



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# Bachelorarbeit

**Stefan Ralfs**

**Kommunikation im Batteriemanagementsystem für  
Elektrofahrzeuge – Hard- und Softwareentwicklung,  
Bewertung des Zeitverhaltens**

*Fakultät Technik und Informatik  
Studiendepartment Informatik*

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

Stefan Ralfs

**Kommunikation im Batteriemanagementsystem für  
Elektrofahrzeuge – Hard- und Softwareentwicklung, Bewertung des  
Zeitverhaltens**

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung  
im Studiengang Bachelor of Science Elektro- und Informationstechnik  
am Department Informations- und Elektrotechnik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuer Prüfer: Prof. Dr.-Ing. Karl-Ragmar Riemschneider  
Zweitgutachter: Prof. Dr.-Ing Lutz Leutelt

Eingereicht am: 27. September 2024

**Stefan Ralfs**

### **Thema der Arbeit**

Kommunikation im Batteriemanagementsystem für Elektrofahrzeuge – Hard- und Softwareentwicklung, Bewertung des Zeitverhaltens

### **Stichworte**

Mikrocontroller, UART, Messung, synchrone Messung, elektrochemischen Impedanzspektroskopie, optische Kommunikation, induktive Kommunikation, Coreless Transformer (CT), IrDA-Standard, Serial Infrared (SIR), C-Quellcode, Jitter, Experimentalaufbau

### **Kurzzusammenfassung**

Diese Arbeit befasst sich mit dem Einfluss zweier Kommunikationsvarianten auf die Synchronität bei der Strom- und Spannungsmessung einer mit Wechselstrom angeregten Batterie für die Erstellung einer elektrochemischen Impedanzspektroskopie. Dabei werden die verwendeten Kommunikationsvarianten auf ihre Echtzeiteigenschaften und ihr Verhalten bei unterschiedlichen Konfigurationen untersucht und ein experimenteller Aufbau zur Erprobung der synchronen Messung entwickelt.

**Stefan Ralfs**

**Title of Thesis**

Communication in the battery management system for electric vehicles - Hardware and software development, evaluation of timing behavior

**Keywords**

Microcontroller, UART, Measurement, Synchronous Measurement, Electrochemical Impedance Spectroscopy, Optical Communication, Inductive Communication, Coreless Transformer (CT), IrDA Standard, Serial Infrared (SIR), C Source Code, Jitter, Experimental Setup

**Abstract**

This work addresses the influence of two communication methods on the synchrony of current and voltage measurements in a battery stimulated by alternating current for the purpose of conducting electrochemical impedance spectroscopy. The communication methods used are examined in more detail for their real-time properties and behavior under various conditions, and an experimental setup is developed to test synchronous measurement.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>viii</b>
<b>Tabellenverzeichnis</b>	<b>x</b>
<b>Abkürzungsverzeichnis</b>	<b>xi</b>
<b>1. Einleitung</b>	<b>1</b>
1.1. Ausgangssituation . . . . .	2
1.2. Aufgabenstellung und Ziel . . . . .	2
<b>2. Grundlagen</b>	<b>3</b>
2.1. Batteriemanagementsystem . . . . .	3
2.1.1. Grundkonzepte des Batteriemanagementsystems . . . . .	3
2.1.2. Verteiltes Batteriemanagementsystem . . . . .	4
2.2. Antriebsbatterien . . . . .	5
2.2.1. Chemische Energiespeicher . . . . .	5
2.2.2. Traktionsbatterie . . . . .	7
2.3. Batteriemesstechnik . . . . .	7
2.4. Elektrochemische Impedanzspektroskopie . . . . .	8
2.4.1. Funktion . . . . .	9
2.4.2. Impedanzspektrum . . . . .	9
2.5. Jitter . . . . .	11
<b>3. Stand der Technik</b>	<b>13</b>
3.1. Kommunikationsvarianten . . . . .	13
3.1.1. Induktive Kommunikation . . . . .	13
3.1.2. Optische Kommunikation im Infrarot-Bereich . . . . .	14
3.2. Kommunikationstopologien . . . . .	16
3.3. Buszugriffsverfahren . . . . .	18
3.3.1. Kontrollierte Buszugriffsverfahren . . . . .	18
3.3.2. Zufällige Buszugriffsverfahren . . . . .	19
<b>4. Voruntersuchungen</b>	<b>21</b>
4.1. Versuchsaufbau . . . . .	21
4.2. Softwareimplementierung . . . . .	23
4.3. Baudatenbestimmung . . . . .	24
4.3.1. Baudatenbestimmung des Digitalisolators . . . . .	25

## Inhaltsverzeichnis

---

4.3.2. Baudatenbestimmung des Infrarot-Trancievers . . . . .	26
4.4. Jitter-Messung . . . . .	30
4.4.1. Jitter des Digitalisolator . . . . .	30
4.4.2. Jitter des Infrarot-Transceiver . . . . .	33
4.5. Einfluss der Strombegrenzung . . . . .	35
4.5.1. Grenzwerte der Funktionalität . . . . .	35
4.5.2. Auswirkungen auf den Bauteil-Jitter . . . . .	38
4.5.3. Auswirkungen auf die Stromaufnahme . . . . .	41
<b>5. Experimentalbau</b>	<b>43</b>
5.1. Systementwurf . . . . .	43
5.2. Hardware- und Systemaufbau . . . . .	45
5.3. Anpassungen . . . . .	46
5.4. Mess- und Versuchsplan . . . . .	47
5.5. Testsoftware . . . . .	50
5.5.1. Matlab Testsoftware . . . . .	51
5.5.2. Master-Modul Software . . . . .	52
5.5.3. Slave-Modul Software . . . . .	53
5.6. Messungen . . . . .	54
5.6.1. Jitter-Messung des Mikrocontrollers (XMC1404) . . . . .	54
5.6.2. Messung zur Untersuchung der Analog-Digital-Wandler (analog-to-digital converter) (ADC)-Verarbeitungszeit . . . . .	56
5.7. Funktionstest . . . . .	58
<b>6. Auswertung und Bewertung</b>	<b>59</b>
6.1. Synchronisation und Synchronisationsfehler . . . . .	59
6.1.1. Berechnung der Verzögerungszeiten . . . . .	59
6.1.2. Einfluss der Verzögerungszeiten auf die Phasenverschiebung . . . . .	60
6.1.3. Bewertung der Synchronisation . . . . .	61
6.2. Ausblick . . . . .	62
<b>Literaturverzeichnis</b>	<b>64</b>
<b>A. Anhang</b>	<b>68</b>
A.1. Kapitel: Voruntersuchungen . . . . .	68
A.1.1. C-Quellcode . . . . .	68
A.1.2. Abhängigkeiten der DAVE-Apps . . . . .	73
A.2. Hardwarespezifisch . . . . .	73
A.2.1. Infrarot-Tranciver Header . . . . .	73

*Inhaltsverzeichnis*

---

A.2.2. Infrarot-Tranciver Quellcode . . . . .	74
<b>Selbstständigkeitserklärung</b>	<b>77</b>

# Abbildungsverzeichnis

1.	Übersicht Batterietechnologien im Ragone Diagramm (aus [4]) . . . . .	6
2.	Impedanzspektren einer Lithiumbatterie bei verschiedenen überlagerten Ladeströmen (Ladezustand (State of Charge) (SoC) = 80%, Temperatur 25°C, Nennkapazität 7 Ah) (aus [32]) . . . . .	10
3.	Pin-Konfiguration des ISOFACE™ digital isolators 2DIB141xF (aus [1]) . . . . .	14
4.	Infrarot Transceiver Modul TFDU4101 (aus [36]) . . . . .	15
5.	Pin-Konfiguration des Infrarot Transceiver Moduls TFDU4101 (aus [36]) . . . . .	16
6.	Schaltplan des Messaufbaus zur Untersuchung der Kommunikationsvarianten . .	22
7.	Hardwareaufbau zur Untersuchung der Kommunikationsvarianten . . . . .	22
8.	Pseudo-Random Bit Sequence (PRBS)-Daten-Übertragung der UART-Schnittstelle des XMC1100 mittels Infrarot-Transceiver bei einer Baudrate von 115.2 kbit/s .	27
9.	PRBS-Daten-Übertragung der UART-Schnittstelle des XMC1100 mittels Infrarot-Transceiver bei einer Baudrate von 204.800 bit/s . . . . .	28
10.	PRBS-Daten-Übertragung der UART-Schnittstelle des XMC1100 mittels Infrarot-Transceiver bei einer Baudrate von 217.600 bit/s . . . . .	28
11.	Frequenz-Histogramm der UART-Schnittstelle des XMC1100 vor (links) und nach dem Digitalisolator ISOFACE (rechts) bei einer Baudrate von 3 Mbit/s . . . . .	31
12.	Darstellung einer UART-Bitrate mit idealer (oben) und mit zu hoher (unten) Eingangsfrequenz . . . . .	32
13.	Frequenz-Histogramm der UART-Schnittstelle des XMC1100 vor (links) und nach (rechts) dem Infrarot-Transceiver bei einer Baudrate von 192 kbit/s . . . . .	33
14.	Darstellung einer UART-Bitrate für Infrared Data Association (IrDA)-Anwendungen mit idealer Übertragungsrate . . . . .	34
15.	PRBS-Datenübertragung bei einer Datenrate 115.2 kbit/s und einem $0.2\ \Omega$ großem Vorwiderstand . . . . .	36
16.	PRBS-Datenübertragung bei einer Datenrate 115.2 kbit/s und einem langsam erhöhtem Widerstandswert von $0.2\ \Omega$ bis $10\ k\Omega$ . . . . .	37
17.	PRBS-Datenübertragung bei einer Datenrate 115.2 kbit/s und einem $6\ k\Omega$ großem Vorwiderstand . . . . .	38
18.	Histogramme bei einer Baudrate von 192 kbit/s unterschiedlichen Vorwiderständen	40
19.	Abhängigkeit des Bauteil-Jitters vom verwendeten Vorwiderstand . . . . .	41
20.	Mögliche Telegramme für die Universal-Asynchronous-Receive-Transmit (UART)-Kommunikation . . . . .	45
21.	Messplatine mit bestücktem Mikrocontroller (XMC1404) und Infrarot-Transceiver (TFDU4101)	46

## *Abbildungsverzeichnis*

---

22.	Ausschnitt der Trägerplatine mit bestückten Digitalisolatoren (2DIB1410F) . . . . .	46
23.	Lichtleitkörper an dem Infrarot-Transceiver der Messplatine . . . . .	46
24.	Schematische Darstellung zur Messung der ADC-Verarbeitungszeit . . . . .	49
25.	Flussdiagramm der Software des Matlab-Tests . . . . .	51
26.	Flussdiagramm der Software des Master-Moduls . . . . .	52
27.	Flussdiagramm der Software des Slave-Moduls . . . . .	53
28.	Jitter des Mikrocontroller XMC1404 bei einer Baudrate von 1.25 Mbit/s . . . . .	55
29.	Jitter des Mikrocontroller XMC1404 bei einer Baudrate von 192 kbit/s . . . . .	56
30.	Messung der ADC-Verarbeitungszeit mit 8 Eingängen . . . . .	57

## Tabellenverzeichnis

1.	Übersicht der Datenraten und Modulationstechniken der IrDA-Standards [15] . . . . .	16
2.	Minuten bis zum Auftreten eines Fehlers in drei Versuchen . . . . .	30
3.	Einfluss des Jitters durch den Digitalisolator bei einer Baudrate von 3 Mbit/s . . . . .	31
4.	Einfluss des Jitters durch den Infrarot-Transceiver bei einer Baudrate von 192 kbit/s	34
5.	Spannung am Vorwiderstand, Strom und Jitter-Werte für verschiedene Widerstände	41
6.	Stromverbrauch bei verschiedenen Vorwiderständen während der IrDA-Übertragung	42
7.	Jitter des Mikrocontrollers XMC1404 bei verschiedenen Baudraten . . . . .	54
8.	Jitter des Mikrocontrollers XMC1404 bei 192 kBit/s . . . . .	55
9.	Gemessene Gesamtverarbeitungszeiten für unterschiedliche Kanalanzahlen . . . . .	57
10.	Berechnete Verarbeitungszeiten pro Kanal durch Subtraktion der Messwerte . . . . .	58
11.	Phasenverschiebung für verschiedene Frequenzen . . . . .	61

## Abkürzungen

**HAW** Hochschule für Angewandte Wissenschaften

**Pkw** Personenkraftwagen

**BMS** Batteriemanagementsystem

**Akku** Akkumulator

**EIS** elektrochemische Impedanzspektroskopie

**ProMoBiS** Progressive Multizell-Verbund-Konzepte für Batteriesysteme mit integrierter Sensorik

**PCLK** Peripherie Clock (Peripheral Clock)

**PRBS** Pseudo-Random Bit Sequence

**BER** Bitfehlerrate (bit error rate)

**UART** Universal-Asynchronous-Receive-Transmit

**IrDA** Infrared Data Association

**ADC** Analog-Digital-Wandler (analog-to-digital converter)

**GPIO** General Purpose Input/Output

**IDE** integrierte Entwicklungsumgebung (integrated development environment)

**SIR** Serial Infrared

**MIR** Medium Infrared

**FIR** Fast Infrared

**VFIR** Very Fast Infrared

**UFIR** Ultra Fast Infrared

**Giga-IR** Giga Infrared

**NRZI** Non-Return to Zero Inverted

**RZI** Return to Zero Inverted

**NRZ** Non-Return to Zero

**RZ** Return to Zero

---

**4PPM** 4 Pulse Position Modulation

**SoC** Ladezustand (State of Charge)

**SoH** Gesundheitszustand (State of Health)

**DoD** Entladetiefe (Depth of Discharge)

# 1. Einleitung

Die heutige Welt setzt bei der Antriebsart von Fahrzeugen im Personennahverkehr immer mehr auf Elektromobilität. Alleine die deutsche Bundesregierung strebt dabei das ehrgeizige Ziel von “mindestens 15 Millionen vollelektrische Personenfahrzeuge (Pkw) bis 2030” an [6]. Dabei ist die verwendete elektrochemische Batterie in der modernen Automobil-Industrie von außerordentlicher Bedeutung. Das korrekte Batteriemanagement stellt demnach eine wichtige Rolle für die heutige Elektromobilität dar.

Der Begriff der “Batterie” ist hierbei eine historisch in dem Bereich etablierte Bezeichnung, die bereits auf die ersten elektrischen Antriebe in der Mitte des 19ten Jahrhunderts zurückgeht. Auch heutzutage ist der Begriff noch Standard und findet weitgehend Verwendung. Dabei bezeichnet der Begriff “Batterie” eigentlich einen elektrischen Speicher, der sich nicht wieder aufladen lässt. Korrekterweise wird in modernen elektronischen Antrieben ein Akkumulator (Akku), also ein wieder aufladbarer elektrischer Energiespeicher verwendet. [4]

Eine adäquate Messung der Pkw-Batterie in Bezug auf den Ladestand, Verbrauch, Lebensdauer und weiteren wichtigen Parametern sind folglich ein essenzieller Bestandteil der modernen Elektromobilität. Das Stichwort in diesem Zusammenhang ist Batteriemanagementsystem (BMS). Das, BMS ist dafür vorgesehen, die verschiedensten Parameter zu erfassen und zu verarbeiten. Unter anderem ist eine der herausforderndsten Aufgaben dabei die präzise Synchronisation zwischen der zentralen Strommessung und den mehreren hundert Zellsensoren bei der Batteriemessung. [4]

Ein wesentlicher Anteil dieser Thesis bezieht explizit auf den Aspekt der Stabilität im Zeitbereich und der Synchronität bei der Spannungsmessung für die Erstellung einer elektrochemische Impedanzspektroskopie (EIS). Es werden im Folgenden die wichtigsten Aspekte und Grundlagen im Zusammenhang mit den genannten Eigenschaften betrachtet und eine ingenieurtechnische Umsetzung erarbeitet.

## **1.1. Ausgangssituation**

Im Rahmen eines vom “Bundesministerium für Wirtschaft und Klimaschutz” geförderten Verbundprojekts namens Progressive Multizell-Verbund-Konzepte für Batteriesysteme mit integrierter Sensorik (ProMoBiS) [16] werden an der Hochschule für Angewandte Wissenschaften (HAW) ingenieurtechnische Konzepte im Bereich der Batteriesysteme in einem kleinen Team von wissenschaftlichen Mitarbeiten entwickelt und einige durch studentische Abschlussarbeiten unterstützt [26]. Ein Teil dessen ist die Erfassung verschiedener Batterie-Parameter mittels EIS. Dazu existieren an der HAW bereits eigene Testaufbauten und Arbeiten. Die bestehenden Projekte und Prototypen werden von wissenschaftlichen Mitarbeitern des Projekts oder von Studenten im Rahmen der Erarbeitung einer Thesis stetig erweitert, neu entwickelt oder genauer auf ihre Randbedingungen und Eigenschaften untersucht.

## **1.2. Aufgabenstellung und Ziel**

Diese Arbeit untersucht die Synchronität der Spannungsmessungen an Batterieblöcken in Kombination mit einer zentralen Strommessung im Rahmen der elektrochemischen Impedanzspektroskopie (EIS). Ein besonderer Schwerpunkt liegt dabei auf der Evaluierung zweier Kommunikationsmethoden zwischen den Mikrocontrollern, die galvanisch voneinander getrennt und taktunabhängig arbeiten. Ziel ist es, die Auswirkungen dieser Kommunikationsansätze auf die Synchronität der Messungen zu analysieren und mögliche Verzögerungen zu bewerten.

Die Hardware zur Spannungs- und Strommessung ist größtenteils vorhanden. Der Fokus dieser Arbeit liegt daher auf der Wahl der geeigneten Topologie für die Kommunikation der Mikrocontroller, um die Synchronität der Messungen sicherzustellen. Getestet und verglichen werden zwei Kommunikationsvarianten: Eine optische Übertragung über Infrarot-Transceiver sowie eine induktive Übertragung mithilfe von Digitalisolatoren.

Ein zentrales Problem stellt der Jitter dar, der durch unterschiedliche Baudaten und Latenzen der Kommunikationsmethoden entstehen kann. Diese Verzögerungen könnten zu einer Beeinträchtigung der Synchronität der Spannungsmessungen führen, was wiederum die Qualität und Genauigkeit der EIS negativ beeinflussen würde. Daher wird in dieser Arbeit untersucht, welche der beiden Kommunikationsmethoden besser geeignet ist, um eine möglichst geringe Verzögerung und hohe Synchronität der Messungen zu gewährleisten.

Das übergeordnete Ziel ist es, eine fundierte Entscheidung über die optimale Kommunikationsmethode zu treffen, die sowohl den Anforderungen an die Synchronität als auch den Bedingungen der galvanischen Trennung gerecht wird, um eine präzise und verlässliche EIS durchführen zu können.

## 2. Grundlagen

In diesem Kapitel werden die Grundlegenden physikalischen Eigenschaften und Konzepte vermittelt, die zu einem angemessenen Verständnis beitragen und bei Einordnung der ingenieurtechnischen Umsetzung relevant sind.

### 2.1. Batteriemanagementsystem

Das Batteriemanagementsystem (BMS) spielt eine zentrale Rolle bei der Überwachung, Steuerung und Optimierung von Batteriesystemen, insbesondere in Anwendungen mit Lithium-Ionen-Batterien. Die Hauptaufgaben des BMS umfassen die Überwachung von Batterieparametern, das Management der Energieflüsse sowie den Schutz vor schädlichen Betriebszuständen wie Überladung oder Tiefentladung. Im Folgenden werden sowohl die Grundkonzepte des BMS als auch das spezielle Konzept des verteilten BMS erläutert.

#### 2.1.1. Grundkonzepte des Batteriemanagementsystems

Das BMS übernimmt mehrere essenzielle Aufgaben, um den Betrieb der Batterie sicher und effizient zu gestalten. Zu den Grundfunktionen zählen:

- **Steuerung des Energieflusses:**

Das BMS setzt Grenzwerte für Lade- und Entladeströme in Abhängigkeit von der Spannung der Batterie, sowohl auf Zell- als auch auf Modulebene. Wenn diese Grenzwerte überschritten werden, können Schutzmechanismen wie interne Schütze aktiviert werden, um die Batterie zu schützen [4].

- **Messung der Batteriezustandsparameter (Batteriemesstechnik):**

Zu den primären Aufgaben des BMS gehört die kontinuierliche Messung der wichtigsten Batterieparameter: Temperatur, SoC, Strom und Spannung. Einige fortschrittliche BMS verfügen zudem über die Fähigkeit, den Innenwiderstand einzelner Zellen zu berechnen, um Rückschlüsse auf die Alterung und den allgemeinen Gesundheitszustand (State of Health) (SoH) der Batterie zu ziehen [20, 33, 4].

- **Balancing:**

Da die Zellen innerhalb eines Batteriekits im Laufe der Zeit unterschiedlich altern und Kapazitätsunterschiede aufweisen, sorgt das BMS für einen Ausgleich, um die Homogenität der Zellen sicherzustellen. Dies geschieht durch sogenannte Balancing-Methoden:

## 2. Grundlagen

---

- **Passives Balancing:** Der Strom durch die energiereicheren Zellen wird mithilfe parallel geschalteter Regelwiderstände reduziert. Dadurch werden die energieärmeren Zellen gleichmäßig bis zum gewünschten Niveau angepasst. Diese Methode ist kosten-günstig, allerdings wird überschüssige Energie in Form von Wärme abgegeben, was die Effizienz reduziert [4].
- **Aktives Balancing:** Hier wird Energie von energiereicheren Zellen auf energiearme Zellen übertragen. Diese Methode ist insgesamt effizienter, erfordert jedoch komplexe und teurere Schaltungen [4].
- **Sicherheitsmechanismen:**  
Das BMS sorgt dafür, dass die Batterie in einem sicheren Betriebsfenster arbeitet, dies schließt den Schutz von Überladung, Tiefenentladung und Überhitzung ein. Einzellensensoren spielen hierbei eine Schlüsselrolle, indem sie detaillierte Informationen über den Zustand jeder Zelle liefern [4].

### 2.1.2. Verteiltes Batteriemanagementsystem

Ein verteiltes BMS setzt auf eine dezentrale Architektur, bei der jede Zelle oder Zellgruppe individuell überwacht und gesteuert wird. Dies erhöht nicht nur die Effizienz und Präzision der Überwachung, sondern verbessert auch die Skalierbarkeit des Systems. In modernen Anwendungen, wie etwa Elektrofahrzeugen und stationären Energiespeichern, kommt diese Architektur häufig zum Einsatz [11].

- **Einzellensensoren:**  
Ein zentrales Element des verteilten BMS sind die Einzellensensoren. Diese erfassen die Spannung, Temperatur sowie den Lade- und Entladestrom jeder Zelle, um sicherzustellen, dass diese innerhalb der sicheren Betriebsgrenzen bleibt. Ein moderner Ansatz kombiniert die Daten von Einzellensensoren mit einer Cloud-basierten Datenverarbeitung, um den Zustand der Batterie in Echtzeit zu überwachen und potenzielle Fehler frühzeitig zu erkennen [11].
- **Vorteile des verteilten BMS:**
  - *Erhöhte Präzision:* Durch die Überwachung auf Zellebene können Ungleichgewichte innerhalb des Batteriesystems genauer erkannt und behandelt werden.
  - *Bessere Skalierbarkeit:* Ein verteiltes BMS lässt sich problemlos auf größere Batteriesysteme anwenden, was es besonders für den Einsatz in Elektrofahrzeugen und erneuerbaren Energiesystemen geeignet macht.

## 2. Grundlagen

---

- *Erhöhte Sicherheit:* Die detaillierte Überwachung einzelner Zellen minimiert das Risiko von gefährlichen Zuständen wie Überladung oder Überhitzung [11].
- **Kommunikationsinfrastruktur:**

Die Kommunikation zwischen den dezentralen Sensoreinheiten und der zentralen Steuerungseinheit erfolgt über industrielle Kommunikationsprotokolle. Ein gängiges Protokoll ist der CAN-Bus (Controller Area Network), der besonders für Anwendungen in Fahrzeugen geeignet ist. In industriellen Umgebungen kommen darüber hinaus Systeme wie Profibus und Profinet zum Einsatz, die eine robuste und zuverlässige Datenübertragung sicherstellen [35]. Diese Bussysteme bieten den Vorteil, dass sie für den Einsatz in rauen Umgebungen entwickelt wurden und eine hohe Störsicherheit bieten.

Durch die Einführung eines verteilten BMS, kombiniert mit modernen Kommunikationsprotokollen wie CAN-Bus oder Profibus, wird die Effizienz und Sicherheit von Batteriesystemen erheblich verbessert. Insbesondere in Anwendungen mit hohen Sicherheitsanforderungen, wie bei Elektrofahrzeugen oder der stationären Energiespeicherung, bieten verteilte BMS deutliche Vorteile [11, 35].

## 2.2. Antriebsbatterien

Antriebsbatterien spielen eine zentrale Rolle in der Elektromobilität und im Bereich der stationären Energiespeicherung. Sie dienen als Herzstück moderner Elektrofahrzeuge, indem sie die notwendige Energie für den Antrieb liefern und gleichzeitig eine zentrale Komponente für die Energiewende darstellen. Durch ihre Fähigkeit, große Energiemengen effizient zu speichern und bei Bedarf abzugeben, ermöglichen sie den Einsatz erneuerbarer Energiequellen wie Solar- oder Windkraft in mobilen und stationären Anwendungen.

### 2.2.1. Chemische Energiespeicher

In der modernen Gesellschaft existieren verschiedene Möglichkeiten, elektrische Energie chemisch zu speichern und wieder abzurufen. Dabei gibt es verschiedene Einsatzmöglichkeiten dieser Speicher, welche ihre Vor- und Nachteile berücksichtigen. Eine Übersicht gängiger chemischer Batterietypen ist im Folgenden aufgelistet.

- **Blei-Batterie (Pb):** Der Blei-Akkumulator ist einer der ältesten und am weitesten verbreiteten Energiespeicher. Er findet vor allem in Autos, USV-Systemen (unterbrechungsfreie Stromversorgung) und größeren stationären Anwendungen Einsatz [7].

## 2. Grundlagen

---

- **Nickel-Cadmium-Batterie (NiCd):** Nickel-Cadmium-Akkus wurden lange in tragbaren Geräten und in der Luftfahrt eingesetzt. Sie sind bekannt für ihre Robustheit und Beständigkeit, auch bei extremen Temperaturen [7].
- **Nickel-Metallhydrid-Batterie (NiMH):** Der NiMH-Akku ist eine Weiterentwicklung des NiCd-Akkus und wurde entwickelt, um die Umweltprobleme von Cadmium zu umgehen. Sie werden häufig in Hybridfahrzeugen und tragbaren Geräten eingesetzt [7].

**Lithium-Ionen-Batterie (Li-Ionen):** Lithium-Ionen-Akkus gehören zu den fortschrittlichsten Energiespeichern und sind die am weitesten verbreitete Technologie in mobilen Geräten, Elektrowerkzeugen und Elektrofahrzeugen [7].

Die Lithium-Ionen-Batterie ist die aktuell gängigste Batterie für die Elektromobilität und wird auch im Rahmen des ProMoBiS-Projekts in der Forschungsgruppe untersucht. Sie hebt sich vor allem durch ihre hohe Energiedichte und Anzahl von Ladezyklen von anderen chemischen Batteriespeichern ab. In der Abbildung 1 werden die Batterietypen in Hinblick auf ihre wichtigste Eigenschaft für den Einsatz als Antriebsbatterie dargestellt. Das Diagramm zeigt in Beziehung zu dem Gewicht anschaulich, dass vor allem höchste Reichweiten sowie sehr hohe Leistungen mit den verschiedensten Lithium Batterien umgesetzt werden können. [4]

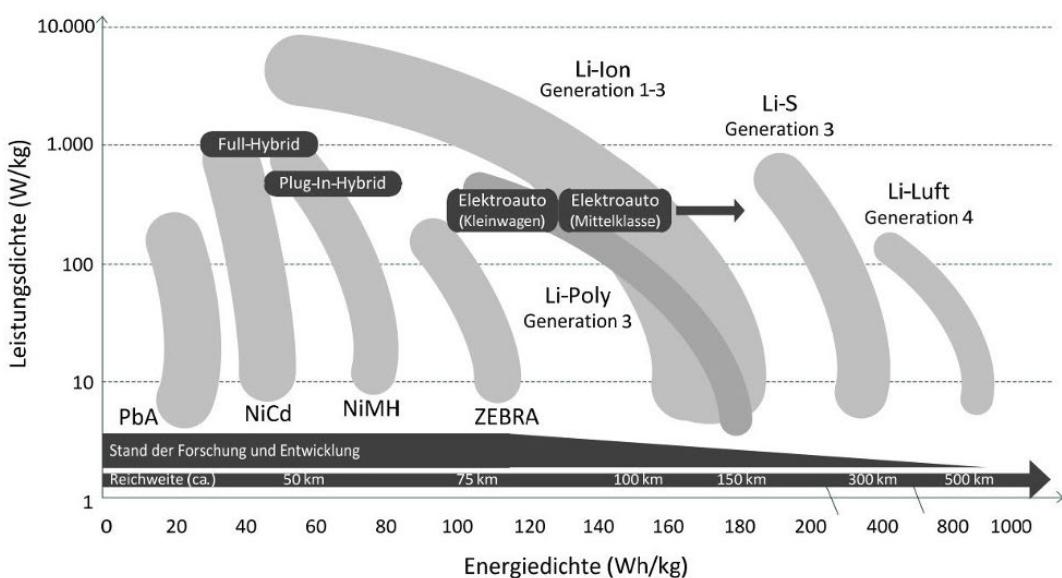


Abbildung 1: Übersicht Batterietechnologien im Ragone Diagramm (aus [4])

## *2. Grundlagen*

---

### **2.2.2. Traktionsbatterie**

Traktionsbatterien bestehen aus mehreren Modulen, die je nach benötigter Spannung entweder parallel oder in Serie geschaltet werden [4].

- **Serienschaltung:**

In der Serienschaltung addieren sich die Einzelspannungen der Zellen, während die Kapazität gleich bleibt. Es ist wichtig zu erwähnen, dass hierbei die Traktionsbatterie durch die Zelle mit der geringsten Leistungsfähigkeit begrenzt ist [4].

- **Parallelschaltung:**

In der Parallelschaltung bleibt die Spannung konstant, während die Kapazität steigt, was höhere Ströme ermöglicht, jedoch Ausgleichsströme verursachen kann [4].

Ein höherer Spannungswert des Batteriestacks reduziert die erforderlichen Ströme, was zu einer geringeren Wärmeentwicklung und einer effizienteren Energieübertragung führt. Dadurch können kleinere Leitungsquerschnitte verwendet werden, was wiederum Materialkosten spart und das Gewicht der Verkabelung verringert. Allerdings erfordert eine hohe Spannung auch den Einsatz spezieller Komponenten, wie Stecker, Isolierungen und Schutzvorrichtungen, um die Spannungsfestigkeit zu gewährleisten und Kurzschlüsse zu vermeiden. Hierbei entsteht ein Mehrkostenaufwand.

Aktuell entwickeln einige Hersteller sogenannte 1000V-Batterien (oft auch "Hochvoltbatterien" genannt), insbesondere für leistungsstarke Fahrzeuge wie Sportwagen. Diese hohe Spannung minimiert die Stromstärke, was nicht nur die Leitungsquerschnitte weiter reduziert, sondern auch die Effizienz des gesamten elektrischen Systems verbessert [4, 14].

### **2.3. Batteriemesstechnik**

Die Messtechnik in Traktionsbatterien spielt eine entscheidende Rolle, um den sicheren und effizienten Betrieb dieser Hochvolt-Batterien zu gewährleisten. Eine Vielzahl von Messgrößen wird kontinuierlich erfasst, um den aktuellen Zustand der Batterie sowie ihre Leistungsfähigkeit zu überwachen. Dabei kommen fortschrittliche Sensoren und Algorithmen zum Einsatz, um Daten in Echtzeit zu ermitteln und auszuwerten.

Die wichtigsten zu messenden Parameter sind:

## 2. Grundlagen

---

- **Lade- und Entladeraten:** Diese werden durch präzise Stromsensoren erfasst, die den Stromfluss in Echtzeit überwachen. Diese Messungen sind entscheidend für die Bestimmung des Ladezustands sowie die Kontrolle der Sicherheit und Leistungsfähigkeit der Batterie [33].
- **Temperatur:** Über mehrere Temperatursensoren, die an verschiedenen Stellen der Batterie installiert sind, wird die Temperaturverteilung überwacht. Dies ist besonders wichtig, da eine Überhitzung zu Schäden oder sogar zu einem thermischen Durchgehen der Zellen führen kann. Die Temperaturüberwachung ist daher ein wesentlicher Bestandteil des Batteriemanagementsystems [20].
- **Ladezustand (State of Charge):** Der Ladezustand der Batterie wird üblicherweise durch die Integration des Stromflusses (Coulomb-Zählung) und die Messung der Zellspannung ermittelt. Ergänzend dazu werden fortgeschrittene Methoden wie die elektrochemische Impedanzspektroskopie (EIS) eingesetzt, um genauere Ergebnisse zu erzielen. Diese Methoden ermöglichen eine präzisere Bestimmung des Ladezustands, insbesondere bei teilgeladenen oder alternden Batterien [33, 4].
- **Alterung:** Die Überwachung des Alterungsprozesses einer Batterie erfolgt durch komplexe Batteriemodelle, die auf den Messungen von Strom, Spannung und anderen Parametern basieren. Mithilfe von Methoden, wie der EIS können Alterungsprozesse frühzeitig erkannt und der Gesundheitszustand (State of Health) der Batterie bewertet werden [4, 20].

Diese Messtechniken sind entscheidend für die Überwachung und den Betrieb von Traktionsbatterien. Sie ermöglichen es, die Effizienz der Batterie zu optimieren, deren Lebensdauer zu verlängern und potenzielle Probleme frühzeitig zu erkennen. Die kontinuierliche Weiterentwicklung dieser Techniken wird zunehmend wichtiger, insbesondere mit Blick auf die wachsenden Anforderungen an Hochvoltbatterien in modernen Elektrofahrzeugen.

### 2.4. Elektrochemische Impedanzspektroskopie

Die elektrochemische Impedanzspektroskopie (EIS) ist eine etablierte Messmethode zur Untersuchung elektrochemischer Systeme. Sie wird in verschiedenen Fachbereichen wie der Medizintechnik, Biochemie und Materialwissenschaften eingesetzt [40, 10, 22]. In dieser Arbeit kommt die EIS im Kontext der Batteriesensorik zum Einsatz und kann verwendet werden, um Aussagen über den aktuellen Batteriezustand zu treffen.

### 2.4.1. Funktion

Die elektrochemische Impedanzspektroskopie (EIS) funktioniert durch die Anregung der Batteriezellen mit einem Wechselstromsignal, dessen Frequenz in einem weiten Spektrum variiert. Dabei wird die Spannungs- oder Stromantwort der Batterie gemessen, um die Impedanz des Systems für jede Frequenz zu berechnen. Da die Impedanz stark von der Frequenz abhängt, wird das Verfahren für eine Vielzahl von Frequenzen wiederholt, um ein vollständiges Impedanzspektrum zu erstellen. Diese Methode ist besonders in der Batterietechnologie wertvoll, da sie tiefere Einblicke in die zugrundeliegenden elektrochemischen Prozesse einer Batterie ermöglicht.

EIS wird in der Batterietechnologie verwendet, um wichtige Zustandsparameter zu bestimmen. Mithilfe von komplexen Batteriemodellen können unter anderem Parameter aus den Messungen abgeleitet werden [33]:

- **Temperatur:** Die Temperatur beeinflusst die elektrochemischen Prozesse und die Leistung der Batterie. Höhere Temperaturen können zu schnellerer Alterung führen.
- **Druck:** Bezieht sich auf den Innendruck in der Batterie, der durch Nebenreaktionen entstehen kann und auf Zellschäden hinweist.
- **Ladezustand (State of Charge):** Gibt den aktuellen Ladezustand der Batterie an.
- **Gesundheitszustand (State of Health):** Beschreibt den Zustand der Batterie im Vergleich zur ursprünglichen Kapazität.

Der Frequenzbereich, der für die EIS relevant ist, hängt von der spezifischen Anwendung ab. In der Batterietechnik liegt der übliche Bereich zwischen 1 mHz und 10 kHz [32]. Ein besonderes Merkmal der EIS ist die Voraussetzung, dass das zu untersuchende System ein lineares, zeitinvariantes Verhalten aufweisen muss. Batterien sind jedoch nicht vollständig linear, weshalb in der Praxis darauf geachtet wird, die Messungen möglichst kurzzuhalten, um Veränderungen im Batteriezustand während der Messung zu minimieren. Dadurch wird sichergestellt, dass eine ausreichende Linearität am aktuellen Arbeitspunkt erhalten bleibt [18].

### 2.4.2. Impedanzspektrum

Das Impedanzspektrum einer Batterie ändert sich in Abhängigkeit des Innenwiderstands der Zelle, dieser hängt an verschiedenen Faktoren ab, unter anderem SoC, SoH und der Temperatur [32]. Diese Veränderungen können genutzt werden, um den aktuellen Zustand der Batterie präzise zu überwachen und frühzeitig Anzeichen von Degradation zu erkennen.

## 2. Grundlagen

---

Impedanzspektren werden typischerweise als Bode- oder Nyquistdiagramme dargestellt. Abbildung 2 zeigt ein Nyquistdiagramm, das die Impedanzspektren einer Lithium-Ionen-Batterie bei verschiedenen Lastströmen darstellt.

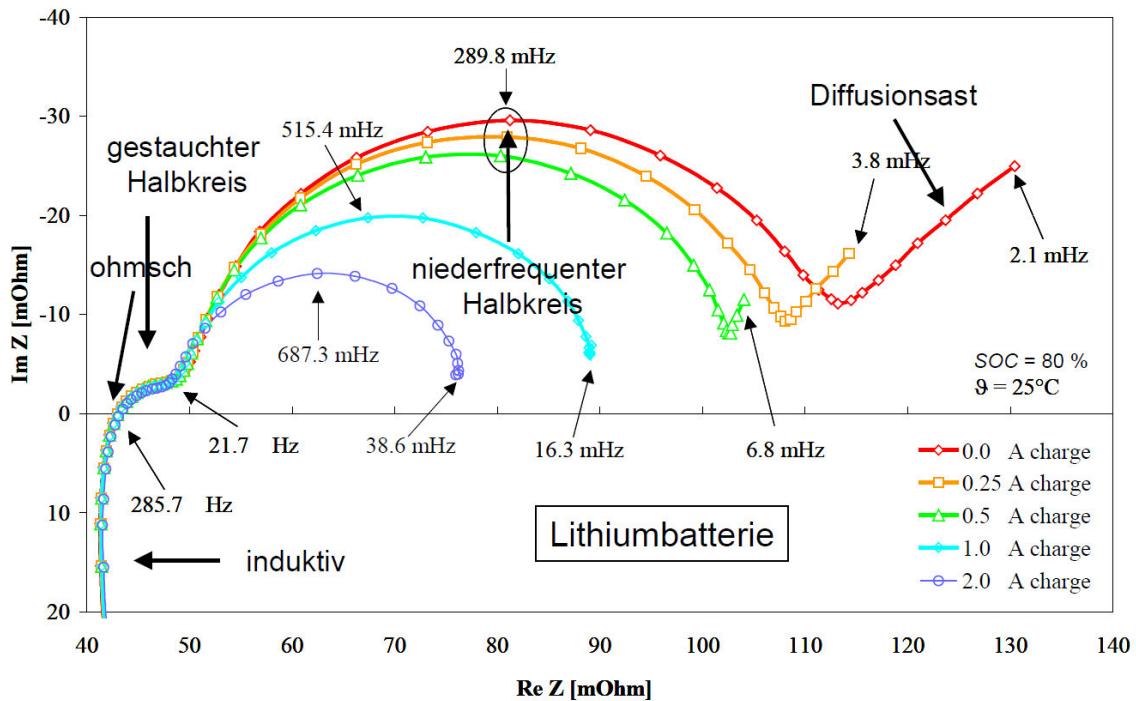


Abbildung 2: Impedanzspektren einer Lithiumbatterie bei verschiedenen überlagerten Ladeströmen (SoC = 80%, Temperatur 25°C, Nennkapazität 7 Ah) (aus [32])

In einem Nyquistdiagramm wird die Realachse für den realen Teil der Impedanz und die Imaginärachse für den imaginären Teil verwendet. Verschiedene Bereiche des Diagramms geben Aufschluss über unterschiedliche physikalische Phänomene [32, 30]:

- **Durchtrittspunkt (ohmsch):** Der Schnittpunkt des Diagramms mit der Realachse bei höheren Frequenzen entspricht dem ohmschen Widerstand der Batterie, der durch den Elektrolyten und die elektrischen Leitungen verursacht wird. Dieser Punkt ist ein interessanter Hinweis auf den Alterungszustand der Batteriezelle.
- **Gestauchter Halbkreis:** Dieser Bereich bei mittleren Frequenzen repräsentiert die Ladungstransferprozesse an den Elektrodenoberflächen und wird durch den Ladungstransferwiderstand und die Doppelschichtkapazität bestimmt.
- **Niederfrequenter Halbkreis:** Bei niedrigeren Frequenzen bildet sich ein weiterer Halbkreis, der Informationen über langsame elektrochemische Prozesse wie Diffusion enthält.

## 2. Grundlagen

---

- **Diffusionsast:** Im niederfrequenten Bereich zeigt sich ein linearer Anstieg, der auf die langsame Diffusion der Ionen im Elektrolyten zurückzuführen ist. Dieser Bereich wird oft als Warburg-Impedanz bezeichnet.
- **Induktiver Bereich:** In seltenen Fällen können sich im hochfrequenten Bereich induktive Effekte zeigen, die auf parasitäre Induktivitäten in den Leitungen oder elektrischen Verbindungen hinweisen.

Die Analyse dieser verschiedenen Bereiche des Impedanzspektrums liefert wertvolle Informationen über den aktuellen Zustand der Batterie, ihre Leistungsfähigkeit und mögliche Degradationserscheinungen.

## 2.5. Jitter

Der Begriff Jitter bezeichnet die Schwankungen im Timing eines elektrischen Signals und kann die Qualität und Zuverlässigkeit eines Systems erheblich beeinträchtigen. Jitter kann auf verschiedene Arten auftreten, abhängig vom Signal und der Anwendung. In Kommunikationssystemen oder bei Taktsignalen spielt Jitter eine entscheidende Rolle, da Verzögerungen oder Unregelmäßigkeiten die Genauigkeit von Datentransfers oder die Synchronisation stören können.

- **Zeitlicher Jitter:** Dieser Jitter-Typ bezieht sich auf Schwankungen im zeitlichen Abstand zwischen aufeinanderfolgenden Signalfanken. Besonders in Hochgeschwindigkeitssystemen, in denen Daten präzise zu festgelegten Zeitpunkten gesendet und empfangen werden müssen, können selbst kleine Schwankungen die Datenintegrität gefährden. Zeitlicher Jitter ist besonders problematisch bei asynchronen Taktsignalen, da er die Synchronisation von Schaltkreisen erschwert [38].
- **Cycle-to-cycle Jitter:** Dieser Jitter-Typ beschreibt Schwankungen im zeitlichen Abstand zwischen aufeinanderfolgenden Taktzyklen eines periodischen Signals. Das bedeutet, dass sich die Dauer eines einzelnen Taks unvorhersehbar ändert. Diese Schwankungen sind besonders kritisch in Systemen, die auf eine konstante Taktfrequenz angewiesen sind, wie Mikroprozessoren oder digitale Signalprozessoren, da sie die Signalverarbeitung beeinträchtigen können. Bei schnellen Analog-Digital-Umsetzern (ADCs) führt cycle-to-cycle Jitter zu Fehlern bei der Umwandlung von analogen in digitale Signale, was die Präzision reduziert [38].

Jitter entsteht durch eine Vielzahl von Ursachen, darunter thermisches Rauschen, elektromagnetische Interferenzen (EMI) oder systemische Unregelmäßigkeiten im Schaltkreis. Um die Auswirkungen von Jitter zu minimieren, werden in der Regel Techniken wie Phasenregelkreise (PLL)

## *2. Grundlagen*

---

oder Taktwiederherstellungsschaltungen eingesetzt, die dazu beitragen, die Präzision und Stabilität eines Systems zu gewährleisten [12].

### 3. Stand der Technik

In diesem Kapitel werden unter anderem die in dieser Arbeit verwendeten Kommunikationsvarianten sowie die dafür konkret verwendeten Bauteile vorgestellt. Des Weiteren wird eine Übersicht über die zur Kommunikation verwendeten Verfahren und Techniken vermittelt.

#### 3.1. Kommunikationsvarianten

Im Rahmen dieser Arbeit werden zwei Lösungsvarianten zur galvanisch getrennten Kommunikation in Betracht gezogen. Dabei handelt es sich bei den Kommunikationsvarianten einmal um eine induktiv gekoppelte Übertragung sowie eine optische Methode. In dieser Sektion werden die beiden Varianten vorgestellt und auf ihre technischen Eigenschaften eingegangen.

##### 3.1.1. Induktive Kommunikation

- **Allgemein:**

Die induktive Kopplung wird häufig zur drahtlosen Übertragung von Energie und Daten verwendet, insbesondere in Systemen, die galvanische Isolation erfordern. Die Technologie basiert auf elektromagnetischer Induktion, bei der zwei Spulen verwendet werden, um Signale durch ein magnetisches Feld zu übertragen, ohne eine physische Verbindung zwischen den Schaltkreisen zu haben. Der Kopplungsfaktor, der die Effizienz der Übertragung beeinflusst, hängt von der Geometrie der Spulen und ihrem Abstand zueinander ab. Diese Technik ist besonders für Nahfeldkommunikation (NFC) und drahtlose Energieübertragung in Anwendungen wie medizinischen Geräten oder isolierten Leistungselektroniksystemen geeignet [8, 27].

In der Praxis wird induktive Kommunikation oft in Digitalisolatoren verwendet, die hohe Datenraten unterstützen. Beispielsweise können moderne Systeme Datenraten von mehreren Mbit/s bei einer Reichweite von wenigen Zentimetern erreichen, wobei die Verzögerungszeiten minimal bleiben. Die Performance hängt dabei stark von der Qualität der verwendeten Spulen und der elektrischen Anpassung ab [8].

- **Bauteil:** (Infineon ISOFACE™ 2DIB141xF)

Das konkrete Bauteil, das in dieser Arbeit für die induktive Kommunikation verwendet wird, ist der *Digitalisolator ISOFACE™ 2DIB141xF* von Infineon. Dieses Bauteil nutzt die Coreless Transformer Technologie, um eine galvanische Trennung zwischen zwei Schaltkreisen

### 3. Stand der Technik

---

zu gewährleisten. Es bietet eine Übertragungsrate von bis zu 40 Mbit/s, was es für Anwendungen mit hohen Datenübertragungsanforderungen besonders geeignet macht. Ein weiteres wichtiges Merkmal dieses Digitalisolators ist die geringe Latenzzeit von etwa 26 ns, was eine schnelle Reaktion in zeitkritischen Anwendungen ermöglicht. Der ISOFACE Digitalisolator bietet außerdem eine flexible Konfiguration der Kanäle und Ausgangspegel, sodass er an unterschiedliche Einsatzszenarien angepasst werden kann [1].

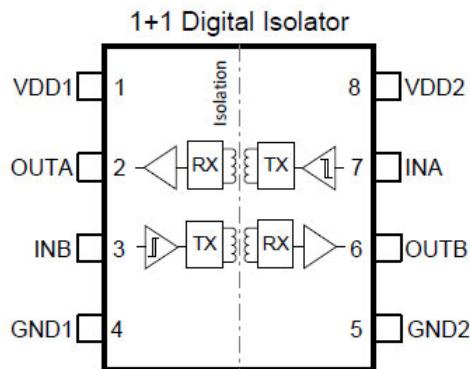


Abbildung 3: Pin-Konfiguration des ISOFACE™ digital isolators 2DIB141xF (aus [1])

#### 3.1.2. Optische Kommunikation im Infrarot-Bereich

- **Technologie:**

Die optische Kommunikationstechnologie nutzt Lichtwellen zur Datenübertragung, wobei Infrarotlicht besonders häufig verwendet wird. Sie bietet eine bidirektionale Punkt-zu-Punkt-Verbindung und ist in vielen Bereichen, darunter drahtlose Kurzstreckenkommunikation und mobile Anwendungen, verbreitet. Infrarot-Datenübertragungssysteme (wie die IrDA-Standards) nutzen typischerweise Wellenlängen im Bereich von 850 bis 900 nm und ermöglichen eine drahtlose Kommunikation über kurze Distanzen ohne die Notwendigkeit einer physischen Verbindung. Durch die Verwendung von Licht als Übertragungsmedium wird ein hohes Maß an Datensicherheit erreicht, da die Signale schwer von außen abzufangen sind [5]. Typische Anwendungen reichen von Fernbedienungen bis zu Datenübertragung in medizinischen Geräten und Kommunikationsmodulen in mobilen Endgeräten.

Die optische Kommunikation im Infrarot-Bereich kann Datenraten von bis zu 1 Gbit/s erreichen, abhängig von den verwendeten Modulationstechniken und Standards. Infrarotkommunikationsprotokolle wie der **IrDA-Standard!** (**IrDA-Standard!**) spezifizieren dabei verschiedene Modulationstechniken und Datenraten, die in unterschiedliche Kategorien

### 3. Stand der Technik

---

unterteilt sind (Tabelle 1), je nach den Anforderungen an die Übertragungsrate und die Energieeffizienz [15].

- **Infrarot-Transceiver:** (TFDU4101 von Vishay)

Für diese Arbeit wird der *Infrarot-Transceiver TFDU4101* von Vishay (Abbildung 4) verwendet. Dieses Bauteil ist für den SIR-Bereich (Standard IrDA) ausgelegt und unterstützt Baudraten von 9.6 kbit/s bis zu 115.2 kbit/s. Die Pulslänge für ein HIGH-Signal beträgt in diesem Bereich nur 3/16 der Bitzeit, um den Anforderungen der Infrarot-LED gerecht zu werden. Dies hilft, Überhitzung zu vermeiden und den Stromverbrauch zu minimieren. Ein weiteres Merkmal ist die umgekehrte Signalübertragung, bei der das durch die Infrarot-LED gesendete Signal beim Empfänger invertiert wird, was auf der verwendeten Return to Zero Inverted (RZI)-Modulation (Return-to-Zero Inverted) basiert. Diese Modulationstechnik stellt sicher, dass das Signal korrekt erfasst wird und eine zuverlässige Synchronisation zwischen Sender und Empfänger gewährleistet ist. Die Photodiode im Empfänger spielt hierbei eine wichtige Rolle in der präzisen Detektion des übertragenen Signals [36].



Abbildung 4: Infrarot Transceiver Modul TFDU4101 (aus [36])

### 3. Stand der Technik

---

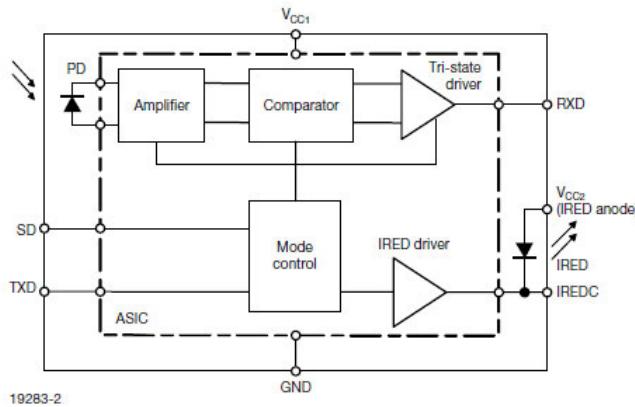


Abbildung 5: Pin-Konfiguration des Infrarot Transceiver Moduls TFDU4101 (aus [36])

Standard	Datenrate	Modulation
Serial Infrared (SIR)	9,6 kbit/s bis 115,2 kbit/s	Return to Zero Inverted (RZI) mit 3/16 Puls
Medium Infrared (MIR)	0,576 Mbit/s bis 1,152 Mbit/s	Return to Zero Inverted (RZI) mit 1/4 Puls
Fast Infrared (FIR)	4 Mbit/s	4 Pulse Position Modulation (4PPM)
Very Fast Infrared (VFIR)	16 Mbit/s	Non-Return to Zero Inverted (NRZI)
Ultra Fast Infrared (UFIR)	96 Mbit/s	Non-Return to Zero (NRZ)
Giga Infrared (Giga-IR)	512 Mbit/s bis 1 Gbit/s	Non-Return to Zero (NRZ)

Tabelle 1: Übersicht der Datenraten und Modulationstechniken der IrDA-Standards [15]

## 3.2. Kommunikationstopologien

Um die Kommunikation zwischen Mikrocontrollern je nach Anwendungsfall effizient umzusetzen, existieren verschiedene Kommunikationsstrategien und Topologien. Diese Topologien bestimmen die physikalische Anordnung und die Datenübertragung zwischen den beteiligten Geräten. Jede Topologie bringt spezifische Vor- und Nachteile hinsichtlich der Skalierbarkeit, Fehler-toleranz und Kosten mit sich. In den folgenden Abschnitten werden die wichtigsten Topologien in Bezug auf deren Einsatzgebiete, übliche Anzahl an Teilnehmern und Kommunikationsdistanzen beschrieben.

### *3. Stand der Technik*

---

- **Bus-Topologie:**

In einer Bus-Topologie sind alle Geräte über eine gemeinsame Kommunikationsleitung (Bus) verbunden. Dies bedeutet, dass alle Teilnehmer gleichzeitig auf dieselbe Leitung zugreifen, wobei ein Kommunikationsprotokoll den Zugriff regelt (z.B. Master-Slave). Bus-Topologien sind einfach und kostengünstig zu implementieren und eignen sich gut für kurze Distanzen und eine begrenzte Anzahl von Teilnehmern, wie beispielsweise in UART-Anwendungen. Ein typischer Anwendungsfall ist die RS-485-Schnittstelle in industriellen Bussystemen. Nachteilig ist die geringe Fehlertoleranz, da ein Ausfall des Busses die gesamte Kommunikation stören kann [35].

- **Ring-Topologie:**

In der Ring-Topologie ist jedes Gerät mit genau zwei anderen verbunden, sodass ein geschlossener Ring entsteht. Die Daten werden in eine Richtung weitergeleitet, bis sie das Ziel erreichen. Diese Topologie eignet sich gut für Anwendungen, bei denen eine gleichmäßige Lastverteilung gewünscht ist. Sie wird oft in industriellen Netzwerken und bei Token-Ring-Netzwerken verwendet. Vorteile sind eine konstante Datenrate und deterministische Zugriffszeiten. Ein Nachteil ist, dass ein Ausfall eines einzelnen Geräts die gesamte Kommunikation beeinträchtigen kann, es sei denn, es werden redundante Ringe eingesetzt [35].

- **Stern-Topologie:**

Bei der Stern-Topologie sind alle Geräte direkt mit einem zentralen Knotenpunkt (z.B. Hub oder Switch) verbunden. Diese Topologie wird häufig in Ethernet-Netzwerken oder bei seriellen Kommunikationssystemen wie UART verwendet. Stern-Topologien bieten hohe Fehlertoleranz, da der Ausfall eines Geräts oder einer Verbindung den Rest des Netzwerks nicht beeinträchtigt. Die zentrale Einheit kann allerdings zum Engpass oder Single-Point-of-Failure werden. Diese Topologie ist ideal für kurze Distanzen, wie sie typischerweise in lokalen Netzwerken (LAN) vorkommen [35].

- **Baum-Topologie:**

Eine Baum-Topologie stellt eine hierarchische Verbindung von Geräten dar, bei der mehrere Stern-Topologien miteinander kombiniert werden. Der Aufbau ähnelt der Struktur eines Baumes, wobei Knoten auf verschiedenen Hierarchieebenen miteinander verbunden sind. Diese Topologie wird oft in großen Netzwerken oder bei hierarchisch organisierten Systemen wie CAN-Bus verwendet. Sie ermöglicht es, viele Geräte auf einer strukturierten und skalierbaren Weise zu verbinden. Der Nachteil liegt in der erhöhten Komplexität und der potenziellen Belastung des zentralen Knotenpunktes [35].

### 3.3. Buszugriffsverfahren

Buszugriffsverfahren sind entscheidend für die Organisation der Kommunikation in Netzwerken und definieren, wie und wann Teilnehmer auf das gemeinsame Kommunikationsmedium zugreifen können. Sie lassen sich grob in deterministische und nicht-deterministische Verfahren unterteilen, sowie nach der Art der Buszuteilung in zentrale und dezentrale Verfahren.

#### 3.3.1. Kontrollierte Buszugriffsverfahren

Bei kontrollierten, auch deterministischen Verfahren genannt, ist der Sender vor Beginn der Übertragung klar definiert. Das ermöglicht es, die maximale Zeit bis zur Datenübertragung (Zykluszeit) vorherzusagen, was für Echtzeitsysteme essenziell ist. Diese Verfahren sind somit echtzeitfähig. Eine Unterteilung erfolgt in zentrale und dezentrale Buszuteilung. [35].

- **Master/Slave-Verfahren:** (zentrale Buszuteilung)

Beim Master/Slave-Verfahren gibt es einen zentralen Controller, den Master, der steuert, welches der Slave-Geräte auf den Bus zugreifen darf. Dieses Verfahren ist deterministisch und echtzeitfähig, da der Master zu jedem Zeitpunkt den Zugriff kontrolliert und den Kommunikationszyklus berechenbar macht. Beispiele sind der I<sup>2</sup>C-Bus oder Modbus [35].

- **Zentrale Buszuteilung:** Der Master entscheidet zentral, welches Gerät senden darf, was die Kommunikation kontrolliert und effizient macht.

- **Token-Prinzip (Tokenbus und Tokenring):** (dezentrale Buszuteilung)

Beim Tokenbus und Tokenring wird ein Token im Netzwerk zirkuliert. Nur das Gerät, das das Token hält, darf Daten senden. Da jedes Gerät irgendwann das Token erhält, ist auch hier die maximale Wartezeit vorhersagbar, was die Echtzeitfähigkeit sicherstellt [35].

- **Dezentrale Buszuteilung:** Es gibt keinen zentralen Master. Stattdessen wird das Token zwischen den Teilnehmern weitergegeben, was die Kontrolle über den Bus verteilt und Ausfallsicherheit erhöht [35].

- \* **Tokenbus:** Verwendet z.B. im Profibus, besonders geeignet für längere Entfernung und größere Netzwerke [35].

- \* **Tokenring:** Hier sind alle Geräte in einem Ring verbunden, und das Token wird von Gerät zu Gerät weitergereicht [35].

### 3. Stand der Technik

---

- **Busarbitration:** (zentrale Buszuteilung)

Beim Busarbitrationsverfahren wird der Buszugriff von einem sogenannten "Link Active Scheduler (LAS)" koordiniert, der in zyklischen Abständen das zu übertragende Objekt abfragt (z.B. "Compel Data"). Jeder Teilnehmer, der dieses Objekt erzeugt (Publisher), antwortet auf diese Abfrage. Andere Teilnehmer (Subscriber) können das übertragene Objekt empfangen und direkt verarbeiten. Der Vorteil des Busarbitrationsverfahrens liegt in der schnellen und synchronisierten Datenübertragung, da alle Teilnehmer gleichzeitig auf die Information zugreifen können. Dieses Verfahren ist deterministisch und ermöglicht Echtzeitkommunikation [35].

- **Zentrale Buszuteilung:** Der LAS übernimmt die zentrale Steuerung des Buszugriffs und synchronisiert die Teilnehmer, was eine zeitgenaue und planbare Kommunikation ermöglicht. Beispiele hierfür finden sich bei "FOUNDATION Fieldbus" oder "WorldFIP" [35].

#### 3.3.2. Zufällige Buszugriffsverfahren

Im Gegensatz dazu sind zufällige Buszugriffsverfahren nicht-deterministisch, da sie keinen festen Kommunikationszyklus besitzen. Die Teilnehmer greifen nur bei Bedarf auf den Bus zu, was zu Kollisionen führen kann, die den Kommunikationsprozess unvorhersehbar machen. Daher sind diese Verfahren in der Regel nicht echtzeitfähig.

- **CSMA (Carrier Sense Multiple Access):**

Beim "Carrier Sense Multiple Access (CSMA)" prüfen die Geräte vor dem Senden, ob der Bus frei ist (Carrier Sense). Ist der Bus belegt, warten sie, bis dieser frei wird. Da keine zentralisierte Kontrolle über die Zugriffsreihenfolge besteht, kann es zu Kollisionen kommen, wenn mehrere Geräte gleichzeitig senden wollen, was die Übertragungszeit unvorhersehbar macht. Dies macht das Verfahren nicht deterministisch und nicht echtzeitfähig [35].

- **CSMA/CD (Collision Detection):** Dieses Verfahren wird beispielsweise im klassischen Ethernet eingesetzt. Wenn es zu einer Kollision kommt, erkennen die Geräte dies und stoppen die Übertragung, um es nach einer zufälligen Zeit erneut zu versuchen [35].
- **CSMA/CA (Collision Avoidance):** Bei CSMA/CA versuchen die Geräte Kollisionen zu vermeiden, indem sie eine zusätzliche Wartezeit nach dem Freiwerden des BusSES einbauen. Dieses Verfahren wird beispielsweise in WLAN-Netzwerken verwendet, wo es wichtig ist, Kollisionen aufgrund der drahtlosen Übertragung zu vermeiden [35].

### *3. Stand der Technik*

---

- **Busarbitration bei zufälligem Zugriff:**

In einigen Systemen wird eine Mischung aus zufälligem Zugriff und Arbitration verwendet, um Kollisionen zu vermeiden. In diesen Fällen überwacht eine zentrale oder dezentrale Einheit den Bus und entscheidet dynamisch, welches Gerät senden darf. Dies kann die Übertragungszeit stabilisieren, führt aber nicht zur vollen Echtzeitfähigkeit [35].

## 4. Voruntersuchungen

In diesem Kapitel werden die beiden Methoden zur Kommunikation mittels eines einfachen Testaufbaus genauer auf ihre Eigenschaften untersucht. Das Ziel dieser Untersuchungen ist es, die Rahmenbedingungen für einen Anwendungsfall optimal einschätzen zu können und abschließend die Vor- und Nachteile der Bauteile gegeneinander abzuwägen.

### 4.1. Versuchsaufbau

Für das spätere ProMoBiS-Projekt sollen Mikrocontroller des Typs XMC1404 von Infineon verwendet werden. Da diese zu dem aktuellen Zeitpunkt der Versuchsdurchführung nicht zur Verfügung stehen, werden für den Versuchsaufbau die Mikrocontroller der Typenreihe XMC1100 von Infineon eingesetzt, um die entsprechenden Bauteile zu testen. Das hat den Vorteil, dass die Softwareimplementierung sehr ähnlich zu dem präferierten Mikrocontroller des Versuchsaufbaus ist.

Die dabei relevanten Eigenschaften, die es zu untersuchen gilt, sind die maximale Baudrate, der Bauteil-Jitter und die maximale Verzögerungszeit bei der Übertragung. Des Weiteren sind die Auswirkungen der Strombegrenzung der Sendediode des Infrarot-Transceivers für die optische Kommunikationsvariante von Interesse. Eine kleine Schaltung mit einem Mikrocontroller zum Senden und einem zum Empfangen von UART-Daten sowie den entsprechenden Kommunikationsbauteilen ist für diese Aufgabe ausreichend.

Da es sich schaltungstechnisch hierbei um einen simplen Aufbau handelt, der eine erste Untersuchung der Bauteile ermöglichen soll, wird die Hardwareimplementierung als Lochrasterpatienten-Aufbau umgesetzt. Zusammengefasst sind die hierbei benötigten Bauteile zwei Mikrocontroller der Typenreihe XMC1100, zwei Infrarot-Transceiver des Typs tfdu4101, ein Digitalisolator vom Typ ISOFACE™ 2DIBx41xF sowie ein Potentiometer zur Strombegrenzung der Sendediode (Abbildung 7). In Abbildung 6 ist der Schaltplan für den Messversuch dargestellt.

#### 4. Voruntersuchungen

---

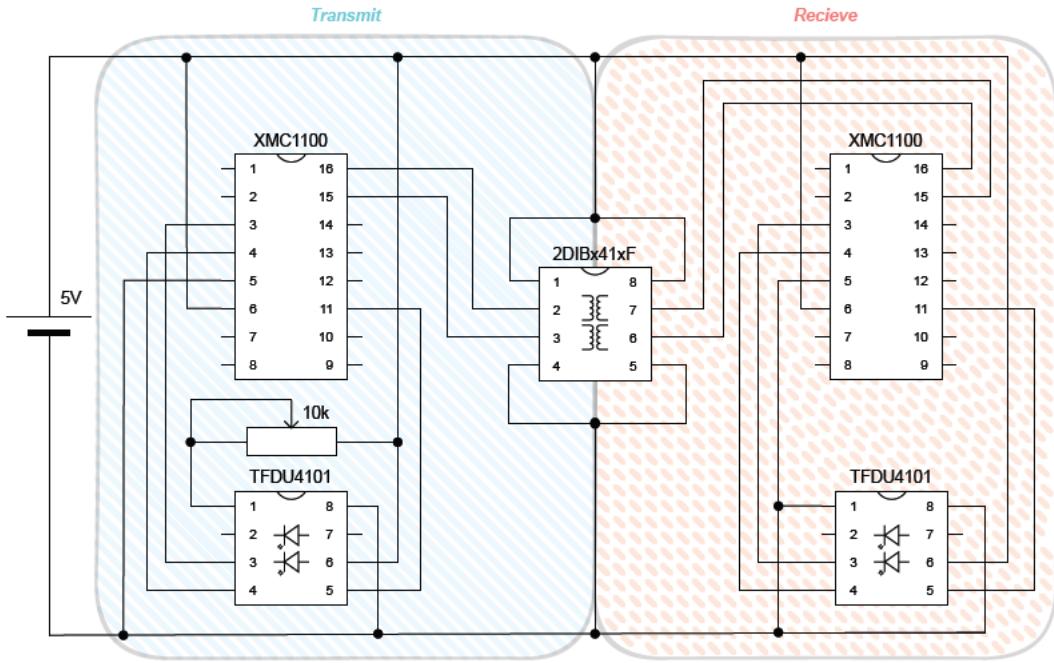


Abbildung 6: Schaltplan des Messaufbaus zur Untersuchung der Kommunikationsvarianten

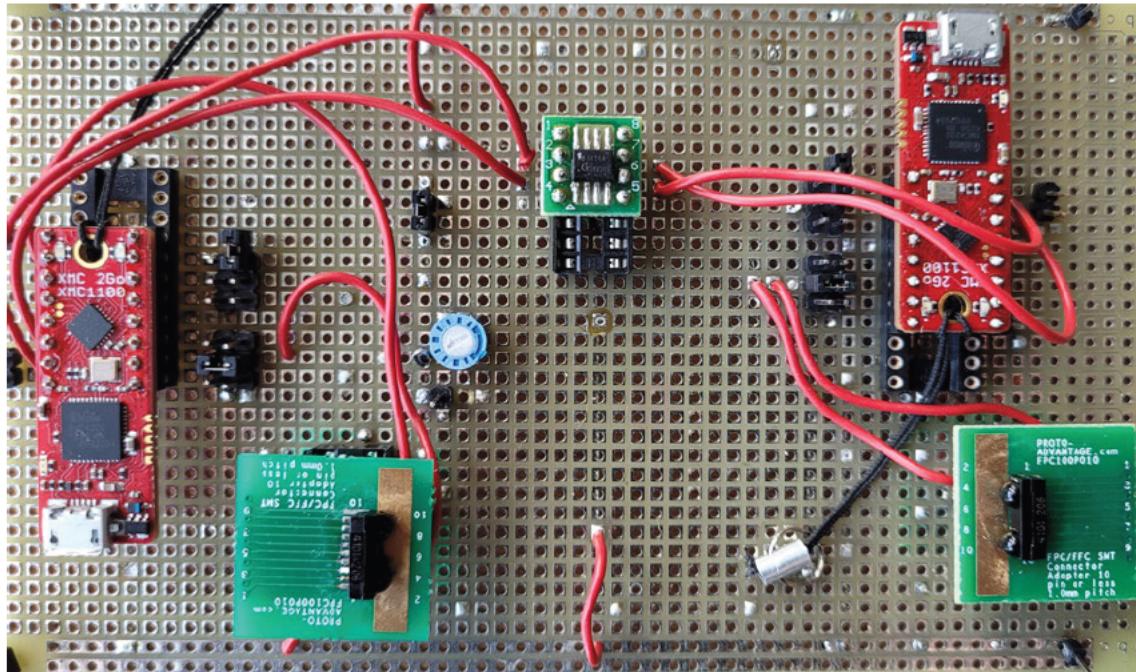


Abbildung 7: Hardwareaufbau zur Untersuchung der Kommunikationsvarianten

## 4.2. Softwareimplementierung

Die Implementierung der Software erfolgte mithilfe der integrierten Entwicklungsumgebung (IDE) *DAVE Version 4.5.0* von Infineon. Die Wahl dieser Entwicklungsumgebung kristallisierte sich heraus, da DAVE einen speziell auf Infineon-Mikrocontroller zugeschnittenes Code-Generator bereitstellt. Die Software bietet eine umfangreiche Bibliothek von Apps, die es ermöglichen, verschiedenste Hardwarekonfigurationen über eine benutzerfreundliche Oberfläche zu definieren. So kann beispielsweise die Konfiguration der UART-Schnittstelle über eine GUI eingestellt werden, woraufhin der notwendige Code automatisch generiert wird, um die gewählten Funktionen zu unterstützen.

Für die Funktionsprüfung der verwendeten Bauteile ist es erforderlich, spezifische Softwareanpassungen vorzunehmen. Im Folgenden werden die wichtigsten Implementierungen beschrieben:

- **UART-Anpassungen:**

Da für beide wesentlichen Bauteile des Systems, den Infrarot-Transceiver und den Digitalisolator, eine UART-Kommunikation notwendig ist, mussten die UART-Konfigurationen jeweils spezifisch konfiguriert werden.

- **Infrarot-Transceiver:**

Für den Infrarot-Transceiver erfolgte die Anpassung der UART-Kommunikation nach dem IrDA-Standard. Dieser Standard erfordert eine Pulslänge von 3/16 sowie einen entsprechenden Abtastzeitpunkt, welcher sich in etwa in der Mitte der Bitdauer befindet, in diesem Fall bei 2/16. Diese spezifische Konfiguration ist allerdings nicht direkt über die UART-App-Maske der DAVE App einstellbar. Aus dem Grund wurde der Code für die Anpassung aus einem der bestehenden Projekte übernommen, wodurch die Anpassung des UART für den Infrarot-Transceiver effizient durchgeführt werden konnte (siehe Anhang A.2.2). Des Weiteren ist eine Invertierung des UART-Ausgangssignals beim Senden notwendig, um den Anforderungen des Infrarot-Transceivers gerecht zu werden.

- **Digitalisolator:**

Für den Digitalisolator *2DIB1410F* (ISOFACE) ist eine Anpassung der UART-Schnittstelle erforderlich. Da das Bauteil invertierte Signale verwendet, musste sowohl der UART-Ausgang beim Senden als auch der UART-Eingang beim Empfangen invertiert werden. Die Pin-Konfiguration erfolgte entsprechend der Schaltungsvorgabe, wie sie in Abbildung 3 dargestellt ist.

#### 4. Voruntersuchungen

---

- **Periodisches Senden von Daten:**

Zum Testen der Implementierung wurde ein periodisches Senden von Daten implementiert. Dies ist durch die Verwendung von Systick-Interrupts realisiert, welche eine zeitgesteuerte Übertragung ermöglichen. Die periodische Datenübertragung ist insbesondere für die Testzwecke in den noch folgenden Untersuchungen des Jitters von Bedeutung. Explizit ist es dadurch möglich, bei der Messung mit dem Oszilloskop die durch den Timer eingestellte Sendefrequenz der UART-Frequenz zu filtern und zielführende Messungen durchzuführen.

- **GPIO-Steuerung:**

Zusätzlich zu den UART-Anpassungen wurden GPIO-Pins konfiguriert, um verschiedene Funktionen des Systems zu testen. Hierzu zählen die Steuerung eines Trigger-Pins sowie die Ansteuerung von LEDs, die optional zur visuellen Rückmeldung während der Tests verwendet wurden. Außerdem wird ein GPIO-Pin zur Ansteuerung eines Infrarot-Transceivers genutzt.

- **Auswahlmodi:**

Da bei den beiden verwendeten Mikrocontroller die zur Übertragung dienenden Bauteile an den jeweils selben Pins der Mikrocontroller angeschlossen sind, ist es möglich denselben Softwarecode mit kleinen Anpassungen für beide Controller zu verwenden. Ob die Software in Sende- oder Empfangsmodul arbeitet, wird über *define*-Präprozessordirektiven bestimmt. Folglich kann die Auswahl der gesendeten UART-Daten zur Übertragung des genutzten Bauteils ausgewählt werden. (Siehe Anhang A.1.1)

### 4.3. Baudratenbestimmung

Das Ziel der Baudratenbestimmung ist, einen Phasenfehler bei der Durchführung der EIS möglichst zu vermeiden. Hierbei ist die maximale Übertragungsrate ein ausschlaggebender Parameter in der Messkette. Die maximale Übertragungsrate limitiert in einem dezentralen Sensorsystem auch die maximale Abtastfrequenz.

Die verwendeten Bauteile unterscheiden sich grundlegend in ihrer maximalen Übertragungsgeschwindigkeit. Während die IrDA-Infrarot-Transceiver laut Datenblatt Baudraten von bis zu 115.2 kbit/s unterstützen [36], ermöglicht der Digitalisolator ISOFACE Baudraten von bis zu 40 Mbit/s [1].

Die maximale Baudrate der UART-Schnittstelle, die mit den XMC1100 2GO Mikrocontrollern eingestellt werden kann, lässt sich mit der folgenden Formel berechnen:

#### 4. Voruntersuchungen

---

$$Baudrate_{max,\mu C} = \frac{PCLK}{Divisor \cdot Oversampling} \quad (4.1)$$

Bei einem minimalen Divisor von zwei, einem minimalen Oversampling von vier und dem Peripherie Clock (Peripheral Clock) (PCLK) des Mikrocontrollers ergibt sich eine maximal mögliche UART-Geschwindigkeit von etwa 8 Mbit/s [2].

##### 4.3.1. Baudratenbestimmung des Digitalisolators

Für den Digitalisolator ist eine Geschwindigkeit von 8 Mbit/s mit der entsprechenden Hardware erreichbar. Derzeit erreichen die Batterien der Elektroautos eine maximale Länge von schätzungsweise 3 m. Aufgrund der stetigen Weiterentwicklung ist eine Erhöhung der Batteriemaße durch größere Automobile ist allerdings nicht ausgeschlossen. Da es bei hohen Baudaten zu unerwünschten leistungstheoretischen Phänomenen kommen kann, ist eine Bewertung der physikalischen Grenze angebracht.

Eine Abschätzung dessen lässt sich mittels einer in der Nachrichtentechnik üblichen Faustformel berechnen, nach der die Wellennatur elektrischer Vorgänge bei Schaltungen vernachlässigt werden darf, deren Größe 1/10 der Wellenlänge nicht überschreitet [23].

Mit dieser Erkenntnis und einer angenommenen typischen Ausbreitungsgeschwindigkeit von elektrischen Impulsen von etwa 2/3 der Lichtgeschwindigkeit ( $c_0$ ) sowie einer maximalen Leitungslänge ( $l_{max}$ ) von 3 m ergibt sich die Formel 4.2 zur Abschätzung der maximal zulässigen Baudate:

$$Baudrate_{max} = \frac{2/3 \cdot c_0}{10 \cdot l_{max}} \quad (4.2)$$

Daraus ergibt sich eine maximale Obergrenze von etwa 6.66 Mbit/s. Da die Stabilität des Systems jedoch stets gewährleistet sein muss, ist es nicht ratsam, diese theoretische Maximalgeschwindigkeit vollständig auszunutzen. Um sicherzustellen, dass die Schaltung einwandfrei funktioniert, wird daher etwa die Hälfte dieser Geschwindigkeit, also 3 Mbit/s, als praktikable und schnellstmögliche Übertragungsrate bevorzugt.

Es ist möglich, durch leitungstechnische Anpassungen höhere Datenraten zu erreichen. Der damit verbundene zusätzliche Arbeitsaufwand und die Berechnung der Kosten sind nicht Gegenstand dieser Arbeit und werden daher nicht weiter angeführt.

### 4.3.2. Baudratenbestimmung des Infrarot-Trancievers

Der optische Infrarot-Transceiver ist wie zuvor erwähnt mit einer maximalen Datenrate von 115.2 kbit/s im Datenblatt angegeben. Aus einer vorherigen Bachelorarbeit gibt es jedoch Hinweise auf einen stabilen, übertakteten Betrieb des Bausteins [9]. Eine erhöhte Datenrate bei dem Infrarot Transceiver kann im Gegensatz zu dem verdrahtetem Optokoppler einen zentralen Unterschied in der Geschwindigkeit des Systems machen. Daher wird die maximale Datenrate im Folgenden näher ermittelt.

Vorab ist es angebracht, die Strombegrenzung der Sendediode des Infrarot-Transceivers zu berücksichtigen. Da es sich um einen Versuchsaufbau zur Untersuchung des Bauteils für den realen Einsatz handelt, wird die Leistung der Diode begrenzt. Dies ist besonders wichtig, da sich die Bauenteile im Versuchsaufbau unmittelbar gegenüberstehen, was im realen Einsatz aufgrund der variierenden Streckenführung nicht zwangsläufig der Fall sein wird. Aus diesem Grund wird zunächst ein Vorwiderstandswert von  $1\text{ k}\Omega$  eingestellt. Eine genauere Betrachtung der Auswirkungen des Vorwiderstands erfolgt in Abschnitt 4.5.

Zunächst wird die Datenrate in gleichmäßigen Schritten erhöht, um dem Bereich näherzukommen, in dem die Technologie ihre Funktionsfähigkeit vollständig verliert. Die gewählte Schrittweite für das weitere Vorgehen beträgt 12.800 kbit/s. Diese Schrittweite wurde gewählt, um eine angemessene Anzahl von Messungen durchzuführen und eine Einschätzung des Verhaltens bei allmählicher Erhöhung der Datenrate zu ermöglichen.

Die Messungen werden mithilfe eines Oszilloskops durchgeführt. Dabei wird die Persistenzfunktion des verwendeten Geräts aktiviert. Diese Funktion ermöglicht es, mehrere erfasste Wellenformen gleichzeitig auf dem Display des Oszilloskops anzuzeigen. Durch die Übertragung von PRBS-Daten über die Schnittstelle ist es möglich, jedes auftretende Bit der UART-Übertragung darzustellen. Für die Analyse der maximalen Übertragungsrate ist, zu beobachten, ab wann Symbole vollständig ausfallen oder überlappen. Zur Veranschaulichung wird in Abbildung 8 die PRBS-Datenübertragung bei der maximalen Datenblattangabe gezeigt.

#### 4. Voruntersuchungen

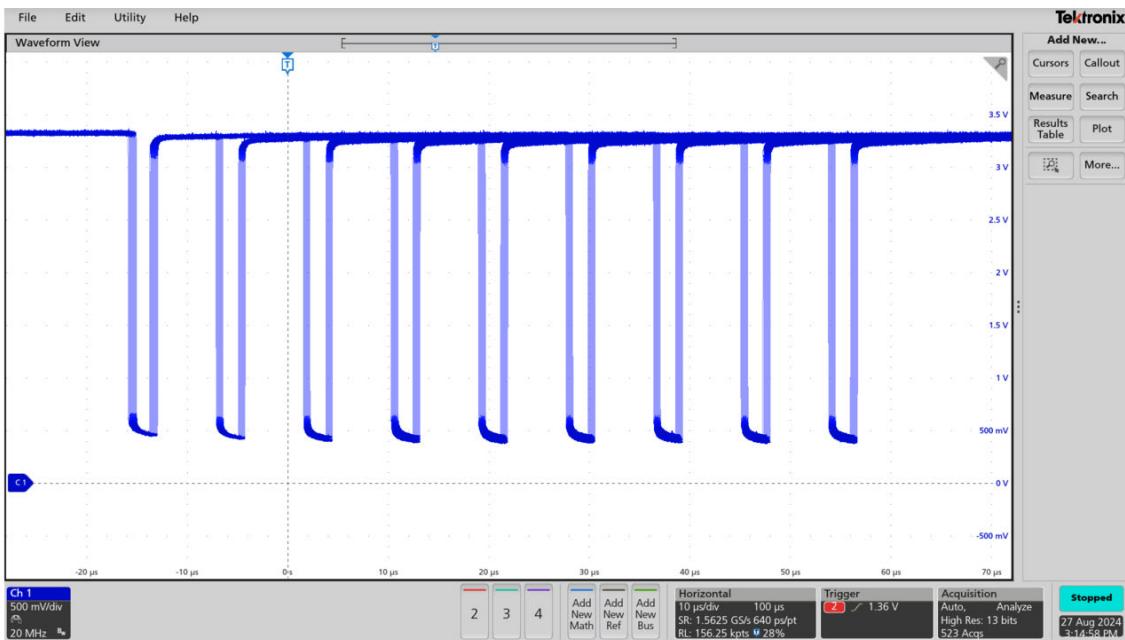


Abbildung 8: PRBS-Daten-Übertragung der UART-Schnittstelle des XMC1100 mittels Infrarot-Transceiver bei einer Baudrate von 115.2 kbit/s

Die durchgeföhrten Messungen zeigen ein funktionierendes System bis zu einer Datenrate von 204.800 bit/s (Abbildung 9). Ab einer Datenrate von 217.600 bit/s treten bei der Datenübertragung eindeutig Fehler auf, bei denen das erste Datenbit nach dem Startbit immer fehlt (Abbildung 10).

#### 4. Voruntersuchungen

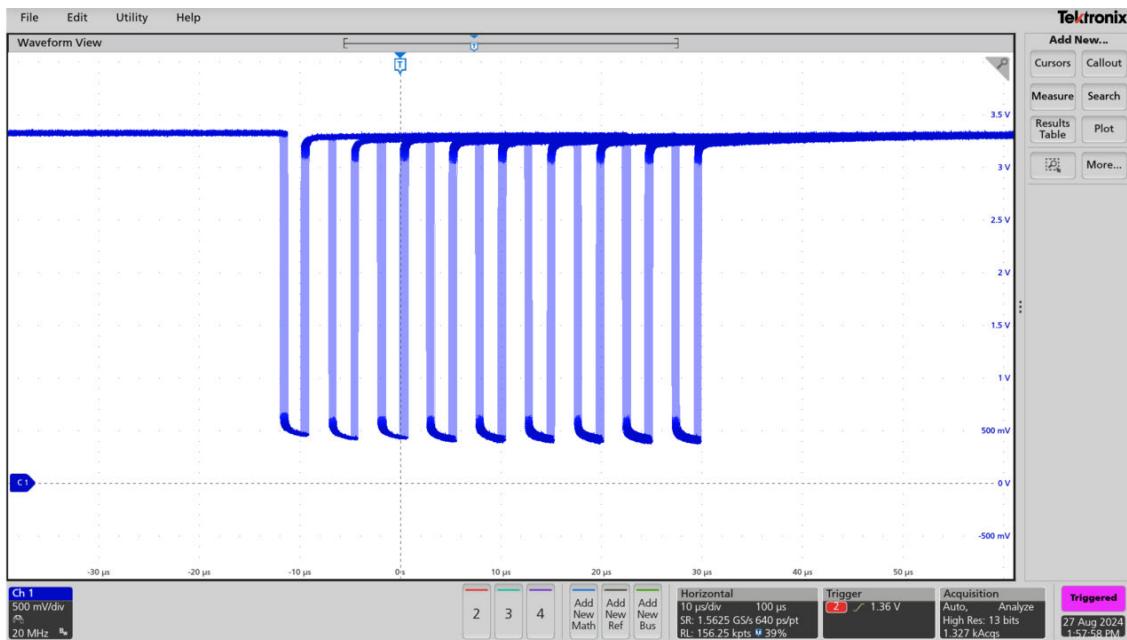


Abbildung 9: PRBS-Daten-Übertragung der UART-Schnittstelle des XMC1100 mittels Infrarot-Transceiver bei einer Baudrate von 204.800 bit/s

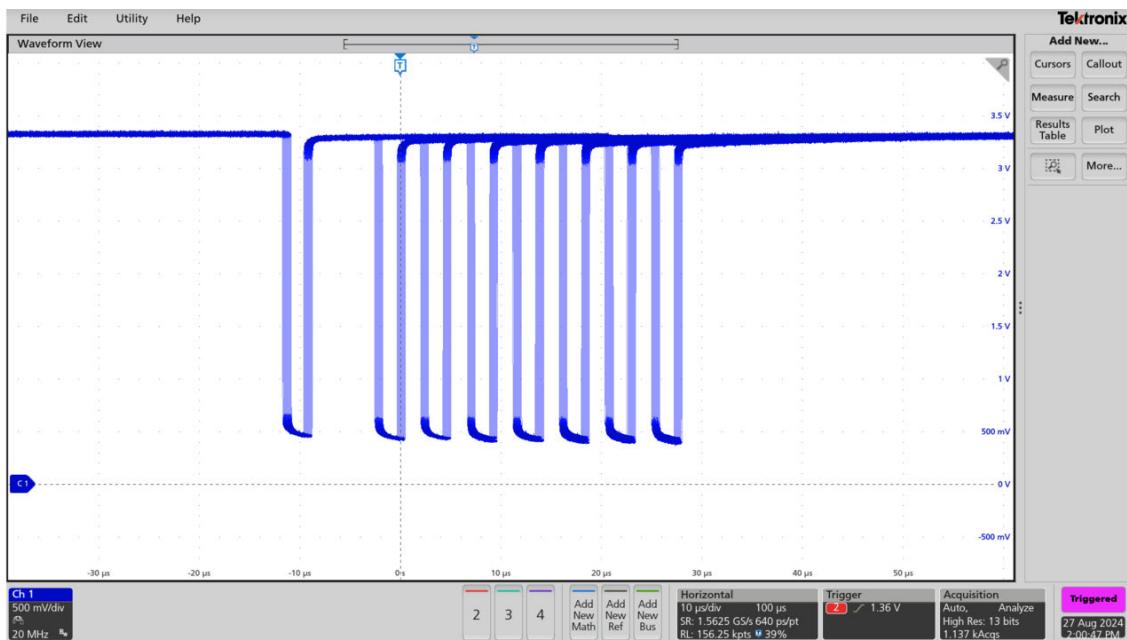


Abbildung 10: PRBS-Daten-Übertragung der UART-Schnittstelle des XMC1100 mittels Infrarot-Transceiver bei einer Baudrate von 217.600 bit/s

Der genaue Wert der Grenze liegt wahrscheinlich bei etwa 210 kbit/s. Die genaue Grenze kann

#### 4. Voruntersuchungen

---

aufgrund von Bauteiltoleranzen und thermischen Einflüssen abweichen

Es fällt in Abbildung 9 auf, dass im Gegensatz zu Abbildung 8 die Zeit, in der ein empfangenes Bit auf Low gesetzt wird, nicht proportional zu den gesendeten HIGH Bits kleiner wird. Genau genommen ändert sich mit zunehmender Baurate der Tastgrad des empfangenen UART-Signals. Somit könnte eine Anpassung des Abtastzeitpunktes erforderlich werden, um diesen idealerweise wieder in der Mitte der High-Phase zu positionieren. Da das Verhältnis der in Betracht kommenden Baudaten in etwa bei 50/50 liegt, wäre eine Anpassung des Abtastzeitpunktes auf 4/16 statt 2/16 eine Möglichkeit, um die Positionierung zu erreichen

Für die Wahl der maximalen Datenrate, mit der der Infrarot-Transceiver übertaktet betrieben werden kann, ist aus Stabilitätsgründen der vorletzte noch stabile Wert der Messreihe zu bevorzugen, bei 12.800 bit/s großen Abständen. Ein zu geringer Abstand zur höchsten Baudate könnte andernfalls zu Problemen führen. Daher fällt die vorläufige Wahl der präferierten Datenrate auf 192 kbit/s.

Bei der Messung mit der Persistenz-Funktion des Oszilloskops und der PRBS-Datenreihe kann es vorkommen, dass zwar grundsätzlich an allen Positionen korrekt Bits übertragen werden, jedoch eine erhöhte Fehlerrate auftreten kann. Eine exakte Messung der Fehlerrate ist sehr komplex und erfordert umfangreiche Hardware wie z.B. einen Logic-Analyzer sowie spezielle Software. Eine genauere Messung wäre möglich, doch im Hinblick auf den dafür erforderlichen Aufwand und die Aufgabenstellung ist dies nicht zweckmäßig, es sei denn, die Fehlerrate ist signifikant höher als die bei einem Betrieb mit 115.2 kbit/s.

Um eine einfache Abschätzung der Fehlerhäufigkeit vorzunehmen, wird daher ein pragmatischer Ansatz gewählt: Das Oszilloskop wird weiterhin mit der Persistenz-Funktion verwendet, jedoch wird anstelle der zuvor gesendeten PRBS-Datenreihe immer nur ein bestimmtes Zeichen gesendet. Mit diesem Verfahren werden auftretende Fehler visuell angezeigt. Zu Beginn der Messung wird eine Stoppuhr gestartet und bei Auftreten eines Fehlers gestoppt. Da dies nicht automatisch erfolgt, sondern durch einen Beobachter, kann es zu Verzögerungen von einigen Sekunden zwischen dem Erkennen eines Fehlers und dem Stoppen der Uhr kommen. Sollten jedoch über einen Zeitraum von mehreren Minuten keine Fehler auftreten, kann von einer sehr geringen Fehlerwahrscheinlichkeit ausgegangen werden.

Die Dauer von mehreren Minuten wurde gemäß den IrDA-Standards aus dem Buch “Wireless LAN Standards and Applications” gewählt, in dem die Bitfehlerrate (bit error rate) (BER) standardmäßig auf weniger als  $10^{-8}$  festgelegt ist.

$$Messzeit_{min} = \frac{\text{Anzahl der gesendeten Bits}}{\text{Baudrate}} \quad (4.3)$$

#### **4. Voruntersuchungen**

---

Bei einer Anzahl gesendeter Bits von  $10^8$  und einer Baudrate von 192 kbit/s ergibt sich eine minimale Messzeit von etwa 520 Sekunden. Um eine adäquate Abschätzung für diese Arbeit vorzunehmen, werden daher drei erfolgreiche Messungen von 10 Minuten (600 Sekunden) angestrebt. Da das Auftreten eines Fehlers rein zufällig ist und selbst bei niedriger BER bereits in den ersten Minuten auftreten kann, wird bei einem Fehler ein neuer Versuch gestartet. Ein erfolgreiches Ergebnis wird daher erst als solches angesehen, wenn alle drei Messungen fehlerfrei verlaufen und die Anzahl der Fehlversuche entsprechend gering bleibt.

Tabelle 2: Minuten bis zum Auftreten eines Fehlers in drei Versuchen

Erfolgreiche Durchläufe (10min)	abgebrochene Durchläufe
3 von 3	0

Die durchgeföhrten Messungen zeigen, dass bei der Verwendung des Infrarot-Transceivers mit einer erhöhten Baudrate von 192 kbit/s keine unverhältnismäßig hohe BER auftritt, sodass eine praktische Anwendung möglich ist.

### **4.4. Jitter-Messung**

Eine für die Messung geeignete Methode ist das periodische Senden eines bestimmten Datenbytes, welches durch abwechselnde High und Low Bits die Sendefrequenz messbar macht. Mit dem Oszilloskop ist eine automatisiert Frequenzmessung möglich, bei der die gemessenen Werte in einer CSV-Datei gespeichert werden. Anhand der Messwerte lassen sich dann die auftretenden Frequenzen entsprechend auswerten und in einer Häufigkeitsverteilung in einem Histogramm darstellen.

Die Bestimmung des Jitters der untersuchten Bauteile lässt sich im Folgenden durch die Messung des Jitters der UART-Schnittstelle mit entsprechender Baudrate durch zwei Messungen bestimmen. Dabei wird am Mikrocontroller Ausgang sowie nach der Übertragung mittels Bauteil gemessen.

#### **4.4.1. Jitter des Digitalisolator**

Die Abbildung 11 zeigt Frequenzen der präferierten Baudrate des Digitalisolators von 3 Mbit/s.

#### 4. Voruntersuchungen

---

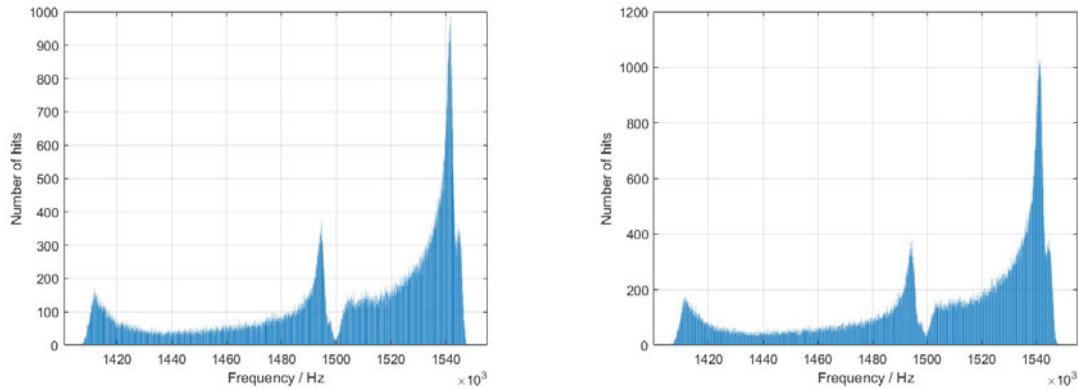


Abbildung 11: Frequenz-Histogramm der UART-Schnittstelle des XMC1100 vor (links) und nach dem Digitalisolator ISOFACE (rechts) bei einer Baudrate von 3 Mbit/s

Durch die in Tabelle 3 abgegebenen Randwerte der Messung lässt sich der Jitter Einfluss in Zahlen ausdrücken. Die Abweichungen von den Randwerten betragen hier nur wenige Hertz, was in Anbetracht der Frequenz von 1.5 MHz nicht einmal ein Promille beträgt.

Tabelle 3: Einfluss des Jitters durch den Digitalisolator bei einer Baudrate von 3 Mbit/s

	Maximalwert	Minimalwert
<b>Vor dem Digitalisolator</b>	1.555.080 Hz	1.400.820 Hz
<b>Nach dem Digitalisolator</b>	1.554.960 Hz	1.400.100 Hz
<b>Abweichung durch den Digitalisolator (absolut)</b>	-120 Hz	720 Hz
<b>Anstieg durch den Digitalisolator (in Promille)</b>	-0,077 %	0,514 %

Für eine Einschätzung zu problematischem Jitter-Verhalten, ist die folgende Abbildung 12 zu betrachten. In der Abbildung Abbildung 12 sind zwei UART-Übertragungen zu sehen, eine mit idealer Eingangsfrequenz (oben) und eine mit zu hoher Eingangsfrequenz (unten). Die roten vertikal angeordneten Pfeile markieren die Abtastzeitpunkte des Mikrocontrollers auf Empfängerseite und die in Blau dargestellten horizontalen Pfeile eine Bitdauer. Das UART-System synchronisiert sich nach einer halben Bitdauer mit dem Startbit und tastet dann periodisch im Abstand einer Bitdauer ab. Wenn diese Eingangsfrequenz zu hoch oder zu niedrig ist in Bezug auf die ideale Frequenz, kommt es zu Fehlern. Wie in der beispielhaften Darstellung in der Abbildung 12 (unten) zu sehen ist, ist die Frequenz zu hoch und das letzte Bit ist fehlerhaft.

#### 4. Voruntersuchungen

---

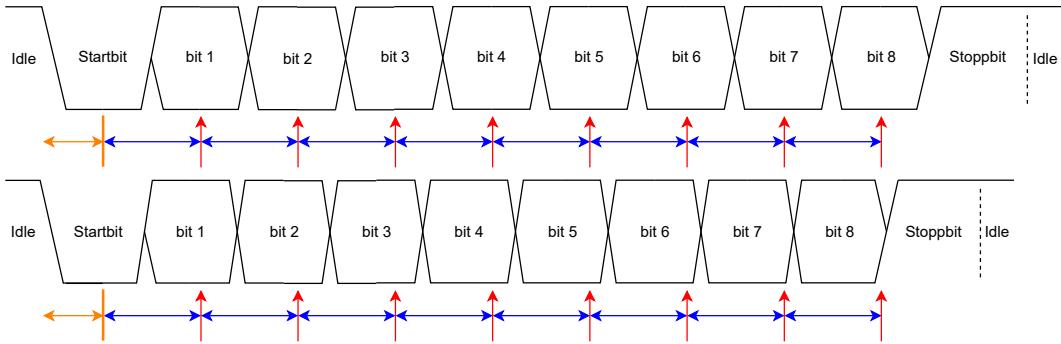


Abbildung 12: Darstellung einer UART-Bitrate mit idealer (oben) und mit zu hoher (unten) Eingangs frequenz

Für UART-Übertragungen mit 8 Datenbits, 1 Stopbit und einem Startbit gilt als Richtwert, dass eine maximale Abweichung von 3 % – 4 % von der idealen Frequenz noch zulässig ist, um eine normale Funktion der UART-Schnittstelle zu ermöglichen. Diese Grenze lässt sich aus einer Worst-Case-Betrachtung ableiten. Für den Worst Case muss der durch die Frequenzabweichung verursachte Fehler nach  $N = 10$  Bits dazu führen, dass es zu einer Abweichung von mindestens  $T_{\text{bit}}/2$  kommen muss, damit es zu Fehlinterpretation kommt. Mathematisch lässt sich das so ausdrücken:

- Anzahl der Bits pro Frame [ $N_{\text{bits}}$ ]
- relative Frequenzabweichung [ $\Delta f$ ]
- Dauer eines Bits [ $T_{\text{bit}}$ ]

$$N_{\text{bits}} \cdot \Delta f \cdot T_{\text{bit}} \geq \frac{T_{\text{bit}}}{2} \quad (4.4)$$

Anschließend lässt sich nun die Variable der Bitzeit aus der Gleichung kürzen, und die Abweichung wie folgt berechnen:

$$\Delta f \geq \frac{1}{2 \cdot N_{\text{bits}}} = \frac{1}{2 \cdot 10} = 0,05 \equiv 5\% \quad (4.5)$$

Hier wird erkennbar, dass bei einer Abweichung von 5% (sowohl positiv als auch negativ) die Verschiebung des Abtastzeitpunkts des letzten Bits so groß wird, dass das letzte Bit sicher fehlerhaft interpretiert wird.

Der durch den Digitalisolator verursachten Jitter ist gering, betrachtet man den durch den Mikrocontroller verursachten Jitter fallen große Abstände zur eingestellten Frequenz auf. Betrachten

#### 4. Voruntersuchungen

---

man die Werte in Tabelle 3 erneut, erhält man folgende Abweichungen für die obere (OG) und untere Grenze (UG):

$$\Delta f_{OG} = \frac{1.554.960 - 1.500.000}{1.500.000} = \frac{54.960}{1.500.000} \approx 0,03664 = 3,664\% \quad (4.6)$$

$$\Delta f_{UG} = \frac{1.400.100 - 1.500.000}{1.500.000} = \frac{-99.900}{1.500.000} \approx -0,0666 = -6,66\% \quad (4.7)$$

Mit einer Abweichung von 6.66 % ist das System so offensichtlich nicht für den Einsatz bei einer Baudrate von 3 Mbit/s zu gebrauchen. Der durch das Bauteil verursachte Jitter ist zwar sehr gering, aber der durch den Mikrocontroller verursachte Jitter bei 3 Mbit/s ist eindeutig zu hoch. Eine Überprüfung des Jitters beim Zielcontroller (XMC1404) ist unabdingbar, um gegebenenfalls eine Verringerung der Baudrate bei der Verwendung des Digitalisolators vorzunehmen.

#### 4.4.2. Jitter des Infrarot-Transceiver

In Abbildung 13 werden die gemessenen Frequenzen vor und nach der Übertragung mittels der Infrarot-Transceiver bei einer Baudrate 192 kbit/s als Histogramme dargestellt. Der direkte Vergleich der Histogramme in Abbildung 13 lässt in diesem Fall einen sichtbaren Jitter erkennen. Es ist hier zu erkennen, dass die vom Mikrocontroller erzeugte Form der Frequenzhäufigkeiten zwar auch so ähnlich übertragen wird, und zusätzlich eine nennenswerte Anzahl von deutlich schnelleren Frequenzen auftreten.

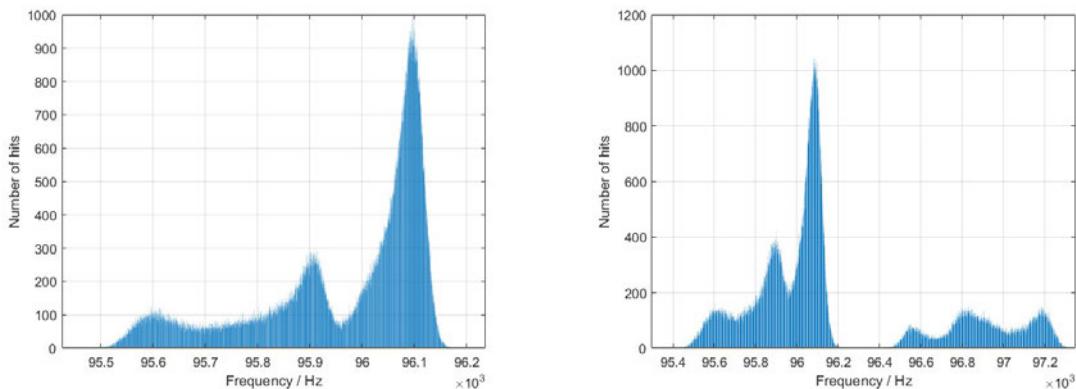


Abbildung 13: Frequenz-Histogramm der UART-Schnittstelle des XMC1100 vor (links) und nach (rechts) dem Infrarot-Transceiver bei einer Baudrate von 192 kbit/s

Durch die Betrachtung der in der Tabelle 4 angegebenen Grenzwerte lässt sich diese unerwartete Anomalie von schnelleren Frequenzen als Zahlenwert ausdrücken. Der untere Grenzwert ist recht

#### 4. Voruntersuchungen

---

unauffällig und nicht kritisch. Der obere hingegen Grenzwert hat eine circa 1 kHz große Abweichung.

Tabelle 4: Einfluss des Jitters durch den Infrarot-Transceiver bei einer Baudrate von 192 kbit/s

	Maximalwert	Minimalwert
<b>Vor dem Infrarot-Transceiver</b>	96.235,9 Hz	95.425,8 Hz
<b>Nach dem Infrarot-Transceiver</b>	97.347,2 Hz	95.299,1 Hz
<b>Abw. durch den IR-Transceiver (absolut)</b>	1111,3 Hz	126,7 Hz
<b>Anstieg durch den IR-Transceiver (in Promille)</b>	11,54 %	1,33 %

Um für den Infrarot-Transceiver eine Einschätzung über die Auswirkungen des Jitters zu erhalten, ist es notwendig, sich einen Überblick über die veränderten Bedingungen zu verschaffen. Da es sich um eine IrDA-Anwendung und daher um eine angepasste UART-Übertragung handelt, ist die Faustformel, welche im vorherigen Abschnitt behandelt wurde, nicht mehr ohne weiteres gültig. Das an der UART-Schnittstelle gemessenen Signal hat wie im Kapitel 4.3 und in Abbildung 14 anschaulich dargestellt nur halbe Bitzeiten. Pro gesendeter logischer 1 ist nur etwa die Hälfte der Bitzeit auf dem Low-Zustand geht und danach wieder in den Idle-Zustand von High.

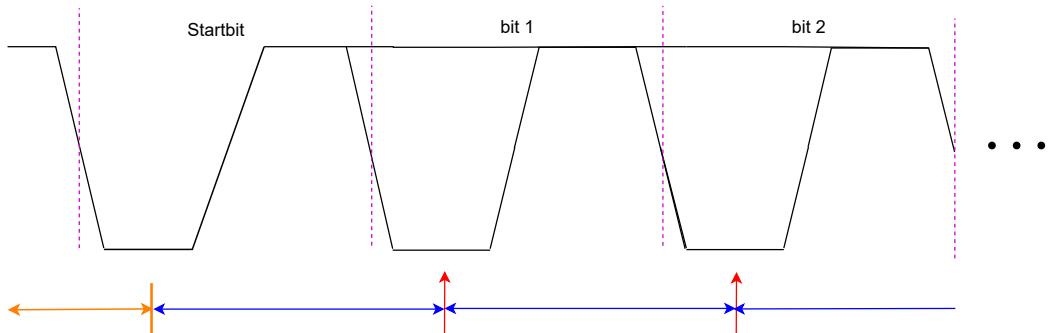


Abbildung 14: Darstellung einer UART-Bitrate für IrDA-Anwendungen mit idealer Übertragungsrate

Die neue Bestimmung der Abweichung wie beim Digitalisolator 4.4 mittels angepasster Formel, bei der der Abtastzeitpunkt verschoben ist, berechnet sich wie folgt:

$$N_{\text{bits}} \cdot \delta f \cdot T_{\text{bit}} \geq \frac{T_{\text{bit}}}{4} \quad (4.8)$$

$$\delta f \geq \frac{1}{4 \cdot N_{\text{bits}}} = \frac{1}{2 \cdot 10} = 0,025 \equiv 2,5\% \quad (4.9)$$

#### 4. Voruntersuchungen

---

Angesichts dieses Grenzwertes für Jitter-Abweichungen wären Abweichungen vom Idealwert im Bereich von bis zu 1.5 % noch akzeptabel. Zur Einschätzung der Jitter-Auswirkungen werden die Abweichungen der in Tabelle 4 gemessenen Maximalwerte der Frequenz berechnet:

$$\delta f_{OG} = \frac{97.347,2 - 96.000}{96.000} = \frac{1.347,2}{96.000} \approx 0,01403 = 1,403\% \quad (4.10)$$

$$\delta f_{OG} = \frac{95.299,1 - 96.000}{96.000} = \frac{-700,9}{96.000} \approx -0,0073 = -0,730\% \quad (4.11)$$

### 4.5. Einfluss der Strombegrenzung

In diesem Abschnitt wird der Einfluss des Vorwiderstands der IrDA-Sendediode auf die Funktionalität und den Einfluss auf den Bauteil-Jitter näher untersucht. Darüber hinaus wird der Stromverbrauch beim Senden analysiert, da ein zu hoher Energieverbrauch die Energiebilanz des BMS negativ beeinflussen könnte.

#### 4.5.1. Grenzwerte der Funktionalität

Das Ziel dieses Versuchs ist es herauszufinden, wie groß der Vorwiderstand der IrDA-Sendediode maximal sein darf, ohne dass die Funktionalität beeinträchtigt wird, und ob dieser Einfluss auf den Jitter hat. Die Messungen wurden mit einem Oszilloskop im Persistenz-Modus durchgeführt. Zur Einstellung des Vorwiderstands wurde ein 10 k $\Omega$ -Potentiometer verwendet.

Zunächst wurde die Übertragung ohne nennenswerten Vorwiderstand untersucht, um den Optimalfall mit maximalem Strom durch die LED zu betrachten. Wie in Abbildung 17 zu sehen ist, zeigt die PRBS-Datenübertragung bei einer Datenrate von 115.2 kbit/s und einem Vorwiderstand von 0.2  $\Omega$  eine einwandfreie Übertragung ohne auffällige Jitter-Effekte.

#### 4. Voruntersuchungen

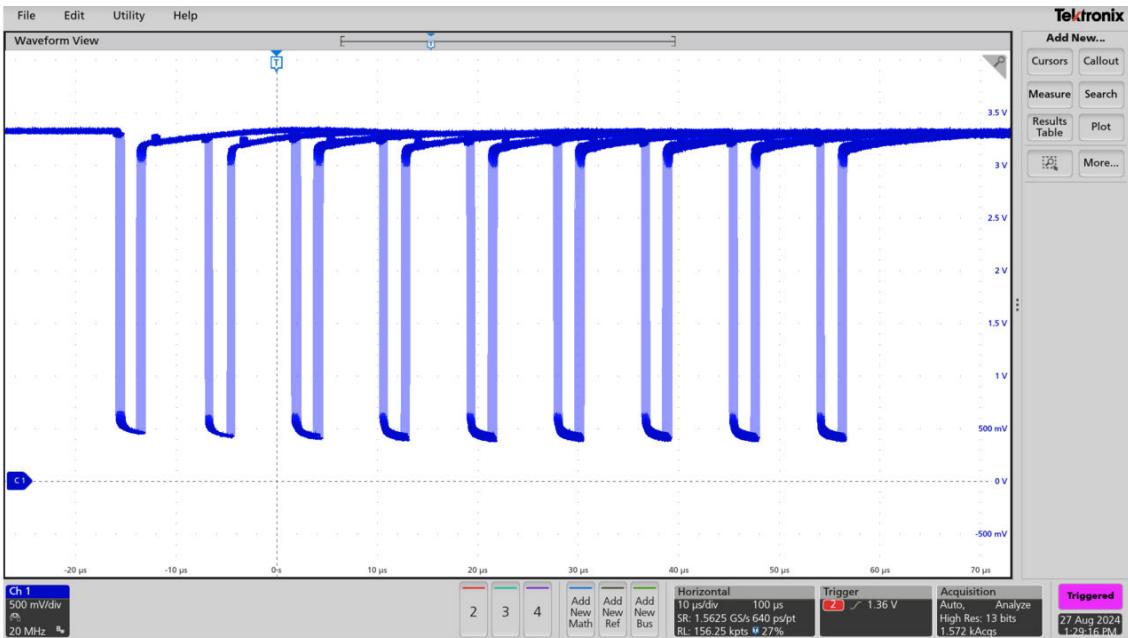


Abbildung 15: PRBS-Datenübertragung bei einer Datenrate 115.2 kbit/s und einem  $0.2 \Omega$  großem Vorwiderstand

Ab einem Vorwiderstand von  $2.4 \text{ k}\Omega$  wurde ein sichtbarer Anstieg des Jitters festgestellt. Dieser Anstieg ist besonders bei dem ersten Bit eines Übertragungsrahmens ausgeprägt. Wie in Abbildung 16 zu sehen, wird der Jitter bei langsamem Erhöhen des Widerstands deutlich sichtbar.

#### 4. Voruntersuchungen

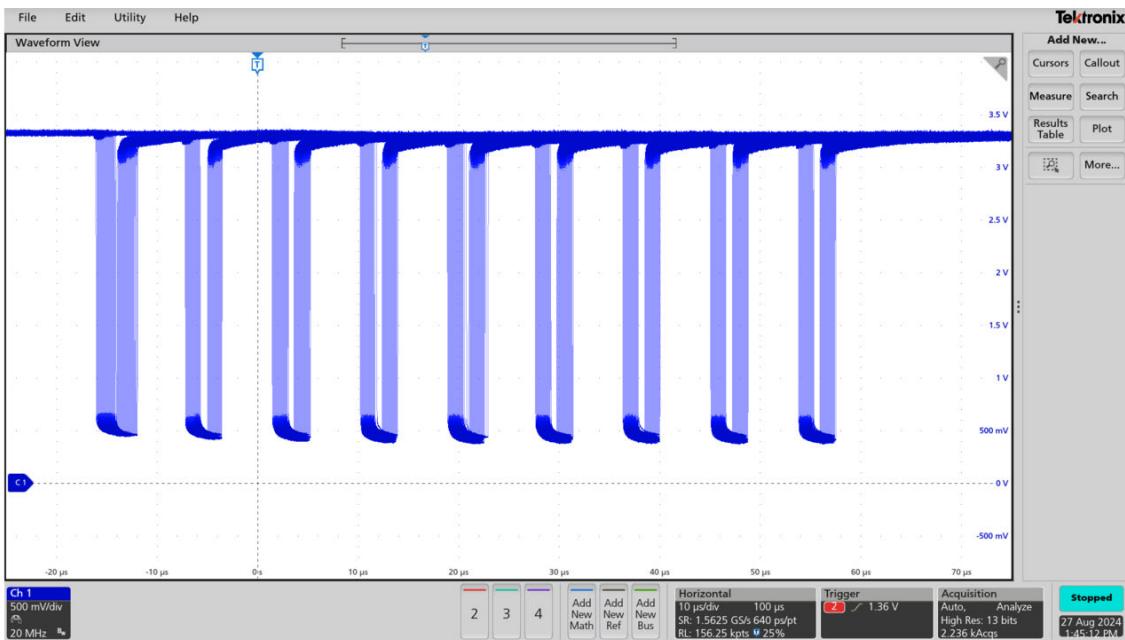


Abbildung 16: PRBS-Datenübertragung bei einer Datenrate 115.2 kbit/s und einem langsam erhöhtem Widerstandswert von  $0.2\Omega$  bis  $10\text{k}\Omega$

Bei einem Vorwiderstand von  $6\text{k}\Omega$  wurde die Funktionsschwelle erreicht. Zu diesem Zeitpunkt sind nur noch vereinzelt Bits korrekt übertragbar, was auf dem Oszilloskop sichtbar ist. Dies ist in Abbildung 17 im Vergleich zu den anderen Abbildungen, welche mit etwa derselben Aufnahmezeit gemacht wurden, gut erkennbar.

#### 4. Voruntersuchungen

---

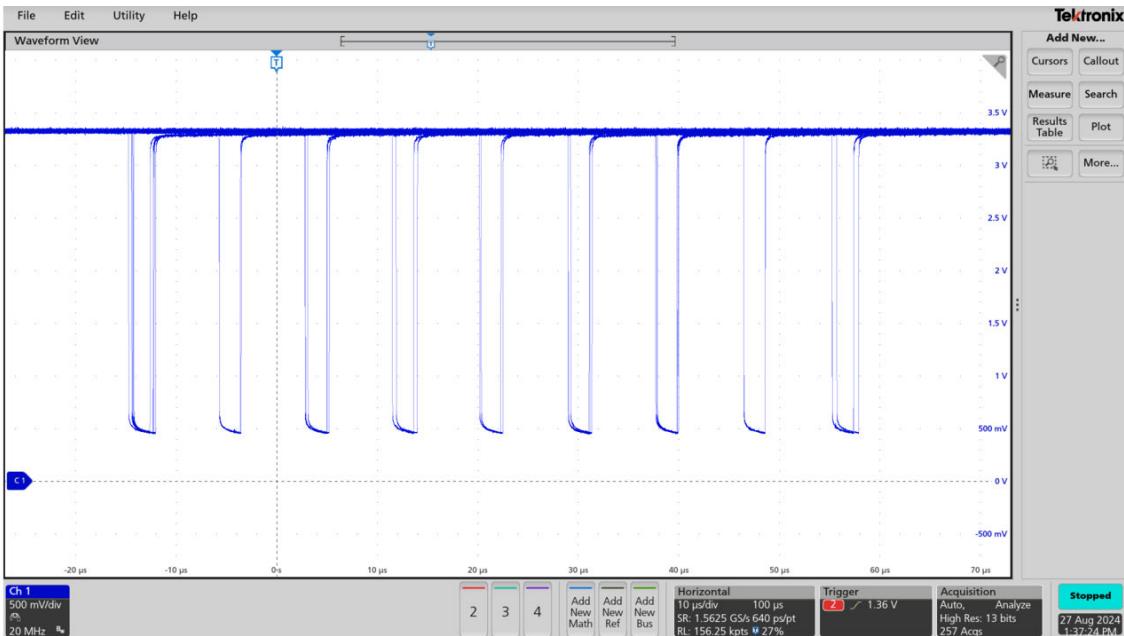


Abbildung 17: PRBS-Datenübertragung bei einer Datenrate 115.2 kbit/s und einem 6 k $\Omega$  großem Vorwiderstand

#### 4.5.2. Auswirkungen auf den Bauteil-Jitter

Wie bereits im Abschnitt 4.4 beschrieben, lässt sich ein deutlicher Einfluss des Vorwiderstands auf den Jitter der IrDA-Datenübertragung beobachten. Dieser Einfluss wird in diesem Abschnitt genauer untersucht. In Abbildung 18 sind sechs Histogramme dargestellt, die die Verteilung des Jitters bei verschiedenen Vorwiderstandswerten zeigen. Die X-Achse der sechs Bilder wurde einheitlich skaliert, um die Vergleichbarkeit zu gewährleisten.

Die in Abbildung 18 gezeigten Histogramme stellen die Verteilung des Jitters bei verschiedenen Vorwiderstandswerten dar. Die X-Achse der sechs Bilder wurde einheitlich skaliert, um die Vergleichbarkeit zu gewährleisten. Bei einem sehr geringen Vorwiderstand von  $0.2 \Omega$  traten keine hohen Frequenzen auf. Mit einem Vorwiderstand von  $25 \Omega$  wurden jedoch bereits erste Anzeichen von Jitter in höheren Frequenzbereichen sichtbar. Dieser Effekt verstärkte sich bei höheren Werten zunehmend, wie in den weiteren Abbildungen zu sehen ist.

Mit zunehmendem Widerstand vergrößert sich der Jitter merklich, insbesondere bei höheren Frequenzen. Je höher der Vorwiderstand, desto stärker ist die Streuung des Jitters, wie in den Abbildungen gut zu erkennen ist.

Zusätzlich zur Analyse des Jitters wurde die Spannung über dem Vorwiderstand mithilfe eines Oszilloskops gemessen, wobei die Maximalwerte des High-Pulses während einer 3/16 Bitzeit auf-

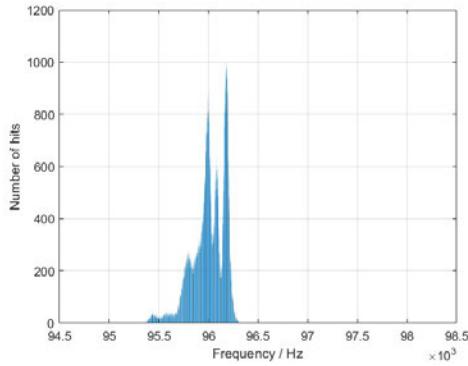
#### *4. Voruntersuchungen*

---

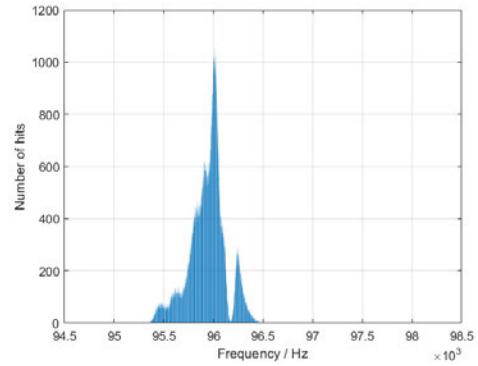
gezeichnet wurden. Der daraus berechnete Strom stellt nicht den Mittelwert über das gesamte High-Bit dar, sondern nur Wert den Spitzenwert. Auf dieser Basis wurden die in Tabelle 5 dargestellten Werte ermittelt, die den Zusammenhang zwischen Strom, Spannung und Jitter verdeutlichen. In Abbildung 19 wird dieser Zusammenhang grafisch dargestellt, um die Korrelation zwischen Stromfluss und Jitter zu verdeutlichen.

#### 4. Voruntersuchungen

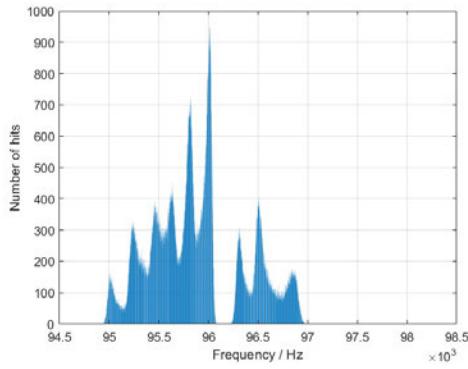
---



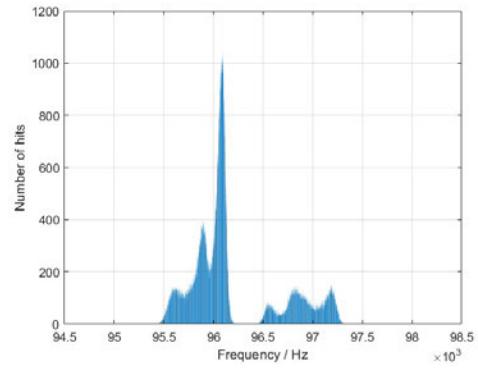
Vorwiderstand  $0.2 \Omega$



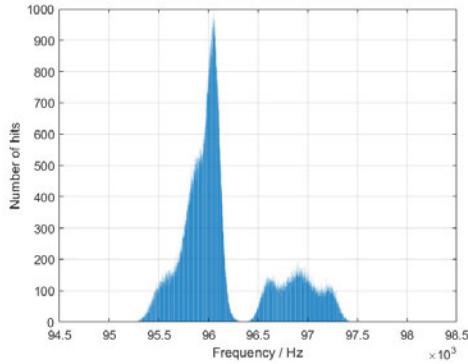
Vorwiderstand  $25 \Omega$



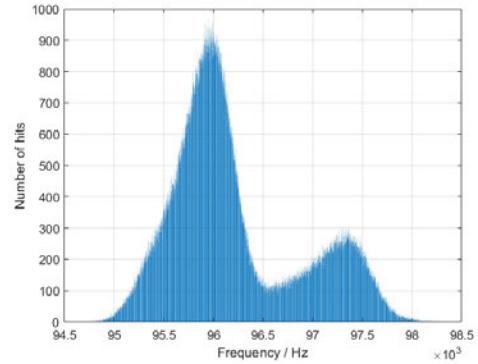
Vorwiderstand  $200 \Omega$



Vorwiderstand  $1 \text{ k}\Omega$



Vorwiderstand  $2 \text{ k}\Omega$



Vorwiderstand  $4 \text{ k}\Omega$

Abbildung 18: Histogramme bei einer Baudrate von 192 kbit/s unterschiedlichen Vorwiderständen

#### 4. Voruntersuchungen

---

Widerstand (Ohm)	Spannung $U$ (V)	Strom $I$ (A)	Maximaler Jitter (Hz)	Minimaler Jitter (Hz)
0,2	0,260	1,30	96.427,2	95.329,4
5	0,9749	0,195	96.517,1	95.173,8
25	1,65	0,066	96.544,7	95.311,3
200	1,9	9,5e-3	97.062,1	94.854,0
1k	1,99	1,99e-3	97.347,2	95.299,1
2k	2,02	1,01e-3	97.611,3	95.094,3
4k	2,035	509e-6	98.930,4	94.535,3

Tabelle 5: Spannung am Vorwiderstand, Strom und Jitter-Werte für verschiedene Widerstände

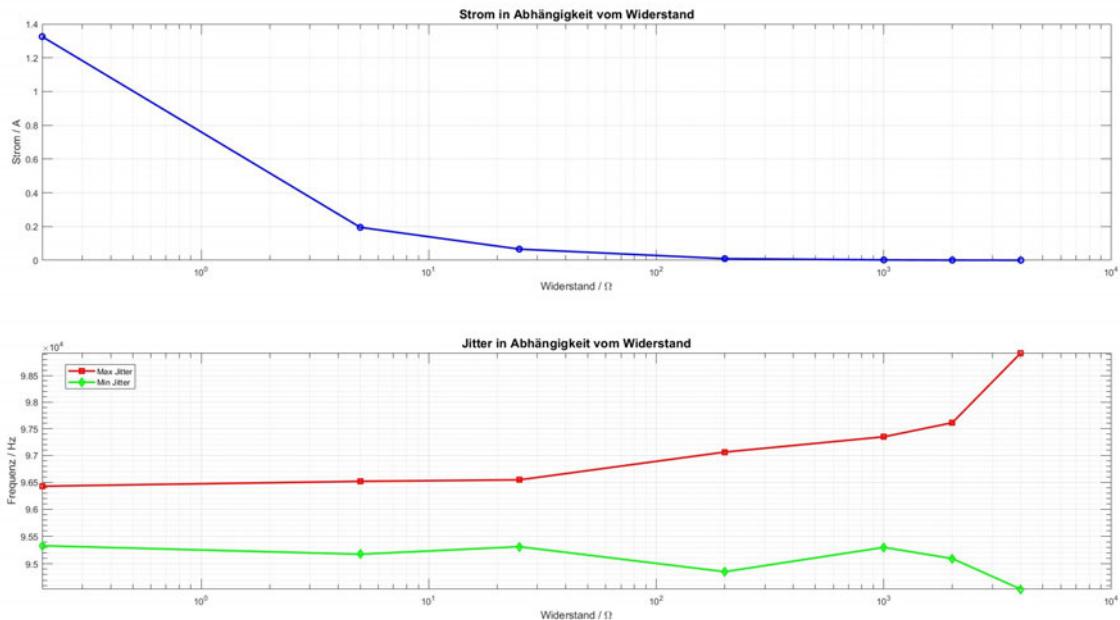


Abbildung 19: Abhängigkeit des Bauteil-Jitters vom verwendeten Vorwiderstand

#### 4.5.3. Auswirkungen auf die Stromaufnahme

Der durchschnittliche Stromverbrauch beim Senden soll abgeschätzt werden. Zum Vergleich dient der Digitalisolator ISOFACE. Der Stromverbrauch beider Technologien (Digitalisolator und Infrarot-Transceiver) wird unter der Annahme verglichen, dass die Ansteuerung durch abwechselnde HIGH- und LOW-Pegel erfolgt, was ein durchschnittliches Szenario darstellt.

Für eine Einschätzung: Der Digitalisolator ISOFACE hat laut Datenblattangaben bei einer Übertragungsrate von 3 Mbit/s einen maximalen Stromverbrauch von 4 mA [1]. Im Vergleich dazu

#### 4. Voruntersuchungen

---

wird der Stromverbrauch des IrDA-Transceivers bei einer Datenrate von 192 kbit/s und einem Vorwiderstand in Abhängigkeit von einem 3/16 HIGH-Puls abgeschätzt. Zusätzlich beträgt der maximale Stromverbrauch im Leerlaufmodus (Idle) während des Empfangens  $110 \mu\text{A}$ , der zum Gesamtstromverbrauch hinzugerechnet werden muss.

Die folgende Tabelle 6 zeigt die berechneten Stromverbräuche für verschiedene Vorwiderstände:

Vorwiderstand (Ohm)	3/16 Pulsstrom (mA)	Strom pro HIGH-Bit (mA)	Durchschn. Stromverbrauch (mA)
0,2	1300	243	121,5
5	195	36,5	18,25
25	66	12,37	6,185
200	9,5	1,78	0,89
1k	1,99	0,37	0,185
2k	1,01	0,187	0,093
4k	0,509	0,095	0,047

Tabelle 6: Stromverbrauch bei verschiedenen Vorwiderständen während der IrDA-Übertragung

Der Digitalisolator überträgt Daten etwa 15 Mal schneller als der IrDA-Transceiver, was für den Vergleich zu berücksichtigen ist. Der Stromverbrauch des IrDA-Transceivers bei einem Vorwiderstand von  $1 \text{k}\Omega$  (durchschnittlich  $0.185 \text{ mA}$ ) ist dem des Digitalisolators am ähnlichsten. Betrachtet man den Jitter in Abbildung 19 sowie die in Abschnitt 4.4 erwähnte 1.5 %-Grenze, scheint ein Vorwiderstand von  $1 \text{k}\Omega$  hinsichtlich des Jitters und in Anbetracht des Versuchsaufbaus, bei dem sich die Infrarot-Transceiver gegenüber stehen, eine gute Wahl für den Betrieb des IrDA-Transceivers zu sein.

Es ist darauf hinzuweisen, dass die Messungen die Sendeleistung nicht konkret berücksichtigten und der Wert von  $1 \text{k}\Omega$  für andere Aufbauten möglicherweise zu hoch ist. Laut Datenblatt des IrDA-Moduls ein Widerstand von  $22\text{Ohm}$  empfohlen wird, um die IrDA-Spezifikation zur Abstrahlleistung einzuhalten. Im Projekt wird anstelle der Lichtausbreitung im Freiraum ein Lichtleitkörper eingesetzt, insofern besteht die Erwartung, dass eine geringere Abstrahlleistung ausreicht. Diese Erwartung wäre am konkreten Aufbau zu überprüfen.

## 5. Experimentalaufbau

In diesem Kapitel werden die Hardwareanforderungen für die zentral synchronisierte Steuerung der ADC für einen praxisnahen Testlauf optimiert. Dabei kommen die zuvor untersuchten Bauteile für die UART Kommunikation zum Einsatz. Außerdem werden die Eigenschaften (ADC-Messzeiten, Jitter) des Experimentalaufbaus zur theoretischen Bestimmung des maximalen Synchronisationsfehlers, der EIS verwendet.

### 5.1. Systementwurf

Das erforderliche System ist, wie in Aufgabenstellung Kapitel 1.2 beschrieben, mittels geeigneter Maßnahmen zu entwerfen. Die im Folgenden aufgelisteten Kriterien und Abwägungen sind für die Entscheidung des Systementwurfs maßgeblich.

- **Topologie:**

Die in Kapitel 3.2 vorgestellten Topologien bieten unterschiedliche Vor- und Nachteile in Bezug auf Komplexität, Kosten, Skalierbarkeit und Echtzeitfähigkeit. Für das geplante System mit 18 Messplatinen und einer zentralen Platine wurde die Entscheidung für eine Bus-Topologie getroffen. Diese Topologie ermöglicht es allen Geräten, über eine gemeinsame Datenleitung zu kommunizieren, was die Implementierung einfach und kostengünstig macht. Besonders bei der Verwendung von UART-Kommunikation ist die Bus-Topologie vorteilhaft, da sie weniger Verdrahtungsaufwand im Vergleich zu anderen Topologien erfordert und die Anzahl der notwendigen Verbindungen reduziert.

Eine Stern-Topologie ist nur bei einer geringeren Anzahl von Teilnehmern sinnvoll gewesen, da sie eine direkte Verbindung jedes Geräts zur zentralen Platine voraussetzt. Dies würde bei 18 Teilnehmern den Verdrahtungsaufwand erheblich erhöhen. Zudem sollen in diesem System Broadcast-Befehle an alle Teilnehmer gleichzeitig gesendet werden, was bei einer Stern-Topologie komplexer zu realisieren wäre.

Auch die Ring- und Baum-Topologien wurden als ungeeignet bewertet. Die Ring-Topologie wäre ineffizient für eine so große Anzahl an Teilnehmern, da der Datenfluss in einem Ring-Netzwerk linear ist und die Übertragungsgeschwindigkeit bei wachsender Teilnehmerzahl abnimmt. Eine Baum-Topologie würde ebenfalls eine aufwändigere Hierarchie in der Kommunikation erfordern, was für die geplante Echtzeit-Kommunikation unvorteilhaft wäre. Deshalb bietet die Bus-Topologie für dieses Szenario die beste Balance zwischen Einfachheit, Skalierbarkeit und Effizienz.

## 5. Experimentalaufbau

---

- **Buszugriffsverfahren:**

In Kapitel 3.3 wurden verschiedene Buszugriffsverfahren vorgestellt, die jeweils unterschiedliche Vor- und Nachteile in Bezug auf Echtzeitfähigkeit, Komplexität und Effizienz aufweisen. Für das geplante System wurde das Master-Slave-Verfahren gewählt. Dieses Verfahren bietet eine zentrale Buszuteilung, bei der der Master die Kontrolle über den Zugriff auf den Bus behält. Dies ist besonders vorteilhaft, wenn wie in diesem System Broadcast-Nachrichten an alle Teilnehmer gleichzeitig gesendet werden sollen. Der Master kann diese Nachrichten steuern und sicherstellen, dass alle Geräte zur gleichen Zeit die relevanten Informationen empfangen.

Das Token-Prinzip wird als weniger geeignet eingestuft, da es eine dezentrale Buszuteilung nutzt, bei der kein einzelner Master existiert, um den Buszugriff zu steuern. Dies würde die zeitgleiche Ansprache aller Teilnehmer erschweren und die Synchronisation bei Broadcast-Nachrichten beeinträchtigen.

Weiterhin muss das System echtzeitfähig sein, um eine präzise und rechtzeitige Datenübertragung zu garantieren. Daher wurde das CSMA-Verfahren ausgeschlossen, da es durch die zufällige Zugriffsregelung keine deterministische Kommunikation ermöglicht und nicht echtzeitfähig ist.

Die Busarbitration, die ebenfalls eine zentrale Buszuteilung vorsieht, ist eine alternative Option gewesen. Sie bietet einen klaren Vorteil in Bezug auf die Synchronisation und Minimierung von Verzögerungen, ist jedoch komplexer und mit einem höheren Implementierungsaufwand verbunden. Da die Anforderungen des Systems bereits durch das Master-Slave-Verfahren ausreichend abgedeckt werden, wurde von der Busarbitration aufgrund der zusätzlichen Komplexität abgesehen.

- **Kommunikationsprotokoll:**

Das Kommunikationsprotokoll sollte möglichst kurz bzw. effizient sein. Für die Mindestanforderung bedarf es einer Adressierung, welche einen Bereich für die Adresse der Mikrocontroller, ein Kommando für Anweisungen und Messdaten zum Übertragen bietet. Aus den Projektanforderungen lassen sich folgende Speichergrößen bestimmen:

- **Adresse:** Für die Adressierung der einzelnen Platinen benötigt es mindestens 19 unterschiedliche Adressen. Diese setzen sich aus 18 Platinen zur Zellmessung sowie einer Masterplatine zusammen. Dafür benötigt es mindestens 5 Bit.
- **Kommando:** Um die unterschiedlichsten Werte der Mikrocontroller auszulesen und andere Anweisungen zu senden, braucht es mindestens 20 bis 30 Kommandos. Diese

## 5. Experimentalaufbau

---

Informationen sind aus ähnlichen Arbeiten der ProMoBiS-Arbeitsgruppe zu entnehmen. Da es bei der Entwicklung des fertigen Systems durchaus dazu kommen kann, sollte hier zusätzlich noch mehr Platz für weitere Kommandos gelassen werden. Aus diesem Grund werden mindestens 6 Bit, das heißt Platz 64 Kommandos benötigt.

- **Messdaten:** Die Daten, welche übertragen werden müssen, haben einen Speicherbedarf von 12 Bit, da das die Auflösung der ADC-Wandlung ist.

Insgesamt werden mindestens 23 Bit zu Datenübertragung benötigt. Da mit jedem UART-Frame 8 Datenbit zur Verfügung stehen, muss ein Telegramm mindestens drei Frames (24 Datenbit) aufgebaut sein. Würde man ein Frame mehr zur Übertragung verwenden, hätte das den Vorteil einer besseren Erweiterbarkeit. In diesem Fall kann man der Adresse und dem Kommando ein eigenes Frame mit 8 Bit freihalten und zusätzlich wären noch 4 weitere Bits zur Verfügung, sodass man beispielsweise die Messsysteme auf einen ADC mit 16 Bit erweitern könnte. Die Entscheidung des Protokolls ist eine Abwägung, welche in Anbetracht des zeitlichen Mehraufwands bei der Datenübertragung getroffen werden muss. Abbildung 20 zeigt die in Erwägung gezogenen Kommunikationstelegramme.



Abbildung 20: Mögliche Telegramme für die UART-Kommunikation

Zusätzlich ist anzumerken, dass eine Broadcast-Nachricht, welche an alle Messplatinen gerichtet werden kann, nur ein einziges Frame mit 8-Bit als Kommando bedarf.

## 5.2. Hardware- und Systemaufbau

Die entsprechende Hardware wurde für den Testaufbau im Rahmen des ProMoBiS-Verbundprojektes bereits auf Prototyp-Ebene entwickelt. Dies beinhaltet mehrere mit den Mikrocontrollern von Infineon der Typenreihe XMC1404 bestückten Platinen (Abbildung 21), auf denen sich ein Infrarot-Transceiver des Typs TFDU4101 befindet. Diese Platinen sollen zum Erfassen der Batterie-Messdaten und zur Kommunikation dienen. Die Platinen lassen sich auf einer größeren Trägerplatine (Abbildung 22) festschrauben, welche in einer finalen Anwendung auf dem Batterieblock eines Automobils befestigt werden kann. Ein entsprechend zur Infrarotübertragung geeigneter Lichtleitkörper (Abbildung 23) existiert ebenfalls. Des Weiteren ist die Trägerplatine mit mehreren Digitalisolatoren des Typs 2DIB141xF bestückt. Für jeden Platinenplatz existiert 1 Digitalisolator mit

## 5. Experimentalaufbau

---

Anbindung an die montierbaren Platinen. Die Trägerplatine umfasst insgesamt 18 Platinenplätze, die genutzt werden können.

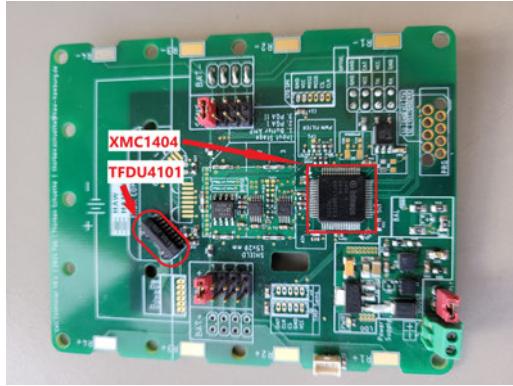


Abbildung 21: Messplatine mit bestücktem Mikrocontroller (XMC1404) und Infrarot-Transceiver (TFDU4101)

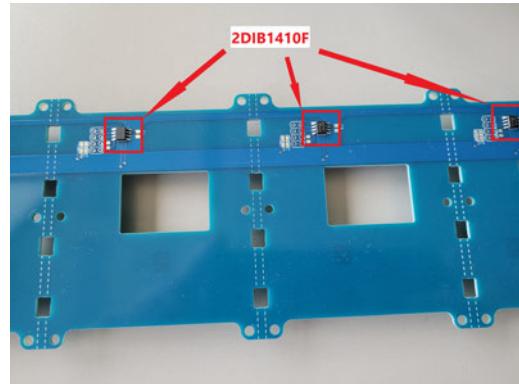


Abbildung 22: Ausschnitt der Trägerplatine mit bestückten Digitalisolatoren (2DIB1410F)

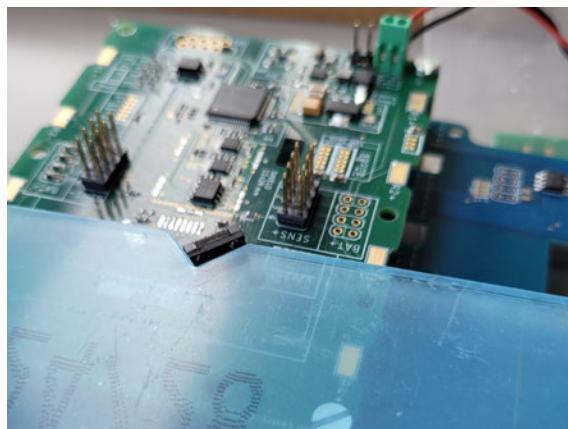


Abbildung 23: Lichtleitkörper an dem Infrarot-Transceiver der Messplatine

### 5.3. Anpassungen

Im Kapitel 4 wurden verschiedene Eigenschaften der verwendeten Kommunikationsbauteile genauer untersucht. Aus den dabei gewonnenen Erkenntnissen lassen sich die hier beschriebenen Anpassungen für den Experimentalaufbau ableiten.

## *5. Experimentalaufbau*

---

- **Baudatenanpassung:**

Aus dem Kapitel 4.4 ist zu entnehmen, dass die in dem Kapitel 4.3 gewählte Baudrate für den Betrieb des Digitalisolators ungeeignet ist. Aus diesem Grund ist eine niedrigere Baudrate für die UART-Verbindung mittels Messung zu bestimmen. Wie in Kapitel 4.4 beschrieben ist, sollte die maximale Abweichung der Frequenz durch den auftretenden Jitter nicht größer sein als  $\pm 3\%$ , um die Funktionalität der Verbindung zu gewährleisten.

- **Vorwiderstandsanpassung:**

In dem Kapitel 4.5 wurden die Auswirkungen verschiedener Vorwiderstände auf ihren Stromverbrauch und auf den Einfluss einer reduzierten Sendeleistung auf den Bauteil-Jitter untersucht. Durch das Testszenario im Kapitel 4.1 lässt sich nicht sagen, ob die höheren Widerstandswerte in diesem Experimentalaufbau auch die nötige Sendeleistung erbringen, um eine geeignete Funktion zu gewährleisten oder ob auf geringere Widerstandswerte zurückgegriffen werden muss.

Beide zur UART-Kommunikation verwendeten Technologien werden in einem Funktionstest auf dem Experimentalaufbau unter Bedingungen entsprechend zum finalen Produkt geprüft. Die hohe Baudrate des Digitalisolators sowie der Vorwiderstand des Infrarot-Transceivers sind dabei die Stellschrauben, welche es gegebenenfalls zu optimieren gilt.

## **5.4. Mess- und Versuchsplan**

- **Jitter-Messung:**

Für die Bestimmung der Synchronität bei der EIS ist es wichtig auch den Jitter bei der UART-Übertragung mit den verwendeten Messplatten und dem hier montierten Mikrocontroller XMC1404 zu kennen. Für die Messung wird praktisch dieselbe Software wie für den XMC1100 in Abschnitt 4.4 des Kapitels 4 verwendet. Der einzige Unterschied ist die Anpassung an den neuen Mikrocontroller.

Zusätzlich wird gemäß der in Abschnitt 5.3 beschriebenen Anpassungen eine neue Baudate für den Digitalisolator gesucht, indem die Histogramme bei verschiedenen geringeren Baudaten aufgenommen und auf ihren Jitter ausgewertet werden. Bei dieser Messung dient ebenfalls die in Abschnitt 4.4 verwendete Software.

- **ADC Messung:**

Für die Synchronität der elektrochemische Impedanzspektroskopie (EIS) ist es entscheidend, die Laufzeit des ADC zu kennen, um den Synchronisationsfehler zuverlässig bewerten

## 5. Experimentalaufbau

---

zu können. In diesem Zusammenhang wird eine ADC-Messung mit acht Kanälen durchgeführt, wobei die Kanäle nacheinander mittels eines Multiplexers abgetastet werden. Jede Messung setzt sich aus einer Abtastzeit und einer Verarbeitungszeit zusammen [3].

Laut Datenblatt, bei einer Versorgungsspannung von 3.3 V und einer ADC-Taktfrequenz von 48 MHz, beträgt die minimale Abtastzeit  $3 \cdot 1/f_{ADC}$  und die Verarbeitungszeit  $20 \cdot 1/f_{ADC}$  bei einer Auflösung von 12 Bit. Das ergibt eine minimale Verarbeitungszeit pro Kanal von etwa 480 ns. In der DAVE-APP ergibt die Konfiguration für die UART-Kommunikation eine Gesamtverarbeitungszeit von 1354 ns, wobei die Abtastzeit flexibel angepasst werden kann. Für diese Berechnung wurde die minimale Abtastzeit von 125 ns festgelegt.

Um die tatsächliche Verarbeitungszeit pro Kanal messtechnisch zu ermitteln, kann die ADC-Messung durch ein externes Signal periodisch getriggert und durch eine Reihe von Messungen quantifiziert werden.

Abbildung 24 veranschaulicht das folgende Vorgehen:

Vor dem Start der ADC-Messung wird ein GPIO-Pin auf "High" gesetzt und nach Abschluss der Messung mithilfe einer Callback-Funktion wieder auf "Low". Dies schafft ein Zeitfenster, das die Dauer der ADC-Verarbeitungszeit repräsentiert. Zwar verursachen das Setzen und Zurücksetzen des General Purpose Input/Output (GPIO)-Pins minimale Verzögerungen, jedoch nimmt bei mehreren Kanälen das gesamte Messfenster proportional zur Anzahl der gemessenen Kanäle zu. Diese Methode ermöglicht es, die reale Verarbeitungszeit pro Kanal präzise zu bestimmen.

## 5. Experimentalaufbau

---

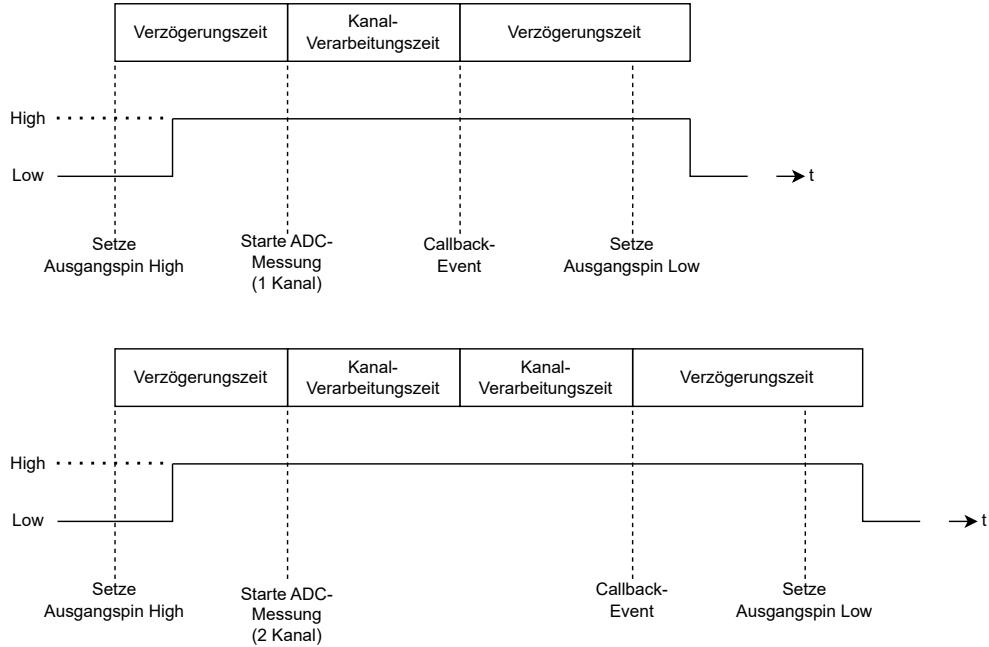


Abbildung 24: Schematische Darstellung zur Messung der ADC-Verarbeitungszeit

- **Funktionstest:**

Um die Funktionalität des Systems abschließend zu validieren, wird ein Funktionstest mit mehreren Mikrocontrollern durchgeführt. Auf der Trägerplatine werden hierfür drei Messmodule montiert, wobei zwei als Slave-Module und eines als Master-Modul agieren.

Im regulären Betrieb einer EIS wird der zentrale Batteriestrom durch ein Wechselstromsignal angeregt und vom Master-Modul gemessen. Die phasenverschobenen Spannungen der einzelnen Batteriezellen werden von den Slave-Modulen erfasst. Um dieses Szenario realistisch zu simulieren, werden im Test zwei phasenverschobene Spannungen mit einem Funktionsgenerator erzeugt. Eine dieser Spannungen repräsentiert den angeregten Batteriestrom, während die andere die entsprechende Spannungsantwort simuliert.

Die aufgezeichneten Spannungswerte werden anschließend auf Synchronität und Phasenverschiebung hin ausgewertet. Für diese Analyse wird eine Matlab-Schnittstelle implementiert, über die die Daten direkt eingelesen werden können. Die Auswertung der Messergebnisse erfolgt mit einem eigens entwickelten Matlab-Skript, das eine präzise Analyse der Phasendifferenzen ermöglicht.

## 5.5. Testsoftware

- **Allgemein:**

Die Kommunikation zwischen den Mikrocontrollern erfolgt über die UART1-Schnittstelle, während die UART0-Schnittstelle zur Anbindung an den PC und zur Kommunikation mit dem Matlab-Skript dient. Der Softwareentwurf dient als Test zur Messung der Spannungen für die EIS bei einer bestimmten, am Funktionsgenerator eingestellten Anregungsspannung. Der Softwareentwurf für den Funktionstest ist in drei Hauptkomponenten unterteilt, die in den Flussdiagrammen Abbildung 25, Abbildung 26 und Abbildung 27 gezeigt sind.

- **Softwarefunktion:**

Nach der Initialisierung des Systems sendet das PC-Programm (Matlab) den Befehl "*Ping*" an das Master-Modul, um die Verfügbarkeit der Slave-Module zu überprüfen. Bestätigt der Master den erfolgreichen Empfang und die Verbindung zu allen Slave-Modulen, beginnt der Messvorgang. Der Befehl "*ADC-Messung*" veranlasst das Master-Modul, die Referenzspannung zu messen, welche den angeregten Strom für die EIS simuliert. Gleichzeitig sendet der Master ein Broadcast-Kommando an die Slave-Module, um die synchrone Spannungsmessung an den Batteriezellen zu starten.

Nach jeder Messung wird eine kurze, frequenzabhängige Verzögerung eingelegt, um sicherzustellen, dass für jede Frequenz eine ausreichend hohe Anzahl an Messpunkten pro Periode aufgenommen wird. Dieser Prozess wird in einer Schleife wiederholt, bis die Laufvariable  $i$  den voreingestellten Wert  $x$  erreicht, wobei  $x$  die gewünschte Anzahl an Messpunkten definiert.

Sobald die erforderliche Anzahl an Messungen abgeschlossen ist, fordert das Master-Modul die gemessenen Spannungswerte von den Slave-Modulen mithilfe des Befehls "*ADC-Kanal-Daten*" ab. Die Anzahl der verwendeten Slave-Module wird durch die Variable  $n$  bestimmt. Die aufgezeichneten Referenzspannungen des Masters sowie die Spannungsdaten der Slave-Module werden in Dateien gespeichert und anschließend in einem separaten Schritt analysiert.

## 5. Experimentalaufbau

---

### 5.5.1. Matlab Testsoftware

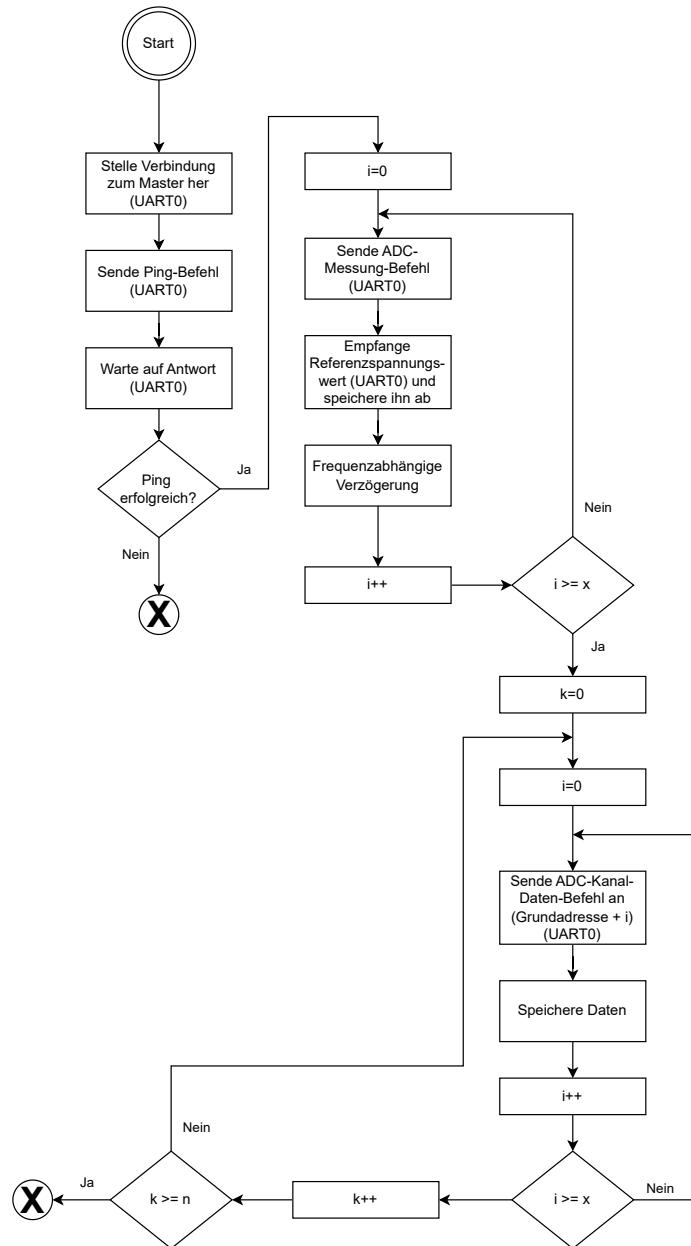


Abbildung 25: Flussdiagramm der Software des Matlab-Tests

## 5. Experimentalaufbau

### **5.5.2. Master-Modul Software**

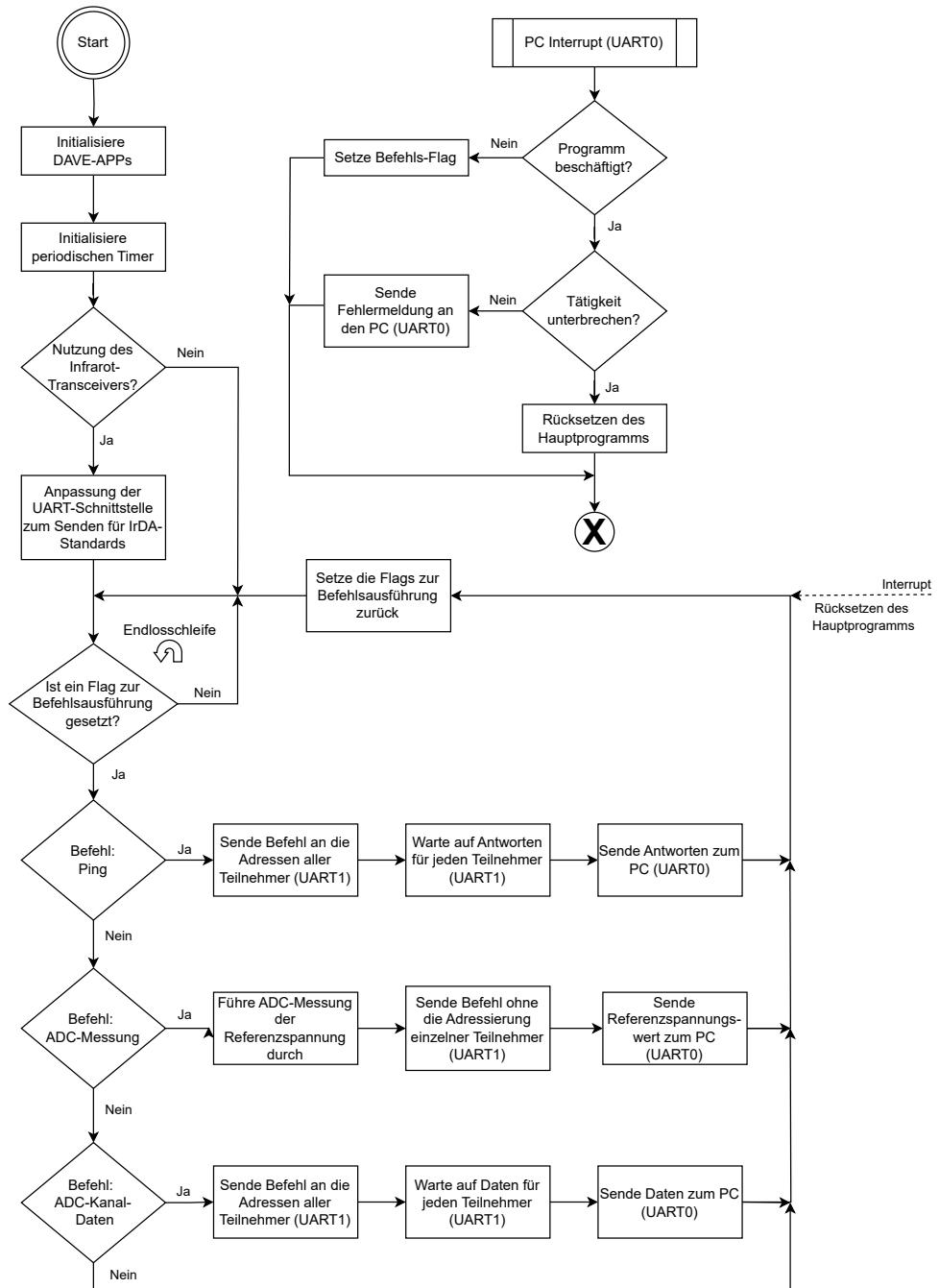


Abbildung 26: Flussdiagramm der Software des Master-Moduls

## 5. Experimentalaufbau

---

### 5.5.3. Slave-Modul Software

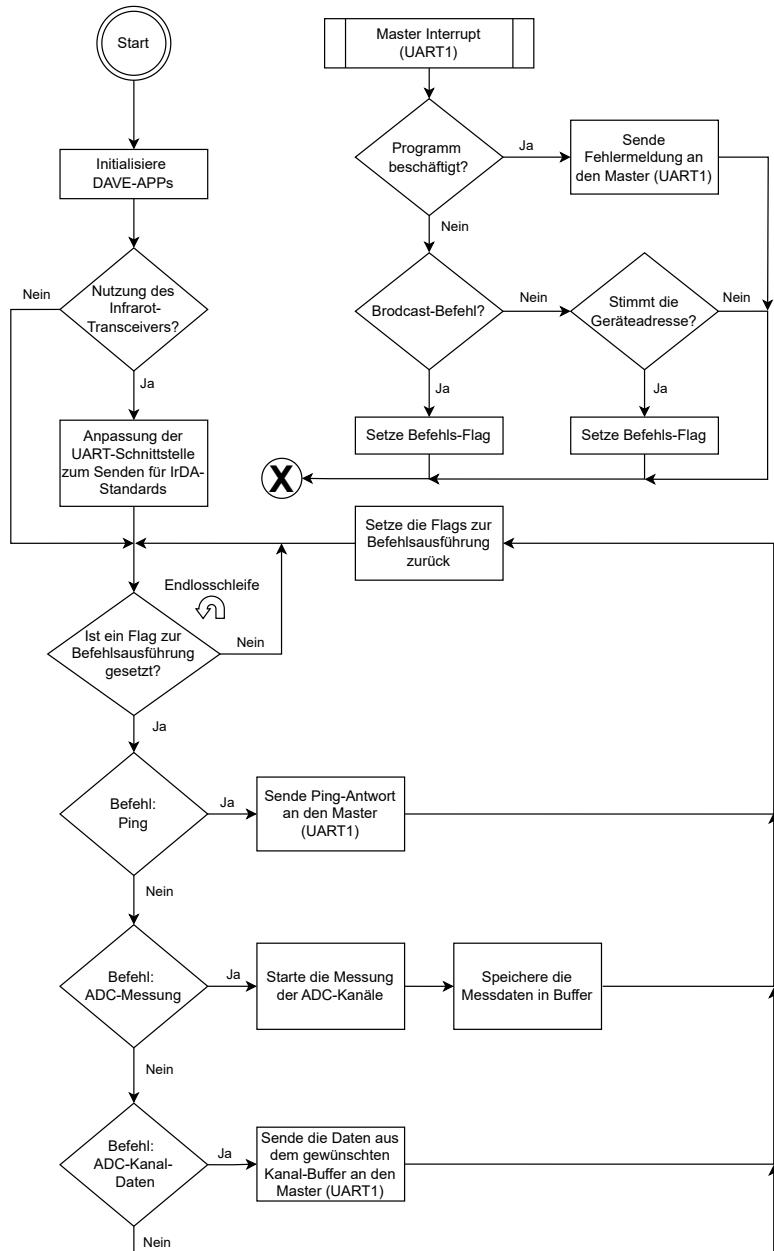


Abbildung 27: Flussdiagramm der Software des Slave-Moduls

## 5.6. Messungen

In diesem Abschnitt werden die in 5.4 beschriebenen Messungen durchgeführt.

### 5.6.1. Jitter-Messung des Mikrocontrollers (XMC1404)

- **Digitalisolator:**

Wie in Abschnitt 5.3 beschrieben, muss die Baudrate für den Betrieb des Digitalisolators neu gewählt werden. Dabei ist es wichtig, dass die Abweichung des maximalen Jitters innerhalb von  $\pm 3\%$  ( $\pm 30\%$ ) bleibt. Zur Bestimmung des Jitters wurden, wie in Kapitel 4.4 beschrieben, mehrere Histogramme für verschiedene Baudaten aufgenommen und die maximalen Frequenzabweichungen von der ideal eingestellten Baudate erfasst. In Tabelle 7 sind die minimalen und maximalen Jitterabweichungen bei unterschiedlichen Baudaten in Promille angegeben. Zusätzlich wurde der prozentuale Anteil für den Anstieg des Jitters durch den Digitalisolator, wie in Tabelle 3 dargestellt, zu den gemessenen Werten des Mikrocontrollers addiert, um den Gesamtjittereinfluss bei der Kommunikation zu ermitteln.

Tabelle 7: Jitter des Mikrocontrollers XMC1404 bei verschiedenen Baudaten

Baudrate	Oberer Jitter (in Promille)	Unterer Jitter (in Promille)	Ges. oberer Jitter (in Promille)	Ges. unterer Jitter (in Promille)
2 Mbit/s	22,71	43,57	22,63	44,08
1.5 Mbit/s	20,29	32,08	20,21	32,59
1.25 Mbit/s	28,24	2,05	28,16	2,56
1 Mbit/s	18,48	19,60	18,40	20,11

Die Analyse zeigt, dass die Wahl der Baudate von 1.25 Mbit/s den Anforderungen an den Jitter entspricht und daher für den Funktionstest mit dem Digitalisolator geeignet ist. Besonders hervorzuheben ist, dass die minimale Abweichung bei dieser Baudate im Vergleich zu den anderen gemessenen Baudaten sehr gering ausfällt. In Abbildung 28 ist das zugehörige Histogramm bei dieser Baudate dargestellt. Auffällig ist, dass höhere Frequenzen, im Gegensatz zu Abbildung 11, nur sehr selten auftreten. Dies könnte sich positiv auf die Bitfehlerrate (bit error rate) (BER) auswirken, da das seltene Auftreten hoher Frequenzen die Wahrscheinlichkeit von Übertragungsfehlern verringert.

## 5. Experimentalaufbau

---

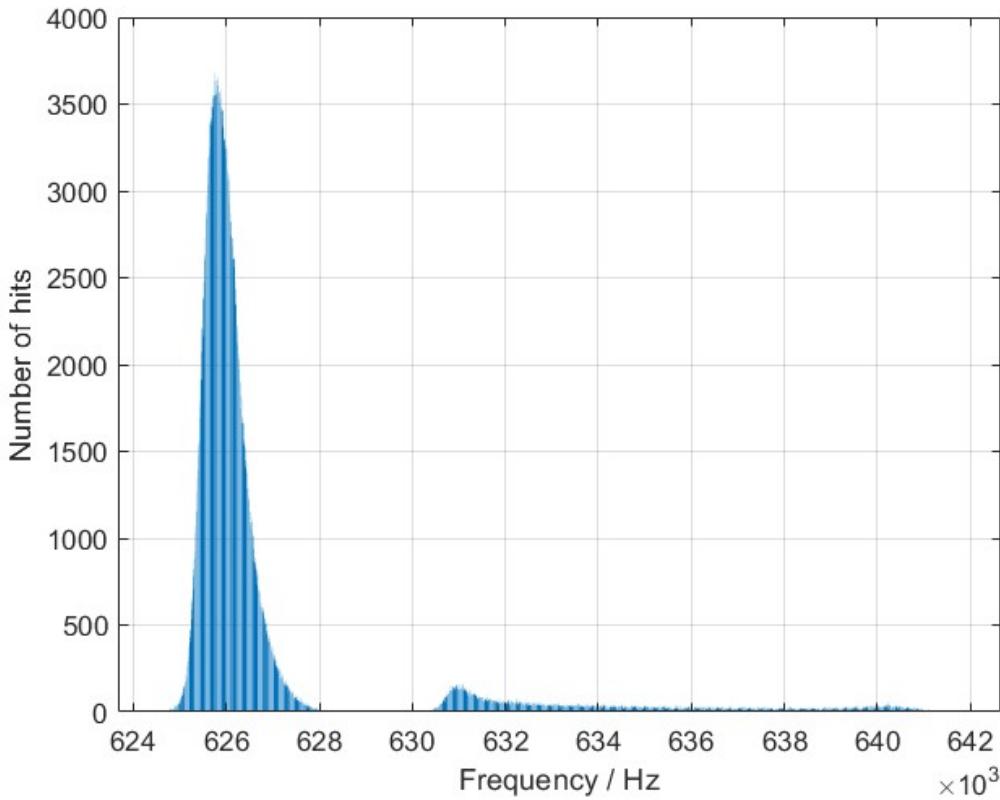


Abbildung 28: Jitter des Mikrocontroller XMC1404 bei einer Baudrate von 1.25 Mbit/s

- **Infrarot-Transceiver:**

Um sicherzustellen, dass der Infrarot-Transceiver bei derselben Baudrate wie Kapitel 4.3 (192 kBit/s) betrieben werden kann, wird der Jitter bei dieser Baudrate für den Mikrocontroller XMC1404 neu bestimmt. Die Tabelle 8 zeigt die Abweichung des Jitters in Promille und den durch den Infrarot-Transceiver verursachten Gesamtjitter bei der UART-Kommunikation. Für die Berechnung des Gesamtjitters wurden die Werte des Jitteranstigs aus Tabelle 4 verwendet. Der Gesamtjitter darf den Wert von 15 Promille nicht überschreiten wie in Kapitel 4.4 beschrieben wird.

Tabelle 8: Jitter des Mikrocontrollers XMC1404 bei 192 kBit/s

Baudrate	Oberer Jitter (in Promille)	Unterer Jitter (in Promille)	Ges. oberer Jitter (in Promille)	Ges. unterer Jitter (in Promille)
192 kbit/s	10,01	-1,98	21,46	-0,65

## 5. Experimentalaufbau

---

Wie in dem Histogramm in Abbildung ?? als auch in der Tabelle 8 zu sehen ist, beträgt die maximale Abweichung des oberen Jitters über 15 Promille, was auf einen höheren Jitter als beim XMC1100 zurückzuführen ist. Daher ist die Baudrate für den XMC14104 nicht geeignet und muss geringer gewählt werden.

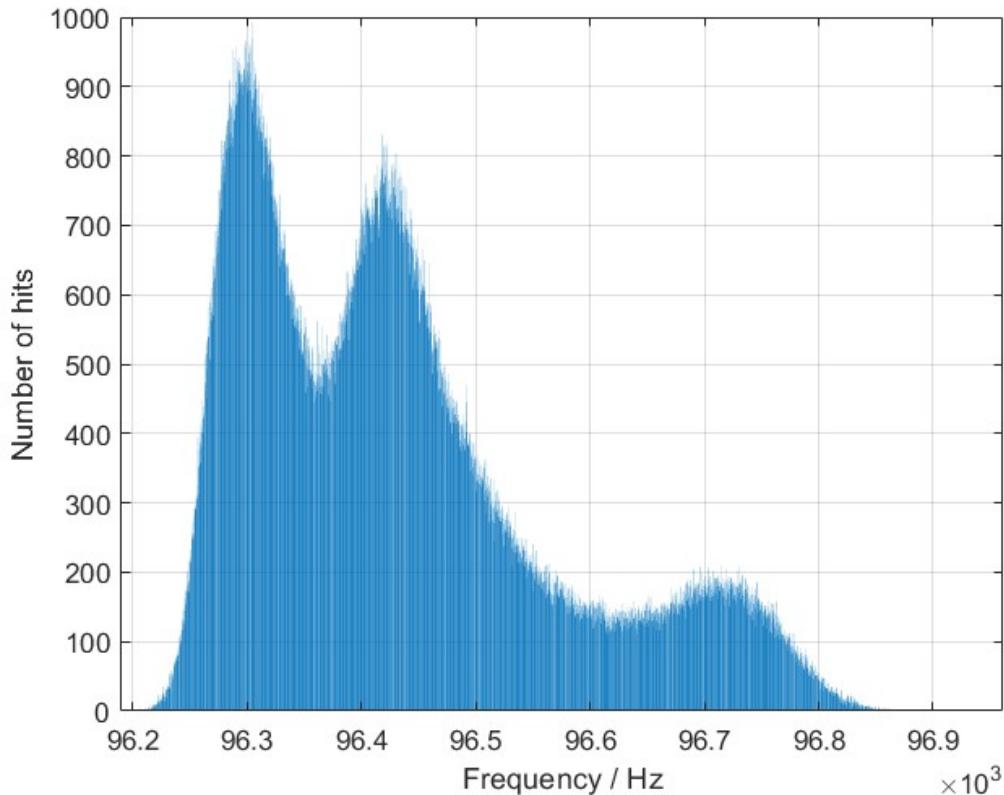


Abbildung 29: Jitter des Mikrocontroller XMC1404 bei einer Baudrate von 192 kbit/s

### 5.6.2. Messung zur Untersuchung der ADC-Verarbeitungszeit

In Abschnitt 5.4 beschrieben, wurde die Messung der ADC-Verarbeitungszeit durchgeführt. Dabei wurden mehrere wie in Abbildung 30 zu betrachtende Messungen mit dem Oszilloskop im Persistenz-Modus durchgeführt.

## 5. Experimentalaufbau

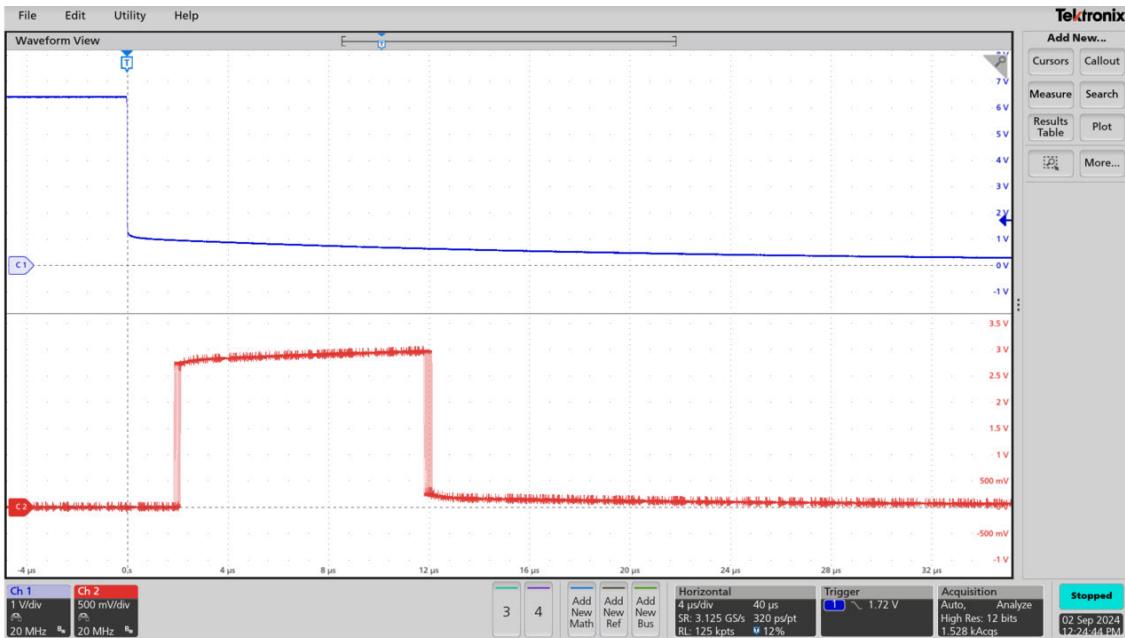


Abbildung 30: Messung der ADC-Verarbeitungszeit mit 8 Eingängen

Die Tabelle 9 zeigt die gemessenen Gesamtverarbeitungszeiten für verschiedene Anzahlen von ADC-Kanälen. Diese Zeiten beinhalten die Verarbeitungszeit pro Kanal sowie eine zusätzliche Verzögerung, die durch das Toggling von GPIO-Pins verursacht wird.

Tabelle 9: Gemessene Gesamtverarbeitungszeiten für unterschiedliche Kanalanzahlen

Anzahl der Kanäle	Messzeit (ns)
8	9968
4	6955
2	5260
1	4506

- **Berechnung der Verarbeitungszeit pro Kanal**

Die Verarbeitungszeit pro Kanal  $t_{Kanal}$  kann aus den Differenzen der gemessenen Zeiten bei unterschiedlichen Kanalanzahlen berechnet werden, indem die Messungen jeweils von der 8-Kanal-Zeit subtrahiert werden:

## 5. Experimentalaufbau

---

$$t_{Kanal,4} = \frac{9968 - 6955}{8 - 4} = \frac{3013}{4} = 753,25 \text{ ns}, \quad (5.1)$$

$$t_{Kanal,2} = \frac{6955 - 5260}{4 - 2} = \frac{1695}{2} = 847,5 \text{ ns}, \quad (5.2)$$

$$t_{Kanal,1} = \frac{5260 - 4506}{2 - 1} = 754 \text{ ns}. \quad (5.3)$$

- **Vergleich der berechneten Kanalverarbeitungszeiten**

Die Tabelle 10 zeigt die berechneten Verarbeitungszeiten pro Kanal für die verschiedenen Messungen im Vergleich.

Tabelle 10: Berechnete Verarbeitungszeiten pro Kanal durch Subtraktion der Messwerte

Kanalanzahl reduziert	Berechnete Kanalzeit (ns)
8 Kanäle auf 4 Kanäle	753,25
4 Kanäle auf 2 Kanäle	847,5
2 Kanäle auf 1 Kanal	754

- **Ergebnisse und Validierung:**

Die berechneten Kanalverarbeitungszeiten variieren leicht in Abhängigkeit davon, welche Messungen herangezogen werden. Dies weist auf mögliche Einflüsse von Messfehlern, zusätzlichen Verzögerungen oder Ungenauigkeiten bei der Messdatenerfassung hin. Die Ergebnisse sind jedoch konsistent genug, um  $t_{Kanal}$  in einem Bereich von ca. 750 ns bis 850 ns zu verorten, was eine plausible Näherung für die Verarbeitungszeit pro Kanal darstellt.

## 5.7. Funktionstest

Der Funktionstest, welcher in Abschnitt 5.4 beschrieben wurde, konnte aufgrund von technischen Schwierigkeiten mit dem JTAG-Interface noch nicht erfolgreich durchgeführt werden. Es war nicht möglich, die beiden Mikrocontroller der Hardware gleichzeitig in der Entwicklungsumgebung mit einem PC zu betreiben.

## **6. Auswertung und Bewertung**

In diesem Kapitel werden die Ergebnisse der Arbeit zusammengefasst und kritisch bewertet. Die wichtigsten Befunde werden analysiert und deren praktische Relevanz im technischen Kontext diskutiert. Abschließend erfolgt ein Ausblick auf mögliche Weiterentwicklungen und Anwendungen der gewonnenen Erkenntnisse.

### **6.1. Synchronisation und Synchronisationsfehler**

Die Synchronisation von Messsystemen spielt eine entscheidende Rolle, insbesondere wenn präzise Messungen durchgeführt werden müssen, wie es bei der elektrochemischen Impedanzspektroskopie (EIS) der Fall ist. Ein wesentlicher Faktor für die Genauigkeit der Messergebnisse ist die exakte zeitliche Abstimmung zwischen den einzelnen Messkomponenten, wie z.B. Strom- und Spannungsaufnahmen. Ein Synchronisationsfehler kann hierbei zu einer Phasenverschiebung führen, die die Messwerte verfälscht und somit die Auswertung und Interpretation der Ergebnisse negativ beeinflusst. Die Auswirkungen von Verzögerungszeiten und Jitter auf die EIS sind daher limitierende Faktoren, die die Abtastfrequenz und die Zuverlässigkeit der Messungen einschränken können.

Das System soll insbesondere zur Ermittlung der EIS eingesetzt werden. Hierzu sollen folgende Hinweise festgehalten werden. Die Auswirkung der Verzögerungszeit und des Jitters ist für die EIS ein limitierender Faktor, der die Abtastfrequenz der Spannungsantwort einschränkt. Es darf sich in einem Block von Abtastwerten keine falsche Zuordnung ergeben. Für die Batterieanalyse sind Anrege-Frequenzen von bis zu 1 kHz relevant. Im Forschungsprojekt sollen beispielsweise Blöcke von 1024 Abtastwerten mit einer Rate von 10 kS/s abgetastet werden. Im Extremfall darf der Fehler, der sich bei der ADC-Synchronisierung ergibt, unter 0,5 Promille liegen. Dies entspricht einem maximalen Fehler von 50 µs. Bei höheren Abtastraten sinkt dieser maximale Fehler proportional. Bei längeren Blocklängen sinkt der zulässige Fehler umgekehrt zur Blocklänge [26].

#### **6.1.1. Berechnung der Verzögerungszeiten**

Für die Synchronisation der Messsysteme wurden zwei verschiedene Übertragungskomponenten untersucht: der Digitalisolator (ISOFACE™ 2DIB141xF von Infineon) und der Infrarot-Transceiver (TFDU4101 von Vishay). Beide Systeme weisen unterschiedliche Verzögerungszeiten und Übertragungscharakteristiken auf, die im Folgenden aufgelistet sind. Dabei ist zu beachten, dass die angegebenen Zeiten die wichtigsten Einflussgrößen darstellen und eine gute Näherung liefern. Es handelt sich um Maximalwerte, nicht um typische Zeiten. Beispielsweise fehlen die Laufzeiten,

## **6. Auswertung und Bewertung**

---

da diese bei den kurzen Strecken vernachlässigt und maßgeblich durch die Latenzen der Bauteile bestimmt werden können. Ebenfalls nicht berücksichtigt wurde die Zeit zur Auslösung des Interrupts, die jedoch bei schnellen Systemen potenziell relevant sein könnte, aber im Rahmen dieser Untersuchung nicht ermittelt wurde.

- **Verzögerungen beim Digitalisolator:**

- 10-Bit-UART-Frame:  $\frac{5}{1,25 \text{ Mbit/s}} = 4 \mu\text{s}$
- UART-Jitter:  $4 \mu\text{s} \cdot 2,05\% = 8,2 \text{ ns}$
- ADC-Verarbeitungszeit für 8 Kanäle:  $8 \cdot 850 \text{ ns} = 6,8 \mu\text{s}$
- Übertragungszeit (Latenz):  $\approx 26 \text{ ns}$  (Datenblattangabe [1])

**Gesamtverzögerung:**  $10,83 \mu\text{s}$

- **Verzögerungen beim Infrarot-Transceiver:**

- 10-Bit-UART-Frame:  $\frac{5}{115,2 \text{ kbit/s}} = 43,4 \mu\text{s}$
- ADC-Verarbeitungszeit für 8 Kanäle:  $8 \cdot 850 \text{ ns} = 6,8 \mu\text{s}$
- UART-Jitter:  $43,4 \mu\text{s} \cdot -1,98\% = -85,9 \text{ ns}$
- Übertragungszeit (Latenz):  $\approx 400 \text{ ns}$  (Oszilloskop-Messung)

**Gesamtverzögerung:**  $50,51 \mu\text{s}$

Während der in Kapitel 5.6 durchgeführten Messungen stellte sich heraus, dass die ursprüngliche Baudrate von 192 kbit/s beim Infrarot-Transceiver zu hoch war, um zuverlässige Ergebnisse zu erzielen. Daher wurde eine Baudrate von 115.2 kbit/s angenommen. Der zusätzlich zu erwarten- de Jitter wurde anhand von Messungen bei 192 kbit/s abgeschätzt, da die Größenordnung dem unteren Jitter prozentual entsprechen sollte.

### **6.1.2. Einfluss der Verzögerungszeiten auf die Phasenverschiebung**

Die Verzögerungszeiten der Systeme führen zu Phasenverschiebungen, die stark von der Frequenz der Signale abhängen. Die Phasenverschiebung  $\phi$  kann allgemein durch folgende Gleichung 6.1 bestimmt werden:

$$\phi = \frac{\Delta t}{T} \cdot 360^\circ = \Delta t \cdot f \cdot 360^\circ \quad (6.1)$$

## 6. Auswertung und Bewertung

wobei:

- $\Delta t$  die Verzögerungszeit in Sekunden,
- $T$  die Periodendauer des Signals in Sekunden ( $T = \frac{1}{f}$ ),
- $f$  die Frequenz des Signals in Hz ist.

Die Phasenverschiebung für die beiden Systeme wird in der folgenden Tabelle 11 für Frequenzen von 1 mHz bis 1 kHz berechnet.

Tabelle 11: Phasenverschiebung für verschiedene Frequenzen

Frequenz (Hz)	Digitalisator ( $\Delta t = 10,83 \mu s$ )	Infrarot-Transceiver ( $\Delta t = 50,51 \mu s$ )
1 mHz	0,000004°	0,000018°
5 mHz	0,000019°	0,000091°
10 mHz	0,000039°	0,000182°
50 mHz	0,000195°	0,000909°
100 mHz	0,000390°	0,001818°
500 mHz	0,001949°	0,009092°
1 Hz	0,003899°	0,018184°
5 Hz	0,019494°	0,090918°
10 Hz	0,038988°	0,181836°
50 Hz	0,194940°	0,909180°
100 Hz	0,389880°	1,818360°
500 Hz	1,949400°	9,091800°
1 kHz	3,898800°	18,183600°

### 6.1.3. Bewertung der Synchronisation

Die Ergebnisse zeigen deutlich, dass die Phasenverschiebung durch die Verzögerungszeiten der verwendeten Übertragungskomponenten stark frequenzabhängig ist. Der Digitalisator weist mit einer Verzögerungszeit von  $10.8 \mu s$  eine geringe Phasenverschiebung auf, die selbst bei Frequenzen von bis zu 1 kHz im akzeptablen Bereich bleibt. Dies zeigt, dass der Digitalisator für präzise EIS-Messungen geeignet ist und den Anforderungen an die Synchronisation gerecht wird.

Der Infrarot-Transceiver hingegen erreicht bei einer Verzögerung von  $50.59 \mu s$  eine deutlich höhere Phasenverschiebung, die bei höheren Frequenzen wie 1 kHz zu Problemen führt. Insbesondere in kritischen Anwendungsfällen, bei denen ein maximaler Synchronisationsfehler von  $50 \mu s$  gefordert wird, stößt der Infrarot-Transceiver an seine Grenzen. Daher ist der Infrarot-Transceiver nur eingeschränkt für die EIS-Messungen geeignet und sollte nur in Anwendungen eingesetzt werden, in denen die Phasengenauigkeit weniger kritisch ist.

## **6. Auswertung und Bewertung**

---

Die Analyse verdeutlicht damit, dass der Digitalisolator die Anforderungen an die Synchronisation besser als der Infrarot-Transceiver erfüllt und somit für präzise EIS-Messungen die bevorzugte Option darstellt.

### **6.2. Ausblick**

In diesem Abschnitt werden die wichtigsten Beobachtungen und Erfahrungen aus den durchgeführten Untersuchungen zusammengefasst und mögliche Weiterentwicklungen sowie zukünftige Optimierungsschritte aufgezeigt.

- Einsatz von Vorwiderständen zur Strombegrenzung:**

Im Abschnitt 4 wurden verschiedene Vorwiderstände zur Strombegrenzung der IrDA-Schnittstelle getestet. Diese Maßnahmen haben gezeigt, dass der Strombedarf der Infrarot-Kommunikation erheblich reduziert werden kann und dies direkt zur Senkung des Energieverbrauchs der Sensorik beiträgt. Der Einsatz von hohen Vorwiderständen hingegen führt auch zu einem signifikanten Anstieg des Jitter bei der Datenübertragung. Zur Herstellung einer Balance zwischen Energieeffizienz und Übertragungsstabilität muss der Widerstand so dimensiniert werden, dass die zulässige Fehlerhäufigkeit nicht überschritten wird. Folglich sollte der Stromfluss durch softwareseitig einstellbare Parameter im Mikrocontroller den jeweiligen Bedingungen angepasst werden. So kann der Strom für jedes Sensormodul individuell geregelt werden, um ein optimales Verhältnis zwischen Übertragungsfehler und Stromverbrauch zu erzielen.

- Optimierung der Baudrate des Infrarot-Transceivers:**

Der Infrarot-Transceiver hat sich aufgrund seiner robusten drahtlosen Verbindung und der Vielseitigkeit als bevorzugte Wahl für die Datenübertragung im ProMoBiS-Projekt herausgestellt. Allerdings könnte die aktuell verwendete Baudrate von 115.2 kBit/s bei höheren Messfrequenzen, insbesondere im Bereich der elektrochemischen Impedanzspektroskopie (z.B. bei 1 kHz), an die Grenze der Synchronität stoßen. In Kapitel 4 wurde gezeigt, dass der Transceiver stabil bei Baudaten bis zu 200 kBit/s betrieben werden kann. Durch eine Optimierung der Mikrocontroller-Taktung, beispielsweise durch den Einsatz von stabileren Taktquellen, könnte der Synchronitätsbereich verbessert werden.

- Erweiterte Funktionsprüfung und Verifizierung:**

Eine abschließende Funktionsprüfung der entwickelten Systeme konnte aufgrund technischer Probleme mit dem JTAG-Interface nicht vollständig durchgeführt werden. Um die

## *6. Auswertung und Bewertung*

---

Zuverlässigkeit und die Eignung des Systems für die angestrebten Anwendungsbereiche sicherzustellen, werden folgende Prüfungen vorgeschlagen:

- **Prüfung der Baudratenstabilität:** Hinsichtlich der Sicherstellung der Stabilität der Datenübertragung, sollten umfangreiche Tests zur Bitfehlerrate (bit error rate) durchgeführt werden. Diese Tests sollten unter realen Bedingungen erfolgen, um die Auswirkungen von Jitter und Synchronisationsabweichungen besser zu verstehen.
- **Validierung der Synchronität:** Zur praktischen Bestimmung der Synchronisationsgenauigkeit ist ein experimenteller Aufbau, siehe Kapitel 5 vorgesehen, bei dem eine bekannte Testspannung von einem Funktionsgenerator eingespeist wird. Die Messergebnisse sollen dann mit den erwarteten Werten verglichen werden, um die Synchronität des Systems zu überprüfen.

Die vorgestellten Maßnahmen und Optimierungsvorschläge bieten eine solide Grundlage für zukünftige Arbeiten und ermöglichen eine kontinuierliche Verbesserung des Systems im Hinblick auf Effizienz, Genauigkeit und Zuverlässigkeit.

## Literaturverzeichnis

- [1] Infineon Technologies AG. "ISOFACE™ digital isolators 2DIBx41xF". Technisches Datenblatt. Dokumentennummer: IFX-ori1669813517077. 2023. URL: [https://www.infineon.com/dgdl/Infineon-2DIBx41xF-DS\\_v01\\_00-EN.pdf?DataSheet-v01\\_00-EN.pdf?fileId=8ac78c8c872bd8d60187370abe7d246c](https://www.infineon.com/dgdl/Infineon-2DIBx41xF-DS_v01_00-EN.pdf?DataSheet-v01_00-EN.pdf?fileId=8ac78c8c872bd8d60187370abe7d246c) (abgerufen am 06.05.2024).
- [2] Infineon Technologies AG. "XMC1100 AB-Step". Technisches Datenblatt. Version: V1.8, 2016-09. 2016. URL: [https://www.infineon.com/dgdl/Infineon-xmc1100\\_AB-DS-v01\\_08-EN.pdf?fileId=5546d4624a0bf290014a4bdaff9325bd](https://www.infineon.com/dgdl/Infineon-xmc1100_AB-DS-v01_08-EN.pdf?fileId=5546d4624a0bf290014a4bdaff9325bd) (abgerufen am 06.05.2024).
- [3] Infineon Technologies AG. "XMC1404 AA-Step". Technisches Datenblatt. Version: V1.5 2024-04. 2024. URL: [https://www.infineon.com/dgdl/Infineon-XMC1400-DataSheet-v01\\_05-EN.pdf?fileId=5546d46250cc1fdf015110a2596343b2](https://www.infineon.com/dgdl/Infineon-XMC1400-DataSheet-v01_05-EN.pdf?fileId=5546d46250cc1fdf015110a2596343b2) (abgerufen am 06.05.2024).
- [4] Kohs Alexander. "Batteriemodell zur Prädiktion des Gesundheitszustands von Lithium-Ionen-Batterien". Universität Stuttgart: Springer Vieweg, 2021. ISBN: 978-3-658-37679-6.
- [5] Francisco J. López-Hernández Asunción Santamaría. "Wireless LAN Standards and Applications". Boston, London: ARTECH HOUSE, INC., 2001. ISBN: 0-89006-943-3.
- [6] Deutsche Bundesregierung. "Nicht weniger fortbewegen, sondern anders". 2024. URL: <https://www.bundesregierung.de/breg-de/schwerpunkte/klimaschutz/nachhaltige-mobilitaet-2044132> (abgerufen am 06.05.2024).
- [7] Claus Daniel and Jürgen O. Besenhard. "Handbook of Battery Materials". John Wiley & Sons, 2012. ISBN: 978-3527326954.
- [8] Christoph Degen. "Inductive coupling for wireless power transfer and near-field communication". In: 2021.1 (2021, EURASIP Journal on Wireless Communications and Networking), p. 121.
- [9] Jonas Ernsting. "Funktionsdemonstrator für die optische Messdatenübertragung in Fahrzeuggittern". Bachelorthesis. HAW, 2020. URL: <http://hdl.handle.net/20.500.12738/14241> (abgerufen am 01.05.2024).
- [10] Birger Fricke et al. "Ortsaufgeloste Impedanzspektroskopie an Brennstoffzellen". In: 99.1/2 (2006, Technische Mitteilungen-Essen-), p. 231.
- [11] Emilio García, Eduardo Quiles, and Antonio Correcher. "Distributed intelligent battery management system using a real-world cloud computing system". In: 23, Sensors.7 (2023), p. 3417.

## Literaturverzeichnis

---

- [12] Johnnie Hancock et al. "Jitter—understanding it, measuring it, eliminating it part 1: Jitter fundamentals". In: 4.4 (2004, High Frequency Electronics), pp. 44–50.
- [13] Mohammad A Hannan et al. "A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations". In: Renewable and Sustainable Energy Reviews, 78 (2017), pp. 834–854.
- [14] "Hochvoltbatterien". 2024. URL: <https://de.wikipedia.org/wiki/Hochvolt> (abgerufen am 12.09.2024).
- [15] "Infrared Data Association". 2024. URL: [https://de.wikipedia.org/wiki/Infrared\\_Data\\_Association](https://de.wikipedia.org/wiki/Infrared_Data_Association) (abgerufen am 02.09.2024).
- [16] Projektträger Jülich/BMWK. "Verbundvorhaben: ProMoBiS". 2024. URL: [https://www.enargus.de/pub/bscw.cgi/?op=enargus.eps2&m=1&v=10&p=2017&psid=1917885\\_0&id=18245409](https://www.enargus.de/pub/bscw.cgi/?op=enargus.eps2&m=1&v=10&p=2017&psid=1917885_0&id=18245409) (abgerufen am 20.08.2024).
- [17] Peter Keil and Andreas Jossen. "Aufbau und parametrierung von batteriemodellen". In: 2012, 19. design&elektronik-Entwicklerforum Batterien & Ladekonzepte.
- [18] Peter Keil and Andreas Jossen. "Aufbau und parametrierung von batteriemodellen". In: 2012, 19. DESIGN&ELEKTRONIK-Entwicklerforum Batterien & Ladekonzepte.
- [19] Lidiya Komsiyska et al. "Critical review of intelligent battery systems: Challenges, implementation, and potential for electric vehicles". In: Energies, 14.18 (2021), p. 5989.
- [20] Reiner Korthauer. "Handbuch lithium-ionen-batterien". Springer, 2013.
- [21] Thomas F Landinger et al. "Power line communications in automotive traction batteries: a proof of concept". In: IEEE. 2020 IEEE International Symposium on Power Line Communications and its Applications (ISPLC), pp. 1–5.
- [22] Michael Leitl. "Strukturchemische und impedanzspektroskopische Untersuchungen an silberionenleitenden Substanzen, Münzmetallthiophosphaten und Kupferargyroditen". PhD thesis. 2008.
- [23] Aining Li. "Vorlesungsunterlage, Kapitel 2: Leitungstheorie". Unveröffentlichte Vorlesungsunterlage. 2023.
- [24] Fabian Mahler. "Optische Datenkommunikation im Batteriemodul von Elektrofahrzeugen". Bachelorthesis. HAW, 2024. URL: <https://reposit.haw-hamburg.de/handle/20.500.12738/15689> (abgerufen am 17.05.2024).
- [25] Oliver Opalko et al. "Physical layer and multi-carrier analysis for power line communication networks in Li-ion batteries for electric and hybrid vehicles". In: IEEE. 2015 IEEE Vehicular Networking Conference (VNC), pp. 243–250.

## Literaturverzeichnis

---

- [26] Forschungsprojekt ProMoBis. "Progressive Multizell-Verbund-Konzepte für Batteriesysteme mit integrierter Sensorik". In: vol. Arbeitsgruppe. 2024.
- [27] Egidio Ragonese et al. "Fully integrated galvanically isolated dc-dc converters based on inductive coupling". In: Springer. 2019, Applications in Electronics Pervading Industry, Environment and Society: APPLEPIES 2018 6, pp. 335–341.
- [28] Florian Rittweger et al. "Sensor Data Communication via Light Guide Body for Monitoring Vehicle Batteries". In: 2022 IEEE Sensors Applications Symposium (SAS), pp. 1–6. DOI: [10.1109/SAS54819.2022.9881250](https://doi.org/10.1109/SAS54819.2022.9881250).
- [29] Valentin Roscher et al. "Synchronisation using wireless trigger-broadcast for impedance spectroscopy of battery cells in 2015 IEEE Sensors Applications Symposium (SAS)". In: 2015, pp. 1–6. DOI: [10.1109/SAS.2015.7133608](https://doi.org/10.1109/SAS.2015.7133608).
- [30] Nico Sassano. "Hard- und Softwareentwicklung für einen drahtlos kommunizierenden Batterie-Zellensor mit funksynchronisierter Messung". Bachelorthesis. HAW, 2013. URL: <http://hdl.handle.net/20.500.12738/6310> (abgerufen am 17.05.2024).
- [31] Nico Sassano, Valentin Roscher, and Karl-Ragmar Riemschneider. "Batterie-Zellensensoren mit drahtloser Kommunikation und verteilter Signalverarbeitung". In: (2016, Automobil-Sensorik: Ausgewählte Sensorprinzipien und deren automobile Anwendung), pp. 45–63.
- [32] Dirk Uwe Sauer. "Grundlagen der Impedanzspektroskopie für die Charakterisierung von Batterien". In: 99.1 (2006, Technische Mitteilungen), pp. 74–80.
- [33] Jan Philipp Schmidt and Thomas Hammerschmidt. "Impedanzsensorik für Batteriezellen in Elektro-Fahrzeugen". In: ed. by Thomas Tille. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018, Automobil-Sensorik 2: Systeme, Technologien und Applikationen, pp. 99–126. ISBN: 978-3-662-56310-6. DOI: [10.1007/978-3-662-56310-6\\_5](https://doi.org/10.1007/978-3-662-56310-6_5). URL: [https://doi.org/10.1007/978-3-662-56310-6\\_5](https://doi.org/10.1007/978-3-662-56310-6_5).
- [34] Matthias Schneider et al. "Automotive battery monitoring by wireless cell sensors". In: IEEE. 2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings, pp. 816–820.
- [35] Gerhard Schnell and Bernhard Wiedemann. "Bussysteme in der Automatisierungs- und Prozesstechnik: Grundlagen, Systeme und Anwendungen der industriellen Kommunikation". 3. Auflage. Berlin, Heidelberg: Springer Vieweg, 2018. ISBN: 978-3-662-55144-6. DOI: [10.1007/978-3-662-55145-3](https://doi.org/10.1007/978-3-662-55145-3).
- [36] Vishay Semiconductors. "TFDU4101 Infrared Transceiver Module (SIR, 115.2 kbit/s) for IrDA® Applications". Technisches Datenblatt. Dokumentennummer: 81288. 2023. URL: [https://www.infineon.com/dgdl/Infineon-xmc1100\\_AB-DS-v01](https://www.infineon.com/dgdl/Infineon-xmc1100_AB-DS-v01)

## Literaturverzeichnis

---

- [\\_08-EN.pdf?fileId=5546d4624a0bf290014a4bdaff9325bd](#) (abgerufen am 06.05.2024).
- [37] Haosen Shi et al. "Effect of timing jitter on time-of-flight distance measurements using dual femtosecond lasers". In: 23.11 (2015, Optics express), pp. 14057–14069.
  - [38] Tektronix. "Understanding and Characterizing Timing Jitter". Accessed: 2024-09-23. 2010. URL: [https://download.tek.com/document/Understanding-and-Characterizing-Timing-Jitter\\_55W\\_16146\\_6.pdf](https://download.tek.com/document/Understanding-and-Characterizing-Timing-Jitter_55W_16146_6.pdf).
  - [39] Wladislaw Waag, Christian Fleischer, and Dirk Uwe Sauer. "Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles". In: Journal of Power Sources, 258 (2014), pp. 321–339.
  - [40] Klaus Wippermann et al. "Grundlagen zur Impedanzspektroskopie an Brennstoffzellen und ortsaufgeloste Messungen". In: 99.1/2 (2006, Technische Mitteilungen-Essen-), p. 96.

## A. Anhang

### A.1. Kapitel: Voruntersuchungen

#### A.1.1. C-Quellcode

```
1  /*
2   * main.c
3   *
4   * Created on:      2024 Jun 17 21:20:00
5   * Author:          Stefan Ralfs
6   * Project:         Preliminary investigations of IrDA and ISOFACE
7   * Description:    This file contains basic functions for controlling
8   *                  and testing certain components that are intended
9   *                  for UART communication.
10  * Revision History:
11  * - 17 Jun 2024: Initial release (Stefan Ralfs)
12  */
13
14 // Declarations from DAVE Code Generation (includes SFR declaration)
15 #include "DAVE.h"
16 // IrDA specific functions
17 #include "src/IRDA/TFDU4101.h"
18
19 /* Function prototypes */
20 void SYSTIMER_usDelay(uint32_t delay_us);
21 void SYSTIMER_msDelay(uint32_t delay_ms);
22 void initializeGPIO(void);
23 uint8_t generate_prbs8(void);
24 uint8_t lfsr = 0xFF; // Starting value for PRBS (Seed)
25
26 /* OPTION A (Choose either IRDA OR ISOFACE) !
27  * Selection of the component that will be
28  * used for data transfer. */
29 #define IRDA_USED // Infrared data transmission
30 // #define ISOFACE_USED // Digital isolator data transmission
31
32 /* OPTION A (Select one or none option) !
33  * Selection of whether and which data
34  * the UART interface should transmit. */
35 #define TRANSMITT_PRBS // PRBS-DATA
```

## A. Anhang

---

```
36 // #define TRANSMITT_BYTE // Specific Byte
37
38 // Length of Data in Bytes
39 #define DATA_LENGTH 1
40
41 uint8_t ReadData_UART0[DATA_LENGTH]; // Data-Buffer IrDA
42 uint8_t ReadData_UART1[DATA_LENGTH]; // Data-Buffer ISOFACE
43 /* If TRANSMITT_BYTE is selected, the initialization value
44 * of data_buffer is the permanently transmitted byte. */
45 uint8_t data_buffer = 0x55;
46 uint32_t timer_1; // Variable for timer initialization
47 volatile bool sendFlag = false; // Flag for sending data
48
49 // Callback-Function triggered by IrDA
50 void UART0_RX(void) {
51     // For testing purposes only
52 }
53
54 // Callback-Function triggered by ISOFACE
55 void UART1_RX(void) {
56     // For testing purposes only
57 }
58
59 void SYSTIMER_ISR_CALLBACK(void *param)
60 {
61     // set flag to send data in main
62     sendFlag=true;
63 }
64
65 int main(void)
66 {
67     DAVE_STATUS_t status;
68
69     /* Initialization of DAVE APPS */
70     status = DAVE_Init();
71
72     if (status != DAVE_STATUS_SUCCESS)
73     {
74         /* Error handler. */
75         XMC_DEBUG("DAVE APPs initialization failed\n");
76
77         while(1U)
```

## A. Anhang

---

```
78         {
79
80     }
81 }
82
83 /* Initialization of GPIO-Ports */
84 initializeGPIO();
85
86 /* Initialization of Systimer for Interrupt */
87 timer_1 = SYSTIMER_CreateTimer(SYSTIMER_TICK_PERIOD_US,
88                                 SYSTIMER_MODE_PERIODIC, SYSTIMER_ISR_CALLBACK, NULL);
89 SYSTIMER_StartTimer(timer_1);
90
91 /* Initialization of IrDA specific settings */
92 #ifdef IRDA_USED
93     initializeIRDA();
94 #endif
95
96 /* Initialization of transmit buffer */
97 #ifdef TRANSMITT_PRBS
98     data_buffer = generate_prbs8();
99 #endif
100    uint8_t *data_ptr = &data_buffer;
101
102 /* Main application code. */
103 while(1U)
104 {
105     #if defined(TRANSMITT_PRBS) || defined(TRANSMITT_BYTE)
106         /* Wait for sending data until the flag is set*/
107         if(sendFlag==true) {
108             sendFlag=false; // reset flag
109
110             #ifdef IRDA_USED
111                 // Transmit data with IRDA
112                 transmitTFDU4101(data_ptr, sizeof(data_buffer));
113
114                 /* set a trigger pulse after sending. */
115                 SYSTIMER_usDelay(20); // Delay for better screen
116                         pictures
117                 DIGITAL_IO_SetOutputHigh(&TEST_0);
118                 DIGITAL_IO_SetOutputLow(&TEST_0);
```

## A. Anhang

---

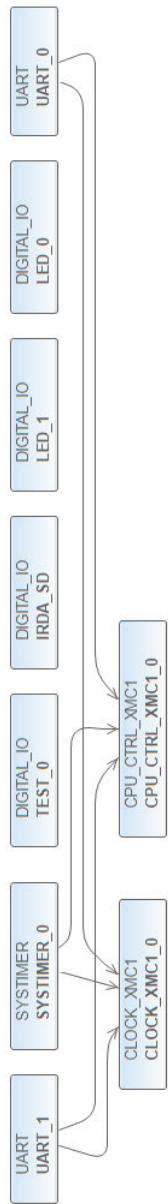
```
118         #endif
119         #ifdef ISOFACE_USED
120             // Transmit data with ISOFACE
121             while (UART_Transmit(&UART_1, data_ptr, sizeof(  
122                 data_buffer)) != UART_STATUS_SUCCESS);
123             /* set a trigger pulse after sending. */
124             DIGITAL_IO_SetOutputHigh(&TEST_0);
125             DIGITAL_IO_SetOutputLow(&TEST_0);
126         #endif
127
128         /* generate new PRBS-Data if selected. */
129         #ifdef TRANSMITT_PRBS
130             data_buffer = generate_prbs8();
131         #endif
132     }
133 }
134 }
135
136 /* Functions. */
137
138 void initializeGPIO(void) {
139     uint8_t i = 0;
140
141     /* Blink Led */
142     for (i = 0; i < 8; ++i) {
143         SYSTIMER_msDelay(50);
144         DIGITAL_IO_ToggleOutput(&LED_0);
145         DIGITAL_IO_ToggleOutput(&LED_1);
146     }
147     /* DIGITAL_IO Pins */
148     DIGITAL_IO_SetOutputLow(&LED_0);
149     DIGITAL_IO_SetOutputLow(&LED_1);
150     DIGITAL_IO_SetOutputLow(&TEST_0);
151     /* Short wait before starting*/
152     SYSTIMER_msDelay(50);
153 }
154
155 /* Generation of 8-bit pseudorandom data. */
156 uint8_t generate_prbs8() {
157     uint8_t bit = ((lfsr >> 7) ^ (lfsr >> 5) ^ (lfsr >> 4) ^ (lfsr >>  
158         3)) & 1;
```

## A. Anhang

---

```
158     lfsr = (lfsr >> 1) | (bit << 7);
159     return lfsr;
160 }
161
162 /* Time delay for an input with microseconds. */
163 void SYSTIMER_usDelay(uint32_t delay_us)
164 {
165     uint32_t start_tick = SYSTIMER_GetTickCount();
166     uint32_t delay_ticks = delay_us / SYSTIMER_TICK_PERIOD_US;
167
168     while ((SYSTIMER_GetTickCount() - start_tick) < delay_ticks)
169     {
170         // Busy wait
171     }
172 }
173
174 /* Time delay for an input with milliseconds. */
175 void SYSTIMER_msDelay(uint32_t delay_ms)
176 {
177     uint32_t start_tick = SYSTIMER_GetTickCount();
178     uint32_t delay_ticks = delay_ms * 1000 / SYSTIMER_TICK_PERIOD_US;
179
180     while ((SYSTIMER_GetTickCount() - start_tick) < delay_ticks)
181     {
182         // Busy wait
183     }
184 }
```

### A.1.2. Abhängigkeiten der DAVE-Apps



## A.2. Hardwarespezifisch

### A.2.1. Infrarot-Tranciver Header

## A. Anhang

---

```
1  /*
2   *  TFDU4101.h
3   *
4   *  Created on: 11 Oct 2023
5   *  Author: Tobias Frahm
6   *
7   *  Revision History:
8   *  - 11 Oct 2023: Initial release (Tobias Frahm)
9   */
10
11
12 #ifndef SRC_IRDA_TFDU4101_H_
13 #define SRC_IRDA_TFDU4101_H_
14
15 #include <stdint.h>
16
17 void initializeIRDA(void);
18 void toggleTFDH4101(void);
19 void enableSendTFDH4101(void);
20 void enableReceiveTFDH4101(void);
21 bool receiveTFDU4101(uint8_t* data_ptr, uint32_t count);
22 void transmitTFDU4101(uint8_t* data_ptr, uint32_t count);
23
24 #endif /* SRC_IRDA_TFDU4101_H_ */
```

### A.2.2. Infrarot-Tranciver Quellcode

```
1  /*
2   *  TFDU4101.c
3   *
4   *  Created on: 11 Oct 2023
5   *  Author: Tobias Frahm
6   *
7   *  Revision History:
8   *  - 11 Oct 2023: Initial release (Tobias Frahm)
9   *  - 17 Jun 2024: Adjusting of initializeIRDA(void) (Stefan Ralfs)
10  */
11
12 #include <stdbool.h>
13 #include "src/IRDA/TFDU4101.h"
14 #include "Dave/Generated/DIGITAL_IO/digital_io.h"
15 #include "Dave/Generated/UART/uart.h"
```

## A. Anhang

---

```
16
17 /**
18 * toggles shutdown mode
19 */
20 void toggleShutdownTFDU4101(void) {
21     DIGITAL_IO_ToggleOutput(&IRDA_SD);
22 }
23
24 /**
25 * Enables shutdown mode
26 */
27 void enableShutdownTFDU4101(void) {
28     DIGITAL_IO_SetOutputHigh(&IRDA_SD);
29 }
30
31 /**
32 * Enables shutdown mode
33 */
34 void disableShutdownTFDU4101(void) {
35     DIGITAL_IO_SetOutputLow(&IRDA_SD);
36 }
37
38 /**
39 * MUST BE CALLED AFTER DAVE INIT
40 * @brief Additional Configuration needed for IDRA-Communication
41 * It configures the TX pulse length to 3/16 of a bit length
42 * and sets the RX sampling point to 2/16 of a bit length
43 */
44 void initializeIRDA(void) {
45
46     // set sample mode to one sample per bit and the sample point to
47     // 2/16 of the bit length (after time quanta 1)
48     UART_0.channel->PCR_ASCMode = (uint32_t) (((UART_0.channel->
49         PCR_ASCMode & (~USIC_CH_PCR_ASCMode_SMD_Msk)
50         & (~USIC_CH_PCR_ASCMode_SP_Msk)) | (((uint32_t) 0UL) <<
51             USIC_CH_PCR_ASCMode_SMD_Pos)
52         | (((uint32_t) 1UL) << USIC_CH_PCR_ASCMode_SP_Pos));
53
54     // enable pulse output and set pulse length to 3/16 of the bit
55     // length
56     UART_0.channel->PCR_ASCMode = (uint32_t) ((UART_0.channel->
57         PCR_ASCMode & (~USIC_CH_PCR_ASCMode_PL_Msk))
58         | (((uint32_t) 6UL) << USIC_CH_PCR_ASCMode_PL_Pos));
```

## A. Anhang

---

```
53     disableShutdownTFDU4101();  
54 }  
55  
56 /**  
57 * Receives the given amount of words  
58 * @param data_ptr pointer to receive buffer  
59 * @param count count Total no of words to be received  
60 */  
61  
62 bool receiveTFDU4101(uint8_t* data_ptr, uint32_t count) {  
63     return (UART_Receive(&UART_0, data_ptr, count) !=  
64         UART_STATUS_SUCCESS);  
65  
66 /**  
67 * Transmits the given amount of words  
68 * @param data_ptr pointer to receive buffer  
69 * @param count total number of words to be transmitted  
70 */  
71 void transmitTFDU4101(uint8_t* data_ptr, uint32_t count) {  
72     while (UART_Transmit(&UART_0, data_ptr, count) !=  
73         UART_STATUS_SUCCESS) {  
74         // busy waiting  
75     }  
76 }
```

## **Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit**

Gemäß der Allgemeinen Prüfungs- und Studienordnung ist zusammen mit der Abschlussarbeit eine schriftliche Erklärung abzugeben, in der der Studierende bestätigt, dass die Abschlussarbeit "ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt wurden. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich zu machen."

*Quelle: §16 Abs. 5 APSO-TI-BM bzw. §15 Abs. 6 APSO-INGI*

### **Erklärung zur selbstständigen Bearbeitung der Arbeit**

Hiermit versichere ich,

Name: \_\_\_\_\_

Vorname: \_\_\_\_\_

dass ich die vorliegende Bachelorarbeit mit dem Thema:

**Kommunikation im Batteriemanagementsystem für Elektrofahrzeuge – Hard- und Softwareentwicklung, Bewertung des Zeitverhaltens**

ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

---

---

Ort

Datum

Unterschrift im Original

