassertechnik gewinnen kleine, autonom betriebene Unterwasserfahrzeuge, sogenannte Autonome Unterwasservehikel (AUV), immer mehr an Bedeutung. Aufgrund der geringen Baugröße ist deren Einsatzdauer jedoch begrenzt, denn ein bedeutender Anteil der Gesamtmasse wird durch die Akkumulatoren bestimmt. Insofern sind Unterwasser-Ladestationen ein wichtiges Hilfsmittel zur Verbesserung der Einsatzmöglichkeiten. Gleichzeitig wäre es wünschenswert, die riesigen Datenmengen, die ein AUV während der Mission sammelt (typischerweise Kartographie- und Bilddaten, aber auch Umweltdaten), direkt am Meeresboden auszulesen. Ziel der aktuellen Entwicklungen sind sogenannte Docking-Stationen, die man sich wie Garagen mit Internetanschluss vorstellen kann. Problematisch in der Unterwassertechnik sind regelmäßig die Steckverbindungen und sehr limitierte Möglichkeiten zur Datenübertragung. Eine Alternative zu Docking-Stationen sind kleine seetaugliche Module, mit denen sowohl Energie als auch Daten berührungslos übertragen werden können. Die Entwicklung und Erprobung eines Prototypen ist Gegenstand dieses Vorhabens. Der innovative Ansatz besteht darin, das magnetische Feld, welches zur Energieübertragung verwendet wird, mit Daten zu modulieren. Insofern ist für die Datenübertragung keine weitere Energie notwendig. Eine Datenübertragung mittels magnetischer Induktion ist auch in stark gestörten Umgebungen einsetzbar, in denen akustische und optische Modems versagen würden.

2.1 Aufgabenstellung

Ziel des Projektes cMODEM ist die Entwicklung und Erprobung eines Prototypen zur induktiven Daten- und Energieübertragung. Der innovative Ansatz besteht darin, das magnetische Feld, welches zur Energieübertragung verwendet wird, mit Daten zu modulieren. Eine Datenübertragung mittels magnetischer Induktion ist auch in stark gestörten Umgebungen einsetzbar, in denen akustische und optische Modems versagen würden.

2.1.1 Beschreibung des Vorhabens

Arbeitspakete

- AP1: Entwicklung eines hochratigen Datenmodems basierend auf magnetischer Induktion (Software- & Algorithmenentwicklung)
- AP2: Entwicklung eines Hardware-Prototypen
- AP3: Aufbau einer Ladestation
- AP4: Integration der Software in die Hardware (Software-Defined-Radio)
- AP5: Durchführung von Unterwassertests & Gesamtsystemoptimierung

- AP6: Dokumentation und Berichterstellung

2.1.2 Arbeitsaufteilung

- Projektleitung und Durchführung durch FH Kiel
- Wissenschaftliche Unterstützung durch TF/CAU
- Praktische Unterstützung durch SubCtech GmbH
- Projektverwaltung durch FuE-Zentrum FH Kiel GmbH

2.2 Aufbau des Berichts

Nach dieser Einleitung folgt im Kapitel 3 eine Beschreibung der ersten Ansätze zur Datenübertragung basierend auf einem SDR-Konzept. Kapitel 4 beschreibt Konzepte zum Entwurf von induktiver Daten- und Energieübertragung. Kapitel 5 enthält die Entwicklungsarbeiten sowie die Messungen des Systems. Im letzten Kapitel folgt eine Zusammenfassung sowie ein Überblick über die nächsten Arbeiten.

3 Grundlagen

In diesem Kapitel werden die Grundlagen für das erstellte Unterwasserkommunikationssystem präsentiert. Eine Teilaufgabe des Projekts besteht im Aufbau eines Subsystems zur digitalen elektromagnetischen Kommunikation. Aufgrund der hohen Dämpfung elektromagnetischer Wellen bei hohen Frequenzen sind in diesem Kapitel Untersuchungen zum Übertragungskanal sowie zur Übertragungsfrequenz beschrieben worden.

Da im Rahmen dieser Arbeit verschiedene Lösungsansätze entwickelt und getestet werden sollen, wurde ein Ansatz mit größtmöglicher Flexibilität gesucht. Diese Fähigkeiten bietet ein möglichst weitgehender softwarebasierter Ansatz. Ein Software Defined Radio (SDR) setzt ein flexibles System mit schnellen AD- und DA-Wandlern und hohe Verarbeitungsgeschwindigkeit voraus.

Gewählt wurde daher ein kombinierter ARM/FPGA-Prozessor (Field Programmable Gate Array). Das bereits auf dem Markt verfügbare System RedPitaya übertraf die geforderten Eigenschaften.

Die Energieübertragung unter Wasser stellt ebenso wie die Datenübertragung eine große Herausforderung dar. Wasserdichte Steckverbindungen sind schwierig auf Dauer dicht zu halten, drahtlose Übertragungsverfahren sind bisher nicht entwickelt oder gar standardisiert. Der Schwerpunkt der Arbeit lag auf der Entwicklung einer neuen induktiven Datenübertragung und darauf aufbauend die Übertragung von Energie. Für die Energieübertragung wurde der Qi-Standard für Unterwasser-Energieübertragung gewählt. Dieser Standard wird von vielen Komponentenherstellern unterstützt und auf absehbare Zeit auch für höhere Leistungen, als die derzeit angestrebten 15 W, spezifiziert.

3.1 Übertragungskanal

Für den Entwurf eines optimalen Kommunikationssystems spielt der Übertragungskanal eine elementare Rolle. Keine Aussagen und Designentscheidungen können getroffen werden, ohne zumindest die Rahmenbedingungen der Arbeitsumgebung des Systems zu kennen. Deshalb soll hier zunächst auf das Prinzip der Übertragung, die angenommene Umgebung sowie anschließend auf die Kanaleigenschaften eingegangen werden, die dann den Übergang zur Signalverarbeitung, behandelt unter Abschnitt 3.2, bildet.

3.1.1 Übertragungsfrequenz

Bei der Wahl der eigenen Sendefrequenzen ist stets auf eine gültige Zuteilung der jeweiligen Zulassungsbehörde, in Deutschland der Bundesnetzagentur zu achten, um andere Teilnehmer nicht zu stören und teils empfindliche Strafen und Entschädigungen bei Verstößen zu

vermeiden. Die Entscheidung für oder gegen einen entsprechenden Frequenzbereich ist also weniger von technischen, als vielmehr juristischen Faktoren abhängig. Eine vollumfängliche Betrachtung dieser Probleme würde jedoch sowohl den inhaltlichen, als auch den zeitlichen Rahmen dieser Arbeit sprengen. Ferner wird die hier entwickelte Übertragungslösung für die Nahfeldkommunikation in Salzwasser eingesetzt, was aufgrund der Leitfähigkeit des Übertragungsmediums eine für andere Nutzer störende Emission ausschließt. Betrachtet man die technischen Grenzen, so sind folgende Faktoren zu berücksichtigen:

1. Wie unter Punkt 3.3.2 gezeigt, beträgt die maximale Eingangsfrequenz des *RedPitaya-Boards* 50 MHz. Da das Modem als SDR-System aufgebaut werden soll, ist die Anzahl der externen Komponenten möglichst klein zu halten, um diesen Ansatz nicht zu untergraben, wodurch externe Mischer nicht berücksichtigt werden.
2. Daten- und Energieübertragung sollen möglichst hohe Reichweiten erzielen. Die Reichweite ist, wie unter Abschnitt 3.1.3 gezeigt, indirekt proportional zur Ausdehnung des Nahfelds.
3. Daten- und Energieübertragung sollen möglichst unabhängig voneinander arbeiten. Dadurch werden Störübertragungen bereits im Vorfeld ausgeschlossen und die gegenseitige Beeinflussung kann minimiert werden. Ferner können Energie- und Datenübertragungen so als unabhängige Teilprojekte behandelt werden, was das Projektrisiko eines Scheiterns einer Komponente reduziert und unabhängige Bearbeitung der Einzelthemen ermöglicht.
4. Der zur Verfügung stehende Bauraum ist sehr begrenzt. Da der benötigte Platz analoger Komponenten, wie Spulen und Kondensatoren mit der elektrischen Größe auch ab- und zunimmt, kann mit höheren Frequenzen meist ein geringerer Bauraum erreicht werden.
5. Für die Energieübertragung ist es sinnvoll, auf gängige Industriestandards setzen zu können. Der einzige Standard, der zumindest in naher Zukunft die erforderlichen Leistungen (Nach Erfahrungen des *AUV Team TomKyle* benötigt ein tauchendes Autonomous Underwater Vehicle (AUV) mittlerer Größe mindestens ca. 100 W) übertragen kann, ist der *Qi*-Standard. Um mit diesem Standard arbeiten zu können, muss die Frequenz der Energieübertragung im Bereich 87...205 kHz liegen [QIWP, 2016].

Aufgrund dieser Faktoren wurde das Band der Energieübertragung auf 100 kHz festgelegt. Für die Mittenfrequenz des Datensignals wurde nicht das unterste Band des Short Range Service (SRS) im Industrial, Scientific and Medical band (ISM) gewählt, sondern eine tiefere Frequenz von 5.5 MHz. In diesem Frequenzband liegen gemäß [BNAFR, 2016] Ozeanographische Radare sowie zivile und militärische Luftfunkdienste. Für eine industrielle Verwertung des Produktes muss die Mittenfrequenz also in jedem Falle angepasst werden, auch wenn in der Arbeitsumgebung unter Wasser eine Beeinflussung außerhalb des Nahfelds annähernd ausgeschlossen werden kann (siehe Abschnitt 3.1.2).

3.1.2 Arbeitsumgebung

Einsatzzweck des entwickelten Modems ist die drahtlose Unterwasserkommunikation mittels elektromagnetischer Wellen. In diesem Abschnitt soll aufgezeigt werden, welche Auswirkung das veränderte Medium gegenüber der üblichen Kommunikation an Luft hat.

Wellenlänge

Die Wellenlänge elektromagnetischer Strahlung ermittelt sich aus:

$$\lambda = \frac{c}{f} \quad (3.1)$$

Die Wellenlänge ist abhängig von der Permittivität (ϵ_r) und der Permeabilität (μ_r) des Mediums:

$$c_{\text{Vakuum}} = \frac{1}{\sqrt{\epsilon_0 \cdot \mu_0}} \quad (3.2)$$

$$c_{\text{Medium}} = \frac{1}{\sqrt{\epsilon_0 \epsilon_r \cdot \mu_0 \mu_r}} \quad (3.3)$$

$$c_{\text{Medium}} = \frac{c_{\text{Vakuum}}}{\sqrt{\epsilon_r \cdot \mu_r}} \quad (3.4)$$

Die Permeabilität ist unabhängig von der Salzkonzentration im Wasser und damit annähernd konstant. Die Berechnung von ϵ_r als Funktion des Salzgehalts ist nicht trivial, [Butler, 1987] gibt die Wellenlänge in Abhängigkeit der Leitfähigkeit (σ) in Salzwasser wie folgt an:

$$\lambda = 1000 \sqrt{\frac{10}{f \cdot \sigma}} \quad (3.5)$$

Für die gewählten Arbeitspunkte Energie- und Datenübertragung zeigt Abbildung 3.1 die sich ergebenden Wellenlängen.

Dämpfung

Die (auch im hier verwendeten Nahfeld gültige) Dämpfung des Feldes ist äquivalent zur Wellenlänge ebenfalls abhängig von Salzgehalt und Frequenz, [Butler, 1987] gibt dazu an:

$$\alpha \left[\frac{\text{dB}}{\text{m}} \right] = 0.0173 \sqrt{f \cdot \sigma} \quad (3.6)$$

Die Freiraumdämpfung eines Kugelstrahlers im Fernfeld ist dagegen lediglich:

$$F = \left(\frac{4\pi \cdot r \cdot f}{c} \right)^2 \quad (3.7)$$

Vergleicht man die Parameter an Luft und in Salzwasser, so werden die Unterschiede schnell deutlich, wie Tabelle 3.1 zeigt.

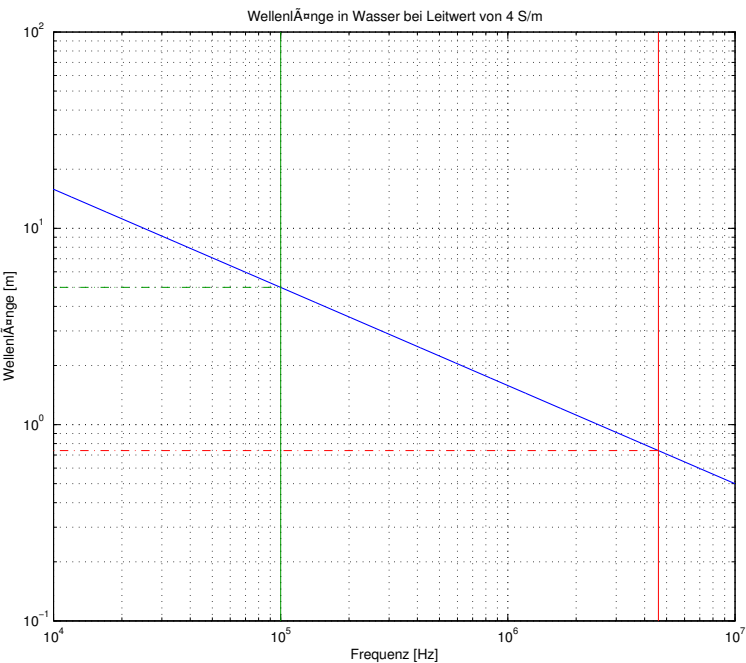


Abbildung 3.1: Wellenlänge in Salzwasser bei $\sigma = 4 \frac{\text{S}}{\text{m}}$

Tabelle 3.1: Dämpfungen für die beiden verwendeten Arbeitsbereiche in Wasser.

	5.5 MHz Luft	5.5 MHz Wasser	100 kHz Luft	100 kHz Wasser
λ	65 m	0.674 m	2998 m	5 m
α		81 $\left[\frac{\text{dB}}{\text{m}}\right]$		11 $\left[\frac{\text{dB}}{\text{m}}\right]$

Diese Werte stellen für die Umgebung in den möglichen Testgebieten ein nahezu unmögliches Szenario dar, da ein Salzgehalt von $4 \frac{\text{S}}{\text{m}}$ in der Ostsee kaum vorkommt. Damit reduziert sich auch Verkürzung der Wellenlänge und die Dämpfung. Stattdessen stellt dieser Wert einen globalen Mittelwert über die Meere dar, da das hier entwickelte Modem nicht nur in der Ostsee Verwendung finden können soll. So werden zu erwartende Probleme in anderen Einsatzgebieten aufgezeigt.

3.1.3 Übertragungsprinzip

Bei der Emission elektromagnetischer Wellen unterscheidet man grundsätzlich zumindest zwei Ausbreitungsregionen: *Nahfeld* und *Fernfeld*. Die Wellenausbreitung im *Fernfeld* kann mit gängigen Modellen gut beschrieben werden. Ausgehend von Maxwell kann mittels $\text{rot} \vec{E} = -\frac{\delta \vec{B}}{\delta t}$ erklärt werden, dass eine zeitliche Änderung des Magnetfeldes (\vec{B}) einen Wirbel des elektrischen Feldes (\vec{E}) bewirkt und umgekehrt. Dadurch bilden sich geschlossene Feldlinien, die eine endlose Fortsetzung dieser Welle bei immer größerer Ausbreitungsoberfläche und damit immer schwächerer lokaler Felddichte bewirken.

Die Gleichungen aus 3.1.2 haben in diesem Feld uneingeschränkte Gültigkeit und die hohe Dämpfung lässt eine Kommunikation über eine Strecke von einem Meter ab Beginn des Fernfeldes als sehr schwierig gestalten.

Das *Fernfeld* beginnt jedoch erst in einer Entfernung von ca. 2λ (unter der Annahme einer gegenüber der Wellenlänge kleinen Antenne) von der Antenne. Das bis zu dieser Entfernung wirkende *Nahfeld*, in dem das hier entwickelte Modem arbeiten soll, verhält sich dagegen anders. Im Nahfeld werden zwei Zonen unterschieden: das *reaktive Nahfeld*, in dem eine direkte Rückwirkung des Empfängers auf den Sender vorliegt sowie die *Fresnel-Region*. Die äußere der beiden Zonen ist die *Fresnel-Region*.

In dieser Region herrscht keine direkte Kopplung zwischen Sender und Empfänger, sie verhält sich diesbezüglich also wie das Fernfeld. Allerdings ist das Zusammenwirken von \vec{B} -Feld und \vec{E} -Feld nicht einfach mit o.a. Gleichungen beschreibbar. So können in diesem Bereich eingebrachte Leiter als Sekundärantennen fungieren, die das Magnetfeld aufnehmen und als neue Antennen ein eigenes Feld aufbauen. Diese Region erstreckt sich von ca. $1 - 2\lambda$.

Innerhalb der einfachen Wellenlänge um die Antenne erstreckt sich das *reaktive Nahfeld*. Innerhalb dieser Region herrscht ein komplexes Zusammenspiel zwischen \vec{B} -Feld und \vec{E} -Feld vor, welche sich hier nicht nur in Ausprägung, sondern auch in Phasenlage auf kurze Distanz stark unterscheiden können. Ferner herrscht hier eine direkte Kopplung zwischen Sendeanterie und Umgebung, insbesondere der ins Nahfeld eingebrachten Empfangsantenne vor. Der extreme Fall kann bei einem Luftspalttransformator beobachtet werden, bei dem die Belastung des einspeisenden Kreises über das Verhältnis der Wicklungen direkt der Belastung durch den Empfänger an der Sekundärwicklung folgt.

Mit zunehmender Nähe zur Antenne (spätestens ab $r \leq \frac{\lambda}{2\pi}$) lassen sich die Effekte besser über die Transformatorgleichungen beschreiben, da die Auswirkungen des Magnetfeldes um den stromdurchflossenen Leiter der Antenne dominieren.

Diese Region soll für das hier entwickelte Übertragungssystem genutzt werden, da bei größeren Abständen einerseits die Dämpfung im Kommunikationskanal zu groß und andererseits der Koppelfaktor für die Energieübertragung zu klein würde.

3.1.4 Elektromagnetische Induktion

Die induzierte Spannung an einem Leiter im Wirkungsbereich eines veränderlichen Magnetfelds wird bestimmt durch die Veränderung des *magnetischen Flusses* Φ :

$$U = \frac{d\Phi}{dt} \quad (3.8)$$

Der magnetische Fluss ist der Teil des *magnetischen Feldes* H , der innerhalb der Fläche der Leiterschleife liegt. Da die Verteilung des Feldes vom umgebenden Medium abhängig ist, wird hierfür die *magnetische Flussdichte* B benötigt.

$$B = \mu \cdot H \quad (3.9)$$

$$\Phi = \int \vec{B} d\vec{A} \quad (3.10)$$

Wird das magnetische Feld von Leiterschleifen erzeugt und die Spannung in einer gleichartigen Schleife wieder induziert, so lässt sich zeigen, dass Φ bei Positionierung der beiden Spulen senkrecht über der jeweiligen Mittenachse maximal wird. Dadurch wird die *magnetische Flussdichte* rotationssymmetrisch zu dieser Achse.

$$\Phi = 2\pi \int \vec{B}(\vec{r}) \cdot \vec{r} d\vec{r} \quad (3.11)$$

3.2 Signalverarbeitung

Das entwickelte Modem soll zur Daten- und Energieübertragung für autonome Unterwasserfahrzeuge dienen, wenn diese sich in der Nähe einer festen Station oder eines anderen Fahrzeugs befinden. Die entscheidenden Eigenschaften dieses sind Datenrate, Fehlerrate, Reichweite und aufgewendete Energie. Die genutzte Energie und zum Teil die erzielte Reichweite unterliegen physikalischen Grenzen, die unter Abschnitt 3.1.3 behandelt und hier nicht weiter thematisiert werden. Die für Daten- und Fehlerrate verantwortlichen Grundlagen werden in diesem Abschnitt besprochen. Ausgangspunkt hierfür ist der in Abschnitt 3.1.2 beschriebene Kanal im Nahfeld um die Sendespule. Wenngleich dieser Kanal aus physikalischer Sicht schwer zu beschreiben ist, so birgt er doch aus Sicht der Signalverarbeitung nicht nur Nachteile:

- negative Effekte durch Mehrwegeausbreitung sind auf dieser kurzen Distanz annähernd unmöglich,
- es können kaum Objekte zwischen Sender und Empfänger eingebracht werden (keine Abschattungseffekte)
- ein Laden des Fahrzeugs erfordert aufgrund der kurzen Reichweiten eine genaue Positionierung oder gar Verankerung des Fahrzeugs, daher kann in der Minimalanforderung (Modem als Steckerersatz) die Annahme eines Linear Time-Invariant (LTI)-Systems gelten,
- schließt man allerdings den Fall der Kommunikation zwischen zwei Fahrzeugen, oder eines sich bewegenden Fahrzeuges mit ein, so muss wegen der vorhandenen Richtcha-

rakteristik des Feldes sowie des komplexen Feldaufbaus von einem Linear Time Variant (LTV)-System ausgegangen werden.

Als Folgerung aus diesen Punkten wird ein System zur Kommunikation in einer LTI-Umgebung realisiert, mit Vorbereitung von Methoden zur Reaktion auf eine sich verändernde Impulsantwort (LTV-Umgebung).

3.2.1 Kanalkapazität

Faktoren, die die Übertragungsrate in der Praxis beeinträchtigen sind zahlreich, lassen sich aber für den hier vorgestellten Kanal auf die bedeutendsten Grenzen einschränken, was unter anderem ein Vorteil der Beschränkung auf die Nahfeldkommunikation ist. Diese Grenzen werden durch das *Shannon-Hartley-Gesetz* (vgl. [Gosling, 1998]) beschrieben und sollen hier kurz erläutert werden.

Bandbreitenlimitierter Kanal

In einem idealen Kanal limitiert lediglich die Bandbreite den Kommunikationskanal. In diesem Falle wird die Kanalkapazität (C) bestimmt durch:

$$C = 2B = 2 \frac{\text{baud}}{\text{s Hz}} \quad (3.12)$$

Es können also 2 Symbole je Sekunde Dauer und Hertz Kanalbandbreite übertragen werden. Bei M Symbole ergibt sich die Kanalkapazität zu:

$$C_b = 2B \cdot \log_2(M) \quad (3.13)$$

Rauschbehafteter Kanal

Ist der Kanal nicht ideal, sondern ein Additive White Gaussian Noise (AWGN)-Kanal, so hat das vorhandene Signal to Noise Ratio (SNR) nach dem *Shannon-Hartley-Gesetz* maßgeblichen Einfluss auf die Kanalkapazität:

$$C_b = B \cdot \log_2 \left(1 + \frac{S}{N} \right) \quad (3.14)$$

Dies stellt allerdings die oberste Grenze in einem reinen AWGN-Kanal mit gleichbleibender Signalleistung dar. Während es theoretisch möglich ist, sich dieser Grenze zu nähern, so bleiben reale Übertragungen davon meist weit entfernt, denn notwendige Schritte wie Synchronisationselemente, Übermittlung von Steuerdaten auf niedrigeren Modulationsstufen, Sendempfangsumschaltung und Folgen von Mehrwegeausbreitungen bleiben dabei komplett unberücksichtigt.

3.2.2 Multiplexing

Zur Übermittlung digitaler Signale stehen im Wesentlichen die Größen Zeit und Frequenz zur Diskriminierung einzelner Komponenten oder Teilnehmer zur Verfügung. Zur Modulation kommen dazu noch Amplitude und Phasenlage hinzu, die sich aber zur Unterscheidung

der Signale mangels zu diesem Zeitpunkt vorhandener Referenz nur bedingt eignen.

Die einfachste Methode, Daten zu übertragen, ist es den Kanal nach einen dieser Kriterien aufzuteilen.

Weist man dazu jedem Kommunikationsteilnehmer einen Zeitabschnitt zu, in dem dieser über die gesamten Ressourcen des Kanals verfügen kann, so spricht man von einem Time Division Multiple Access (TDMA)-System. Wird dagegen der Übertragungskanal in mehrere Unterkanäle aufgeteilt, um mehreren Teilnehmern eine zeitgleiche Kommunikation mit aufgeteilter Bandbreite zu ermöglichen, so ist dies ein Frequency Division Multiple Access (FDMA)-System. Das letzte Verfahren mit wesentlicher Verbreitung zum Multiplexing elektromagnetischer Übertragungskanäle ist das Code Division Multiple Access (CDMA)-Verfahren. In diesem Verfahren wird jedes Sendesignal mit einer dem Sender eigenen Spreizfolge gefaltet. Diese Folgen haben sehr gute Autokorrelationseigenschaften untereinander aber sehr schlechte Kreuzkorrelationseigenschaften gegenüber den Spreizsequenzen anderer Kommunikationsteilnehmer. Dadurch kann ebenfalls von jedem Teilnehmer gleichzeitig auf den Kanal zugegriffen werden.

3.2.3 OFDM

Im Gegensatz zu den unter 3.2.2 angesprochenen Multiplexingverfahren, bei denen es um die Aufteilung der Kanalressourcen an verschiedene Teilnehmer geht, bestehen Zugriffsprobleme auf einzelne Kanalressourcen auch, wenn der Kanal zwischen den Teilnehmern aufgeteilt ist. So kann es Störungen im Frequenzbereich geben, die das SNR auf einzelnen Frequenzberei-

¹Quelle: Lars Wolff; Acoustic Underwater Channel and Network Simulator (Masterprojektarbeit Wintersemester 2011/12)

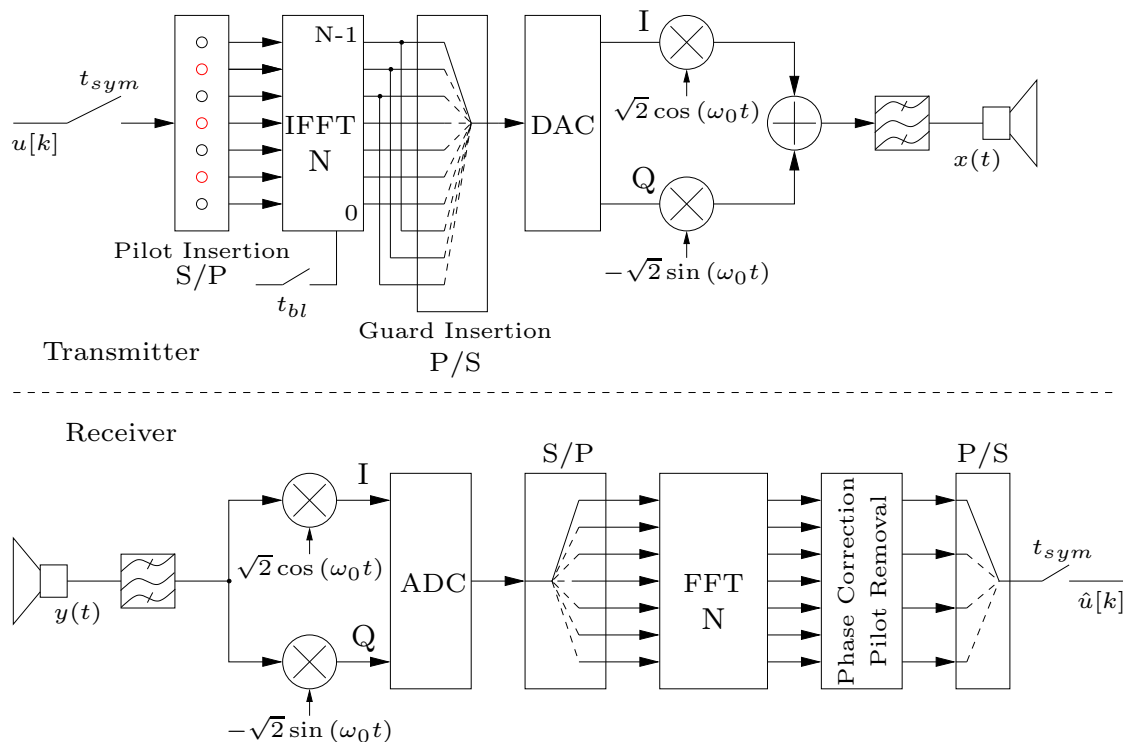


Abbildung 3.2: Überblick über die OFDM-Kommunikation¹

chen des Kanals stark verschlechtern, oder es besteht die Möglichkeit starker Mehrwegeausbreitung, die zu einem mehrfachen, zeitversetzten Empfang des gleichen Signals führt. Bei der Verwendung gängiger Einträgerverfahren, also die einfache Modulation des Datensignals auf einen einzelnen Träger, kann dies zum Abbruch der Kommunikation führen, wenn die Empfangsfehler nicht mehr korrigierbar sind, oder zumindest zum Einsatz robusterer und damit weniger bandbreiteneffizienter Modulations- oder Fehlerkorrekturverfahren zwingen. Um diesen Problemen zu begegnen, kann man den Kanal in mehrere Unterkanäle oder Subträger aufteilen, um Störungen gegen einzelne Subträger mit Abschaltung nur der betroffenen Subträger zu begegnen. Dabei tritt das Problem auf, dass bei Wechsel von einem Datenpaket, oder Frame auf jedem Träger das Modulationssymbol wechseln kann. Da hier natürlich Sprünge im Signalverlauf auftreten können, wird effektiv eine Rechteckfensterung auf jedem einzelnen Subträger vorgenommen, was sich als überlagernde SI-Funktion spektral niederschlägt und damit zu Übersprechen auf andere Subträger führt. Möchte man dies vermeiden, so kann man den Subträgerabstand vergrößern, bis das Übersprechen ausreichend klein wird, oder man versucht die Subträger auf die jeweiligen Nullstellen der anderen Subträger zu legen. Denn entspricht die Länge eines Sendeframes genau der inversen des Subträgerabstandes, so liegt jede Trägerfrequenz eines Subträgers auf den Nullstellen der SI-Funktion aller anderen Subträger.

Ferner kann gezeigt werden, dass die Addition der Mischerergebnisse der einzelnen Subträger genau der Inversen diskreten Fouriertransformation der zu modulierenden Datensignale eines Frames entspricht und umgekehrt die Demodulation der diskreten Fouriertransformation des komplexen Basisbandsignals. Dank schneller und günstiger digitaler Signalprozessoren, lässt sich der gesamte Aufbau damit kostengünstig in Software realisieren. Da die diskrete Fouriertransformation symmetrisch ist, kann durch Kopie des letzten Teils des Sendesignals vor den Anfang ein begrenzter Schutz gegen Inter-Symbol-Interference geschaffen werden, durch das sogenannte Guard-Intervall.

3.3 Das RedPitaya-System

Die Firma *Red Pitaya* [rote Drachenfrucht] ist eine Ausgründung der slowenischen Firma *Instrumentation Technologies*, eines Herstellers von Messinstrumenten für Teilchenbeschleuniger. Das bis heute einzige Produkt, zunächst einfach *RedPitaya-Board*, inzwischen *STEM^{lab}*, ist ein Ein-Platinen-Computer, bestehend aus folgenden für das Projekt relevanten Komponenten:

- einem *Xilinx* ZYNQ[®]7010-System On Chip (SOC),
- einem 14 bit Analog Digital Converter (ADC) ,
- einem 14 bit Digital Analog Converter (DAC) ,
- 512 MB RAM,
- einer *Lantiq* XWAY[™] 7071 Ethernet Phy

Das ZYNQ[®]7010 ist die kleinste Ausbaustufe der Produktreihe aus kombinierten ARM[™]-Prozessoren und FPGA auf einem einzelnen Chip. Dieses Systemdesign erlaubt eine Kom-

bination der Flexibilität von Programmen in Hochsprachen für High-Level Aufgaben bei gleichzeitiger Ausnutzung der Vorteile eines FPGA, namentlich der durch die große Parallelisierbarkeit ermöglichten hohen Rechenleistungen und Hardwarenähe der erzielten Realisierungen.

Die Logikkomponente des Systems wird durch einen ArtixTM-7-FPGA mit 28000 Logikzellen, 17600 Look Up Tables (LUT) und 2.1 Mbit FPGA-eigenem Block-Random Access Memory (RAM) gestellt.

Die Prozessorkomponente besteht aus einem dualcore ARMTM Cortex[®]-A9 Prozessor, der hier mit einer Neon[®]-Vector Floating Point (VFP)V3 Floating Point Unit (FPU) ausgestattet ist und daher auch Single Instruction, Multiple Data (SIMD)-Anweisungen beherrscht. Im Gegensatz zur gewöhnlichen Single Instruction, Single Data (SISD)-Betriebsart, bei der für jede Rechenoperation beide Operanden sowie der Operationscode geladen werden müssen, da hier nur ein Rechenwerk verwendet wird, kann bei diesem Betriebsmodus die gleiche Operation gleichzeitig auf mehrere Datensätze angewendet werden [Menge, 2005].

3.3.1 Interfaces

Neben den bloßen Kapazitäten des Prozessors ist die Betrachtung der in- und externen Interfaces zur Datenübertragung, um die Kommunikation mit externen Komponenten, wie ADC/DAC, oder RAM sowie internen Bestandteilen, wie die Kommunikation zwischen der Programmable Logic (PL) (FPGA-Komponente) und dem Processing System (PS) (ARM-Kern) zu ermöglichen. Auf rein externe Kommunikationsschnittstellen, wie Serial Peripheral Interface (SPI) oder Universal Asynchronous Receiver Transmitter (UART) soll hier dagegen nicht eingegangen werden.

Die im Rahmen dieser Arbeit verwendeten Schnittstellen gehören zur Gruppe der Advanced Microcontroller Bus Architecture (AMBA)-Interfaces. AMBA ist ein von ARM entwickelter und eingeführter, offener Standard zur intra-chip Kommunikation. Während der gesamte Teil des hier beschriebenen Standards unter [AMBA, 2017] abgerufen werden kann, wird hier das in der intra-PL- sowie PS-PL-Kommunikation verwendete Protokoll beschrieben.

Dieses Advanced eXtensible Interface (AXI) genannte Protokoll erlaubt eine schnelle parallele Datenübertragung auf Chipebene, bei der die Übertragung eines vollen Datenwortes im besten Fall in nur einem Taktzyklus erfolgen kann. Zur Vermeidung von Overhead ist die Implementierung in drei Varianten möglich, auf die hier alle kurz eingegangen werden soll. AXI-Interfaces werden in der Regel mit Datenbreiten von 16, 32, 64 oder 128 bit eingesetzt, auch wenn diese Beschränkung nicht Teil der AMBA-Spezifikation ist, sondern in diesem Fall der Implementierung in einigen Intellectual Property (IP)-Cores von *Xilinx* geschuldet ist.

AXI-Full

Ein voll ausgebautes AXI-Interface besteht aus fünf einzelnen parallelen Kanälen:

- Read Address Channel
- Write Address Channel
- Read Data Channel

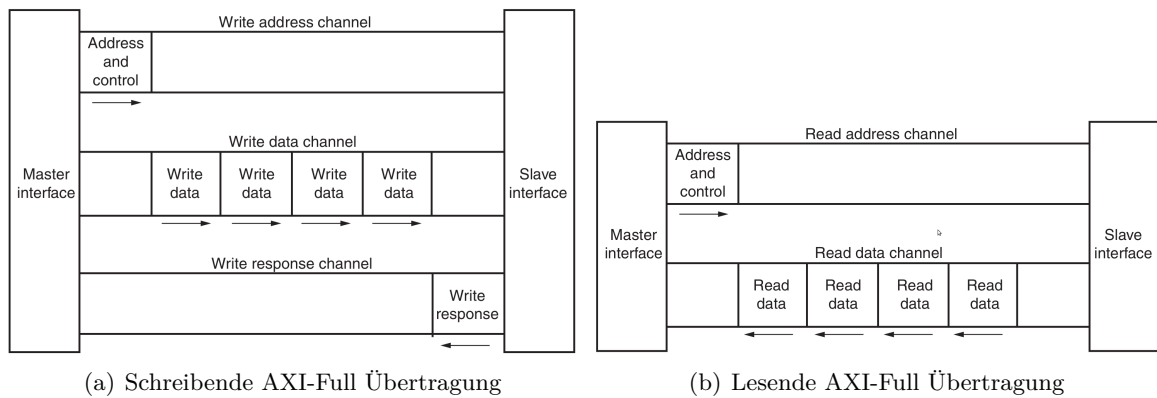


Abbildung 3.3: Lesende und schreibende AXI-Full Kommunikation; Quelle: [UG761, 2011]

- Write Data Channel
- Write Response Channel

Da sowohl Lese- als auch Schreibkanäle vollständig mit eigenen Daten- und Adresskanälen ausgestattet sind, ist eine bidirektionale Datenübertragung möglich. Daten werden dabei in Bursts mit einer Länge von maximal 256 Datenpaketen übertragen. Die Abfolge von Lesenden und schreibenden Übertragungen werden in [UG761, 2011] dargestellt.

AXI-Lite

Das AXI-Lite-Interface unterscheidet sich vom AXI-Full-Interface insofern, als dass keine Burstübertragungen möglich sind. Während also im Full-Modus bis zu $\frac{256}{257} = 99.6\%$ der Übertragungszeit (im Lesemodus bei einer Adressierung und 256 Datenbursts) für die Übertragungszeit genutzt werden kann, sinkt dieses Verhältnis hier auf 50 %. Dieses Interface wird demzufolge hauptsächlich für Konfigurationsdaten verwendet, da es durch die einfachere Implementierung auch weniger Ressourcen des FPGA zur Implementierung benötigt.

AXI-Stream

Das AXI-Stream-Interface wurde für Anwendungen geschaffen, in denen hohe Datenraten bei unidirektionalen Verbindungen benötigt werden. Für diesen Zweck enthält dieses Interface nur den Schreibkanal eines AXI-Full-Interfaces mit den Unterschieden, dass die Adressierung entfällt und die Anzahl der Datenbursts nicht begrenzt ist. Durch die annähernd vollständige Auslastung des Interfaces sind mit dieser Form des AXI-Interfaces Datenraten von bis zu $128 \frac{\text{bit}}{\text{Hz}}$ möglich.

3.3.2 AD-/DA-Wandlung

Der größte Vorteil des *RedPitaya-Boards* gegenüber der großen Auswahl an *ARM-FPGA-Entwicklungsboards* sind die integrierten 2-Kanal ADC und DAC. Sowohl ADC, als auch DAC erreichen auf jeweils zwei Kanälen je $125 \frac{\text{MSamples}}{\text{s}}$. An dieser Stelle werden die Spezifikationen der Wandler gemäß Herstellerdokumentation [RPHD, 2017] kurz aufgelistet, da diese die Schnittstelle zwischen dem implementierten SDR und der entwickelten Hardware darstellen.

DAC

Als DAC kommt in diesem System ein DAC1401D125HL von *Integrated Device Technology* mit anschließendem Ausgangsverstärker und -filter zum Einsatz. Aus der Samplerate von $125 \frac{\text{MSamples}}{\text{s}}$ ergibt sich gemäß Nyquist-Theorem eine maximale Signalfrequenz von $\frac{125 \text{ MHz}}{2} = 62.5 \text{ MHz}$. Das Ausgangsfilter hat gemäß Amplitudengang des Herstellers allerdings eine -3 dB-Grenzfrequenz von 50 MHz.

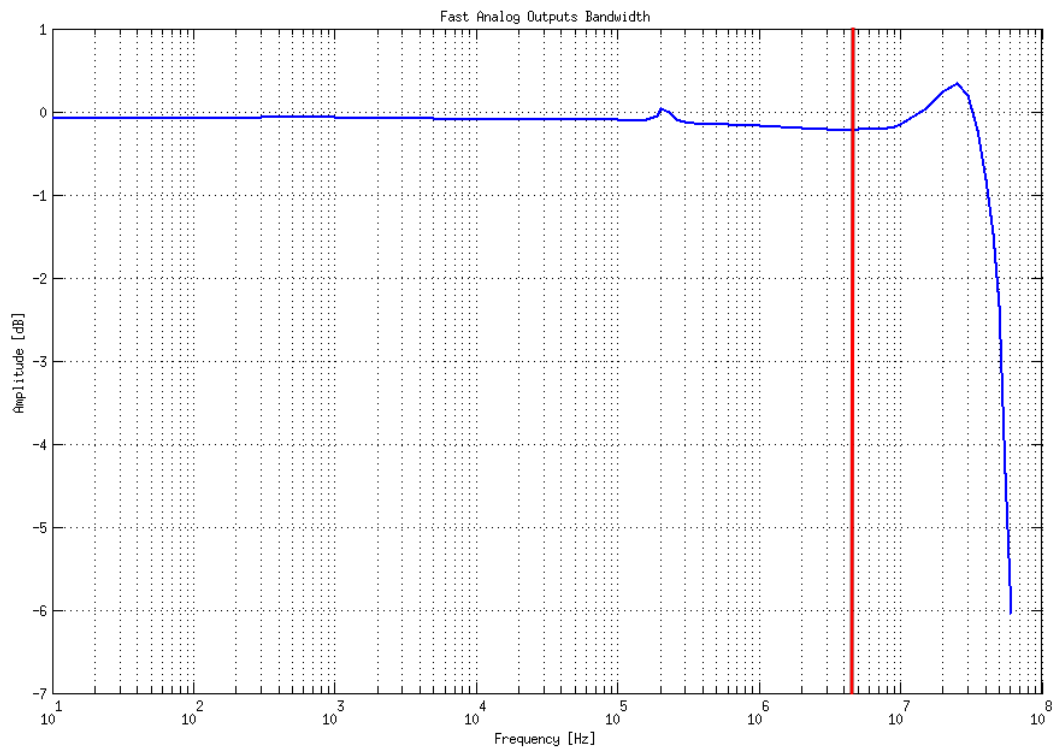


Abbildung 3.4: Amplitudengang des *Redpitaya-Board*-DAC [gewählter Arbeitsbereich in rot eingetragen]; Quelle: [RPHD, 2017]

Der Amplitudengang dieser Ausgangsstufe des Boards ist über beinahe den gesamten Frequenzbereich annähernd konstant, im gewählten Arbeitsbereich besteht eine Dämpfung von ca. 0.2 dB. Das Frontend der Platine besitzt eine gesamte Ausgangsleistung von 10 dBm an 50 Ω , wodurch sich über $U_{pp} = \sqrt{\frac{P \cdot R}{2}} = \sqrt{\frac{10^{-2} \text{ W} \cdot 50 \Omega}{2}}$ eine Ausgangsamplitude von 1 V ergibt.

ADC

Der im *RedPitaya-Board* verwendete ADC LTC2145CUP-14 ist ein 14 bit ADC von *Linear Technologies*. Es sind die zwei Betriebsarten „LV“ und „HV“ möglich, die die Charakteristik des Eingangsverstärkers und -filters ändern, so dass im „LV“-Modus keine und im „HV“-Modus eine -26 dB-Dämpfung im Durchlassbereich erzielt wird ([RPHD, 2017]). Auch dieses Filter hat eine fixe -3 dB-Grenzfrequenz von 50 MHz.

Neben der theoretischen Auflösung eines ADC ist auch immer die effektive Auflösung, oder Effective Number Of Bits (ENOB) wichtig. Dieser Wert berücksichtigt die Fehler während der Wandlung, die selbst unter idealen Bedingungen in jedem ADC auftreten. Wenngleich der Wandler ein Ergebnis mit einer Breite von 14 bit liefert, so sind nur die obersten ENOB davon

wirklich verwertbar, während die untersten 14 – ENOB bit lediglich Rauschen enthalten. Im Umkehrschluss lässt sich ein SNR aus der Auflösung eines ADC errechnen. Je höher ein ADC auflöst, desto besser hebt sich das Signal aus dem Quantisierungsrauschen heraus. Eine Auflösung von $N = 14$ bit ergibt ein maximal mögliches SNR von:

$$\text{SNR} = N \text{ dB} \cdot 6.02 - 1.25 \text{ dB} = 14 \cdot 6.02 \text{ dB} - 1.25 \text{ dB} = 83.03 \text{ dB} \quad (3.15)$$

Für das *RedPitaya-Board* sind in der Dokumentation ([RPHD, 2017]) Angaben für ein SNR von ca. 80 dB und damit ein ENOB von:

$$\text{ENOB} = \frac{\text{SNR} + 1.25 \text{ dB}}{6.02 \text{ dB}} = \frac{80 \text{ dB} + 1.25 \text{ dB}}{6.02 \text{ dB}} = 13.5 \text{ bit} \quad (3.16)$$

Dieses Ergebnis liegt ungewöhnlich nah am idealen Wert und erscheint damit zweifelhaft. Das Datenblatt des ADC [DS2145, 2012] gibt für den Frequenzbereich bis 50 MHz ein SNR von 73 dB an, woraus sich ein wesentlich niedrigeres ENOB ermitteln lässt:

$$\text{ENOB} = \frac{73 \text{ dB} + 1.25 \text{ dB}}{6.02 \text{ dB}} = 12.3 \text{ bit} \quad (3.17)$$

Es wurde daher der schlechtere und plausiblere Wert des Datenblatts übernommen.

3.3.3 Software

Der Hersteller *RedPitaya* liefert mit seinen *RedPitaya-Boards* nicht nur die Hardware, sondern auch eine komplette Softwareumgebung aus. Diese Softwareumgebung umfasst ein Betriebssystem auf *Debian* oder *Ubuntu*-Basis, eine Shared Library zur Steuerung rudimentärer Funktionen und zum Laden eines FPGA-Images sowie ein Application Programming Interface (API) zur Kommunikation mit dem auf *NginX* basierenden *Web UI*-System des Herstellers. Während die Möglichkeit zum Laden verschiedener Images über ein Webinterface für ein variables Messinstrument einen gangbaren Ansatz darstellt, so ist dieser für ein eigenständiges Modem, wie hier geplant, ungeeignet. Neben dem benötigten Ressourcenaufwand für Webserver und Reverse Proxy fallen hier die Einschränkungen und der erhöhte Aufwand für die Nutzung anderer Softwarebibliotheken negativ ins Gewicht.

Da die mitgelieferte Software ohne Modifikation nur eine Speicherung von 16.8 kSamples im Block RAM (B-RAM) zulässt, die dann manuell abgerufen (ADC), bzw. weiter nachgeschoben werden müssen (DAC) können an dieser Stelle auch keine Synergieeffekte aus anderen Projekten auf Basis der bereitgestellten Softwareumgebung gewonnen werden.

Benötigt wird dagegen eine Softwarebasis, die kontinuierliches Sampling mit einstellbarer Samplingrate durch den ADC, sowie automatischer Transport der Ausgangsdaten zum DAC bereitstellt. Die Vorbereitung zu Beginn der Thesis umfasste also die Suche nach einem Basissystem, was diese Arbeiten erledigen konnte, da aufgrund des Projektumfanges jede Arbeit an diesen grundlegenden Punkten, wertvolle zeitliche Ressourcen binden würde.

Es mussten folglich Lösungen für den gesamten Teil Basiseinrichtung, des Bootups sowie des Betriebssystems gefunden werden.

Betriebssystem

Die Wahl des Betriebssystems enthält zunächst die Entscheidung, ob überhaupt eines verwendet werden soll. Gegen ein Betriebssystem sprechen hauptsächlich die effizientere Ressourcennutzung sowie der direkte Hardwarezugriff und damit genaueres Timing auch vom *ARM*-Prozessor aus. Da für zeitkritische Aufgaben allerdings der *FPGA* genutzt werden kann und die effiziente Ressourcennutzung einen erheblichen Aufwand in der Softwareerstellung bedeutet, sind diese Vorteile auch mit erheblichen Nachteilen verbunden. Auch die Benutzung fremder Bibliotheken, was für ein dermaßen umfangreiches Projekt (vor allem unter dem Aspekt der Wart- und Erweiterbarkeit) in dem engen zeitlichen Rahmen unabdingbar ist, wird erschwert, da nur geschlossene Bibliotheken verwendet werden können.

Für die Nutzung eines Betriebssystems spricht die einfache Nutzung von Softwarebibliotheken (siehe Abschnitt 3.4), die Verwendung des Schedulers zur einfacheren Nutzung beider Prozessorkerne, die enthaltene Speicherverwaltung und die deutlich vereinfachte Nutzung der verbauten Hardware auf dem *RedPitaya-Board*, allem voran die Netzwerk-Phy. Bei der Betrachtung der möglichen Betriebssystemalternativen besaß lediglich *Linux* vollumfänglichen Support des Herstellers, weshalb hier keine Alternativen in Betracht gezogen wurden. Bei der Auswahl der begleitenden Distribution war die Zahl der Alternativen wieder deutlich größer. Eingeschränkt konnte diese allerdings wieder unter dem Gesichtspunkt der einfachen Wartbarkeit und des möglichst niedrigen Wartungs- und Erstellungsaufwandes. Es kamen ferner nur Distributionen mit einer guten Unterstützung der *ARMhf*-Plattform in Frage. Die Auswahl musste schließlich zwischen der von *Xilinx* selbst unterstützten *PetaLinux* und einer Debian-artigen Distribution gewählt werden. Aufgrund der größeren Nutzerbasis und der Möglichkeit, in Zukunft die für die Unterwasserrobotik relevanten Frameworks *ROS* und *MOOSE* einsetzen zu können, fiel die Entscheidung auf *Debian*. Die Installation konnte bereits im Vorfeld dieser Arbeit unter Zuhilfenahme des *Red Pitaya Notes*-Projektes (siehe Abschnitt 3.4.3), wenn auch in abgewandeltem Umfang und Form, durchgeführt werden.

Bootvorgang

Der Bootvorgang des *ZYNQ*-Systems erfordert gemäß [UG821, 2015] vier Schritte:

1. Zunächst wird der First Stage BootLoader (FSBL) aufgerufen. Im Falle einer SD-Karte als Bootmedium muss dieser zusammen mit dem initial zu ladenden *FPGA*-Bitstream und dem Second Stage BootLoader (SSBL) am Beginn der ersten *FAT32*-formatierten Partition liegen. Dieser Bootloader setzt grundlegende Einstellungen, wie die Konfiguration der Phase Locked Loop (PLL) zum Betrieb der internen Taktgeneratoren, Aktivierung der verschiedenen internen Interfaces sowie die Startadresse des SSBL.
2. Der anschließend aufgerufene SSBL, in diesem Falle *U-Boot* ist für die Konfiguration und Initialisierung des Bootvorgangs des eigentlichen Betriebssystems verantwortlich. Hier werden Einstellungen wie der nutzbare Bereich des Arbeitsspeichers, die verwendete Standardausgabe sowie Startparameter für den Linuxkernel gesetzt. Anschließend wird der genutzte Kernel ggf. entpackt, an eine konfigurierbare Speicherstelle im RAM kopiert und ein Verweis auf das ebenfalls in den Arbeitsspeicher abgelegte *Devicetree*-File übergeben.

3. Diese *Devicetree* Datei enthält die genutzten Systemkomponenten, insbesondere deren verwendete Systemadressen, zugewiesene Interruptnummern und zur Zuordnung des richtigen Treibers notwendige Identifikationszeichenketten. *Devicetrees* werden in einer eigenen Scriptsprache formuliert und vor Nutzung mittels eines Compilers übersetzt. *Xilinx* liefert hierzu in seinem Software Development Kit (SDK) einen Generator für alle grundlegenden Module, wie *Xilinx*-eigenen und IP-Cores mit. Daher müssen nur noch Anpassungen für eigene IP-Cores sowie nicht von *Xilinx* bereitgestellte Hardware vorgenommen werden. Da *Devicetrees* die Einbindung andere *Devicetrees* vorsehen, kann für die eigenen Anpassungen ein Overlay geschrieben werden, was den mittels SDK erstellten *Devicetree* einbindet und mit den eigenen Änderungen überlagert.
4. Der *Linux*-Kernel wird schließlich geladen und kann seinerseits dank der Informationen der vorangegangenen Schritte die richtigen Treiber auswählen und starten. *Xilinx* stellt auch für diesen Schritt einen geringfügig angepassten *Linux*-Kernel bereit, der auf den *ZYNQ*-Prozessoren lauffähig ist und mittels des im *SDK* ausgelieferten Compilers erstellt werden kann.

Einlesen der ADC-Samples

Der *ZYNQ*-Prozessor erlaubt über ein AXI-Full-Interface (64 bit) den Zugriff auf den RAM-Controller des PS, was das Abspeichern von Daten aus dem FPGA-Teil des Systems mit einer theoretischen maximalen Datenrate von $64 \text{ bit} \cdot 150 \text{ MHz} = 1200 \frac{\text{MB}}{\text{s}}$ ermöglicht.

Dieser Zugriff löst das Problem des Abspeicherns der ADC-Daten, zumal dafür bereits quelloffene IP-Cores verfügbar sind (siehe Abschnitt 3.4.3).

Reservierung des Arbeitsspeichers

Problematisch beim direkten Zugriff auf Bereiche des Arbeitsspeichers ist lediglich, dass die verwendeten Bereiche nicht anderweitig belegt sein dürfen. Da alle Betriebssysteme auch und vor allem den eigenen Arbeitsspeicher verwalten, kann und wird es hier schnell zu Konflikten kommen. Um diese zu vermeiden, wird dem Betriebssystem der zur Speicherung der Eingangsdaten verwendete Bereich nicht überlassen, sondern ein Abschnitt am oberen Ende des Arbeitsspeichers reserviert. Nachteilig bei dieser Methode ist aber, dass einerseits der Speicherbereich nur aufwendig geändert werden kann und dem Betriebssystem dieser Speicher natürlich nicht zur Verfügung steht. Eine dynamische Methode zur Reservierung kohärenten Speichers, wie in 4.4.2 beschrieben wäre eine Alternative an dieser Stelle gewesen, aber da sich die benötigte Speichermenge dank fester Abstrakten nicht zur Laufzeit ändert, wurde diese Möglichkeit hier nicht gewählt, zumal der maximalen Größe des reservierbaren kohärenten Speichers in *Linux*-Systemen praktische Grenzen gesetzt sind.

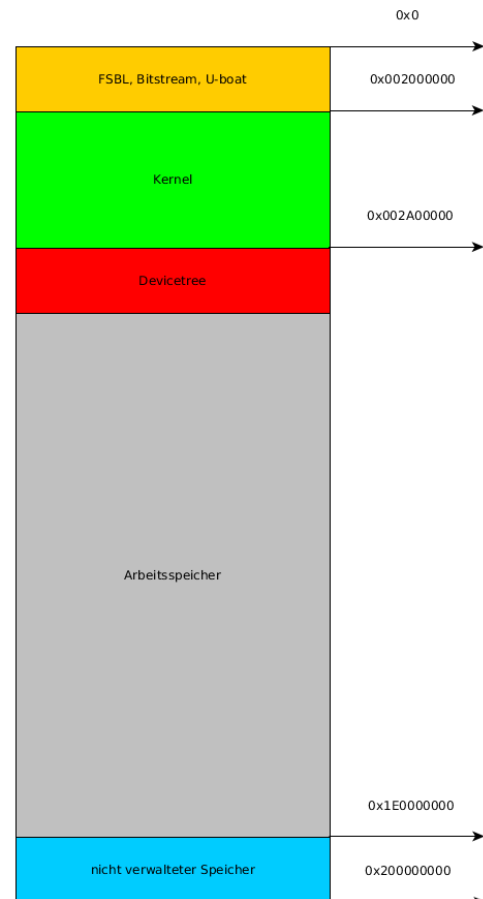


Abbildung 3.5: Beispielhafte Aufteilung des Arbeitsspeichers

Übermittlung der DAC-Samples

Für die Übermittlung der generierten Daten an die Ausgabelogik ist diese Vorgehensweise nicht zielführend. Zwar könnte man auch hier einen Bereich des Arbeitsspeichers reservieren und nach Kopieren der Daten in diesen Bereich die Start- und Endadressen übergeben, allerdings müsste neben dieser Übermittlung auch noch eine Abfrage für den Abschluss der Übertragung erfolgen, denn im Gegensatz zum Eingang ist es im Ausgang nicht gewünscht, permanent die vorliegenden Daten zu übermitteln, sondern es sollen nur zu übermittelnde

Daten gesendet werden. Es kann zwar statt eines Pollings auch dafür ein Interrupt und die Reservierung des Arbeitsspeicherbereiches verwendet werden, allerdings ist es dann sinnvoller die gesamte Übertragung zu automatisieren, um den Prozessor von dieser Arbeit entlasten zu können. Zu diesem Zweck existieren Direct Memory Access (DMA)-Controller, oder im Falle des *ZYNQ* DMA-IP-Cores (vgl. [PG021, 2016]). Diese Controller übernehmen den Transport von Daten aus dem Arbeitsspeicher in ein AXI-Stream-Interface, was unserem Anwendungsfall genau entspricht. Für diese Controller stellt *Xilinx* rudimentäre Treiber bereit, die aber lediglich als Grundlage für eine eigene Treiberentwicklung dienen sollen (vgl. [DMADrivers, 2017]). Daher musste diese Aufgabe auch im Rahmen dieser Arbeit erfüllt werden

3.4 Verwendete Libraries

Wie bereits in der Einführung dieses Kapitels ausführlich beschrieben, erlaubt die große Bandbreite der Arbeit keine ausführliche Bearbeitung aller Teilaspekte. Da dies jedoch in modernen Entwicklungsprozessen durchaus üblich ist, wird dafür üblicherweise direkt auf die Arbeiten anderer aufgebaut. Im Falle von per Software lösbarer Probleme kann dies einfach durch den Einsatz von aus anderen Projekten entstandenen Bibliotheken erfolgen. Während der Einsatz einiger Bibliotheken aus dem modernen Programmieralltag nicht wegzudenken ist und diese keiner Erwähnung bedürfen, so soll an dieser Stelle doch auf einige Bibliotheken eingegangen werden, deren Kenntnis nicht vorausgesetzt werden kann.

3.4.1 Qt

Ursprünglich ab 1990 von Haavard Nord Eirik Chambe-Eng als „object-oriented display system“ [Haavard Nord] erdacht und entwickelt, ist die *Qt*-Library inzwischen immer weiter aufgebaut und auf viele andere Anwendungsgebiete ausgedehnt worden. Während nach wie vor ein starker Fokus auf das Hauptaufgabengebiet, die Abstrahierung grafischer Benutzeroberflächen von der Betriebssystemebene, gelegt wird, werden auch andere Anwendungsfelder erschlossen. Durch die Festlegung auf die Programmiersprache C++ (nebst der selbstentwickelten Scriptsprache *QML*) kann der Kompromiss zwischen Abstrahierung nicht zwingend hardwarenaher Komponenten, wie graphischer Darstellung einerseits und sehr systemnaher Arbeit bei Aufgaben mit hohen Anforderungen an Bearbeitungsgeschwindigkeit andererseits gefunden werden.

Die konsequente Abstrahierung von Programmierschnittstellen mittels Nutzung einer *Signal-Slot*-Methode, bei der jeweils verbundene Funktionen über einen threadeigenen *EventHandler* verwaltet werden, ermöglicht gemeinsam mit der einfachen Threadverwaltung sehr einfach die Skalierung an andere Laufzeitumgebungen und Prozessorsysteme.

Zur Verwendung dieser Library ist zwingend die Nutzung eines Buildsystems notwendig, wobei der Anwender die Wahl zwischen dem bereitgestellten *QMake* oder dem verbreiteten *CMake* Buildsystem hat.

Da für dieses Projekt ohnehin eine *CMake*-Crosscompilerumgebung eingerichtet werden musste, waren außer dem einmaligen Cross-Compiling der Library keine Anforderungen zu erfüllen. Im vorliegenden Projekt wurde *Qt* daher zur Abstrahierung der einzelnen Programmobjekte,

Threadverwaltung und Erstellung des Transmission Control Protocol (TCP)/Internet Protocol (IP)-Servers sowie Verwaltung von dessen Verbindungen genutzt.

3.4.2 Liquid-DSP

Das noch recht junge *Liquid-DSP*-Projekt entstand durch Joseph Gaeddert 2011 im Rahmen seiner Dissertation „Facilitating Wireless Communications through Intelligent Resource Management on Software-Defined Radios in Dynamic Spectrum Environments“ und hat das Ziel, eine leichtgewichtige, aber effiziente Bibliothek für SDR-Anwendungen zu entwickeln. Die Library umfasst mittlerweile viele für moderne Kommunikationsverfahren notwendige Werkzeuge, von der einfachen Finite Impulse Response (FIR)-Filter-Implementierung bis hin zu einem einfachen OFDM-Verfahren (siehe [Gaeddert, 2014]). Die Bibliothek ist in C verfasst, um auch auf verschiedenen Mikroprozessoren arbeiten zu können, bei denen der größere Speicherbedarf von C++-Programmen problematisch werden kann. Im vorliegenden Projekt wurde diese Library für die Implementierung der Ein- und Mehrträgerverfahren im komplexen Basisband verwendet.

3.4.3 Red Pitaya Notes

Wenngleich keine Library im engeren Sinne, stellt der an der Universite catholique de Louvain tätige Forscher Pavel Demin unter dem Projektnamen *Red-Pitaya-Notes* eine Sammlung von IP-Cores für die Verwendung von Komponenten des *RedPitaya-Boards* quelloffen zur Verfügung.

Aus diesem Projekt wurde beispielsweise die Ansteuerung von ADC und DAC, der *RamWriter* genannte direkte schreibende Zugriff auf den RAM von der PL aus sowie Ansätze zur Verbesserung der Umsetzung der QAM im FPGA verwendet.

3.4.4 Der Qi-Standard

Die Energieübertragung des Qi-Standards basiert, wie auch die umgesetzte Datenübertragung auf induktiver Kopplung eines Magnetfeldes. Eine typische Übertragungsspule im Qi-Standard hat ca. 20 Windungen in maximal 2 Lagen und einen Weicheisenboden. Die Induktivität ist mit ca. $24\ \mu\text{H}$ ebenso wie der Einkopplungskondensator mit $200\ \text{nF}$ vorgegeben. Daraus ergibt sich eine Resonanzfrequenz von ca. $140\ \text{kHz}$.

Die reale Resonanzfrequenz von Sender- und Empfängerspule können hiervon abweichen, müssen aber zwingend im Bereich $100\ \text{kHz}$ - $200\ \text{kHz}$ liegen.

Die genaue Übertragungsfrequenz wird zwischen Sender und Empfänger ausgehandelt, nachdem ein erster Kontakt senderseitig durch Änderung der Last erkannt wurde.

Der Qi-Standard sieht hierzu neben der reinen Energieübertragung auch eine schmalbandige Datenübertragung vor. Kommunikation vom Sender zum Empfänger erfolgt hierbei mittels auf die Energieübertragungsfrequenz aufmodulierten BPSK-Signalen mit einer Baudrate von 2kBaud. Da der Empfänger keinen selbstbetriebenen Kommunikationskanal hat, antwortet dieser durch eine der eigentlichen Energieaufnahme überlagerten Lastmodulation, welche als Amplitudenmodulation im Sender erkannt werden kann.

Hierbei werden neben regelmäßigen Pings und frei verfügbaren Datenfeldern vor allem die

Übertragungsmetriken, also die aufgewendete und empfangene Energiemenge, übertragen. Entsteht hierbei eine erhebliche Diskrepanz, so wird die Übertragung eingestellt, da fremde Objekte im Übertragungsweg vermutet werden, die eine zusätzliche induktive Last darstellen.

4 Entwurf

Das erstellte Übertragungssystem enthält alle Komponenten von Übernahme der Daten, über den Nachrichtenversand im Unterwasserkanal, bis zur Weitergabe empfangener Daten vom Empfängermodem an den dort angeschlossenen Teilnehmer. Dabei wurden so viele Schritte wie möglich in einem SDR -System gelöst, so dass die Funktionsparameter vom Datenmodulationsverfahren, über die Codierung bis hin zur tatsächlichen Signalmischung in das Übertragungsband in Software parametrisiert und somit einfach angepasst werden kann. Diese Anpassungen müssen allerdings innerhalb der Funktionsgrenzen des Systems erfolgen.

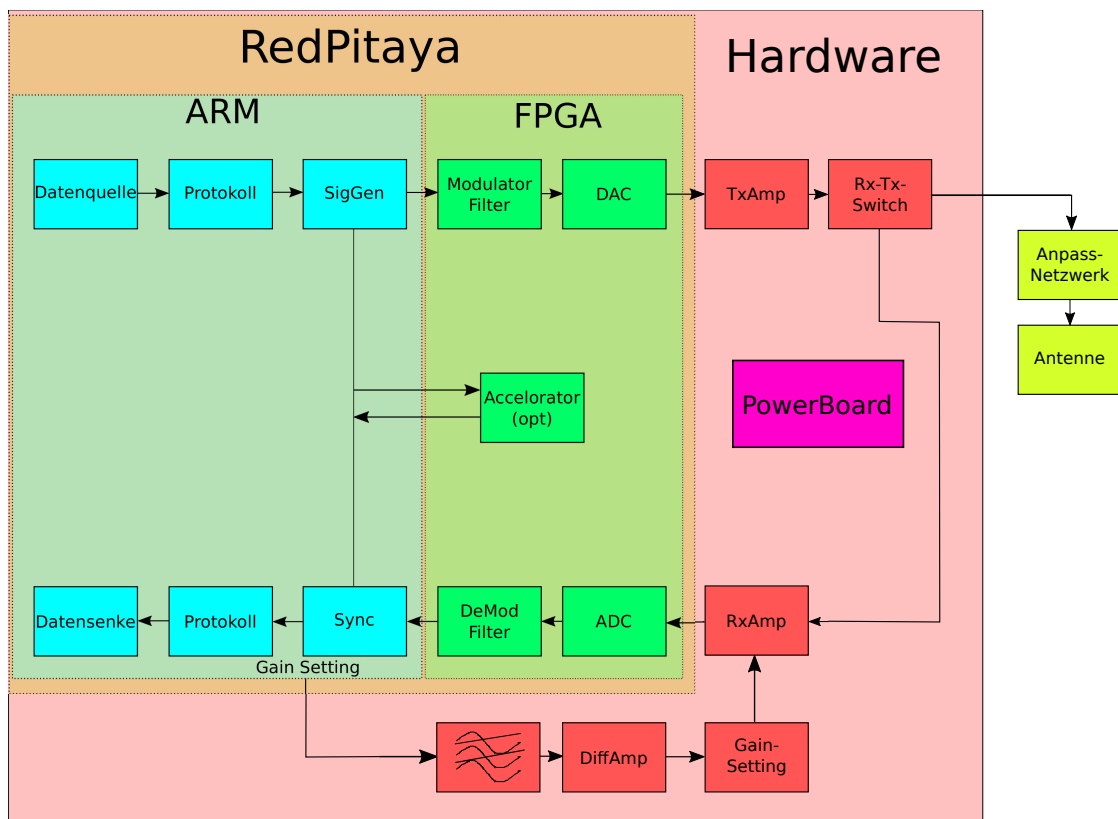


Abbildung 4.1: Schematische Darstellung des erstellten Übertragungssystems (Kommunikationsteil)

Im nachfolgenden Kapitel soll der Grundentwurf des Systems erläutert werden, indem die gewählten Verfahren beschrieben und die Übergabeschnittstellen der einzelnen Komponenten aufgezeigt werden, die ein Zusammenwirken des Gesamtsystems ermöglichen.

Das System besteht aus den fünf Hauptkomponenten *Antennensystem*, *Transmissionboard*, *FPGA-Logik*, *Powerboard*, sowie die *ARM-Software* auf Basis des installierten *Linux*-Systems.

4.1 Antennensystem

Beim Entwurf des Antennensystems ist zunächst die grundlegende Frage des gewünschten Antennendesigns zu klären. Im Abschnitt 3.1.2 wurde bereits gezeigt, dass mit zunehmendem Salzgehalt eine Übertragung im Fernfeld aufgrund der hohen Dämpfung und des dadurch sinkenden SNR immer aufwendiger wird. Eine Kompensation dieser Dämpfung durch bloße Steigerung der Sendeleistung erscheint dort ebenfalls nicht erfolgversprechend.

Unter Annahme einer Übermittlung auf der gewählten Kommunikationsfrequenz ($f = 4.6 \text{ MHz}$) mit einer Sendeleistung von 5 W und einer Übertragungsstrecke von $r = 1 \text{ m}$ bei einem Salzgehalt $\sigma = 4 \frac{\text{S}}{\text{m}}$, erreicht den Empfangsverstärker bei angepasster Antenne noch eine Leistung von:

$$P_{\text{out}} = 5 \text{ W} \approx 37 \text{ dBm} \quad (4.1)$$

$$\alpha = 0.0173 \sqrt{f \cdot \sigma} = 81 \frac{\text{dB}}{\text{m}} \quad (4.2)$$

$$P_{\text{in}} = P_{\text{out}} - \alpha \cdot 1 \text{ m} = -44 \text{ dBm} \approx 0.04 \text{ mW} \quad (4.3)$$

Da die Antenne gemäß Abschnitt 4.2.2 dazu mit einer Impedanz von 9Ω angepasst werden muss, liegt das Signal am Empfangsverstärker noch mit einer Amplitude von

$$\hat{u} = \sqrt{2 \cdot P_{\text{in}} \cdot R} = 26.8 \text{ mV} \quad (4.4)$$

an.

Um die volle Amplitude des ADC von $\pm 1 \text{ V}$ ausnutzen zu können, muss das Signal vom Eingangsverstärker daher hier bereits um $\approx 24.5 \text{ dB}$ verstärkt werden. Eine Verdopplung dieser Übertragungsstrecke ergibt für die benötigte Eingangsverstärkung bereits einen Wert von $\approx 98 \text{ dB}$. Es ist offensichtlich, dass die Dämpfung so stark ist, dass die Erhöhung der Sendeleistung zu keiner nennenswerten Steigerung der Übertragungsentfernung führt.

Da diese Ergebnisse gar unter Annahme eines verlustfreien Betriebes der Antennen erfolgten, müssen weitere Verluste möglichst vermieden werden. Antennen mit einer guten Richtcharakteristik können daher helfen, diese Verluste einzugrenzen. Gleichzeitig ist zumindest im Bereich der Energieübertragung ein guter Kopplungsfaktor für eine verlustarme Übertragung entscheidend.

Daher fiel die Entscheidung zugunsten von *Magnetic-Loop-Antennen*. Diese entsprechen im Fernfeldeinsatz einer Antenne mit schmaler Richt- und Übertragungscharakteristik. Im reaktiven Nahfeld dagegen verhalten sich Sender und Empfänger wie das Spulenpaar eines Luftspalttransformators.

Aufgrund der Schmalbandigkeit dieser Antennen werden diese, zur Erzeugung eines Datenkanals hoher Bandbreite auf leicht versetzte Resonanzfrequenzen angepasst. Die Übertragungsfunktion eines solchen Spulenpaars weist dann zwar eine Dämpfung im Bereich der Resonanzpunkte gegenüber einem ideal angepassten Paar auf, allerdings entsteht zwischen den beiden Resonanzfrequenzen ein Bereich geringer Amplitudenbetragsvarianz und linearem Phasengang.

4.1.1 Spulenentwurf

Der vorgegebene Druckkörper begrenzte den maximalen Spulendurchmesser auf 82 mm. Der freie Bauraum gemeinsam mit dem verbleibenden Ohmschen Widerstand der kompensierten Spule wiederum begrenzt die Gesamtlänge beider Spulen auf ca. 100 mm. Die Kommunikationsspule wurde daher auf 13 Wicklungen und die Leistungsspule auf 40 Wicklungen aus 1.76 mm^2 Kupferdraht festgelegt, was mit einer theoretischen Gesamtlänge von 79.5 mm noch genug Platz für suboptimale Wicklung, Isolation und Anschlüsse lässt.

Die entworfenen Spulen wurden anschließend in *FEKO* für den Nahfeldbereich simuliert, um die Stärke des erzeugten Feldes sowie dessen Verteilung zu prüfen. Die eingespeiste Leistung entspricht in der Simulation bereits dem angestrebten Einsatzfall von 5 W Leistung im Kommunikationsteil und 200 W im Leistungsbereich.

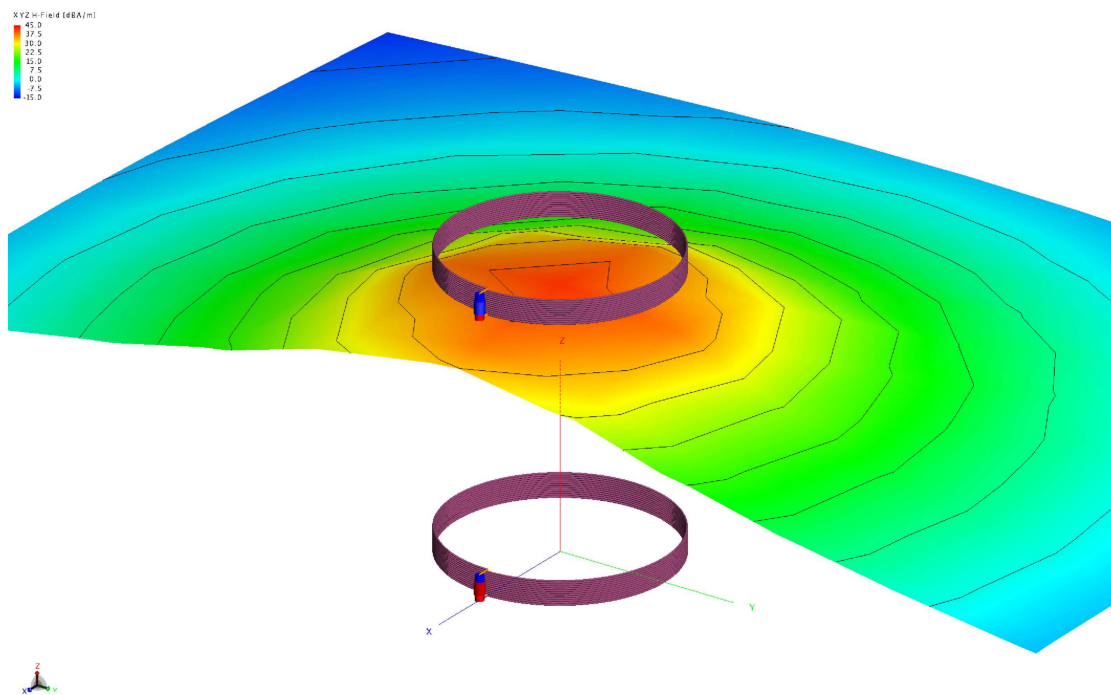


Abbildung 4.2: Felddarstellung der Kommunikationsspule

Erkennbar ist in der Simulation die deutliche Richtwirkung des erzeugten Feldes entlang der Spulenachse.

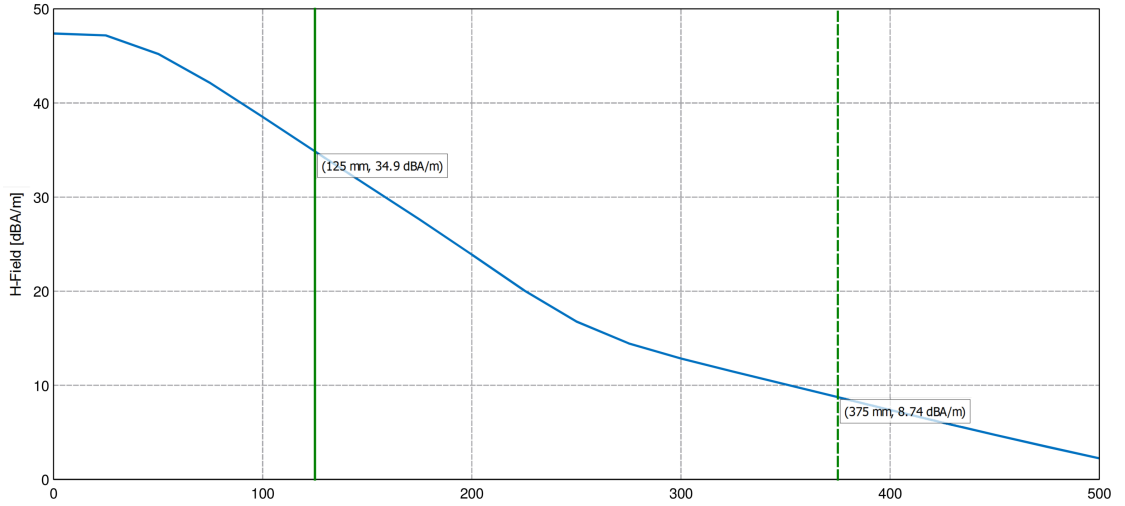


Abbildung 4.3: Feldstärke in Abhängigkeit der Entfernung auf der Spulenachse der Kommunikationsspule

Wie aufgrund der starken Dämpfung erwartet, nimmt die Feldstärke über die Entfernung schnell ab. Die erste Markierung stellt die einbaubedingte minimale Entfernung zwischen den Spulen im fertigen Druckkörper dar.

Die hier theoretisch maximal induzierbare Spannung kann nun mittels der Zusammenhänge aus 3.1.4 ermittelt werden. Dies wird hier einmal exemplarisch für den Punkt der höchsten Annäherung der Kommunikationsspulen bei einer aus den Simulationsergebnissen abgelesenen Verteilung der *magnetischen Flussdichte* von

$$\vec{H}(\vec{r}) = \vec{e}_z \cdot 34 \text{ dB} \frac{\text{A}}{\text{m}} - \vec{r} \cdot 300 \text{ dB} \frac{\text{A}}{\text{m}} \quad (4.5)$$

ermittelt und führt bei einem Spulenradius von 41 mm zu einer mittleren Flussdichte von

$$\vec{H} = 21.7 \cdot \vec{e}_z \text{ dB} \frac{\text{A}}{\text{m}}, \quad (4.6)$$

$$\vec{B} \approx \mu_0 \cdot 12.1 \cdot \vec{e}_z \frac{\text{A}}{\text{m}}. \quad (4.7)$$

Mit dieser Vereinfachung wird der *magnetische Fluss* zu

$$\Phi = \pi r^2 \cdot B = 8 \cdot 10^{-8} \text{ Wb} \quad (4.8)$$

und bei reiner Trägerwelle (4.6 MHz) zu

$$\Phi(t) = \sqrt{2} \cdot 8 \cdot 10^{-8} \sin(\omega t) \text{ Wb} \quad (4.9)$$

$$\dot{\Phi}(t) = \omega \cdot \sqrt{2} \cdot 8 \cdot 10^{-8} \cos(\omega t) \text{ Wb} = 3.3 \cos(\omega t) \text{ V} \quad (4.10)$$

Diese Spannung an einer unbelasteten Empfangsspule stellt den theoretischen Idealfall dar, der in der Praxis nicht erreicht wird. Eine geringere Entfernung ist in diesem Fall nicht wünschenswert, da die Kopplung zwischen den beiden Spulen sonst zu stark ansteigen würde und die volle Sendeamplitude auf der Empfangsspule induziert würde.

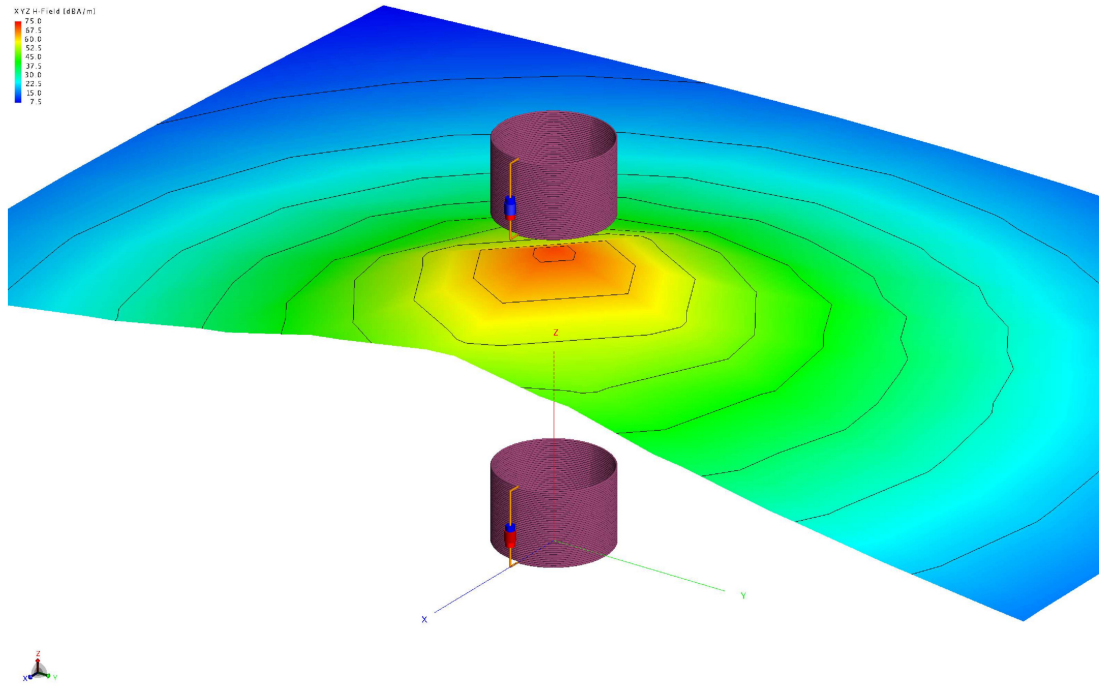


Abbildung 4.4: Felddarstellung der Leistungsspule

Das Feld der Leistungsspule hat aufgrund der größeren Spulenlänge eine stärkere Richtcharakteristik, was der gewünschten Anwendung entgegen kommt. Die Ausrichtung der Spule muss zwar genauer erfolgen, die eingekoppelte Energie ist dafür umso größer.

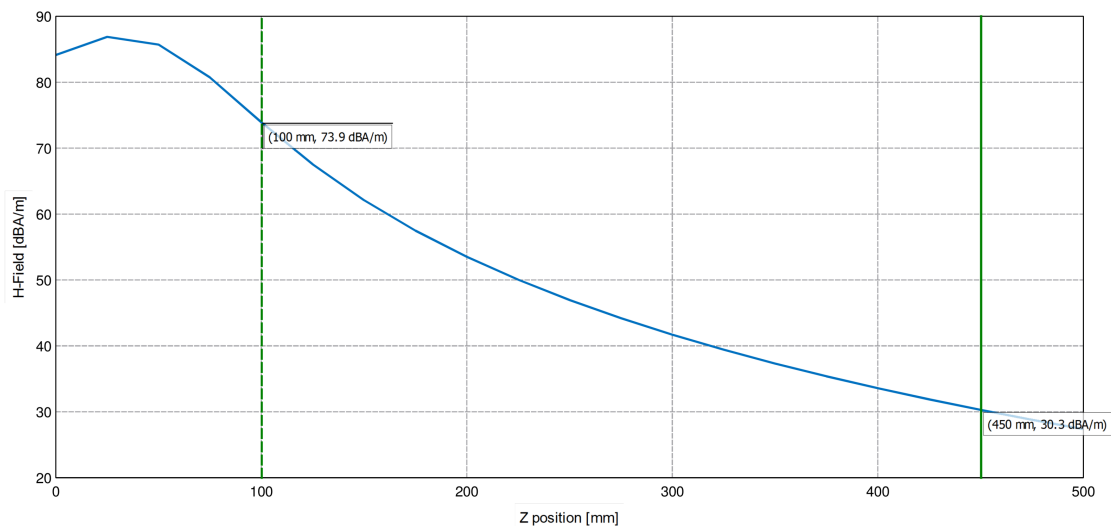


Abbildung 4.5: Feldstärke in Abhängigkeit der Entfernung auf der Spulenachse der Leistungsspule

Auch wenn die Feldstärke aufgrund der größeren Signalleistung deutlich größer ist, fällt sie nach Verlassen des spulenumschlossenen Bereiches zügig ab. Aufgrund der wegen der größeren Wellenlänge ausgeprägteren Richtcharakteristik geschieht dies allerdings langsamer.

4.2 Hardware

Auch wenn die Signalerzeugung des Systems komplett in Software erfolgt, kann auf ein analoges Frontend nicht verzichtet werden. Ein- und ausgehende Signale müssen verstärkt und gegeneinander isoliert werden. Diese elementar wichtigen Aufgaben sind allerdings keineswegs trivial und erfordern einige Designentscheidungen, die die komplette Arbeitsweise des Modems bestimmen werden. In diesem Abschnitt soll auf diese Probleme und die hier gewählten Lösungsansätze eingegangen werden. Dabei beschränkt sich dieser Abschnitt zunächst auf den Kommunikationsteil des Modems. Der Leistungsteil folgt unter Abschnitt 4.2.5

4.2.1 Sendeverstärker

Eine Verstärkung des Ausgangssignals des *RedPitaya-Boards* ist nötig, da die Ausgangsstufen der internen Ausgangsverstärker gemäß [RPHD, 2017] nur ca. 9 dBm an Ausgangsleistung liefern können.

Wenngleich die Auswahl des konkreten Bauteiles normalerweise nicht das ausschlaggebende Entscheidungskriterium im Entwurf des Konzeptes sein sollte, so diktiert der Markt hier das Design. Nur wenn auch verfügbare Komponenten gefunden werden können, ist eine praktische Umsetzung möglich. Schließlich beeinflusst die Wahl des Sendeverstärkers maßgeblich die Anzahl der notwendigen Verstärkerstufen, die benötigten Systemspannungen, die gewählte Impedanz der Sendeantenne sowie einige Anforderungen an die Sende- Empfangsumschaltung.

An den Sendeverstärker des Modems wurden folgende Anforderungen gestellt:

- Sendeleistung: $\geq 5 \text{ W} = 37 \text{ dBm}$
- zulässige Signalfrequenz: $\geq 5 \text{ MHz}$
- Preis: $\leq 15 \text{ €}$ je Stück
- geringer Implementierungsaufwand
- Möglichkeit des Abschaltens bei Nichtverwendung
- thermische Robustheit, da kaum externe Kühlung möglich ist. Wasser steht zwar in der Umgebung des Fahrzeuges zur Verfügung, ein Kühlsystem ist aber bei kleineren Druckkörpern unüblich, weil fehleranfällig.

Neben der Entwicklung einer eigenen Verstärkerstufe, die aber den zeitlichen Rahmen der Arbeit gesprengt hätte, konnte lediglich der Digital Subscriber Line (DLS)-Leitungstreiber *LT1210* alle Bedingungen erfüllen.

Ausgehend von dieser Wahl konnten weitere Designentscheidungen getroffen werden:

Ausgangsimpedanz

Die Impedanz der Antenne ist ein wesentliches Designmerkmal dieser und sollte, um die optimale Ausbeute zu erzielen, auf den Signalgenerator abgestimmt sein. Der gewählte Sendeverstärker *LT1210* kann laut [DS1210, 1996] einen maximalen Strom von 1.1 A liefern.

Während die maximale Ausgangsamplitude bei $\pm 15\text{ V}$ Versorgungsspannung erzielt werden kann, so wird hier aufgrund der größeren Verfügbarkeit und des günstigeren Preises leistungsstärkerer 12 V Spannungswandler mit dieser reduzierten Versorgungsspannung gearbeitet (siehe Abschnitt 5.1). Mit dieser Versorgungsspannung kann eine Ausgangsamplitude von 9.2 V ausgegeben werden. Bei einem Spitzenstrom von 1.1 A führt dies zu einer Ausgangsimpedanz von 8.4Ω .

Die Impedanz des Systems wird damit auf 9Ω festgelegt, um auch im Falle von Toleranzen nicht in die Strombegrenzung zu kommen und durch das dann auftretende Clipping nichtlineare Verzerrungen im Verstärker zu erzeugen.

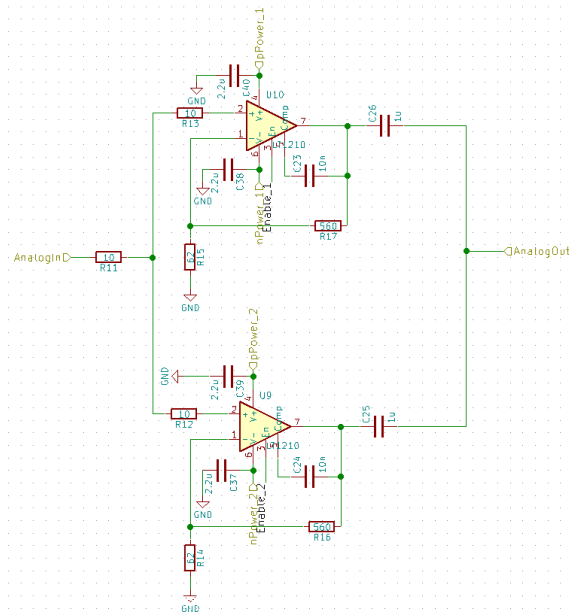
Ausgangsleistung

Bei maximaler Aussteuerung führt die ermittelte Ausgangsimpedanz zu folgender Sinusleistung:

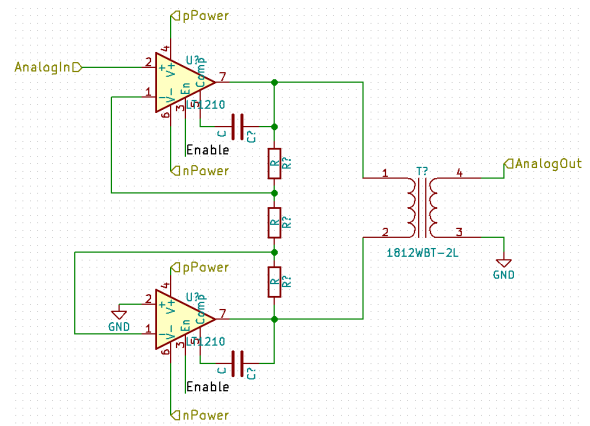
$$u_{\text{eff}} = \frac{\hat{u}}{\sqrt{2}} = 6.5\text{ V} \quad (4.11)$$

$$P = \frac{u_{\text{eff}}^2}{R} = 4.7\text{ W} \quad (4.12)$$

Die geforderte Ausgangsleistung wird also bereits mit einem Verstärker annähernd erreicht. Möglichkeiten der Steigerung der Leistung sind ebenfalls vorhanden:



(a) Parallelisierung der Verstärker



(b) Komplementärschaltung der Verstärker

Abbildung 4.6: Verstärkerschaltungen zur Steigerung der Ausgangsleistung

Soll die Leistung verdoppelt werden, so ist die erste Möglichkeit eine Parallelisierung von Ausgangsverstärkern, wobei auf eine möglichst geringe Phasendifferenz der Ausgangssignale geachtet werden muss, um die Blindleistung im System gering zu halten. Ferner muss aufgrund möglicher unterschiedlicher Offsetspannung eine AC-Kopplung zwischen den beiden Ausgangsverstärkern vorgesehen werden.

Die zweite Möglichkeit zur Steigerung der Ausgangsleistung ist die Komplementärschaltung zweier Ausgangsverstärker. Dabei werden diese durch invertierende und nichtinvertierende Ansteuerung in genau 180° Phasenlage bei gleicher Amplitude gebracht. Dies sorgt an den Ausgängen für eine verdoppelte Spannungsdifferenz. Der Nachteil dieser Methode ist, dass das Ausgangssignal danach keinen Massebezug mehr hat und, so dieser wie in diesem Fall benötigt wird, mittels Übertrager hergestellt werden muss. Übertrager haben allerdings grundsätzlich keine neutrale Übertragungsfunktion, verändern also den Übertragungskanal zusätzlich und arbeiten nicht verlustfrei. Dies gilt vor allem bei einer hohen Anforderung an die Signaldynamik.

Für das System wurde letztlich eine Parallelisierung der Verstärker vorbereitet.

4.2.2 Empfangsverstärker

Aufgrund der in den Abschnitten 3.1 sowie 4.1 angeführten starken Zunahme der Dämpfung mit dem Sender-Empfänger-Abstand entsteht im Empfangsfall ein Problem. Die Datenübertragung soll auch in Salzwasser nicht nur in direktem Kontakt, sondern auch in (geringem) Abstand zwischen Sender und Empfänger ermöglicht werden. Auch wenn, wie unter Abschnitt 3.2 erläutert, zunächst von einem LTI-System ausgegangen wird, so müssen doch zumindest verschiedene Einbaupositionen und Fahrzeugmodelle im späteren Einsatz berücksichtigt werden.

Eine Möglichkeit, dieses Problem zumindest etwas zu verringern, ist der Einsatz eines Programmable Gain Amplifier (PGA) oder Variable Gain Amplifier (VGA). Diese Verstärker ermöglichen die Veränderung der Verstärkung mittels Anlegen einer definierten Spannung (VGA) bzw. durch ein digitales Interface (PGA). Der DAC des *RedPitaya-Board* verfügt, wie in Abschnitt 3.3.2 zwar über zwei Kanäle, allerdings soll dieser für künftige Entwicklungen freigehalten werden. Aus diesem Grund war ein PGA die erste Wahl. Da ein solcher mit ausreichend hohem Verstärkungsbereich aber nicht zu vertretbaren Preisen erhältlich war, wurde die Suche um VGA erweitert und anschließend der *AD8337* aufgrund seines großen Verstärkungsbereichs ausgewählt. Die Verstärkung kann hier in einem Bereich von 24 dB variiert werden. Zusätzlich dazu ist in diesem Integrated Circuit (IC) eine zweite, über Widerstände konfigurierbare Verstärkerstufe integriert, was unter Berücksichtigung der Einschränkungen des Verstärkertyps (*Current-Feedback*-Verstärker wie dieser verlieren bei Einstellung zu großer Feedbackwiderstände stark an Bandbreite) und gängiger Widerstandswerte die Verstärkungsbereiche 18...42 dB, bzw. 21...45 dB erlaubt.

Die Einstellung der Verstärkung in diesen Grenzen erfolgt durch das Anlegen eines analogen Signals im Bereich $-0.7\text{ V} \dots +0.7\text{ V}$, wobei Spannungen bis zur Größe der Versorgungsspannung ausdrücklich zulässig sind.

Zum Erreichen dieser Steuerspannung wurde mittels eines sekundären Verstärkers ein Differenzverstärker mit einer Verstärkung von 0.5 entworfen. Dieser wird mit zwei Spannungen des 1.8 V-Extension Headers des *RedPitaya-Boards* angesprochen. Diese werden intern von einem entwickelten durch das PS konfigurierbaren Pulse Width Modulation (PWM)-Modul gesteuert (siehe Abschnitt 4.3.1). Der Referenz Ausgang sollte dabei voll ausgesteuert, der Kontrollausgang mit einem modulierten Signal belegt werden. Zwischen PWM-Pin und Ein-

gang des Differenzenverstärkers wurde zur Glättung des Signals ein RC-Tiefpass vorgesehen, da Signale auf dem Verstärkungseingang auf das verstärkte Signal moduliert werden.

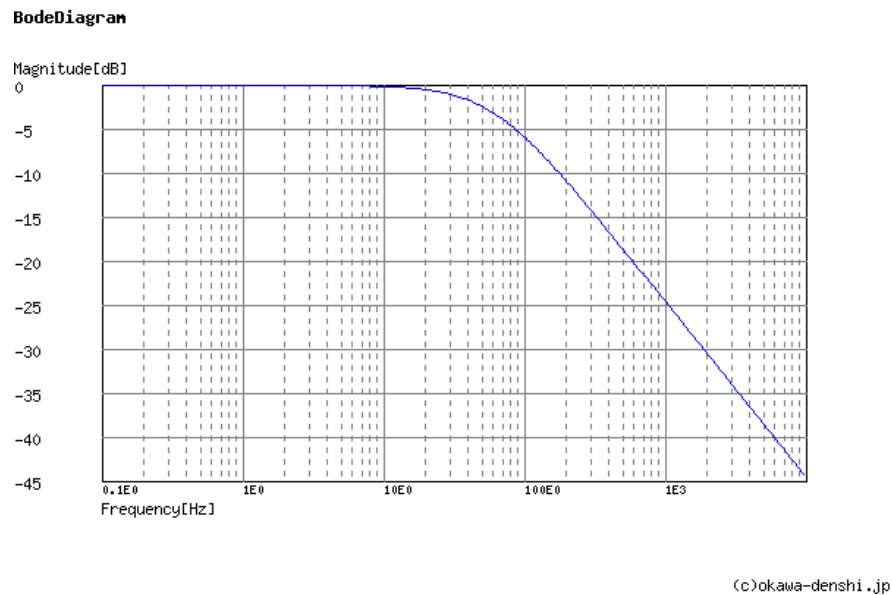


Abbildung 4.7: berechnetes Bode-Diagramm des PWM-Filters der Empfangsverstärkungseinstellung

Da die relative Bewegung von autonomen Wasserfahrzeugen, insbesondere in der Nähe anderer Fahrzeuge oder Ladestationen stets vergleichsweise langsam ist, sich die Dämpfung also selbst in einem ungünstigen Fall nicht ruckartig ändert, wurde dieser Tiefpass für eine Grenzfrequenz von 50 Hz ausgelegt.

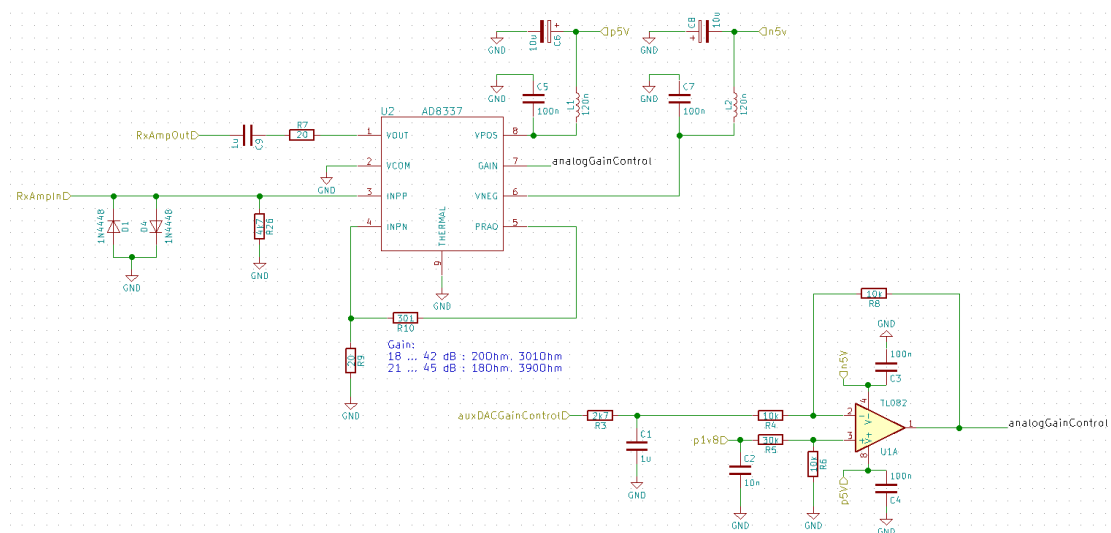


Abbildung 4.8: Einstellbarer Empfangsverstärker samt Differenzverstärker mit Eingangstiefpass

4.2.3 Sende- und Empfangsumschaltung

Die vergleichsweise hohe Sendeleistung bei der gleichzeitig großen Empfindlichkeit der Eingangsverstärker macht eine Trennung zwischen Sende- und Empfangssignal unumgänglich.

Einerseits ist eine große Sendeleistung gewünscht, um die Ausbreitungsverluste auszugleichen, andererseits ist eine möglichst empfindliche Eingangsstufe angestrebt, um auch die sehr stark gedämpften Signale noch detektieren zu können. Diese Kombination kann zu unerwünschten Effekten von der Übersteuerung der Eingangsstufe, bis zu deren Zerstörung führen. Andererseits haben Leistungsverstärker meist eine sehr geringe Ausgangsimpedanz, die parallel zum Empfangsverstärker die Impedanz im Empfangsfall verändert. Daher muss einerseits das Sendesignal für den Empfangsverstärker soweit gedämpft werden, dass dieser keinen Schaden nehmen, oder das Signal stören kann und andererseits der Sendeverstärker im Empfangsfall möglichst hochohmig werden, um das Empfangssignal nicht weiter zu dämpfen. Zur Trennung dieser Signale werden normalerweise je nach Einsatzzweck und Anforderungen meist verschiedene Lösungen parallel verfolgt. Grundsätzlich werden die unter Abschnitt 3.2.2 angesprochenen Verfahren des Zeit- oder Frequenzmultiplexings verwendet. Auf die Anwendbarkeit der beiden Verfahren auf das hier entwickelte System soll anschließend eingegangen werden.

Frequenzmultiplexing

Zur Trennung der beiden Signalwege besteht die erste Möglichkeit der gegenseitigen Isolation im Frequenzbereich. Im einfachsten Falle weist das Kommunikationssystem getrennte Kanäle für den Sende- und Empfangskanal auf, die sich mittels Bandpassfilter trennen lassen. Dazu wird zumindest im Empfangszweig ein Bandpass hinzugefügt, der die starken Signale des Sendeverstärkers dämpft. Eingesetzt wird das Verfahren, beispielsweise bei Mobilfunkverfahren wie GSM und LTE, da hier die Mobilteilnehmer niemals untereinander, sondern stets mit einer festen Gegenstelle kommunizieren. Im entwickelten System birgt diese Lösung allerdings einige Probleme. Während in klassischen Mobilfunknetzen eine klare Abgrenzung zwischen den zur Netzinfrastruktur gehörenden Basisstationen und den Mobilfunkclients getroffen werden kann, ist dies hier nicht der Fall. Ein Kommunikationsteilnehmer, wie beispielsweise ein AUV muss in der Lage sein, sowohl mit Basisstationen, als auch mit anderen Fahrzeugen zu kommunizieren. Dies ist bei Verwendung dieser Multiplexertechnik nur möglich, wenn entweder jedes Modem Sende- und Empfangsfrequenzen fest vorgegeben bekommt und nie ein zweites Modem mit gleicher Frequenzaufteilung im gleichen Einsatzgebiet verwendet wird, oder wenn der Bandpassfilter per Software anpassbar ist.

Dies kann durch Switched Capacitor (SC)-Filter realisiert werden. Bei diesem Filtertyp werden die für den zu realisierenden Bandpass verwendeten Operationsverstärker nicht mit Widerständen, sondern mit Kondensatoren eingestellt. Die Grenzfrequenzen können dann in schmalen Grenzen mittels Frequenz angelegter Taktsignale eingestellt werden.

Da der Einsatz von Frequenzmultiplexing aber einen Teil der ohnehin schon geringen Bandbreite für den Rückkanal belegt, wird derzeit auf den Einsatz dieser Technik verzichtet.

Zeitmultiplexing

Die naheliegendste Möglichkeit der Trennung zwischen Sende- und Empfängersignal ist die zeitbasierte Umschaltung. Durch Aus- und Einkopplung des Sende- bzw. Empfangszweiges wird die notwendige Dämpfung erreicht. Zu beachten bei der Wahl des Schalters sind Fre-

quenzbereich, Umschaltzeit, Trennung zwischen Steuer- und Signalkreis, schaltbare Leistung sowie die resistive und kapazitive Kopplung zwischen Antenne und abgeschaltetem Zweig.

Anforderungen

Um die beste Methode der Sende-Empfangsumschaltung auswählen zu können, ist ein Blick auf die Anforderungen des erstellten Systems wichtig, um die verfügbaren Methoden daran messen zu können.

Dämpfung

Der Eingangszweig muss so stark vom Sendezweig entkoppelt werden, dass die Eingangsstufen des ADC und damit auch die der Eingangsverstärker im Sendefall nicht überlastet werden. Das Ausgangssignal des Sendeverstärkers hat, wie unter Punkt 4.2.1 dargestellt, eine Spitzenspannung von 9.2 V an einer Impedanz von 9 Ω . Da am Eingang des ADC die zulässige Höchstspannung von 1 V nicht überschritten werden sollte, lässt sich die erforderliche Dämpfung errechnen. Der Eingangsverstärker hat, wie unter Punkt 4.2.2 dargestellt eine maximale Eingangsverstärkung von 45 dB.

Die erforderliche Dämpfung der Sende-Empfangsumschaltung für den Optimalfall ist also:

$$D_{\text{TxRx}} = D_{\text{Tx}} + D_{\text{Rx}} = 19.3 \text{ dB} + 45 \text{ dB} = 64.3 \text{ dB} \quad (4.13)$$

Nutzt man die Belastbarkeit des Eingangs des gewählten Empfangsverstärkers (bis zu angelegter Versorgungsspannung gemäß [AD8337, 2017]) aus, so ist auch eine Dämpfung von 6 dB noch vertretbar, aber mit nichtlinearen Verzerrungen am ADC des *RedPitaya-Boards* verbunden.

Eingangsimpedanz

Ferner darf der Sendeverstärker mit den geringen Ausgangsimpedanzen den Eingangswiderstand nicht so sehr senken, dass ein zu großer Teil der Eingangsleistung daran abfällt. Im Empfangsmodus sind Eingangsverstärker sowie Sendeverstärker mit dazugehörigen Schaltern sowie die Antenne parallel geschaltet. Da die Impedanz des Eingangsverstärkers gegenüber den beiden anderen Zweigen sehr groß ist, kann dieser Zweig vernachlässigt werden. Es verbleiben also Sendeverstärker samt Schalter parallel zur Sende-/Empfangsantenne. Die Impedanz des Schalters für den Sendeverstärker muss im ausgeschalteten Fall so groß sein, dass die Empfangsantenne nicht durch diesen belastet wird. Die zulässige Reduktion des Gesamtimpedanz wurde auf maximal 1 % der reinen Antennenimpedanz festgelegt, um das Eingangssignal nicht unnötig zu dämpfen. Ausgehend von einer Antennenimpedanz von 9 Ω bestimmt sich die zulässige Sendeverstärkerimpedanz zu:

$$Z_{\text{ges}} \geq \frac{Z_{\text{Ant}} \cdot Z_{\text{Amp}}}{Z_{\text{Ant}} + Z_{\text{Amp}}} \approx 0.99 \cdot Z_{\text{Ant}} \quad (4.14)$$

$$\frac{1}{Z_{\text{Amp}}} = \frac{1}{0.99 \cdot Z_{\text{Ant}}} - \frac{1}{Z_{\text{Ant}}} \quad (4.15)$$

$$Z_{\text{Amp}} \geq 891 \Omega \quad (4.16)$$

Die Sendeverstärker LT1210 haben im gewählten Frequenzbereich im ausgeschalteten Fall laut [DS1210, 1996] noch eine Impedanz von $100\,\Omega$.

Umschaltgeschwindigkeit

Die letzte wichtige Eigenschaft stellt die Umschaltgeschwindigkeit dar. Da während des Umschaltens weder gesendet, noch empfangen werden kann, sollten diese Totzeiten möglichst klein gehalten werden, um die Nutzung des Kanals zu maximieren. Grundsätzlich müssen die Schaltzeiten daher wesentlich kleiner als die Länge der Sendepakete sein. Bei einer Paketlänge von 20 ms und einer angenommenen maximalen Umschaltzeit von 1 ms ergibt sich ein maximales Sende- zu Umschaltzeitverhältnis von

$$\eta = \frac{T_{\text{Sym}}}{T_{\text{Sym}} + 2 \cdot T_{\text{Switch}}} = \frac{20}{22} = 90.9\% \quad (4.17)$$

inklusive Ein- und Ausschaltzeit.

Lösungen

Nachdem die Anforderungen geklärt sind, werden an dieser Stelle die gängigen Lösungen für das Problem präsentiert und gegenübergestellt. Anschließend wird der gewählte Ansatz genauer vorgestellt.

RF-Switch

RF-Switches sind ICs, die als Umschalter für Signale im Radio Frequency (RF)-Bereich entwickelt und hergestellt werden. Diese Umschalter bieten in der Regel direkt eine Single Pole Double Throw (SPDT)-Konfiguration und benötigen daher keine zweiten Schalter und abgesehen von Blockkondensatoren meist auch nur wenig zusätzliche Beschaltung. Eingesetzt werden diese Schalter vor allem im Mobil- und Datenfunk, da es in diesen Bereichen ebenfalls auf eine hohe Sperrdämpfung ankommt und die erforderliche Sendeleistung in den Mobilteilen meist weniger als 30 dBm beträgt.

Da die meisten relevanten Datenfunkbänder allerdings bei ca 700 – 850 MHz im Mobilfunk, oder gar erst bei 2.4 GHz bei Bluetooth/WLAN beginnen, gibt es nur wenige ICs, die bereits im Frequenzbereich von 5.5 MHz arbeiten.

Die Schaltzeiten dieser Switches liegen im Bereich 10^{-8} s. Beispielhaft wurde zum Vergleich hier der RF-Switch *HMC784* ausgewählt. Mit einer schaltbaren Leistung von 40 dBm, einer Schaltzeit von 50 ns sowie einer Dämpfung von > 60 dB erfüllt dieser formal die Bedingungen. Allerdings ist die angegebene schaltbare Leistung trotz DC-Fähigkeiten erst bei > 100 MHz gültig (siehe [HMC784, 2017]). Es konnten im Zuge dieser Arbeit keine RF-Switches gefunden werden, die die Leistungsanforderungen erfüllten und zu vertretbaren Preisen ($\leq 10\,\text{€}$) lieferbar waren.

Solid-State-Relay/Solid-State-Switch

Solid State Relays werden im niederfrequenten Bereich immer öfter eingesetzt, um mechanische Relais zu ersetzen, da sie deutlich langsamer verschleifen und günstiger zu produzieren sind. Die Anforderung an die schaltbare Leistung wird erfüllt, aber aufgrund der großen in-

ternen Kapazitäten werden weder die Anforderung an die Umschaltzeit noch an Dämpfung oder Eingangsimpedanz erfüllt.

Schaltdioden

Schaltdioden werden in Funkanlagen häufig eingesetzt, um Eingangsverstärker zu schützen. Dazu werden zwei Dioden seriell gegen Masse geschaltet, ein Abgriffpunkt zwischen den beiden Dioden D1 und D2 sowie Anode der ersten Diode bieten die beiden Abgriffe (an C1 bzw. C2) des Systems. Die Schaltung muss an beiden Eingängen entkoppelt und mit zwei Spannungsquellen (V2 und V3) versehen werden. Wird nun eine Bypassspannung an den Dioden dermaßen angelegt, dass die seriell zum Signal liegende Diode (D1) leitet und die gegen Masse liegende Diode (D2) sperrt, so kann ein hochfrequentes Signal bis zu einer Amplitude der Hälfte der zwischen V1 und V2 bestehenden Spannungsdifferenz abzüglich Vorwärtsspannung der Diode durchgelassen werden. Wird die Polarität der angelegten Spannung invertiert, so sperrt die serielle Diode (D1) und die gegen Masse liegende Diode (D2) wird durchlässig, der Eingang wird entkoppelt.

Diese Schaltung wird vor allem verwendet, wenn der Sendeverstärker im Empfangsfall eine hohe Impedanz aufweist und nur der Empfangsverstärker entkoppelt werden muss. Dies ist darin begründet, dass Dioden mit einem hohen Durchlassstrom meist eine relativ große Sperrkapazität aufweisen, was die Dämpfung des Signals einerseits und den Verlust andererseits erhöht.

Während also Umschaltzeiten abhängig von der Entkopplung der Steuerspannungen und des Signals sehr kurz sind, ist die Wahl von Dioden, die sowohl einen Strom von 1.1 A aushalten, als auch eine geringe Kapazität aufweisen sehr schwierig.

Zudem kommt eine bloße Entkopplung des Empfangsverstärkers leider aufgrund der geringen Impedanz des Sendeverstärkers im Ausschaltfall nicht in Frage.

MEMS / Reed-Relais

Ein Reed-Relais, oder Reed-Schalter, besteht aus einer in einem Glaskolben eingeschmolzenen ferromagnetischen Kontaktzunge. Mittels Anlegen einer äußeren Spannung an die den Kolben umgebende Spule wird im Inneren des Kolbens ein Magnetfeld aufgebaut, wodurch die Zunge bewegt und der Kontakt geschlossen wird. Diese Konstruktion ermöglicht einerseits eine gute galvanische Trennung zwischen Schalt- und Arbeitskreis und andererseits für ein mecha-

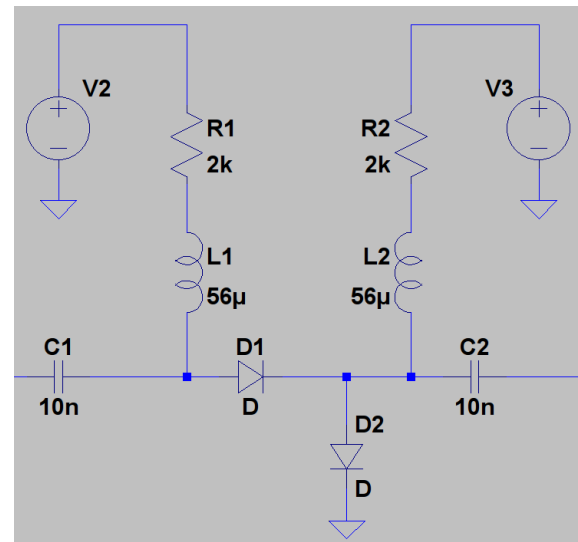


Abbildung 4.9: Prinzipaufbau Send- und Empfangsumschaltung mit Schaltdioden

nisches Relais vergleichsweise kurze Umschaltzeiten im Bereich weniger Millisekunden, was trotzdem die schlechteste Umschaltzeit im Vergleich darstellt. Konventionelle Relais haben in der Regel wegen der größeren bewegten Massen eine wesentlich größere Umschaltzeit und fallen daher wegen Verletzung der Anforderung an die Schaltzeiten aus dieser Betrachtung. Während die Schaltzeiten vergleichsweise groß ausfallen, sind bei dieser Art von Schalter die Isolation des Steuerzweiges sowie die Dämpfung im ausgeschalteten Zustand sehr groß, da hier eine physische Trennung der beiden Kontakte vorgenommen wird.

Der zulässige Strom verhält sich bei diesen Schaltern meist indirekt proportional zur Baugröße und damit auch der Schaltzeit, da mit der Stärke der Schaltzunge auch deren Trägheit zunimmt. Die angegebenen Schaltzeiten gelten allerdings bereits für den hier benötigten Leistungsbereich.

Auswahl

Aufgrund der ausschließenden Faktoren bei allen anderen Möglichkeiten fiel die Wahl schließlich auf ein Reed-Relais. Das ausgewählte Relais *Meder SIL 1A72-71* hat eine Schaltzeit von ca. 1 ms bei einer Kapazität von 1 pF im geöffneten Zustand, was bei 5.5 MHz einem Blindwiderstand von ca. 29 kΩ entspricht.

Diese Kapazität ist hier auch entscheidend für die Dämpfung zwischen Sendebereich und Empfangsbereich, denn aus dem Widerstandsteiler zwischen Reed-Relais und Eingangswiderstand des Empfangsverstärkers folgt der maximale Eingangswiderstand, um die Anforderung von −64.3 dB Dämpfung einzuhalten:

$$Z_{Rx} = \frac{(Z_{Sw} + Z_{Sw}) \cdot U_{Rx}}{U_{Tx}} = \frac{29 \text{ k}\Omega \cdot 10^{\frac{-64.3}{20}} \text{ V}}{9.2 \text{ V}} = 1.9 \Omega \quad (4.18)$$

Folglich muss für eine Einhaltung dieses Kriteriums am Eingangsverstärker ein zweiter, komplementär angesteuerter, Schalter gegen Masse vorgesehen werden. In dieser Arbeit wurde darauf verzichtet, da, wie erläutert, dadurch kein Schaden entsteht, sondern lediglich das selbst gesendete Signal verzerrt empfangen wird.

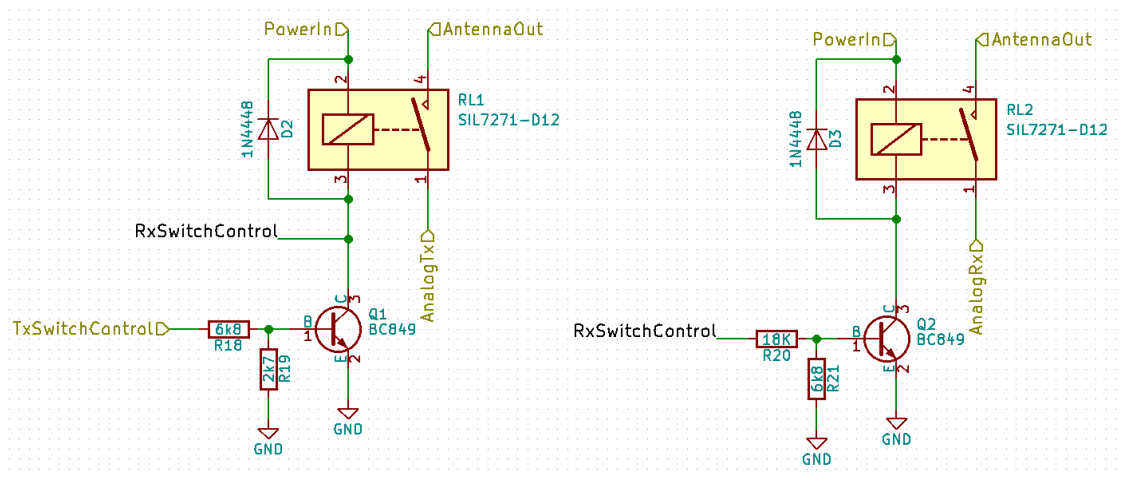


Abbildung 4.10: Sende-Empfangs-Umschalter mit Steuertransistoren

4.2.4 Energieversorgung

Die für die vorgestellten Schaltungen benötigten Spannungen von $\pm 12\text{ V}$ für den Sendeverstärker, $\pm 5\text{ V}$ für den Empfangsverstärker, dessen Differenzenverstärker zur Einstellung des Verstärkungsfaktors sowie die benötigte Versorgungsspannung des *RedPitaya-Boards* ($+5\text{ V}$) können aufgrund des angestrebten Einsatzszenarios an AUVs nicht extern bereitgestellt, sondern müssen im System erzeugt werden.

Zu diesem Zweck werden hier Schaltregler verwendet, da die Fahrzeuge in der Regel Gleichspannungen zwischen 18 V und 24 V bereitstellen.

Obwohl der Sendeverstärker bis zu 1.1 A Spitzenstrom benötigt, wurde eine Ausgangsleistung von 12 W jeweils für die $\pm 12\text{ V}$ -Versorgung als ausreichend erachtet, sofern diese von Pufferkondensatoren gestützt werden kann.

Für die $\pm 5\text{ V}$ -Versorgung wurde ein maximaler Versorgungsstrom von 1 A bei 5 V und 0.5 A bei -5 V überschlägig berechnet. Aufgrund dieser Anforderungen wurden Schaltregler der *OKI-78SR*-Serie von *muRata PS* gewählt. Für spätere Erweiterungen wurde noch ein Versorgungsweig mit $\pm 2.5\text{ V}$ vorgesehen, der allerdings hier noch nicht benötigt wird.

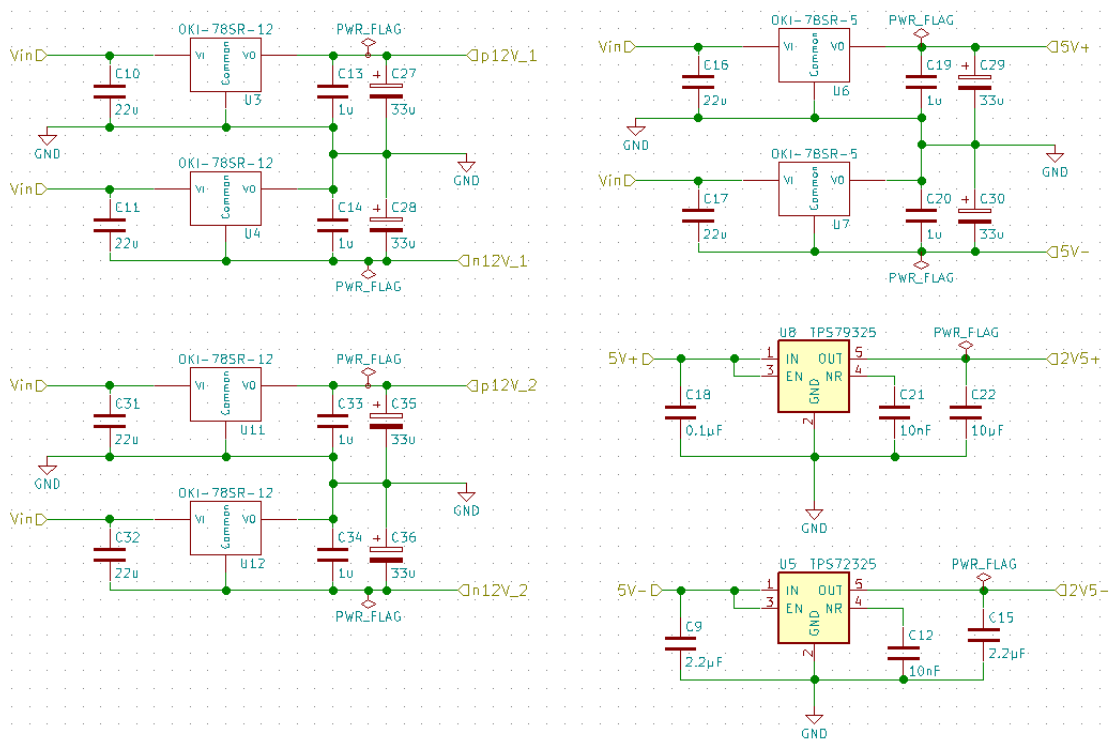


Abbildung 4.11: Spannungsregulierung durch Schaltregler

Mit diesen Schaltreglern, jeweils in Beschaltung für positive und negative Ausgangsspannung besteht ein zulässiger Eingangsspannungsbereich von $15 - 36\text{ V}$, der momentan durch einen kleineren Spannungsbereich der Abblockkondensatoren auf $15 - 25\text{ V}$ begrenzt wird, die Anforderungen aber noch immer erfüllt. Da eine Parallelisierung von Schaltreglern aufgrund möglicher Instabilitäten im Schwingungsverhalten problematisch ist, wurde für den optionalen zweiten Sendeverstärker ein zweiter Versorgungsstrang geschaffen.

4.2.5 Leistungsübertragung

Für die Übertragung der Ladeenergie war eine Entscheidung zwischen einer Eigenentwicklung und einigen Industriestandards zu treffen. Derzeit haben sich am Markt zwei Standards zum drahtlosen Laden etablieren können *Qi* und *Powermat*. Während für den *Qi*-Standard bereits der Bereich *Medium Power* (bis 200 W) mehrfach angekündigt wurde, gibt es bis zum Zeitpunkt der Arbeit weder einen veröffentlichten Standard, noch ICs dafür.

Eigene Versuche wurden nach mehreren erfolglosen Anläufen aus Zeitmangel abgebrochen. Das Partnerprojekt an der Technischen Fakultät der Christian Albrecht Universität zu Kiel erzielte allerdings wenige Tage vor Abschluss der Arbeit Erfolge bei der Übertragung von 200 W im 77 kHz-Band.

4.3 FPGA-Logik

Da das entwickelte System als SDR ausgeführt werden sollte, muss die komplette Signalerzeugung in Software erfolgen. Dies beinhaltet insbesondere auch die Modulation und die Ansteuerung der Hardwareein- und ausgänge. Der FPGA bietet sich aufgrund der Parallelisierbarkeit der Hardware und der hohen Arbeitsgeschwindigkeit als Plattform für alle hardwarenahen Aufgaben an. Dies umfasst die Ansteuerung des Sende-Empfangs-Umschalters, die Erzeugung des Hilfssignals zur Einstellung der Empfangsverstärkung sowie die Implementierung des IQ-Modulators.

4.3.1 QAM

Die Übermittlung komplexer Signale, wie sie bei der Verwendung hochstufiger Modulationsverfahren entstehen, über einen reellen Kanal, wie den hier gewählten Nahfeldkanal erfordert einen Mischer, der die imaginären und reellen Bestandteile in einem Signal übermitteln und dabei ihre Orthogonalität beibehalten kann. Dies gelingt mittels Quadraturamplitudenmodulation. Hierbei werden die reellen und komplexen Anteile des Basisbandsignals auf einen Träger mit gleicher Frequenz, aber mit 90° Phasenverschiebung moduliert, das Summensignal übertragen und im Empfänger die Signalkomponenten durch Demodulation wieder getrennt. In diesem Abschnitt wird sowohl Modulator als auch Demodulator kurz erläutert.

IQ-Modulator

Der IQ-Modulator ist im Wesentlichen eine Kombination aus zwei Mischern, die unabhängig voneinander zwei Datensignale auf zwei Trägersignale mischen. Die Besonderheit hierbei ist, dass die zwei Trägersignale sich nicht in ihrer Frequenz, sondern lediglich der Phasenlage unterscheiden. Das Summensignal dieser beiden Mischergebnisse enthält dann die beide Komponenten des komplexen Basisbandes zeitgleich auf einem Träger. Dieses Verfahren hat den Vorteil, dass sowohl Imaginär-

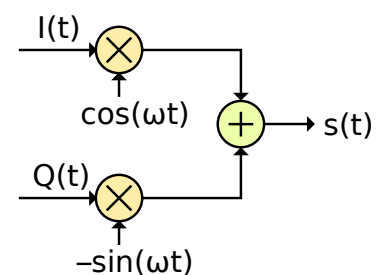


Abbildung 4.12: Prinzip des IQ-Modulators¹

¹Quelle: Walter Dvorak, Wikimedia Commons, Titel: „Prinzip QAM Modulation“

als auch Realteil des Signals gleichzeitig und unter den gleichen Bedingungen übertragen werden können.

Mathematisch lässt sich die Modulation auch als die Multiplikation zweier komplexer Signale darstellen:

$$s_c(t) = (I(t) + jQ(t)) \cdot (\cos(\omega t) - j \sin(\omega t)) \quad (4.19)$$

$$s_c(t) = I(t) \cdot \cos(\omega t) + Q(t) \cdot \sin(\omega t) + j(Q(t) \cdot \cos(\omega t) - I(t) \cdot \sin(\omega t)) \quad (4.20)$$

Das Sendesignal ist hingegen rein reellwertig:

$$s(t) = \text{Re}\{s_c(t)\} = I(t) \cdot \cos(\omega t) + Q(t) \cdot \sin(\omega t) \quad (4.21)$$

Aufgrund dieser Darstellung wird schnell ersichtlich, dass zur Implementierung statt zweier Modulatoren und eines Addierers auch ein komplexer Multiplizierer verwendet werden kann. Ein solcher Multiplizierer steht als IP für den FPGA zur Verfügung.

IQ-Demodulator

Der Demodulator besteht, genau wie der Modulator ebenfalls aus zwei unabhängigen Mischern. Zur Demodulation wird das Empfangssignal wieder mit dem Trägersignal multipliziert. Im Gegensatz zu konventionellen Mischern muss das Trägersignal hier aber nicht nur die gleiche Frequenz, sondern auch die gleiche Phasenlage aufweisen, da eine Phasenverschiebung des Trägersignals direkt an die komplexen Bestandteile des Basisbandes weitergegeben wird. Da Träger- bzw. Referenzsignal in aller Regel lokal auf dem Empfangsgerät erzeugt wird und die Taktgeber in verschiedenen Geräten stets Bauteiltoleranzen unterliegen, ist eine solche kontinuierliche Phasendrehung unvermeidlich. Ist diese Drehung nicht zu groß, so kann diese durch die Verwendung von komplexen Referenzsignalen und Phasenreglern allerdings auch nach der Demodulation kompensiert werden. Auch die Demodulation kann als komplexe Multiplikation komplexer Signale verstanden werden.

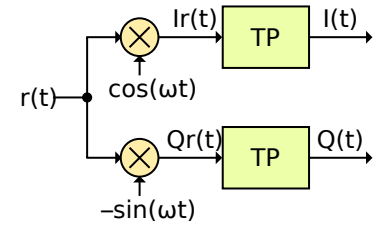


Abbildung 4.13: Prinzip des IQ-Demodulators²

$$y(t) = r(t) \cdot c(t) \quad (4.22)$$

$$y(t) = (I(t) \cdot \cos(\omega t) + Q(t) \cdot \sin(\omega t)) \cdot (\cos(\omega t) - j \sin(\omega t)) \quad (4.23)$$

Unter Berücksichtigung von $\sin(\omega t) \cdot \cos(\omega t) = \frac{1}{2} \sin(2\omega t)$, $\sin^2(\omega t) = \frac{1}{2}(1 - \cos(2\omega t))$ sowie $\cos^2(\omega t) = \frac{1}{2}(1 + \cos(2\omega t))$ löst sich diese Multiplikation auf zu:

$$y(t) = \frac{1}{2} I(t) + j \frac{1}{2} Q(t) - \frac{1}{2} I(t) \sin(2\omega t) + \frac{1}{2} I(t) \cos(2\omega t) - \frac{1}{2} Q(t) \cos(2\omega t) - \frac{1}{2} Q(t) \sin(2\omega t) \quad (4.24)$$

²Quelle: Walter Dvorak, Wikimedia Commons, Titel: „Prinzip QAM Demodulation“

Filtert man die auftretenden Signale doppelter Frequenz jetzt heraus und verstärkt das Basisbandsignal, so erhält man wieder das Ursprungssignal.

Die gleichen Gesetzmäßigkeiten gelten natürlich sinngemäß auch bei zeitdiskreten Signalen, wie sie im FPGA vorliegen.

Verzögerung des Sendesignals und Steuerung der Verstärkung des Empfangsverstärkers

Aufgrund der Umschaltzeiten des Sende-Empfangsschalters ist eine Verzögerung des Sendesignals nötig, während gleichzeitig das Schaltsignal an die Steuertransistoren des Sende-Empfangs-Schalters übermittelt wird. Nach Ablauf der konfigurierbaren Wartezeit wird das Signal für den DAC freigegeben.

Diese Verzögerung funktioniert, indem die *DATA_VALID* und *DATA_READY*-Signale des AXI-Stream Interfaces auf 0 gehalten werden, bis die Wartezeit abgelaufen ist. *DATA_READY* signalisiert dem Sender, dass der Empfänger bereit für neue Daten ist und *DATA_VALID*, dass die im *DATA*-Bereich liegenden Daten für diesen Takt gültig sind. Bei ausbleibenden Signalen stoppt die weitere Kommunikation, um langsamere Teilnehmer nicht zu benachteiligen.

Zur Steuerung der Verstärkung des Empfangsverstärkers wurde ein einfaches PWM-Modul in Very High Speed Integrated Circuit Hardware Description Language (VHDL) entwickelt und mit einem AXI-Light-Interface versehen. Somit können Ausgabewerte einfach im laufenden Betrieb angepasst werden. Die Auflösung ist zur Synthesezeit anpassbar, aber auf 32 bit begrenzt.

4.4 ARM-Prozessor

Auf dem ARM-Prozessor erfolgt die Bereitstellung des Benutzerinterfaces, die Protokollbehandlung, die Signalverarbeitung im komplexen Basisband sowie die Übergabe der Daten mittels DMA. Mit Ausnahme des Treibers für den DMA wurde der gesamte Code in *C++* unter Verwendung der unter Abschnitt 3.4 erwähnten Bibliotheken erstellt.

4.4.1 CModem-Programm

Da die Kommunikation zwar über den Unterwasserkanal stattfindet, aber niemals dort beginnt, oder endet ist eine Software nötig, die von der Entgegennahme der Daten von der Datenquelle bis zur Übergabe der übertragenen Daten an die Datensenke die Funktionen des Modems steuert.

Dafür wurde hier das CModem-Programm geschrieben. Dieses Programm erlaubt die Funktionsparameter des Modems, wie Sendefrequenz, Bandbreite, Anzahl der Subträger sowie derzeit noch die Adresse des Zielmodems sowie den Verstärkungsfaktor des Empfangsverstärkers einzustellen. Alle nachfolgend angeführten Programmbestandteile sind objektorientiert und mit Hinblick auf Multithreading entwickelt worden, denn auch wenn der Prozessor derzeit nur zwei Kerne zur Verfügung hat, ist einerseits nicht sicher, ob zumindest Teile des Programms auch Anwendung auf anderen Plattformen finden werden. Außerdem erzeugen Threads bei modernen Schedulingern kaum Overhead, erlauben aber eine bessere Verteilung der Last auf die einzelnen Prozessorkerne.

Benutzerinterface

Da in diesem Projekt ein Übertragungskanal erstellt werden sollte, mussten Verluste und Übertragungsfehler an allen anderen Stellen möglichst ausgeschlossen werden. Gleichzeitig ist das Modem möglichst unter Wasser, zumindest aber in einem geschlossenen Druckkörper untergebracht. Ein direktes Mensch-Maschine-Interface auf dem Gerät selbst war also ausgeschlossen. Unter Verwendung des auf dem *RedPitaya-Boards* vorhandenen Ethernet-Anschlusses wurde schließlich ein Ethernet-Server geschaffen. Eingehende Verbindungsversuche resultieren in Erzeugung eines eigenen Sockets für jede Verbindung und Entgegennahme und Ausgabe aller Daten an dieses Interface.

Diese simple Lösung erlaubt sowohl den Versand von Textnachrichten, als auch Dateien mittels selbstentwickelten Programmen oder *Linux*-Boardmitteln wie *netcat*.

Nachrichtenprotokoll

Die grundsätzliche Frage bei der Übertragung der Daten ist, welcher Wert auf Sicherheit der Übertragung gelegt werden soll. Sollen Multimediadaten, wie Audio- oder Videostreams übermittelt werden, ist oftmals eine latenzarme Übertragung wichtiger als eine vollständige oder auch fehlerfreie Übermittlung. Da das Modem zum Abruf der Messdaten autonomer Unterwasserfahrzeuge entwickelt wird, wird Wert auf die Vollständigkeit der Daten gelegt. Die zu übermittelnden Daten müssen daher organisiert und die Übertragung überwacht werden. Deshalb wurde ein einfaches Protokoll unter folgenden Kriterien erstellt:

- Da die Übertragungen des Modems zwischen den einzelnen Kommunikationsteilnehmern, wie unter 4.2.3 beschrieben, zeitmultiplext werden, kann jederzeit ein Sendekonflikt (gleichzeitiges Senden) mit einem bekannten, oder neu hinzugekommenen Partner entstehen.
- Aufgrund der geringen Reichweite kann ein Partner den Empfangsbereich schnell verlassen.
- Daten sollen keinesfalls unbemerkt unvollständig oder fehlerhaft übertragen werden.
- Eine Übertragungsbestätigung soll möglichst, aber nicht zwingend erfolgen, denn Mehrfachsendungen werden in Kauf genommen.
- Der Protokolloverhead soll möglichst gering gehalten werden.

Während alle Anforderungen auch mit gängigen Protokollen, wie TCP, User Datagram Protocol (UDP) / IP gelöst werden können, so erfüllt keines der Protokolle alle Kriterien. Da das System nicht duplexfähig, die Umschaltvorgänge vergleichsweise langsam sind sowie der Übertragungsoverhead bei kurzen Übertragungen deutlich stärker ins Gewicht fällt (da dieser je Frame anfällt), sollten längere/zusammengefasste Übertragungen mit relativ seltenen Paket- und Teilnehmerwechseln angestrebt werden. Dies lässt sich durch Wartezyklen erreichen, die, damit sich Teilnehmer nicht durch gleichzeitige Sendeveruche überhören können, in Maßen mit zufälligen Wartezeiten versehen werden.

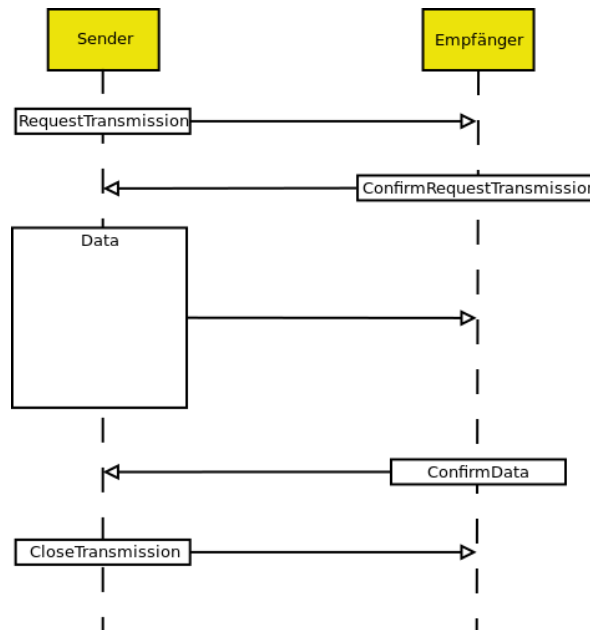


Abbildung 4.14: Sequenzdiagramm des Protokolls

Sendesignale

Wie bereits in Abschnitt 3.4.2 erwähnt, erfolgt die Erzeugung der Sendesignale mittels der in der Library *Liquid-DSP* verfügbaren Funktionen (s. [Gaeddert, 2014]). Verwendet werden hierbei sowohl Ein- als auch Mehrträgerverfahren in Form der Strukturen *FlexFrame* und *OFDMFlexFrame*.

FlexFrame

Die *FlexFrame*-Struktur kann in sechs Phasen eingeteilt werden:

Abbildung 4.15: Aufbau der FlexFrame-Struktur³

1. **Anfahrphase**: In diesem Abschnitt wird das Ausgangssignal schrittweise erhöht, um steile Transienten im Sendesignal zu vermeiden. Der Empfänger nutzt diese Phase mit bekannter Anstiegsgeschwindigkeit, um die Automatic Gain Control (AGC) anzupassen.
2. **Präambel**: Die nachfolgende Präambel besteht aus einer Reihe von wechselnden Binary Phase Shift Keying (BPSK)-Symbolen, um ein möglichst schnelles Einstellen der Empfänger-PLL auf das ankommende Signal zu gewährleisten.

³Quelle: Joseph Gaeddert, Titel: „Framing Structure“

3. Autokorrelationssequenz: Die gesendete Autokorrelationssequenz (63 bit M-Sequenz) dient zwei Zwecken: Einerseits ist es nach der Präambel noch möglich, dass die PLL des Empfängers sich auf eine genau gedrehte Phasenlage eingestellt hat, da das BPSK-Signal zur Unterscheidung nicht genug Symbole bietet. Andererseits dient sie natürlich der zeitlichen Synchronisation der Struktur.
4. Header: Der Header besteht aus einer BPSK Header-Präambel, die dem Empfänger Paketlänge, Modulationsverfahren und Kanalkodierung des Payloadfeldes übermittelt, gefolgt von einem 14 Byte Datenfeld, wovon in diesem Projekt fünf Byte für Protokollinformationen genutzt werden.
5. Payload: Die Payload besteht aus den eigentlichen Nutzerdaten. Die Länge, Modulationsart und bis zu zwei Kanalkodierungen sowie eine Checksummenmethode können frei gewählt werden.
6. Abfallphase: Entsprechend zur Anfahrtphase gibt es auch eine Abfallphase, um Transienten zu vermeiden.

OFDMFlexFrame

Die *OFDMFlexFrame*-Struktur folgt einem ähnlichen Ansatz, wie die *FlexFrame*-Struktur, verfügt aber über keinen Einträger-Header, weshalb der Empfänger die Anzahl der gewählten Subträger kennen muss, und kann in drei Phasen eingeteilt werden:



Abbildung 4.16: Aufbau der OFDMFlexFrame-Struktur⁴

1. Pilotenphase: In dieser Phase werden mindestens zwei Pilotensymbole ohne Cyclic Prefix gesendet, um eine Synchronisation zu ermöglichen. Anschließend folgt ein Referenzsymbol mit Cyclic Prefix, um die Synchronisation weiter zu verbessern.
2. Header: Der Header besteht hier aus einem oder mehreren OFDM-Symbolen, allerdings ohne BPSK Header-Präambel, weshalb dem Empfänger Paketlänge, Modulationsverfahren und Kanalkodierung des der Payloadsymbole in den ersten sechs Bytes des Headers übermittelt werden, weshalb dem Nutzer nur ein 8 Byte Datenfeld zur Verfügung steht, wovon hier, wie beim Einträgerverfahren fünf Byte für Protokollinformationen genutzt werden.
3. Payload: Die Payload besteht wieder aus den eigentlichen Nutzerdaten. Die Länge, Modulationsart und bis zu zwei Kanalkodierungen sowie eine Checksummenmethode können wieder frei gewählt werden.

⁴Quelle: Joseph Gaeddert, Titel: „OFDM Framing Structure“

4.4.2 DMA-Treiber

Zur Übermittlung der Daten des komplexen Basisbandes an den FPGA kommt hier ein DMA zum Einsatz. DMAs arbeiten wie Koprozessoren mit einem sehr beschränkten Aufgabenfeld. Die einzige Aufgabe ist der Transport von Daten von einem Speicherbereich in einen anderen. Man unterscheidet hierbei in Linux-Systemen grundsätzlich zwischen DMA mit zusammenhängendem Speicher und *Scatter-Gather*-DMA. DMA mit zusammenhängendem Speicher arbeiten einen kontinuierlichen Speicherbereich von Anfang bis Ende ab. *Scatter-Gather*-DMA wiederum wurden entwickelt, weil kohärenter zusammenhängender Speicher in von Betriebssystemen verwalteten Systemen sehr begrenzt ist. Um die Speicherverwaltung effizient arbeiten zu lassen, wird die reservierbare Speichergröße stark begrenzt. Daher werden für *Scatter-Gather*-DMA mehrere kleine Speicherbereiche reserviert und mittels einer Tabelle im DMA miteinander verknüpft. Da die für dieses Projekt erforderliche Speichermenge aber noch zusammenhängend reservierbar ist, kann hier darauf verzichtet werden.

Der hier verwendete DMA greift von der PL über den AXI-HP-Bus direkt auf den Arbeitsspeicher zu. Um dem DMA jetzt Daten zum Transport zu übergeben, müssen diese an die richtige Speicherstelle im Arbeitsspeicher abgelegt werden. Da aber die Speicheradressen in *Linux*-Systemen virtuelle und keine physikalischen Adressen sind, kann nicht ohne weiteres auf diese Adressen zugegriffen werden. Es wird neben dem Physikalischen Adressraum grundsätzlich zwischen *Kernel*-Raum und *User*-Raum unterschieden.

Dieses Sicherheitsmerkmal, das verhindern soll, dass Kernelprogramme dazu gebracht werden können, Benutzercode mit erhöhten Rechten auszuführen, verhindert die Übersetzung der Adressen. Lediglich Programme im *Kernel*-Raum können diese Übersetzung in den Physikalischen Adressraum vornehmen.

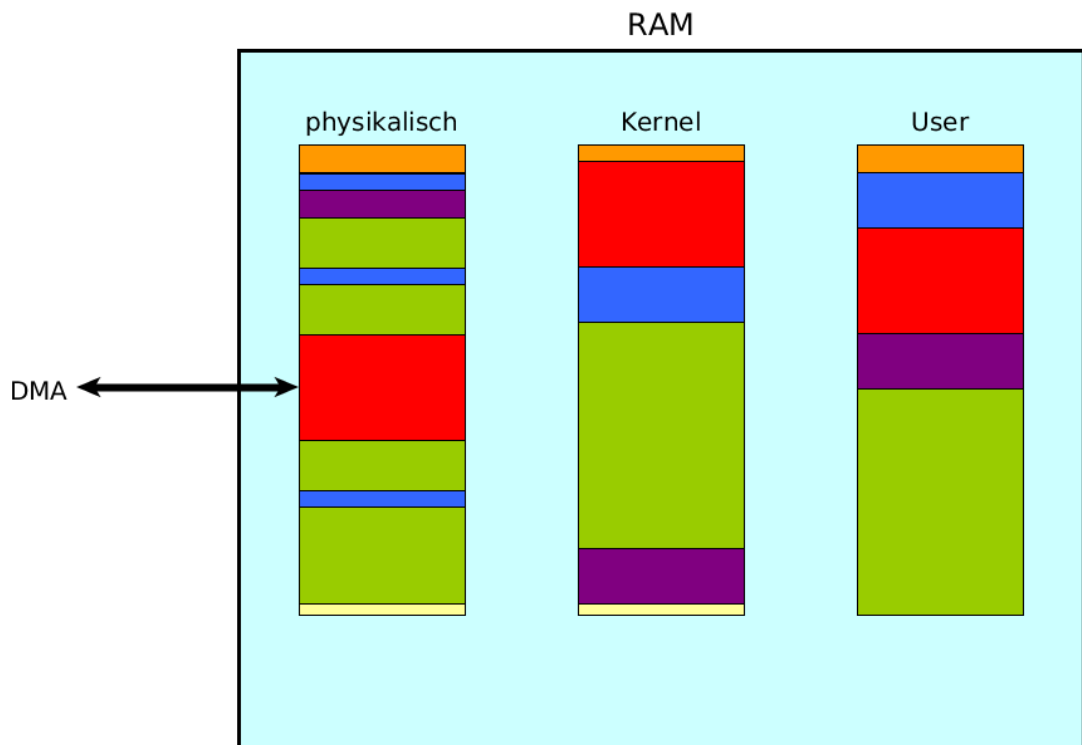


Abbildung 4.17: Adressierung des Arbeitsspeichers

Es wurde daher ein Kernaltreiber auf Basis des von *Xilinx* zum DMA-IP ausgelieferten Basis-treibers entwickelt, der wiederum mittels *System-Calls* vom CModem-Programm angesprochen werden kann.

Dazu erstellt der DMA-Treiber ein Filedescriptor unter „/dev/cmodem-axidma-Tx“ im Dateisystem. Mittels der Systemcalls „CMODEM_GET_DMA_SETTINGS“ sowie „CMODEM_SET_DMA_SETTINGS“ können Einstellungen, wie erforderliche RAM-Größe, Timeoutzeit eingestellt sowie der blockierende Modus ein- bzw. ausgestellt werden. Im blockierenden Modus gibt der Systemcall „CMODEM_START_TRANSFER“ erst ein Ergebnis zurück, wenn der Transfer abgeschlossen ist, im nicht blockierenden Modus ist der Nutzer für die Überwachung der Transaktion verantwortlich. Dies kann beispielsweise geschehen, indem der Füllstand des an den DMA angeschlossenen FiFo-Speichers geprüft wird. Dieser fasst ca. 8.2kSamples. Somit können Sendeblocke dann an das AXI-Stream-Interface der Verarbeitungskette im FPGA übergeben werden, indem auf den Filedescriptor „/dev/cmodem-axidma-Tx“ Daten abgelegt werden und der Transfer über Systemcall gestartet wird.

4.5 Qi-Umsetzung zur Energieübertragung

Um die Entwicklungszeit möglichst kurz zu halten, war es Ziel, die Energieübertragung nicht komplett neu zu entwerfen, sondern käuflich zu erwerbende Systemkomponenten einzusetzen, um einen Demonstrator aufbauen zu können.

Da der Medium-Power-Standard zum Zeitpunkt des Projektabschlusses nicht abschließend spezifiziert war, sollten Komponenten aus dem Low-Power-Bereich eingesetzt werden.

Sender

Senderseitig fiel die Entscheidung hierbei zugunsten den BQ501210 von Texas Instruments, da dieser sowohl in einem handlötbaren Package verfügbar war, als auch die erforderliche Leistung übertragen kann. Für den Entwurf wurde sich am Referenzdesign orientiert, was eine Kalibrierung und sogar Abschaltung der Foreign-Object-Detection vorsieht und als eigenständiges Modul umgesetzt werden kann. Daher wurde eine im CModem stapelbare Komponente entworfen und umgesetzt. Wie alle Qi-Sender besteht der Leistungsteil hier auch aus einer H-Brücke, die die Spule als Last betreibt.

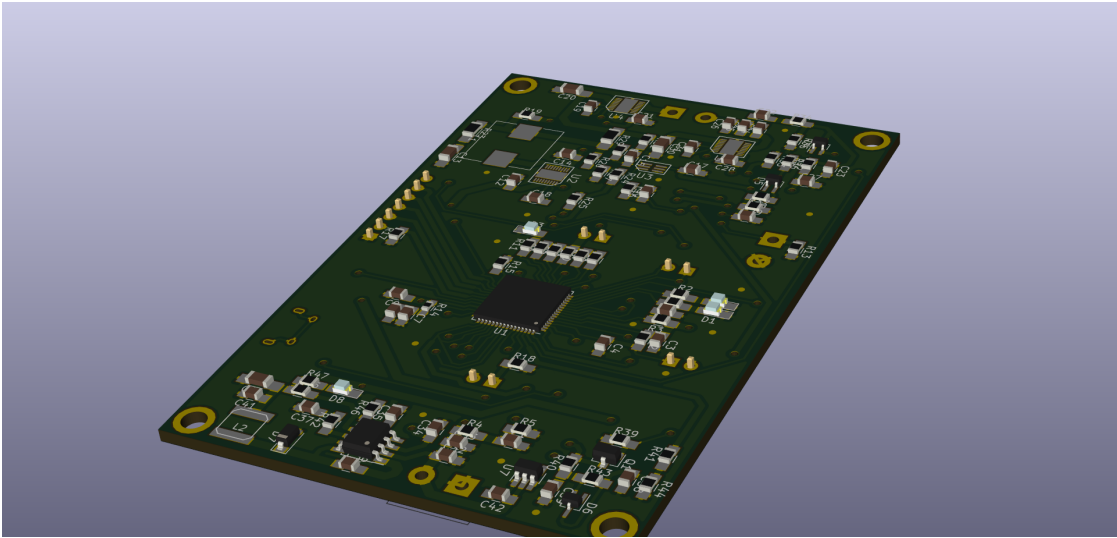
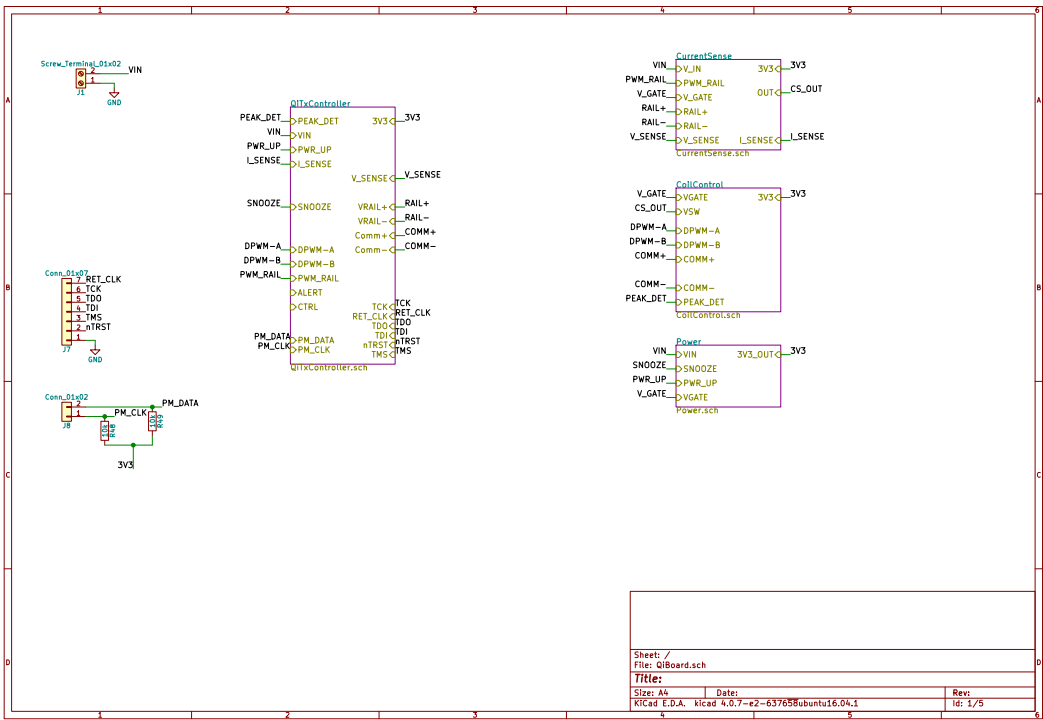


Abbildung 4.18: Rendering des Qi-Senderboards

5 Implementierung und Messung der Komponenten

Ein großer und meist stark unterschätzter Teil der Entwicklungsarbeit befasst sich mit der Umsetzung der erdachten Konzepte und Entwürfe in funktionsfähige Prototypen. Frühestens an dieser Stelle fallen Fehler im Konzept und Inkompatibilitäten auf, die wieder neue Anpassungen erfordern. Dieses Kapitel beschreibt folglich die Umsetzung und Vermessung der in Abschnitt 4 entworfenen Teilkomponenten, wodurch am Ende ein funktionsfähiges Übertragungssystem entstand.

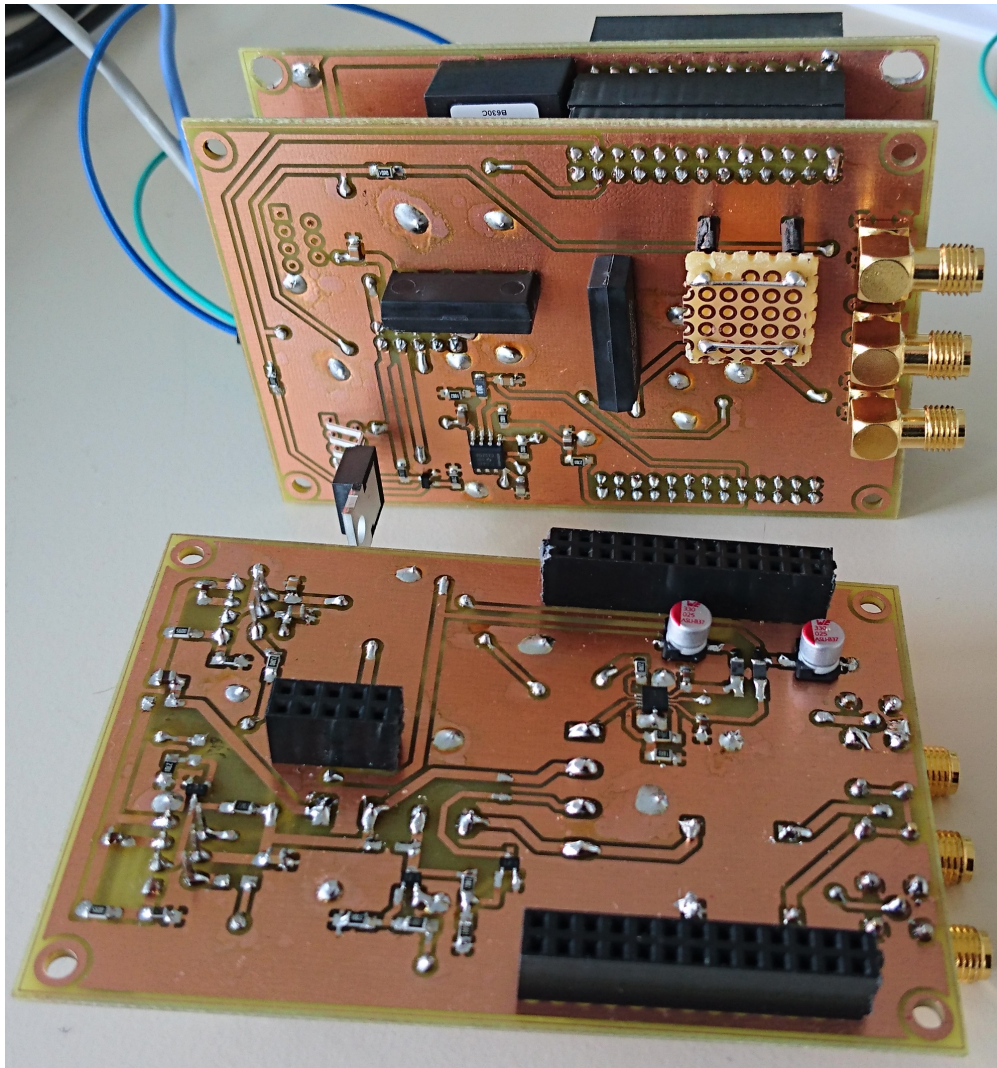


Abbildung 5.1: Vorder- und Rückseite des *TransmissionBoards* mit und ohne aufgestecktes *PowerBoard*

5.1 Hardware

Um den beschränkten Bauraum optimal ausnutzen zu können, musste die Hardware von Beginn an möglichst klein konstruiert werden, ohne die für die Prototypenphase unerlässliche Testbarkeit des Systems zu verlieren. Es wurde daher ein Stapelsystem verwendet, was die *Extension Header*, die teilweise genutzt werden, in die entwickelte Hardware durchzureichen. Da in diesem System sowohl Eingangsverstärker mit möglichen hohen Verstärkungen als auch Schaltwandler mit hoher Systemleistung kombiniert werden müssen, fiel die Entscheidung zu Gunsten eines Konzeptes mit zwei Platinen.

Das *PowerBoard* umfasst alle Spannungswandler samt der dazugehörigen Beschaltung. Das *TransmissionBoard* dagegen enthält nur noch zur Signalverarbeitung nötige Elemente. Dadurch konnten Störungen bereits im Vorfeld reduziert werden. Als Software für Schaltplan- und Platinenentwicklung kam *KiCad* zum Einsatz.

5.1.1 Übertragungsspulen

Die entworfenen Spulen konnten gemäß den Überlegungen aus 4.1.1 problemlos per Hand gewickelt und unter Zuhilfenahme eines Netzwerkanalysators anschließend vermessen werden. Nach mehreren Versuchen konnte ein passendes Spulenpaar geschaffen werden, wovon eine Spule hiervon kurz betrachtet wird.

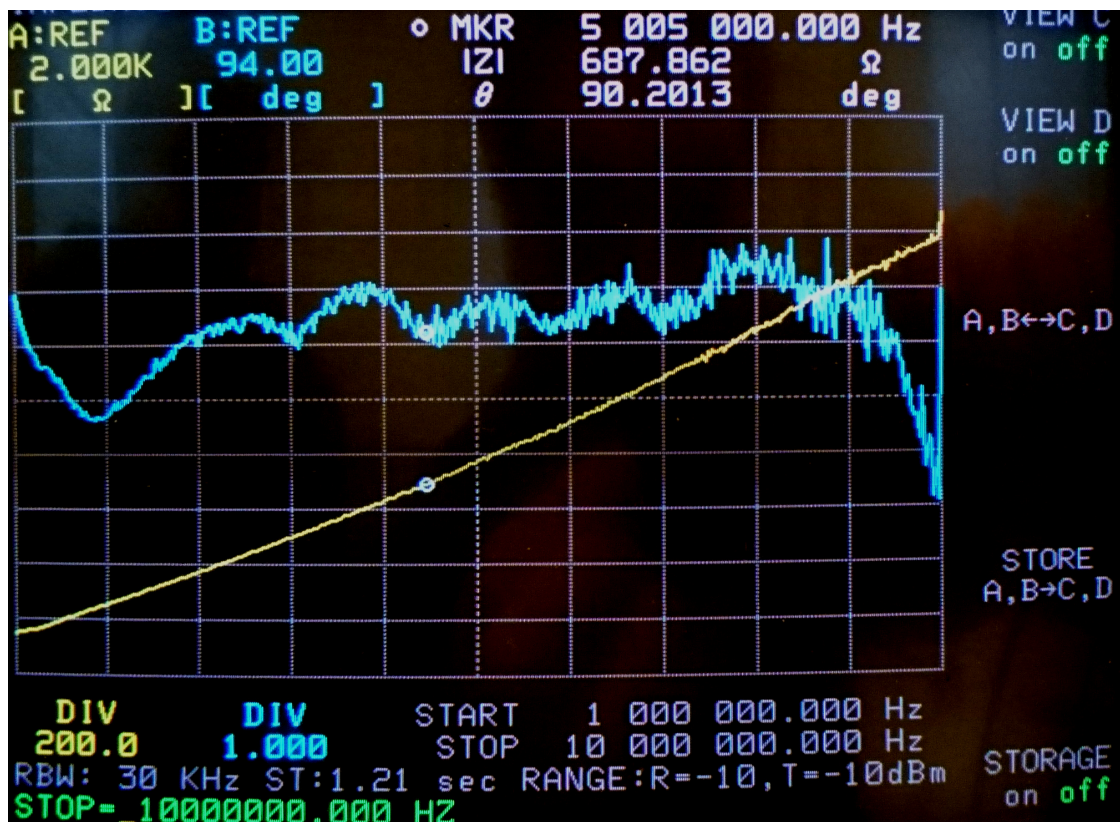


Abbildung 5.2: Impedanzverlauf der unangepassten Spule

Auf der Übertragungsfrequenz ergab sich, wie auf Abbildung 5.2 zu erkennen, sich eine Impedanz von:

$$Z_{\text{Spule}} = \left| 608 \cdot e^{j90.8^\circ} \right| \Omega = (8.5 + j 607.9) \Omega \quad (5.1)$$

Der Realteil entsprach damit annähernd genau der gewünschten Impedanz von 9Ω .

Nach Implementierung der Anpassungsschaltung für beide Spulen, konnte das Übertragungsverhalten Spule zu Spule gemessen werden.

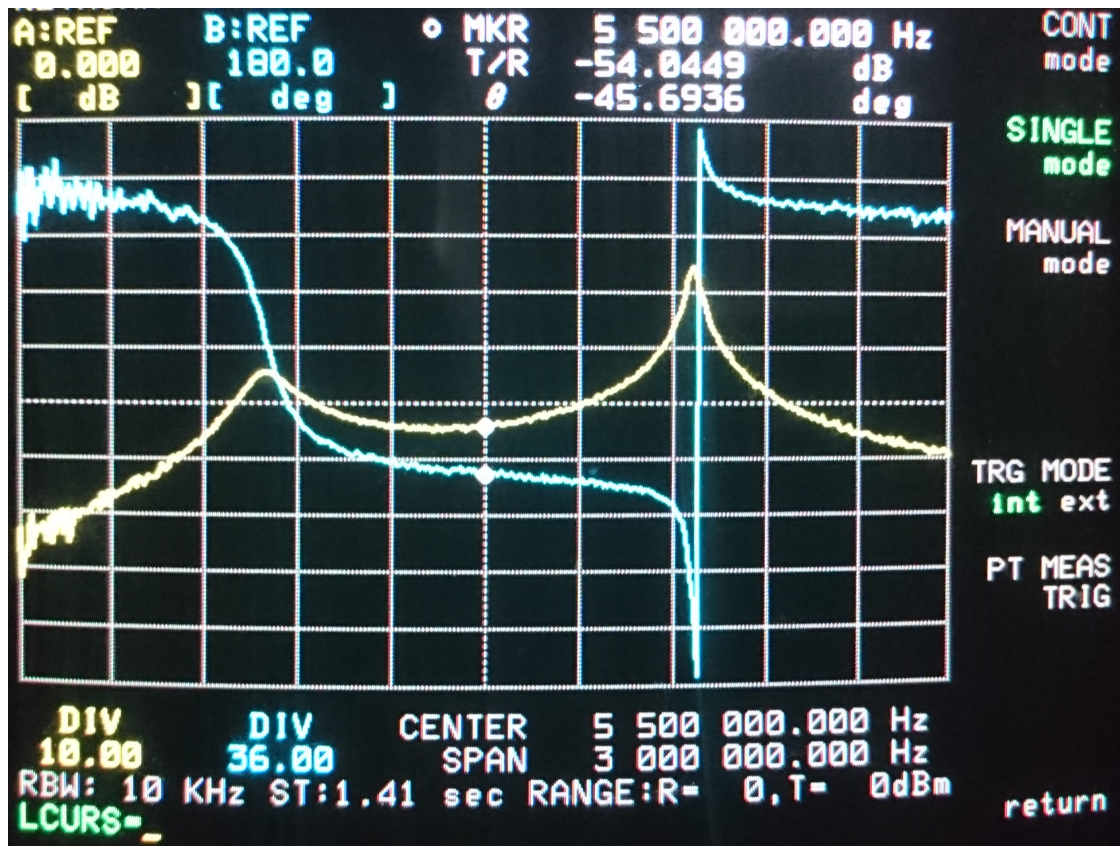


Abbildung 5.3: Amplituden- und Phasengang der Kommunikationsspulen (Abstand: 15 cm)

Die Verstimmung der beiden Spulen bewirkte tatsächlich die Schaffung eines linearphasigen Frequenzbereichs zwischen den beiden Spulen. Dieser weist zwar keine konstante Dämpfung auf, dieser Dämpfungsverlauf ist allerdings bekannt, was bei der weiteren Entwicklung des Systems genutzt werden kann. Beispielsweise kann die Sendeleistung einzelner OFDM Subträger eines Frames daran angeglichen werden. In Rahmen dieser Arbeit fand diese Methode jedoch noch keine Anwendung. Bei dem gewählten Abstand von 15 cm liegt die Dämpfung im Bereich $-54 \text{ dB} \leq D \leq -44 \text{ dB}$. die Grenzen des Kommunikationsbandes sind also:

$$f_c = 5.5 \text{ MHz}$$

$$B = 900 \text{ kHz}$$

5.1.2 TransmissionBoard

Das *TransmissionBoard* beinhaltet Sende- und Empfangsverstärker sowie die Sende-Empfangs-Umschaltung mittels zweier Reed-Relais. Ferner wurde ein Steckplatz für eine Antennenanpassungsschaltung vorgesehen, so dass diese zukünftig nicht mehr direkt an der Antenne erfolgen muss. Der Empfangsverstärker wurde wegen der großen möglichen Eingangsverstärkung mit Abstand zu anderen Baugruppen und nahe am Antenneneingang platziert, damit so wenig Störeinkopplungen wie möglich erfolgen können.

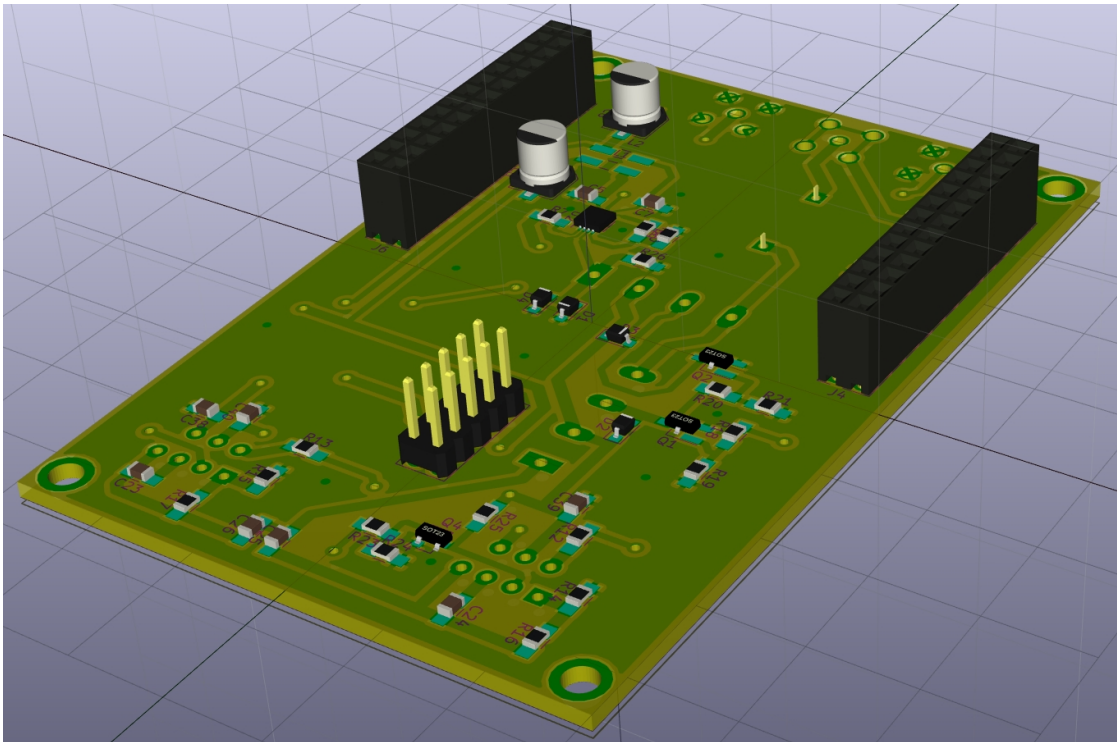


Abbildung 5.4: Rendering der Unterseite des *TransmissionBoards*

Eingangsverstärker

Neben dem Übertragungskanal der Spulen sollte natürlich auch der Frequenzgang des Eingangsverstärkers einen möglichst flachen Amplitudengang und einen linearen Phasengang aufweisen, was mittels Netzwerkanalysator überprüft werden konnte. Dazu wurde der Eingangsverstärker in der Mitte des Verstärkungsbereiches betrieben, was gemäß Entwurf $\frac{24}{2} \text{ dB} + 21 \text{ dB} = 33 \text{ dB}$ entsprechen sollte. Gemessen wurden 34.6 dB bei 5.5 MHz.

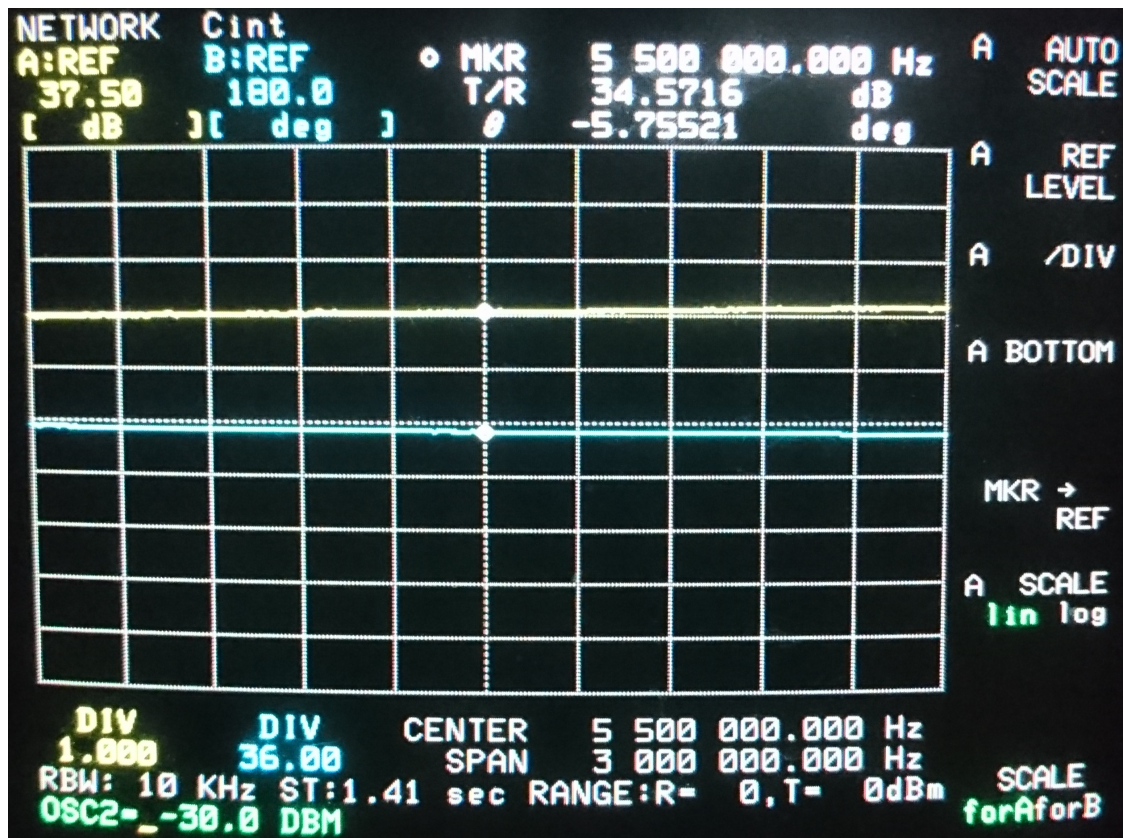


Abbildung 5.5: Amplituden- und Phasengang des Eingangsverstärkers

Neben einer guten Übertragungsfunktion ist aber vor allem der Rauschanteil des verstärkten Signals von Bedeutung. Daher wurde noch eine Messung im Zeitbereich vorgenommen. Zur Bestimmung des SNR sind sowohl Messungen des verrauschten Signals, als auch Messungen des Rauschens selbst nötig. Leider waren alle verfügbaren Signalgeneratoren bei den niedrigen Ausgabepegeln sehr stark lastabhängig und damit eine aussagekräftige Messung nicht möglich.

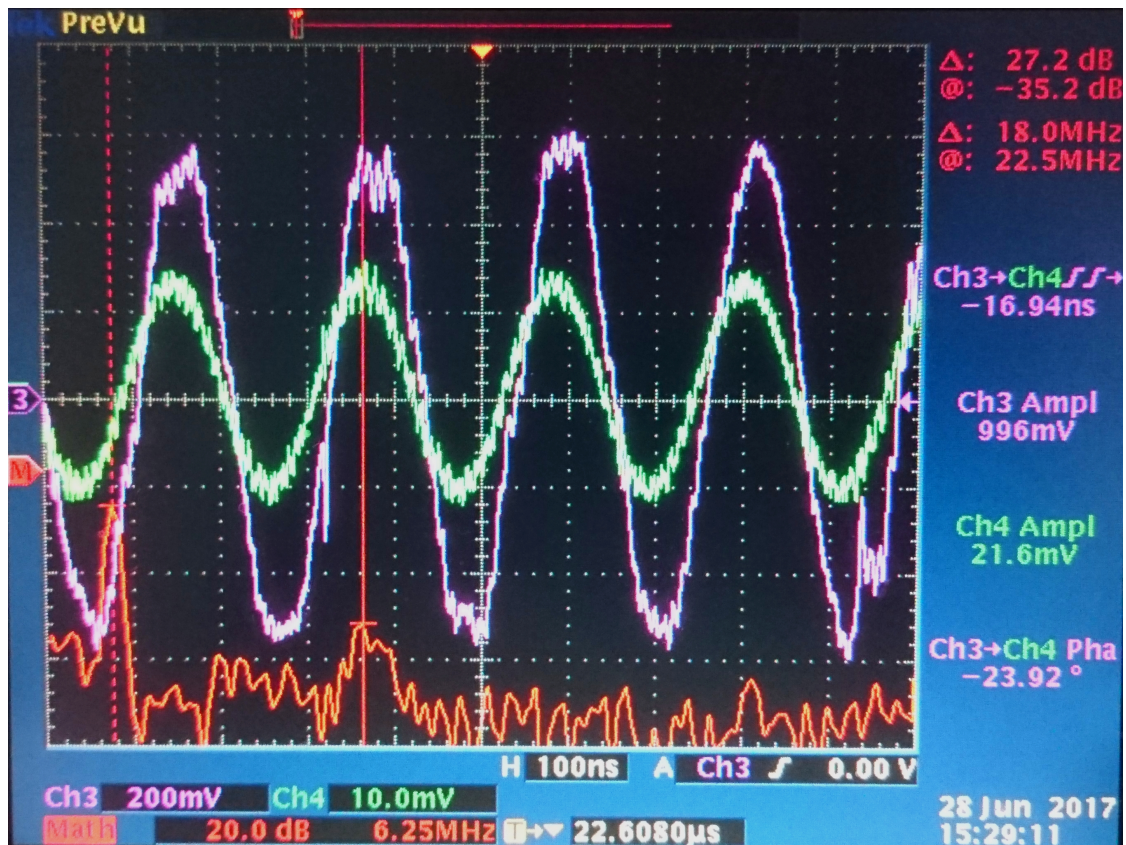


Abbildung 5.6: Verstärkungsmessung des Eingansverstärkers

Die größten Rauschanteile liegen, wie in Abbildung 5.6 ersichtlich, allerdings ohnehin außerhalb des gewählten Bandes und werden von den digitalen Empfangsfiltern unterdrückt. Lässt man für den verbleibenden Bereich die Schmalbandannahme zu, laut der sich die Leistungsdichte proportional zu den Effektivwerten der Spannungen verhält, da die Signal und Rauschleistungsdichte bei sehr kleiner Bandbreite sich dem Effektivwert annähert, so kann man ein SNR von mindestens 30 dB abschätzen.

Die optimale Kanalkapazität des Systems ergibt sich daher zu:

$$C_b = B \cdot \log_2 \left(1 + \frac{S}{N} \right) = 900 \text{ kHz} \cdot \log_2 (31 \text{ dB}) = 4.46 \frac{\text{MBit}}{\text{s}} \quad (5.2)$$

Sende-Empfangs-Umschaltung

Der Sende-Empfangs-Schalter ist ein Zusammenspiel der mechanischen Sende- und Empfangsumschalter und der im FPGA implementierten Umschaltlogik, die die Signalverzögerung und Generierung der Steuersignale übernimmt. Die Verzögerungszeit wurde bei einer im Datenblatt (siehe [SIL1A72, 2015]) angegebenen Umschaltzeit von 1 ms auf 1.3 ms gesetzt, um ein vollständiges Umschalten und Entprellen zu gewährleisten.

Betrachtet man allerdings die Messung der Umschaltzeit im Oszilloskop (Abbildung 5.7), so kann im Optimalfall bereits nach 80 μs die Umschaltung und nach 400 μs ein statischer Strom über das Reed-Relais registriert werden.

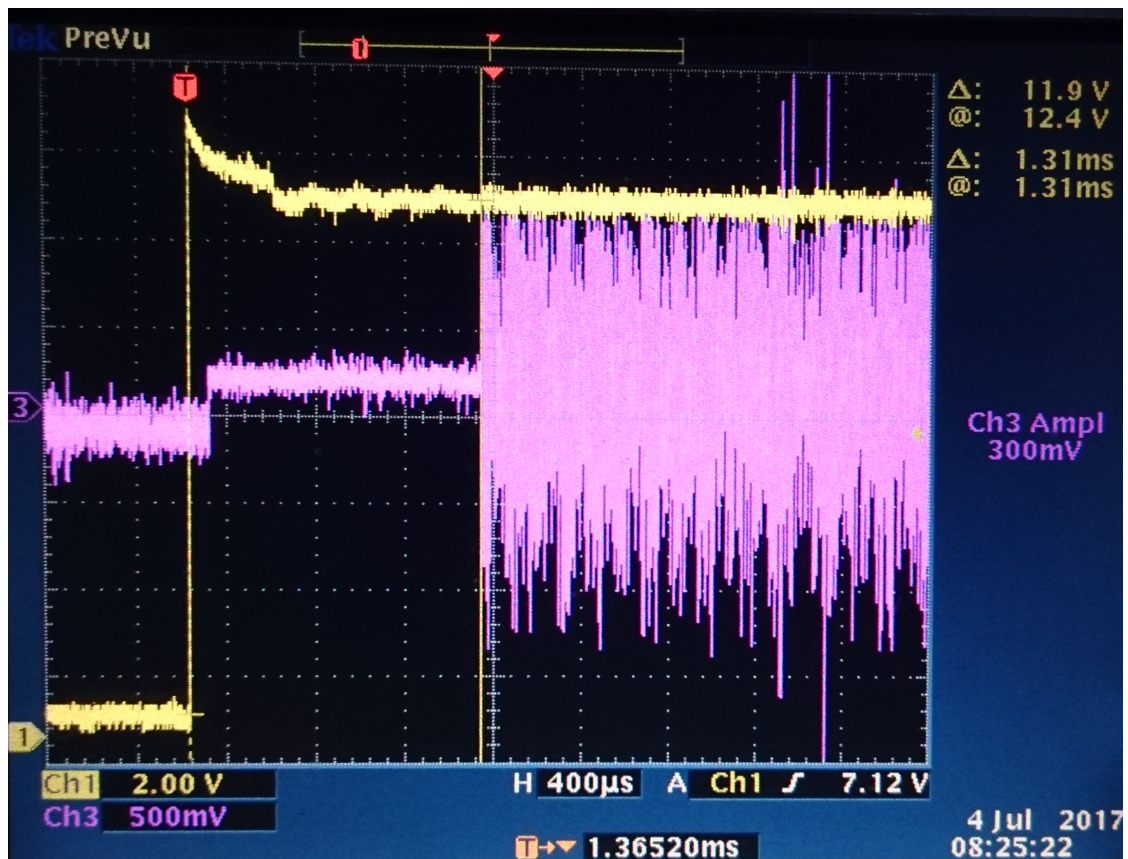


Abbildung 5.7: Umschaltmoment des Sende-Empfangsumschalters

Vermessung des Kanals

Die Vermessung des Kanals vom Sendeverstärker zum Empfangsverstärker soll hier die Funktionalität der Hardware belegen. Da die Ergebnisse mit den Ergebnissen der Untersuchung der einzelnen Elemente der Hardware vergleichbar bleiben sollen, wurde die Verstärkung des Sendeverstärkers an der Sendefrequenz (5.5 MHz) auf 0 dB kalibriert und anschließend die Strecke erneut vermessen.

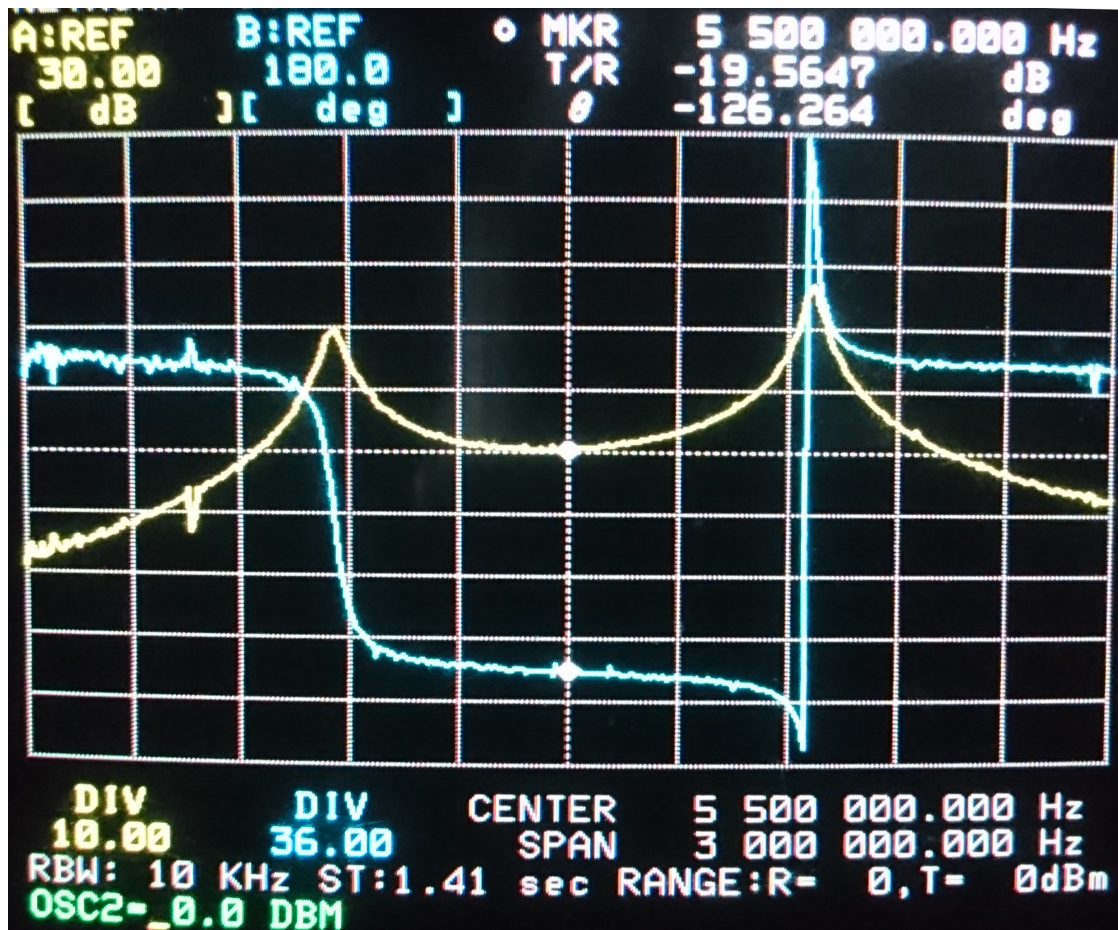


Abbildung 5.8: Amplituden- und Phasengang des gesamten Übertragungsweges (Abstand: 15 cm)

Die frequenzabhängige Dämpfung ist mit Sende- und Empfangsverstärker deutlich höher, so liegt der Unterschied zwischen niedrigster und höchster Überhöhung der Außenfrequenzen des Bandes gegenüber der Mittenfrequenz jetzt bei 15 dB anstatt der bisherigen 10 dB. Die Kompensation ist also für die zukünftige Entwicklung durchaus bedeutsam.

5.2 FPGA-Logik

Die in Abschnitt 4.3 erläuterten Verfahren wurden in der *Xilinx*-eigenen Entwicklungsumgebung *Vivado* entworfen. Mit Ausnahme des PWM-Generators sowie der Verzögerungslogik für die Sende-Empfangs-Umschaltung konnten für alle Aufgaben IP-Cores aus dem *Xilinx*-Repository oder der *Red-Pitaya-Notes*-Library verwendet werden. Die Implementierung des Modulators ist eine Variante des unter [XAPP1113, 2008] vorgestellten Entwurfs für digitale Modulatoren und Empfänger mittels FPGA unter Zuhilfenahme von Konzepten und IP-Cores des *RedPitaya*-Projektes. Soweit nicht explizit erläutert, bestehen alle übrigen Blöcke aus *Xilinx* IP-Cores.

5.2.1 Empfänger

Die Empfangsseite des FPGA-Designs besteht lediglich aus den drei Blöcken ADC-Ansteuerung, IQ-Demodulator und RAM-Writer. Während die erst- und letztgenannten Blöcke IP-Cores aus dem *Red-Pitaya-Notes*-Projekt sind, war dies beim IQ-Demodulator nicht so einfach lösbar.

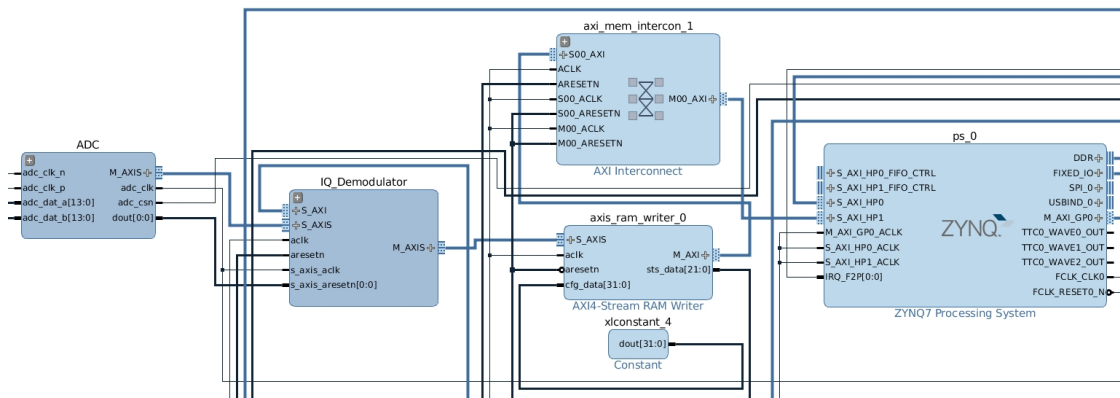


Abbildung 5.9: Übersicht des Blockdesigns der Empfangsseite des FPGA

Der IQ-Demodulator enthält zunächst einen konfigurierbaren Direct Digital Synthesis (DDS)-Compiler zur Erzeugung des Trägersignales. Das Signal wird sowohl in Phase als auch mit 90° Phasenverschiebung erzeugt, um die I- und Q-Phase demodulieren zu können. Die Ausgabe erfolgt in der gleichen Frequenz, in der auch die Samples des ADC eintreffen, also 125 MHz. Das Eingangssignal wird zunächst mit dem Trägersignal komplex multipliziert (vgl.4.3.1), eine Phasenkorrektur kann zu diesem Zeitpunkt nicht erfolgen, da noch keine Kenntnisse über eventuelle Pilotinformationen vorliegen. Dies ist aufgrund des modularen Designs hier nicht möglich, da sonst eine Änderung des Multiplexings oder Modulationsverfahrens immer auch im FPGA erfolgen müsste; somit wird der Phasenfehler bis in die Signalverarbeitung im ARM-Kern weitergereicht.

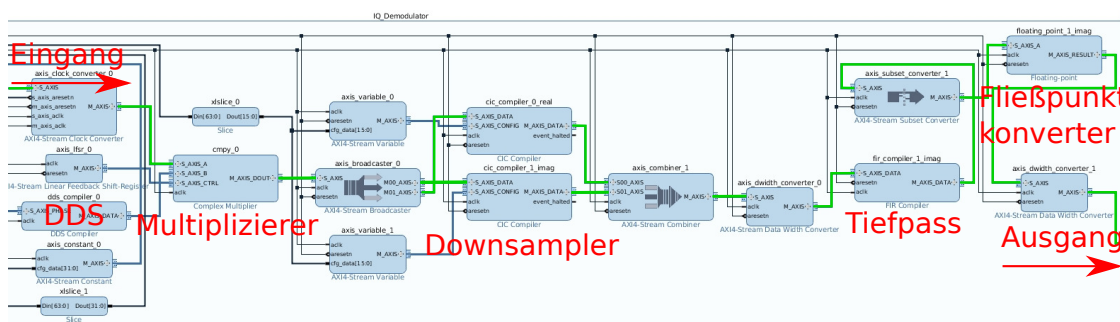


Abbildung 5.10: Darstellung des IQ-Demodulators, der Signalweg des Empfangssignals ist in grün hervorgehoben.

Erst nach der Phasentrennung erfolgt das Downsampling auf die doppelte Signalfrequenz. Die in den verwendeten Cascaded Integrator-Comb (CIC)-Compilern implementierten und den nachfolgenden vom Hersteller empfohlenen Kompensationsfiltern (siehe [PG140, 2016])

erweiterten Tiefpassfilter dämpfen alle ungewünschten Mischprodukte heraus. Die abschließende Umwandlung in 32 bit Fließpunktzahlen spart diese Umwandlung im ARM-Prozessor ein.

Pakete, damit keine Übertragungen initiiert werden und stellt ihn für einen Wert ca. 30 % über der erwarteten Übertragungszeit neu ein. Bei jedem empfangenen Datenpaket prüft der Empfänger anhand der Informationen im Header, ob das jeweilige Paket bereits vollständig empfangen wurde, und speichert eine Quittierung eines jeden Frames in einem dafür vorgesehenen Frametyp. Ein einzelner Frame wird im Anschluss an den Datenempfang als Quittierung gesendet. Ist ein Paket vollständig, wird es an alle mit dem eigenen TCP-Port verbundenen Teilnehmer gesendet. Empfängt der Sender die Quittierung, löscht er das Paket auf seiner Seite und sendet ein Signal zum Schließen der Verbindung. Bleiben Frames beim Empfänger aus, so sendet dieser eine Anfrage zur erneuten Übermittlung nach einer konfigurierbaren Anzahl von Timerüberläufen. Empfängt der Sender keine Quittierung, sendet er wiederum die Daten, nach erneuter Sendeabfrage, einfach erneut.

5.4 Übertragung der Frames

Da die eigene Software einerseits, die *Liquid-DSP*-Library andererseits die Möglichkeit zur Aufzeichnung von Rohdaten, bzw. Modulationsdiagrammen bietet, kann die Qualität der übertragenen Daten gut abgeschätzt werden. Da aus zeitlichen Gründen noch kein systematischer Test über verschiedene Übertragungsstrecken unter wechselnden Bedingungen stattfinden konnte, werden hier nur einzelne Messergebnisse aufgezeigt. Alle Messungen erfolgen mit dem Modem im geschlossenen Druckkörper unter verschiedenen Bedingungen. Gesendet wurde jeweils ein Einträgersignal mit einer QAM16-Payloadmodulation. Da keine Messung des Salzgehaltes vorgenommen werden konnten, ist leider die quantitative Aussage der Messergebnisse bisher noch sehr eingeschränkt. Allerdings konnte ein grundsätzlicher Funktionsnachweis des Modems erbracht werden. Übermittelt wurden jeweils 40 Byte bei einer einfachen Hamming(7,4)-Kodierung.

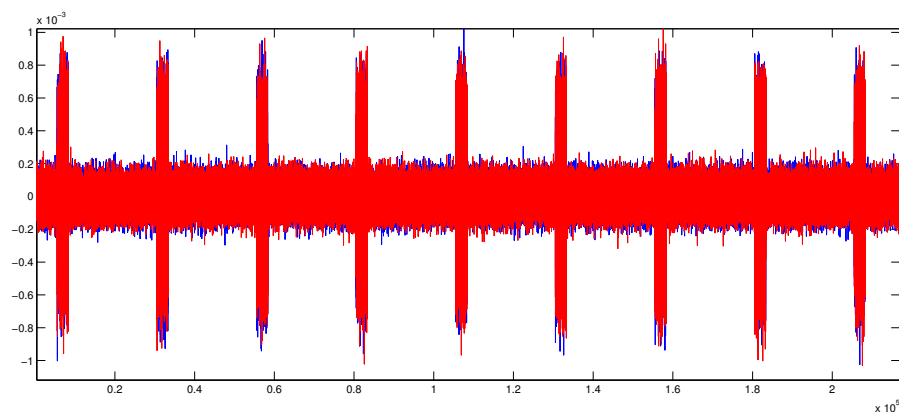


Abbildung 5.13: Empfangenes Zeitsignal bei einer Unterwasserübertragung in der Schwentimündung bei ca. 20 cm Abstand

Bereits im Zeitsignal wird ersichtlich, dass der Signalpegel klar aus dem Rauschen heraussteht. Das aus den Daten ermittelte SNR betrug 12 dB, was, wie erwartet, auf einen deutlich geringeren Salzgehalt als $4 \frac{\text{S}}{\text{m}}$ hinweist.

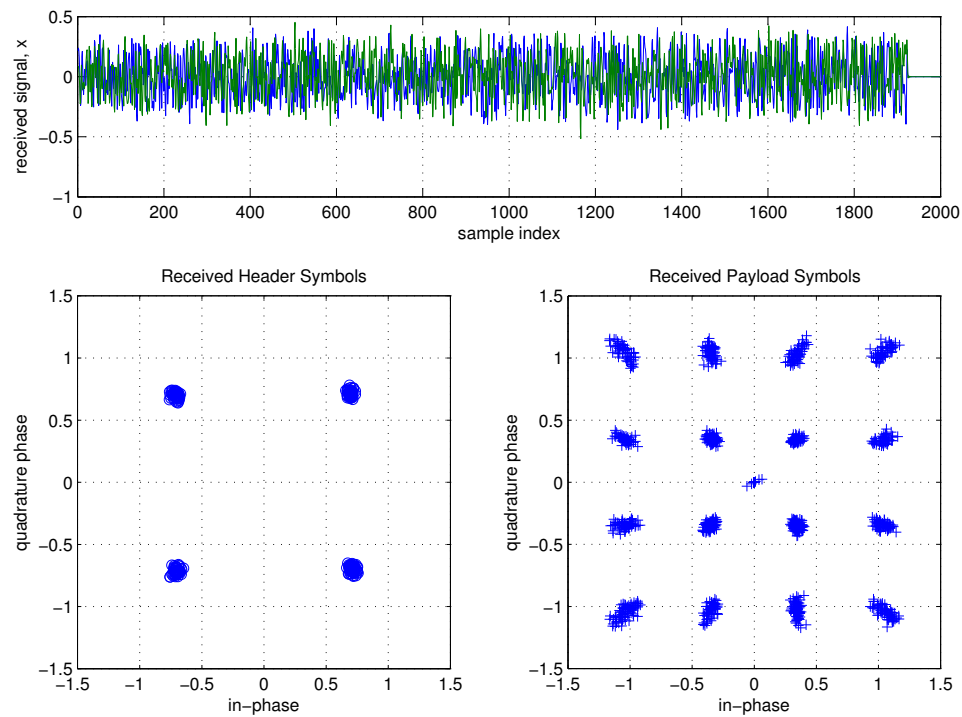


Abbildung 5.14: Modulationsdiagramm bei einer Übertragung an Luft in ca. 30 cm Abstand

Die Übertragung an Luft zeigt, dass das System grundsätzlich gute Ergebnisse liefern kann. Es lag hier nach Ergebnis der Kanalkodierung kein fehlerhaft zugeordnetes Symbol vor, so dass das Modulationsverfahren sogar noch gesteigert werden könnte. Die als 0-Symbole gekennzeichneten Elemente im Payloadbereich entstehen, da die Library byteweise arbeitet und *Zero-Padding* betreibt, weshalb bei diesen nicht durch acht teilbaren Kanalkodierungslängen zu Nullsymbolen führt.

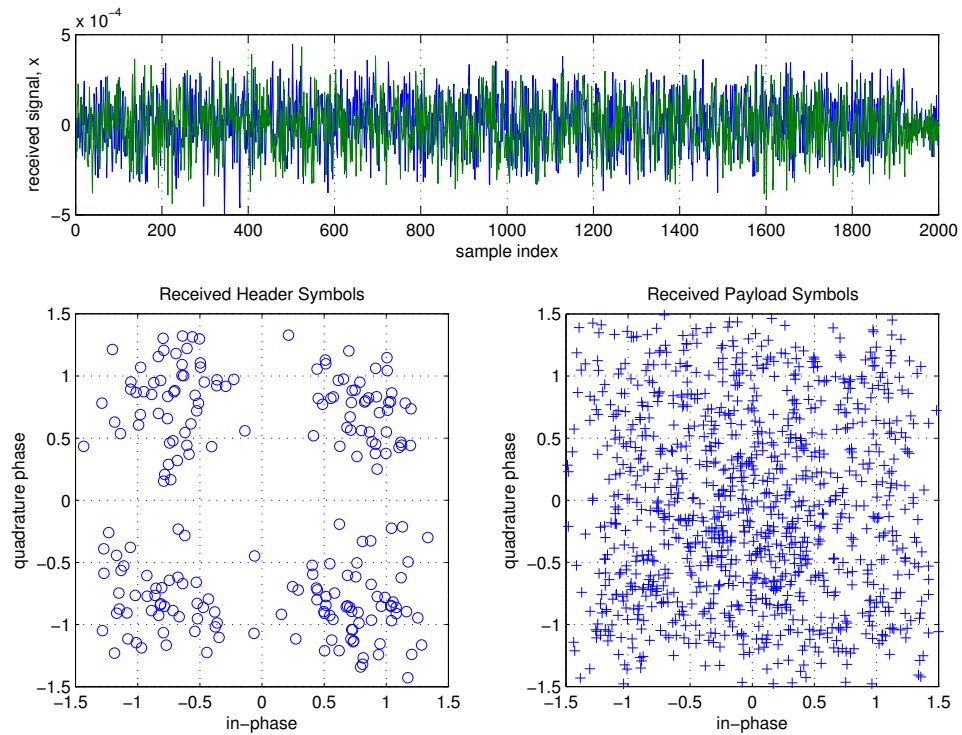


Abbildung 5.15: Modulationsdiagramm bei einer Übertragung unter Wasser in ca. 40 cm Abstand

Diese Messung zeigte nicht rekonstruierbare Fehler im Payload, der Blick auf das Modulationsdiagramm zeigt warum. Der Header konnte dank seines geringeren Modulationsschema noch erfolgreich demoduliert werden, die Payloaddaten bereits nicht mehr.

6 Zusammenfassung und Ausblick

6.1 Zusammenfassung

Das Ziel dieser Arbeit war es, ein induktives Daten- und Energieübertragungssystem für den Einsatz unter Wasser zu entwickeln. Aufgrund der hohen Dämpfung elektromagnetischer Wellen bei hohen Frequenzen wurden Analysen zum Übertragungskanal durchgeführt. Basierend auf diesen Übertragungskanaleigenschaften wurden verschiedene Lösungsansätze für die Datenübertragung entwickelt und getestet. Ein Ansatz mit größtmöglicher Flexibilität bietet ein Software Defined Radio (SDR) mit einem flexiblen System und schnellen AD- und DA-Wandlern.

Gewählt wurde daher ein kombinierter ARM/FPGA-Prozessor (Field Programmable Gate Array). Das bereits auf dem Markt verfügbare System RedPitaya übertraf die geforderten Eigenschaften.

Die Energieübertragung unter Wasser stellt ebenso wie die Datenübertragung eine große Herausforderung dar. Wasserdichte Steckverbindungen sind schwierig auf Dauer dicht zu halten, drahtlose Übertragungsverfahren sind bisher nicht entwickelt oder gar standardisiert. Der Schwerpunkt der Arbeit lag auf der Entwicklung einer neuen induktiven Datenübertragung und darauf aufbauend die Übertragung von Energie. Unter Verwendung moderner Bibliotheken der Signalverarbeitung wurden gängige Modulationsverfahren wie z.B. QAM und OFDM implementiert.

Für die Energieübertragung wurde der Qi-Standard für Überwasser-Energieübertragung gewählt. Dieser Standard wird von vielen Komponentenherstellern unterstützt und auf absehbare Zeit auch für höhere Leistungen, als die derzeit angestrebten 15 W, spezifiziert.

Nach dem Aufbau des Demonstrators wurden Test im Labor und an der Kieler Förde erfolgreich durchgeführt.

6.2 Ausblick

Das System mit seinem modularen Platinendesign und der konsequenten Umsetzung des SDR-Prinzips ist ein guter Ausgangspunkt für weitere Optimierungen im cModem. Es kann die Anzahl der genutzten Träger sowie die Anpassung von Trägerfrequenz und Bandbreite ohne Softwareänderung mittels einer Konfigurationsdatei, Modulationsverfahren, Kanalkodierung und Subträgerzuordnung mit kleinen Änderungen in der Software auf dem ARM-Prozessor angepasst werden. Selbst für die Auslagerung von Berechnungen in den FPGA

stehen noch Ressourcen zur Verfügung. Die Erweiterung des *Qi*-Standards mit einer Ausgangsleistung bis 200 W kann den Teil der Energieübertragung für AUVs attraktiver machen.

Literaturverzeichnis

- [AD8337, 2017] AD8337 (2017). *General-Purpose, Low Cost, DC-Coupled VGA*. Analog Devices. Rev. D.
- [AMBA, 2017] AMBA (2017). ARM AMBA Specification. <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ih0022d/index.html>. Version vom: 06.07.2017.
- [BNAFR, 2016] BNAFR (2016). *FREQUENZPLAN gemäß § 54 TKG über die Aufteilung des Frequenzbereichs von 0 kHz bis 3000 GHz auf die Frequenznutzungen sowie über die Festlegungen für diese Frequenznutzungen*. Bundesnetzagentur.
- [Butler, 1987] Butler, L. (1987). Underwater Radio Communication. *Amateur Radio*.
- [DMADrivers, 2017] DMADrivers (2017). SoftIP DMA'S Linux driver for Microblaze and Zynq and Zynq Ultrascale+ MPSoC. <http://www.wiki.xilinx.com/DMA+Drivers+-+Soft+IPs>. Version vom: 06.07.2017.
- [DS1210, 1996] DS1210 (1996). *1.1A, 35MHz Current Feedback Amplifier*. Linear Technology. Rev. B.
- [DS2145, 2012] DS2145 (2012). *14-Bit, 125Mbps Low Power Dual ADC*. Linear Technology. Rev. A.
- [Gaeddert, 2014] Gaeddert (2014). *User's Manual for Version 1.2.0*. Liquid-DSP. Rev. 1.2.
- [Gosling, 1998] Gosling, W. (1998). *Radio Antennas and Propagation*. Newnes, London, 1. Aufl. edition.
- [HMC784, 2017] HMC784 (2017). *GaAs MMIC 10 WATT T/R SWITCH DC - 4 GHz*. Analog Devices. Rev. 00.0808.
- [Menge, 2005] Menge, M. (2005). *Moderne Prozessorarchitekturen: Prinzipien und ihre Realisierungen (German Edition)*. Springer, 2005 edition.
- [PG021, 2016] PG021 (2016). *PG021: LogiCORE IP Product Guide*. Xilinx. Rev. 7.1.
- [PG140, 2016] PG140 (2016). *CIC Compiler v4.0 LogiCORE IP Product Guide*. Xilinx. Rev. v4.0.
- [Proakis, 2001] Proakis, J. G. (2001). *Digital Communications*. McGraw-Hill, New York.
- [QIWP, 2016] QIWP (2016). *The Qi Wireless Power Transfer System Power Class 0 Specification; Parts 1 and 2: Interface Definitions*. Wireless Power Consortium. Rev. 1.2.2.
- [RPHD, 2017] RPHD (2017). Red Pitaya Hardware Documentation. <http://redpitaya.readthedocs.io/en/latest/doc/developerGuide/125-14/fastIO.html>. Version vom: 05.07.2017.
- [SIL1A72, 2015] SIL1A72 (2015). *Single-In-Line Reed Relays*. MEDER electronic. Rev. 1.
- [UG761, 2011] UG761 (2011). *UG761: AXI Reference Guide*. Xilinx. Rev. 13.1.
- [UG821, 2015] UG821 (2015). *UG821: Zynq-7000 All Programmable SoC Software Developers Guide*. Xilinx. Rev. 12.0.

- [XAPP1113, 2008] XAPP1113 (2008). *Designing Efficient Digital Up and Down Converters for Narrowband Systems*. Xilinx. Rev. v1.0.

Abbildungsverzeichnis

1.1	cModem1	5
1.2	cModem2	6
3.1	Wellenlänge in Salzwasser bei $\sigma = 4 \frac{\text{S}}{\text{m}}$	12
3.2	Überblick über die OFDM-Kommunikation	16
3.3	Lesende und schreibende AXI-Full Kommunikation; Quelle: [UG761, 2011] . .	19
3.4	Amplitudengang des <i>Redpitaya-Board</i> -DAC [gewählter Arbeitsbereich in rot eingetragen]; Quelle: [RPHD, 2017]	20
3.5	Beispielhafte Aufteilung des Arbeitsspeichers	24
4.1	Schematische Darstellung des erstellten Übertragungssystems (Kommunikationsteil)	28
4.2	Felddarstellung der Kommunikationsspule	30
4.3	Feldstärke in Abhängigkeit der Entfernung auf der Spulenachse der Kommunikationsspule	31
4.4	Felddarstellung der Leistungsspule	32
4.5	Feldstärke in Abhängigkeit der Entfernung auf der Spulenachse der Leistungsspule	32
4.6	Verstärkerschaltungen zur Steigerung der Ausgangsleistung	34
4.7	berechnetes Bode-Diagramm des PWM-Filters der Empfangsverstärkungseinstellung	36
4.8	Einstellbarer Empfangsverstärker samt Differenzverstärker mit Eingangstiefpass	36
4.9	Prinzipaufbau Sende- und Empfangsumschaltung mit Schaltdioden	40
4.10	Sende-Empfangs-Umschalter mit Steuertransistoren	41
4.11	Spannungsregulierung durch Schaltregler	42
4.12	Prinzip des IQ-Modulators	43
4.13	Prinzip des IQ-Demodulators	44
4.14	Sequenzdiagramm des Protokolls	47
4.15	Aufbau der FlexFrame-Struktur	47
4.16	Aufbau der OFDMFlexFrame-Struktur	48
4.17	Adressierung des Arbeitsspeichers	49
4.18	Rendering des Qi-Senderboards	51
4.19	Rendering des Qi-Empfängerboards	52
5.1	Vorder- und Rückseite des <i>TransmissionBoards</i> mit und ohne aufgestecktes <i>PowerBoard</i>	53
5.2	Impedanzverlauf der unangepassten Spule	54
5.3	Amplituden- und Phasengang der Kommunikationsspulen (Abstand: 15 cm) .	55

5.4	Rendering der Unterseite des <i>TransmissionBoards</i>	56
5.5	Amplituden- und Phasengang des Eingangsverstärkers	57
5.6	Verstärkungsmessung des Eingangsverstärkers	58
5.7	Umschaltmoment des Sende-Empfangsumschalters	59
5.8	Amplituden- und Phasengang des gesamten Übertragungsweges (Abstand: 15 cm)	60
5.9	Übersicht des Blockdesigns der Empfangsseite des FPGA	61
5.10	Darstellung des IQ-Demodulators, der Signalweg des Empfangssignals ist in grün hervorgehoben.	61
5.11	Übersicht des Blockdesigns der Sendeseite des FPGA	63
5.12	Darstellung des IQ-Modulators, der Signalweg des Empfangssignals ist in grün hervorgehoben.	63
5.13	Empfangenes Zeitsignal bei einer Unterwasserübertragung in der Schwentine- mündung bei ca. 20 cm Abstand	64
5.14	Modulationsdiagramm bei einer Übertragung an Luft in ca. 30 cm Abstand .	65
5.15	Modulationsdiagramm bei einer Übertragung unter Wasser in ca. 40 cm Abstand	66

