

Demonstrator für hochratige RFID- und NFC-Systeme

MICHAEL ROLAND, BSc

DIPLOMARBEIT

eingereicht am
Fachhochschul-Masterstudiengang
EMBEDDED SYSTEMS DESIGN
in Hagenberg

im Juni 2009

© Copyright 2009 Michael Roland
Alle Rechte vorbehalten

Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und die aus anderen Quellen entnommenen Stellen als solche gekennzeichnet habe.

Hagenberg, am 19. Juni 2009

Michael Roland, BSc

Inhaltsverzeichnis

Erklärung	iii
Vorwort	vii
Kurzfassung	viii
Abstract	ix
1 Einleitung	1
1.1 Zielsetzung	1
1.2 Gliederung	2
2 Radio Frequency Identification (RFID)	3
2.1 Prinzipieller Aufbau eines RFID-Systems	3
2.1.1 Leseinheit	4
2.1.2 Transponder	4
2.2 Varianten von RFID-Systemen	4
2.3 Induktiv gekoppelte Systeme	5
2.3.1 Physikalische Grundlagen	6
2.3.2 Energieversorgung	13
2.3.3 Datenübertragung	13
2.4 Eigenschaften aktueller Systeme	18
2.4.1 Proximity cards	18
2.4.2 FeliCa	20
2.4.3 Vicinity cards	20
2.4.4 Near Field Communication	21
3 Grundlagen für höhere Datenraten	23
3.1 Einsatzgebiete für RFID- und NFC-Systeme mit höheren Datenraten	23
3.1.1 Beispiel: ePass	24
3.1.2 Beispiel: NFC	25
3.2 Systemkonzepte	26
3.2.1 System für höhere Datenraten	27

3.2.2	Parameter für höhere Datenraten	28
3.2.3	Transponderkonzept für höhere Datenraten	31
3.2.4	Filterung	32
3.3	Prototypenentwicklung	32
3.3.1	Lesegerät	32
3.3.2	Transponder	33
3.3.3	Probleme und Einschränkungen	34
4	System zur Nutzdatenübertragung im Downlink	38
4.1	Überblick über die Systemkomponenten	38
4.2	PC-Anbindung	39
4.2.1	Anforderungen	39
4.2.2	Auswahl	40
4.2.3	Schaltungsentwurf	41
4.2.4	Implementierung für das FPGA	41
4.2.5	Implementierung für den PC	45
4.2.6	Performance	45
4.3	Transponder	47
4.3.1	Anforderungen	47
4.3.2	Auswahl der Komponenten	47
4.3.3	Schaltungsentwurf	49
4.3.4	Implementierung für das CPLD	49
4.3.5	Analyse des gesendeten Signals	51
4.4	Lesegerät	52
4.4.1	Implementierung für das FPGA	53
4.4.2	Implementierung für den PC	59
4.5	Programmiergerät für den Transponder	59
4.5.1	Ablauf des Programmiervorgangs	60
4.5.2	Implementierung für das FPGA	61
4.5.3	Implementierung für den PC	62
5	Protokoll zur Nutzdatenübertragung im Downlink	63
5.1	Auswahlkriterien	63
5.1.1	Limitierende Faktoren	64
5.1.2	Anforderungen	65
5.2	Protokollentwurf	68
5.2.1	Zugriffskontrollschicht	69
5.2.2	Datensicherungsschicht	72
5.3	Implementierung	75
5.3.1	Sender	75
5.3.2	Empfänger	76
5.3.3	Integration in den Transponder und das Lesegerät	78
5.4	Performance	79
5.4.1	Zugriffskontrollschicht	80

Inhaltsverzeichnis	vi
5.4.2 Datensicherungsschicht	81
6 Demonstrator: Bildübertragung	84
6.1 Gesamtsystem	84
6.2 Ergebnisse der Bildübertragung	86
7 Zusammenfassung und Ausblick	88
7.1 Ausblick	88
7.1.1 Passiver Transponder	88
7.1.2 Stabiler Zustand des CPLD	89
7.1.3 Aufbereitung des Empfangssignals	89
7.1.4 Uplinkkanal	90
Abkürzungsverzeichnis	91
Symbolverzeichnis	93
Literaturverzeichnis	94

Vorwort

Diese Diplomarbeit entstand im Rahmen meines Studiums Embedded Systems Design an der Fachhochschule Oberösterreich, Campus Hagenberg, Fakultät für Informatik/Kommunikation/Medien. Die Thematik dieser Arbeit ist Teil des prämierten FIT-IT-Forschungsprojekts VHD (Very High Data-rate) – High Speed Air-Interface and IC Architecture for Contactless Smartcards and NFC – einer Kooperation zwischen NXP Semiconductors, der Technischen Universität Graz, dem NFC Research Lab der Fachhochschule Oberösterreich und der Universität Klagenfurt.

Zuerst möchte ich mich bei Dipl.-Ing. Dr. Harald Witschnig und den Kollegen bei NXP Semiconductors für ihre kompetente fachliche Betreuung bedanken. Zudem danke ich Prof. (FH) Dipl.-Ing. Mag. Dr. Josef Langer und Dipl.-Ing. (FH) Christian Saminger, die mich von Seiten der Fachhochschule Oberösterreich betreut haben.

Großer Dank gilt meinen Eltern, die mich und meine Ausbildung in jeglicher Weise förderten. Weiters bedanke ich mich bei meinen Freunden für ihre Unterstützung und ihr Verständnis während meines Studiums und vor allem während ich meine Diplomarbeit verfasste.

Kurzfassung

RFID (Radio Frequency Identification) und NFC (Near Field Communication) sind kontaktlose Übertragungstechnologien zur Kommunikation mit Chipkarten und mobilen Geräten. Chipkarten, NFC-Geräte und deren Anwendungen unterliegen einer stetigen Weiterentwicklung. Diese neuen Technologien und Anwendungsmöglichkeiten stoßen jedoch durch die geringe Übertragungsgeschwindigkeit der momentan standardisierten RFID-Technologie rasch an ihre Grenzen. Die maximal mögliche Übertragungsrate liegt bei 848 kBit/s. Um diesen limitierenden Faktor zu eliminieren, beschäftigt sich das FIT-IT-Forschungsprojekt VHD (Very High Datarate) – High Speed Air-Interface and IC Architecture for Contactless Smartcards and NFC – mit der Implementierung höherer Datenraten für RFID- und NFC-Systeme.

Diese Diplomarbeit behandelt die Entwicklung einer „Demonstratorplattform“, welche die Datenübertragung mit einer Übertragungsgeschwindigkeit von bis zu 6,78 MBit/s zeigt. Ausgehend von einer Analyse der bestehenden Konzepte und Prototypen wird eine entsprechende Demonstratorhardware konstruiert. Aufbauend auf dieser Hardware wird eine vollständige Datenübertragungsstrecke implementiert: Mit einem PC (Personal Computer) werden Daten über eine kontaktbehaftete Schnittstelle auf eine Chipkarte gespeichert. Diese Chipkarte überträgt die Daten über die kontaktlose VHD-Schnittstelle auf einen weiteren PC.

Die kontaktlose Übertragungsstrecke bestimmt die Anforderungen an ein passendes Übertragungsprotokoll. Diese Anforderungen werden zusammengefasst und für die Implementierung des Protokolls herangezogen. Abschließend wird die Funktionsfähigkeit des Demonstratorsystems anhand einer Beispielanwendung gezeigt. Diese Anwendung demonstriert die Übertragung einer digitalen Fotografie über die VHD-Schnittstelle.

Abstract

RFID (Radio Frequency Identification) and NFC (Near Field Communication) are wireless data transmission technologies. They are used for the communication with smart cards and mobile devices. Smart cards, NFC devices and their applications are subject to continuous development. The improvement of these technologies and the development of new applications reach a limit defined by the low transmission speed of the currently standardized RFID technology. Current RFID systems have a maximum data rate of 848 kbps. The FIT-IT research project VHD (Very High Datarate) – High Speed Air-Interface and IC Architecture for Contactless Smartcards and NFC – is devoted to eliminating this restraining factor by implementing higher data rates for RFID and NFC systems.

This diploma thesis deals with the development of a “demonstrator” platform, that demonstrates the data transmission with a data rate of up to 6.78 Mbps. Based on an analysis of current concepts and prototypes, new demonstrator hardware is built. A whole data transmission link is implemented on top of this hardware: A PC (personal computer) writes data through a wire-based interface onto a transponder (“chip card”). This transponder transmits the data across the wireless VHD interface to another PC.

The wireless link defines the requirements for an appropriate communication protocol. These requirements are summarized and are, then, used to implement that protocol. Finally, a sample application scenario proves the operability of the demonstrator system. This application scenario demonstrates the transmission of a digital photo across the VHD link.

Kapitel 1

Einleitung

RFID, *Radio Frequency Identification*, ist eine kontaktlose Übertragungstechnologie. Ursprünglich wurde sie für einfache Identifikationsanwendungen eingesetzt. Heute besteht ein zunehmender Trend, kontaktbehaftete Chipkartensysteme durch diese kontaktlose Technologie zu ergänzen oder gar zu ersetzen. Chipkarten, sogenannte Smartcards, können einfache Speicherkarten oder komplexe Prozessorkarten mit aufwändigen Betriebssystemen sein. Smartcards gewinnen zunehmend an Bedeutung. Beispiele für die vielfältigen Anwendungen sind Telefonwertkarten, Mitarbeiterkarten, SIM-Karten (*Subscriber Identity Module*, das ist jene Karte, die ein Mobiltelefon seinem Besitzer bzw. dessen Mobiltelefonanschluss zuordnet) und Bankomat- und Kreditkarten. Chipkarten haben jedoch nicht immer die Form einer gewöhnlichen Kunststoffkarte, denn z. B. auch viele moderne Reisepässe zählen zu den Smartcard-Anwendungen. Durch NFC, *Near Field Communication*, eine Weiterentwicklung der RFID-Technologie, ist diese kontaktlose Übertragungstechnologie nicht nur auf Chipkarten beschränkt. Kontaktlose Mikrochips können in Alltagsgegenstände oder Werbeplakate integriert sein. Mobiltelefone können sogar sowohl die Funktionen einer kontaktlosen Smartcard als auch die eines entsprechenden Lesegerätes übernehmen.

1.1 Zielsetzung

Durch die zunehmende Weiterentwicklung der Smartcard-Mikrochips, welche vor allem durch die Neuerungen am Sektor der On-Chip-Speichertechnologien vorangetrieben wird [26], steigen die Speicherkapazität und die Rechengeschwindigkeit kontaktloser Chipkarten stark an. Während die Zugriffszeit bei früheren Systemen maßgeblich durch die Berechnungsgeschwindigkeit und die Speicherzugriffszeiten beeinflusst wurde, hängt diese bei aktuellen und zukünftigen Systemen vor allem von der Übertragungsgeschwindigkeit ab.

Diese Arbeit entstand im Rahmen des Forschungsprojekts VHD (*Very*

High Datarate) – High Speed Air-Interface and IC Architecture for Contactless Smartcards and NFC. Dieses Forschungsprojekt beschäftigt sich damit, die kontaktlose Smartcard-Schnittstelle an diese Trends anzupassen. Durch höhere Datenraten soll der durch die derzeit standardisierten Übertragungsgeschwindigkeiten entstehende Engpass ausgeglichen werden.

Ziel dieser Diplomarbeit ist die prototypische Umsetzung neu erforschter Übertragungskonzepte. Während bisher bestehende Prototypen nur die prinzipielle Funktionsfähigkeit dieser Konzepte bestätigen, soll ein neuer Demonstrator die Übertragung einer vollständigen Datei zeigen. Der fertige Demonstrator hat die Aufgabe die Übertragung einer Bilddatei von einem PC auf eine kontaktlose Chipkarte und damit weiter auf einen anderen PC zu ermöglichen.

1.2 Gliederung

In einem ersten Schritt erörtert diese Diplomarbeit die Ursprünge, die Aufgaben und die Funktionsweise von RFID-Systemen (Kap. 2). In Kapitel 3 werden anschließend die Ziele und die möglichen Aufgaben für RFID- und NFC-Systeme mit höheren Datenraten anhand von Beispielen erläutert. Darüber hinaus gibt dieses Kapitel einen Überblick über die, in vorhergehenden Diplomarbeiten entwickelten, Systemkonzepte und über den Status der Prototypenentwicklung zum Thema VHD. In zwei weiteren Schritten wird der Demonstrator entwickelt. Kapitel 4 beschreibt dazu zunächst ein weiterentwickeltes Systemkonzept. Kapitel 5 vervollständigt den Demonstrator anschließend durch ein geeignetes Übertragungsprotokoll. Abschließend wird der vollständige Demonstrator zur Bildübertragung dokumentiert (Kap. 6) und ein Ausblick auf weiterführende Ergänzungen gegeben (Kap. 7).

Kapitel 2

Radio Frequency Identification (RFID)

Dieses Kapitel gibt einen Überblick über die Grundlagen und Anwendungen von RFID-Systemen.

Als RFID, *Radio Frequency Identification*, bezeichnet man kontaktlose Identifikationssysteme. Dabei werden Daten zwischen einem kontaktlosen Datenträger (*Transponder*) und einer Schreib-/Leseinheit (im Weiteren einfach als Leseinheit, engl. *Reader*, bezeichnet) übertragen.

Eine der ersten kommerziellen Anwendungen von RFID ist die, bereits in den 1960er Jahren eingesetzte, elektronische Diebstahlsicherung in Kaufhäusern [29]. In den darauf folgenden zwei Jahrzehnten wurden die Anwendungsbereiche Transport und Logistik, Zutrittskontrolle und Tieridentifikation erschlossen [29].

Heute hat RFID ein umfangreiches Anwendungsgebiet, das weit über einfache Identifikationsaufgaben hinausreicht. So wird diese Technologie z. B. für elektronische Bezahlssysteme und elektronische Reisepässe eingesetzt.

Durch die kontaktlose Schnittstelle tritt kein mechanischer Verschleiß auf. Weiters sind RFID-Transponder praktisch nicht anfällig gegen Verschmutzung. Deshalb stellt diese Technologie eine vorteilhafte Alternative zur kontaktbehafteten Chipkarte dar. Der Funktionsumfang von RFID-Transpondern und RFID-Chipkarten reicht von einfachen Identifikationstoken, über wiederbeschreibbare Datenspeicher, bis hin zu Prozessorkarten mit Speicherplatz für mehrere Anwendungsprogramme.

2.1 Prinzipieller Aufbau eines RFID-Systems

Abbildung 2.1 zeigt den grundlegenden Aufbau eines RFID-Systems. Dieses besteht immer aus den zwei Komponenten Leseinheit und Transponder [4].

2. Radio Frequency Identification (RFID)

4

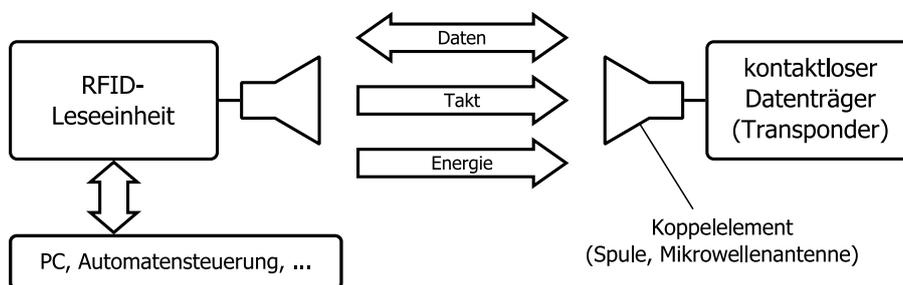


Abbildung 2.1: Grundbestandteile jedes RFID-Systems [4]

2.1.1 Leseinheit

Das Lesegerät bildet die Brücke zwischen der kontaktlosen Schnittstelle und einem weiteren System (z. B. PC, Automatensteuerung, Zutrittskontrollsystem). Es besteht typischerweise aus einer Treiber- und Modulationseinheit, einer Demodulationseinheit, einer Kontrolleinheit und einem Koppellement [4, 29]. Die Treiber- und Modulationseinheit versorgt den Transponder mit Energie, Takt und Daten. Die Demodulationseinheit empfängt die vom Transponder zurückgesendeten Daten. Die Kontrolleinheit steuert die Kommunikation mit dem Transponder. Das Koppellement ist je nach Variante des RFID-Systems eine Spule, eine Mikrowellenantenne oder eine Elektrodenfläche.

2.1.2 Transponder

Der Transponder besteht üblicherweise aus einem Koppellement und einer integrierten Schaltung. Die integrierte Schaltung enthält die analoge RFID-Schnittstelle, eine Kontrolleinheit zur Verarbeitung von Befehlen und einen Speicher [4, 29]. Die Kontrolleinheit variiert dabei vom einfachen Zustandsautomaten bis hin zum *Java Card* basierten Prozessor.

2.2 Varianten von RFID-Systemen

Es gibt zahlreiche Variationen von RFID-Systemen. Die wesentlichen Unterscheidungsmerkmale sind dabei das physikalische Übertragungsprinzip, die Betriebsfrequenz und die Reichweite. Nach dem Übertragungsprinzip lassen sich, nach [29], drei Gruppen unterscheiden:

- *Kapazitiv gekoppelte Systeme* nutzen ein starkes, hochfrequentes elektrisches Feld zur Energie- und Datenübertragung zwischen dem Lesegerät und dem Transponder. Die Funktionsfähigkeit ist jedoch stark orientierungs- und umgebungsabhängig. Weiters ist die Reichweite auf

2. Radio Frequency Identification (RFID)

5

das Nahfeld beschränkt. Dadurch finden sich für diese Systeme nur wenige Einsatzmöglichkeiten.

- *UHF-Backscatter-Systeme* nutzen die elektromagnetische Wellenausbreitung zur Kommunikation zwischen dem Lesegerät und dem Transponder. Ähnlich der Radartechnik, reflektiert der Transponder zur Datenübertragung Energie zurück zur Leseinheit (*Backscattering*). Die Betriebsfrequenzen liegen im UHF-Frequenzband (*Ultra High Frequency*, 300 MHz bis 3 GHz). UHF-Systeme arbeiten im Fernfeld und haben üblicherweise eine Reichweite von mehreren Metern.
- *Induktiv gekoppelte Systeme* nutzen, analog zum Prinzip des Transformators, das Induktionsgesetz zur Energie- und Datenübertragung zwischen dem Lesegerät und dem Transponder. Es werden vorwiegend Betriebsfrequenzen im Bereich von 125 kHz und 13,56 MHz eingesetzt. Nachdem das magnetische Wechselfeld zur Kommunikation verwendet wird, arbeiten diese Systeme im Nahfeld.

Heute werden vorwiegend induktiv gekoppelte Systeme und UHF-Backscatter-Systeme eingesetzt, wobei in dieser Diplomarbeit nur induktiv gekoppelte Systeme mit einer Betriebsfrequenz von 13,56 MHz behandelt werden.

Eine weitere Einteilung lässt sich anhand der Energieversorgung des Transponders treffen. Der Transponder kann passiv, semi-passiv oder aktiv sein [4].

- Passive Transponder beziehen ihre gesamte Versorgungsenergie aus dem Feld des Lesegerätes.
- Semi-passive (bzw. Batterie gestützte) Transponder besitzen eine eigene Energieversorgung. Sie sind jedoch nicht in der Lage ein Hochfrequenzsignal zu erzeugen und benötigen deshalb zur Kommunikation das Feld des Lesegerätes.
- Aktive Transponder besitzen eine eigene Energieversorgung und erzeugen ihr eigenes Hochfrequenzsendesignal. Sie sind daher unabhängig vom Feld des Lesegerätes. Man bezeichnet sie auch als *Short Range Radio Devices*.

2.3 Induktiv gekoppelte Systeme

Als induktiv gekoppelte Systeme bezeichnet man jene RFID-Systeme, die das magnetische Nahfeld stromdurchflossener Leiterspulen zur Kopplung von Leseinheit und Transponder verwenden.

2.3.1 Physikalische Grundlagen

Zur Betrachtung induktiv gekoppelter Systeme sind einige physikalische Grundkenntnisse notwendig. Diese werden in den nachfolgenden Abschnitten kurz beschrieben. Als Grundlage dafür wurden [4, 11, 28] verwendet.

Magnetisches Feld

Jeder Stromfluss, d. h. jedes Bewegen einer elektrischen Ladung, bewirkt ein magnetisches Feld [4]. Das magnetische Feld ist ein Vektorfeld. Das bedeutet, dass jedem Raumpunkt ein Vektor bzw. ein Betrag und eine Richtung zugeordnet werden. Magnetische Felder lassen sich daher mit Hilfe von Feldlinien darstellen [28]. Abbildung 2.2 zeigt die magnetischen Feldlinien um einen stromdurchflossenen Leiter. Das magnetische Feld wird durch die

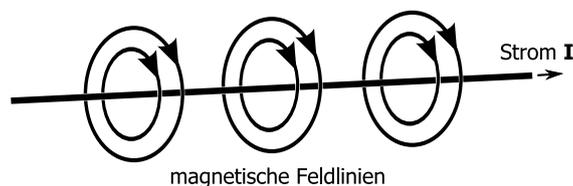


Abbildung 2.2: Magn. Feldlinien um einen stromdurchflossenen Leiter [4, 28]

materialunabhängige magnetische Feldstärke \mathbf{H} beschrieben. Diese wird in Ampere pro Meter [A/m] angegeben.

Magnetische Spannung

Die magnetische Spannung U_m ist die Summe der Stromstärken I_n aller Ströme durch eine von einem Weg \mathbf{s} umschlossene Fläche. Sie ist gleich dem Kurvenintegral der magnetischen Feldstärke \mathbf{H} entlang dieses geschlossenen Weges \mathbf{s} [28],

$$U_m = \sum_n I_n = \oint_{\mathbf{s}} \mathbf{H} \, d\mathbf{s}. \quad (2.1)$$

Magnetische Feldstärke

Aus der Definition der magnetischen Spannung ergibt sich der Zusammenhang zwischen der Anordnung von stromdurchflossenen Leitern und dem sie umgebenden magnetischen Feld. So ist die Feldstärke im Abstand r eines geraden Leiters (Abb. 2.2)

$$H = \frac{I}{2\pi \cdot r}. \quad (2.2)$$

2. Radio Frequency Identification (RFID)

7

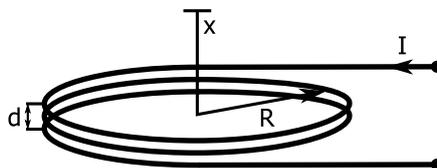


Abbildung 2.3: Stromdurchflossene Leiterspule mit $d \ll R$ [4]

Die Feldstärke im Abstand x vom Mittelpunkt einer zylindrischen Leiterspule (Abb. 2.3) mit N Windungen, der Breite d und dem Radius R ergibt sich, unter der Bedingung $d \ll R$, zu

$$H = \frac{I \cdot N \cdot R^2}{2 \cdot \sqrt{(R^2 + x^2)^3}}. \quad (2.3)$$

Abbildung 2.4 zeigt den Feldstärkeverlauf im Nahbereich verschiedener

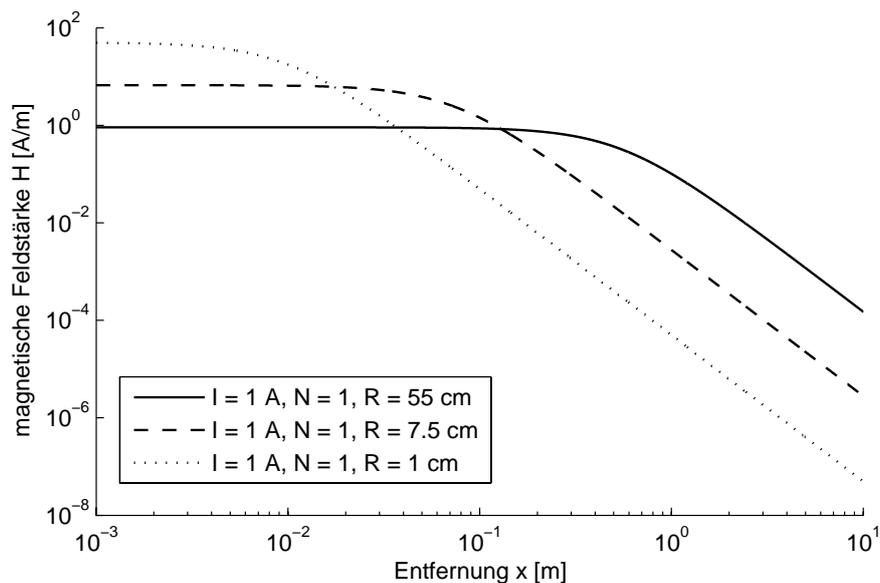


Abbildung 2.4: Magn. Feldstärke von Spulen in Abhängigkeit von Spulradius R und Abstand x zum Spulenzentrum [4]

Leiterspulen bzw. Antennen. Dabei ist erkennbar, dass Spulen mit kleinen Durchmessern in kurzen Entfernungen eine höhere Feldstärke als große Spulen aufweisen. Große Spulen liefern hingegen in großen Entfernungen eine höhere Feldstärke als kleine Spulen. Es lässt sich also für jeden Abstand x ein optimaler Antennenradius R bestimmen.

2. Radio Frequency Identification (RFID)

8

Die Feldstärke im Inneren einer langen zylindrischen Leiterspule, d. h. $d \gg R$, ergibt sich zu

$$H = \frac{I \cdot N}{d}, \quad (2.4)$$

und ist somit homogen [28].

Magnetische Flussdichte

Die magnetische Flussdichte B gibt an, welche Kraft ein Magnetfeld auf eine bewegte Ladung ausübt. Sie wird in Tesla [T] angegeben. Die Flussdichte entspricht der Feldliniendichte. Sie ist materialabhängig und proportional zur magnetischen Feldstärke:

$$\mathbf{B} = \mu_0 \cdot \mu_r \cdot \mathbf{H}. \quad (2.5)$$

Dabei ist μ_0 die magnetische Feldkonstante. Diese gibt die Permeabilität, d. h. die magnetische Leitfähigkeit, von Vakuum an. μ_r ist die relative Permeabilität des durchfluteten Materials.

Magnetischer Fluss

Der magnetische Fluss Φ entspricht der Anzahl der Feldlinien, die eine bestimmte Fläche \mathbf{A} durchsetzen. Er wird in Weber [Wb] angegeben und berechnet sich zu

$$\Phi = \int_{\mathbf{A}} \mathbf{B} \, d\mathbf{A}. \quad (2.6)$$

Induktivität

Die Flussverketzung Ψ durch eine Spule entspricht der Summe aller Teilflüsse Φ durch die einzelnen Spulenebenen:

$$\Psi = \sum_n \Phi_n. \quad (2.7)$$

Jede Spulenebene entspricht einer Windung. Strom und Fläche sind daher für jede der N Ebenen identisch:

$$\Psi = N \cdot \Phi = N \cdot B \cdot A = N \cdot \mu_0 \cdot \mu_r \cdot H \cdot A. \quad (2.8)$$

Die Induktivität L einer Spule ist das Verhältnis zwischen der Flussverketzung Ψ und der Stromstärke I durch die Spule und wird in Henry [H] angegeben:

$$L = \frac{\Psi}{I}, \quad (2.9)$$

2. Radio Frequency Identification (RFID)

9

Für eine lange zylindrische Leiterspule ergibt sich daher

$$L = \frac{N \cdot \mu_0 \cdot \mu_r \cdot H \cdot R^2 \pi}{I} \quad (2.10)$$

$$= \frac{N \cdot \mu_0 \cdot \mu_r \cdot R^2 \pi}{I} \cdot \frac{I \cdot N}{d} \quad (2.11)$$

$$= N^2 \cdot \mu_0 \cdot \mu_r \cdot \frac{R^2 \pi}{d}. \quad (2.12)$$

Gl. (2.12) zeigt, dass die Induktivität nur von der Bauform (Windungen N , Radius R und Länge d) der Spule und den Materialeigenschaften (relative Permeabilität μ_r) des durchfluteten Raumes abhängt.

Gegeninduktivität

Schiebt man zwei lange zylindrische Spulen mit unterschiedlichen Radien R_1 und R_2 und gleicher Länge d vollständig ineinander, so durchsetzt der Fluss der einen Spule die jeweils andere Spule. Es sei N_1 die Windungszahl, I_1 der Strom und R_1 der Radius der äußeren Spule und N_2 die Windungszahl, I_2 der Strom und R_2 der Radius der inneren Spule. Die gemeinsame Fläche beider Spulen ist somit $A_2 = R_2^2 \pi$. Die durch die äußere Spule verursachte Flussdichte ($I_2 = 0$) berechnet sich zu

$$B_1 = \mu_0 \cdot \mu_r \cdot \frac{N_1 \cdot I_1}{d}. \quad (2.13)$$

Der gesamte durch die äußere Spule in der inneren Spule verursachte verkettete Fluss Ψ_{21} ist daher

$$\Psi_{21} = N_2 \cdot B_1 \cdot A_2 = \mu_0 \cdot \mu_r \cdot N_1 \cdot N_2 \cdot I_1 \cdot \frac{R_2^2 \pi}{d}. \quad (2.14)$$

Die durch die innere Spule verursachte Flussdichte ($I_1 = 0$) berechnet sich zu

$$B_2 = \mu_0 \cdot \mu_r \cdot \frac{N_2 \cdot I_2}{d}. \quad (2.15)$$

Der gesamte durch die innere Spule in der äußeren Spule verursachte verkettete Fluss Ψ_{12} ist daher

$$\Psi_{12} = N_1 \cdot B_2 \cdot A_2 = \mu_0 \cdot \mu_r \cdot N_1 \cdot N_2 \cdot I_2 \cdot \frac{R_2^2 \pi}{d}. \quad (2.16)$$

Die Stromkreise der Spulen beeinflussen sich also gegenseitig über den magnetischen Fluss. Die Gegeninduktivität M_{21} der Spule 2 zur Spule 1 ist das Verhältnis zwischen dem verketteten Fluss Ψ_{21} und der Stromstärke I_1 durch Spule 1:

$$M_{21} = \frac{\Psi_{21}}{I_1} = \mu_0 \cdot \mu_r \cdot N_1 \cdot N_2 \cdot \frac{R_2^2 \pi}{d}. \quad (2.17)$$

2. Radio Frequency Identification (RFID)

10

Die Gegeninduktivität M_{12} der Spule 1 zur Spule 2 ist das Verhältnis zwischen dem verketteten Fluss Ψ_{12} und der Stromstärke I_2 durch Spule 2:

$$M_{12} = \frac{\Psi_{12}}{I_2} = \mu_0 \cdot \mu_r \cdot N_1 \cdot N_2 \cdot \frac{R_2^2 \pi}{d}. \quad (2.18)$$

Die Gegeninduktivitäten stimmen also überein: $M = M_{21} = M_{12}$.

Kopplungsfaktor

Der Kopplungsfaktor k ist nach [4] definiert als

$$k = \frac{M}{\sqrt{L_1 \cdot L_2}}. \quad (2.19)$$

Er gibt Aufschluss über die Verkopplung zweier Spulen bzw. Antennen [4]. Ist der Kopplungsfaktor $k = 0$, so sind die beiden Spulen vollständig entkoppelt. Sie beeinflussen sich daher nicht. Ein Kopplungsfaktor $k = 1$ bedeutet, dass die Spulen vom selben magnetischen Fluss durchsetzt sind [4].

Induktion

Variiert man den magnetischen Fluss durch eine Leiterspule über die Zeit, dann entsteht an den Spulenden eine Spannung. Diesen Vorgang nennt man Induktion. Die induzierte Spannung U_{ind} ist also

$$U_{ind} = -\frac{d\Psi}{dt}. \quad (2.20)$$

Die Selbstinduktion einer stromdurchflossenen Spule mit der konstanten Induktivität L ist

$$U_{ind} = -L \cdot \frac{dI}{dt}. \quad (2.21)$$

Transformator

Hat man zwei gekoppelte Spulen, dann bewirkt eine Stromänderung in einer Spule eine Flussänderung in beiden Spulen. Durch die Flussänderung wird in der zweiten Spule eine Spannung induziert. Abbildung 2.5 zeigt das Ersatzschaltbild eines Transformators, d. h. zweier gekoppelter Spulen. Aus der Definition der induzierten Spannung (Gl. (2.20)) ergeben sich folgende Zusammenhänge zwischen Strom, Spannung und Induktivität:

$$U_{i1} = L_1 \cdot \dot{I}_1 - M \cdot \dot{I}_2, \quad (2.22)$$

$$U_{i2} = -L_2 \cdot \dot{I}_2 + M \cdot \dot{I}_1, \quad (2.23)$$

$$U_2 = U_{i2} - R_2 \cdot I_2 \quad (2.24)$$

$$= -L_2 \cdot \dot{I}_2 + M \cdot \dot{I}_1 - R_2 \cdot I_2. \quad (2.25)$$

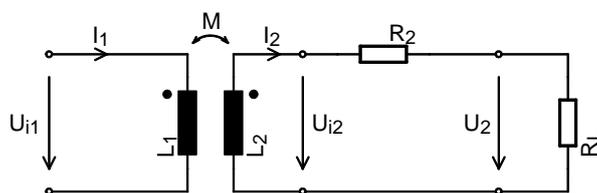


Abbildung 2.5: Ersatzschaltbild magnetisch gekoppelter Leiterschleifen [4]

Mit sinusförmigen Strömen, wie sie auch bei RFID-Systemen zum Einsatz kommen, kann die komplexe Wechselstromrechnung verwendet werden:

$$U_{i1} = j\omega L_1 \cdot I_1 - j\omega M \cdot I_2, \quad (2.26)$$

$$U_{i2} = -j\omega L_2 \cdot I_2 + j\omega M \cdot I_1, \quad (2.27)$$

Durch Umformen und Einsetzen des Kopplungsfaktors (Gl. (2.19)) ergibt sich, unter Vernachlässigung des Innenwiderstands, folgender Zusammenhang zwischen U_{i1} und U_{i2} :

$$U_{i2} = k \cdot \sqrt{\frac{L_2}{L_1}} \cdot U_{i1} + j\omega L_2 \cdot I_2 \cdot (k^2 - 1). \quad (2.28)$$

Für den Fall, dass Spule 2 unbelastet ist, $R_L \rightarrow \infty \rightsquigarrow I_2 = 0$, gilt

$$U_{i2} = k \cdot \sqrt{\frac{L_2}{L_1}} \cdot U_{i1}, \text{ bzw.} \quad (2.29)$$

$$U_{i2} = j\omega \cdot k \cdot \sqrt{L_1 \cdot L_2} \cdot I_1. \quad (2.30)$$

Die Spannungen an den Spulendenen des Transformators sind also proportional zueinander. Der Proportionalitätsfaktor ist abhängig vom Kopplungsfaktor und vom Verhältnis der Eigeninduktivitäten der beiden Spulen.

Schwingkreis

Schaltet man, wie in Abbildung 2.6, parallel zu Spule 2 eine Kapazität C_2 , dann erhält man einen Parallelschwingkreis. Der Schwingkreis hat die Re-

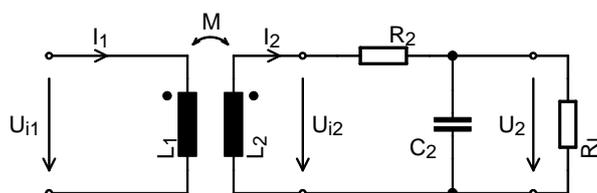


Abbildung 2.6: Ersatzschaltbild mit Parallelschwingkreis [4]

2. Radio Frequency Identification (RFID)

12

sonanzfrequenz

$$f_r = \frac{1}{2\pi \cdot \sqrt{L_2 \cdot C_2}}. \quad (2.31)$$

Durch den Schwingkreis ist der Zusammenhang aus Gl. (2.30) nicht mehr gegeben:

- Für Frequenzen $f \ll f_r$ hat die Kapazität C_2 kaum eine Auswirkung auf das Verhalten des Transformators.
- Für Frequenzen $f \gg f_r$ wirkt die Kapazität C_2 dämpfend und führt daher zu einer niedrigeren Ausgangsspannung U_2 .
- Für Frequenzen f nahe f_r ist der Schwingkreis in Resonanz. Dadurch ist die Ausgangsspannung U_2 um ein vielfaches höher, als ohne die Kapazität C_2 . Dies wird als Spannungsüberhöhung bezeichnet.

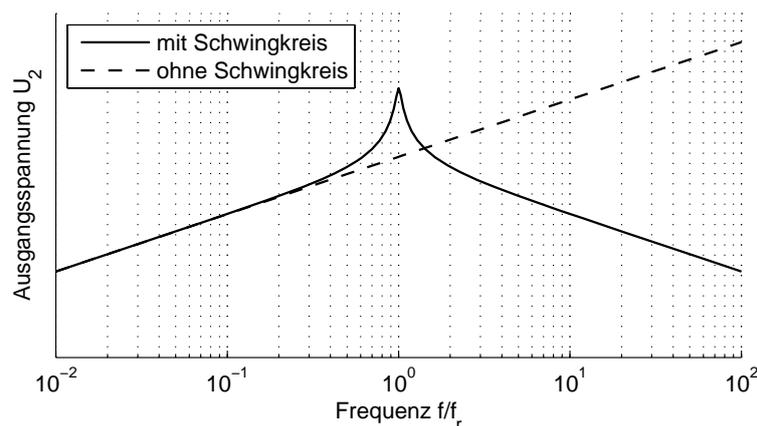


Abbildung 2.7: Spannungsüberhöhung durch Resonanz im Vergleich zur einfachen Spule ohne Schwingkreis [4]

Ein Maß für die Spannungs- und Stromüberhöhung im Schwingkreis ist der Gütefaktor Q [4]. Dieser gibt an, wie schnell die Schwingung im Schwingkreis abklingt. Ein niedriger Gütefaktor bedeutet, dass die Schwingung rasch abklingt. Daher ist auch die Spannungsüberhöhung gering. Ein hoher Gütefaktor bedeutet, dass der Schwingkreis lange nachschwingt. Die Spannungsüberhöhung ist deshalb groß.

Für Abbildung 2.6 berechnet sich der Gütefaktor zu

$$Q = \frac{1}{\frac{R_2}{\omega L_2} + \frac{\omega L_2}{R_L}}. \quad (2.32)$$

2.3.2 Energieversorgung

Induktiv gekoppelte RFID-Systeme nutzen das Prinzip des Transformators zur Energie- und Datenübertragung. Ihr prinzipieller Aufbau folgt Abbildung 2.6. Zur Energieübertragung speist die Leseinheit ihre Antenne (Spule 1) mit einem sinusförmigen Strom – dem Trägersignal. Nach Gl. (2.30) wird dadurch in der Transponderantenne (Spule 2) eine Spannung induziert. Diese Wechselspannung wird anschließend mit Hilfe eines Brückengleichrichters gleichgerichtet.

Bei RFID-Systemen treten üblicherweise Kopplungsfaktoren $k \ll 1$ auf. Aus diesem Grund ist der Schwingkreis am Transponder bzw. dessen Resonanzüberhöhung bei der Trägerfrequenz notwendig, um den Transponderchip mit einer ausreichend hohen Spannung zu versorgen. Ein hoher Gütefaktor bewirkt also eine bessere Energieversorgung des Transponders.

Gleichzeitig kann der Kopplungsfaktor eines Systems sehr stark schwanken. Dies geschieht z. B. durch Abstands- oder Lageänderung des Transponders oder durch unterschiedliche Materialien im Umfeld der Leseantenne. Daher durchstreift die Amplitude der induzierten Spannung einen relativ großen Bereich von einigen Volt bis zu einigen hundert Volt. Transponderchips arbeiten jedoch nur in einem sehr eingeschränkten Spannungsbereich. Der übliche Bereich für die Versorgungsspannung von ICs (*Integrated circuits*, engl. integrierte Schaltungen) liegt derzeit zwischen 0,8 und 5 Volt. Es ist daher nicht nur eine Spannungsüberhöhung, sondern auch eine Spannungsbegrenzung notwendig. Diese Spannungsbegrenzung kann mit Zenerdioden oder spannungsabhängigen Shuntwiderständen realisiert werden.

2.3.3 Datenübertragung

Bei der Datenübertragung zwischen Lesegerät und RFID-Transponder gibt es zwei Übertragungskanäle: Ein Kanal wird für die Datenübertragung vom Lesegerät zum Transponder und der zweite für die Datenübertragung vom Transponder zum Lesegerät verwendet.

Während bei der Energieversorgung ein hoher Gütefaktor bessere Ergebnisse erzielt, bedeutet ein hoher Gütefaktor für die Datenübertragung lange Nachschwingzeiten und damit eine geringe Bandbreite. Nach [2] gilt für den Zusammenhang zwischen Gütefaktor Q und Bandbreite B_w :

$$B_w = \frac{f_r}{Q}. \quad (2.33)$$

Für einen optimalen Gütefaktor muss also ein Kompromiss zwischen Energieversorgung und Bandbreite gefunden werden.

Modulationsverfahren

Für die Datenübertragung werden verschiedene digitale Modulationsverfahren (Abb. 2.8) verwendet:

- n -ASK (*Amplitude-shift keying*, engl. Amplitudenumtastung) bezeichnet ein n -stufiges Modulationsverfahren, bei dem die n verschiedenen Symbole als unterschiedliche Amplitudenwerte des Trägersignals dargestellt werden. Eine besondere Variante des ASK ist OOK (*On-off keying*, engl. Ein-Aus-Tastung). Dabei wird das Trägersignal an- bzw. ausgeschaltet.
- n -PSK (*Phase-shift keying*, engl. Phasenumtastung) bezeichnet ein n -stufiges Modulationsverfahren, bei dem die n verschiedenen Symbole als unterschiedliche Phasenlagen des Trägersignals dargestellt werden.
- n -FSK (*Frequency-shift keying*, engl. Frequenzumtastung) bezeichnet ein n -stufiges Modulationsverfahren, bei dem die n verschiedenen Symbole als unterschiedliche Frequenzen dargestellt werden.

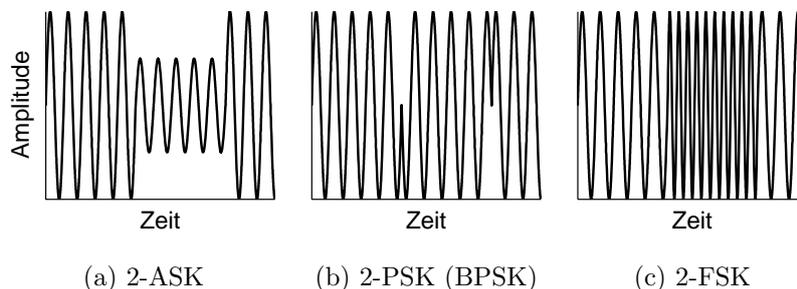


Abbildung 2.8: Digitale Modulationsverfahren

Codierungsverfahren

Um die Daten an das Modulationsverfahren und den Übertragungskanal anzupassen, werden diese codiert. Für RFID-Systeme sind die folgenden Codierungsverfahren (Abb. 2.9) besonders wichtig [4]:

- Bei der NRZ-L-Codierung (*Non-return-to-zero level*) wird jedes Datenbit als ein bestimmter Signalpegel codiert.
- Bei der unipolaren RZ-Codierung (*Return-to-zero*) wird eine logische Null als bestimmter Signalpegel codiert. Für eine logische Eins wechselt der Signalpegel für die erste Hälfte der Symboldauer.

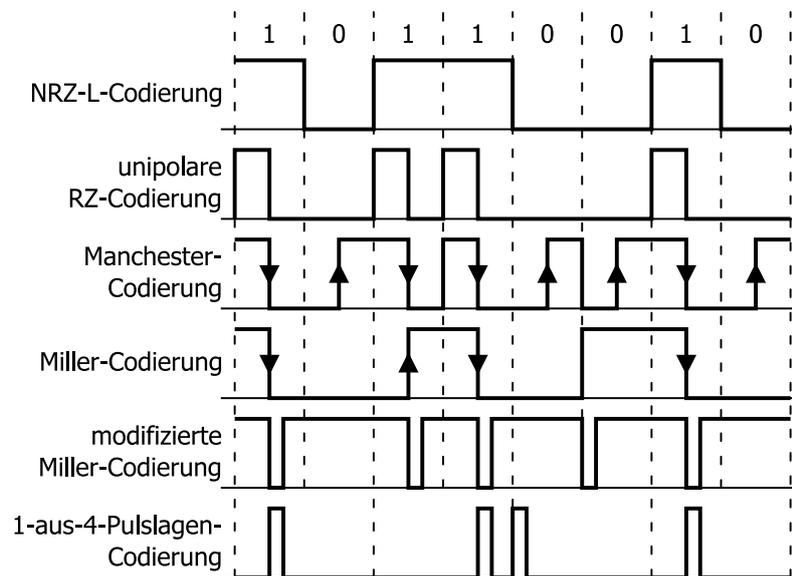


Abbildung 2.9: Codierungsverfahren [4]

- Bei der Manchestercodierung wird jedes Datenbit als eine bestimmte Signalfanke (in der Symbolmitte) codiert. Eine logische Eins wird auf den einen Pegelwechsel und eine logische Null auf den anderen Pegelwechsel abgebildet.
- Bei der Millercodierung wird eine logische Eins als Pegelwechsel (in der Symbolmitte) codiert. Bei einer logischen Null bleibt der Pegel unverändert. Bei mehreren aufeinander folgenden Nullen wird für jede folgende Null der Pegel gewechselt.
- Bei der modifizierten Millercodierung wird jede Flanke eines entsprechenden Miller-codierten Signals durch einen Puls codiert.
- Bei der Pulslagen-Codierung wird jedes Datensymbol als Position eines Pulses fester Breite innerhalb einer Symboldauer codiert. Bei einer 1-aus- N -Codierung kann der Puls also an N verschiedenen Positionen innerhalb des Symbols auftreten. Damit können N Werte (bzw. $\log_2(N)$ Bits) in einem Symbol dargestellt werden.

Datenübertragung vom Lesegerät zum Transponder

Diese Übertragungsrichtung wird auch als Uplink bezeichnet. Sie entspricht der Richtung der Energieversorgung. Es liegt also nahe, das Trägersignal, welches auch zur Energieversorgung genutzt wird, mit dem Datenstrom zu modulieren.

Für die Datenübertragung vom Lesegerät zum Transponder eignen sich verschiedene digitale Modulationsverfahren. In den handelsüblichen induktiv gekoppelten RFID-Chipkarten-Systemen werden vorwiegend ASK und PSK eingesetzt [15, 17, 21]. FSK wird hingegen in diesen Systemen nicht verwendet.

Die am häufigsten eingesetzten induktiven RFID-Chipkarten-Systeme in Europa sind Proximity- [17] und Vicinity-Cards [21]. Diese Systeme verwenden ausschließlich zweistufige ASK-Modulationsverfahren. Dabei wird, je nach System, die Amplitude entweder 100 Prozent (Abb. 2.10(a)) oder zehn Prozent (Abb. 2.10(b)) moduliert.

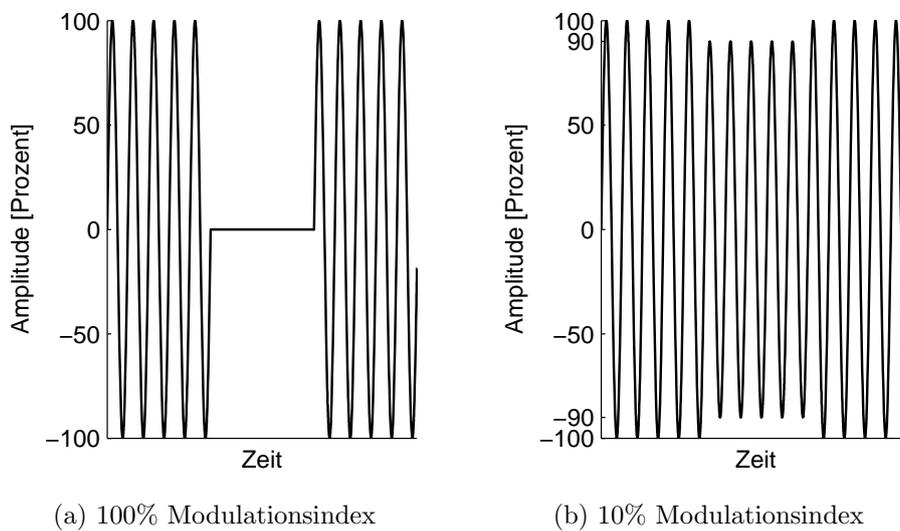


Abbildung 2.10: Zweistufige Amplitudenumtastung (2-ASK)

Datenübertragung vom Transponder zum Lesegerät

Diese Übertragungsrichtung wird auch als Downlink bezeichnet. Die Kopplung der Spulen verursacht nicht nur eine Wirkung des Lesegeräts auf den Transponder (Gl. (2.23)), sondern auch eine Wirkung des Transponders auf das Lesegerät (Gl. (2.22)). Dieses Prinzip wird für die Datenübertragung vom Transponder zum Lesegerät verwendet.

Das Lesegerät sieht den Transponder als transformierte Impedanz Z' . Abbildung 2.11 zeigt, wie sich diese aus der tatsächlichen Transponderimpedanz Z und der induktiven Kopplung ergibt. Änderungen der Transponderimpedanz bewirken daher Amplituden- bzw. Phasenänderungen der Spannung an der Antenne des Lesegerätes. Variiert nun der Transponder seine Impedanz, dann lassen sich diese Änderungen vom Lesegerät als Spannungsänderungen detektieren. Dieses Modulationsverfahren wird als Last-

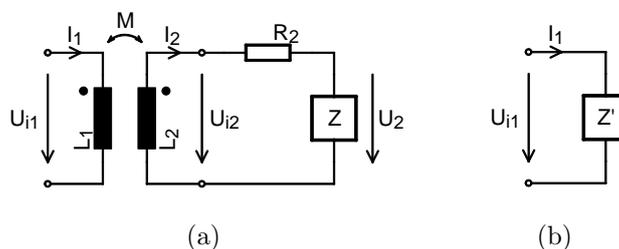


Abbildung 2.11: Überführung der Transponderimpedanz Z (a) in die transformierte Transponderimpedanz Z' (b)

modulation bezeichnet.

Für RFID-Systeme gibt es zwei mögliche Arten der Lastmodulation [4]. Zum einen gibt es die ohmsche Lastmodulation. Dabei wird ein zusätzlicher Modulationswiderstand R_{mod} parallel zum Lastwiderstand R_L des Transponders geschaltet. Dieser bewirkt eine reine Amplitudenmodulation der Antennenspannung am Lesegerät. Zum anderen gibt es die kapazitive Lastmodulation. Dabei wird eine zusätzliche Modulationskapazität C_{mod} parallel zur Schwingkreiskapazität C_2 geschaltet. Diese bewirkt sowohl eine Amplituden- als auch eine Phasenmodulation der Antennenspannung am Lesegerät.

Bei Proximity- [17] und Vicinity-Cards [21] wird diese Lastmodulation mit einem Hilfsträger durchgeführt. Abbildung 2.12 zeigt das Spektrum

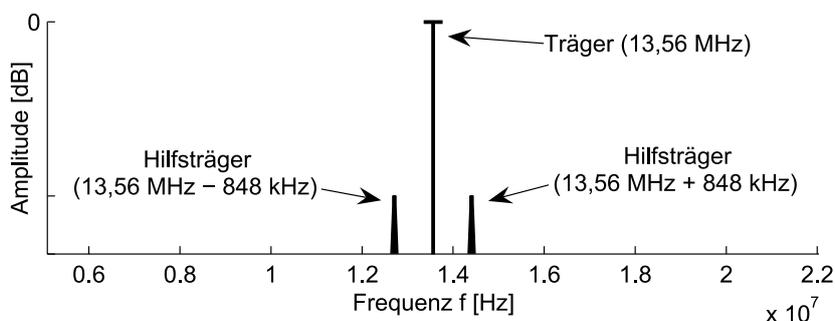


Abbildung 2.12: Lastmodulation eines 13,56-MHz-Trägersignals mit einem 848-kHz-Hilfsträger

eines lastmodulierten Signals. Dabei ist zu erkennen, dass die Modulationsseitenbänder eine deutlich geringere Energie als das Trägersignal aufweisen. Die Modulation mit einem Hilfsträger ist also notwendig, um das dominierende, für den Energietransport verwendete Trägersignal vom Datensignal zu trennen. Dadurch wird eine Filterung und Detektion des Datensignals im Lesegerät möglich.

2.4 Eigenschaften aktueller Systeme

Es gibt verschiedene Standards für induktiv gekoppelte RFID-Systeme. Für kontaktlose Chipkarten werden drei Standards besonders häufig eingesetzt:

- *Proximity cards* (ISO/IEC 14443 [16–19]),
- *Vicinity cards* (ISO/IEC 15693 [20–22]) und
- *FeliCa*.

Während ISO/IEC 14443 (*proximity*, engl. Nähe) und FeliCa auf sehr kurze Distanzen (bis zu 10 Zentimeter) ausgelegt sind, ist ISO/IEC 15693 (*vicinity*, engl. Umgebung) auch für größere Distanzen vorgesehen. Alle drei Systeme haben gemeinsam, dass es sich um induktiv gekoppelte Systeme mit einer Betriebsfrequenz (f_c) von 13,56 MHz handelt.

2.4.1 Proximity cards

Proximity-Systeme werden typischerweise für elektronische Bezahl-, Ticket- und Identifikationsanwendungen verwendet. Mifare, Calypso, elektronische Reisepässe und verschiedene andere Prozessorkarten basieren auf diesem Standard.

Der Standard ISO/IEC 14443 ist in vier Teile gegliedert:

- Teil 1 [16] beschreibt den physikalischen Aufbau von Proximity-Transpondern (PICC, *Proximity integrated circuit card*). Dazu zählen die Antennenabmessungen und die maximale Feldstärke.
- Teil 2 [17] beschreibt OSI-Schicht 1, d. h. die physikalische Schnittstelle zwischen Proximity-Lesegeräten (PCD, *Proximity coupling device*) und PICCs. Es wird der Energie- und Datentransfer definiert.
- Teil 3 [18] und Teil 4 [19] beschreiben die OSI-Schichten ab Schicht 2, d. h. den Verbindungsaufbau, die dazu notwendigen Antikollisionsverfahren, den Aufbau der Datenrahmen, den Datenaustausch und die gleichzeitige Aktivierung mehrerer PICCs.

Beim Datentransfer wird der Standard in Typ-A- und Typ-B-Systeme unterteilt. Beide Systeme unterstützen die Datenraten 106 kBit/s ($f_c/128$), 212 kBit/s ($f_c/64$), 424 kBit/s ($f_c/32$) und 848 kBit/s ($f_c/16$). Weiters verwenden beide Systeme die Lastmodulation mit einem 848-kHz-Hilfsträger zur Datenübertragung vom Transponder zum Lesegerät.

Typ A

Für die Kommunikation vom PCD zur PICC werden eine 2-ASK-Modulation mit einem Modulationsindex von 100 Prozent und eine modifizierte Millercodierung verwendet. Für die Datenübertragung in die entgegengesetzte Richtung wird das Datensignal manchestercodiert und der Hilfsträger mit OOK moduliert.

Typ-A-Systeme verwenden ein binäres Suchverfahren zur Antikollision mehrerer Transponder. Für den Datenaustausch werden drei verschiedene Frameformate definiert:

- Kurzframes (Abb. 2.13) enthalten nur sieben Datenbits und werden für den Beginn der Datenübertragung verwendet.
- Standardframes (Abb. 2.14) enthalten ein oder mehrere Datenbytes und werden für den normalen Datenaustausch verwendet.
- Antikollisionsframes werden für das Antikollisionsverfahren verwendet. Sie sind ähnlich aufgebaut wie Standardframes, allerdings wird ein Teil des Frames (Transponderauswahl) vom PCD zur PICC gesendet und alle mit dieser Auswahl übereinstimmenden Transponder antworten mit dem Rest des Frames.

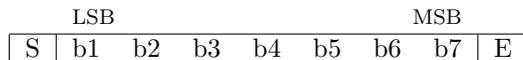


Abbildung 2.13: Kurzframe, bestehend aus Startbit S, sieben Datenbits und Endsequenz E [18]

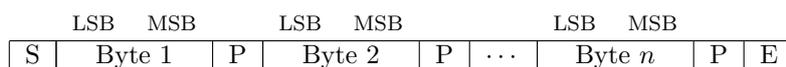


Abbildung 2.14: Standardframe, bestehend aus Startbit S, n Datenbytes (mit je einem Paritätsbit P) und Endsequenz E [18]

Typ B

Für die Kommunikation vom PCD zur PICC werden eine 2-ASK-Modulation mit einem Modulationsindex von 10 Prozent und eine NRZ-L-Codierung verwendet. Für die Datenübertragung in die entgegengesetzte Richtung wird das Datensignal ebenfalls NRZ-L-codiert und der Hilfsträger BPSK-moduliert.

Typ-B-Systeme verwenden ein ALOHA-Verfahren mit Zeitraster (*slotted ALOHA*) zur Antikollision mehrerer Transponder. Für den Datenaustausch werden mehrere Zeichen (Abb. 2.15) in ein Frame (Abb. 2.16) verpackt.

2. Radio Frequency Identification (RFID)

20

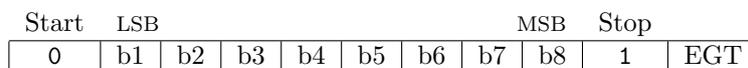


Abbildung 2.15: Zeichen, bestehend aus Startbit, acht Datenbits, Stoppbit und Totzeit (EGT, extra guard time) [18]

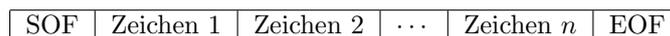


Abbildung 2.16: Frame, bestehend aus Startsequenz SOF, n Datenzeichen und Endsequenz EOF [18]

2.4.2 FeliCa

FeliCa (*Felicity Card*) ist ein proprietäres kontaktloses Smartcard-System von Sony. Es ist in Bezug auf Einsatzgebiete, Reichweite, Funktionsweise und Datenraten vergleichbar mit Proximity-Systemen. Es wurde deshalb auch als Typ-C-System für den Standard ISO/IEC 14443 vorgeschlagen. Jedoch ist es nicht in diesen Standard aufgenommen worden.

FeliCa wird heute hauptsächlich in Asien verwendet. Vor allem in Japan sind viele verschiedene elektronische Bezahlungssysteme auf der Basis von FeliCa verbreitet.

2.4.3 Vicinity cards

Vicinity-Systeme werden typischerweise für Zutrittskontrollsysteme und Logistik-Anwendungen verwendet.

Der Standard ISO/IEC 15693 ist in drei Teile gegliedert:

- Teil 1 [20] beschreibt den physikalischen Aufbau von Vicinity-Transpondern (VICC, *Vicinity integrated circuit card*). Dazu zählen die Antennenabmessungen, die maximale Feldstärke und verschiedene weitere physikalische Maximalbelastungen.
- Teil 2 [21] beschreibt OSI-Schicht 1, d. h. die physikalische Schnittstelle zwischen Vicinity-Lesegeräten (VCD, *Vicinity coupling device*) und VICCs. Es wird der Energie- und Datentransfer definiert.
- Teil 3 [22] beschreibt die OSI-Schichten ab Schicht 2, d. h. den Verbindungsaufbau, dazu notwendige Antikollisionsverfahren, den Aufbau der Datenrahmen und den Datenaustausch.

Für die Kommunikation vom VCD zur VICC wird eine 2-ASK-Modulation mit einem Modulationsindex von wahlweise 10 oder 100 Prozent und eine Pulslagen-Codierung verwendet. Je nach Datenrate wendet man eine 1-aus-256- (1,66 kBit/s) oder eine 1-aus-4-Pulslagen-Codierung (26,48 kBit/s)

2. Radio Frequency Identification (RFID)

21

an. Für die Datenübertragung in die entgegengesetzte Richtung kommt die Lastmodulation mit Hilfsträger zum Einsatz. Das Datensignal wird dazu manchestercodiert. Wahlweise moduliert man den Hilfsträger anschließend mit dem OOK- oder dem FSK-Verfahren. Der Hilfsträger hat bei der OOK-Modulation eine Frequenz von 424 kHz ($f_c/32$). Für die FSK-Modulation wird ein Hilfsträger mit 424 kHz ($f_c/32$) und ein weiterer mit 484 kHz ($f_c/28$) verwendet. Mit diesen Verfahren lassen sich Übertragungsraten mit bis zu 26,69 kBit/s erzielen.

Für den Datenaustausch werden zwei Frameformate definiert:

- Anfrageframes (Abb. 2.17) werden verwendet um Befehle und Daten vom VCD zur VICC zu senden.
- Antwortframes (Abb. 2.18) werden verwendet um Daten von der VICC zum VCD zu senden.

SOF	Flags (1 Byte)	Befehl (1 Byte)	Parameter (n Byte)	Daten (m Byte)	CRC (2 Byte)	EOF
-----	-------------------	--------------------	--------------------------	----------------------	-----------------	-----

Abbildung 2.17: Anfrageframe, bestehend aus Startsequenz (SOF), Markierungsbits (Flags), Befehlscode, Parametern, Daten, Prüfsumme (CRC) und Endsequenz (EOF) [22]

SOF	Flags (1 Byte)	Parameter (x Byte)	Daten (y Byte)	CRC (2 Byte)	EOF
-----	-------------------	--------------------------	----------------------	-----------------	-----

Abbildung 2.18: Antwortframe, bestehend aus Startsequenz (SOF), Markierungsbits (Flags), Parametern, Daten, Prüfsumme (CRC) und Endsequenz (EOF) [22]

2.4.4 Near Field Communication

Eine Weiterentwicklung dieser Standards ist NFC¹, *Near Field Communication* [23, 24]. Neben einer standardisierten Speicherstruktur definiert NFC auch, wie NFC-Geräte miteinander und mit einer auf den vorhergehenden Standards basierenden Infrastruktur kommunizieren können: NFC-Geräte sind in der Lage als Transponder (*Card emulation mode*) oder als Leseeinheit (*Reader/writer mode*) zu agieren. Zusätzlich können NFC-Geräte auch direkt miteinander kommunizieren (*Peer-to-peer mode*).

¹Aufbauend auf der Standardisierung durch die ISO/IEC [23, 24] wird NFC durch das NFC Forum (<http://www.nfc-forum.org/>) spezifiziert.

2. Radio Frequency Identification (RFID)

22

NFC ist für kurze Reichweiten (bis zu 10 Zentimeter) vorgesehen. Typische Anwendungen sind elektronische Bezahl- und Ticketsysteme, Smartposter (das sind Plakate mit RFID-Transpondern, auf denen z. B. Telefonnummern für SMS-Dienste (*Short Message Service*) oder Links zu Internetangeboten gespeichert sind), Bluetooth- und WLAN-Pairing (d. h. der, zum Verbindungsaufbau notwendige, Austausch von Identifikations- und Authentifikationsinformationen drahtlos kommunizierender Geräte) und andere Steueraufgaben.

Kapitel 3

Grundlagen für höhere Datenraten

Von den in Kapitel 2 betrachteten kontaktlosen Smartcard-Systemen erreichen Proximity-Systeme nach dem Standard ISO/IEC 14443 die höchste Bruttodatenübertragungsrate¹. Diese liegt bei 848 kBit/s. Das Forschungsprojekt VHD (*Very High Datarate*) – *High Speed Air-Interface and IC Architecture for Contactless Smartcards and NFC* – beschäftigt sich mit der Erhöhung dieser Datenrate um ein Vielfaches. Dabei wird gleichzeitig auf Rückwärtskompatibilität zum aktuellen Standard geachtet. Die angestrebten Werte für die gesteigerte Datenrate liegen bei bis zu 13,56 MBit/s.

Dieses Kapitel beschreibt die Vorteile hochratiger RFID- und NFC-Systeme für bestehende und zukünftige Anwendungen. Weiters gibt es einen Überblick über die bisherigen Aktivitäten zur Erhöhung der Datenrate kontaktloser Smartcards.

3.1 Einsatzgebiete für RFID- und NFC-Systeme mit höheren Datenraten

Ein Hauptgrund für die Erhöhung der Datenrate ist das zunehmende Datenvolumen. Kontaktlose Smartcard-Systeme sind längst über einfache Identifikationsanwendungen, dem ursprünglichen Einsatzgebiet von RFID, hinausgewachsen.

¹Die Bruttodatenübertragungsrate ist die Geschwindigkeit mit der alle Daten, d. h. sowohl Nutzdaten als auch Steuerinformationen, übertragen werden.

3.1.1 Beispiel: ePass

Ein momentan viel diskutiertes Thema ist der *ePass*, der elektronische Reisepass. Die erste Generation² der österreichischen elektronischen Reisepässe enthält bereits alle Daten der maschinenlesbaren Zone, den vollständigen Namen, den Geburtsort und ein Foto (mit den Abmessungen 413 Pixel × 531 Pixel) des Passinhabers, die ausstellende Behörde und Informationen zum Zugriffsschutz und zur Echtheitsbestimmung des Dokuments. Insgesamt sind etwa 22,5 Kilobyte an auslesbaren Daten gespeichert. Die zum Auslesen und Verarbeiten notwendige Zeit wurde mit zwei verschiedenen frei verfügbaren Anwendungen zum Auslesen von Reisepässen gemessen und in Tabelle 3.1 dargestellt. Die Messungen zeigen, dass bereits bei dieser ge-

Tabelle 3.1: Lesedauer eines österr. Reisepasses der ersten Generation

Anwendung	Lesegerät	Dauer
RFID IO tools ^a	Omniquey 5321 RFID	15 Sekunden
PassportTest1 ^b	Nokia 6131 NFC	30 Sekunden

^aEine von Adam Laurie entwickelte Python-Anwendung zum Auslesen von RFID-Transpondern (<http://rfidiot.org/>).

^bEine selbst entwickelte *Java Micro Edition* Anwendung zum Auslesen elektronischer Reisepässe.

ringen Menge an Informationen ein Zeitaufwand von deutlich mehr als zehn Sekunden zum Auslesen der Reisepassdaten besteht.

Dabei ist jedoch zu beachten, dass nur ein relativ geringer Anteil an dieser Zugriffszeit für die Datenübertragung aufgewendet wird. Die durch die ICAO (International Civil Aviation Organization) und die ISO durchgeführten Interoperabilitätstests haben Lesezeiten zwischen 3 und 30 Sekunden für einen etwa 20 Kilobyte großen Datensatz ergeben [13]. Eine Aufstellung und Zeitabschätzung der verwendeten Befehle (APDU, *Application protocol data unit*) ergab, dass die Übertragung bei 106 KBit/s jedoch nur etwa 2,3 Sekunden und bei 848 KBit/s sogar nur etwa 0,3 Sekunden dauert. Messungen der Übertragungszeit mit einem CLT Move von Comprion ergaben allerdings, dass die verwendeten APDUs in deutlich kürzere Pakete des kontaktlosen Übertragungsprotokolls zerlegt werden. Dies lässt sich durch die limitierten Pufferspeicher des Reisepass-ICs erklären. Dadurch kommt es jedoch zu einer beträchtlichen Erhöhung der Übertragungszeit.

Darüber hinaus basieren die momentanen kontaktlosen Chipkarten-ICs auf deutlich älteren Technologien [26]. Durch neuere Technologien sind bereits viel schnellere Speicherzugriffe und damit wesentlich geringere Verar-

²Die erste Generation der österreichischen Reisepässe mit Datenträger wurde zwischen dem 16. Juni 2006 und dem 30. März 2009 ausgestellt.

beitungszeiten möglich. Dadurch stellt die niedrige Datenrate gerade bei neuen IC-Technologien einen Engpass dar.

Die zweite Generation³ der österreichischen elektronischen Reisepässe enthält als weiteres biometrisches Merkmal zwei Fingerabdrücke. Zukünftige Reisepässe sollen alle zehn Fingerabdrücke enthalten. Des Weiteren können auf dem elektronischen Datenträger andere Merkmale, wie die Signatur des Passinhabers, und weitere Informationen, wie Visa, gespeichert werden.

Das dadurch enorm ansteigende Datenvolumen führt zu einer Erhöhung der Lesezeit. Mit der Implementierung höherer Datenraten lässt sich diese gesteigerte Lesezeit wieder verkürzen.

3.1.2 Beispiel: NFC

Eine weitere Technologie, die von höheren Datenraten der RFID-Schnittstelle profitiert, ist NFC mit seinen zahlreichen Anwendungsmöglichkeiten. Gerade im Zusammenhang mit NFC-fähigen Mobiltelefonen sind hohe Übertragungsraten vorteilhaft.

TCP/IP über NFC

In [7] wird die Übertragung von TCP/IP-Datenpaketen (TCP/IP, *Transmission control protocol/Internet protocol*, ist eine Familie von Netzwerkprotokollen) über die kontaktlose NFC-Schnittstelle beschrieben. Nachdem Webserver auch zunehmend in Embedded Systeme integriert werden, kann in Zukunft auch die TCP/IP-Datenübertragung über NFC an Bedeutung gewinnen. Durch den Overhead und die Nutzdatenmenge, die sowohl mit dem Tunneln einer TCP/IP-Verbindung als auch mit den darauf aufbauenden Protokollen verbunden sind, ist man durch die momentan spezifizierten Datenraten sehr eingeschränkt. In [7] wird eine maximal erreichbare durchschnittliche Datenrate von weniger als 1,5 kByte/s angegeben.

Sensoren mit NFC-Schnittstelle

Eine neue und vielversprechende Anwendung von NFC sind Sensoren, die Mobiltelefone und andere NFC-fähige Geräte als Benutzerschnittstelle verwenden [9]. Solche Sensoren können z. B. beim Hausbau fest in Wänden (vor allem von Feuchträumen) installiert werden. Später können dann Messwerte mit dem Mobiltelefon, durch einfache Berührung der Wand ausgelesen, werden. Die dabei übertragenen Sensordaten können von einfachen Momentanwerten bis zu Aufzeichnungen von Langzeitmessungen reichen. Gerade für Langzeitmesswerte ist eine schnelle Datenübertragung nützlich.

³Die zweite Generation der österreichischen Reisepässe mit Datenträger wird seit dem 30. März 2009 ausgestellt.

Mobiltelefone als grafische Benutzerschnittstelle

Mobiltelefone sind heutzutage bereits viel mehr als einfache Telefone. Sie enthalten auch Fotoapparat, Webbrowser, E-Mail-Programm, Computerspiele, Navigationssystem und zahlreiche weitere Anwendungen. Das Mobiltelefon ist aus dem alltäglichen Leben kaum mehr wegzudenken. Aus diesem Grund bietet es sich an, das Mobiltelefon als überall verfügbare Benutzerschnittstelle einzusetzen. Mit NFC lässt sich eine Vielfalt an solchen Anwendungen realisieren.

Während einfache Steueraufgaben [27] nur ein geringes Datenvolumen haben und somit auch mit geringen Übertragungsraten realisierbar sind, gibt es auch zahlreiche Anwendungen, die große Datenmengen übertragen und deshalb auf höhere Datenraten angewiesen sind. Beispiele für Anwendungen mit großen Datenmengen sind

- Smartposter mit aktiven Inhalten, wie z. B. Videoclips,
- Zeitschriften, die Klingeltöne, Trailer oder ähnliche aktive Elemente enthalten,
- Haushaltsgeräte, deren Bedienungsanleitung man mit einer einfachen Berührung auf das Mobiltelefon oder das eBook-Lesegerät lädt,
- elektronische Zeitschriften, die man am Kiosk mit einer einfachen Berührung bezahlen und direkt auf das Mobiltelefon oder das eBook-Lesegerät laden kann, und
- Beschreibungen, Audio- und Videoinhalte zu Ausstellungsstücken in Museen.

Gemeinsam genutzte Wiedergabegeräte

Es gibt auch Anwendungen bei denen das mobile Gerät nicht die zentrale Steuereinheit, sondern viel mehr der Datenspeicher ist. Beispiele dafür sind die Interaktion mit Videoprojektoren, Fotodruckern, Personalcomputern (z. B. zur Einbindung von Web-2.0-Anwendungen wie YouTube⁴ oder Facebook⁵) und öffentlichen Bildschirmen (z. B. elektronische Anschlagbretter). Das mobile Gerät ist dabei der Datenspeicher. Mit einer kurzen Berührung können Präsentationen, Fotos, Videos und Texte übertragen werden.

3.2 Systemkonzepte

Diese Diplomarbeit beschäftigt sich mit der Entwicklung eines Demonstrators zur hochratigen Datenübertragung im Downlink, d. h. vom Transponder

⁴YouTube (<http://youtube.com/>) ist eine Online-Plattform zum Austausch von Videos.

⁵Facebook (<http://facebook.com/>) ist eine Online-Plattform zur Formung sozialer Netzwerke.

zum Lesegerät. Im Weiteren wird deshalb nur diese Übertragungsrichtung in Bezug auf die höheren Datenraten behandelt. Die nachfolgenden Untersuchungen und Realisierungen richten sich an den in [1, 8, 12] erarbeiteten Konzepten.

3.2.1 System für höhere Datenraten

In [1] wird ein System vorgestellt, welches das Senden von Daten von einem Transponder zu einem Lesegerät mit Datenraten bis zu 13,56 MBit/s ermöglicht. Abbildung 3.1 zeigt das Blockschaltbild dieses Systems. Es be-

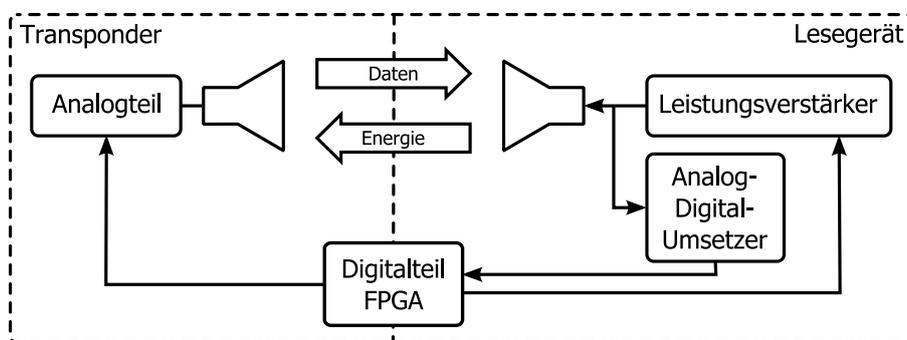


Abbildung 3.1: Blockschaltbild eines RFID-Systems zur hochrätigen Datenübertragung im Downlink [1]

steht aus einem semi-passiven Transponder und einem Lesegerät.

Der Transponder setzt sich aus einer Antenne, einer aus diskreten Bauteilen aufgebauten analogen Modulationsschaltung und einem Digitalteil zusammen. Eine weitere Antenne, ein Leistungsverstärker, ein Analog-Digital-Umsetzer und ein Digitalteil bilden das Lesegerät.

Der Analogteil des Transponders wurde aus einer bestehenden Referenzimplementierung eines Proximity-Transponders nach [14] übernommen. Die Bauform der Transponderantenne entspricht einer *Class 1 PICC* nach [16].

Für das Lesegerät wurde eine PCD-Testantenne nach [14] verwendet. Als Alternative zu dieser Antenne wird ein Zwei-Antennen-System [1, 8], d. h. eine Auftrennung in eine Antenne zur Energieversorgung und eine weitere Antenne zum Datenempfang, beschrieben. Dieses Konzept wäre besonders leistungsfähig, weil die Güte der Energie abstrahlenden Antenne praktisch keine Auswirkung auf die Datenübertragung zur Empfangsantenne hat. Damit lässt sich eine gute Energieübertragung bei gleichzeitig hoher Bandbreite realisieren. Aufgrund der, in [1] beschriebenen, patentrechtlichen Situation ist dieses Konzept jedoch für eine kommerzielle Nutzung ungeeignet.

Als Leistungsverstärker kommt der Verstärker aus einem I•CODE-System von NXP Semiconductors zum Einsatz. Die Analog-Digital-Umsetzung findet mit Hilfe eines AD9430 von Analog Devices statt. Dieser hat eine

Auflösung von 12 Bit und eine maximale Umsetzungsrate von 210 Millionen Abtastpunkten pro Sekunde.

Die Digitalteile des Transponders und des Lesegerätes sind gemeinsam in einem FPGA (*Field-programmable gate array*, engl. vor Ort programmierbarer Logik-IC) untergebracht. Ein FPGA besteht aus Logikblöcken, deren Funktion und Zusammenschaltung im Nachhinein festlegbar ist. Dazu werden Programmiersprachen, sogenannte Hardwarebeschreibungssprachen, wie z. B. VHDL (*Very high speed integrated circuits hardware description language*) und Verilog verwendet.

Transponder und Lesegerät teilen sich ein gemeinsames FPGA und einen gemeinsamen Systemtakt. Sie arbeiten deshalb synchron zueinander. Zur Signalgenerierung im Transponder steht daher bereits der Takt des Trägersignals (bzw. ein Vielfaches davon) zur Verfügung. Auch bei der Signalauswertung müssen dadurch nur die Signallaufzeiten durch die einzelnen Systemkomponenten berücksichtigt werden.

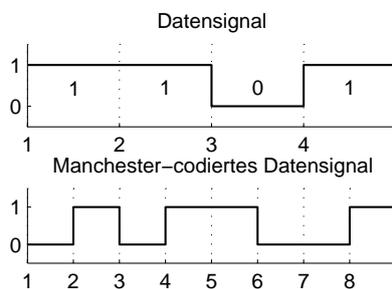
3.2.2 Parameter für höhere Datenraten

Modulation und Codierung

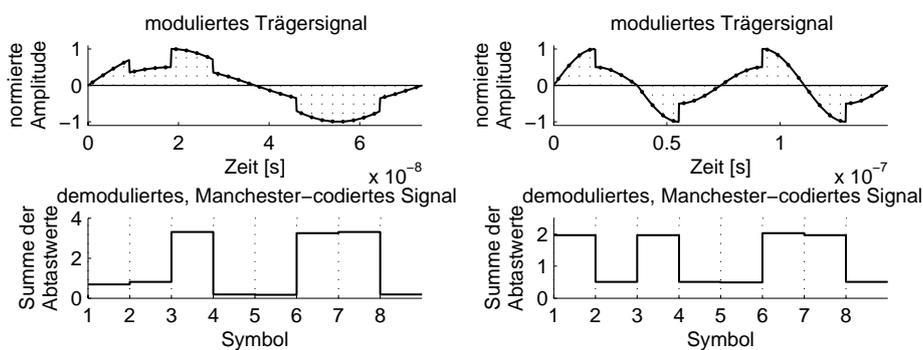
Anhand von vier Faktoren (dem Energiegehalt, der Komplexität der Codierung, der Komplexität der Decodierung und der Fehleranfälligkeit) wurden in [1] mehrere Codierungsverfahren auf ihre Eignung für ein RFID-System mit höheren Datenraten untersucht. Die Entscheidung fiel dabei auf die Manchestercodierung zusammen mit einer Lastmodulation ohne Hilfsträger. Das entspricht einer BPSK-Modulation eines Hilfsträgers dessen Frequenz mit der Bitrate übereinstimmt. Die Manchestercodierung hat einen konstanten Energiegehalt von 50 Prozent, d. h. das HF-Feld ist für die Hälfte der Übertragungszeit nicht moduliert. Die Codesequenz ist leicht zu erzeugen, nachdem sich jedes Bit aus sich selbst und einer invertierten Version zusammensetzt. Die Decodierung ist besonders einfach, weil kein Schwellenwertentscheider notwendig ist. Jedes Bit kann durch einen Vergleich der beiden Bithälften ermittelt werden. Dadurch ist die Decodierung auch unempfindlich gegenüber dynamischen Effekten wie z. B. einer Bewegung des Transponders im Feld des Lesegerätes und der damit verbundenen Amplitudenschwankung. Um diese einfache Decodierung zu ermöglichen muss auch die Datenübertragungsrate bestimmte Eigenschaften aufweisen.

Datenübertragungsrate

Abbildung 3.2 zeigt das lastmodulierte Signal bei Verwendung einer Manchestercodierung und unterschiedlichen Datenübertragungsraten. Dieses Signal wurde anschließend durch Integration decodiert. In Abbildung 3.2(b) ist erkennbar, dass eine Rekonstruktion des codierten Sendesignals bei einer Datenrate von 54,24 MBit/s nicht mehr möglich ist. Bei einer Datenrate von

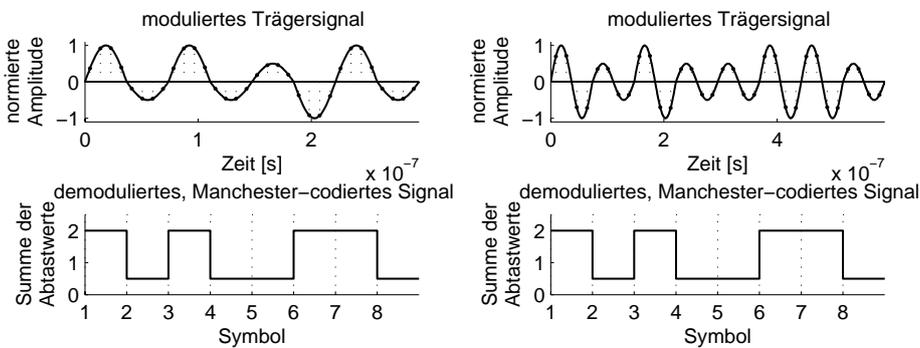


(a) codiertes Datensignal



(b) 54,24 MBit/s

(c) 27,12 MBit/s



(d) 13,56 MBit/s

(e) 6,78 MBit/s

Abbildung 3.2: Lastmodulation ohne Hilfsträger mit Manchestercodierung bei unterschiedlichen Datenübertragungsraten und anschließende Rekonstruktion durch Aufsummieren von vier Abtastwerten (bzw. deren Absolutbetrag) pro Sendesymbol

27,12 MBit/s (Abb. 3.2(c)) ist zwar eine Rekonstruktion möglich, jedoch weist das modulierte Signal Sprungstellen auf. Erst bei einer Datenrate von 13,56 MBit/s ergibt sich ein kontinuierliches Signal. Die Datenrate kann also nur 13,56 MBit/s oder gerade Teiler davon annehmen.

Um zusätzlich auch eventuelle Unterschiede bei der Modulation der positiven und der negativen Halbwelle auszugleichen muss die Datenrate sogar den Wert 6,78 MBit/s oder einen gerade Teiler davon annehmen. Damit besteht ein Teilsymbol aus gleich vielen positiven und negativen Halbwellen. Unterschiede zwischen positiven und negativen Halbwellen wirken dadurch auf jedes Symbol gleich.

Gütefaktor

Ein weiterer Parameter ist der Gütefaktor. Die Güte gibt an, auf wieviele Signalperioden sich eine einzelne Signaländerung auswirkt. Während eine hohe Güte bedeutet, dass viel Energie über das HF-Feld übertragen werden kann, ist damit eine zunehmende Trägheit des Datensignals verbunden. Ist diese Trägheit größer als die Symboldauer, dann kommt es zu Intersymbolinterferenz (ISI). Jedes Symbol wirkt daher auch auf nachfolgende Symbole. Somit wird die Rekonstruktion des Sendesignals deutlich aufwendiger. Für das Lesegerät wird eine Antenne mit der Güte $Q = 30$ verwendet.

Modulationsindex

Der Modulationsindex, ebenfalls ein wichtiger Systemparameter, ist das Verhältnis zwischen der Amplitude des einhüllenden Datensignals und der gesamten Trägeramplitude. Nachdem in diesem Systemkonzept keine Filterung vor der Digitalisierung vorgesehen ist, bestimmt der Modulationsindex die notwendige Auflösung des Analog-Digital-Umsetzers. Neben dem Kopplungsfaktor, dem Gütefaktor und dem Abstand zwischen Transponder- und Leseantenne, ist der Modulationswiderstand maßgeblich für den Modulationsindex verantwortlich. Der Modulationswiderstand ist jener Widerstand mit dem der gesamte Lastwiderstand des Transponders zur Modulation variiert wird. Weil der Modulationswiderstand parallel zum Lastwiderstand des Transponders geschaltet wird, bewirkt ein kleiner Modulationswiderstand eine große Modulationstiefe. Das bedeutet jedoch auch, dass in jenen Phasen, in denen der Modulationswiderstand zugeschaltet ist, ein Großteil der Energie vom Modulationswiderstand aufgenommen wird und nur wenig Energie für die Transponderschaltung zur Verfügung steht. Nachdem in [1] nur ein semi-passiver Transponder zum Einsatz kommt, und damit die Energieübertragung nur eine untergeordnete Rolle spielt, wird ein Modulationswiderstand von weniger als 30 Ohm vorgeschlagen. Weiters lassen sich die Pausen in der Energieübertragung durch Stützkapazitäten ausgleichen. Diese müssen so dimensioniert sein, dass die minimal notwendige Versorgungsspannung über die Dauer der größtmöglichen Modulationsphase aufrechterhalten wird.

3.2.3 Transponderkonzept für höhere Datenraten

Das bisher beschriebene System verwendet eine PICCC-Referenzimplementierung nach [14]. Dieser Analogteil wurde für die derzeit standardisierten Datenraten entwickelt. Deshalb stellt [12] ein neuartiges Transponderkonzept vor, mit dem Datenübertragungsraten bis zu 6,78 MBit/s erreichbar sind. Abbildung 3.3 zeigt die wesentlichen Bestandteile dieses analogen Transpon-

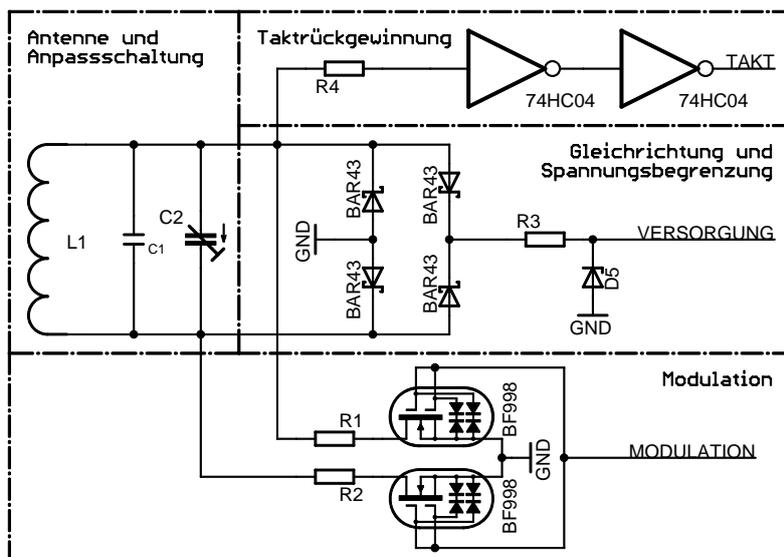


Abbildung 3.3: Transponder-Front-End zur hochrätigen Datenübertragung im Downlink [12]

der-Front-Ends. Die Antenne, die Anpassschaltung, die Gleichrichtung und die Spannungsbegrenzung entsprechen im Wesentlichen dem Referenzaufbau. Die Modulationsschaltung wurde so überarbeitet, dass auch die notwendigen kurzen Schaltzeiten keine Probleme verursachen. Sie besteht nun aus den zwei Modulationswiderständen und zwei N-Kanal-MOS-Feldeffekttransistoren (MOS-FETs) des Typs BF998 von NXP Semiconductors. Zur Taktrückgewinnung werden zwei Inverter in Serie geschaltet, um die Sinusschwingung des Trägersignals in ein Rechtecksignal mit (zum Digitalteil passenden) Logikpegeln umzuformen.

Der Digitalteil des Transponders ist bei diesem neuen Konzept in einem CPLD (*Complex programmable logic device*, engl. komplexer, programmierbarer Logik-IC) untergebracht. Ein CPLD besteht, wie ein FPGA, aus Logikblöcken, deren Funktion und Zusammenschaltung im Nachhinein mit Hilfe von Hardwarebeschreibungssprachen festlegbar ist. Jedoch sind auf CPLDs typischerweise viel weniger Logikblöcke verfügbar. Durch die Trennung der Digitalteile von Lesegerät und Transponder sind diese nur mehr über das HF-Feld gekoppelt. Des Weiteren ist durch die geringe Stromauf-

nahme des CPLDs ein passiver Transponderbetrieb möglich.

3.2.4 Filterung

In Abschnitt 2.3.3 wurde gezeigt, dass das zur Energieübertragung genutzte Trägersignal um ein Vielfaches größer ist, als das informationstragende Modulationssignal. Aus diesem Grund schlägt [12] eine Quarzbandsperre zur Unterdrückung des 13,56-MHz-Trägersignals im analogen Empfangssignal vor. Eine Quarzbandsperre hat einen sehr schmalen Sperrbereich von nur 6 kHz [32] und ist deshalb gut zur Trägerunterdrückung geeignet.

3.3 Prototypentwicklung

Ein Prototyp wurde entwickelt und um die jeweils erarbeiteten Konzepte erweitert. So entstand eine erste Demonstrationsumgebung. Der schematische Aufbau dieses Prototyps ist in Abbildung 3.4 dargestellt.

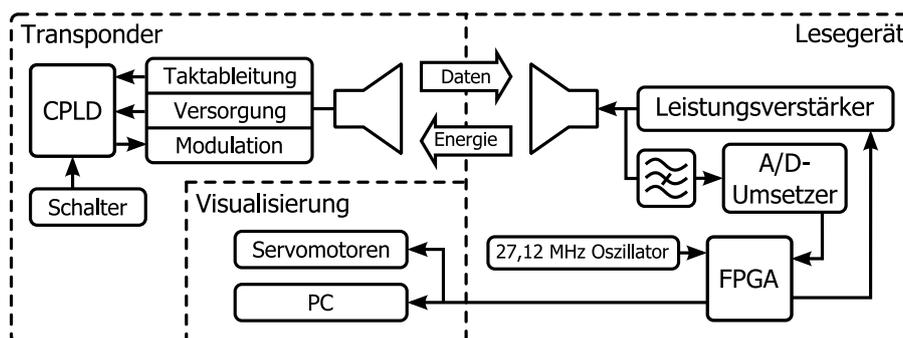


Abbildung 3.4: Blockschaltbild der Demonstratorplattform

3.3.1 Lesegerät

Das Lesegerät besteht aus einem Verstärker, einer Antenne, einer Quarzbandsperre, einem Analog-Digital-Umsetzer und einem FPGA. Das FPGA, ein Spartan-II XC2S150 von Xilinx, ist das Kernstück des Lesegeräts. Darauf finden die Trägergenerierung und die digitale Verarbeitung des Empfangssignals statt. Abbildung 3.5 zeigt das Blockschaltbild des Digitalteils.

Der externe 27,12-MHz-Takt wird mit der DLL (*Delay-locked loop*) des FPGAs zu einem 108,48-MHz-Systemtakt vervierfacht. Eine DLL ist eine Schaltung, die mit Hilfe einer einstellbaren Verzögerungskette und eines Regelkreises die Frequenz eines Taktsignals vervielfachen oder dividieren und die Phasenlage verändern kann.

Die Analog-Digital-Umsetzer-Steuerung tastet das Empfangssignal mit 27,12 MHz ab. Dabei ist die Phasenlage so eingestellt, dass Verzögerungen

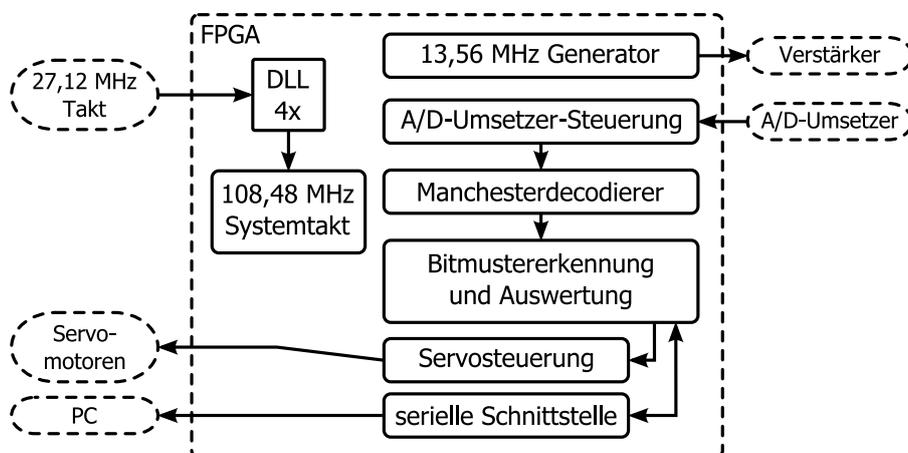


Abbildung 3.5: Blockschaltbild des Lesegerätes

durch den Verstärker, die Übertragungsleitungen und die Filterung ausgeglichen werden. Somit erhält man von jeder Trägerperiode den Maximalwert und den Minimalwert.

Der Manchesterdecodierer addiert den Betrag des Maximalwertes und des Minimalwertes jeder Trägerperiode und ermittelt so die Einhüllende des Empfangssignals. Anschließend werden jeweils zwei aufeinander folgende Punkte der Einhüllenden miteinander verglichen. Das Ergebnis dieses Vergleichs sind zwei manchesterdecodierte Empfangssignale. Je nachdem, ob man die erste Bithälfte mit der zweiten Bithälfte oder die zweite Bithälfte mit der ersten Bithälfte des nachfolgenden Bits vergleicht, gibt es zwei Möglichkeiten das Empfangssignal zu decodieren. Weil das Lesegerät und der Transponder zwar takt synchron aber nicht bitsynchron arbeiten, hat das Lesegerät im Vorhinein keine Kenntnis über die Richtigkeit einer der beiden Decodierungsvarianten.

Die Bitmustererkennung ist so aufgebaut, dass sie den empfangenen Datenstrom nach vier, fest definierten Bitmustern durchsucht. Wird eines der Bitmuster mehrmals in Folge erkannt, so wird dieses Bitmuster als detektiert markiert. Über die serielle Schnittstelle wird daraufhin eine Statusinformation an den PC weitergegeben, wo eine Auswertung in einer grafischen Oberfläche stattfindet. Zusätzlich bewegt die Servosteuerung zwei Servomotoren, um den Vergleich zwischen der hohen und der bisherigen Datenrate zu visualisieren.

3.3.2 Transponder

Der Transponder besteht aus den in Abschnitt 3.2.3 beschriebenen Teilen. Mit dem CPLD, einem CoolRunner-II XC2C64A, wird ein manchestercodierter Datenstrom erzeugt. Abbildung 3.6 zeigt das Blockschaltbild des

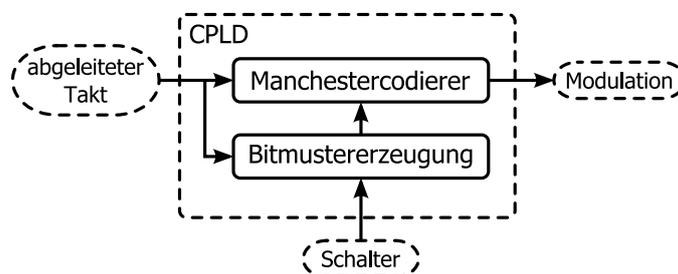


Abbildung 3.6: Blockschaltbild des Transponders

CPLDs.

Die Bitmustererzeugung wählt, in Abhängigkeit der Schalterstellungen, eines von vier vordefinierten Bitmustern aus. Jede Sequenz ist acht Bit lang und wird endlos wiederholt, sobald die Energieversorgung des Transponders ausreichend ist.

Der Manchestercodierer formt den Bitstrom in eine manchestercodierte Symbolfolge um und gibt diese an die Modulationsschaltung aus.

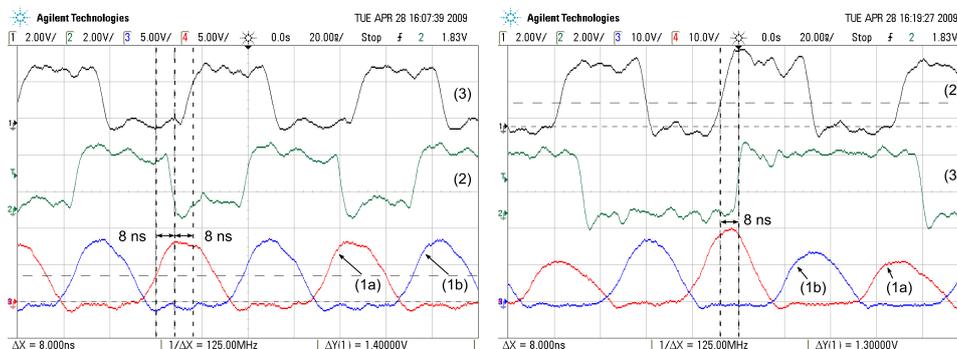
3.3.3 Probleme und Einschränkungen

Der Prototyp weist jedoch einige Probleme und Einschränkungen auf.

Modulation im Transponder

Abbildung 3.7 zeigt die Verzögerung von der Taktableitung bis zum ausgegebenen Modulationssignal im Transponder. Es ist zu erkennen, dass jede Inverterstufe der Taktableitung das Taktsignal um 8 ns verzögert. Insgesamt ergibt sich dadurch eine Phasenverschiebung von 16 ns zwischen dem empfangenen Trägersignal und dem Systemtakt. Die Verzögerung durch den CPLD beträgt ebenfalls 8 ns. Weiters kommt noch eine Schaltverzögerung durch die MOS-FETs hinzu. Insgesamt ergibt sich dadurch eine Verzögerung von deutlich mehr als einer halben Trägerperiode, d. h. von mehr als 37 ns.

Aus diesem Grund wird ein Modulationszustand erst auf die nächsten zwei Halbwellen angewendet. Problematisch daran ist jedoch, dass diese Verschiebung etwas länger als eine halbe Trägerperiode dauert. Dadurch werden die einzelnen Periodenhälften unterschiedlich stark moduliert. In Abbildung 3.8 ist die Modulation mehrerer Sendesymbole abgebildet. Während das erste Halbwellensignal (1a) immer etwa dieselben Pegel für den modulierten bzw. den nichtmodulierten Zustand aufweist, sind beim zweiten Halbwellensignal (1b, 1c) starke Pegelunterschiede über einen Modulationszustand hinweg erkennbar. Die Ursache dafür ist die Schaltverzögerung des Modulationstransistors. Diese bewirkt, dass der Schaltzustand bei der auf



(a) Systemtaktableitung aus dem empfangenen Trägersignal: erste (1a) und zweite (1b) Halbwellen des Trägers, erste (2) und zweite (3) Inverterstufe der Takttableitung

(b) Verzögerung zwischen dem Taktsignal und dem Modulationssignal: erste (1a) und zweite (1b) Halbwellen des modulierten Trägers, Taktsignal (2), Modulationssignal (3)

Abbildung 3.7: Verzögerungen im Transponder

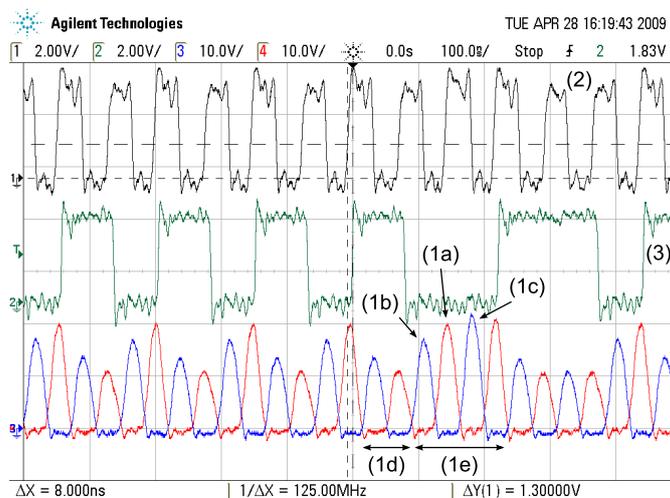


Abbildung 3.8: Moduliertes Trägersignal am Transponder: erste (1a) und zweite (1b, 1c) Halbwellen des modulierten Trägers, modulierte (1d) und unmodulierte (1e) Phasen, Taktsignal (2), Modulationssignal (3)

das Schaltsignal folgenden Halbwellen noch nicht vollständig erreicht ist, wodurch nicht die gesamte Halbwellen den gewünschten Schaltzustand (bzw. Modulationszustand) erfährt.

Filterung im Lesegerät

Im momentanen Demonstratoraufbau ist, abgesehen von der Trägerunterdrückung, keine Filterung vorgesehen. Auch die Trägerunterdrückung löst, in ihrer aktuellen Implementierung, Probleme aus: Die Kombination aus dem verwendeten 27,12-MHz-Oszillator und der Vervierfachung des Taktes mit Hilfe der DLL des FPGAs führt dazu, dass das generierte 13,56-MHz-Trägersignal Frequenzschwankungen (Jitter) aufweist. Diese Schwankungen bewegen sich in Größenordnungen die den engen Sperrbereich von 6 kHz der Bandsperre überschreiten. Dadurch kommt es zu großen Amplitudenschwankungen im gefilterten Signal, die eine Demodulation des Empfangssignals extrem erschweren.

Der Entwurf der analogen Filterstruktur (Antialiasing und verbesserte Trägerunterdrückung) und die Charakterisierung und Behebung der Schwankungen im Trägersignal würde den Rahmen dieser Arbeit sprengen und sind Thema einer weiteren Diplomarbeit.

Auch eine digitale Filterung ist in diesem Prototyp noch nicht vorhanden. In [1] wurde allerdings bereits gezeigt, dass durch ein Entzerrerfilter deutliche Verbesserungen des Empfangssignals erzielbar sind.

Signalabtastung und Demodulation im Lesegerät

In Bezug auf die Signalabtastung und Demodulation weist der Demonstrator gleich mehrere Einschränkungen auf: Zum einen wird die Auswahl des Abtastzeitpunktes nicht dynamisch ermittelt, sondern durch grobe Schätzung und manuelles Abstimmen bestimmt und bereits in der Entwicklungsphase fest vorgegeben. Zum anderen wird nur ein Bruchteil der möglichen Abtastrate genutzt.

Die Verzögerung zwischen dem generierten Trägersignal und dem Empfangssignal wird durch die Komponenten entlang dieses Pfades bestimmt. Dazu zählen der Verstärker, die Antennen und deren Kopplung, die analogen Filter und die Leitungsverzögerungen. Die meisten dieser Teilverzögerungen sind statisch, d. h. sie verändern sich nicht. Durch den variablen Abstand zwischen Transponder- und Leseantenne, bzw. durch eine Bewegung des Transponders im Feld der Leseantenne, tritt jedoch auch eine dynamische Verzögerung auf. Solche dynamischen Effekte bleiben bei der Signalabtastung gänzlich unberücksichtigt. Auch die Auswirkung der statischen Verzögerung wird lediglich durch Schätzung in die Auswahl einbezogen.

Durch den 108,48-MHz-Systemtakt und den hochwertigen A/D-Umsetzer lässt sich problemlos eine Abtastrate von 108,48 MHz, das sind acht Abtastpunkte pro Trägerperiode, erreichen. Der Prototyp nutzt jedoch nur zwei Abtastpunkte pro Trägerperiode: die beiden Scheitelwerte. Die Scheitelwerte werden jedoch durch die ungenau bestimmte Signalverzögerung und die dynamischen Effekte nicht exakt getroffen. Dies hat zur Folge, dass nicht

die maximale Amplitude des modulierten Datensignals zur Verfügung steht. Mit viermal so vielen Abtastwerten würden sich die Auswirkungen dieser Ungenauigkeiten in der Wahl der Abtastzeitpunkte reduzieren lassen.

PC-Anbindung

Dieser Prototyp kommuniziert mit dem PC über einen seriellen UART (*Universal asynchronous receiver-transmitter*, engl. universell einsetzbarer asynchroner Empfänger und Sender) mit einer RS-232-Schnittstelle. Die serielle Schnittstelle des PCs ist dabei, aufgrund ihres Aufbaus, typischerweise auf eine maximale Symbolrate von 115200 Baud, d. h. 115200 Symbolen pro Sekunde, beschränkt. Mit acht Datenbits, einem Startbit, einem Stoppbit und keinem Paritätsbit lässt sich also eine maximale Datenrate von 92,16 kBit/s erzielen. Die Datenrate reicht also bereits für die standardisierten Übertragungsraten der kontaktlosen Schnittstelle nicht aus. Im ersten Prototyp werden deshalb auch nur Statusinformationen zur Visualisierung und nicht die tatsächlich übertragenen Daten an den PC weitergegeben.

Weitere einschränkende Faktoren

Ein, in Hinsicht auf eine Weiterentwicklung des bestehenden Prototyps, sehr einschränkender Faktor ist der Systemtakt von 108,48 MHz. Durch diese hohe Taktfrequenz, in Kombination mit dem nicht mehr auf dem aktuellen Stand der Entwicklung stehenden FPGA, ist die Tiefe der realisierbaren kombinatorischen Logik stark begrenzt. Aufwendige Zustandsautomaten oder der Vergleich längerer Bitsequenzen lassen sich nicht für diese hohe Taktfrequenz implementieren.

Eine ähnliche Einschränkung weist das CPLD auf. Der limitierende Faktor ist dabei allerdings nicht die Taktfrequenz, sondern die verfügbaren Logikressourcen.

Für zukünftige Demonstrationsanwendungen gibt es noch ein weiteres Problem: Der Demonstrator unterstützt nur eine Übertragungsrichtung, den Downlink. Es gibt jedoch keine kontaktlose Kommunikationsmöglichkeit vom Lesegerät zum Transponder.

Kapitel 4

System zur Nutzdatenübertragung im Downlink

Aus den bisher betrachteten Grundlagen und Konzepten wird ein Demonstrator entwickelt, mit dem eine Datenübertragung mit bis zu 6,78 MBit/s möglich ist. In diesem Kapitel werden die Anforderungen an die einzelnen Systemkomponenten definiert und in ein entsprechendes System umgesetzt.

4.1 Überblick über die Systemkomponenten

Die zentralen Bestandteile des Systems sind der Transponder und das Lesegerät, die über die kontaktlose Schnittstelle kommunizieren. Ziel ist es, Daten mit einem PC auf einen Transponder zu speichern und anschließend vom Transponder über die kontaktlose Schnittstelle auf einen anderen PC zu übertragen. Das System aus Transponder und Lesegerät muss also um mehrere Bestandteile erweitert werden:

- Es ist eine Anbindung des Lesegerätes an einen PC notwendig.
- Am PC müssen die Daten empfangen und ausgewertet werden.
- Der Transponder benötigt einen Datenspeicher.
- Der Datenspeicher des Transponders muss von einem PC aus beschreibbar sein.

Abbildung 4.1 zeigt den Zusammenhang zwischen diesen Komponenten.

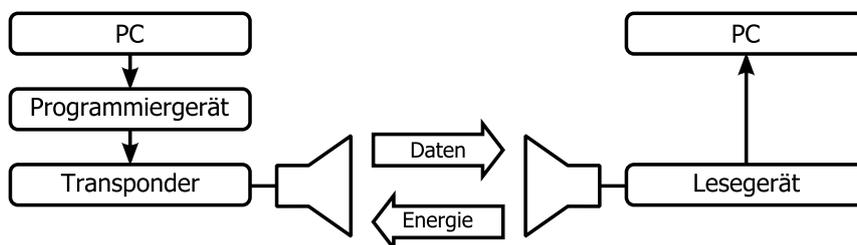


Abbildung 4.1: Komponenten und Verkettung der Übertragungsstrecke von einem PC zu einem anderen PC über das VHD-System

4.2 PC-Anbindung

Die serielle Schnittstelle des bestehenden Prototyps erreicht (unter Microsoft Windows mit gängiger PC-Hardware) Datenraten im Bereich von 100 kBit/s. Damit ist sie für hohe Datenraten ungeeignet. Es muss daher eine alternative Schnittstelle gefunden werden.

4.2.1 Anforderungen

Die Anforderungen an eine solche Schnittstelle sind vielfältig. Allen voran steht, neben geringen Kosten, die Erreichbarkeit höherer Datenraten. Über die kontaktlose Schnittstelle werden Übertragungsraten bis zu 6,78 MBit/s erzielt. Es muss deshalb eine Geschwindigkeit von mindestens 6,78 MBit/s möglich sein, um die über das HF-Feld empfangenen Daten an einen PC weiterzuleiten.

Natürlich sind für die neue PC-Anbindung zumindest dieselben Fähigkeiten wie für die bestehende Kommunikationsschnittstelle notwendig. Konkret bedeutet das, dass eine Übertragung in beide Richtungen, also sowohl vom Lesegerät zum PC, als auch vom PC zum Lesegerät, möglich sein muss. Darüber hinaus ist eine Flusskontrolle wichtig, um Datenverluste durch überfüllte Empfangspufferspeicher zu vermeiden.

Es muss eine Schnittstelle ausgewählt werden, die in aktuell verfügbarer PC-Hardware vorhanden ist. Weiters soll der PC-seitige Implementierungsaufwand so gering wie möglich bleiben. Eventuell notwendige Treiber müssen deshalb bereits verfügbar sein und ein einfaches Interface bereitstellen.

Am verwendeten FPGA-Entwicklungsboard steht nur eine geringe Zahl von ungenutzten Ein-/Ausgabepins zur Verfügung. Das FPGA XC2S150 stellt selbst keine geeignete PC-Schnittstelle bereit. Nachdem eine Integration in den bestehenden Prototyp vorgesehen ist, muss diese Schnittstelle also mit den (wenigen) vorhandenen Pins auskommen.

Abschließend soll auch der Implementierungsaufwand für eine eventuell notwendige Schaltung und für die Einbindung in den VHDL-Entwurf gering gehalten werden.

4.2.2 Auswahl

Im ersten Schritt gilt es eine passende PC-Schnittstelle auszuwählen. In Tabelle 4.1 werden einander verschiedene gebräuchliche Ein-/Ausgabeschnittstellen gegenübergestellt. Die serielle Schnittstelle wurde bereits zuvor aus-

Tabelle 4.1: Vergleich verschiedener PC-Schnittstellen

Schnittstelle	Verfügbarkeit	Geschwindigkeit
Seriell (RS-232/UART)	–	100 kBit/s
Parallelport (IEEE 1284)	–	2 MBit/s
USB (USB 2.0)	++	480 MBit/s
Firewire (IEEE 1394)	+	3200 MBit/s
Ethernet (IEEE 802.3)	++	10 GBit/s

geschlossen. Der Parallelport ist nur für Geschwindigkeiten bis 2 MBit/s geeignet und scheidet daher ebenfalls aus. Für USB, Firewire und Ethernet stellt eine Datenrate von 6,78 MBit/s prinzipiell kein Problem dar. Um Ethernet in Kombination mit dem Betriebssystem Microsoft Windows sinnvoll einzusetzen muss das Übertragungsprotokoll auf dem Netzwerkprotokoll TCP/IP aufsetzen. Die Entwicklung oder Adaptierung eines solchen Protokollstacks für das FPGA ist darum zu aufwändig. Für Firewire ergab die Recherche ebenfalls keine, auf PC- und FPGA-Seite, einfach implementierbare Lösung.

Von den verglichenen Schnittstellen bleibt also nur USB (*Universal Serial Bus*). Basierend auf USB gibt es eine Vielzahl von fertigen Lösungen. Diese reichen von USB-Steuerbausteinen über Mikrocontroller mit integrierter USB-Schnittstelle bis zu vollständigen Konvertern zwischen USB und einer zum FPGA kompatiblen Schnittstelle. Das Ansteuern eines USB-Controllers ist aufwändig. Deshalb konzentrierten sich die weiteren Recherchen auf fertige Konverter-ICs. Ein Großteil dieser Lösungen unterstützt zwar nicht die volle Datenrate von USB 2.0 (480 MBit/s), jedoch sind mit USB 1.0 Geschwindigkeiten bis zu 12 MBit/s erreichbar.

FTDI bietet eine Reihe von ICs, welche PC-seitig wie eine gewöhnliche serielle Schnittstelle angesprochen werden können. Für alle diese ICs steht außerdem ein Treiber zur Verfügung. Dieser Treiber erschließt die Zusatzfunktionen der Chips. Die API (*Application programming interface*, Schnittstelle zwischen Treiber und Anwendungsebene) ist für alle FTDI ICs einheitlich und ähnelt in ihrer Struktur und ihrer Namenskonvention der API für die serielle Schnittstelle. Aus diesem Grund wurden verschiedene dieser Produkte auf ihre Eignung untersucht.

Tabelle 4.2 vergleicht mehrere ICs des Herstellers FTDI. FT8U232AM, FT232B und FT232R stellen eine serielle UART-Schnittstelle zur Verfügung. In der bestehenden VHDL-Implementierung ist bereits ein serieller UART

Tabelle 4.2: Vergleich verschiedener Produkte des Herstellers FTDI

Typ	Verfügbarkeit	Geschwindigkeit	I/O-Pins zum FPGA	zusätzliche Komponenten
FT8U232AM	–	1,6 MBit/s	6	Quarz, Widerstände
FT8U245AM	–	8,0 MBit/s	12	Quarz, Widerstände
FT232B	+	2,4 MBit/s	6	Quarz, Widerstände
FT245B	+	8,0 MBit/s	12	Quarz, Widerstände
FT232R	+	2,4 MBit/s	6	keine
FT245R	+	8,0 MBit/s	12	keine

Quelle: <http://www.ftdichip.com/>

implementiert, daher wäre die Verwendung dieser Mikrochips mit dem geringsten Implementierungsaufwand verbunden. Allerdings sind die erreichbaren Übertragungsraten zu niedrig. FT8U245AM, FT245B und FT245R haben eine parallele 8-Bit-Schnittstelle. Es müsste deshalb eine Ansteuerung dieser Schnittstelle für das FPGA programmiert werden. Alle diese ICs erreichen die notwendige Geschwindigkeit von 6,78 MBit/s. Es wurde der FT245R ausgewählt, weil, abgesehen von ein paar Kondensatoren, keine externen Komponenten notwendig sind.

4.2.3 Schaltungsentwurf

Basierend auf dem FT245R wurde eine Schaltung entwickelt und als PCB (*Printed circuit board*, engl. gedruckte Schaltung) realisiert. Der Schaltplan ist in Abbildung 4.2 dargestellt. Acht Datenleitungen (D0 bis D7) und vier Steuerleitungen (RXF#, RD#, TXE# und WR) verbinden den Schnittstellenchip mit dem FPGA. Außerdem wird die Schaltung über das FPGA-Entwicklungsboard mit Strom versorgt.

4.2.4 Implementierung für das FPGA

Mit dem FT245R lassen sich Daten byteweise senden und empfangen, wobei sich der IC wie ein FIFO-Speicher (*First in, first out*) verhält. D. h. die Daten werden in derselben Reihenfolge abgearbeitet, in der sie in den Speicher geschrieben wurden. Für jede dieser Übertragungsrichtungen stehen zwei Steuerleitungen für ein Handshakeprotokoll zur Verfügung. Das Datenblatt des FT245R gibt diese Protokolle als Zeitdiagramme (Abb. 4.3) an.

Wenn im Empfangs-FIFO-Speicher Daten zum Lesen bereitstehen, dann wird das Signal RXF# vom FT245R auf den Low-Pegel (logische 0) gezogen. Daraufhin kann das FPGA mit einer fallenden Flanke am Signal RD# die Empfangsbereitschaft signalisieren. Mit einer steigenden Flanke am Signal RD# liest das FPGA das Datenbyte von den acht Datenleitungen ein. Anschließend zieht der FT245R das Signal RXF# wieder auf einen High-Pegel

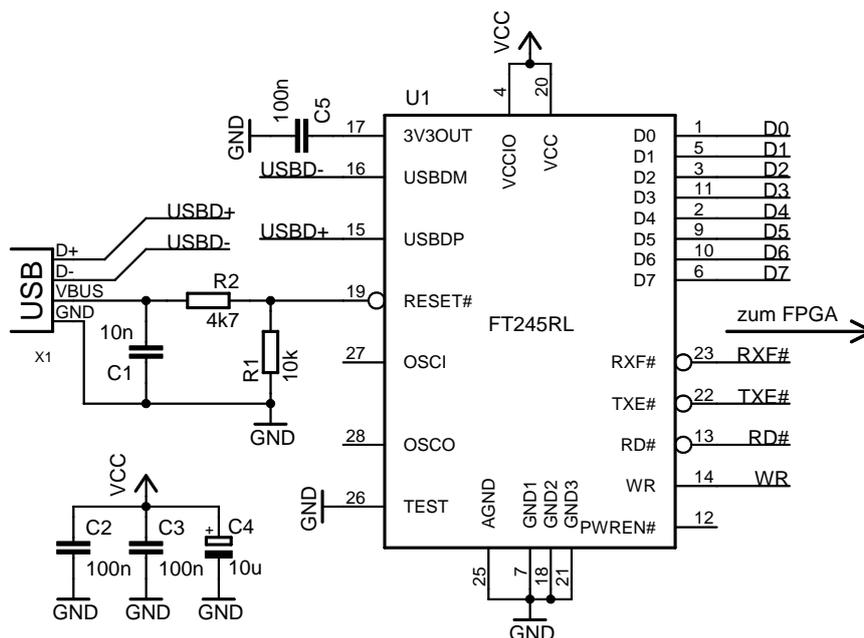


Abbildung 4.2: Beschaltung des USB-Schnittstellen-ICs

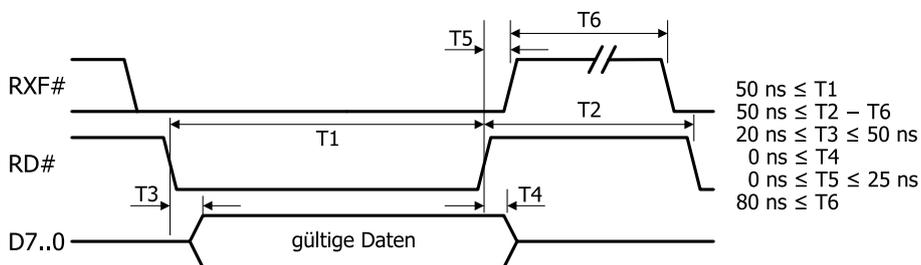
(logische 1).

Wenn der Sende-FIFO-Speicher Daten aufnehmen kann, dann liegt am Signal TXE# vom FT245R ein Low-Pegel an. In diesem Zustand kann das FPGA mit einer steigenden Flanke am Signal WR die Sendebereitschaft signalisieren und ein Datenbyte an den acht Datenleitungen anlegen. Bei einer fallenden Flanke am Signal WR liest der FT245R das Datenbyte ein. Anschließend zieht der FT245R das Signal TXE# so lange auf einen High-Pegel, bis der Sende-FIFO-Speicher erneut Daten aufnehmen kann.

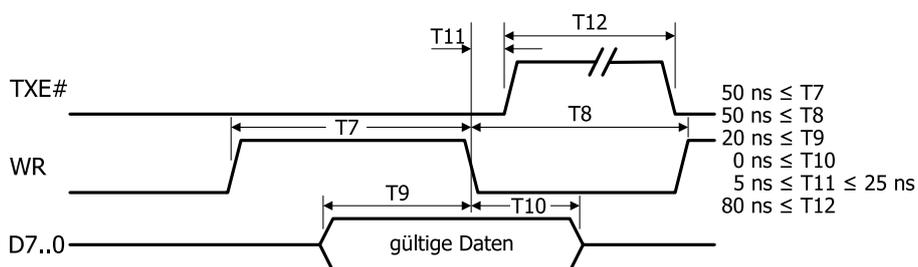
Für das FPGA wurde die Anbindung des Schnittstellen-ICs in der Form eines endlichen Zustandsautomaten mit implizitem Datenpfad (FSMD) implementiert. Abbildung 4.4 zeigt diesen Zustandsautomaten. Dieser ist in einen Sende- und einen Empfangspfad unterteilt. Wenn das FPGA empfangsbereit ist und im FT245R empfangbare Daten vorhanden sind, wird der Empfangspfad betreten. Andernfalls wird geprüft, ob Daten zu Senden sind. Ist dies der Fall und ist der FT245R bereit Daten entgegenzunehmen, dann wird der Sendepfad durchlaufen.

Der Empfänger beginnt damit die Dauer T2, also die minimale Pause zwischen zwei Lesezyklen, abzuwarten. Anschließend wird der Lesepuls mit der Dauer T1 auf dem Signal RD# ausgegeben. Mit dem Ende des Lesepulses wird das Datenbyte von den Datenleitungen in einen Zwischenspeicher eingelesen. Abschließend wird der zwischengespeicherte Wert über ein Handshake (Abb. 4.5) an das angeschlossene VHDL-Design weitergereicht.

4. System zur Nutzdatenübertragung im Downlink



(a) Lesezugriff (Übertragung vom USB zum FPGA)



(b) Schreibzugriff (Übertragung vom FPGA zum USB)

Abbildung 4.3: Zeitdiagramm des Handshakeprotokolls [5]

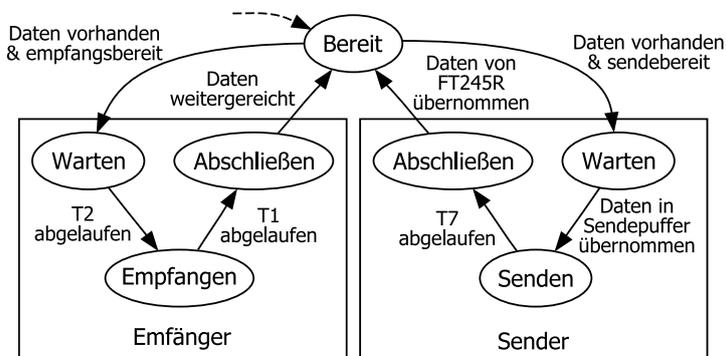


Abbildung 4.4: Zustandsautomat der FT245R-Anbindung

Der Sender beginnt damit, die Daten vom angeschlossenen VHDL-Design mittels des in Abbildung 4.6 dargestellten Handshakes in einen Zwischenspeicher zu übernehmen. Anschließend wird der Schreibpuls mit der Dauer T7 auf dem Signal WR ausgegeben. Gleichzeitig wird das zwischengespeicherte Datenbyte auf die Datenleitungen gelegt. Abschließend wird darauf gewartet, dass der FT245R, mit dem Signal TXE#, die Übernahme der Daten quittiert.

Programm 4.1 zeigt die Schnittstelle der VHDL-Implementierung. Dabei

4. System zur Nutzdatenübertragung im Downlink

44

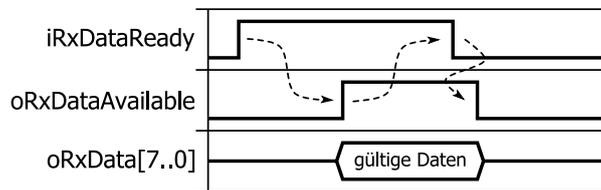


Abbildung 4.5: Handshake zur Datenweitergabe an den angeschlossenen VHDL-Entwurf

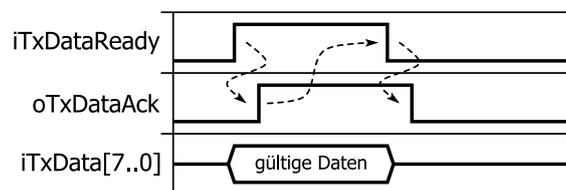


Abbildung 4.6: Handshake zur Datenübernahme vom angeschlossenen VHDL-Entwurf

Programm 4.1: Schnittstelle der FT245R-Anbindung in VHDL

```

1  entity FT245R_FIFO is
2    generic (
3      gClkPeriod : natural := cClkPeriod);
4    port (
5      iClk          : in std_ulogic;
6      inResetAsync : in std_ulogic;
7
8      -- FT245R Schnittstelle
9      ioData : inout std_logic_vector(7 downto 0);
10     onRd   : out  std_ulogic;
11     inRxF  : in   std_ulogic;
12     oWr    : out  std_ulogic;
13     inTXE  : in   std_ulogic;
14
15     -- interne Schnittstelle: Empfänger
16     iRxDataReady : in  std_ulogic;
17     oRxData      : out std_logic_vector(7 downto 0);
18     oRxDataAvailable : out std_ulogic;
19
20     -- interne Schnittstelle: Sender
21     iTxDat      : in  std_logic_vector(7 downto 0);
22     iTxDatReady : in  std_ulogic;
23     oTxDataAck  : out std_ulogic);
24 end entity FT245R_FIFO;

```

ist iClk ein Taktsignal mit der Periodendauer gClkPeriod. inResetAsync ist ein Resetsignal. Die anderen Ports entsprechen den Signalen des FT245R und der intern verwendeten Schnittstelle aus den Beschreibungen zum Hand-

shake.

4.2.5 Implementierung für den PC

Obwohl der FT245R unter Microsoft Windows wie eine serielle Schnittstelle installiert und angesprochen werden kann, ist es für die volle Datenrate von bis zu 8 MBit/s notwendig, einen speziellen Treiber zu verwenden. Dieser wird, genauso wie der serielle Schnittstellentreiber, vom Hersteller FTDI bereitgestellt. Der Treiber stellt mehrere APIs für die Programmierung in unterschiedlichen Umgebungen zur Verfügung. Zum einen kann die Anwendungsschnittstelle in den Programmiersprachen C/C++ über Funktionsaufrufe genutzt werden. Zum anderen gibt es für die Verwendung in der Microsoft .NET-Umgebung eine eigene Klasse, deren Instanzen die angeschlossenen FTDI-Schnittstellen-ICs repräsentieren.

Nachfolgend sind die wichtigsten Funktionsaufrufe der Programmierschnittstelle im Überblick beschrieben:

- Mit der Funktion `FT_ListDevices` kann man alle installierten FTDI-Schnittstellen-ICs auflisten und deren Seriennummern und Beschreibungen auslesen.
- Die Funktion `FT_OpenEx` wird verwendet, um ein Gerät anhand der Seriennummer oder der Beschreibung zu öffnen.
- Mit der Funktion `FT_Purge` lassen sich der Sende- und der Empfangspufferspeicher des FT245R leeren.
- Mit einem Aufruf von `FT_GetStatus` liest man den Füllstand des Sende- und des Empfangspufferspeichers aus.
- Mit der Funktion `FT_Write` werden ein oder mehrere Byte in den Sendepuffer geschrieben.
- Mit der Funktion `FT_Read` werden ein oder mehrere Byte aus dem Empfangspuffer gelesen.
- Die Funktion `FT_Close` wird verwendet, um ein geöffnetes Gerät zu schließen.

Abbildung 4.7 zeigt den typischen Ablauf bei der Verwendung dieser Funktionen.

4.2.6 Performance

Um sicherzustellen, dass der FT245R funktioniert und die notwendige Geschwindigkeit von 6,78 MBit/s erreicht, wurde eine Anwendung zur Performancemessung entwickelt. Diese Anwendung misst die maximal mögliche

4. System zur Nutzdatenübertragung im Downlink

46

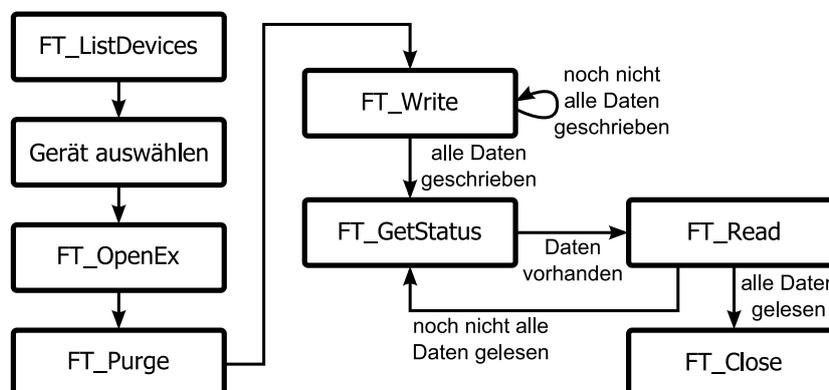


Abbildung 4.7: Ablaufdiagramm einer typischen Interaktion mit der FTDI-API

Übertragungsgeschwindigkeit vom FPGA zum PC. Ein entsprechend entworfenes VHDL-Design sendet ständig Daten an den FT245R. Am PC wurde ein Programm geschrieben, das die Daten empfängt, überprüft und die erreichte Datenrate ausgibt.

Mit Hilfe dieser Messung konnte festgestellt werden, dass bei der Übertragung keine Daten verloren gehen. Das Messergebnis ist in Abbildung 4.8 dargestellt. Die erwartete Datenrate von 8 MBit/s konnte trotz mehrerer

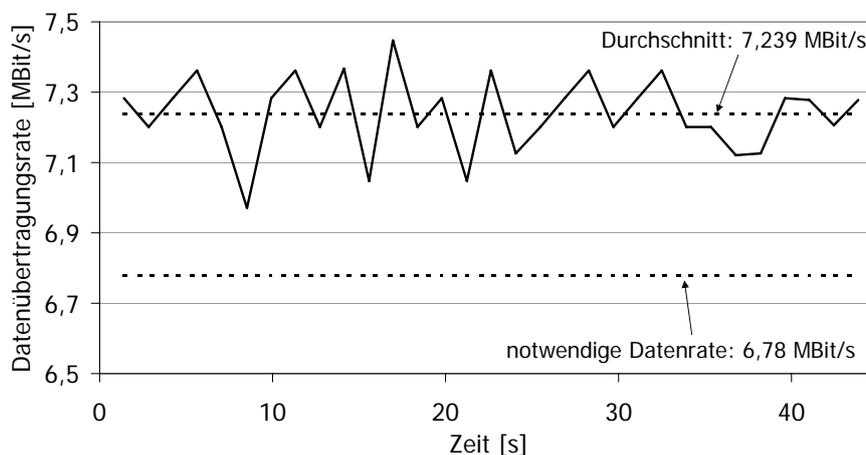


Abbildung 4.8: Performancemessung der Übertragung vom FPGA zum PC

Optimierungen des verwendeten C++-Programms nicht erreicht werden. Mit einer durchschnittlichen Übertragungsrate von 7,239 MBit/s ist dennoch die Anforderung (6,78 MBit/s) erfüllt.

4.3 Transponder

Der bestehende passive Transponder ist für die Übertragung von Nutzdaten ungeeignet. Das CPLD verfügt nicht über ausreichende Logikressourcen. Zudem ist kein Datenspeicher vorhanden. Aus diesem Grund wurde ein neuer Transponder entwickelt.

4.3.1 Anforderungen

Die wichtigste Anforderung an den neuen Transponder ist das Speichern von Daten. Dieser nicht-flüchtige Speicher muss mit mindestens 6,78 MBit/s auslesbar sein. Die Schnittstelle des Speichers soll möglichst einfach sein. Weil das CPLD nur eine geringe Anzahl an I/O-Pins aufweist und durch die Verwendung vieler I/O-Pins die verfügbaren Logikressourcen stark eingeschränkt werden, ist es notwendig, den Speicher mit einer geringen Anzahl an Signalleitungen zu steuern. Auch der Aufwand zum Auslesen der Daten soll gering sein. Neben dem Auslesen des Speichers über das CPLD soll das Lesen und Schreiben des Speichers über eine externe Programmierschnittstelle möglich sein. Versorgungsspannung und Signalpegel müssen zu jenen des CPLDs passen.

Der Transponder soll abgesehen von der Übertragung von Nutzdaten noch verschiedene Messungen bezüglich der Leistungsaufnahme der einzelnen Komponenten ermöglichen. Aus diesem Grund muss jeder Teil von der Stromversorgung trennbar sein.

Beim bestehenden Transponder wird die Versorgungsspannung über eine Zenerdiode auf 3,3 Volt begrenzt. Messungen ergaben, dass die Spannung trotzdem auf über 4 Volt ansteigen kann. Für das CPLD des bestehenden Transponders ist ein Betrieb mit einer Kernversorgungsspannung von 1,8 Volt und einer I/O-Versorgungsspannung zwischen 1,5 Volt und 3,3 Volt spezifiziert. Deshalb ist eine exakte Spannungsregelung notwendig, die sowohl eine passende Kernversorgungsspannung als auch eine passende I/O-Versorgungsspannung liefert.

Nachdem die erfolgreiche Übertragung von Daten im Mittelpunkt dieser Diplomarbeit steht, ist die Entwicklung eines semi-passiven Transponders ausreichend. Dennoch sollten Komponenten mit möglichst geringer Leistungsaufnahme ausgewählt werden, um in Zukunft auch einen passiven Betrieb zu ermöglichen.

4.3.2 Auswahl der Komponenten

Als erstes muss ein geeigneter Datenspeicher gefunden werden. Eine Recherche ergab, dass Flash-Speicher ausreichende Speichergrößen (ab einem Megabyte) aufweisen. Nachdem Speicher mit parallelen Schnittstellen ausgeschlossen werden mussten, weil, in Kombination mit einem CPLD, nur

eine geringe Anzahl an I/O-Pins verfügbar ist, bleiben nur serielle Schnittstellen übrig. Bei Flash-Speichern gibt es davon im Wesentlichen zwei Arten: I²C (*Inter-integrated circuit bus*) und SPI (*Serial peripheral interface bus*). Während I²C nur bis 3,4 MBit/s spezifiziert ist [25], gibt es Speicher deren SPI-Schnittstelle mit über 100 MHz getaktet werden kann.

SPI-Flash-Speicher sind mit Größen bis zu 16 Megabyte erhältlich. Um die Kosten gering zu halten wurde für den neuen Transponder der Speicher-IC M25P32 von Numonyx ausgewählt. Dieser hat ein Megabyte Speicherplatz und erlaubt Taktfrequenzen bis 75 MHz (und damit Geschwindigkeiten bis 75 MBit/s). Der Speicher kann in einem Zug gelesen und seitenweise geschrieben werden, wobei eine Speicherseite 256 Byte umfasst. Der IC benötigt eine Versorgungsspannung von 3 bis 3,3 Volt und benötigt beim Lesen einen Betriebsstrom von 4 Milliampere.

Der bestehende Transponder enthält ein CPLD vom Typ CoolRunner-II XC2C64A. Dieses CPLD verfügt nur über 64 1-Bit-Register. Erste Tests mit komplexeren VHDL-Beschreibungen haben gezeigt, dass die verfügbaren Logikressourcen dieses CPLDs rasch aufgebraucht sind. Aus diesem Grund ist für den neuen Transponder ein größeres CPLD notwendig. Die Suche wurde dabei auf die CoolRunner-II-Serie von Xilinx beschränkt. Diese Serie besitzt laut [12] eine sehr geringe Leistungsaufnahme und ist somit für den Betrieb in einem passiven Transponder gut geeignet. Um verschiedene CPLDs dieser Serie zu vergleichen, enthält der neue Transponder drei verschiedene CPLDs (XC2C64A, XC2C128 und XC2C256). Tabelle 4.3 zeigt die spezifizierte Stromaufnahme dieser ICs.

Tabelle 4.3: CPLDs der CoolRunner-II-Serie von Xilinx [30]

Typ	Stromaufnahme	
	typisch ^a	maximal ^b
XC2C64A	17 μ A	5 mA
XC2C128	19 μ A	10 mA
XC2C256	21 μ A	27 mA

^abei 0 MHz und 25 °C

^bbei 50 MHz und 70 °C

Die Entwicklung eines neuen Transponders basiert auf dem bestehenden Konzept nach [12]. Die Antenne und die Schaltungen zur Taktableitung und zur Modulation werden unverändert übernommen. Die Schaltung zur Begrenzung der Versorgungsspannung erzielt nicht die notwendigen Spannungspegel und muss daher angepasst werden. Der Speicher erfordert Logikpegel und eine Versorgungsspannung im Bereich von 3,3 Volt. Die CPLDs benötigen jeweils eine Kernversorgungsspannung von 1,8 Volt. Spannungsregler sind ICs, die eine Eingangsspannung mit einem größeren

Spannungsbereich auf eine bestimmte Ausgangsspannung einstellen. Damit lässt sich die, durch eine Zenerdiode begrenzte, Versorgungsspannung auf die benötigten Werte regeln. Es werden ein TPS79118 und ein TPS79133 von Texas Instruments verwendet. Das sind Spannungsregler mit geringem Spannungsabfall (LDO, *Low-dropout regulator*).

Als Alternative zur Versorgung über das HF-Feld kann der Transponder auch mit einer Batterie oder einem Netzteil versorgt werden.

4.3.3 Schaltungsentwurf

Unter Berücksichtigung der Anforderungen wurden die ausgewählten Komponenten zu einem neuen Transponder zusammengesetzt. Der schematische

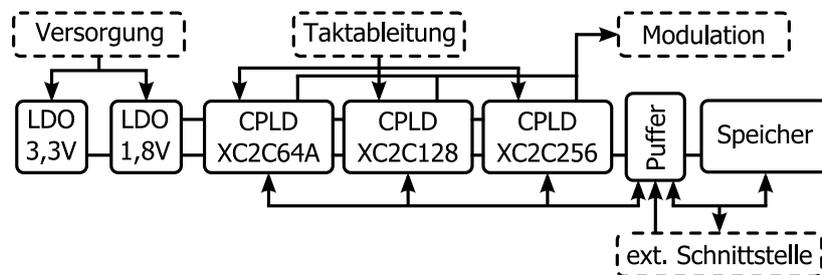


Abbildung 4.9: Schematischer Aufbau des Transponders

Aufbau ist in Abbildung 4.9 dargestellt.

Der neue Schaltungsentwurf setzt auf dem analogen Front-End des bestehenden Transponders (Abb. 3.3) auf. Die, mit einer Zenerdiode, begrenzte Versorgungsspannung wird mit je einem LDO-Regler auf 3,3 Volt und auf 1,8 Volt geregelt. Die CPLDs und der Speicher sind einzeln über Drahtbrücken mit der Stromversorgung, den Modulationstransistoren, dem Speicher und untereinander verbindbar. Das ermöglicht Messungen zur Stromaufnahme. Zudem kann im Nachhinein bestimmt werden, welches CPLD für die Modulation und das Auslesen des Speichers verwendet wird. Auch die Kombination mehrerer CPLDs ist möglich. Die CPLDs sind zudem über einen Bustransceiver durch ein Steuersignal vom Speicher trennbar. Dieses Steuersignal und die SPI-Bus-Schnittstelle des Speichers werden aus dem Transponder herausgeführt, um ein externes Lesen und Schreiben des Transponderspeichers zu ermöglichen.

4.3.4 Implementierung für das CPLD

Die Ablaufsteuerung und das Auslesen und Senden der Daten findet im CPLD statt. Der VHDL-Entwurf besteht aus vier wesentlichen Bestandteilen: einer zentralen Ablaufsteuerung, dem Auslesen des Flash-Speichers, der Verarbeitung der ausgelesenen Daten und dem Manchestercodieren der

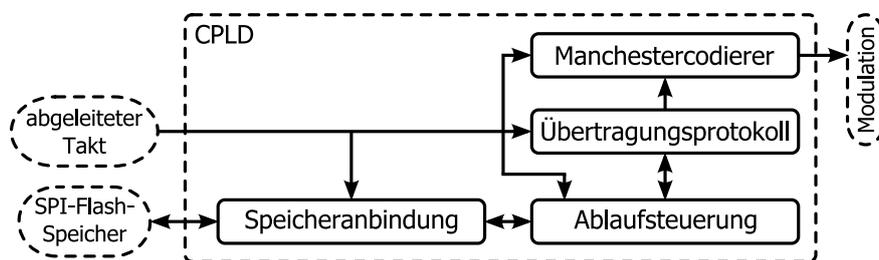


Abbildung 4.10: Blockschaltbild des Transponders

Daten. Abbildung 4.10 zeigt diese Verarbeitungsblöcke. Die folgenden Abschnitte beschreiben die Aufgabe und die Funktionsweise der einzelnen Komponenten. Wie die gelesenen Daten vor dem Senden verarbeitet werden, wird anhand des Übertragungsprotokolls in Kapitel 5 erklärt.

Manchestercodierung

Der Manchestercodierer ist ein einfacher Zustandsautomat, welcher ein Bit einliest und anschließend über zwei Taktzyklen hinweg einmal invertiert und einmal ohne Invertierung ausgibt. Dadurch entsteht ein manchestercodierter Datenstrom.

Speicheranbindung

Der Flash-Speicher hat eine SPI-Bus-Schnittstelle. Der SPI-Bus besteht aus einer Takt- und einer Datenleitung vom Busmaster zu allen Busteilnehmern, einer gemeinsamen Datenleitung von den Busteilnehmern zum Busmaster und jeweils einer Auswahlleitung vom Busmaster zu den einzelnen Busteilnehmern. Das CPLD übernimmt bei der Kommunikation die Rolle des Busmasters.

Zur Kommunikation mit EEPROM- und Flash-Speichern über SPI gibt es einheitliche Befehle. Dadurch kann der Speicher im Nachhinein einfach gegen einen größeren Speicher getauscht werden. Jeder Befehl beginnt mit dem Setzen der Auswahlleitung auf einen Low-Pegel und mit der Aktivierung des Chips. Anschließend wird der Bittakt auf der Taktleitung ausgegeben. In dieser Taktperiode erfolgt auch die Ein- und Ausgabe auf den Datenleitungen. Der Master sendet einen Befehl und wartet anschließend auf eine optionale Antwort. Jeder Befehl besteht dabei aus einem Instruktionsbyte und optionalen Parametern bzw. Daten. Abschließend zieht der Master die Auswahlleitung wieder auf einen High-Pegel um das Befehlsende zu signalisieren. Tabelle 4.4 gibt einen Überblick über die Befehle zum Lesen eines SPI-Flash-Speichers. READ liest die Daten aus dem Speicher aus. FAST_READ ermöglicht das Lesen mit einer höheren Taktfrequenz.

Tabelle 4.4: SPI-Befehle zum Lesen eines EEPROM-/Flash-Speichers

Instruktion	Parameter	Antwort
READ (0x03)	Adresse (3 Byte)	Daten (1 bis ∞ Byte)
FAST_READ (0x0B)	Adresse (3 Byte) + 1 Füllbyte	Daten (1 bis ∞ Byte)

Das Lesen des Speichers wurde mit einem Zustandsautomat realisiert. Der Automat übernimmt die Kommunikation über den SPI-Bus und ist durch die Ablaufsteuerung über ein Start-Stopp-Signal steuerbar. Als Reaktion auf das Startsignal wird damit begonnen, den Status des Speichers zu prüfen. Sobald der Speicher bereit ist, wird ein Lesebefehl für die Speicheradresse 0 gesendet. Der Speicher beantwortet den Lesebefehl indem er alle Daten ab dieser Adresse sendet, wobei die Übertragung nach dem Ende des Speichers wieder bei Adresse 0 fortsetzt. Der Master bricht den Lesevorgang ab, sobald die Ablaufsteuerung den Stoppbefehl gibt.

Ablaufsteuerung

Die Ablaufsteuerung koordiniert das Lesen und die Weiterverarbeitung der Daten. Der Ablauf beginnt mit einem Startsignal an die Speicheranbindung. Die ersten drei ausgelesenen Bytes müssen die Größe des Speichers angeben. Anschließend werden die Datenbytes aus dem Speicher gelesen und zum Senden über die VHD-Schnittstelle weiterverarbeitet. Sobald alle Daten aus dem Speicher gelesen sind, bricht die Ablaufsteuerung den Lesevorgang mit dem Stoppsignal ab und startet den gesamten Ablauf von vorne.

4.3.5 Analyse des gesendeten Signals

Die Auswirkungen des modulierenden Transponders auf das Empfangssignal wurden einerseits im Frequenzbereich und andererseits im Zeitbereich gemessen.

Frequenzbereich

Zunächst wurde das Spektrum des Empfangssignals ohne Modulation durch den Transponder aufgezeichnet. Anschließend wurde die Messung mit Modulation durch den Transponder wiederholt. Abbildung 4.11 zeigt die beiden Spektren im Vergleich. Es sind deutlich die durch die Modulation entstehenden spektralen Komponenten rund um das Trägersignal zu erkennen. Wie für die Lastmodulation erwartet, dominiert das Trägersignal.

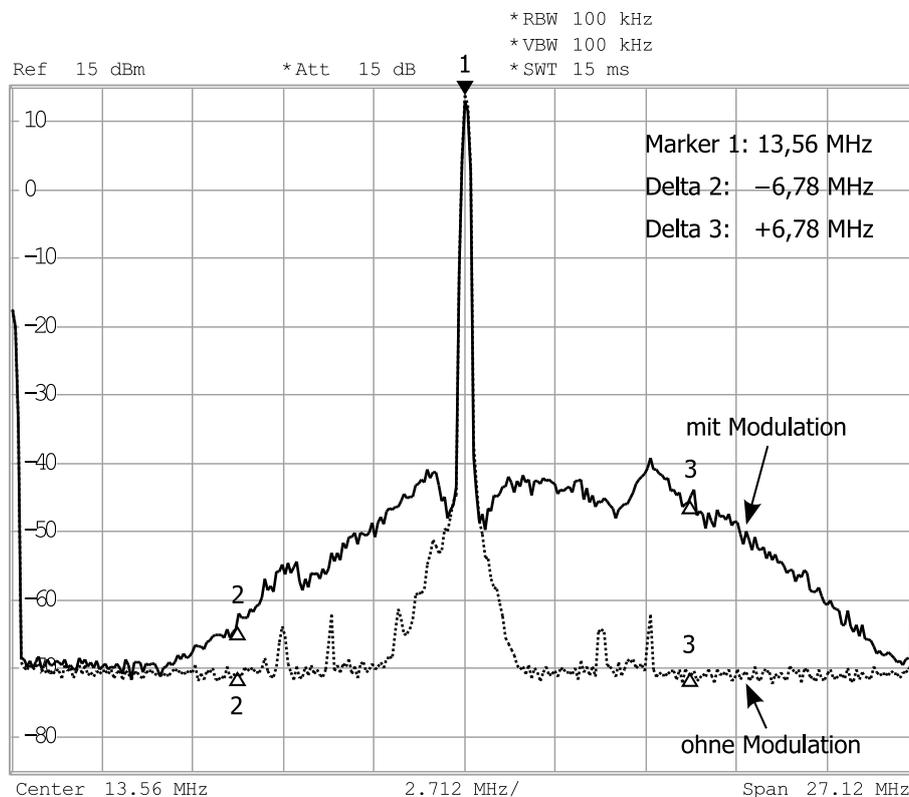


Abbildung 4.11: Spektrum des Empfangssignals

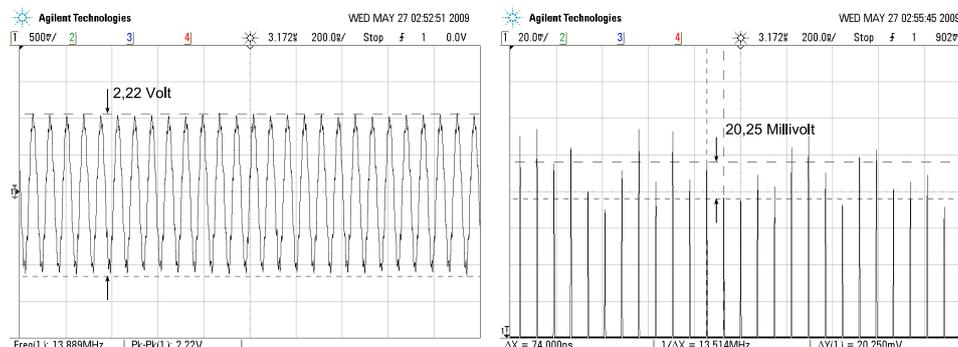
Zeitbereich

Im Zeitbereich wurde der Modulationsindex, d. h. der Anteil der Amplitudenänderung durch die Modulation an der gesamten Signalamplitude untersucht. Abbildung 4.12 zeigt die gesamte Signalamplitude und den Unterschied zwischen modulierten und unmodulierten Phasen. Während die Signalamplitude etwa 2,22 Volt beträgt, misst der Abstand zwischen den Modulationszuständen nur etwa 20 Millivolt. Das entspricht einem Modulationsindex von weniger als einem Prozent.

4.4 Lesegerät

Für das Lesegerät wurde die bestehende Hardware weitgehend übernommen und um die USB-Schnittstelle ergänzt. Eine Trägerunterdrückung durch die Quarzbandsperre findet, wegen der, in Abschnitt 3.3.3, erwähnten Probleme mit der Frequenzstabilität, nicht mehr statt.

Das Lesegerät erzeugt daher ein 13,56-MHz-Trägersignal und digitalisiert das empfangene Signal mit einem Analog-Digital-Umsetzer. Die empfan-



(a) gesamte Signalamplitude

(b) Amplitudenänderung durch Modulation

Abbildung 4.12: Moduliertes Empfangssignal im Zeitbereich

genen Daten werden anschließend über die neu entwickelte USB-Anbindung zu einem PC übertragen. Der PC wertet die Daten anschließend aus.

4.4.1 Implementierung für das FPGA

Die VHDL-Beschreibung des Digitalteils hat durch zahlreiche Änderungen und Anpassungen über die Zeit ihre Struktur und Übersichtlichkeit verloren. Darüber hinaus verwendet das VHDL-Design des ersten Demonstrators einen einheitlichen Systemtakt von 108,48 MHz. Dadurch ist die realisierbare Logik stark eingeschränkt. Aus diesem Grund wurde der gesamte Digitalteil neu aufgebaut. Abbildung 4.13 zeigt die Struktur des neuen Lesegeräts.

Systemtakt

Zunächst wurde die Instanzierung der FPGA-eigenen DLLs so umgestaltet, dass, statt einem einheitlichen Systemtakt von 108,48 MHz, ein allgemeiner 54,24-MHz-Systemtakt und ein 108,48-MHz-Abtasttakt zur Verfügung stehen. Die Beschaltung der DLLs orientiert sich an [31] und ist in Abbildung 4.14 dargestellt. Die erste DLL verdoppelt den externen 27,12-MHz-Takt. Die zweite DLL verwendet diesen 54,24-MHz-Takt als Eingangssignal und gibt den Takt selbst und ein dazu synchrones Taktsignal mit der doppelten Frequenz (108,48 MHz) in je einen globalen Taktpuffer (BUFG) aus. Zur Verbindung der beiden DLLs stand kein globaler Taktpuffer mehr zur Verfügung. Deshalb musste das Taktsignal über einen I/O-Pin aus dem FPGA herausgeführt und über einen Takteingangspin zurück in das FPGA gespeist werden.

4. System zur Nutzdatenübertragung im Downlink

54

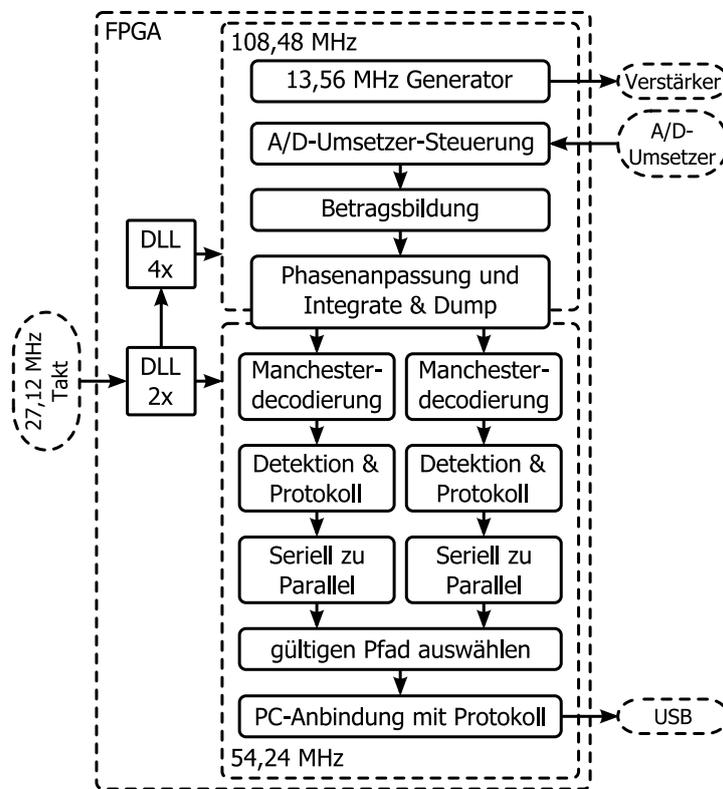


Abbildung 4.13: Blockschaltbild des Digitalteils des Lesegeräts

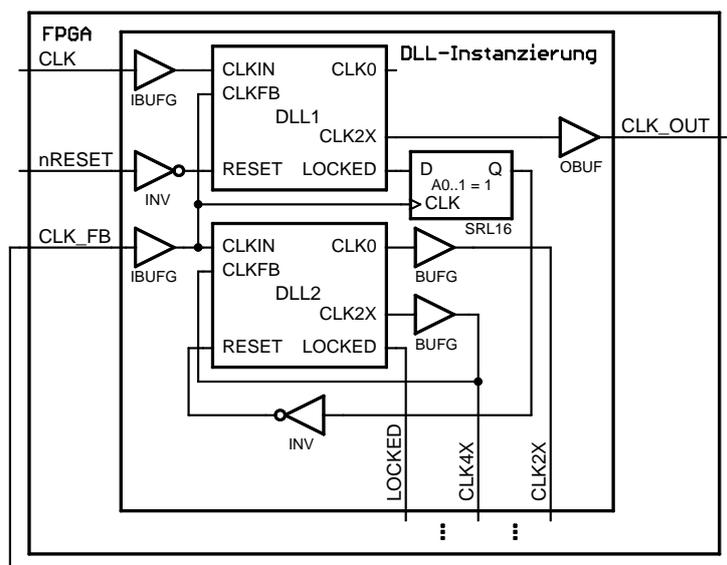


Abbildung 4.14: Blockschaltbild der DLL-Instanzierung

Trägererzeugung

Der 13,56-MHz-Generator erzeugt das Trägersignal indem, nach je vier Perioden des 108,48-MHz-Takts, der Pegel des Ausgangssignals gewechselt wird.

Signalabtastung

Die A/D-Umsetzer-Steuerung tastet das Empfangssignal mit 108,48 MHz ab. Im ersten Demonstrator wurde anschließend nur jeder vierte Abtastwert weiterverwendet, wodurch sich die Abtastrate auf 27,12 MHz reduzierte. Dadurch war das Empfangskonzept sehr ineffizient. Um zumindest brauchbare Abtastwerte zu erhalten, wurden die Abtastwerte (durch grobe Schätzung und manuelles Abstimmen) so ausgewählt, dass sie möglichst nahe bei den Spitzenwerten jeder Periode des Empfangssignals lagen (Abb. 4.15). Im wei-

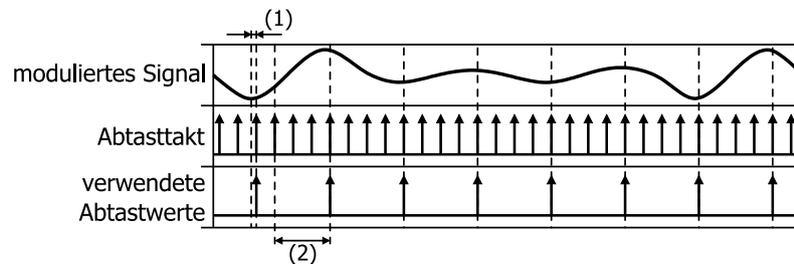


Abbildung 4.15: Auswahl eines der vier Abtastwerte (2) bei einer Reduzierung der Abtastrate auf 27,12 MHz und Abweichung (1) des ausgewählten Abtastwertes vom idealen Abtastzeitpunkt (Scheitelwert)

terentwickelten Prototyp werden nun sieben der acht Abtastwerte genutzt. Um die Auswirkungen von Ungenauigkeiten bei der Einstellung der Abtastzeitpunkte und bei der Zuordnung der Abtastzeitpunkte zu den einzelnen Modulationsphasen zu minimieren, bleibt jeder achte Abtastwert ungenutzt.

Von den Abtastwerten wird der Absolutbetrag gebildet. Der *Integrate-and-Dump*-Block (engl. integrieren und ausgeben) summiert jeweils die Abtastwerte einer Periode auf. Jedes Ergebnis dieser Integration ist ein Symbol des manchestercodierten Empfangssignals. Die Phasen Anpassung stellt dabei sicher, dass immer die Abtastwerte einer Modulationsphase zusammengefasst werden. Dieses Verhalten lässt sich durch die Relation zwischen dem generierten Trägersignal und dem als Startwert gewählten Abtastwert konfigurieren (Abb. 4.16). Dadurch werden die Verzögerungen zwischen dem generierten Trägersignal und dem Empfangssignal sowie die Phasenverschiebung zwischen dem Trägersignal und dem aufmodulierten Signal ausgeglichen. Vereinfachend wird dazu angenommen, dass sich die Phasenlage durch Abstandsänderungen zwischen dem Transponder und der Leseantenne um weniger als eine Periode des Abtasttaktes ändert. Durch diese Annahme

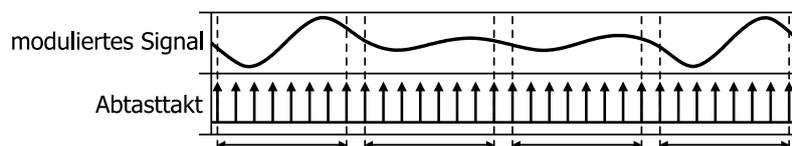


Abbildung 4.16: Kombination der Abtastwerte zu einer Signalperiode

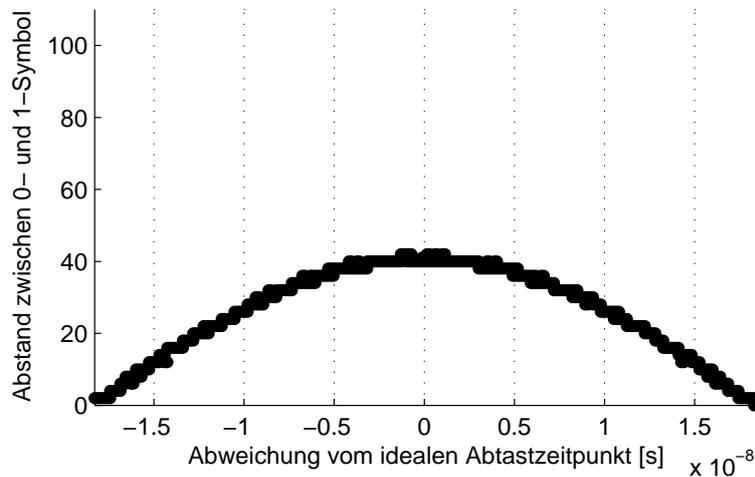
kann die Phasenanpassung unabhängig vom Abstand zwischen dem Transponder und der Leseantenne vorgenommen werden.

Die Abtastung des Empfangssignals mit dem 12-Bit-Analog-Digital-Umsetzer wurde in Matlab modelliert. Dabei wurde untersucht wie sich Abweichungen des Abtastzeitpunktes vom optimalen Zeitpunkt auswirken. Abbildung 4.17 zeigt die Ergebnisse dieser Simulation. Während der Abstand zwischen den beiden Bithälften bei zwei Abtastwerten pro Periode für große Abweichungen vom idealen Abtastzeitpunkt gegen 0 geht, sinkt dieser bei sieben Abtastwerten pro Periode auch bei Abweichungen von bis zu einer viertel Trägerperiode nicht unter 45. Der angegebene Abstand entspricht dabei den im FPGA verwendeten Binärwerten. Im Gegensatz zur Variante mit zwei Abtastwerten ist die Variante mit sieben Abtastwerten daher deutlich robuster gegenüber schlecht eingestellten Abtastzeitpunkten und anderen Störeinflüssen.

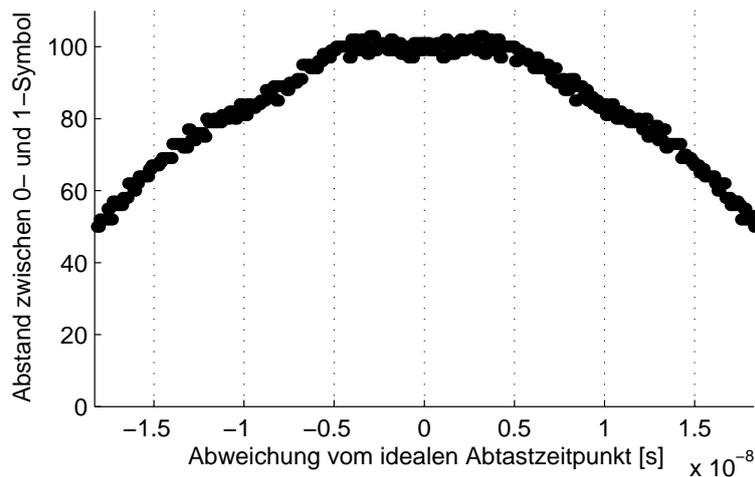
Manchesterdecodierung

Der Manchesterdecodierer vergleicht zwei aufeinanderfolgende Empfangssymbole miteinander und bestimmt so, ob es sich um eine steigende oder eine fallende Signalfanke (und damit um eine übertragene Eins oder Null) handelt. Es gibt also keinen absoluten Schwellenwert, sondern nur die Relation zwischen den verglichenen Werten. Die Decodierung kennt dabei nur die zwei Entscheidungszustände 0 und 1. Der Fall, dass beide Werte exakt gleich sind, wird als extrem unwahrscheinlich angesehen und deshalb vernachlässigt.

Das Empfangssignal ist zum Zeitpunkt der Decodierung noch nicht bit-synchron. Je nachdem, welches Empfangssymbol der Decodierer als erste Bithälfte heranzieht, gibt es zwei Möglichkeiten die Folge von Empfangssymbolen zu decodieren. Abbildung 4.18 zeigt diese Problematik. Aus diesem Grund spaltet sich die Verarbeitung des Empfangssignal ab der Manchesterdecodierung in zwei Pfade auf. Die beiden Manchesterdecodierer arbeiten jeweils um ein Symbol versetzt. Die Synchronisation auf Bitebene muss mit Hilfe des Übertragungsprotokolls erfolgen.



(a) bei zwei Abtastwerten pro Periode



(b) bei sieben Abtastwerten pro Periode

Abbildung 4.17: Auswirkung der Abweichung des tatsächlichen Abtastzeitpunktes vom idealen Abtastzeitpunkt auf die Differenz zwischen dem 0- und dem 1-Symbol (entsprechend dem Binärwert im FPGA) ohne Berücksichtigung von Störeinflüssen bei einer angenommenen Modulationstiefe von einem Prozent

Detektion und Verarbeitung der empfangenen Daten

Dieser Verarbeitungsblock erkennt den Beginn und das Ende des empfangenen Datenstroms und verarbeitet die Daten anhand eines Übertragungsprotokolls. Dieses Übertragungsprotokoll und die Implementierung des Verarbeitungsblocks werden in Kapitel 5 beschrieben.

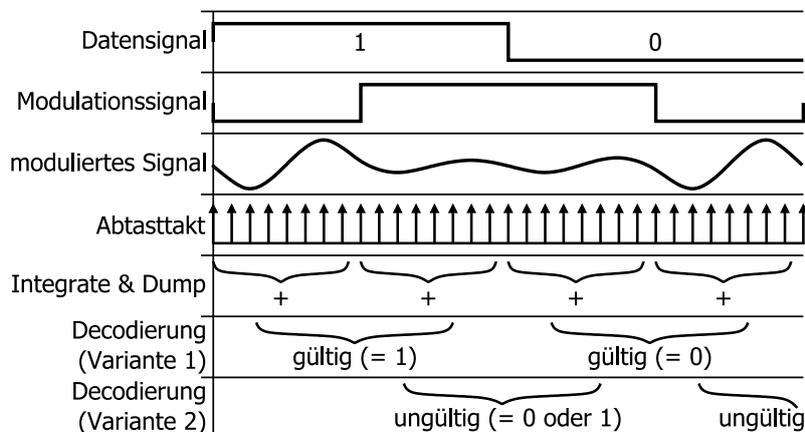


Abbildung 4.18: Zwei verschiedene Möglichkeiten ein manchestercodiertes Signal zu decodieren

Über die Information, ob einer der Empfangspfade einen gültigen Datenstrom detektiert hat, wird ein Pfad ausgewählt und dessen Daten zur USB-Schnittstelle weitergegeben.

Übertragung zum PC

Die PC-Anwendung soll neben den empfangenen Daten auch Statusinformationen über den Zustand der Übertragung erhalten. Zu diesen Informationen zählen das Erkennen und der Verlust der Verbindung mit einem Transponder. Die PC-Anbindung ist byteorientiert. Es wurde daher ein byteorientiertes Protokoll zur Kommunikation mit dem PC entwickelt. Die übertragenen Informationen bestehen dabei aus Daten- und Steuerzeichen. Tabelle 4.5 zeigt die Steuerzeichen dieses Protokolls. ESC ist das Escape-

Tabelle 4.5: Steuerzeichen des Übertragungsprotokolls

Zeichen	Bytedarstellung
ESC	0xFF
LOCK	0xA5
LOST	0xB4
SYNC	0xC3

Zeichen. LOCK signalisiert, dass sich ein Transponder in Reichweite des Lesegerätes befindet. Mit LOST wird dem PC mitgeteilt, dass ein zuvor erkannter Transponder nicht mehr innerhalb der Reichweite des Lesegerätes ist. SYNC gibt an, dass eine Synchronisation zwischen dem Transponder und dem Lesegerät durchgeführt wurde. Die Steuerzeichen (auch Daten mit

demselben Bitmuster wie das Escape-Zeichen) werden durch Voranstellen eines Escape-Zeichens von den Daten abgegrenzt.

Zur Verarbeitung des Protokolls werden die vom Transponder empfangenen Daten in einem FIFO-Puffer zwischengespeichert. Statusinformationen haben eine höhere Priorität als die Daten und werden deshalb sofort (ohne Puffer) an die PC-Anbindung übermittelt. Wenn keine Statusinformationen zu senden sind, werden die Daten aus dem FIFO-Speicher gelesen, eventuell notwendige Escape-Zeichen eingefügt und an die PC-Anbindung weitergegeben.

4.4.2 Implementierung für den PC

Es wurde eine PC-Anwendung zum Empfangen der Daten vom Lesegerät entwickelt. Nachdem das Lesegerät mangels eines Uplink-Kanals nicht in der Lage ist, den vom Transponder kommenden Datenfluss zu steuern, ist auch keine Steuerung des Lesegeräts durch den PC vorgesehen. Die PC-Anwendung liest den Datenstrom des Lesegerätes in einer Endlosschleife ein. Anschließend verarbeitet sie diesen byteweise. Abbildung 4.19 zeigt das Zustandsdiagramm zur Verarbeitung der empfangenen Zeichen. Im empfan-

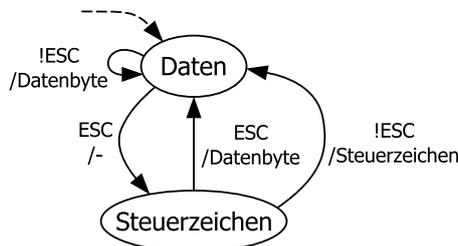


Abbildung 4.19: Zustandsdiagramm der Verarbeitung des Protokolls zwischen Lesegerät und PC

genen Bytestrom wird das Escape-Zeichen gesucht. Wird dieses gefunden, dann wird das darauffolgende Byte als Steuerzeichen behandelt: ESC wird als Datenbyte verarbeitet. LOCK und LOST markieren den Transponder als erkannt bzw. nicht mehr erkannt. SYNC zeigt eine erfolgreich durchgeführte Synchronisation an. Andere Steuerzeichen sind nicht spezifiziert und lösen einen Fehler aus. Alle Zeichen, die keine Steuerzeichen sind (und daher kein vorangestelltes Escape-Zeichen haben) werden als Datenbytes behandelt und zur weiteren Verarbeitung in einer Warteschlange (FIFO-Speicher) abgelegt.

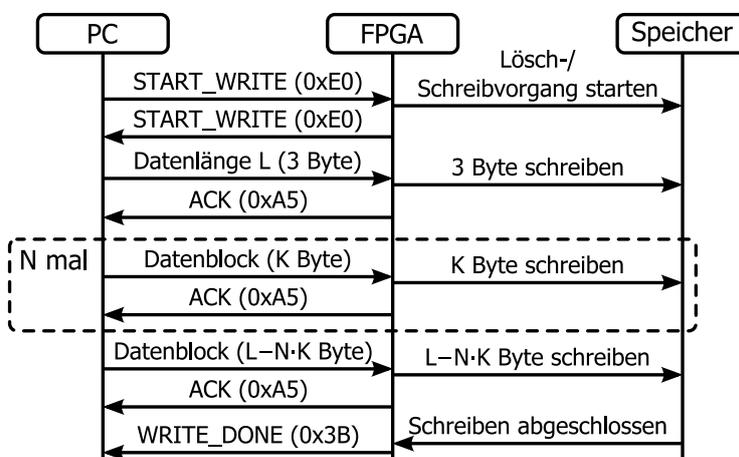
4.5 Programmiergerät für den Transponder

Ein Programmiergerät soll eine möglichst einfache und schnelle Möglichkeit bieten, den Transponderspeicher mit dem PC zu beschreiben und auszule-

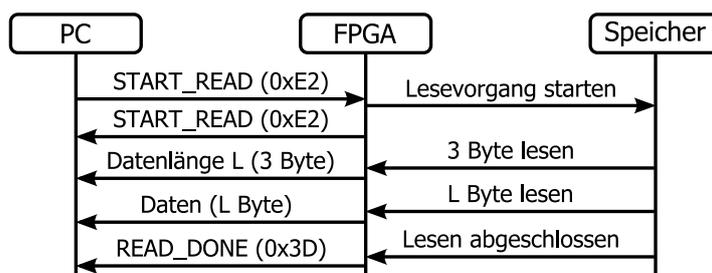
sen. Deshalb wurde ein solches Programmiergerät entwickelt. Die Kommunikation zwischen dem PC und dem Flash-Speicher übernimmt ein FPGA. Das FPGA wird über die USB-Anbindung aus Abschnitt 4.2 mit dem PC verbunden. Der Transponder enthält bereits einen Programmieranschluss. Dieser kann direkt an das FPGA-Entwicklungsboard angeschlossen werden.

4.5.1 Ablauf des Programmiervorgangs

Die Interaktion zwischen PC und FPGA basiert auf einem einfachen Protokoll (Abb. 4.20): Der PC beginnt den Schreib- bzw. Lesevorgang, indem



(a) Schreibvorgang



(b) Lesevorgang

Abbildung 4.20: Ablauf der Kommunikation zwischen PC und FPGA

ein entsprechendes Startzeichen (START_WRITE bzw. START_READ) gesendet wird. Das FPGA startet den Vorgang und antwortet mit demselben Startzeichen.

Bei einem Schreibvorgang sendet der PC zunächst drei Byte, welche die gesamte Datenlänge angeben. Das FPGA bestätigt den Erhalt mit einem

Bestätigungszeichen (ACK). Anschließend überträgt der PC die zu schreibenden Daten. Dabei wird jeweils ein Datenblock fester Größe gesendet und danach auf ein ACK-Zeichen gewartet. Auch der letzte Datenblock wird mit einem ACK-Zeichen bestätigt. Sobald der letzte Datenblock in den Speicher geschrieben wurde, schließt das FPGA den Schreibvorgang mit einem Steuerzeichen (WRITE_DONE) ab.

Bei einem Lesevorgang sendet das FPGA nach dem Startzeichen drei Byte, welche die gesamte Datenlänge angeben. Anschließend überträgt das FPGA alle gespeicherten Daten zum PC. Nachdem der Flash-Speicher in einem Zug auslesbar ist und die PC-Anbindung ohnehin eine automatische Flusskontrolle enthält, ist keine blockweise Bestätigung durch den PC vorgesehen. Sobald das letzte Datenbyte gesendet wurde, schließt das FPGA den Lesevorgang mit einem Steuerzeichen (WRITE_DONE) ab.

4.5.2 Implementierung für das FPGA

Am FPGA ist das Programmiergerät in zwei Blöcke gegliedert. Ein Funktionsblock steuert die Kommunikation zwischen dem PC und dem FPGA und der zweite Funktionsblock kommuniziert mit dem Flash-Speicher. Der Lesevorgang ist vergleichbar mit dem Lesevorgang aus der Transponderimplementierung. Bei einem Schreibvorgang empfängt das FPGA-Entwicklungsboard die Daten vom PC, puffert diese und schreibt sie zeilenweise in den Speicher. Durch das Zwischenspeichern in einem Pufferspeicher, wird sichergestellt, dass immer ausreichend Daten vorhanden sind, um den Speicher mit maximaler Taktfrequenz anzusteuern.

Beim Transponderspeicher handelt es sich um einen Flash-Speicher. Tabelle 4.6 enthält eine Übersicht über die SPI-Befehle. Für diese Art von

Tabelle 4.6: SPI-Befehle zum Schreiben bzw. Löschen eines Flash-Speichers

Instruktion	Parameter	Antwort
WREN (0x06)		
RDSR (0x05)		Daten (1 bis ∞ Byte)
WRSR (0x01)	Daten (1 Byte)	
BE (0xC7)		
PP (0x02)	Adresse (3 Byte) + Daten (1 bis 256 Byte)	

Speicher muss eine bestimmte Schreibsequenz eingehalten werden: In einem ersten Schritt wird der Schreibzugriff mit dem Befehl WREN (*Write enable*) freigegeben. Dann kann der Speicher mit dem Befehl BE (*Bulk erase*) vollständig gelöscht werden. Alle Datenbits befinden sich danach im Zustand einer logischen 1. Das ist notwendig, weil der Programmierbefehl die Datenbits nur von einer logischen 1 auf eine logische 0 (aber nicht umgekehrt)

setzen kann. Anschließend werden jeweils Speicherseiten zu 256 Byte mit dem Befehl PP (*Page program*) in den Speicher programmiert. Nach jedem PP- und BE-Befehl muss der Status mit dem Befehl RDSR (*Read status register*) gelesen und ausgewertet werden. Dieser Befehl gibt das Statusbyte zurück, wobei dieses solange wiederholt wird, bis der Befehl vom FPGA (Busmaster) abgebrochen wird. Es darf solange kein neuer Schreib- oder Lesebefehl an den Speicher gesendet werden, bis das im Statusbyte enthaltene WIP-Bit (*Write in progress bit*, engl. der Schreibvorgang ist noch nicht abgeschlossen) deaktiviert wurde.

4.5.3 Implementierung für den PC

Die PC-Seite des Programmiergeräts besteht aus einer C++-Anwendung. Diese Applikation ermittelt die Größe der zu speichernden Datei und lädt diese in den Arbeitsspeicher. Anschließend wird eine Schreibsequenz gestartet und die Dateigröße (als 3-Byte-Wert) gefolgt von den Datenblöcken an das FPGA gesendet. Danach wird ein Lesezyklus durchgeführt. Die Anwendung vergleicht die gelesene Dateigröße und die gelesenen Daten mit der gesendeten Datei. Der Schreibvorgang war erfolgreich, wenn diese Informationen übereinstimmen.

Kapitel 5

Protokoll zur Nutzdatenübertragung im Downlink

Im vorhergehenden Kapitel wurde die Entwicklung einer neuen Demonstrator-Hardware beschrieben. In einem nächsten Schritt soll die Demonstrationsumgebung eine Verbindung zwischen Transponder und Lesegerät aufbauen und anschließend Daten übertragen. Dazu müssen Regeln zur Kommunikation zwischen dem Transponder und dem Lesegerät aufgestellt werden. Diese Regeln für den Kommunikationsablauf und den Aufbau der Datenstrukturen werden in einem Übertragungsprotokoll bzw. einer Hierarchie von Übertragungsprotokollen vereinbart. Eine Protokollhierarchie wird auch als *Protocolstack* (engl. Protokollstapel) bezeichnet.

In diesem Kapitel werden die Anforderungen an ein solches Protokoll definiert. Anhand dieser Kriterien erfolgt die Entwicklung eines geeigneten Protokollstapels. Abschließend werden diese Konzepte in Hard- und Software umgesetzt und in den Demonstrator integriert.

5.1 Auswahlkriterien

Es existiert bereits eine Vielfalt an Kommunikationssystemen. Nicht jedes Protokoll ist für den Einsatz in jedem System geeignet. Daher verwenden die verschiedenen Kommunikationssysteme unterschiedliche Schnittstellen und Protokolle. Um für die Anwendung in einem bestimmten System und mit einer bestimmten Kommunikationsschnittstelle in Frage zu kommen, muss ein Protokoll verschiedene Bedingungen erfüllen. Diese Anforderungen werden durch die Eigenschaften des Systems und der Kommunikationsschnittstelle und durch die Aufgaben des Protokolls festgelegt.

5.1.1 Limitierende Faktoren

Die Demonstratorplattform weist einige limitierende Faktoren auf. Diese schränken die Möglichkeiten bei der Auswahl eines Protokolls drastisch ein.

Übertragungskanal

Ein schwerwiegender Faktor ist der verfügbare Übertragungskanal. Der Demonstrator verfügt nur über einen Downlinkkanal. Ein Uplinkkanal mit höheren Datenraten ist Thema aktueller Forschungsarbeit. Zum Zeitpunkt der Entwicklung des Demonstrators befanden sich diese Konzepte jedoch noch nicht in einem einsatzfähigen Zustand. Als Alternative zu einem hochratigen Uplinkkanal wurde der Uplinkkanal derzeit standardisierter RFID-Systeme nach ISO/IEC 14443 untersucht. Die Integration eines solchen Übertragungskanals in den Prototyp würde jedoch den Rahmen dieser Diplomarbeit sprengen. Daher ist die Kommunikation nur im Simplexbetrieb (Übertragung vom Transponder zum Lesegerät ohne Rückkanal) möglich [10].

Übertragungsfehler

Laut [12] liegt die Fehlerhäufigkeit bei der Übertragung vom Transponder zum Lesegerät etwa zwischen 10^{-4} und 10^{-3} Fehlern pro übertragenem Bit. Über die VHD-Schnittstelle ist also keine fehlerfreie Übertragung möglich. Daher muss das Protokoll in der Lage sein, Übertragungsfehler zu erkennen und zu korrigieren.

Datenübertragungsrate

Verschiedene Übertragungsprotokolle ermöglichen eine dynamische Beeinflussung der Datenrate in Abhängigkeit der Fehlerhäufigkeit. Ohne Rückkanal ist es jedoch unmöglich, den Transponder über die Übertragungsqualität in Kenntnis zu setzen. Eine dynamische Änderung der Datenübertragungsrate ist daher nicht möglich. Die Übertragung erfolgt deshalb immer mit 6,78 MBit/s.

Synchronisation

Der Transponder leitet seinen Systemtakt, und in weiterer Folge den Symboltakt der Sendesymbole, vom Trägersignal ab. Das Trägersignal wird wiederum vom Lesegerät erzeugt. Aus diesem Grund sind Transponder und Lesegerät bereits Symboltaktsynchron.

Anders ist die Lage in Bezug auf ganze Bits und die Bildung von Übertragungsrahmen (Frames). Ohne Rückkanal kann das Lesegerät den Beginn eines Bits (ein Bit besteht aus zwei Symbolen) nicht beliebig festle-

gen. Stattdessen muss das Übertragungsprotokoll geeignete Elemente enthalten, die eine eindeutige Synchronisation auf Bitebene, und damit eine korrekte Manchesterdecodierung, ermöglichen. Ebenso muss gewährleistet sein, dass Übertragungsrahmen definierte, unverwechselbare Anfangs- und Endsequenzen haben.

Implementierungsaufwand

Am Transponder stellt auch der Implementierungsaufwand ein klares Limit dar. Als größtes CPLD ist am Transponder ein XC2C256 vorhanden. Dieses enthält 16 Funktionsblöcke mit jeweils einer Produktterm-Matrix (56 Produktterme) und 16 Makrozellen [30]. Eine Makrozelle enthält wiederum jeweils ein Flipflop. Das CPLD schränkt daher die implementierbare Logik und den verfügbaren Zustandsspeicher (maximal 256 Registerbits) deutlich ein.

5.1.2 Anforderungen

Das Übertragungsprotokoll muss dafür sorgen, dass die gesendeten Daten fehlerfrei beim Empfänger ankommen. Weiters müssen die Anforderungen und die limitierenden Faktoren des VHD-Systems berücksichtigt werden.

Transpondererkennung

Das Übertragungsprotokoll muss den Transponder erkennen, sobald sich dieser in der Lesereichweite der Empfangsantenne befindet. Die Erkennung, Antikollision und Aktivierung mehrerer Transponder ist nicht vorgesehen. Nachdem nur ein unidirektionaler Übertragungskanal, vom Transponder zum Lesegerät, zur Verfügung steht, und der Transponder über keinerlei Empfangsmöglichkeit verfügt, darf sich nur ein Transponder im HF-Feld befinden. Der Transponder wird durch das Vorhandensein des Trägersignals aktiviert. Durch die fehlende Empfangseinrichtung ist es für den Transponder unmöglich zu erkennen ob weitere Transponder aktiv sind. Nachdem eine Manchesterdecodierung ohne Hilfsträger verwendet wird und keine Multiplexverfahren eingesetzt werden, kann auch das Lesegerät die Empfangssignale mehrerer Transponder nicht voneinander trennen.

Die Erkennung ob sich ein Transponder in der Reichweite des Lesegerätes befindet lässt sich über die Erkennung von Synchronisationssequenzen realisieren. Wird eine bestimmte Anzahl dieser Sequenzen empfangen, dann ist ein Transponder vorhanden. Wird anschließend über einen Zeitraum, welcher größer als das Wiederholungsintervall der Synchronisation ist, keine Synchronisationssequenz mehr empfangen, dann kann angenommen werden, dass sich der Transponder außerhalb der Reichweite des Lesegerätes befindet. Die Pause zwischen den Synchronisationsphasen bestimmt daher die

Geschwindigkeit mit der erkannt wird, ob ein Transponder innerhalb oder außerhalb der Reichweite ist.

Synchronisation

Wie die Transpondererkennung kann auch die Synchronisation der Bitübertragung (d. h. der Manchesterdecodierung) mit der Hilfe von Synchronisationssequenzen realisiert werden. Dazu müssen beide Varianten der Manchesterdecodierung betrachtet werden. Wird in einem Decodierungspfad eine bestimmte Anzahl dieser Sequenzen erkannt, dann kann dieser Pfad als gültig und der alternative Pfad als ungültig betrachtet werden.

Framing

Durch periodisch wiederholte Synchronisationssequenzen kommt es zur Rahmenbildung (Framing) [10]. D. h. ein Datenframe ist von Synchronisationszeichen umschlossen. Die Rahmenbildung muss eindeutig sein und innerhalb des Datenblocks auftretende Bitfolgen dürfen nicht fälschlicherweise als Synchronisationszeichen erkannt werden. Dazu muss sichergestellt werden, dass bestimmte Bitmuster niemals im Datenteil enthalten sein können [10]. In Abhängigkeit davon, ob das Übertragungsprotokoll bit- oder zeichenorientiert ist, lässt sich das über Stopfbits bzw. Stopfzeichen erzielen [10].

Fehlererkennung

Eine wichtige Anforderung an die Kommunikation ist eine fehlerfreie Übertragung. Daher muss das Übertragungsprotokoll in der Lage sein, zu erkennen ob die empfangenen Daten gültig sind. Ungültige Daten müssen verworfen bzw. korrigiert werden. Die Erkennung von Bitübertragungsfehlern lässt sich mit redundanten Informationen bewerkstelligen [10]. Dazu müssen die Daten in Pakete zerlegt werden. Jedes Paket besteht aus mehreren Datenbits. Für jedes Datenpaket wird vom Sender eine Prüfsumme berechnet und an das Datenpaket angehängt. Die Prüfsumme kann dabei ein oder mehrere Bits umfassen. Der Empfänger bildet die Prüfsumme anschließend erneut und vergleicht sie mit der empfangenen Prüfsumme.

Durch das Prüfsummenverfahren werden allerdings nicht einzelne fehlerhafte Bits erkannt. Ein erkannter Bitfehler bedeutet immer, dass ein ganzes Datenpaket als ungültig identifiziert und deshalb verworfen wird. Umso größer ein Datenpaket ist, desto größer ist die Wahrscheinlichkeit, dass ein Bitfehler in diesem Paket auftritt. Je höher die Bitfehlerrate ist, desto kleiner müssen daher die Datenpakete sein. Gleichzeitig enthalten Datenpakete neben den Nutzdaten auch noch Steuerinformationen wie z. B. die Prüfsumme. Nachdem die Steuerinformationen typischerweise eine feste, von der Paketgröße unabhängige Länge haben, sinkt mit abnehmender Paketgröße auch die Nutzdatenrate. Diese wird durch das Verhältnis zwischen

Nutzdaten und Steuerinformationen bestimmt. Es muss daher ein Optimum zwischen der Bitfehlerrate (und damit der Paketgröße) und der Nutzdatenrate ermittelt werden.

Fehlerkorrektur

Um die gesendeten Daten fehlerfrei und vor allem vollständig zu empfangen, reicht es nicht aus, fehlererkennende Maßnahmen zu ergreifen. Durch die Fehlererkennung werden zwar fehlerhafte Datenpakete verworfen. Die darin enthaltenen Daten gehen dadurch aber dauerhaft verloren. Die Fehlererkennung garantiert den fehlerfreien Empfang der Daten. Sie stellt allerdings nicht sicher, dass die gesendeten Daten vollständig beim Empfänger ankommen.

Maßnahmen zur Sicherung des vollständigen Datenempfangs bezeichnet man als Fehlerkorrektur. Es gibt verschiedene Verfahren zur Korrektur von Übertragungsfehlern. Zu diesen Verfahren zählen fehlerkorrigierende Codierungen und die wiederholte Übertragung fehlerhafter Datenblöcke.

Fehlerkorrigierende Codierungen basieren auf demselben Prinzip wie die Fehlererkennung: Durch redundante Informationen lassen sich Bitfehler identifizieren. Wenn man die Redundanz geeignet wählt, dann lassen sich auch Rückschlüsse auf die tatsächlich gesendeten Daten ziehen. Ein Beispiel für eine Codierung zur Erkennung und Korrektur bestimmter Klassen von Bitfehlern ist der Hamming-Code [10].

Um erkannte, aber nicht durch die Codierung korrigierbare Fehler zu beheben, hilft nur eine erneute Übertragung der betroffenen Datenpakete. Das dabei am häufigsten eingesetzte Verfahren ist ARQ (*Automatic repeat request*, engl. automatische Aufforderung zur Wiederholung) [10]. Bei diesem Verfahren wird jeder empfangene Datenblock überprüft und mit einer Antwort an den Absender quittiert (ACK, *Acknowledge*) oder abgelehnt (NAK, *Not acknowledge*). Wurde ein Paket abgelehnt, so wird es umgehend erneut gesendet, ansonsten setzt der Sender die Übertragung mit dem nächsten Datenpaket fort. Es existieren auch Abwandlungen dieses Verfahrens bei denen nur positive oder nur negative Bestätigungen gesendet werden. Zudem werden Quittierungen zum Teil für mehrere Datenblöcke gemeinsam gesendet.

Nachdem die VHD-Schnittstelle des Demonstrators allerdings nur im Simplexbetrieb arbeitet, steht kein Rückkanal für Statusinformationen zur Verfügung. Das ARQ-Verfahren ist daher für diese Anwendung ungeeignet. Es muss also ein Korrekturverfahren gefunden werden, das auch für den Simplexbetrieb einsetzbar ist.

Es ist daher naheliegend, andere Kommunikationsnetze mit ähnlichen Einschränkungen zu betrachten. Eine Gruppe solcher Systeme sind Rundfunknetze. Bei diesen gibt es in der Regel, so wie beim VHD-System, keinen Rückkanal zum Rundfunksender. Die Datenübertragung in digitalen Rund-

funknetzen erfolgt typischerweise nach dem MPEG-Standard (*Moving Picture Experts Group*). Neben Audio- und Videodaten unterstützt der Standard auch Kanäle zur Übertragung von Steuerinformationen [3]. Diese Übertragung ist im Standard ISO/IEC 13818-6 (DSM-CC, *Digital Storage Media Command and Control*) spezifiziert. Speziell das in diesem Standard beschriebene Karussellverfahren scheint ein geeignetes Übertragungsverfahren zur Fehlerkorrektur zu sein.

Das Datenkarussell (*Data carousel*) ist in [3] beschrieben: Die Datenele-

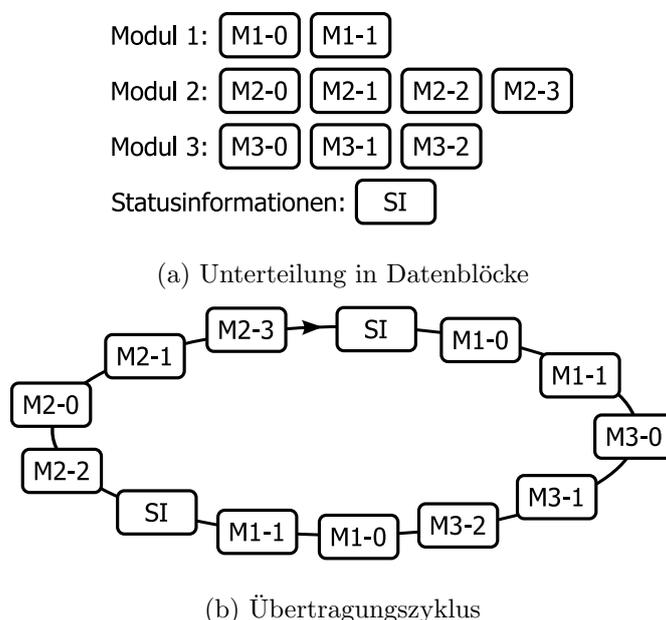


Abbildung 5.1: Datenkarussellverfahren [3]

mente (für DSM-CC sind das Steuernachrichten und Datenmodule) werden in kleinere Blöcke fester Länge zerteilt (Abb. 5.1(a)). Anschließend werden diese Datenblöcke zyklisch gesendet (Abb. 5.1(b)). DSM-CC sieht dabei vor, dass ein Datenblock je nach Priorität auch mehrfach in einem Zyklus vorkommen kann. Die Datenpakete sind durch eine laufende Nummer eindeutig identifizierbar. Der Empfänger kann daher, unabhängig davon bei welchem Paket die Übertragung startet und in welcher Reihenfolge die Pakete ankommen, jedes empfangene, gültige Datenpaket umgehend verwenden um das Datenmodul zusammzusetzen.

5.2 Protokollentwurf

Die erörterten Anforderungen müssen nun in einem Übertragungsprotokoll kombiniert werden. Das Übertragungsprotokoll besteht dabei aus Regeln für

den Kommunikationsablauf und den dazu verwendeten Datenstrukturen. Es setzt auf der durch die VHD-Schnittstelle vorgegebenen Bitübertragungsschicht auf und ist selbst in eine Zugriffskontrollschicht und eine Datensicherungsschicht unterteilt. Während die Bitübertragungsschicht der ersten Schicht des OSI-Referenzmodells (*Open systems interconnection reference model*) entspricht, sind die Zugriffskontroll- und die Datensicherungsschicht der zweiten Schicht zuzuordnen.

5.2.1 Zugriffskontrollschicht

Diese Schicht ist für die Übertragung von Datenrahmen zuständig. Die Datenrahmen sorgen für die Synchronisation auf Bitebene. Dazu wird der Datenteil von Synchronisationszeichen (SYN) umschlossen. Die Synchronisationssequenzen haben die Aufgabe, eine eindeutige Entscheidung für einen der beiden Manchesterpfade zu ermöglichen. Um Anhand des Synchronisationszeichens den gültigen Decodierungspfad auszuwählen, muss das Bitmuster dieser Sequenz bestimmte Anforderungen erfüllen.

Die SYN-Sequenz darf nicht innerhalb der Nutzdaten vorkommen. Das lässt sich durch Stopfbits sicherstellen. Die Synchronisationssequenz muss dazu eine lange Folge gleicher Bits, d. h. eine lange Sequenz aus lauter Einsen oder Nullen, enthalten. Im Datenteil wird dann das Auftreten solcher Bitfolgen durch Einfügen einer Eins bzw. einer Null verhindert. Enthält also das SYN-Zeichen eine Folge aus sechs Einsen, dann muss im Datenstrom spätestens nach jeder fünften Eins eine Null eingefügt (gestopft) werden. Nachdem dieser Synchronisationsmechanismus den richtigen Manchesterpfad bestimmt, sind zusätzliche Vorkehrungen notwendig. Wenn im Datenstrom sechs aufeinanderfolgende Nullen auftreten würden, dann werden diese im ungültigen Decodierungspfad als sechs Einsen erkannt. Das signalisiert wiederum das Vorhandensein einer Synchronisationssequenz. Daher muss im Datenstrom auch spätestens nach jeder fünften Null eine Eins eingefügt werden, um das fälschliche Auftreten einer SYN-Sequenz in beiden Manchesterpfaden zu verhindern.

Um eine zuverlässige Synchronisation zu garantieren, muss jedes einzelne SYN-Zeichen in einer Sequenz aus SYN-Zeichen eindeutig abgrenzbar sein. Teile von aneinandergrenzenden Synchronisationszeichen dürfen also nicht erneut ein SYN-Zeichen ergeben. Für eine Bitfolge b aus zwei 8-Bit-SYN-Zeichen muss daher

$$\forall i \in \{1..7\} : \exists n \in \{0..7\} : b[n] \neq b[n + i] \quad (5.1)$$

gelten. Dieselbe Bedingung muss auch für ein SYN-Zeichen mit angrenzenden Datenbits, d. h. für eine Bitfolge b aus einem 8-Bit-SYN-Zeichen und acht Datenbits, gelten.

Wegen den zwei Pfaden der Manchesterdecodierung darf auch eine Folge von falsch decodierten SYN-Zeichen keine Sequenz enthalten die dem SYN-

Zeichen entspricht. Ansonsten wäre es nicht möglich den korrekten Decodierungspfad zu ermitteln. Abbildung 5.2 zeigt die Decodierungsvarianten

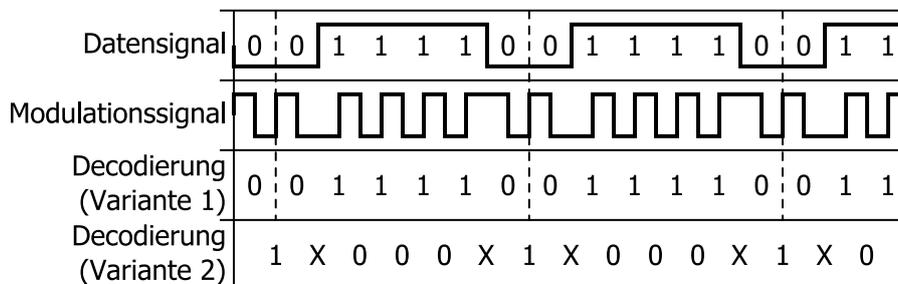


Abbildung 5.2: Decodierungsvarianten für die wiederholte Bitfolge 011110

für die Bitsequenz 011110. Die Fehldecodierung (Variante 2) ergibt also die Sequenz X000X1. Wobei X sowohl eine 1 als auch eine 0 sein kann. Für eine Bitfolge b aus zwei 8-Bit-SYN-Zeichen und die entsprechende Fehldecodierung \bar{b} muss daher für jede Verschiebung i von 0 bis 7

$$\forall i \in \{0..7\} : \exists n \in \{0..7\} : b[n] \neq \bar{b}[n + i] \quad (5.2)$$

gelten. Die Bitsequenz 011110 erfüllt somit diese Bedingung.

Als Synchronisationssequenz wurde das Bitmuster 01111110 (0x7E) ausgewählt. Dieses 8-Bit-Zeichen erfüllt alle geforderten Bedingungen.

Der Frameaufbau ist in Abbildung 5.3 dargestellt. Ein Datenrahmen be-

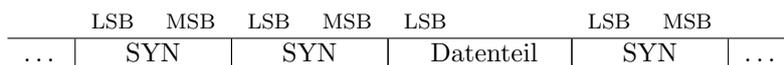


Abbildung 5.3: Frameaufbau

ginnt mit mindestens 12 SYN-Zeichen, gefolgt von maximal N Datenbits, und endet mit einem SYN-Zeichen. Die Daten beginnen, wie bei Protokollen nach dem Standard ISO/IEC 14443, mit dem LSB (*Least significant bit*). In die Datenbits werden Stopfbits eingefügt. Dazu wird an fünf aufeinanderfolgende Einsen eine Null und an fünf aufeinanderfolgende Nullen eine Eins angehängt. Dabei ist zu beachten, dass auch das eingefügte Bit zur darauffolgenden Bitfolge gezählt wird. Das bedeutet, dass nach einer eingefügten Eins nur mehr vier Einsen in Form von Datenbits folgen müssen, bis eine Null als Stopfbit notwendig ist.

Neben der Synchronisation auf Bit- und Frameebene hat die SYN-Sequenz auch noch die Aufgabe die Transpondererkennung zu ermöglichen. Dazu muss das Lesegerät messen, ob innerhalb eines bestimmten Zeitraumes SYN-Zeichen empfangen wurden. Der maximale Zeitabstand zwischen zwei

Synchronisationssequenzen entspricht der maximalen Framegröße. Die maximale Framegröße beeinflusst daher die Geschwindigkeit, mit der erkannt wird, ob sich ein Transponder innerhalb der Reichweite des Lesegeräts befindet oder diese wieder verlassen hat. Mit der Datenrate R , der maximalen Anzahl N an Datenbits und der maximalen Anzahl S an Stopfbits berechnet sich die maximale Zeit t_{max} , die benötigt wird um zu erkennen, dass der zuvor detektierte Transponder die Reichweite des Lesegeräts verlassen hat, zu

$$t_{max} = \frac{N + S}{R}. \quad (5.3)$$

Die maximale Anzahl S an Stopfbits wird durch eine Bitfolge aus abwechselnd vier Nullen und vier Einsen erreicht. Bei dieser Sequenz wird nach jedem dieser Viererblöcke ein Stopfbit eingefügt. Somit ist jedes fünfte Bit des Datenteils ein Stopfbit. Für die maximale Zeit t_{max} ergibt sich daher

$$t_{max} = \frac{5}{4} \cdot N \cdot \frac{1}{R}. \quad (5.4)$$

Tabelle 5.1 gibt einen Überblick über den Zusammenhang zwischen der maximalen Rahmengröße N und der Erkennungszeit t_{max} für eine Datenrate von 6,78 MBit/s.

Tabelle 5.1: Zusammenhang zwischen der maximalen Rahmengröße N und der Erkennungszeit t_{max}

N	t_{max}
256 Bit	47,2 μ s
512 Bit	94,4 μ s
1024 Bit	188,8 μ s
2048 Bit	377,6 μ s
3072 Bit	566,4 μ s
4096 Bit	755,2 μ s
5120 Bit	944,0 μ s
6144 Bit	1132,8 μ s
8192 Bit	1510,4 μ s

Das Lesegerät hat keine Möglichkeit die Datenübertragung und damit den Transponder zu beeinflussen. Solange der Transponder keine Datenbits überträgt, müssen ständig Synchronisationszeichen gesendet werden. So wird die Transpondererkennung aufrechterhalten.

In Abhängigkeit der Rahmengröße lässt sich die maximal mögliche effektive Datenübertragungsrate ermitteln. Bei 12 SYN-Zeichen und N Datenbits pro Frame ergibt sich unter der Annahme, dass keine Stopfbits notwendig

sind, eine maximale effektive Datenrate

$$R_{eff,max} = \frac{\frac{5}{4} \cdot N}{\frac{5}{4} \cdot N + 12 \cdot 8 \text{ Bit}} \cdot R = \frac{5 \cdot N}{5 \cdot N + 384 \text{ Bit}} \cdot R. \quad (5.5)$$

In Abbildung 5.4 wird dieser Zusammenhang grafisch dargestellt.

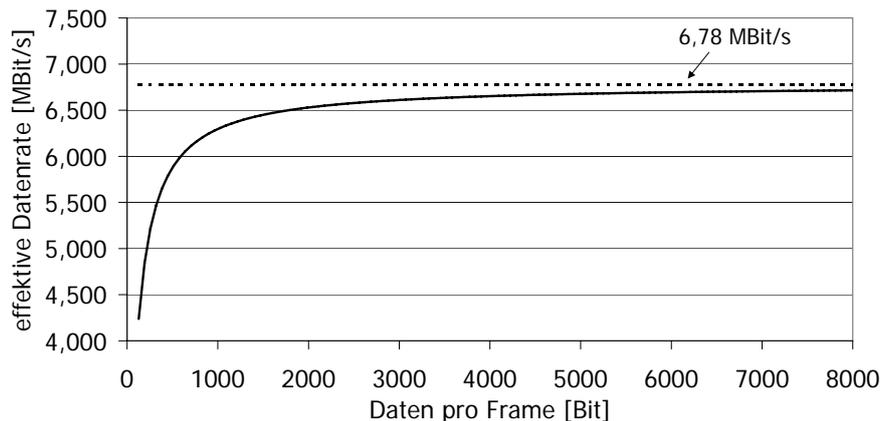


Abbildung 5.4: Maximale effektive Datenrate $R_{eff,max}$ der Zugriffskontrollschicht in Abhängigkeit der Rahmengröße N bei fehlerfreier Übertragung

5.2.2 Datensicherungsschicht

Diese Schicht ist für die zuverlässige Übertragung von Datenpaketen zuständig. Die Datenpakete ermöglichen die Erkennung und Korrektur von Übertragungsfehlern.

Zur Fehlererkennung enthält jedes Paket eine Prüfsumme. Analog zur Prüfsumme für die Datenübertragung nach ISO/IEC 14443 wird ein zyklisches Redundanzprüfverfahren (CRC, *Cyclic redundancy check*) verwendet. Bei diesem Prüfsummenverfahren betrachtet man den Bitstrom als Polynom. Der Divisionsrest einer Polynomdivision durch ein Generatorpolynom entspricht der CRC-Prüfsumme [10]. Die Prüfsumme wird an die Daten angehängt. Der Empfänger führt eine Polynomdivision über die empfangenen Daten (inklusive der CRC-Prüfsumme) durch [10]. Wenn der Divisionsrest nicht null ergibt, dann wurde ein Übertragungsfehler erkannt [10]. Umgekehrt gilt diese Aussage jedoch nicht: Ergibt der Divisionsrest null, so ist zwar das zwar die Restwahrscheinlich für einen Übertragungsfehler gering, eine hundertprozentige Sicherheit ist aber nicht gegeben.

Wie bei Systemen nach ISO/IEC 14443 Typ A wird ein Generatorpolynom des Grads 16, zusammen mit dem Initialwert `0x6363`, verwendet. Das Polynom $x^{16} + x^{12} + x^5 + 1$ ist in der Literatur [10] unter dem Namen CRC-CCITT bekannt. Nach [10] erkennt man damit zumindest jeden

1-Bit-Fehler, alle Fehler mit einer ungeraden Bitanzahl und alle Burstfehler (das sind unterbrechungsfreie Bitfehlerfolgen) mit einer Länge von maximal 16 Bit. Zudem werden alle 2-Bit-Fehler erkannt, wenn die Länge der Eingabedaten kürzer als 4096 Byte ist. Des Weiteren ist für jeden 17 Bit langen Burstfehler nur ein Fehler nicht detektierbar.

Zur Fehlerkorrektur wird eine vereinfachte Variante des Datenkarussellverfahrens eingesetzt. Bei dem verwendeten Datenkarussell gibt es nur ein Datenmodul. Dieses Modul entspricht dem gesamten Transponderspeicher. Die Nutzdaten werden in Pakete unterteilt. Abbildung 5.5 zeigt den Pa-

Kennung (1 Byte)	ID (2 Byte)	Datenlänge M (3 Byte)	Paketlänge L (1 Byte)	Daten (L Byte)	Prüfsumme (2 Byte)
---------------------	----------------	----------------------------	----------------------------	----------------------	-----------------------

Abbildung 5.5: Paketaufbau

ketaufbau. Jedes Paket besteht aus einem Paketkopf, einem Datenteil und einer Prüfsumme. Der Paketkopf beginnt mit einer Kennung. Die Kennung 0xD8 gibt an, dass es sich um ein Datenpaket handelt. Sie ist in der aktuellen Implementierung die einzige gültige Kennung. Dadurch lässt sich der empfangene Bitstrom einfach nach beginnenden Datenpaketen durchsuchen. Die Paketidentifikation (ID) ist eine laufende Nummer, mit der jedes Paket eindeutig identifizierbar ist. Damit kann der Empfänger die Datenpakete sortieren und zu einer Datei kombinieren. Dabei wird der Empfänger durch die Datenlänge M unterstützt. Diese spezifiziert die Gesamtlänge der Datei. Anhand dieses Wertes kann der Empfänger die Vollständigkeit der empfangenen Daten feststellen. Die Paketlänge L gibt die Anzahl der Nutzdatenbytes an. Die zwei Prüfsummenbytes enthalten die CRC-CCITT-Prüfsumme über den Paketkopf und die Nutzdaten.

Der Transponder sendet die Datenpakete zyklisch in Form eines Datenkarussells (Abb. 5.6). Der Empfänger prüft die einzelnen Datenpakete auf

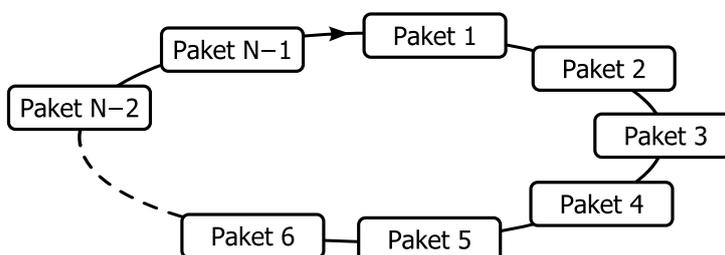


Abbildung 5.6: Datenkarussell

ihre Gültigkeit. Ungültige Pakete werden verworfen. Gültige Pakete, deren Paketidentifikationsnummer noch keinem zuvor empfangenen, gültigen Paket entspricht, werden in einer Liste gespeichert. Sobald die Summe der

gespeicherten Datenbytes der Gesamtdateigröße M entspricht, wurde die Datei vollständig empfangen.

In Abhängigkeit der Paketgröße, d. h. der Anzahl der enthaltenen Nutzdaten, und der Übertragungsrate der Zugriffskontrollschicht lässt sich die maximal mögliche effektive Datenübertragungsrate ermitteln. Bei L Datenbytes und 9 Bytes an Steuerinformationen ergibt sich eine maximale effektive Datenrate

$$R_{eff,max} = \frac{L}{L+9} \cdot \frac{5 \cdot N}{5 \cdot N + 384 \text{ Bit}} \cdot R, \quad (5.6)$$

wobei N der Paketlänge in Bit, $(L+9) \cdot 8 \text{ Bit}$, entspricht. Dadurch erhält man

$$R_{eff,max} = \frac{L}{L+9} \cdot \frac{5 \cdot (L+9) \cdot 8 \text{ Bit}}{5 \cdot (L+9) \cdot 8 \text{ Bit} + 384 \text{ Bit}} \cdot R \quad (5.7)$$

$$= L \cdot \frac{5}{5 \cdot (L+9) + 48} \cdot R \quad (5.8)$$

$$= \frac{5 \cdot L}{5 \cdot L + 93} \cdot R. \quad (5.9)$$

In Abbildung 5.7 wird dieser Zusammenhang grafisch dargestellt.

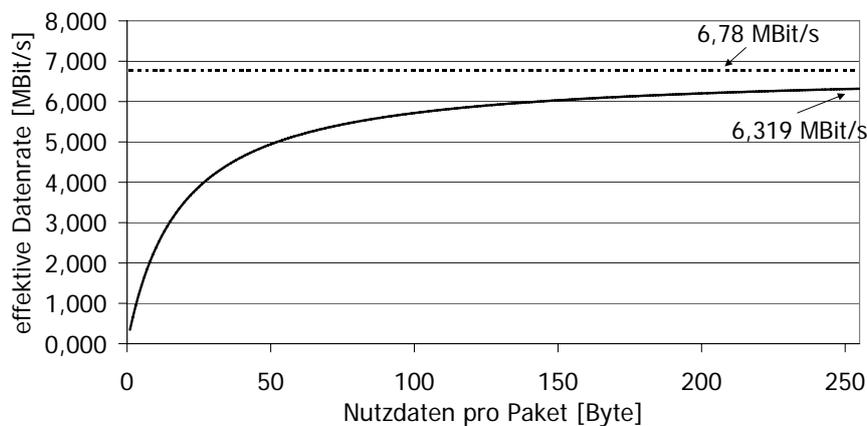


Abbildung 5.7: Maximale effektive Datenrate $R_{eff,max}$ der Datensicherungsschicht in Abhängigkeit der Datenbytes pro Paket (L) bei fehlerfreier Übertragung

Die tatsächliche Datenübertragungsrate liegt jedoch deutlich unter der maximal möglichen. Das liegt daran, dass bereits bei einem einzelnen Bitfehler ein ganzes Datenpaket verworfen wird. Durch nur ein einzelnes ungültiges Datenpaket ist ein weiterer Durchlauf des Datenkarussells notwendig, um das Datenpaket gültig zu empfangen. Wenn allerdings auch in diesem Durchlauf dasselbe Datenpaket einen Übertragungsfehler enthält ist erneut ein weiterer Durchlauf notwendig. Mit jedem Zyklus des Datenkarussells steigt

die Wahrscheinlichkeit des vollständigen Empfangens der Datei. Jedoch gibt es keine hundertprozentige Sicherheit dafür, dass alle Teile der Datei in endlicher Zeit gültig beim Empfänger ankommen.

Umso kleiner die Datenpakete sind, desto geringer ist die Wahrscheinlichkeit für einen darin enthaltenen Bitfehler. Daher steigt der Anteil an gültig empfangenen Paketen mit abnehmender Paketgröße. Während also die Datenübertragungsrate mit zunehmender Paketgröße zuerst ansteigt, sinkt diese ab einer bestimmten Paketgröße (bedingt durch die Fehlerrate) wieder ab.

5.3 Implementierung

Zur Implementierung des Protokolls in den Demonstrator wurde die Schichtentrennung beibehalten. Die Verarbeitung der Zugriffskontrolschicht findet auf der Demonstrator-Hardware (CPLD und FPGA) statt. Die Datensicherungsschicht wird mit PC-Programmen verarbeitet.

5.3.1 Sender

Der Transponder ist aus der Sicht des Übertragungsprotokolls der Sender. Der Sender hat die Aufgabe, die gespeicherten Daten in Pakete zu zerlegen, diese wiederum in Datenrahmen zu verpacken und über die VHD-Schnittstelle zu senden.

Datensicherungsschicht

Ursprünglich wurde der Ansatz verfolgt, die Datensicherungsschicht als Teil des CPLDs zu implementieren. Die insgesamt notwendige VHDL-Beschreibung des Transponders ist jedoch so komplex, dass mit dem CPLD XC2C256 nicht genug Ressourcen dafür zur Verfügung stehen. Erst mit einem doppelt so großen CPLD, dem XC2C512 von Xilinx, lässt sich die Datensicherungsschicht auf CPLD-Ebene realisieren. Tabelle 5.2 zeigt den Ressourcenbedarf

Tabelle 5.2: Ressourcenbedarf der VHDL-Beschreibung des Transponders inklusive der Datensicherungsschicht auf einem XC2C512

Makrozellen	Produktterme	Registerbits
350 von 512	1460 von 1792	184 von 512

der VHDL-Beschreibung auf einem XC2C512.

Daher übernimmt eine PC-Software die Aufgaben der Datensicherungsschicht. Das Programm zerlegt die Eingabedatei in Pakete fester Größe und Erzeugt den Paketkopf und die Prüfsumme. Die Datenpakete erhalten dabei eine fortlaufende Nummer die ihrer Anordnung in der Datei entspricht.

Zudem wird die Datenlänge M für jedes Paket auf die Größe der Eingabedatei gesetzt. Die Programmieranwendung schreibt diesen Datenstrom aus fertigen Datenpaketen in den Transponderspeicher. Die daraus ausgelesenen Daten entsprechen also bereits der Ausgabe der Datensicherungsschicht.

Zugriffskontrollschicht

Das CPLD übernimmt die Kommunikation auf der Ebene der Zugriffskontrollschicht. Die Verarbeitung wird mit einem Zustandsautomaten (Abb. 5.8) durchgeführt. Dieser erhält die aus dem Transponderspeicher gelesenen Da-

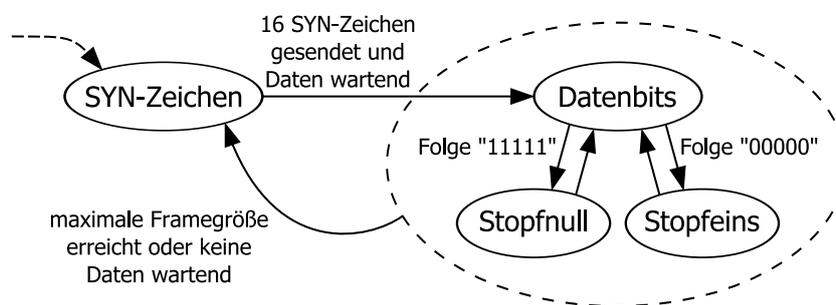


Abbildung 5.8: Zustandsdiagramm der Verarbeitung der Zugriffskontrollschicht im Sender

tenbytes. Es wird mit einer Sequenz aus mindestens 16 SYN-Zeichen begonnen. Sobald Daten bereitstehen, beginnt der Zustandsautomat die Daten bitweise zu senden. Um Wiederholungen von logischen Nullen und Einsen zu zählen, gibt es einen 0-Zähler und einen 1-Zähler. Jede logische Null erhöht den 0-Zähler und setzt den 1-Zähler zurück. Umgekehrt erhöht jede logische Eins den 1-Zähler und setzt den 0-Zähler zurück. Nach einer Folge von fünf logischen Einsen wird eine logische Null und nach fünf logischen Nullen eine logische Eins angehängt. Die Zählerstände berücksichtigen auch die Bits der Synchronisationssequenz. Während der Synchronisationssequenz fügt der Automat jedoch keine Stopfbits ein. Sobald die maximale Anzahl an Datenbits für einen Datenrahmen erreicht wurde oder keine Daten mehr bereitstehen, ist der Datenrahmen abgeschlossen und es wird wieder eine Sequenz aus mindestens 16 SYN-Zeichen gesendet.

5.3.2 Empfänger

Das Lesegerät ist aus der Sicht des Übertragungsprotokolls der Empfänger. Der Empfänger hat die Aufgabe, die gesendeten Daten über die VHD-Schnittstelle zu empfangen, den Transponder anhand der SYN-Sequenzen zu erkennen, die Stopfbits zu entfernen und die gesendete Datei aus den Datenpaketen zu rekonstruieren.

Zugriffskontrollschicht

Das FPGA ist für die Verarbeitung der Zugriffskontrollschicht zuständig. In einem ersten Schritt werden die empfangenen Bits durch ein 8-Bit-Schieberegister (Abb. 5.9) geführt. So sind SYN-Zeichen im empfangenen Daten-

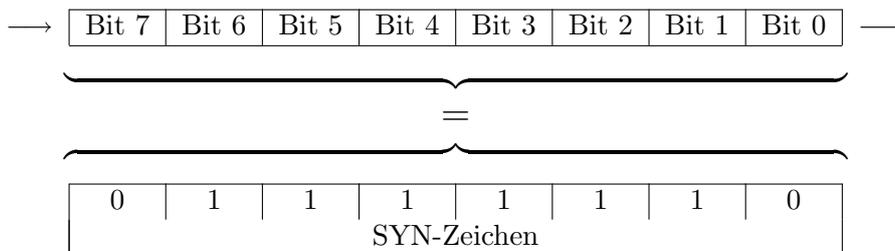


Abbildung 5.9: Schieberegister zum Vergleich der empfangenen Daten mit dem SYN-Zeichen

strom schnell erkennbar. Ein Zustandsautomat (Abb. 5.10) verarbeitet die Ausgabewerte des Schieberegisters. Solange sich kein Transponder in der

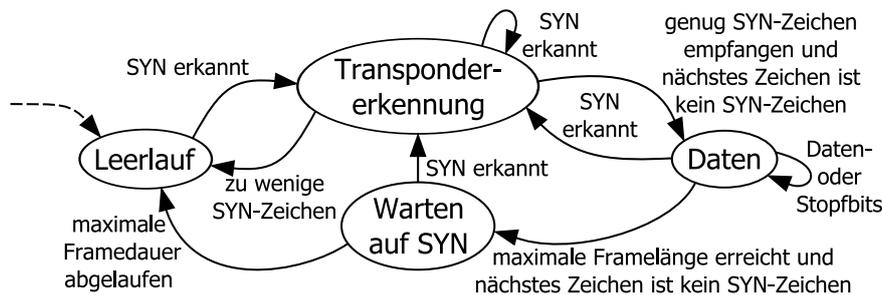


Abbildung 5.10: Prinzipieller Ablauf der Verarbeitung der Zugriffskontrollschicht im Empfänger

Empfangsreichweite des Lesegeräts befindet bleibt der Automat im Leerlauf. Wenn ein SYN-Zeichen detektiert wird, beginnt der Automat mit der Transpondererkennung. Ein Transponder gilt als erkannt, sobald zwölf aufeinanderfolgende SYN-Zeichen aufgetreten sind. Werden weniger als zwölf SYN-Zeichen empfangen bevor ein anderes Zeichen gefunden wurde, dann bricht der Zustandsautomat die Transpondererkennung ab und wechselt wieder in den Leerlauf. Sobald nach mindestens zwölf SYN-Zeichen ein anderes Datenbyte empfangen wurde wechselt der Automat in den Datenempfangszustand. In diesem Zustand liest der Automat die empfangenen Daten bitweise ein, zählt, wie bereits im Sender, die logischen Nullen und Einsen und entfernt die Stopfbits. Sobald wieder ein SYN-Zeichen erkannt wurde, führt der Automat erneut eine Transpondererkennung durch. Wenn die Anzahl an empfangenen Datenbits die maximale Framegröße überschreitet, bevor

ein SYN-Zeichen erkannt wurde, wird noch für eine maximale Framedauer auf eine Synchronisationssequenz gewartet, bevor signalisiert wird, dass der Transponder die Empfangsreichweite verlassen hat und der Zustandsautomat zurück in den Leerlauf wechselt.

Datensicherungsschicht

Die Datensicherungsschicht benötigt aufgrund des Datenkarussellverfahrens zumindest die Größe der gesendeten Datei an Speicherplatz. Diese Speichermenge steht auf dem FPGA-Entwicklungsboard nicht zur Verfügung. Darüber hinaus soll der Datenempfang kontinuierlich am PC dargestellt werden. Daher wurde die Datensicherungsschicht in Form einer PC-Anwendung realisiert.

Wie in Abschnitt 4.4.2 beschrieben, übergibt das FPGA die Daten und Steuerinformationen der Zugriffskontrollschicht byteweise an den PC. Am PC bewirkt ein neu erkannter Transponder, dass die gesamte Verarbeitung zurückgesetzt wird, d. h. alle bereits empfangenen Pakete wieder verworfen werden. Die empfangenen Daten sind in einer Warteschlange (FIFO-Speicher) abgelegt. Zur Verarbeitung dieser Warteschlange wurde ein Parser, d. h. ein Analysealgorithmus, entwickelt. Der Parser durchsucht die empfangenen Daten nach der ersten Paketbeginnkennung (0xD8). Alle vor dieser Kennung empfangenen Daten enthalten kein vollständiges Paket und können deshalb verworfen werden. Wenn die verbleibende Datenlänge ausreichend ist, um ein vollständiges Paket zu enthalten, verarbeitet und speichert der Parser den Paketkopf, die Daten und die Prüfsumme. Anschließend wird die Prüfsumme über das Paket gebildet. Stimmt diese mit der erwarteten Prüfsumme überein, so ist das Paket gültig und wird vollständig aus der Warteschlange gelöscht. Falls vorher noch kein gültiges Paket mit der Identifikationsnummer dieses Pakets empfangen wurde, fügt der Empfänger das Paket an der passenden Position in die Empfangsdatei ein. Die Empfangsdatei ist dabei ein Vektor. Dieser wird über die Paketidentifikationsnummer indiziert und enthält jeweils einen Zeiger auf den Speicherort des zugehörigen Datenpakets. Ist das Paket ungültig, wird es verworfen. Zusätzlich löscht der Parser die Paketbeginnkennung aus der Warteschlange. Dadurch beginnt der nächste Parserdurchlauf beim nächsten Zeichen und ignoriert so eine eventuell falsch erkannte Paketbeginnkennung.

Sobald die Gesamtgröße der empfangenen Teildaten der in allen Paketen angegebenen Dateigröße entspricht, ist die Datei vollständig und kann weiterverarbeitet werden.

5.3.3 Integration in den Transponder und das Lesegerät

Die VHDL-Beschreibungen der Protokollverarbeitung müssen nun in das System des Transponders und in das System des Lesegeräts integriert werden.

CPLD

Das Transpondersystem ist in einem CPLD untergebracht. Nach der Einbindung des Protokolls muss nun analysiert werden, ob die Verarbeitungslogik mit den knappen Ressourcen des CPLDs auskommt.

Tabelle 5.3: Ressourcenbedarf des VHDL-Designs des Transponders auf einem XC2C256

Makrozellen	Produktterme	Registerbits
164 von 256	669 von 898	125 von 256

Tabelle 5.3 zeigt den Ressourcenbedarf auf einem XC2C256. Die Ressourcen des CPLDs reichen daher aus. Das Design kann dabei mit einer maximalen Taktfrequenz von 26,385 MHz betrieben werden. Der 13,56 MHz-Systemtakt erfüllt diese Bedingung.

FPGA

Die Auswertung des Empfangssignals inklusive der Verarbeitung der Datenrahmen ist in einem FPGA untergebracht. Nach der Einbindung des Protokolls muss auch hier analysiert werden, ob die Verarbeitungslogik mit den Ressourcen des FPGAs auskommt.

Tabelle 5.4: Ressourcenbedarf des VHDL-Designs des Lesegeräts auf einem XC2S150

Look-Up-Tabellen (mit 4 Eingängen)	711 von 3456
Registerbits	385 von 3456
RAM-Blöcke	1 von 12
Globale Taktpuffer	2 von 4
Takteingänge	2 von 4
DLLs	2 von 4

Tabelle 5.4 zeigt den Ressourcenbedarf auf einem XC2S150. Die Ressourcen des FPGAs reichen daher aus. Das Design kann dabei mit einer maximalen externen Taktfrequenz von 61,950 MHz betrieben werden. Der externe Takt mit einer Frequenz von 27,12 MHz erfüllt diese Bedingung.

5.4 Performance

Um die Performance des VHD-Systems und des VHD-Übertragungsprotokolls zu charakterisieren, wurden Messungen zur Bestimmung der Fehlerrate und der Datenrate durchgeführt. In einem ersten Schritt erfolgt die

Betrachtung der Zugriffskontrollschicht. Anschließend folgt die Datensicherungsschicht.

5.4.1 Zugriffskontrollschicht

Zur Bestimmung der Bitfehlerrate (BER, relative Häufigkeit der Bitfehler) wurde eine Anwendung entwickelt, mit der pseudozufällige Bitfolgen, in Datenrahmen der Zugriffskontrollschicht verpackt, übertragen und am PC ausgewertet werden. Als pseudozufällige Bitfolge kommt eine M-Sequenz zum Einsatz. Eine M-Sequenz ist eine, durch ein lineares rückgekoppeltes Schieberegister erzeugte, periodische Bitfolge maximaler Länge [6]. Das Schieberegister hat eine Länge von 9 Bit. Jede Periode ist daher $2^8 - 1 = 511$ Bit lang. Aufgrund der verwendeten M-Sequenz lässt sich der Empfänger ebenfalls durch dieses Schieberegister realisieren. Zu Beginn füllt der Empfänger das Schieberegister mit den ersten neun empfangenen Bits. Anschließend wird das nächste Bit der M-Sequenz berechnet und mit dem empfangenen Bit verglichen. Stimmen diese überein, so ist kein Bitfehler aufgetreten. Sind die beiden Bits unterschiedlich so entspricht dies einem Bitfehler.

Durch das Übertragungsprotokoll der Zugriffskontrollschicht können neben einfachen Bitfehlern innerhalb der Datenbits auch fehlerhafte Stopfbits auftreten. Wird z. B. eine Folge aus fünf Einsen aufgrund eines Bitfehlers unterbrochen, dann behandelt der Empfänger das Stopfbit als normales Datenbit. Dadurch verschiebt sich die empfangene Sequenz im Vergleich zur vorausberechneten Sequenz. Um diese Verschiebung laufend zu korrigieren, erfolgt die weitere Berechnung der M-Sequenz im Empfänger nicht anhand des vorausberechneten Bits, sondern anhand des tatsächlich empfangenen Bits. Ein Nachteil dieser Berechnungsmethode liegt jedoch darin, dass einzelne Bitfehler unter Umständen eine ganze Serie von fälschlicherweise erkannten Bitfehlern auslösen. Nachdem eine detaillierte Messung der Bitfehlerhäufigkeit bereits in [12] durchgeführt wurde, dient der hier ermittelte Wert lediglich einem Vergleich mit den bekannten Ergebnissen. So lässt sich feststellen, ob der vollständig überarbeitete Prototyp gemeinsam mit dem neu entwickelten Übertragungsprotokoll eine ähnliche Performance wie sein Vorgänger erreicht. Die mit dem neuen Messaufbau bestimmte Bitfehlerhäufigkeit kann als Obergrenze für die tatsächlich erreichte Bitfehlerrate angesehen werden.

Zur Messung der Bitfehlerhäufigkeit wurde der Transponder nacheinander in verschiedenen Abständen zwischen 0 und 45 Millimetern über der Antenne des Lesegeräts fixiert. Die Mittelpunkte beider Antennen fallen auf eine Achse zusammen. Für jeden Abstand erfolgte eine Messwertaufzeichnung über mehrere Minuten.

Die Messergebnisse für einen Abstand von 0 Millimetern sind in Abbildung 5.11 dargestellt. Über eine Dauer von 250 Sekunden wurde eine mittlere Bitfehlerrate von $1,38 \cdot 10^{-4}$ Fehlern pro Bit bei einer mittleren

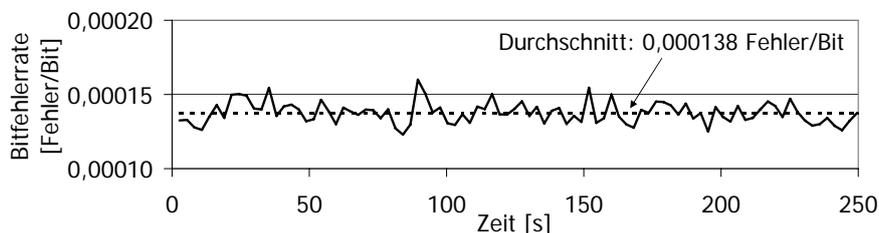


Abbildung 5.11: Bitfehler rate bei einer Framelänge von 2048 Bit in der Mitte der Leseantenne ohne Abstand

Übertragungsrate von 6,185 MBit/s gemessen. Die Bitfehlerhäufigkeit liegt daher etwa im erwarteten Wertebereich (zwischen 10^{-4} und 10^{-3} Fehlern pro Bit).

Für jede Serie aus Messwerten wurde der Mittelwert berechnet. In Abbildung 5.12 ist das Ergebnis der Bitfehler ratesmessung dargestellt.

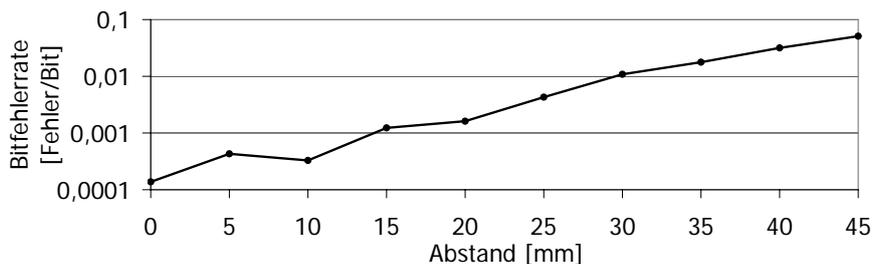


Abbildung 5.12: Bitfehler rate bei einer Framelänge von 2048 Bit in der Mitte der Leseantenne mit verschiedenen Abständen

5.4.2 Datensicherungsschicht

Auf der Ebene der Datensicherungsschicht wurde die Paketfehler rate (PER), d. h. der Anteil der verworfenen Pakete zu den insgesamt übertragenen Paketen, gemessen. Ziel dieser Messungen ist die Bestimmung der optimalen Paketgröße. Das ist jene Paketgröße, bei der die gegenläufigen Faktoren, Paketfehler rate und Nutzdatenanteil, ihr Optimum erreichen und damit die effektive Datenrate maximal ist.

Zur Messung der Paketfehlerhäufigkeit werden Zufallsdaten in Pakete zerlegt und am Transponder gespeichert. Der Transponder ist in einem Abstand von einem Zentimeter über der Antenne des Lesegeräts fixiert und die Mittelpunkte beider Antennen fallen wieder auf einen Punkt zusammen. Ein PC-Programm wertet anschließend die empfangenen Daten aus und zählt die gültigen und fehlerhaften Datenpakete. Darüber hinaus ermittelt es die effektive Datenübertragungsrate. Der gesamte Vorgang wird für

unterschiedliche Rahmen- und Paketgrößen wiederholt.

Abbildung 5.13 zeigt die Paketfehlerrate in Abhängigkeit der Paketgröße und der Rahmengröße. Die Messergebnisse bestätigen, dass die Paketfeh-

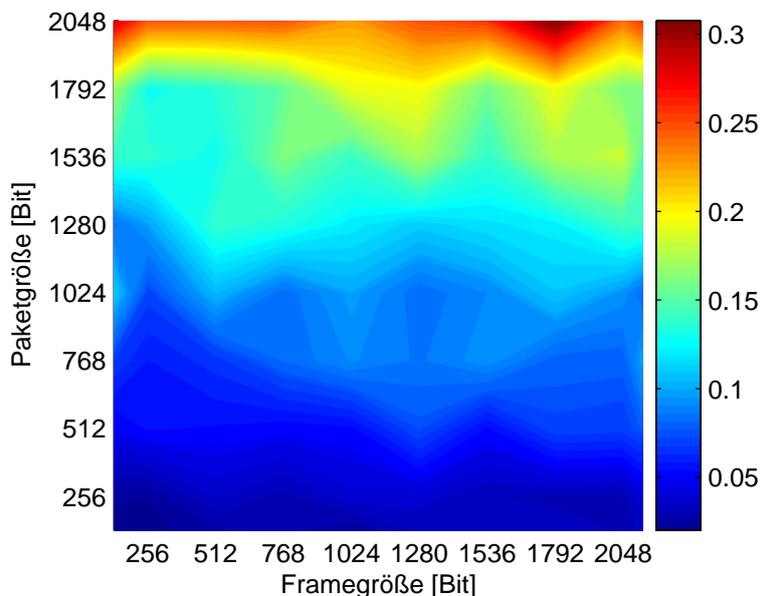


Abbildung 5.13: Paketfehlerhäufigkeit (PER), in Fehlern pro Paket, in Abhängigkeit der Paketgröße und der Rahmengröße

lerhäufigkeit mit zunehmender Paketgröße ansteigt. Umso kleiner die Datenpakete sind, desto geringer ist der Anteil an fehlerhaft empfangenen Paketen. Weiters ist deutlich erkennbar, dass die Rahmengröße keinen Einfluss auf die Paketfehlerrate hat. Die Paketfehlerhäufigkeit kann dadurch also auch einfach in Abhängigkeit der Paketgröße dargestellt werden (Abb. 5.14).

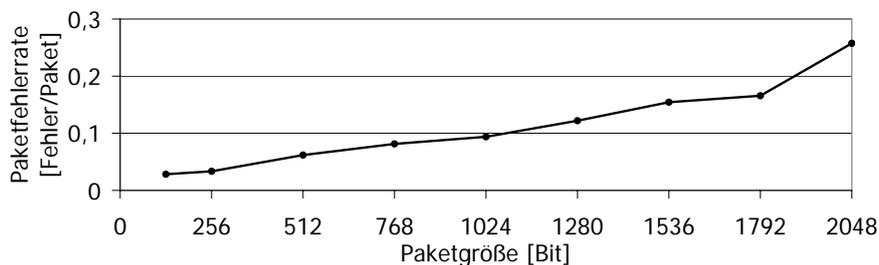


Abbildung 5.14: Paketfehlerhäufigkeit (PER), gemittelt über alle Rahmengrößen, in Abhängigkeit der Paketgröße

Als weiterer Messwert wurde die Anzahl an gültig empfangenen Datenbytes gemessen. Dieser Wert ist abhängig von der Paketfehlerrate und dem,

durch die Paket- und Rahmengröße bestimmten, Nutzdatenanteil. Mit Hilfe dieser Messgröße lässt sich die effektive Nutzdatenübertragungsrate bestimmen. Diese ist in Abbildung 5.15 in Abhängigkeit der Paketgröße und der Rahmengröße dargestellt. Die Messergebnisse zeigen, dass eine optimale

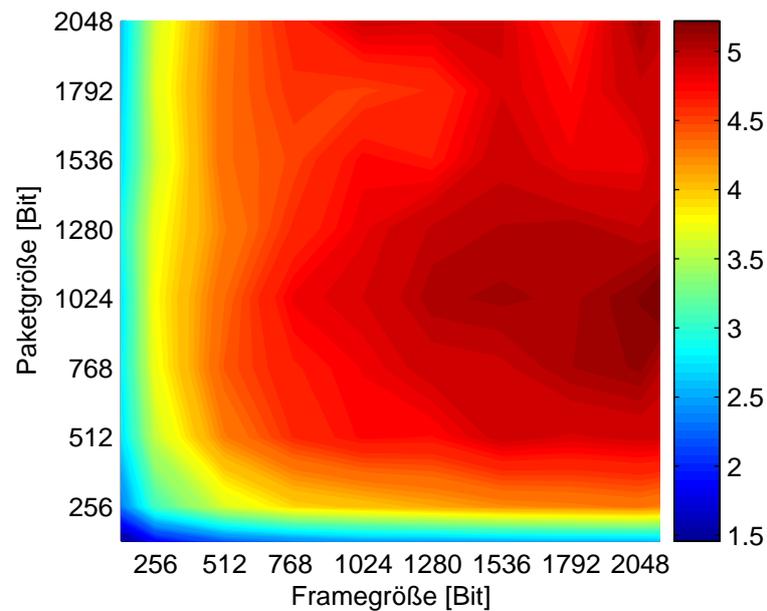


Abbildung 5.15: Effektive Nutzdatenübertragungsrate, in MBit/s, in Abhängigkeit der Paketgröße und der Rahmengröße

Datenübertragung bei einer Paketgröße von 1024 Bit und der maximalen Framegröße erreicht wird. Es ist daher sinnvoll, diese Parameter für die Datenübertragung zu verwenden.

Kapitel 6

Demonstrator: Bildübertragung

Die Systemkomponenten und die Implementierung des Übertragungsprotokolls wurden zu einem vollständigen Demonstrator zusammengefügt. Mit diesem Demonstrator ist es nun möglich, eine Datei mit dem PC über ein Programmiergerät auf dem Transponder zu speichern. Diese Datei lässt sich anschließend über die VHD-Schnittstelle zum Lesegerät und von dort wieder weiter zu einem PC übertragen.

Zur Vorführung dieser Datenübertragung (z. B. auf einer Messe), ist es notwendig, diese Datenübertragung anschaulich zu gestalten. Anhand der in Abschnitt 3.1 erarbeiteten beispielhaften Einsatzgebiete für höhere Datenraten, wurde die Übertragung einer digitalen Fotografie als typischer Anwendungsfall ausgewählt. Sowohl bei einem Reisepass, als auch bei verschiedenen NFC-Anwendungen werden Bilddaten über die kontaktlose Schnittstelle übertragen.

6.1 Gesamtsystem

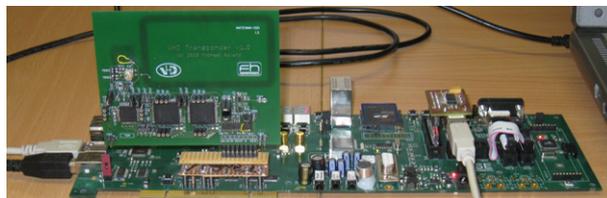


Abbildung 6.1: Programmiergerät beim Speichern einer Datei auf dem Transponder

Das Gesamtsystem besteht aus einem Programmiergerät (Abb. 6.1), einem Transponder (Abb. 6.3), einem Lesegerät (Abb. 6.2) und je einem PC-

6. Demonstrator: Bildübertragung

85

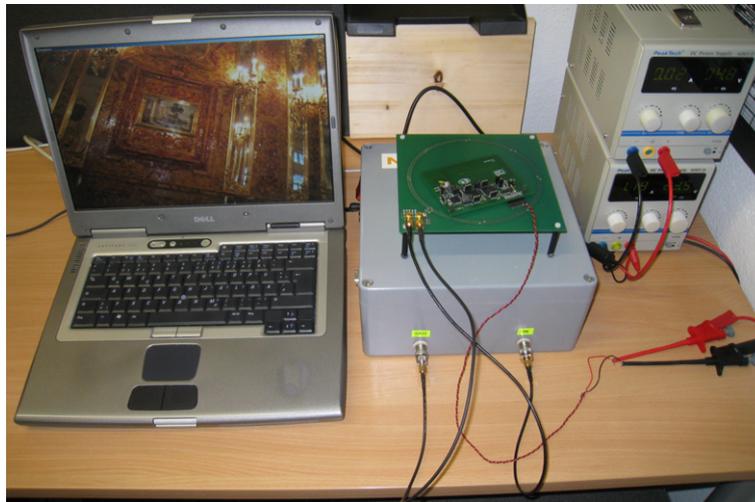


Abbildung 6.2: Lesegerät beim Lesen einer Datei aus dem Transponder



Abbildung 6.3: Transponder mit Datenspeicher

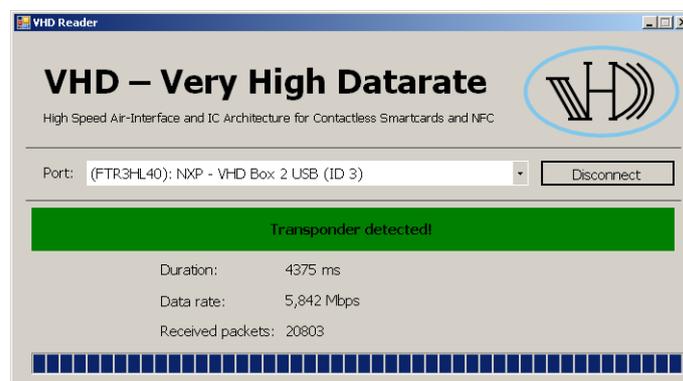


Abbildung 6.4: Anwendung zur Interaktion mit dem Lesegerät

Programm zur Kommunikation mit dem Programmiergerät bzw. dem Lesegerät.

Die Anwendung zur Interaktion mit dem Lesegerät (Abb. 6.4) wurde so erweitert, dass sie die empfangene Datei nicht nur einfach abspeichert, sondern auch unmittelbar auf dem Bildschirm anzeigt.

6.2 Ergebnisse der Bildübertragung

Um die Bildübertragung zu analysieren, wurde die Datenübertragung mit verschiedenen Abständen zwischen dem Transponder und der Antenne des Lesegeräts durchgeführt. Dabei wurde gemessen, wie lange die Übertragung einer vollständigen Bilddatei benötigt (Abb. 6.5). Die daraus resultierende

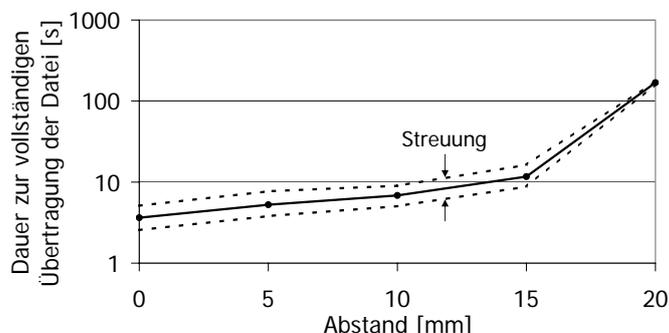


Abbildung 6.5: Dauer zur vollständigen Übertragung der Datei in Abhängigkeit des Abstands zwischen dem Transponder und der Antenne des Lesegeräts

effektive Nutzdatenübertragungsrate ist in Abbildung 6.6 als Funktion des Antennenabstandes dargestellt. Es ist erkennbar, dass diese Datenrate deut-

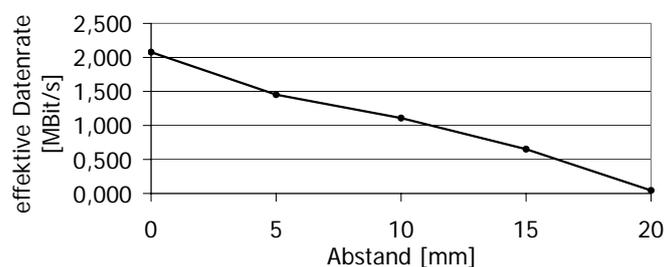


Abbildung 6.6: Effektive Nutzdatenübertragungsrate in Abhängigkeit des Abstands zwischen dem Transponder und der Antenne des Lesegeräts

lich unter dem in Abbildung 5.15 ermittelten Wert liegt. Das liegt daran, dass einzelne Teilpakete der Datei nur sehr selten fehlerfrei übertragen werden,

6. Demonstrator: Bildübertragung

87

während der Großteil der Teilpakete häufig fehlerfrei beim Empfänger ankommt.

Die Ergebnisse lassen erkennen, dass der Demonstrator für Reichweiten von bis zu 15 Millimetern eine brauchbare Übertragung ermöglicht.

Kapitel 7

Zusammenfassung und Ausblick

In den vorhergehenden Kapiteln wurden bestehende Konzepte und Prototypen zu einer Demonstratorplattform erweitert. Mit Hilfe eines neuen Übertragungsprotokolls lassen sich nun vollständige Dateien übertragen. Auf der Basis dieser Plattform wurde anschließend ein Demonstrationsszenario zur Übertragung einer Bilddatei aufgebaut. Anhand dieser Beispielanwendung ließ sich die Funktionsfähigkeit der Plattform erfolgreich zeigen. Darüber hinaus bestätigten Messungen der Fehlerhäufigkeit dieses Ergebnis.

Die entworfenen Schaltungen, die entwickelten Programme (inklusive Quelltext) und die Messergebnisse befinden sich auf der beigelegten CD-ROM.

7.1 Ausblick

Trotz der erfolgreichen Durchführung einer Datenübertragung sind einige Punkte unbehandelt geblieben.

7.1.1 Passiver Transponder

Der für diesen Demonstrator entwickelte Transponder ist semi-passiv, d. h. batteriegestützt. Mit Hilfe einer Energiebilanz könnte man ermitteln, ob ein passiver Betrieb möglich ist. Dazu müsste die Energieaufnahme der einzelnen Transponderkomponenten der verfügbaren Energie aus dem HF-Feld des Lesegeräts gegenübergestellt werden. Für eine exakte Energiebilanz ist daher eine Analyse der Stromaufnahme der einzelnen Komponenten (Taktableitung, CPLD, Flash-Speicher, . . .) notwendig. Des Weiteren sind bei der Stromaufnahme Unterschiede zwischen dem laufenden Betrieb und der Initialisierungsphase zu beachten. Beim CPLD kann z. B. davon ausgegangen

werden, dass die Initialisierungsphase deutlich mehr Energie benötigt.

7.1.2 Stabiler Zustand des CPLD

Der momentane batteriebetriebene Transponder sieht keine Möglichkeit vor, das CPLD durch ein Resetsignal in einen definierten Zustand zu bringen. Der Resetzustand ist daher nur durch ein ab- und anklappen der Stromversorgung (Batterie) erreichbar. Für einen kontinuierlichen Betrieb innerhalb des HF-Feldes des Lesegeräts würde das noch kein Problem darstellen. Nachdem der Transponder jedoch das HF-Feld des Lesegeräts im laufenden Betrieb mehrmals betreten und wieder verlassen kann, ist die Taktversorgung des CPLDs unregelmäßig. D. h. der Takt wird zu beliebigen Zeitpunkten unterbrochen und wieder fortgesetzt. Zudem können, durch die Bewegung des Transponders und die damit verbundene Schwankung der Feldstärke, entstehende Störungen zusätzliche Taktflanken bewirken. Wenn durch das unregelmäßige Taktsignal die Spezifikation des CPLDs verletzt wird, kann das CPLD in ungültige Zustände übergehen. Diese Zustände können typischerweise nur durch einen Reset verlassen werden.

Eine Möglichkeit zur Behebung dieser Problematik wäre eine Schaltung, die immer beim Betreten des HF-Feldes des Lesegeräts einen Reset des CPLDs durchführt.

7.1.3 Aufbereitung des Empfangssignals

Im aktuellen Demonstrator wird keinerlei Aufbereitung des Empfangssignals durch entsprechende analoge oder digitale Filter durchgeführt.

Wie in Abschnitt 2.3.3 gezeigt, überwiegt das ausgesendete Trägersignal deutlich gegenüber dem Modulationsanteil. Daher benötigt der Analog-Digital-Umsetzer eine hohe Auflösung, damit das Datensignal rekonstruiert werden kann. Wird das Empfangssignal vor der Digitalisierung durch analoge Filter geeignet aufbereitet, kann der Detektionsaufwand im Empfänger deutlich reduziert werden.

Es gibt verschiedene Methoden zur Unterdrückung des Trägersignals. Zum einen lässt sich das 13,56-MHz-Trägersignal durch ein schmalbandiges Filter (z. B. Quarzbandsperr) reduzieren. Zum anderen kann die kombinierte Antenne des Lesegeräts in eine Sendeantenne zur Abstrahlung des Trägersignals und eine Empfangsantenne zum Empfang des auf den Träger aufmodulierten Signals aufgeteilt werden. Durch spezielle Bauformen der Empfangsantenne lässt sich das Trägersignal ohne zusätzliche Filterstrukturen praktisch vollständig aus dem Empfangssignal eliminieren. Wie bereits in Abschnitt 3.2.1 erwähnt, wird die kommerzielle Nutzung dieses Konzepts durch mehrere Patente erschwert. Allerdings wäre die Implementierung im Rahmen einer prototypischen Demonstratorplattform davon nicht betroffen. Somit ist auch die weitere Untersuchung von Zwei-Antennen-Systemen für

zukünftige Demonstratorplattformen interessant.

7.1.4 Uplinkkanal

Die Implementierung des Uplinkkanals vom Lesegerät zum Transponder würde mehrere Vorteile mit sich bringen. Statt dem Karussellverfahren könnte ein gezielteres Fehlerkorrekturverfahren (z. B. ARQ) eingesetzt werden. Weiters ist dadurch eine dynamische Anpassung der Datenübertragungsrate an die Fehlerhäufigkeit möglich und somit eine Verbesserung der Übertragung möglich.

Ein Uplinkkanal mit höherer Datenrate ist Gegenstand laufender Forschungsarbeit. Für einfache Statusmeldungen reicht jedoch auch ein Rückkanal auf der Basis bestehender RFID-Systeme, mit einer niedrigen Datenübertragungsrate, aus. Für diese Übertragung sind bereits bestehende Implementierungen vorhanden. Es ist daher anzunehmen, dass die Implementierung eines langsamen Rückkanals in die Demonstratorplattform ohne größere Probleme möglich ist.

Abkürzungsverzeichnis

APDU	Application protocol data unit (Kommunikationseinheit auf Anwendungsebene)
API	Application programming interface (Programmierschnittstelle)
ARQ	Automatic repeat request (automatische Wiederholanforderung)
ASK	Amplitude-shift keying (Amplitudenumtastung)
BER	Bit error rate (Bitfehlerhäufigkeit)
BPSK	Binary phase-shift keying (binäre Phasenumtastung)
CCITT	Comité Consultatif International Téléphonique et Télégraphique
CPLD	Complex programmable logic device (komplexer programmierbarer Logikbaustein)
CRC	Cyclic redundancy check (zyklische Redundanzprüfung)
DLL	Delay-locked loop
DSM-CC	Digital Storage Media Command and Control
EEPROM	Electrically erasable and programmable read-only memory
FIFO	First in, first out
FPGA	Field-programmable gate array (programmierbarer integrierter Schaltkreis)
FSK	Frequency-shift keying (Frequenzumtastung)
FSM	Finite state machine (endlicher Zustandsautomat)
HF	Hochfrequenz
IC	Integrated circuit (integrierter Schaltkreis)
IEC	International Electrotechnical Commission
I ² C	Inter-integrated circuit bus
ISI	Intersymbolinterferenz

ISO	International Organization for Standardization
LDO	Low-dropout regulator (Spannungsregler)
LSB	Least significant bit (Bit mit dem niedrigsten Stellenwert)
MOS-FET	Metalloxid-Feldeffekttransistor
MPEG	Moving Picture Experts Group
MSB	Most significant bit (Bit mit dem höchsten Stellenwert)
NFC	Near Field Communication
NRZ-L	Non-return-to-zero level
OOK	On-off keying (Ein-Aus-Tastung)
OSI	Open Systems Interconnection
PC	Personal Computer
PCB	Printed circuit board (gedruckte Schaltung)
PCD	Proximity coupling device
PER	Packet error rate (Paketfehlerhäufigkeit)
PICC	Proximity integrated circuit card
PSK	Phase-shift keying (Phasenumtastung)
RFID	Radio Frequency Identification
SIM	Subscriber Identity Module
SMS	Short Message Service
SPI	Serial peripheral interface bus
TCP/IP	Transmission control protocol/Internet protocol
UART	Universal synchronous receiver-transmitter
UHF	Ultrahochfrequenz
USB	Universal Serial Bus
VCD	Vicinity coupling device
VHD	Very High Datarate
VHDL	Very high speed integrated circuits hardware description language
VICC	Vicinity integrated circuit card
WLAN	Wireless local area network

Symbolverzeichnis

A	Fläche [m ²]
B	magnetische Flussdichte [T]
B_w	Bandbreite [Hz]
C	Kapazität [F]
f_r	Resonanzfrequenz [Hz]
H	magnetische Feldstärke [A/m]
I	Stromstärke [A]
k	Kopplungsfaktor
L	Induktivität [H]
M	Gegeninduktivität [H]
μ_0	magnetische Feldkonstante [$4\pi \cdot 10^{-7}$ H/m]
μ_r	relative Permeabilität
Φ	magnetischer Fluss [Wb]
Ψ	magnetische Flussverkettung [Wb]
Q	Gütefaktor
U_{ind}	induzierte Spannung [V]
U_m	magnetische Spannung [A]
Z	Impedanz [Ω]

Literaturverzeichnis

- [1] J. A. E. BRUCKBAUER: *Simulation und Realisierung eines hochratigen 13 MHz RFID Systems für den Downlink*. Diplomarbeit, Fachhochschule Hagenberg, Studiengang Hardware/Software Systems Engineering, Hagenberg, Austria, Sep. 2006.
- [2] J. J. CARR: *Practical antenna handbook*. McGraw-Hill Professional, 4th ed., 2001.
- [3] R. S. CHERNOCK, R. J. CRINON, J. R. MICK, and M. A. DOLAN: *Data Broadcasting: Understanding the ATSC Data Broadcast Standard*. McGraw-Hill Professional, 2001.
- [4] K. FINKENZELLER: *RFID-Handbuch*. Hanser München Wien, 4. Aufl., Aug. 2006.
- [5] FTDI CHIP: *FT245R USB FIFO IC Datasheet V1.05*, Jan. 2006.
- [6] J. D. GIBSON (ed.): *The Communications Handbook*. CRC Press, 2nd ed., 2002.
- [7] S. GRÜNBERGER and J. LANGER: *Analysis and test results of tunneling IP over NFCIP-1*. In *Proceedings of the 1st Intl. IEEE Workshop on Near Field Communication*, pp. 93–97, Hagenberg, Austria, Feb. 2009. IEEE Computer Society.
- [8] G. HAIDER: *Ein RFID-System mit hohen Datenraten*. Diplomarbeit, Johannes Kepler Universität Linz, Institut für Nachrichtentechnik/Informationstechnik, Linz, Austria, Feb. 2006.
- [9] M. HILLUKKALA, M. HEISKANEN, and A. YLISAUKKO-OJA: *Practical implementations of passive and semi-passive NFC enabled sensors*. In *Proceedings of the 1st Intl. IEEE Workshop on Near Field Communication*, pp. 69–74, Hagenberg, Austria, Feb. 2009. IEEE Computer Society.
- [10] W. P. KOWALK und M. BURKE: *Rechnernetze: Konzepte und Techniken der Datenübertragung in Rechnernetzen*. B. G. Teubner Stuttgart, 1994.

- [11] K. KÜPFMÜLLER: *Einführung in die theoretische Elektrotechnik*. Springer Berlin Heidelberg New York, 10. Aufl., 1973.
- [12] A. MAIER: *Entwurf & Analyse von Transceiverstrukturen in hoch-ratigen RFID-Systemen*. Diplomarbeit, Fachhochschule Technikum Kärnten, Studiengang Telematik/Netzwerktechnik, Gratkorn, Austria, Juni 2007.
- [13] NEW MEDIA DEVELOPMENT ASSOCIATION: *Test Results of the e-Passport Interoperability Test Session at Tsukuba in March 8-10, 2005*, Apr. 2005.
- [14] NORM ISO/IEC 10373-6:2001: *Identification cards – Test methods – Part 6: Proximity cards*.
- [15] NORM ISO/IEC 10536-3:1996: *Identification cards – Contactless integrated circuit(s) cards – Part 3: Electronic signals and reset procedures*.
- [16] NORM ISO/IEC 14443-1:2008: *Identification cards – Contactless integrated circuit cards – Proximity cards – Part 1: Physical characteristics*.
- [17] NORM ISO/IEC 14443-2:2001: *Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 2: Radio frequency power and signal interface*.
- [18] NORM ISO/IEC 14443-3:2001: *Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 3: Initialization and anticollision*.
- [19] NORM ISO/IEC 14443-4:2008: *Identification cards – Contactless integrated circuit cards – Proximity cards – Part 4: Transmission protocol*.
- [20] NORM ISO/IEC 15693-1:2000: *Identification cards – Contactless integrated circuit(s) cards – Vicinity cards – Part 1: Physical characteristics*.
- [21] NORM ISO/IEC 15693-2:2006: *Identification cards – Contactless integrated circuit cards – Vicinity cards – Part 2: Air interface and initialization*.
- [22] NORM ISO/IEC 15693-3:2001: *Identification cards – Contactless integrated circuit(s) cards – Vicinity cards – Part 3: Anticollision and transmission protocol*.

- [23] NORM ISO/IEC 18092:2004: *Information technology – Telecommunications and information exchange between systems – Near Field Communication – Interface and Protocol (NFCIP-1)*.
- [24] NORM ISO/IEC 21481:2005: *Information technology – Telecommunications and information exchange between systems – Near Field Communication Interface and Protocol -2 (NFCIP-2)*.
- [25] NXP SEMICONDUCTORS: *I²C-bus specification and user manual (Rev. 03)*, June 2007.
- [26] J. PEARSON and T. MOISE: *The Advantages of FRAM-based Smart ICs for Next-gen Electronic IDs*. Wireless Design & Development Asia, May 2008.
- [27] I. SÁNCHEZ, J. RIEKKI, and M. PYYKKÖNEN: *Touch & Compose: Physical User Interface for Application Composition in Smart Environments*. In *Proceedings of the 1st Intl. IEEE Workshop on Near Field Communication*, pp. 61–66, Hagenberg, Austria, Feb. 2009. IEEE Computer Society.
- [28] H. STÖCKER (Hrsg.): *Taschenbuch der Physik*. Verlag Harri Deutsch, Frankfurt am Main, 5., korr. Aufl., 2004.
- [29] H. WITSCHNIG und E. MERLIN: *Über Geschichte, physikalische Grundlagen und Applikationen der RFID-Technologie*. Elektrotechnik und Informationstechnik, 123(3):61–71, März 2006.
- [30] XILINX: *CoolRunner-II CPLD Family Product Specification V3.1*, Sept. 2008.
- [31] XILINX: *Using Delay-Locked Loops in Spartan-II/III FPGAs V1.2*, June 2008.
- [32] J. ZIRBESEGGER, M. GEBHART, E. MERLIN, and E. LEITGEB: *Extending the analogue performance of integrated 13.56 MHz proximity reader chips*. Elektrotechnik und Informationstechnik, 124(11):369–375, Nov. 2007.