

DISSERTATION

Betriebliche, messtechnische Online-Überwachung von Schienenfahrzeugen

ausgeführt zum Zwecke der Erlangung des akademischen Grades
eines Doktors der technischen Wissenschaften
unter der Leitung von

Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Johann Wassermann
Institut für Mechanik und Mechatronik, E325

eingereicht an der Technischen Universität Wien
Fakultät für Maschinenwesen und Betriebswissenschaften

von

Bernhard Groß
9326099
Akaziengasse 48-50/47/13, 1230 Wien

Wien, im September 2014

Kurzfassung

Im Betrieb einer Straßenbahn können, abhängig von der Qualität der Fahrstrecke (Gleiszustand und Zustand des Gleisunterbaus), Beeinflussungen des Fahrkomforts aber auch erhöhte Bauteilbelastungen durch Schwingungsanregungen über das Fahrwerk inklusive dem Antriebssystem auftreten. Zur Analyse solcher im Betrieb festgestellten Komfortbeeinflussungen werden als Stand der Technik spezielle Messfahrzeuge eingesetzt, die das Erregerspektrum entlang der Fahrstrecke aufnehmen und eine Analyse ermöglichen. Nachteil dieser Vorgehensweise ist, dass das Messfahrzeug völlig andere Lastzustände für die Fahrstrecke darstellt und die dabei gemessenen Erregerspektren nur eine Näherung der realen Situation widerspiegeln. Darüber hinaus wird die Problemanalyse dadurch erschwert, dass das Messfahrzeug ein völlig anderes Schwingungsverhalten als das tatsächlich eingesetzte Schienenfahrzeug aufweist.

Das Ziel dieser Dissertation war daher die Entwicklung eines Messsystems, das bei Bedarf im konkret betroffenen Fahrzeug installiert werden kann und somit im normalen Fahrgastbetrieb repräsentative Messergebnisse liefert. Ein wesentliches Kriterium für die Praxistauglichkeit eines solchen Systems stellt der, bei der Installation des Systems, erforderliche Verkabelungsaufwand dar. Im Idealfall sollten die im Fahrzeug verteilten Komponenten des Messsystems autark vom Bordnetz sein (z.B. lokaler Energiespeicher oder Versorgung über elektromagnetische Felder) und die Messdaten drahtlos übertragen. Konkret soll das Messsystem in der Lage sein über eine Dauer von 10 bis 20 h Messdaten von zumindest 26 im gesamten Fahrzeug verteilten Sensoren mit einer Abtastrate von 1 kS/s aufzuzeichnen.

Stand der Technik sind einerseits Oberflächenwellensensoren und passive RFID-Systeme (Radio Frequency Identification), die per Funk ausgelesen werden können und keine zusätzliche Energiequelle benötigen, und andererseits drahtlose Sensornetzwerke und Telemetriesysteme, die Messdaten ebenfalls per Funk übertragen, jedoch in der Regel mittels Batterien oder Akkumulatoren mit Energie versorgt werden. Aus den zu Oberflächenwellensensoren und passiven RFIDs veröffentlichten Leistungsdaten geht hervor, dass beide Systeme unter anderem auf Grund der geringen überbrückbaren Auslesedistanz für einen Einsatz in Schienenfahrzeugen nicht geeignet sind. Bei drahtlosen Sensornetzwerken und Telemetriesystemen besteht wiederum die Problematik, dass diese entweder einen zu geringen Datendurchsatz für Messungen mit Abtastraten im Bereich von 1 kS/s aufweisen

(drahtlose Sensornetzwerke) oder nicht für Messungen mit einer Vielzahl von Sensoren ausgelegt sind (Telemetriesysteme).

Um eine möglichst hohe Flexibilität des Messsystems zu erzielen sieht das, im Rahmen dieser Arbeit entwickelte, Konzept im gesamten Fahrzeug verteilte Messknoten mit lokalem Datenspeicher vor, wobei die einzelnen Knoten mittels eines Mesh-Funknetzwerkes miteinander kommunizieren. Die lokale Datenspeicherung erlaubt einerseits den Einsatz einer Funktechnologie mit geringem Energiebedarf und andererseits kann ein Verlust von übertragenen Messdaten auf Grund von elektromagnetischer Interferenz ausgeschlossen werden.

Speziell die Auswahl von Komponenten mit geringem Energiebedarf ermöglicht in weiterer Folge neben einer Versorgung der Messknoten durch das 24 V-Bordnetz des Fahrzeuges auch den Einsatz von Akkumulatoren als Energiequelle. Die in der Regel geringe Reichweite von Funktechnologien mit niedrigem Energiebedarf wird bei diesem Konzept durch die Verwendung der Mesh-Topologie ausgeglichen. Darüber hinaus ist durch diese Topologie eine hohe Robustheit gegenüber der Abschattung einzelner Funkverbindungen gegeben.

Die Steuerung des Messsystems erfolgt über das Funknetz, wobei hier neben den grundlegenden Funktionen (z.B. Start und Stopp der Messung) ebenfalls die Konfiguration einzelner Messkanäle (Abtastrate, Verstärkung, etc.) sowie die Übertragung und Anzeige der aktuellen Messwerte eines einzelnen Sensors vorgesehen sind. Nach Abschluss der Messung können die Messdaten durch Entfernen der Massenspeicher (SDHC-Karten) aus den Messknoten entnommen werden und in weiterer Folge am PC ausgewertet werden.

Die Erprobung der entwickelten Schaltungen erfolgte im Rahmen eines speziell für die einfache Handhabung im Labor ausgelegten Laborsystems. Neben den Messungen und Tests im Labor wurde mit den Knoten des Laborsystems ebenfalls eine Testfahrt in einer Straßenbahn auf dem Werksgelände der Firma Bombardier absolviert. Die Analyse der hierbei aufgezeichneten Daten hat gezeigt, dass in der realen Betriebsumgebung beim Einsatz von Funktechnologien mit Paketverlusten durch elektromagnetische Interferenz gerechnet werden muss, wobei dies jedoch natürlich keinerlei Einfluss auf die am Knoten aufgezeichneten Daten hatte.

Das im Rahmen dieser Arbeit entwickelte und erprobte Konzept ist auf Grund seiner Flexibilität und Leistungsfähigkeit nicht auf den Einsatz in Schienenfahrzeugen beschränkt und kann auf Grund nicht erforderlicher Kabelverbindungen bei einer Vielzahl von messtechnischen Aufgabenstellungen vorteilhaft eingesetzt werden.

Danksagung

Endlich ist es geschafft! Nach einigen interessanten und teils anstrengenden Jahren ist die Arbeit an meiner Dissertation abgeschlossen. Viele Personen hatten ihren Anteil daran, dass die Arbeit erfolgreich fertiggestellt bzw. überhaupt durchgeführt werden konnte. Diesen möchte ich an dieser Stelle meinen Dank aussprechen.

Zu besonderem Dank bin ich meinem Betreuer, Herrn Ao.Prof. Dr. Johann Wassermann, sowie der Firma Bombardier und der Technologieagentur der Stadt Wien (ZIT) verpflichtet. Sie haben dieses überaus interessante Projekt ins Leben gerufen, die finanziellen Mittel für dessen Durchführung zur Verfügung gestellt und mir die Möglichkeit gegeben mich mit diesem Thema zu befassen. Darüber hinaus bin ich Herrn Dr. Johann Wassermann für die hervorragende fachliche Betreuung und die mühevollen Arbeit des Korrekturlesens dankbar.

Weiters möchte ich mich bei der gesamten Arbeitsgruppe für die freundschaftliche Arbeitsatmosphäre, die vielen wertvollen Anregungen und die mir entgegengebrachte Hilfsbereitschaft bedanken. All dies hat wesentlich zum Gelingen dieser Arbeit beigetragen. Besonders hervorzuheben sind in dieser Hinsicht Herr Ing. Manfred Neumann und Herr Dr. Alexander Schulz, die mir bei Fragen immer mit Rat und Tat zur Seite standen. Auch bei Herrn Dr. Harald Sima (damals noch Cand.ing.) möchte ich mich an dieser Stelle speziell für die vielen Stunden bedanken, die er in die Programmierung des Firmwaretreibers für den DOSonChip-IC und dessen „Zähmung“ investiert hat.

Frau Renate Mühlberger und Frau Ruth Polterauer dürfen ebenfalls nicht unerwähnt bleiben. Dank ihrer unermüdlichen Arbeit im Sekretariat waren sie eine unverzichtbare Hilfe bei der Abhaltung von Lehrveranstaltungen und der Betreuung der Studenten.

Zu guter Letzt möchte ich mich ganz besonders herzlich bei meinen Eltern bedanken, die in jeglicher Hinsicht die Grundsteine für meinen Weg gelegt und mich auf diesem immer unterstützt haben.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation und Ziel der Arbeit	1
2	Stand der Technik	3
2.1	Oberflächenwellensensoren	3
2.2	Radio Frequency Identification	6
2.3	Drahtlose Sensornetzwerke	6
2.4	Telemetriesysteme	10
2.5	Messsysteme für Schienenfahrzeuge	11
3	Konzeptentwicklung	13
3.1	Erster Lösungsansatz	13
3.1.1	Anforderungen an das Messsystem	13
3.1.2	Grundkonzept des Messsystems	15
3.1.3	Entwicklungsmethode für die Umsetzung des Lösungskonzepts	17
3.2	Allgemeine Grundlagen	17
3.2.1	Funktechnologien im 2.4 GHz-Band	18
3.2.1.1	IEEE 802.11 – WLAN	20
3.2.1.2	Bluetooth	21
3.2.1.3	IEEE 802.15.4	22
3.2.1.4	IEEE 802.11, Bluetooth und IEEE 802.15.4 im Vergleich	26
3.2.1.5	Koexistenz von IEEE 802.15.4	27
3.2.2	ZigBee	28
3.2.3	Akkumulortechnologien	31
3.2.3.1	NiMH-Akkumulatoren	33
3.2.4	Flash-Basierte Speicherkarten	35
3.2.5	Hoch- und Tiefpassfilter 1. Ordnung	37
3.2.6	Piezoelektrische Beschleunigungssensoren	39
3.2.6.1	IEPE-Beschleunigungssensoren	40
3.3	Detailliertes Lösungskonzept	41
3.3.1	Funktion des Messsystems	42
3.3.2	Aufbau eines Messknotens	44
4	Das Laborsystem	46
4.1	Überblick über die Architektur und Funktion des Laborsystems	48

4.2	Basismodul	53
4.2.1	Beschreibung der Hardware	53
4.2.1.1	Funk-Transceiver	53
4.2.1.2	Energieversorgung	60
4.2.1.3	Strombedarf	61
4.2.2	Beschreibung der Firmware	62
4.2.2.1	RS232-Kommunikation	63
4.2.2.2	Netzwerkmanagementfunktionen	67
4.2.2.3	Z-Stack	72
4.3	Datenspeicher	77
4.3.1	Beschreibung der Hardware	77
4.3.1.1	FAT32-Treiber-IC – DOSonChip	80
4.3.1.2	FAT32-Firmwaretreiber	82
4.3.1.3	Zeitbasis	84
4.3.1.4	Strombedarf der SD-Karte	91
4.3.2	Beschreibung der Firmware	92
4.3.2.1	Endlicher Zustandsautomat	92
4.3.2.2	Verarbeitung der Abtastwerte	92
4.3.2.3	Dateiformat	95
4.3.2.4	Kommunikationsbeispiel	96
4.4	Allgemeine Signalaufbereitung	98
4.4.1	Eingangsschaltkreis und Verstärker	99
4.4.2	Pegelanpassung	100
4.4.3	Antialiasing-Filter	102
4.4.4	Spannungsversorgung der analogen Elektronik	104
4.4.5	Funktionstests	106
4.4.6	Strombedarf	108
4.4.7	Entwicklungsgeschichte	108
4.5	IEPE-Sensorinterface	109
4.5.1	DC/DC-Konverter	110
4.5.2	Stromquelle LT3092	111
4.5.3	Spannungsteiler	112
4.5.4	Hochpass und Verstärker	112
4.5.5	Frequenzgänge des IEPE-Sensorinterfaces	116
4.5.5.1	IEPE-Interface exklusive Sensor	117
4.5.5.2	IEPE-Interface inklusive Sensor	118
4.5.6	Strombedarf	121
4.5.7	Entwicklungsstufen	122
4.5.8	IEPE-Interface V3.1	122
4.6	USB-Interface	123
4.7	Funktionsgenerator (DDS-Modul)	127

4.8	Testfahrt	129
4.8.1	Messungen im Stillstand	131
4.8.1.1	Verbindungsqualität	131
4.8.1.2	Übertragung eines ADC-Kanals per Funk	133
4.8.1.3	Lokal aufgezeichnete Messdaten	134
4.8.2	Messungen mit Fahrtanteilen	134
4.8.2.1	Übertragung eines ADC-Kanals per Funk	136
4.8.2.2	Aufgezeichnete Messdaten	136
4.8.3	Zusammenfassung	137
5	Mesh-DAQ	138
5.1	Steuerlogik	138
5.1.1	Periphere Logik	141
5.1.2	Zentrale CPLD-Logik	143
5.1.2.1	Funktionsbelegung des Konfigurationsspeichers	143
5.1.2.2	Simulation der CPLD-Logik	145
5.1.2.3	CPLD-Timing	148
5.1.3	Prototyp der Steuerlogik	151
5.1.4	Funktionstests	153
5.1.5	Strombedarf	155
5.2	Energieversorgung	156
5.2.1	Abschätzung des Energiebedarfs eines Knotens	157
5.2.2	DC/DC-Konverter	158
5.2.2.1	LTC3112	159
5.2.2.2	LT3580	163
5.2.2.3	TPS61081	164
5.2.3	Ladeschaltung LTC4011 und PowerPath™ Controller LTC4412	165
5.3	RX-/TX-Jitter-Kompensation des Z-Stack	169
6	Zusammenfassung	173
A	Messdaten	175
B	Funktionen des Zustandsautomaten	190
C	Kommandocodes des Datenspeichermoduls	193
C.1	Systemzeit einstellen	193
C.2	Standard-Antwortadresse festlegen	193
C.3	Standard-Antwortadresse abfragen	194
C.4	Status des Moduls abfragen	194
C.5	Synchronisierung der ADC-Zeitbasis	195
C.6	Zeitbasis des ADCs per Funk starten	195
C.7	Abtastung starten	195
C.8	Funkübertragung eines ADC-Kanals starten	196

C.9	Analoge Signalaufbereitung konfigurieren	196
C.10	Konfiguration der analogen Signalaufbereitung auslesen	197
C.11	Funkübertragung eines ADC-Kanals stoppen	197
C.12	Aufzeichnung auf SD-Karte starten	198
C.13	Aufzeichnung stoppen	198
C.14	Abtastung stoppen	198
D	Pinbelegungen der Module des Laborsystems	199
D.1	Basismodul	200
D.2	Datenspeicher	203
D.3	Allgemeine Signalaufbereitung	205
D.4	IEPE-Interface	207
D.5	USB-Interface	209
D.6	Funktionsgenerator	212
D.7	Verbindungstabellen	215
E	Pinbelegung der Steuerlogik-Prototypen	217
E.1	CPLD-Platine	217
E.2	CPLD-Erweiterungsplatine	225
F	Kalibrierdatenblätter der Sensoren	229
	Abkürzungsverzeichnis	233
	Abbildungsverzeichnis	236
	Tabellenverzeichnis	242
	Literaturverzeichnis	246

1 Einleitung

Die vorliegende Arbeit entstand im Rahmen der Forschungsgemeinschaft Straßenbahn (FOGE Straba) zwischen der Fakultät für Maschinenwesen und Betriebswissenschaften und der Firma Bombardier. Ein Forschungsschwerpunkt dieser Kooperation war die „Betriebliche, messtechnische Online-Überwachung der neuen Straßenbahnen“. Da das in weiterer Folge beschriebene Konzept bzw. Messsystem nicht primär auf den Einsatz in Straßenbahnen eingeschränkt ist und entsprechend der Rückmeldung der Firma Bombardier im Bereich der Bahn prinzipiell der Bedarf nach einem solchen Messsystem besteht, wurde die „Betriebliche, messtechnische Online-Überwachung von Schienenfahrzeugen“ als Titel für die vorliegende Dissertation gewählt.

1.1 Motivation und Ziel der Arbeit

Im Betrieb einer Straßenbahn können, abhängig von der Qualität der Fahrstrecke (Gleiszustand und Zustand des Gleisunterbaus), Beeinflussungen des Fahrkomforts aber auch erhöhte Bauteilbelastungen durch Schwingungsanregungen über das Fahrwerk inklusive dem Antriebssystem auftreten. Besonders nachteilig ist, wenn im Erregerspektrum nennenswerte Amplituden bei oder in der Nähe einer Eigenfrequenz des gesamten Wagenkastens oder einer Teilstruktur auftreten.

Zur Analyse solcher, im Betrieb festgestellten, Komfortbeeinflussungen werden als Stand der Technik spezielle Messfahrzeuge eingesetzt, die das Erregerspektrum entlang der Fahrstrecke aufnehmen und eine Analyse ermöglichen. Nachteil dieser bekannten Vorgehensweise ist, dass das Messfahrzeug völlig andere Lastzustände für die Fahrstrecke darstellt und die dabei gemessenen Erregerspektren nur eine Näherung der realen Situation widerspiegeln. Zusätzlich kann auf Grund eines völlig anderen Schwingungsverhaltens des Messfahrzeuges (Schwingungseinleitung auf den Wagenkasten durch die Drehgestelle und der Ausführung des Wagenkastens selbst) dieser nicht direkt in die Analyse des aktuellen Problems mit einbezogen werden. Hier muss in weiterer Folge eine sehr aufwändige Bestimmung des Übertragungsverhaltens vom Drehgestell bis zum Problembereich durch-

geführt werden.

Für schwingungstechnische Untersuchungen an Schienenfahrzeugen würde man im Idealfall ein Messsystem benötigen, das bei Bedarf im konkret betroffenen Fahrzeug installiert werden kann und somit repräsentative Messergebnisse liefert. Speziell der mit der Installation verbundene zeitliche und materielle Aufwand stellt ein wesentliches Kriterium für die Praxistauglichkeit eines Systems dar und beeinflusst die mit dem Einsatz des Systems verbundenen Kosten wesentlich. Üblicherweise werden die Messgrößen direkt an der Messstelle mittels Sensoren erfasst und die Signale anschließend in analoger oder digitaler Form mittels Kabel an ein zentrales Gerät zur Datenverarbeitung und Aufzeichnung übertragen. Entsprechend den Vorgaben der Firma Bombardier umfasst der typische Anwendungsfall bis zu 26 Sensorpositionen, die über eine Fahrzeuglänge von etwa 40 m verteilt sind. Ein konventioneller Aufbau würde hunderte Meter an Kabelverbindungen erfordern, die darüber hinaus unsichtbar für den Fahrgast verlegt werden müssten um eine betriebliche Nutzung zu gewährleisten.

Es ist daher naheliegend – soweit technisch möglich – die Kabelverbindungen durch Funkverbindungen zu ersetzen. Da eine Funkstrecke jedoch kein abgeschirmtes Übertragungsmedium nutzt, ist diese einerseits anfällig gegenüber Störaussendungen im Umfeld und andererseits ist das in Summe pro Zeiteinheit über einen Funkkanal übertragbare Datenvolumen begrenzt. Hierdurch kann eine konventionelle Systemarchitektur nicht Eins zu Eins auf ein funkbasiertes Messsystem übertragen werden und es müssen spezielle Vorkehrungen getroffen werden um beispielsweise Datenverlust durch elektromagnetische Interferenz zu vermeiden.

Ziel dieser Arbeit ist es daher, ein möglichst flexibles Mess-, Signalübertragungs-, Speicher- und Signalauswerteverfahren für ein „Onboard-Messsystem“ zu entwickeln, welches bei Bedarf mit möglichst geringem Aufwand (Verkabelung, etc.) in der neuen Fahrzeuggenerationen installiert werden kann, ohne dass die betriebliche Nutzung hierdurch beeinträchtigt wird.

2 Stand der Technik

Funkbasierte Sensoren können in aktive Sensoren, semi-aktive Sensoren sowie passive Transponder eingeteilt werden [73]. Das Unterscheidungskriterium ist hierbei die Art und Weise auf welche der Sensor mit Energie versorgt wird. Aktive Sensoren werden mittels einer lokalen Energiequelle wie z.B. Batterien betrieben, semi-aktive Sensoren beziehen Energie berührungslos mittels induktiver Kopplung oder durch starke RF-Signale und passive Transponder werden durch das Funk-Abfragesignal selbst mit der notwendigen Energie versorgt.

Die Abfrage passiver Funktransponder erfolgt durch ein Abfragesignal (Downlink), auf das der Sensor mit dem entsprechenden Antwortsignal (Uplink) antwortet. Abfrage und Antwortsignal müssen hierbei separiert sein um eine Auswertung der Antwort zu ermöglichen. Diese Separation kann entweder im Frequenzbereich (FDD¹ – Frequency Domain Division) oder im Zeitbereich (TDD – Time Domain Division) erfolgen [73].

In den folgenden Abschnitten wird der Stand der Technik sowohl passiver Transponder als auch aktiver Sensorsysteme erläutert. Wobei das Hauptaugenmerk darauf gelegt wurde, einen Überblick über die Funktion sowie die technische Leistungsfähigkeit inkl. der Beschränkungen der jeweiligen Technologien zu geben. Detaillierte Ausführungen hinsichtlich spezieller Details würden auf Grund der umfangreichen Literatur den Rahmen dieser Arbeit sprengen. Diesbezüglich wird daher auf die angeführte Literatur verwiesen.

Die behandelten passiven Systeme umfassen Oberflächenwellensensoren sowie RFIDs (Radio Frequency Identification) und die aktiven Systeme drahtlose Sensornetzwerke und Telemetriesysteme.

2.1 Oberflächenwellensensoren

Oberflächenwellensensoren beruhen auf der Ausbreitung von akustischen Oberflächenwellen in piezoelektrischen Substraten.

¹ Die verwendeten Abkürzungen sind im Abkürzungsverzeichnis auf Seite 233 zusammengestellt.

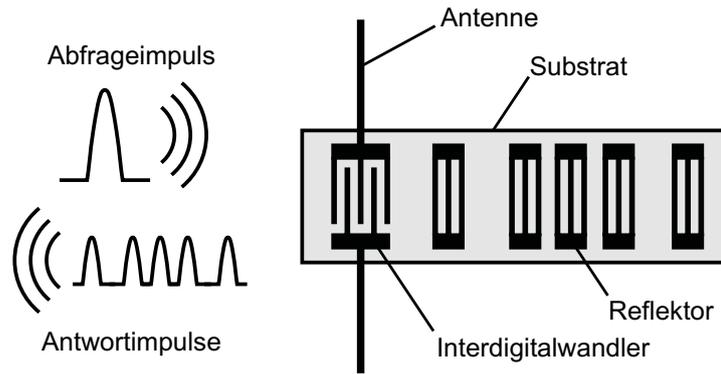


Abbildung 2.1: Aufbau eines Oberflächenwellensensors nach [77]

Abbildung 2.1 zeigt den typischen Aufbau eines auf dem Prinzip der reflektiven Verzögerungsleitung beruhenden Oberflächenwellensensors. Die Funktionsweise ist hierbei folgende: Der Abfrageimpuls des Lesegerätes wird vom Sensor mittels einer Antenne empfangen und im Interdigitalwandler in eine Oberflächenwelle umgesetzt, die sich in Oberflächennähe im Substrat ausbreitet. Durch, auf der Oberfläche aufgebraachte, Reflektoren wird die ursprüngliche Welle teilweise zurück in Richtung des Interdigitalwandlers reflektiert und generiert dort einen Antwortimpuls, der über die Antenne abgestrahlt wird. In Summe besteht das Antwortsignal des Sensors somit – abhängig von der Anzahl der Reflektoren – aus mehreren zeitlich versetzten Impulsen, wobei der zeitliche Abstand zwischen diesen von der Geometrie (Reflektorabstand) sowie der Ausbreitungsgeschwindigkeit der Oberflächenwelle abhängig ist. Der Einfluss verschiedener physikalischer Größen wie z.B. der Temperatur, der Masse eines Films auf der Sensoroberfläche, der mechanischen Dehnung, etc. auf die Ausbreitungsgeschwindigkeit der Welle bzw. zum Teil auch auf die geometrische Distanz zwischen den Reflektoren führt dabei zu einer zeitlichen Verschiebung der Antwortimpulse der betroffenen Reflektoren. Durch Auswertung des zeitlichen Abstandes der Einzelimpulse kann somit der Wert der eigentlichen Messgröße bestimmt werden. Details hinsichtlich der Ausbreitung von akustischen Wellen sowie der diversen physikalischen Einflussgrößen auf deren Ausbreitungsverhalten siehe [7].

Neben dem soeben beschriebenen direkten Sensorprinzip besteht ebenfalls die Möglichkeit indirekt wirkende Messsysteme basierend auf reflektiven Verzögerungsleitungen zu verwirklichen. Hierbei wird ein zusätzlicher Interdigitalwandler eingesetzt, dessen Reflexionsverhalten über die Impedanz eines externen Sensors gesteuert wird [30, 73]. Die Oberflächenwelle selbst wird in diesem Fall in erster Linie zur zeitlichen Trennung von Abfrageimpuls und Sensorantwort genutzt.

Abgesehen vom Prinzip der reflektiven Verzögerungsleitung existieren ebenfalls Sensoren die auf dem Prinzip des Resonators basieren [73].

Ein selektives Ansprechen einzelner Sensoren ist auf Grund des passiven Funktionsprinzips nur mittels Richtfunk möglich, da alle Sensoren im Empfangsbereich des Abfragesignals ein Antwortsignal aussenden. Die einzelnen Antwortsignale aller Sender im Empfangsbereich des Lesegerätes werden in weiterer Folge durch dieses, als überlagertes Gesamtsignal, detektiert. Eine gleichzeitige Abfrage und Unterscheidung von mehr als einem Sensor ist durch die Verwendung von mehreren Reflektoren möglich. Ein Teil der Reflektoren wird hierbei für die Unterscheidung der einzelnen Sensoren benötigt, wobei das Funktionsprinzip dem eines Barcodes ähnelt. Die Position der einzelnen Reflektoren ist derart zu wählen, dass sich die einzelnen, den Reflektoren zugeordneten Funkimpulse zeitlich nicht überlagern. Bei einer Verwendung von 14 Zeitfenstern für die Reflektorimpulse können sieben Sensoren abgefragt und unterschieden werden [73].

Eine einzelne Abfrage dauert nur wenige Mikrosekunden, weshalb prinzipiell bis zu 10^5 Abfragen pro Sekunde möglich sind [74]. Auf Grund der hohen Signaldämpfung der Sensoren und der gesetzlichen Vorschriften bezüglich der maximal zulässigen Sendeleistung des Lesegerätes können bei einer einzelnen Abfrage, entsprechend [74], Entfernungen von 26 cm bis zu einigen Metern überbrückt werden. Durch Mittelung mehrerer Abfragen kann diese Distanz vergrößert werden. Spezifikationen hinsichtlich der erreichbaren Abfrageentfernung und der hierbei erzielten effektiven Ausleserate sind in der bekannten Literatur nur selten ausgeführt. Entsprechend [74] sind bei Mittelungen über 100 ms im 434 MHz-ISM-Band (ISM – Industrial, Scientific and Medical) 5 bis 10 m und im 2.45 GHz-ISM-Band² 1 bis 2 m erzielbar. Eine aktuellere Publikation des selben Autors [75] stellt ein Auslesegerät vor, das im 2.45 GHz-ISM-Band arbeitet und Entfernungen bis 10 m, bei einer effektiven Ausleserate von 50 Messwerten pro Sekunde, überbrücken kann.

Eine viel versprechende Methode zur Verbesserung der Reichweite stellt die Frequenzspreizung mittels orthogonaler Frequenzkodierung dar. Durch Anwendung dieser Technik soll die Signaldämpfung eines Oberflächenwellensensors mit ID-Funktionalität von 30 dB oder mehr auf etwa 6 bis 15 dB reduziert werden können [58]. Ein weiterer Vorteil ist die höhere Diversität hinsichtlich der Identifizierungscodes, wodurch eine höhere Anzahl von Sensoren ohne Kollisionen gleichzeitig betrieben werden können [44]. Versuche bzw. Messungen mit dem in [44] vorgestellten Prototypen eines Multisensorsystems basierend auf orthogonaler Frequenzkodierung wurden jedoch lediglich mit vier gleichzeitig abgefragten

² Abweichend von der üblichen Konvention wird im Rahmen dieser Arbeit durchgehend der im englischsprachigen Raum übliche Punkt anstatt des Kommas als Dezimaltrennzeichen verwendet.

Sensoren in einem Abstand von maximal 4.5 cm zur Antenne des Lesegeräts publiziert.

Die mit Hilfe von Oberflächenwellensensoren erfassbaren Messgrößen sind vielfältig [30, 73, 74] und umfassen unter anderem die Größen Temperatur, Druck, Dehnung, Beschleunigung, Kraft und Drehmoment. Auf Grund ihres einfachen und robusten Aufbaus sind Oberflächenwellensensoren speziell auch für Einsätze unter rauen Umgebungsbedingungen prädestiniert [110].

2.2 Radio Frequency Identification (RFID)

Wie Oberflächenwellensensoren können RFIDs ebenfalls als vollkommen passive Systeme, bestehend aus Lesegerät und Transponder, ausgeführt werden. Im Gegensatz zu den Oberflächenwellensensoren beinhalten RFID-Transponder jedoch einen Mikrochip und werden in der Regel zur Identifikation von Gegenständen oder Lebewesen eingesetzt. Die Energieversorgung eines passiven Transponders erfolgt hierbei je nach Frequenzbereich über das magnetische oder elektromagnetische Feld des Lesegerätes. Besagtes Feld dient ebenfalls der Kommunikation, die in der Regel durch Lastmodulation oder Backscattering erfolgt [23]. Die Besonderheit besteht hierbei darin, dass der Transponder – im Gegensatz zur konventionellen Funktechnologie – kein eigenes Feld aufbaut und somit lediglich eine geringe Energie für die Kommunikation erforderlich ist. Typischerweise können mit passiven RFID-Systemen derzeit Distanzen im Bereich von 3 m überbrückt werden [23]. Unter Verwendung von Batterien zur Versorgung der Elektronik (aktive Transponder) kann die Reichweite bis auf ca. 15 m gesteigert werden [23].

Neben der reinen Identifikation von Gegenständen war und ist die Integration von Sensoren bzw. die Entwicklung von RFID-basierten Messsystemen mit rein passiven Transpondern Ziel von aktuellen Entwicklungsprojekten [27, 76, 111]. Die erzielbaren Ausledistanzen liegen jedoch maximal im Bereich von einigen Metern und der Einsatz von Sensoren ist in der Regel auf MEMS-Sensoren (MEMS – Microelectromechanical Systems) mit geringer Verlustleistung beschränkt. Entsprechend [76] sind beispielsweise maximale Distanzen von 4.3 m möglich und es können Sensoren mit einem Strombedarf von 500 μA (bei 1.8 V) mit dem vorgestellten System betrieben werden.

2.3 Drahtlose Sensornetzwerke

Drahtlose Sensornetzwerke bestehen aus einzelnen, mit Sensoren bestückten, Knoten die mittels eines Funknetzwerkes miteinander kommunizieren können. Ein einzelner Knoten

besteht hierbei in der Regel aus einem Mikrocontroller, einem oder mehreren Sensoren und einem Funktransceiver samt erforderlichem Energiespeicher (Batterien). Typische Anwendungsgebiete von drahtlosen Sensornetzwerken sind Applikationen bei denen weitläufige und/oder schwer zugängliche Gebiete bzw. Bereiche mittels Sensoren überwacht werden müssen, wobei hierbei auch mobile Knoten [39] zum Einsatz kommen können. Ziel dieser Überwachung kann sowohl die Erfassung von physikalischen Größen (z.B. Temperatur, Luftfeuchtigkeit, etc. [87, 105]) oder Zuständen z.B. strukturelle Integrität von Brücken [112]) als auch in weiterer Folge die Detektion und/oder Lokalisierung von bestimmten Ereignissen (z.B. Waldbränden [83] oder Vulkanausbrüchen [109]), Gegenständen [46] oder Lebewesen [39, 87] sein. Die erfassten Daten werden in der Regel über die Funkschnittstelle zu einer oder mehreren Datensinken weitergeleitet, die die Daten schließlich über eine leistungsfähige Datenverbindung an einen Server bzw. zum Benutzer zur Aufzeichnung und Auswertung übermittelt. Je nach Anwendung kann die Anzahl von ein paar bis zu einigen tausend Knoten reichen [113]. Innerhalb des Sensornetzwerkes werden die Funkpakete im Multi-Hop-Verfahren bei Bedarf über eine beliebige Anzahl von Zwischenknoten vom Absender an den Empfänger weitergeleitet.

Ausschlaggebend für den Einsatz von drahtlosen Sensornetzwerken sind in der Regel die bei konventionellen Systemen anfallenden hohen Kosten (Material und Zeitaufwand) für die Verkabelung zwischen den Sensoren und einer zentralen Datenerfassungseinheit. Die Verwendung von drahtlosen Sensornetzwerken ist jedoch nur dann finanziell interessant, wenn die mit dem Wechsel des Energiespeichers verbundenen Kosten im Laufe der Einsatzdauer die eingesparten Verkabelungskosten nicht übersteigen. Je nach Anwendung kann die Batterielebensdauer einige Monate oder Jahre betragen, jedoch auch deutlich darunter liegen.

Ein wesentliches Auslegungskriterium bei der Entwicklung von Sensornetzwerken ist daher ein möglichst geringer Energiebedarf der Knoten. Üblicherweise werden 8 oder 16 Bit-Mikrocontroller der Firmen Atmel, Texas Instruments und Microchip eingesetzt [29, 37], die einen geringen Strombedarf aufweisen. Neuere Hardwaredesigns beruhen zumeist auf Funktransceivern [29], die dem Standard IEEE 802.15.4 [34] entsprechen. Dieser Standard definiert für WPANs (Wireless Personal Area Network) [18] die Bitübertragungsschicht und die MAC-Schicht (MAC – Media Access Control). Es handelt sich hierbei um einen Kurzstrecken-Funkstandard mit geringem Energiebedarf und einer Reichweite von etwa 125 m [9]. Neben dem Einsatz von Bauteilen mit geringer elektrischer Leistung werden die einzelnen Komponenten (inkl. Mikrocontroller), sofern diese nicht benötigt werden, in einen Breitschafts- bzw. Power-Down-Modus versetzt. Die Kombination von aufladbaren

Energiespeichern (z.B. Akkumulatoren oder Superkondensatoren) und Energy-Harvesting beispielsweise mittels Solarzellen [1,69,81] erlaubt je nach Anwendung Betriebsdauern von einigen Jahren ohne einen Wechsel des Energiespeichers. Bei Verwendung von Superkondensatoren als Energiespeicher sind sogar Einsatzdauern bis zu 20 Jahren möglich [81].

Resultierend aus der Wahl von Komponenten mit geringem Energiebedarf sind sowohl die Rechenleistung als auch die per Funk erzielbare Reichweite und Datentransferrate stark limitiert. Funk-Transceiver entsprechend dem Standard IEEE 802.15.4 stellen im 2.4 GHz-Band eine Datentransferrate von 250 kbit/s zur Verfügung. Unter Berücksichtigung des im Standard beschriebenen Paketaufbaus ergibt sich nach [72] für eine einfache Punkt-zu-Punkt-Verbindung eine maximale Übertragungsrate von 225 kbit/s. Bei Multi-Hop-Übertragungen ist dieser Wert durch die Anzahl der sich im Interferenzbereich befindlichen Knoten zu teilen. Im Interferenzbereich tritt eine gegenseitige Beeinflussung der Transceiver auf, wobei jedoch ein erfolgreicher Datenaustausch zwischen den betroffenen Transceivern über diese Distanz nicht gegeben sein muss (Interferenzbereich > Empfangsbereich) [4]. In der Praxis ist die im Netzwerk erzielbare Datenübertragungsrate von einer Vielzahl von Faktoren abhängig. Diese sind unter anderem: die Funktechnologie (Bitübertragungs- und die MAC-Schicht), die Netzwerktopologie, das Netzwerkprotokoll, die Anzahl der Knoten, die gegenseitige Lage der Knoten, die Verbindungsqualität zwischen den Knoten, die Anzahl und Lage der gleichzeitig sendenden Knoten, die Paketlänge der Funknachrichten, die Anzahl und Lage der Senke(n), etc. Die tatsächlich mittels IEEE 802.15.4 erreichbaren Werte können somit beträchtlich unter den genannten Maximalwerten liegen. Die im Folgenden diesbezüglich zitierten Datentransferraten wurden sämtlich unter Verwendung von IEEE 802.15.4-konformen Transceivern ermittelt. Für Single-Hop-Übertragungen sind in der Literatur Werte von ca. 50 kbit/s [22, 62], ca. 110 kbit/s [12] und ca. 140 kbit/s [72] angeführt. Bei Multi-Hop-Übertragungen sind deutlich geringere Datentransferraten dokumentiert. Entsprechend [12] liegt bei drei Zwischenknoten die Datenrate bei ca. 10 kbit/s. Etwas höhere Werte sind in [72] angeführt. Laut den dort veröffentlichten Untersuchungen sind ebenfalls bei drei Zwischenknoten etwa 30 kbit/s bzw. unter Verwendung von „Conditional Immediate Transmission“ ca. 55 kbit/s erreichbar. Bei einer sehr hohen Anzahl von Zwischenknoten reduziert sich die Datentransferrate weiter. Entsprechend [42] konnten bei 47 Zwischenknoten 4.4 kbit/s erzielt werden.

Die kontinuierliche Erfassung und Übermittlung von mehreren Kanälen mit hohen Abtastraten im Bereich von 100 S/s und darüber ist, wie die obigen Werte zeigen, speziell im Fall von Multi-Hop-Übertragungen nicht praktikabel. Es existieren jedoch trotz dieser

Einschränkung Anwendungsbereiche von drahtlosen Sensornetzwerken, die hohe Abtastfrequenzen erfordern. Es handelt sich hierbei beispielsweise um die Überwachung der strukturellen Integrität von Brücken [43, 69, 112], die Überwachung von Vulkanen [109] oder die Analyse von Vogelgesang [8]. Gemeinsam ist diesen Anwendungen, dass die Daten im lokalen Speicher des Knotens abgelegt werden und danach entweder per Funk in komprimierter [112] oder unkomprimierter [43, 109] Form übertragen werden oder direkt im Knoten verarbeitet und lediglich die Ergebnisse der Analyse übertragen bzw. lokal gespeichert werden [8, 69]. Auf Grund der meist geringen Speichergröße des lokalen Datenspeichers und der geringen Datenübertragungsrate der Funkverbindung sind bei erster Lösung jedoch in der Regel lediglich kurzzeitige Erfassungen (Datenmenge 512 kB bei 1 kS/s [43], Aufzeichnungsdauer 60 s bei 100 S/s [109], Aufzeichnungsdauer 1 ms bei 10 MS/s [69]) möglich.

In der Literatur ist darüber hinaus der Einsatz von Massenspeichern in Form von SD-Karten mit dem Ziel der Energieeinsparung bzw. Verlängerung der Betriebsdauer [8], dem Schutz vor Datenverlust [8, 78] und der Hilfestellung bei der Problembehebung [79] beschrieben. Eine Energieeinsparung ist laut [8] dadurch gegeben, dass für das Abspeichern von 100 kB Daten auf SD-Karten 245.6 mJ benötigt werden und für das Abspeichern im EEPROM der Mica-Netzwerkknoten 2450 mJ bzw. für das Versenden 16 473 mJ aufgewendet werden müssen. Vorteilhaft kann die lokale Speicherung von Daten jedoch nur dann eingesetzt werden, wenn während des Betriebs nicht die gesamten aufgezeichneten Daten per Funk übertragen werden müssen. Dies ist beispielsweise dann der Fall, wenn lediglich Minima, Maxima, Mittelwerte, etc. übermittelt werden bzw. die aufgezeichneten Daten einen hohen Anteil redundanter Informationen enthalten [8]. Neben der erzielbaren Energieeinsparung können die bei Ausfällen der Funkverbindung zwischen Teilen des Netzwerks oder zwischen Datensenke bzw. Gateway des Sensornetzwerkes und dem für die Datenaufzeichnung verantwortlichen Server die ansonsten unwiederbringlich verlorenen Messdaten aus den lokalen Massenspeichern nachträglich per Funk oder Kartenleser ausgelesen werden. Hierfür können entsprechend [78] dezidierte Knoten eingesetzt werden, die die per Funk übertragenen Daten der umliegenden Knoten aufzeichnen und somit jeweils einen Teil des Netzwerkes abdecken. Es ist somit nicht erforderlich jeden Knoten mit einem Massenspeicher auszurüsten.

Da es sich bei Sensornetzen um verteilte Systeme handelt und jeder einzelne Knoten eine eigene Zeitbasis besitzt, ist – speziell im Hinblick auf lange Einsatzdauern – die Synchronisierung dieser Zeitbasen für eine korrekte Funktion des Gesamtsystems bei vielen Anwendungen unerlässlich [82, 86]. Dies trifft insbesondere dann zu, wenn die Messdaten

einer modalen Analyse unterzogen werden wie z.B. im Rahmen der Untersuchung der strukturellen Integrität von Bauwerken [43]. Je nach Anforderung findet eine Vielzahl von Verfahren Anwendung, wobei diesbezüglich auf die Literatur [82, 86] verwiesen wird. Die erzielbare Genauigkeit liegt dabei im Bereich von wenigen Mikrosekunden bis wenigen Millisekunden [82, 86].

Auf Grund der umfangreichen Literatur ist es nahezu unmöglich im Rahmen dieses Kapitels einen umfassenden und detaillierten Überblick über die Technologie der drahtlosen Sensornetzwerke zu geben. Diesbezüglich muss daher auf die einschlägige Literatur wie z.B. [41, 113] verwiesen werden.

2.4 Telemetriesysteme

Analog zu den drahtlosen Sensornetzwerken dienen Telemetriesysteme der drahtlosen Übermittlung von Messdaten. Der wesentliche Unterschied zwischen beiden Systemen besteht darin, dass Telemetriesysteme keine Multi-Hop-Verbindungen beherrschen. In der Regel ist jeweils einem Empfänger genau ein Sender zugeordnet, wobei dieser Sender jedoch üblicherweise mehrere Sensorkanäle aufweisen kann.

Das unter der Bezeichnung *Wireless Sensor Network* geführte System der Firma Microstrain (<http://www.microstrain.com>) ähnelt insofern bereits stark drahtlosen Sensornetzwerken, als es den Betrieb von mehreren Sensorknoten mit einem Empfänger erlaubt [66]. Die maximale Distanz zwischen Sender und Empfänger ist jedoch auf die Single-Hop-Reichweite der Funkschnittstelle begrenzt. Diese beträgt unter idealen Bedingungen je nach verwendeter Antenne 70 bis 280 m [66]. Zum Einsatz kommen bei diesem System ebenfalls Transceiver, die dem Standard IEEE 802.15.4 entsprechen. Es gelten daher ähnliche Restriktionen bezüglich der Datentransferrate wie bereits in Abschnitt 2.3 beschrieben.

Typische Telemetriesysteme beschränken sich auf jeweils einen, dem Empfänger zugeordneten, Sender und finden vornehmlich bei der Übertragung von Messgrößen zwischen bewegten Objekten (rotierenden Wellen, Fahrzeugen, etc.) bzw. schwer zugänglichen Messstellen und einer Basiseinheit Anwendung. Weitere Hersteller von Telemetriesystemen sind beispielsweise die Firmen Advanced Telemetry International (<http://www.atitelemetry.com>), Kraus Messtechnik GmbH (<http://www.kmt-gmbh.com>), Manner Sensortelemetrie GmbH (<http://www.sensortelemetrie.de/>) und Telemetrie-Messtechnik Schnorrenberg (<http://www.telemetry-world.com/>).

2.5 Messsysteme für Schienenfahrzeuge

Spezielle Literatur hinsichtlich drahtloser Messsysteme für den Einsatz in Schienenfahrzeugen war zum Zeitpunkt des Projektstarts (2008) nicht verfügbar. Stand der Technik sind hier nach wie vor kabelgebundene Systeme, wie sie z.B. in der Dresdner Messstrassenbahn [10] eingesetzt werden. Der hierbei notwendige technische Aufwand zur Sicherstellung eines ungestörten Fahrgastbetriebes ist jedoch enorm. Auf Grund dieser Tatsache ist eine Installation vergleichbarer Systeme im konkreten Bedarfsfall nicht praktikabel. Der Einsatz von Funktechnologien beschränkt sich derzeit auf die Kommunikation zwischen Messknoten im Fahrzeug, die der Betriebsüberwachung dienen, und einer ortsfesten Basiseinheit [70].

Kurz vor Abschluss der Entwicklungsarbeiten im Rahmen des Projekts „Betriebliche, messtechnische Online-Überwachung der neuen Straßenbahnen“ wurde in der Publikation [48] das Messsystem ViFDAQ vorgestellt, das auf dem selben Grundkonzept beruht wie das im Rahmen dieser Arbeit entwickelte System. Dieses, ohne das Wissen des Autors, zeitlich parallel entwickelte System besteht ebenfalls aus einzelnen Messknoten mit lokalen Massenspeichern, die mittels Funk vernetzt und gesteuert werden können. Das Hauptaugenmerk lag bei diesem System jedoch auf einer kompakten Baugröße (30 mm x 20 mm x 15 mm) und einem autarken Betrieb mittels Lithium-Ionen-Akkumulatoren. ViFDAQ wurde für den Einsatz in der Automobilbranche entwickelt und besitzt integrierte Sensoren für Beschleunigungen (3D), Drehraten (3D), magnetische Flussdichten (3D), Temperatur, Luftfeuchtigkeit und Luftdruck. Es handelt sich hierbei vornehmlich um fix integrierte MEMS-Sensoren. Speziell bei schwingungstechnischen Aufgabenstellungen kann die Verwendung von integrierten Sensoren jedoch problematisch sein. Einerseits auf Grund der relativ hohen mitschwingenden Masse des Geräts (Rückwirkung auf das Messobjekt) und andererseits, wenn die strukturelle Anbindung des Sensors über die Platine bis zum Befestigungspunkt des Gerätes nicht ausreichend steif ausgeführt ist. Darüber hinaus muss das Gerät bezüglich der auftretenden Schwingungsamplituden stabil ausgeführt sein. Analoge Eingänge für externe Sensoren sind bei ViFDAQ zwar prinzipiell vorhanden, eine analoge Signalaufbereitung und Filterung oder speziell angepasste Sensorinterfaces, wie bei dem im Rahmen dieser Arbeit entwickelten System, stehen jedoch nicht zur Verfügung. Hinsichtlich der (Summen-)Abtastrate bzw. der Performance beim Schreiben auf SD-Karten sind in [48] keine Daten enthalten.

Das Grundkonzept der per Funk vernetzten Messknoten mit lokalen Datenspeichern liegt beiden Systemen zu Grunde, in der konkreten Ausführung bzw. Ausrichtung un-

terscheiden sie sich jedoch deutlich. ViFDAQ ist äußerst kompakt und vornehmlich auf den Betrieb mit den internen Sensoren ausgerichtet, während bei der Entwicklung im Rahmen dieser Arbeit die Verwendung von externen, handelsüblichen Sensoren im Fokus stand und daher eine flexible Signalaufbereitung inklusive Filterung und tauschbare Sensorinterfaces vorgesehen wurden. Ob die Performance von ViFDAQ für die betriebliche, messtechnische Online-Überwachung von Schienenfahrzeugen ausreichend ist, kann anhand der publizierten Daten nicht beurteilt werden.

3 Konzeptentwicklung

Im Folgenden wird die Entwicklung des Lösungskonzeptes ausgehend von den Anforderungen an das System und einem ersten, aus dem Stand der Technik folgenden, Lösungsansatz (Abschnitt 3.1) beschrieben. Hierbei wird auf die wesentlichen Technologien, die zur Umsetzung des ersten Lösungsansatzes erforderlich sind, und deren Einschränkungen eingegangen (Abschnitt 3.2) und in weiterer Folge ein detailliertes Lösungskonzept erarbeitet (Abschnitt 3.3).

3.1 Erster Lösungsansatz

Auf die Motivation, die zur Durchführung des gegenständlichen Entwicklungsprojekts geführt hat und den Zweck des Messsystems wurde bereits in Abschnitt 1.1 eingegangen. An dieser Stelle sollen nun die daraus resultierenden Anforderungen an das Messsystem erläutert und ein adäquates Lösungskonzept vorgestellt werden.

3.1.1 Anforderungen an das Messsystem

Mit dem zu entwickelnden Messsystem soll im konkreten Anlassfall die neue Generation von Straßenbahnen mit möglichst geringem Aufwand – speziell im Hinblick auf die Verkabelung – ausgerüstet werden können. Den Idealfall würden hier drahtlose Sensoren darstellen, die ebenfalls über das elektromagnetische Feld eines zentralen Basisgerätes versorgt werden. Die Recherche zum Stand der Technik hat jedoch gezeigt, dass sowohl Oberflächenwellensensoren (Abschnitt 2.1) als auch passive RFID-Transponder (Abschnitt 2.2) bezüglich der überbrückbaren Distanzen, der Anzahl der gleichzeitig auslesbaren Sensoren und der erzielbaren Abstraten starken Einschränkungen unterworfen sind und ein Messsystem auf Basis dieser Technologien derzeit nicht praktikabel ist.

Um den Verkabelungsaufwand möglichst gering zu halten muss daher auf gängige Funktechnologien zur Datenübertragung zurückgegriffen werden. Diesbezüglich ist zu beachten, dass bei Funkübertragungen immer mit elektromagnetischer Interferenz und einem daraus

resultierenden Verlust von Datenpaketen zu rechnen ist. Weiters kann durch Gegenstände und Personen eine zusätzliche Dämpfung des Funksignals erfolgen, wodurch im schlimmsten Fall einzelne Funkverbindungen zwischen bestimmten Geräten des Messsystems nicht mehr für die Kommunikation zur Verfügung stehen. Neben der begrenzten Datenübertragungsrate müssen diese Faktoren bei der Auslegung des Messsystems ebenfalls berücksichtigt werden, da andernfalls Messdaten unwiderruflich verloren gehen können. Aus der limitierten Datenübertragungsrate folgt im Übrigen eine obere Grenze der per Funk übertragbaren Messkanäle, die je nach Funktechnologie, der Anzahl von Bits pro Abtastwert und der Abtastrate bereits im einstelligen Bereich angesiedelt sein kann. Da das System flexibel an die spätere Messaufgabe anpassbar sein soll, ist jedoch eine möglichst gute Skalierbarkeit hinsichtlich der Anzahl der verfügbaren Messkanäle erforderlich, die mit konventionellen funkbasierten Messsystemen aktuell nicht erreicht werden kann.

Die Ausrüstung jedes einzelnen Sensors mit einer eigenen Funkschnittstelle ist auf Grund des hohen Hardwareaufwandes und der damit verbundenen Kosten prinzipiell nicht vorteilhaft. Sinnvollerweise werden daher mehrere Sensoren mittels Kabelverbindungen an einem Knoten des Messsystems betrieben. Die hierfür erforderlichen Kabel sind in weiterer Folge durch eine geschickte Wahl der Positionen der Systemknoten in Relation zu den Messstellen so zu verlegen, dass ein ungestörter Fahrgastbetrieb möglich ist. Die räumliche Abdeckung der gesamten Straßenbahn mit Sensoren muss durch das System selbstverständlich ebenfalls gegeben sein.

Die Energieversorgung der Knoten kann prinzipiell mittels des 24 V-Bordnetzes erfolgen, wobei hier mit kurzen Spannungseinbrüchen zu rechnen ist und daher ein Pufferakkumulator vorgesehen werden muss. Der reine Betrieb mittels Akkumulator muss jedoch prinzipiell ebenfalls unterstützt werden, da in bestimmten Bereichen wie z.B. im Bereich des Drehgestells eine einfache Anbindung an das Bordnetz nicht möglich ist. Dies wäre beispielsweise mit Hilfe eines optionalen Zusatzakkumulators denkbar, der nicht im Gehäuse der Systemknoten integriert ist sondern extern in dessen unmittelbarer Umgebung untergebracht wird.

Entsprechend den Vorgaben der Firma Bombardier muss das System zumindest 26 Messkanäle aufweisen. Eine Wiedergabe des Standard-Messstellenplanes ist an dieser Stelle aus Geheimhaltungsgründen leider nicht möglich. Aus diesem geht unter anderem hervor, dass pro Knoten vier Sensorkanäle erforderlich sind. Der relevante Frequenzbereich der Messgrößen erstreckt sich von 0 bis 100 Hz.

Um eine ausreichende zeitliche Auflösung der Messsignale sicherzustellen wurde daher in weiterer Folge eine Abtastrate von 1 kS/s als Anforderung definiert. Zu messen sind in

erster Linie Beschleunigungen sowie die Position (mittels GPS) und die Geschwindigkeit des Fahrzeuges (Signale am Fahrzeugbus).

Weiters soll das System möglichst flexibel hinsichtlich der einsetzbaren Sensortypen aufgebaut sein und eine geringe Baugröße aufweisen. Eine einfache Handhabung/Bedienung ist selbstverständlich ebenfalls gefordert.

Die grundlegenden Anforderungen an das zu entwickelnde Messsystem lauten daher zusammengefasst:

- Geringer Verkabelungsaufwand,
- keine Einschränkung des Fahrgastbetriebes,
- Robustheit hinsichtlich elektromagnetischer Störungen und der Abschattung einzelner Funkverbindungen durch Hindernisse/Objekte im Fahrzeug,
- gute Skalierbarkeit hinsichtlich der Anzahl der Messkanäle,
- räumliche Abdeckung der gesamten Straßenbahn mit Sensoren,
- Energieversorgung mittels 24 V-Bordnetz oder Akkumulatoren,
- insgesamt zumindest 26 Sensorkanäle mit einer Abtastrate von 1 kS/s pro Kanal,
- vier Sensorkanäle pro Knoten,
- hohe Flexibilität hinsichtlich der einsetzbaren Sensortypen,
- geringe Baugröße und
- einfache Handhabung/Bedienung.

3.1.2 Grundkonzept des Messsystems

Derzeit erfüllt kein am Markt erhältliches System die genannten Anforderungen vollständig, wobei das Funktionsprinzip der drahtlosen Sensornetzwerke hinsichtlich der Robustheit und Erweiterbarkeit sehr gute Eigenschaften aufweist. Lediglich die geringen erzielbaren Abtastraten verhindern einen direkten Einsatz dieser Technologie.

Das Lösungskonzept beruht daher im Wesentlichen auf dem Prinzip der drahtlosen Sensornetzwerke, sieht zusätzlich jedoch die lokale Speicherung der Messdaten am jeweiligen Systemknoten vor. Die Steuerung des Gesamtsystems erfolgt mittels Funk, wobei hier

ebenfalls die Übertragung eines frei wählbaren Messkanals per Funk zur Überwachung durch den Benutzer vorgesehen ist. Das Konzept ermöglicht so einerseits eine beliebige Erweiterbarkeit des Messsystems hinsichtlich der Anzahl der Messkanäle, ohne Einschränkung durch die zur Verfügung stehende Übertragungsbandbreite, und schließt andererseits Aufzeichnungslücken auf Grund von Daten-Paketverlusten durch elektromagnetische Interferenz bei der Funkübertragung aus. Weiters wird durch einen modularen Aufbau mit wechselbaren Sensorinterfaces die Verwendung von beliebigen handelsüblichen Sensoren ermöglicht und die einzelnen Systemknoten können somit flexibel an die jeweilige Messaufgabe angepasst werden.

Der prinzipielle Aufbau eines Systemknotens, ist in Abbildung 3.1 dargestellt. Die Datenübertragung innerhalb des Messsystems erfolgt mittels eines Mesh-Netzwerkes mit Multi-Hop-Funktionalität (siehe Abbildung 3.2), das eine robuste Kommunikation innerhalb des Systems ermöglicht. Wie in Abbildung 3.2 dargestellt können mit Hilfe dieser

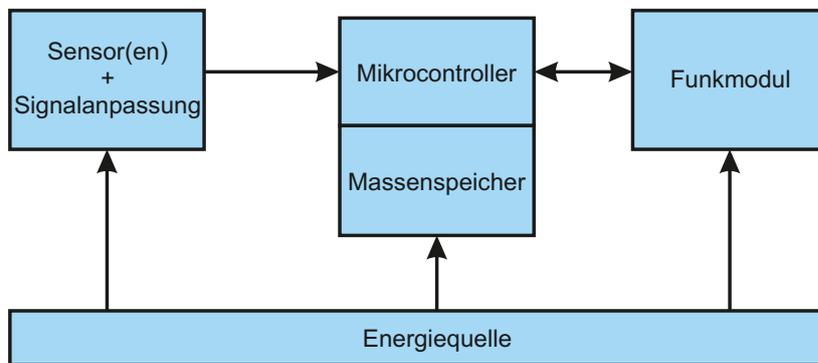


Abbildung 3.1: Prinzipieller Aufbau eines Systemknotens

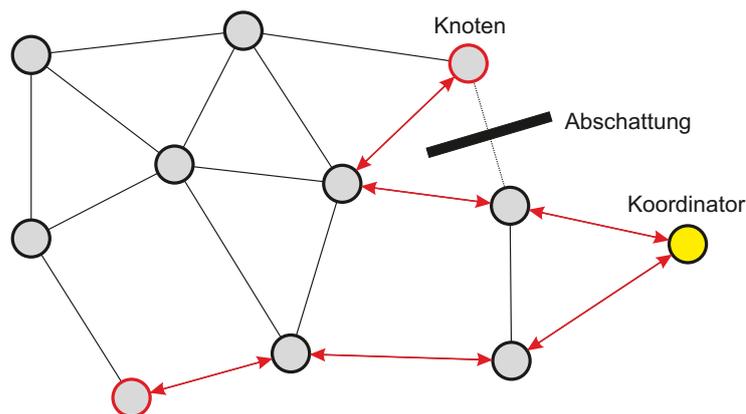


Abbildung 3.2: Mesh-Topologie

Netzwerktopologie bei Bedarf Daten über mehrere Zwischenknoten vom Sender an den Empfänger weitergeleitet werden. Der Ausfall von einzelnen direkten Verbindungen zwischen bestimmten Knoten kann auf diese Weise ebenfalls kompensiert werden. Diese Topologie erlaubt somit einerseits die räumliche Abdeckung der gesamten Straßenbahn und ist andererseits bei entsprechend dicht verteilten Systemknoten sehr robust gegen Ausfälle von einzelnen Knoten bzw. Abschattungen von Funkverbindungen durch Personen oder Objekte.

3.1.3 Entwicklungsmethode für die Umsetzung des Lösungskonzepts

Die Basis für die Entwicklung der Technologie des Messsystems stellt ein speziell hinsichtlich einer einfachen Handhabung im Labor ausgelegtes Laborsystem dar. Dieses ist modular ausgelegt, wobei die einzelnen Funktionen als Funktionsprototypen in Form einzelner Platinen ausgeführt sind, die durch Steckverbinder zu einem Messknoten kombiniert werden können. Hieraus resultiert eine ausgesprochen flexible Entwicklungsplattform bei gleichzeitig minimiertem Entwicklungsaufwand, da Redesigns im Wesentlichen auf den jeweiligen Funktionsprototypen beschränkt bleiben. Die Beschreibung der Hard- und Software des Laborsystems befindet sich im Kapitel 4.

Weitere Entwicklungsschritte, die der konkreten Umsetzung der im Rahmen des Laborsystems erarbeiteten Technologien und Erkenntnisse in ein Messsystem für Straßenbahnen dienen, sind in Kapitel 5 zusammengefasst.

3.2 Allgemeine Grundlagen

Im Rahmen der Dissertation wurde sowohl die Hard- als auch die Firmware des Laborsystems inklusive einiger essentieller Komponenten des angestrebten Messsystems entwickelt. Auf allgemeine Grundlagen wie die Funktion von Mikrocontrollern oder programmierbaren Logikbausteinen sowie deren Programmierung in C bzw. VHDL (Very high speed integrated circuit Hardware Description Language) kann aus Platzgründen nicht näher eingegangen werden. Details hinsichtlich der verwendeten Mikrocontroller (MSP430 von Texas Instruments) können beispielsweise in [15] bzw. den jeweiligen Family User Guides [96, 97] und den zugehörigen Datenblättern [93, 94] entnommen werden. Weiters bietet [59] einen guten Überblick über die Technologie der programmierbaren Logikbausteine, wobei auf deren konkrete Programmierung jedoch nicht näher eingegangen wird.

Diese Lücke schließen z.B. [13, 14], die die Grundlagen der Programmierung auf RTL-Ebene (RTL – Register Transfer Level) in VHDL beschreiben.

Die in weiterer Folge behandelten Grundlagen beschränken sich im Wesentlichen auf jenes Wissen, das eine entscheidende Rolle bei der Entwicklung bzw. Auslegung des Systems spielte. Konkret wird ein kurzer Überblick über die im 2.4 GHz-Band vorherrschenden Funktechnologien, die gängigen Akkumulatortechnologien sowie die derzeit verfügbaren Flash-basierten Speicherkartentypen gegeben. Weiters wird kurz auf Hoch- und Tiefpassfilter 1. Ordnung, Piezo-Beschleunigungssensoren und den Universal Serial Bus (USB) eingegangen.

3.2.1 Funktechnologien im 2.4 GHz-Band

Für industrielle, wissenschaftliche und medizinische Anwendungen wurden von der International Telecommunication Union (ITU) die so genannten ISM-Bänder (Industrial, Scientific and Medical) definiert [107]. Diese Frequenzbereiche können lizenzfrei zur Kommunikation per Funk genutzt werden, wobei hier jedoch Auflagen bezüglich Sendeleistung, Zugriffsverfahren etc. zu beachten sind [19]. Abhängig von den regionalen Vorschriften stehen jedoch nur bestimmte ISM-Bänder zur Verfügung. Weltweit sogar nur ein einziges, das 2.4 GHz-Band von 2.4 bis 2.5 GHz. Abbildung 3.3 zeigt einen groben Überblick über die regional verfügbaren ISM-Bänder.

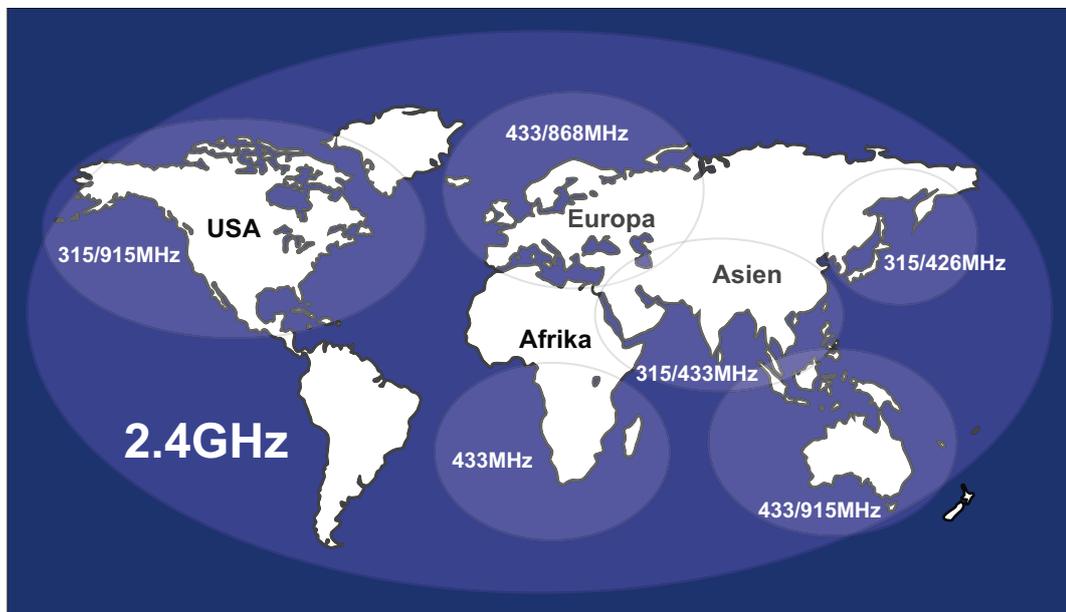


Abbildung 3.3: Lokale Verfügbarkeit der ISM-Bänder nach [102]

Ein weltweit einsetzbares, auf Funktechnologie basierendes, Messsystem muss daher zwangsläufig das 2.4 GHz-Band nützen. Auf Grund der weltweiten Verfügbarkeit arbeiten eine Vielzahl von Geräten in diesem Frequenzbereich, weshalb entsprechend robuste Übertragungsverfahren eingesetzt werden müssen. Unter anderem liegen die weit verbreiteten Standards IEEE 802.11b/g/n (WLAN) und Bluetooth in diesem Frequenzbereich. Darüber hinaus nutzen IEEE 802.15.4, ein im Bereich der Sensornetzwerke und Telemetriesysteme häufig eingesetzter Standard, sowie einige proprietäre Technologien wie z.B. nanoNET dieses Frequenzband.

Neben der starken Nutzung des 2.4 GHz-Bandes besteht ein wesentlicher Nachteil darin, dass bei Frequenzen im Gigahertz-Bereich Wassermoleküle durch elektromagnetische Wellen effizient in Schwingung versetzt werden können – Mikrowellenherde arbeiten daher auch in diesem Frequenzbereich [45]. Im Gegenzug werden die elektromagnetischen Wellen folglich stark gedämpft. Da der menschliche Körper etwa 55 bis 60 % Wasser enthält muss in Gebäuden bzw. Fahrzeugen mit einer entsprechenden Signaldämpfung durch Personen gerechnet werden.

Der Datenaustausch per Funk ist ein komplexes Aufgabengebiet. Einerseits muss dafür gesorgt werden, dass die Daten fehlerfrei und zuverlässig an der angestrebten Gegenstelle empfangen werden und andererseits soll in der Regel auch verhindert werden, dass Unbefugte die gesendeten Daten abhören können oder gar in das Netzwerk eindringen können. Eine übliche Gliederung der Teilprozesse eines solchen Kommunikationssystems ist in Abb. 3.4 links dargestellt. Es handelt sich hierbei um das OSI-Modell [114], wobei OSI für Open Systems Interconnection steht, das von der ISO (International Organization for Standardization) entwickelt wurde. Dieses besteht aus sieben Schichten (Layern). Jede dieser Schichten übernimmt gewisse Aufgaben und nutzt hierfür jeweils die Funktionen der unterhalb liegenden Schichten.

Von der IEEE (Institute of Electrical and Electronics Engineers) wurde dieses Modell dahingehend modifiziert, dass der Link Layer nochmals, wie in Abb. 3.4 rechts dargestellt, in die Media Access Control und die Logic Link Control Schicht unterteilt wurde.

Publizierte Standards betreffen unterschiedliche Schichten, so deckt der Standard IEEE 802.15.4 (siehe Abschnitt 3.2.1.3) beispielsweise lediglich die physikalische und die MAC-Schicht ab. Ein funktionsfähiges Netzwerk kann in weiterer Folge jedoch nur aufgebaut werden, wenn die Funktionen der darüber liegenden Schichten ebenfalls definiert und implementiert werden. Hierfür kommen einerseits maßgeschneiderte Protokolle (Eigen- oder Fremdentwicklung) und andererseits weitere aufbauende Standards in Frage.

In den folgenden Abschnitten wird kurz auf die prinzipiell für das Messsystem in Fra-

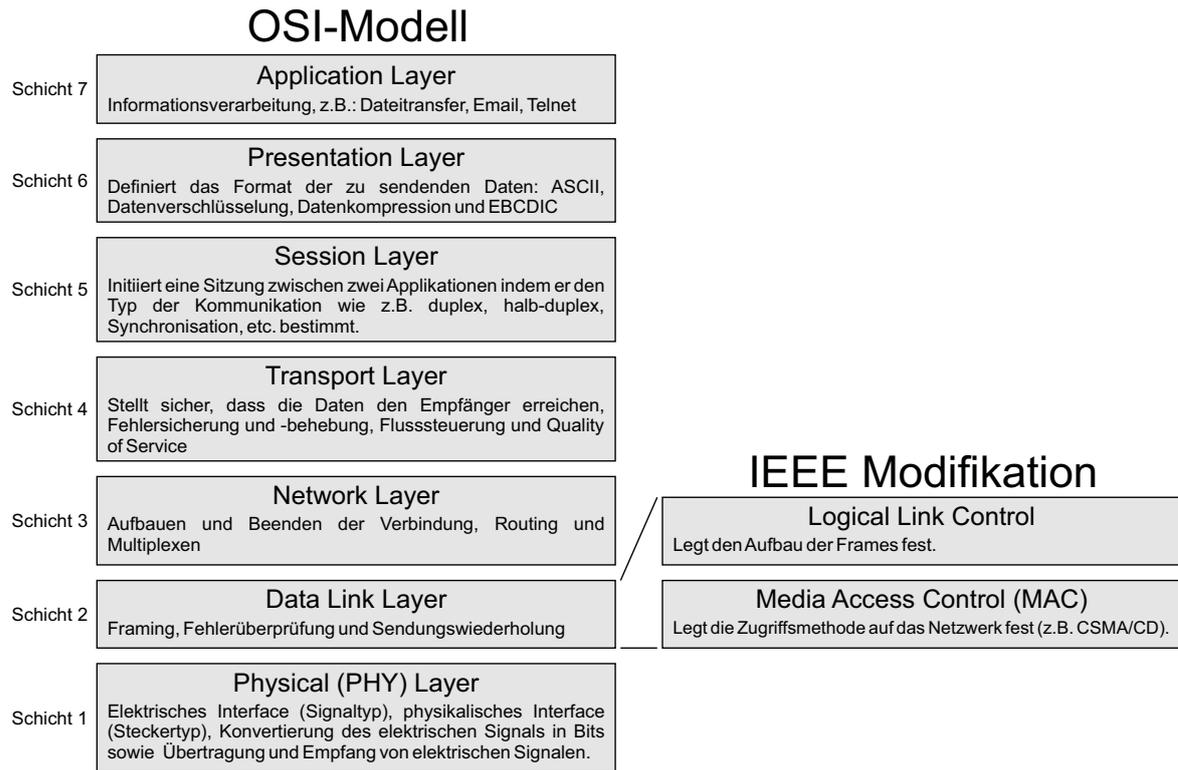


Abbildung 3.4: OSI-Modell mit IEEE-Erweiterung [18]

ge kommenden Funktechnologien eingegangen. Der Schwerpunkt liegt jedoch auf IEEE 802.15.4, da dieses in weiterer Folge tatsächlich eingesetzt wurde.

3.2.1.1 IEEE 802.11 – WLAN

WLAN steht für Wireless Local Area Network und bezeichnet eigentlich eine Netzwerkkategorie. Es handelt sich hierbei um ein lokales Funknetzwerk, dessen Ausdehnung in der Regel auf ein Gebäude bzw. Firmengelände beschränkt ist. Im allgemeinen Sprachgebrauch wird unter einem WLAN jedoch meist ein Netzwerk verstanden, das dem IEEE-Standard 802.11 entspricht. IEEE 802.11 beschreibt allerdings nur die physikalische und die MAC-Schicht, weshalb für den tatsächlichen Betrieb eines WLANs zusätzlich IEEE 802.2 (Logical Link Control) und TCP/IP (Transmission Control Protocol / Internet Protocol) benötigt werden. IEEE 802.11 wurde laufend erweitert, wobei die jeweiligen Entwicklungsstufen durch nachgestellte Buchstaben gekennzeichnet sind.

So wurde im Jahr 2003 beispielsweise 802.11g [32] und 2009 802.11n [33] verabschiedet. 802.11g bietet eine maximale Bruttodatenrate von 54 Mbit/s und 802.11n bis zu 600 Mbit/s, wobei bei 802.11n parallele Datenströme (max. 4) zum Einsatz kommen und

zusätzlich die Verwendung des 5 GHz-Bandes vorgesehen ist. Im 2.4 GHz-Bandes sind abhängig von den lokalen rechtlichen Bestimmungen bis zu 14 Kanäle verfügbar, wovon jedoch nur drei nahezu überlappungsfrei sind [45].

Entsprechend dem Standard [31] können sowohl Ad-hoc-Netze als auch Infrastrukturnetze gebildet werden. Bei Ad-hoc-Netzen erfolgt die Kommunikation direkt zwischen den Teilnehmern in Form von Peer-to-Peer-Verbindungen. Im Gegensatz dazu erfolgt die Kommunikation bei einem Infrastrukturnetz über eine stationäre Basisstation (Access Point). Es liegt also eine Sterntopologie zu Grunde. Die Basisstation fungiert dabei in der Regel auch als Router oder Gateway um eine Verbindung zu anderen Netzwerken zu ermöglichen. Zumeist werden kleine Heimnetze über die Basisstation ans Internet angebunden.

Die Kapazität und der Abdeckungsbereich eines WLANs lassen sich über den Mehrzellenbetrieb erweitern. Hierbei werden die Basisstationen über ein lokales kabelgebundenes Netzwerk verbunden, wobei sich die Abdeckungsbereiche der Funkzellen in der Regel überlappen. Die teilnehmenden Geräte werden bei einem Ortswechsel im Bedarfsfall von einer Basisstation an die benachbarte weitergegeben. In Summe erscheinen die über Kabel verbundenen Funkzellen für den Benutzer daher wie ein einziges ausgedehntes Funknetz.

Weiterführende Details zu IEEE 802.11 sind beispielsweise in [45] enthalten.

3.2.1.2 Bluetooth

Bluetooth [11] ist eine Funktechnologie zur Überbrückung kurzer Entfernungen mit dem Ziel Kabelverbindungen zwischen portablen und/oder ortsgebundenen elektronischen Geräten zu ersetzen. Die Hauptziele bei der Entwicklung waren vor Allem eine hohe Robustheit, ein geringer Energiebedarf sowie niedrige Kosten. Die maximale Bruttodatenübertragungsrate beträgt bei Bluetooth 3 Mbit/s.

Die hohe Robustheit soll im Wesentlichen durch ein als „Frequency Hopping“ bezeichnetes Verfahren erreicht werden. Hierbei erfolgt ein periodischer Wechsel zwischen den 79 Bluetooth Kanälen im 2.4 GHz-Band. Ein Funkkanal wird jeweils zwischen 625 μ s und 3.125 ms belegt, danach erfolgt ein Frequenzsprung. In welcher Reihenfolge die Frequenzen genutzt werden, hängt dabei von der Bluetooth-Device-Adresse des Masters ab. Durch die kurzzeitige Verwendung der einzelnen Kanäle wird die Koexistenz von Bluetooth mit anderen Funksystemen bzw. Störquellen im 2.4 GHz-Band erleichtert, da dauerhafte Datenverluste durch elektromagnetische Interferenz, wie sie z.B. bei zwei Funkverbindungen in überlappenden Frequenzbereichen auftreten würden, vermieden werden. Bluetooth bietet zusätzlich die Möglichkeit eines adaptiven Frequenzsprungverfahrens, bei dem belegte bzw. gestörte Kanäle nicht genutzt werden.

An einem Bluetooth-Funknetz, das auch als Piconet bezeichnet wird, können ein Master und maximal 7 Slaves aktiv beteiligt sein. Die Kommunikation erfolgt hierbei immer zwischen Master und Slave und nie direkt zwischen zwei Slaves. Als Netzstruktur ergibt sich daher für ein einzelnes Piconet die in Abb. 3.5 dargestellte Sterntopologie. Die Verknüpfung mehrerer Piconets wird als Scatternet bezeichnet. Dies ist möglich, da Slaves zeitversetzt (Time Division Multiplex) Mitglied verschiedener Piconets sein können. Analoges gilt für einen Master, wobei dieser jedoch nur in einem Piconet als Master und in den übrigen Netzen als Slave fungieren kann (siehe Abb. 3.5).

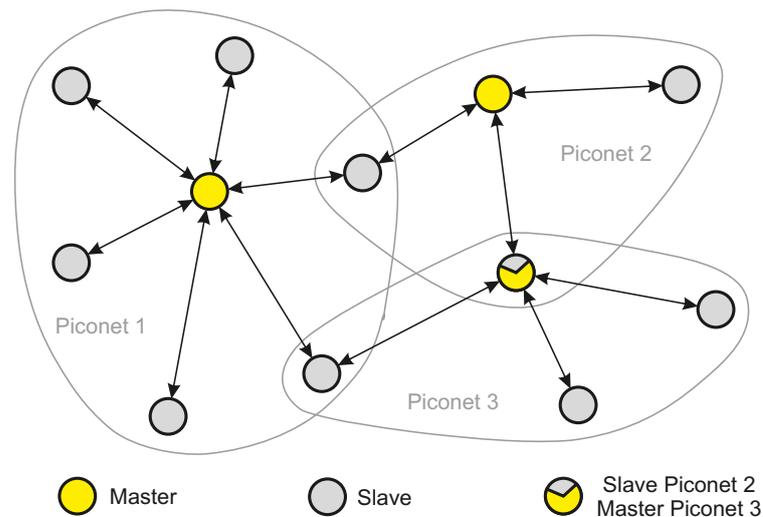


Abbildung 3.5: Bluetooth Pico- und Scatternets

Der Standard selbst enthält jedoch darüber hinaus keine Vorschriften oder Algorithmen zur Erzeugung eines Scatternets. Um ein solches zu bilden muss festgelegt werden welche Knoten jeweils ein Piconet bilden, welche Rollen die einzelnen Knoten übernehmen (Master oder Slave) und welche Knoten als Verbindungsglieder zwischen den Teilnetzen dienen. In weiterer Folge werden Algorithmen benötigt um einen Datenaustausch zwischen den einzelnen Knoten des Scatternets zu ermöglichen. Diese werden ebenfalls nicht durch den Standard abgedeckt und sind Gegenstand der Forschung (z.B. [67]).

Weiterführende Informationen sind in [11] bzw. auch in [45] enthalten.

3.2.1.3 IEEE 802.15.4

Der Standard IEEE 802.15.4 behandelt Personal Area Networks (PANs) mit niedriger Datenübertragungsrate. Im Standard [35] ist bezüglich der wesentlichen Eigenschaften der Funktechnologie folgendes ausgeführt:

The standard provides for ultra low complexity, ultra low cost, ultra low power consumption, and low data rate wireless connectivity among inexpensive devices. The raw data rate is high enough (250 kbit) to satisfy a set of applications but is also scaleable down to the needs of sensor and automation needs (20 kbit) for wireless communications.

Der Standard bietet also eine drahtlose Kommunikation mit niedrigen Datenraten zwischen preisgünstigen Geräten bei sehr niedriger Komplexität, sehr niedrigen Kosten und sehr geringem Energiebedarf. Die Brutto-Datenübertragungsrate ist hoch genug (250 kbit/s) um eine drahtlose Kommunikation bei vielen Applikationen zu ermöglichen und kann für Sensor- und Automatisierungsaufgaben nach unten hin angepasst werden (20 kbit/s oder niedriger).

IEEE 802.15.4 definiert hierfür verschiedene physikalische Schichten, die in unterschiedlichen Frequenzbändern positioniert sind. Im Folgenden wird ausschließlich auf die im 2.4 GHz-Band definierte OQPSK-Modulation (OQPSK – Offset Quadrature Phase-Shift Keying) eingegangen, da derzeit in diesem Frequenzbereich lediglich für diese Modulation entsprechende Transceiver erhältlich sind.

Generell definiert der Standard zwei Arten von Geräten, die in einem Netzwerk vorhanden sein können: Full Function Devices (FFD) und Reduced Function Devices (RFD). Ein FFD bietet die volle Funktionalität und kann als Koordinator die Verwaltung des Netzwerks übernehmen. RFDs sind für sehr einfache Anwendungen wie z.B. Lichtschalter vorgesehen und bieten dementsprechend eine minimale Netzwerkfunktionalität. Die Anforderungen an die Hardware und der benötigte Arbeitsspeicher sind daher gering, wobei jedoch bei dieser Geräteklasse die Einschränkung besteht, dass RFDs nur mit einem FFD assoziiert sein können und nur mit dem zugehörigen FFD direkt kommunizieren können.

Im 2.4 GHz-Band stehen 16 Kanäle zur Verfügung, die mit Kanalnummer und Mittenfrequenz in Abbildung 3.6 schematisch dargestellt sind. Hinsichtlich der Netzwerktopologie werden durch den Standard die Sterntopologie und Peer-to-Peer-Netzwerke (siehe Abbildung 3.7) unterstützt.

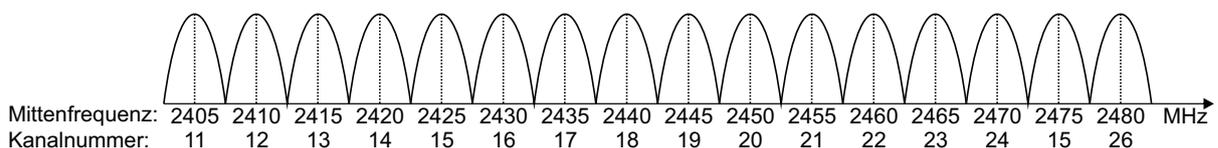


Abbildung 3.6: IEEE 802.15.4 Funkkanäle im 2.4 GHz-Band nach [18]

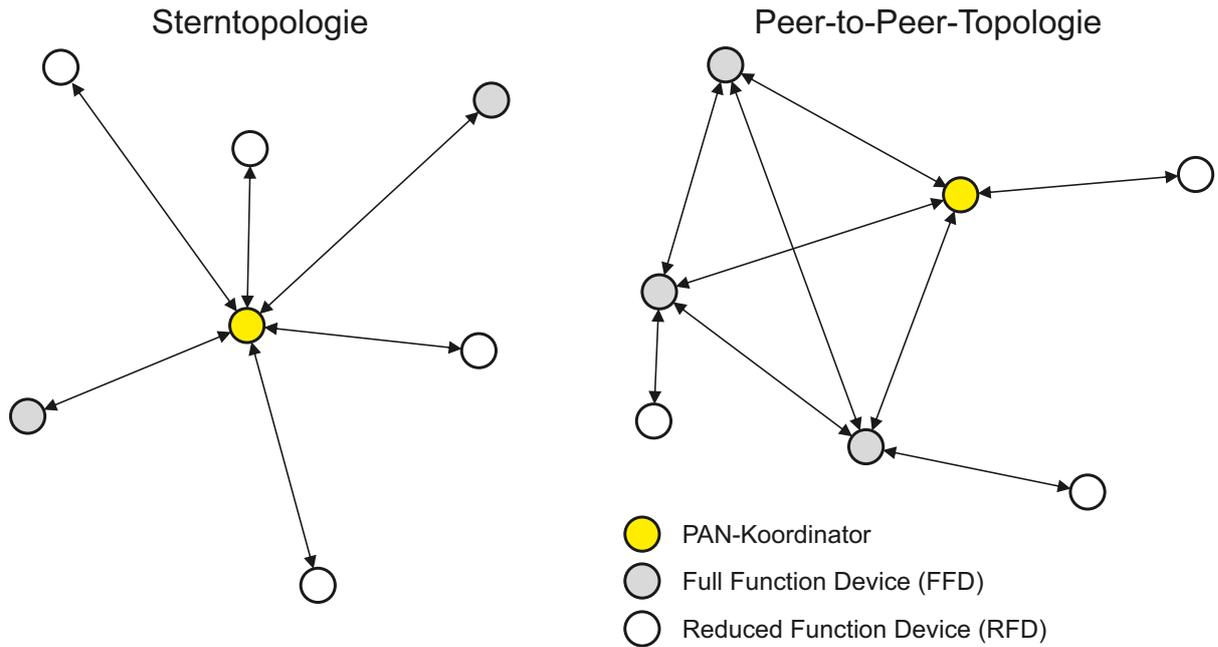


Abbildung 3.7: Stern- und Peer-to-Peer-Topologie

Der Standard IEEE 802.15.4 definiert lediglich die physikalische (PHY) und die MAC-Schicht (siehe Abbildung 3.8). Der allgemeine Aufbau eines IEEE 802.15.4 Pakets ist in Abbildung 3.9 dargestellt. Jede der Schichten generiert Daten die an die darunter liegende Schicht weitergeleitet werden. Die nach unten übergebenen Daten stellen hierbei jeweils die Nutzlast (Payload) der darunterliegenden Schicht dar. In der Regel bettet die jeweilige Schicht die Nutzlast zwischen einem Header und Footer ein und übergibt den resultierenden Datenstrom wiederum an die unterhalb liegende Schicht. Die Daten werden also quasi eingerahmt, weshalb man hier auch von Frames (Rahmen) spricht. Im konkreten Fall beschreibt der Standard 802.15.4 lediglich die Bitarbitrierungs- bzw. die physikalische Schicht und die MAC-Schicht. Das Paket besteht daher aus den Anteilen

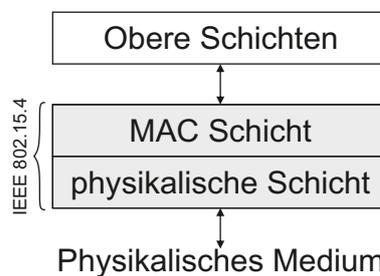


Abbildung 3.8: IEEE 802.15.4 Schichten

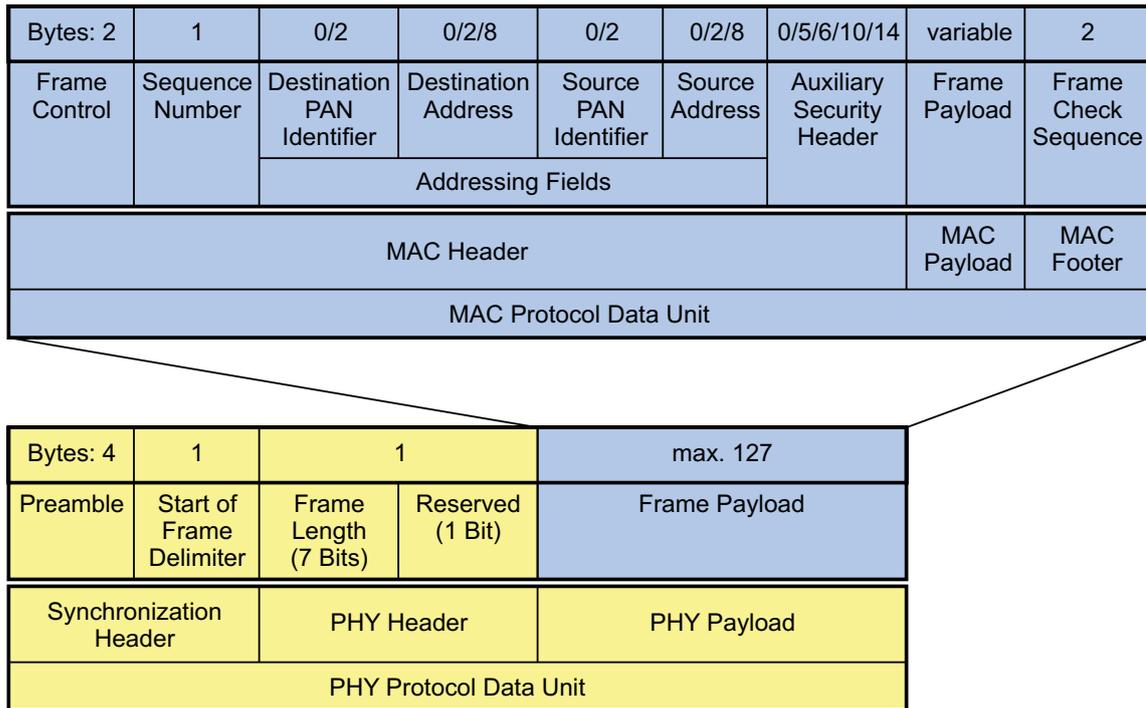


Abbildung 3.9: Aufbau des IEEE 802.15.4 MAC-Datenrahmens nach [34]

der physikalischen Schicht und der MAC-Schicht, wobei der Anteil der MAC-Schicht die Nutzlast des Frames der physikalischen Schicht darstellt.

Am Beginn des Pakets steht der Synchronisierungheader, der aus Preamble und Start of Frame Delimiter (SFD) besteht. Die Präambel dient zur Synchronisation im Empfänger und durch den SFD wird der Beginn des Headers der physikalischen Schicht markiert. Die Länge des gesamten Pakets wird mit Hilfe von 7 Bits im PHY-Header kodiert. Anschließend wird die Nutzlast der physikalischen Schicht (PHY-Payload) übertragen.

Diese Nutzlast besteht wiederum aus den von der MAC-Schicht übergebenen Daten. Sie enthalten den MAC-Header, die MAC-Payload und den MAC-Footer. Der Typ des MAC-Frames (Beacon, Data, Acknowledgement oder MAC-Command) sowie der weitere Inhalt des Headers werden über das Feld „Frame Control“ festgelegt. Weiters enthält der MAC-Header eine Sequenznummer (Sequence Number), die nach jedem gesendeten Paket um Eins erhöht wird. Über diese Nummer können eingelangte Empfangsbestätigungen (Acknowledge) konkret einem zuvor abgesendeten Paket zugeordnet werden. Empfänger- und Absenderadresse inklusive der jeweiligen Netzwerk-ID (PAN ID) sind ebenfalls Bestandteil des Headers. Nach dem Header folgen die Nutzlast des MAC-Frames und in der Frame Check Sequence eine mittels CRC (Cyclic Redundancy Check) generierte Prüfsumme.

Zur Absicherung gegen den Verlust von Nachrichten sind nach dem Standard Empfangsbestätigungen (Acknowledges) vorgesehen. Durch das Setzen eines Bits im Feld Frame Control des MAC-Headers kann durch den Sender eine Empfangsbestätigung vom Sender gefordert werden. Bleibt diese aus, so wird die Nachricht erneut gesendet. Nach dem Standard sind hier je nach Konfiguration bis zu sieben neuerliche Abstrahlungen der Nachricht vorgesehen.

Eine Ausführliche Behandlung von IEEE 802.15.4 würde den Rahmen dieser Arbeit sprengen. Es wird daher an dieser Stelle diesbezüglich vor allem auf [34] bzw. mit Einschränkungen auch auf [18, 28] verwiesen.

Wie bereits erläutert, definiert der Standard lediglich die beiden untersten Schichten des IEEE-Modells. Ein standardisiertes Protokoll, das auf 802.15.4 aufbaut und Mesh-Netzwerkfunktionalität zur Verfügung stellt, ist beispielsweise ZigBee (siehe Abschnitt 3.2.2). Hersteller von IEEE 802.15.4-konformen Funk-Transceivern (z.B. Texas Instruments, Freescale, Microchip, etc.) bieten für ihre Mikrocontroller in der Regel neben proprietären Protokoll-Stacks auch entsprechende ZigBee-Stacks an. Für eine verhältnismäßig rasche Implementierung eines Mesh-Netzwerkes bietet sich daher vor allem ZigBee an.

Im Bereich der drahtlosen Sensornetze wird in Kombination mit IEEE 802.15.4-konformen Transceivern häufig das Open Source Betriebssystem TinyOS [26] eingesetzt. Entsprechend der regen Forschungstätigkeit im Bereich der drahtlosen Sensornetze ist der Sourcecode vieler Protokolle für TinyOS verfügbar [57]. Da TinyOS im Rahmen dieser Arbeit nicht eingesetzt werden konnte (siehe Abschnitt 4.1), wird in weiterer Folge auf TinyOS nicht näher eingegangen. Die entsprechenden Grundlagen können beispielsweise in [47] nachgelesen werden.

3.2.1.4 IEEE 802.11, Bluetooth und IEEE 802.15.4 im Vergleich

Neben der erzielbaren Datenrate ist für batteriebetriebene Geräte vor allem der zu erwartende Strombedarf ein wesentliches Kriterium für die Wahl eines Funkstandards. In Tabelle 3.1 sind beispielhaft die typischen Werte von Funkmodulen der Firma Microchip mit einem vergleichbaren Link Budget von ca. 95 dBm angeführt. Da das Link Budget angibt welche Funkausbreitungsdämpfung maximal zwischen Sender und Empfänger zulässig ist, ist es ein Maßstab für die überbrückbare Entfernung.

Unter Vernachlässigung des Antennengewinns sowie der Kabel- und Anschlussverluste kann das Link Budget LB aus der Differenz zwischen der Sendeleistung P_{TX} und der

Funkstandard	Modul	E_{RX}	P_{TX}	I in mA		Datenrate
		in dBm	in dBm	RX	TX	in kBit/s
IEEE 802.11bg	RN171	-83	12	40	180	54 000
Bluetooth	RN41	-80	16	35	65	3000
IEEE 802.15.4	MRF24J40MA	-94	0	19	23	250

Tabelle 3.1: Richtwerte für die Empfindlichkeit des Empfängers, der Sendeleistung, des Strombedarfs beim Empfangen (RX) und Senden (TX) bei +3.3 V sowie der maximalen Datenrate von IEEE 802.11bg, Bluetooth und IEEE 802.15.4 anhand von Modulen der Firma Microchip

Empfindlichkeit des Empfängers E_{RX} näherungsweise bestimmt werden:

$$LB \doteq P_{TX} - E_{RX}. \quad (3.1)$$

Bezüglich näherer Details zum Link Budget wird auf [45] verwiesen.

Wie in Tabelle 3.1 ersichtlich, steigt der Strombedarf mit der maximal erzielbaren Datenübertragungsrate deutlich an. IEEE 802.15.4 weist deutlich niedrigere Werte im Empfangsmodus auf, der in der Regel auch der vorherrschende Betriebszustand ist. Auf Grund der deutlich höheren Datenübertragungsrate relativiert sich der Strombedarf beim Senden, da die zum Absetzen der Nachricht benötigte Zeit damit natürlich auch sinkt.

Der geringe Energiebedarf und die Möglichkeit der Peer-To-Peer-Kommunikation, die den einfachen Aufbau von Mesh-Netzwerken ermöglicht, begründen den häufigen Einsatz von IEEE 802.15.4 im Bereich der drahtlosen Sensornetzwerke. In weiterer Folge waren diese beiden Eigenschaften auch ausschlaggebend für die Wahl von IEEE 802.15.4 für das geplante Messsystem.

3.2.1.5 Koexistenz von IEEE 802.15.4 mit anderen Funktechnologien

Mit der steigenden Zahl von Geräten, die im 2.4 GHz-Band per Funk kommunizieren, steigt die Wahrscheinlichkeit von Daten-Paketverlusten durch elektromagnetische Interferenz ebenfalls entsprechend an. Die Koexistenz dieser Technologien gewinnt daher immer mehr an Bedeutung.

Die geringe Sendeleistung von IEEE 802.15.4 führt zwar zu einem geringen Energiebedarf, sie macht die Technologie jedoch auch anfälliger für Störungen durch Sender mit deutlich höherer Sendeleistung. Entsprechend [80] ist hier in erster Linie IEEE 802.11 kritisch. Die Ergebnisse in [80] haben gezeigt, dass im schlimmsten Fall bei sehr hoher

Auslastung von IEEE 802.11b und überlappenden Frequenzen bei gleichzeitigen in unmittelbarer Nähe betriebenen IEEE 802.15.4 Funkstrecken 92 % der Pakete gestört werden. Da WLAN-Netze jedoch zumeist auf den überlappungsfreien Kanälen 1, 6 und 11 arbeiten, verbleiben für IEEE 802.15.4 die Kanäle 15, 20, 25 und 26 in der Regel frei für einen störungsfreien Betrieb [88].

Berücksichtigt man, dass in Straßenbahnen in der Regel nicht mit dem gleichzeitigen Betrieb von mehreren WLAN-Netzen mit hoher Auslastung zu rechnen ist und der Betreiber des Fahrzeuges gegebenenfalls Einfluss auf das Datenaufkommen des/der WLAN-Netze bzw. ebenfalls auf die belegten Frequenzbereiche hat, so ist beim Einsatz von IEEE 802.15.4 in Straßenbahnen in der Praxis nicht mit grundsätzlichen Problemen zu rechnen.

3.2.2 ZigBee

Von der ZigBee Alliance wurde für Anwendungen in der Gebäude- und Heimautomatisierung, in der Unterhaltungs- und Haushaltselektronik, in der Medizintechnik, in der Telekommunikation, im Energiemanagement (Stichwort Smart Meters), etc. der ZigBee-Standard entwickelt, der auf IEEE 802.15.4 aufsetzt. Abbildung 3.10 zeigt den prinzipiellen Aufbau des Protokoll-Stacks inklusive der jeweiligen Verantwortungsbereiche. Die eigentliche Funktionalität des Geräts wird hierbei durch bis zu 240 Applikationsobjekte

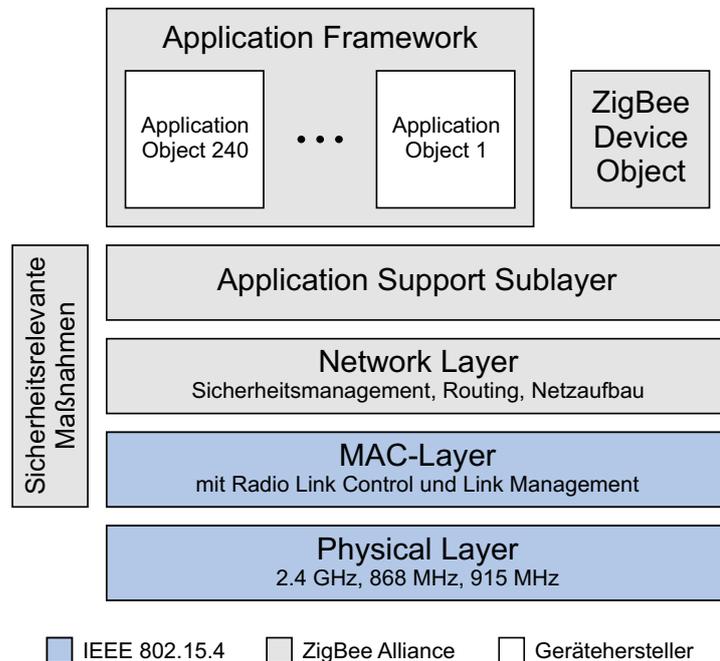


Abbildung 3.10: ZigBee Protokollarchitektur nach [45]

(Application Objects), die vom Hersteller des Gerätes zu implementieren sind, festgelegt.

Um die Interoperabilität von Geräten verschiedener Hersteller zu gewährleisten wurden so genannte Applikationsprofile (Application Profiles) entwickelt. Diese legen für bestimmte Anwendungen den Aufbau der ausgetauschten Nachrichten im Detail fest. Dadurch wird erreicht, dass mit einem Lichtschalter oder einer Fernbedienung des Herstellers A die Lampe des Herstellers B ein- und ausgeschaltet werden kann. Natürlich unter der Voraussetzung, dass die Geräte beider Hersteller das entsprechende Applikationsprofil nützen. Eine von den Applikationsprofilen vollständig abgekoppelte Entwicklung ist natürlich ebenfalls möglich. Dies kann erforderlich sein, wenn z.B. für die entsprechende Anwendung kein entsprechendes Applikationsprofil zur Verfügung steht, die bestehenden Profile die Funktionalität des Produktes einschränken würden, oder wenn ein geschlossenes proprietäres System geschaffen werden soll.

Der Standard beinhaltet derzeit zwei Varianten des Protokolls: ZigBee und ZigBee PRO. Einer der grundlegenden Vorteile von ZigBee PRO gegenüber ZigBee ist, dass die Adressen der Knoten stochastisch vergeben werden und sich dadurch keine wesentliche Beschränkung der Ausdehnung des Netzwerkes auf Grund des Adressvergabeschemas ergibt. Der bei ZigBee eingesetzte Cskip-Algorithmus [28] beruht auf einer hierarchischen Eltern-Kind-Beziehung der Knoten untereinander und reserviert für jeden Elternknoten eine bestimmte Anzahl an Adressen für dessen Kinder. Die Anzahl der reservierten Adressen ist dabei unabhängig davon ob diese tatsächlich benötigt werden. Da diese Adressen somit quasi an die nähere Umgebung eines Knotens gebunden sind, stehen sie an anderer Stelle nicht für eine Erweiterung des Netzes zur Verfügung. Konkret können sich zwischen einem beliebigen Knoten und dem Koordinator maximal drei Zwischenknoten befinden.

In weiterer Folge wird ZigBee synonym für ZigBee und ZigBee PRO verwendet, da die weiteren Ausführungen für beide Varianten gelten – sofern nicht explizit auf Unterschiede hingewiesen wird.

Neben der 16 Bit-Netzwerkadresse, die jedes Gerät bei jedem Beitritt zu einem Netzwerk erneut erhält, muss jeder Knoten auch eine vom Hersteller fix vergebene 64 Bit lange, so genannte MAC-Adresse, besitzen. Diese 64 Bit-Adresse wird für eine eindeutige Identifikation der Geräte benötigt, da die vergebene Netzwerkadresse sich bei jedem Beitritt ändern kann. Die MAC-Adresse muss eindeutig sein, das heißt, es darf weltweit nur ein Gerät mit einer bestimmten Adresse existieren. Erreicht wird dies dadurch, dass von der IEEE, 24 Bit lange Organizational Unique Identifiers (OUI) gegen Gebühr vergeben werden. Dieser OUI bildet den höherwertigen Abschnitt der MAC-Adresse, die übrigen Bits werden vom Hersteller selbst bestimmt.

In einem ZigBee-Netzwerk können drei Geräteklassen existieren: Koordinator, Router und Endgerät. Pro Netzwerk muss immer exakt ein Koordinator vorhanden sein, der für den Aufbau des Netzwerkes verantwortlich ist. Dem Netzwerk können in weiterer Folge Router und/oder Endgeräte beitreten, wobei nur Router als Zwischenknoten fungieren und Nachrichten weiterleiten können. Abb. 3.11 zeigt beispielhaft einen möglichen Aufbau eines solchen Netzwerkes.

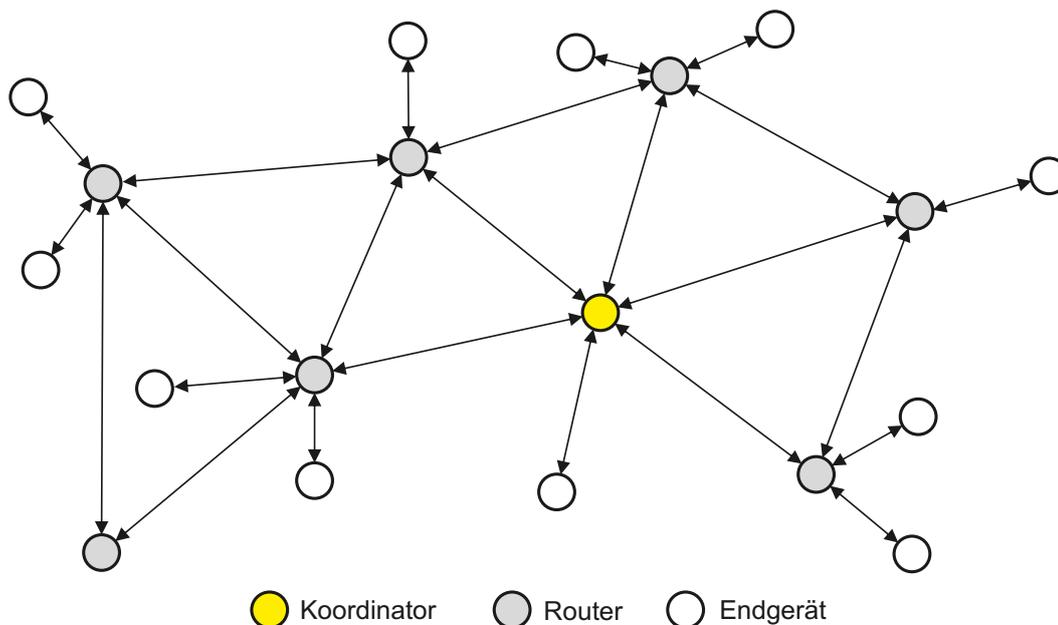


Abbildung 3.11: Beispiel eines ZigBee-Netzwerkes

Im Betrieb erfolgt die Adressierung im ZigBee-Netzwerk mittels der 16 Bit-Netzwerkadresse. Neben der Adressierung eines konkreten Geräts im Netzwerk können ebenfalls Gruppen bzw. alle Geräte im Netzwerk mittels Group- bzw. Broadcasts angesprochen werden. Der Radius eines Broadcasts lässt sich dabei mittels eines Parameters auf eine bestimmte Anzahl von Zwischenknoten, die die Nachricht erneut ausstrahlen, beschränken. Zu beachten ist hierbei, dass Broadcasts nicht über Empfangsbestätigungen abgesichert werden, wie dies bei einer, an einen bestimmten Knoten, adressierten Nachricht der Fall ist. Dies geschieht einerseits automatisch durch den MAC-Layer, der bei einem Ausbleiben der Empfangsbestätigung des unmittelbaren Empfängers die Nachricht bis zu drei Mal erneut sendet. Und andererseits auf Anforderung durch das Application Object zusätzlich auch von Endpunkt zu Endpunkt der gesamten Route des Funkpakets. Hier sind ebenfalls maximal drei erneute Sendeversuche durch den Absender der Nachricht vorgesehen, bevor der Sendevorgang mit einer Fehlermeldung beendet wird.

Entsprechend [28] ist mit ZigBee eine Bruttodatenübertragungsrate von etwa 25 kBit/s zu erreichen, wobei für jede einzelne Teilstrecke (Sender zu Empfänger) der Route durch das Netzwerk etwa 10 ms zu veranschlagen sind.

Details wie das Finden von Routen, die Sicherheitsfeatures, etc. kann der zahlreichen Literatur [17] [20] [21] [28] bzw. dem Standard selbst entnommen werden.

3.2.3 Akkumulortechnologien

Um 1800 herum entdeckte Alessandro Volta, dass ein kontinuierlicher Stromfluss zwischen zwei, in einem Elektrolyten eingetauchten, metallischen Elektroden fließt und legte damit den Grundstein für die heutige Batterie- und Akkumulortechnologie. Batterien werden oft auch als Primärzellen und wiederaufladbare Akkumulatoren als Sekundärzellen bezeichnet. Galvanische Zellen können prinzipiell mit einer Vielzahl von chemischen Elementen aufgebaut werden. Die derzeit gebräuchlichsten Zellsysteme bei Akkumulatoren sind die Blei-, die Nickel-Metallhydrid- (NiMH-) und die Lithium-Ionen-Zelle. Abhängig vom Zellsystem besitzen Primär- und Sekundärzellen jeweils unterschiedliche Leerlaufspannungen sowie eine unterschiedliche spezifische Energie und spezifische Leistung. Bei vielen Anwendungen, beispielsweise im Automobilbereich (Hybridantrieb) oder bei Power-Tools (Akku-Bohrmaschinen, etc.), ist gleichzeitig eine hohe spezifische Energie und Leistung gefordert. Wie das Ragone-Diagramm in Abb. 3.12 zeigt, sind dies jedoch konträre Eigenschaften. Zellen, die für eine hohe spezifische Leistung optimiert sind, bieten diese lediglich mit Abstrichen hinsichtlich der spezifischen Energie.

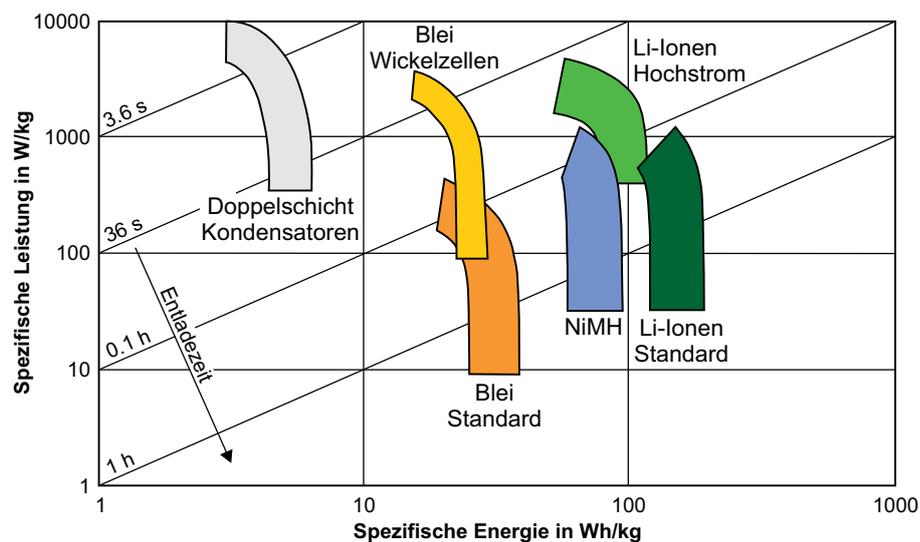


Abbildung 3.12: Ragone Diagramm nach [38]

Lithium-Ionen-Zellen verfügen derzeit über die höchste spezifische Energiedichte und werden deshalb in vielen mobilen Geräten (Mobiltelefonen, Laptops, Tablets, MP3-Playern, etc.) eingesetzt. Im Vergleich zu NiMH-Zellen erfordert der sichere Einsatz von Lithium-Ionen-Zellen jedoch einen deutlich höheren schaltungstechnischen Aufwand. Eine aktive Strombegrenzung, ein Tiefentladeschutz, sowie ein Schutz vor Überspannung sind bei Lithium-Ionen-Zellen unbedingt vorzusehen. Weiters ist bei einer Serienschaltung von Lithium-Ionen-Zellen eine aktive Überwachung und ein Abgleich des Ladezustandes der einzelnen Zellen erforderlich (Balancing). Fehlen oder versagen die entsprechenden Schutzeinrichtungen, so ist im schlimmsten Fall mit einer Zellöffnung mit Flammerscheinung (venting with flame) zu rechnen. Entsprechend dem höheren Gefahrenpotential im Vergleich zu z.B. Nickelzellen gelten für Lithiumzellen daher auch strenge Transportvorschriften [38].

Die nutzbare Kapazität einer Zelle hängt im Wesentlichen von der Temperatur, dem Entladestrom und dem Zustand der Zelle (Alterung) ab. Da sowohl chemische Reaktionen als auch Diffusionsvorgänge bei höheren Temperaturen schneller ablaufen und die Leitfähigkeit von Elektrolyten ebenfalls mit der Temperatur ansteigt, steigt die nutzbare Kapazität einer Zelle prinzipiell mit der Temperatur. Neben den eigentlichen Lade- und Entladereaktionen ergeben sich allerdings auch raschere, unerwünschte Nebenreaktionen, was zu einer erhöhten Selbstentladung und einer verstärkten Alterung führt. Bis etwa 40 °C steigt beispielsweise bei NiMH-Zellen die tatsächliche Kapazität an, wohingegen sie oberhalb dieser Temperatur fällt sie jedoch wieder ab [38].

Das Absinken der nutzbaren Kapazität bei einer Erhöhung des Entladestromes ist unter anderem darauf zurück zu führen, dass die Transportvorgänge in der Zelle eine gewisse Trägheit besitzen. Weiters steigen mit dem Strom natürlich auch die ohmschen Verluste an. Der Lade- bzw. Entladestrom wird häufig als C-Rate angegeben. 1 C entspricht hierbei der Nennkapazität des Akkumulators geteilt durch eine Stunde. Entlädt man beispielsweise einen Akkumulator mit einer Nennkapazität von 2000 mA h bei einem Strom von 1000 mA, so kann dies äquivalent als Entladung mit 0.5 C beschrieben werden.

Abhängig vom jeweiligen Zelltyp kommen unterschiedliche Ladeverfahren zum Einsatz, wobei an dieser Stelle nicht auf die Besonderheiten der einzelnen Akkumulatortypen eingegangen wird. Nähere Details bezüglich der Funktionsweise und der Verwendung von Akkumulatoren können z.B. [38] entnommen werden.

Auf Grund der Sicherheitsproblematik bei Lithium-Akkumulatoren kommen diese als Energiequelle für ein Messsystem, das in einem Massentransportmittel eingesetzt werden soll, nicht in Betracht. Wie das Ragone-Diagramm zeigt, stehen NiMH-Zellen unter den

gebräuchlichen Technologien hinsichtlich der spezifischen Energie an zweiter Stelle. Wegen ihrer guten Verfügbarkeit und ihres verhältnismäßig unkomplizierten Verhaltens fiel die Wahl daher auf diesen Akkumulatortyp. Im folgenden Abschnitt wird daher kurz auf die, für den konkreten Einsatz, wesentlichen Eigenschaften von NiMH-Zellen eingegangen.

3.2.3.1 NiMH-Akkumulatoren

NiMH-Zellen sind im Temperaturbereich von -20 bis $+45$ °C einsetzbar [49], wobei bei -20 °C nur etwa 80 % der Nennkapazität zur Verfügung stehen [38].

Durch Selbstentladung reduziert sich bei herkömmlichen NiMH-Zellen bei einer Temperatur von $+20$ °C nach 30 Tagen die nutzbare Kapazität auf ca. 75 bis 80 % des Nennwertes [38]. Dieser Effekt ist temperaturabhängig, wobei die Höhe der Selbstentladung mit steigender Temperatur zunimmt. Bei 0 °C sind beispielsweise nach 30 Tagen etwa 90 bis 95 % und bei $+40$ °C nur noch ca. 50 bis 70 % der Nennkapazität verfügbar. Moderne NiMH-Zellen wie z.B. die Eneloop von Sanyo sind speziell mit dem Ziel einer niedrigen Selbstentladung entwickelt worden. Ein Eneloop-Akkumulator in der Bauform AA (HR-3UTGA) verfügt nach einem Jahr Lagerung bei Raumtemperatur noch über eine entnehmbare Kapazität von etwa 85 %.

Weiters ist die entnehmbare Kapazität vom Entladestrom abhängig. Abbildung 3.13 zeigt den typischen Verlauf der Zellspannung einer Eneloop-Zelle in Abhängigkeit von Ladezustand und Entladestrom. Die Abhängigkeit der insgesamt nutzbaren Kapazität vom Entladestrom ist hier ebenfalls deutlich erkennbar.

Typischerweise erfolgt die Ladung von NiMH-Zellen mit konstantem Ladestrom. Die vollständige Ladung der Zelle wird in der Regel über die Zellenspannung erkannt. Unmittelbar vor dem Erreichen des Vollladezustandes weist die Zellenspannung ein Maximum auf. In weiterer Folge kann über den anschließenden Abfall dieser Spannung das Ende des Ladevorganges bestimmt werden. Dies wird als $-\Delta U$ -Kriterium bezeichnet. Der typische Verlauf der Zellspannung bei einem Ladevorgang mit konstantem Ladestrom unter Verwendung des $-\Delta U$ -Kriteriums zur Erkennung der Vollladung ist in Abbildung 3.14 dargestellt. Nähere Informationen bezüglich Ladeverfahren und Erkennung der Vollladung sind in [38] ausgeführt.

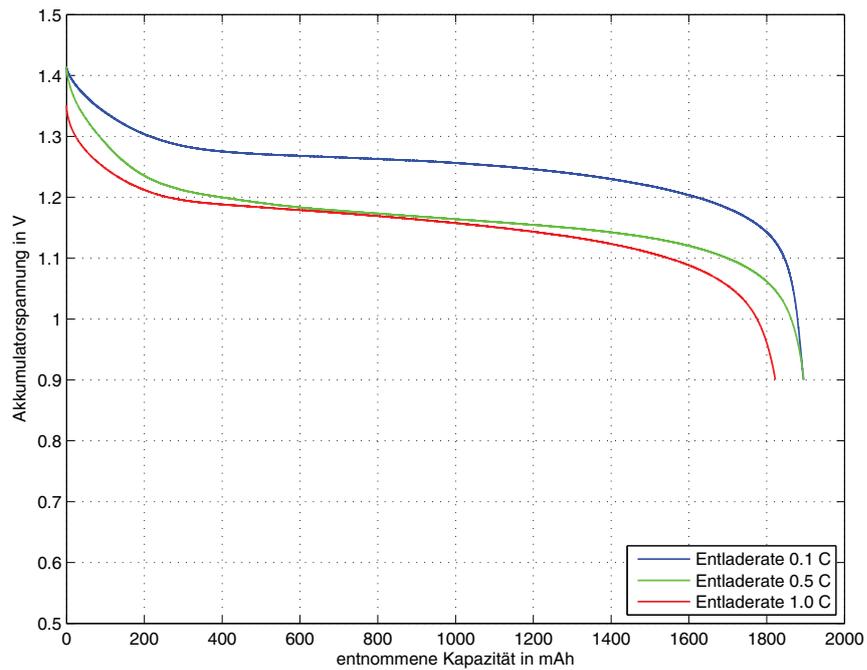


Abbildung 3.13: Spannung einer Sanyo Eneloop HR-3UTG AA-Zelle mit einer typischen Nennkapazität von 2000 mA h bei verschiedenen Entladeraten

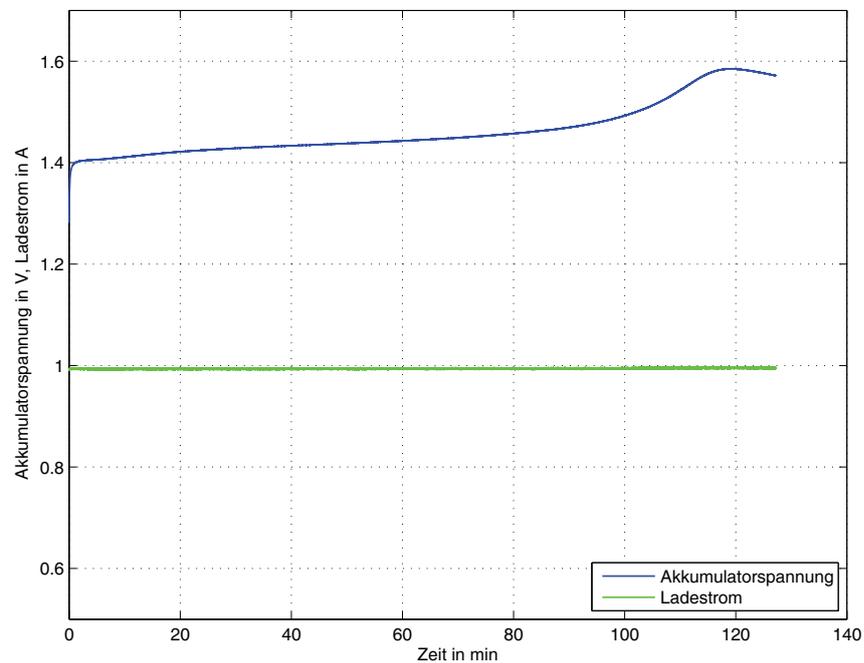


Abbildung 3.14: Ladezyklus einer Sanyo Eneloop HR-3UTG AA-Zelle beim Laden mit 0.5 C

3.2.4 Flash-Basierte Speicherkarten

Flash-Speicher sind kompakte und robuste nichtflüchtige Speichermedien, die aus dem täglichen Leben nicht mehr wegzudenken sind. Ihre Robustheit beruht vor allem darauf, dass die Information in Form von Ladungen mittels eines Floating-Gate-Transistors gespeichert wird. Da hier keine beweglichen Teile wie z.B. bei Festplatten vorhanden sind, sind Flash-Speicher weitgehend unempfindlich gegenüber Erschütterungen.

Abhängig von der Verschaltung der einzelnen Speicherzellen wird zwischen NAND- und NOR-Flash unterschieden. NAND-Flash wird auf Grund der geringeren effektiven Fläche der einzelnen Zellen hauptsächlich in Massenspeichern eingesetzt. Es hat jedoch den wesentlichen Nachteil, dass der wahlfreie Zugriff auf einzelne Zellen deutlich langsamer ist als beim NOR-Flash und, dass das Löschen und Schreiben nur blockweise erfolgen kann. Die Anzahl der Lösch- und Schreibzyklen ist beim NAND-Flash je nach konkreter Technologie auf etwa 10 000 bis 100 000 Zyklen beschränkt [63].

Abb. 3.15 zeigt den prinzipiellen Aufbau einer Flash-Speicherkarte. Sie besteht im Wesentlichen aus einem Controller und den eigentlichen NAND-Speichern. Der Controller wiederum besteht aus einem Speicherinterface zum Ansteuern des Speichers, einem Protokollinterface, das für die Kommunikation nach außen verantwortlich ist, und dem eigentlichen Mikrocontroller. Der Controller ist in Summe dafür verantwortlich, dass die vom Lesegerät angeforderten oder gesendeten Daten vom Flash-Speicher abgerufen oder dort gespeichert werden. Üblicherweise übernimmt der Controller auch die Aufgabe des Wear-Levellings, das dafür sorgen soll, dass die Blöcke der Flash-Speicher möglichst gleichmäßig durch die Lösch- und Schreibzyklen belastet werden. So soll vermieden werden, dass bestimmte Speicherbereiche deutlich öfter beschrieben werden als andere und dadurch entsprechend früher das Ende ihrer Lebensdauer erreichen.

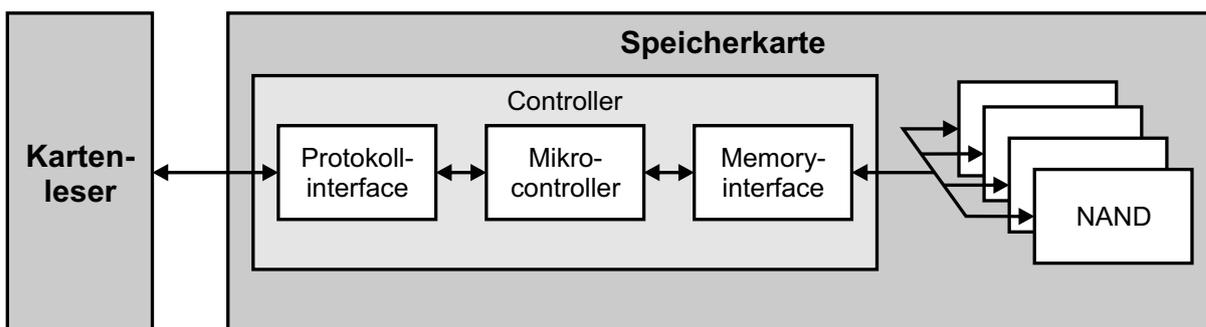


Abbildung 3.15: Aufbau einer Flash-Speicherkarte nach [63]

Die gebräuchlichsten Typen von Flash-Speicherkarten sind:

- SmartMedia (SM),
- Multimedia Card (MMC),
- CompactFlash (CF),
- Memory Stick (MS),
- xD-Picture Card (xD) und
- Secure Digital (SD).

Die Vielzahl der Typen rührt daher, dass die Hersteller versuchten ihre eigenen Entwicklungen zu etablieren. So wurde z.B. die Smart Media Card von Toshiba, die Multimedia Card von Igentix und SanDisk, die CompactFlash Card von Sandisk, der Memory Stick von Sony und die xD-Picture Card von Olympus und FujiFim entwickelt. Leider sind die zugehörigen Spezifikationen in der Regel nicht frei erhältlich. Die CF-Spezifikation ist mit 100 USD noch relativ erschwinglich, um Zugriff auf die vollständige SD-Spezifikation zu erhalten muss man jedoch der SD Association beitreten, womit ein Mitgliedsbeitrag in der Höhe von 2000 USD pro Jahr verbunden ist.

Auf Grund der weiten Verbreitung von SD-Karten, deren kompakten Abmessungen und der Tatsache, dass hier eine vereinfachte Spezifikation [5] frei erhältlich ist, sind SD-Karten für Projekte mit verhältnismäßig geringen Anforderungen an die Lese- und Schreibraten gut geeignet. Der geringe Datendurchsatz resultiert daher, dass im Rahmen der vereinfachten Spezifikation mittels des SPI-Busses (SPI – Serial Peripheral Interface) auf die Karte zugegriffen wird.

Der SD-Standard ließ lediglich Karten bis zu einer Größe von 2 GB zu und wurde daher 2006 erweitert. Die neu spezifizierte Secure Digital High Capacity (SDHC) Card ermöglicht Kapazitäten bis zu 32 GB. Die physikalischen Abmessungen blieben unverändert.

Derzeit existieren drei Gehäusevarianten, die normale SD(HC)-Karte (24 mm x 32 mm), die miniSD(HC)-Karte (20 mm x 21.5 mm) und die microSD(HC)-Karte (11 mm x 15 mm). Speziell die microSDHC-Karte bietet auf geringem Raum einen für Messsysteme bei weitem ausreichenden Speicher.

Bezüglich näherer Details hinsichtlich der Ansteuerung, etc. von SD(HC)-Karten wird auf die Literatur [5, 16] verwiesen.

3.2.5 Hoch- und Tiefpassfilter 1. Ordnung

Einfache Hoch- und Tiefpässe erster Ordnung können sowohl passiv als RC-Kombination oder aktiv mittels Operationsverstärkern gebildet werden (siehe Abbildung 3.16). Neben der im Passband wählbaren Verstärkung A_0 bieten die dargestellten aktiven Filter den Vorteil einer niedrigen Ausgangsimpedanz. Filter höherer Ordnung werden daher in der Regel mit Hilfe von Operationsverstärkern aufgebaut. Details hierzu sind z.B. in [40, 103] ausgeführt.

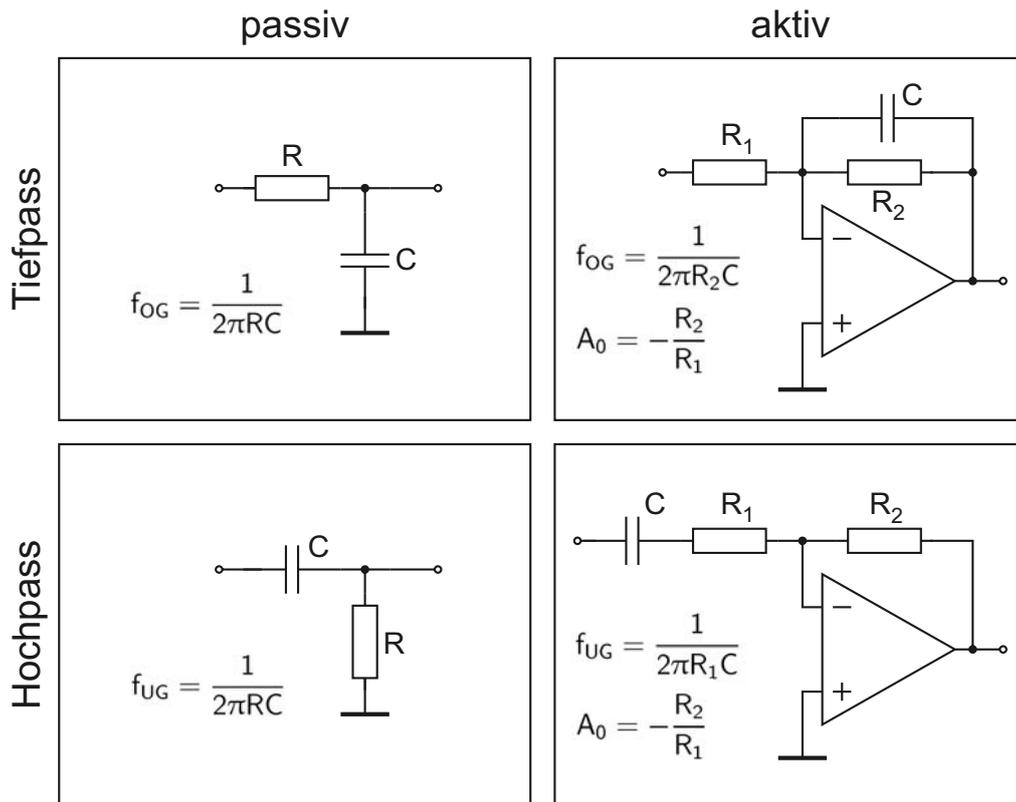


Abbildung 3.16: Passive und aktive Hoch- und Tiefpässe 1. Ordnung

Auf Grund ihrer Einfachheit werden passive RC-Filter trotz ihres flachen Übergangs (± 20 dB pro Dekade) vom Pass- in das Stopppband bzw. vice versa häufig eingesetzt. Für den Amplituden- bzw. Phasengang erhält man nach kurzer Rechnung

$$|\underline{A}| = \frac{1}{\sqrt{1 + \omega^2 R^2 C^2}} \quad (3.2)$$

bzw.

$$\varphi = -\arctan(\omega RC). \quad (3.3)$$

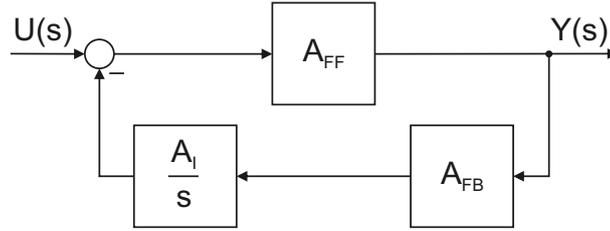


Abbildung 3.17: Hochpass mittels negativem Feedback

Eine weitere interessante Möglichkeit zur Verwirklichung eines Hochpasses erster Ordnung besteht durch die Verwendung eines Integrators im Rückkoppelungsweig eines Systems mit negativer Rückkoppelung entsprechend Abbildung 3.17. Für die Übertragungsfunktion des dargestellten Systems erhält man nach einfacher Rechnung

$$H(s) = \frac{A_{FF}}{1 + \frac{A_{FF}A_{FB}A_I}{s}}. \quad (3.4)$$

Durch die Substitution $s = j\omega$ erhält man aus (3.4) den Frequenzgang

$$H(j\omega) = \frac{A_{FF}}{1 + \frac{A_{FF}A_{FB}A_I}{j\omega}}. \quad (3.5)$$

Für Frequenzen weit oberhalb der unteren Grenzfrequenz ($f \gg f_{UG}$) erhält man durch den Grenzübergang $\omega \rightarrow \infty$ die Verstärkung des Systems im Passband

$$H_\infty = \lim_{\omega \rightarrow \infty} H(j\omega) = \lim_{\omega \rightarrow \infty} \frac{A_{FF}}{1 + \frac{A_{FF}A_{FB}A_I}{j\omega}} = A_{FF}. \quad (3.6)$$

Der Amplitudengang des Systems folgt durch Bildung des Betrags aus (3.4):

$$|H(j\omega)| = \frac{A_{FF}\omega}{\sqrt{A_{FF}^2 A_{FB}^2 A_I^2 + \omega^2}}. \quad (3.7)$$

Unter Berücksichtigung von (3.6) erhält man schließlich aus (3.7) mit $\omega = 2\pi f_{UG}$ die Bestimmungsgleichung für die -3 dB-Grenzfrequenz f_{UG}

$$|H(j2\pi f_{UG})| = \frac{A_{FF}}{\sqrt{2}}. \quad (3.8)$$

Für f_{UG} gilt daher

$$f_{UG} = \frac{A_{FF}A_{FB}A_I}{2\pi}. \quad (3.9)$$

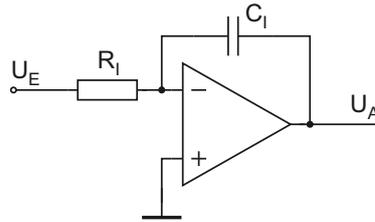


Abbildung 3.18: Invertierender Integrator

Mittels der in Abbildung 3.18 dargestellten Schaltung kann mit geringem Aufwand ein (invertierender) Integrator aufgebaut werden. Für die Ausgangsspannung der Schaltung gilt:

$$U_A = -\frac{1}{R_I C_I} \int_0^t U_E(\tau) d\tau + U_{A,0}. \quad (3.10)$$

Setzt man in weiterer Folge für die Verstärkung $A_I = (R_I C_I)^{-1}$ in (3.9) ein und wählt $A_{FB} < 0$ um negative Rückkoppelung zu erreichen, so erhält man mit $\hat{A}_{FB} = -A_{FB}$

$$f_{UG} = \frac{A_{FF} \hat{A}_{FB}}{2\pi R_I C_I}. \quad (3.11)$$

Analog zum passiven RC-Hochpass resultiert aus großen Zeitkonstanten $\tau = RC$ eine niedrige untere Grenzfrequenz. Zusätzlich kann entsprechend (3.11) durch Wahl von $\hat{A}_{FB} < 1$ die untere Grenzfrequenz des betrachteten Systems bei, im Vergleich zum RC-Hochpass, gleicher Verstärkung im Passbandbereich ($A_{FF} = 1$) und gleich gewähltem τ um den Faktor \hat{A}_{FB} reduziert werden. Die Bildung eines Hochpasses mittels Rückkoppelung kann daher, wie in Abschnitt 4.5 beschrieben, vorteilhaft für die Kompensation eines Offsetdrifts eingesetzt werden, wenn niedrige untere Grenzfrequenzen des Messkette erzielt werden müssen.

3.2.6 Piezoelektrische Beschleunigungssensoren

Zur Analyse von schwingungstechnischen Problemen werden häufig piezoelektrische Beschleunigungssensoren eingesetzt. Diese sollen daher auch als primärer Sensor bei dem zu entwickelnden Messsystem eingesetzt werden. Der prinzipielle Aufbau eines solchen Sensors ist in Abb. 3.19 dargestellt. Im Gehäuse ist eine seismische Masse mit Scheiben aus piezoelektrischem Material (z.B. Quarz oder Blei-Zirkonat-Titanat) unter Vorspannung montiert. Die Ladung am Ausgang des Sensors ist im Bereich weit unterhalb der Eigen-

frequenz des Sensors dabei proportional zur Beschleunigung des Befestigungspunktes des Sensors (hohe Abstimmung). Zur weiteren Verarbeitung und Aufzeichnung muss diese Ladung mittels eines Ladungsverstärkers in eine proportionale Spannung umgewandelt werden. Ein Abfluss von Ladungsträgern über Isolationsmaterialien oder die Einbringung von zusätzlichen Ladungen über Reibung (triboelektrischer Effekt – beim Biegen des Kabels) muss dabei soweit wie möglich verhindert werden. Aus diesem Grund müssen zwischen Sensor und Ladungsverstärker spezielle Kabel eingesetzt werden, wobei die Kabellänge möglichst kurz gehalten werden sollte.

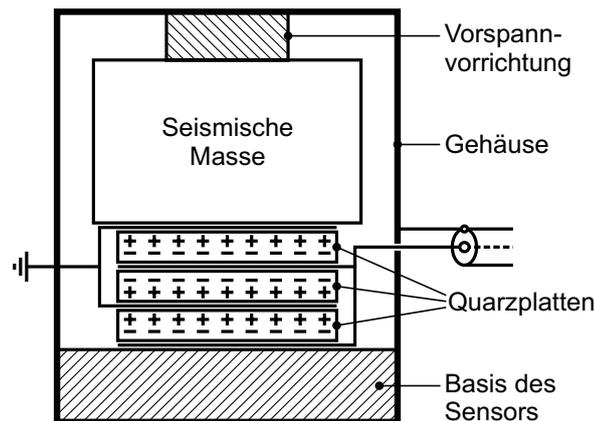


Abbildung 3.19: Schematischer Aufbau eines Piezo-Beschleunigungssensors (Logitudinaltyp) nach [85]

3.2.6.1 IEPE-Beschleunigungssensoren

Eine Sonderform von piezoelektrischen Beschleunigungssensoren stellen die IEPE-Sensoren dar. IEPE steht für *Integrated Electronics Piezo Electric*, also für piezoelektrische Sensoren mit integrierter Elektronik. Der Ladungsverstärker ist bei diesem Sensortyp bereits in das Gehäuse integriert, wodurch am Sensorausgang ein Spannungssignal mit niedriger Ausgangsimpedanz zur Verfügung steht. Hierdurch entfällt im Gegensatz zu konventionellen Piezo-Beschleunigungssensoren die Notwendigkeit zur Verwendung von teuren Spezialkabeln. Die Signalübertragung erfolgt mittels handelsüblicher Koaxialkabel, deren Länge einige 100 m betragen kann. Die Versorgung des Sensors erfolgt durch eine Konstantstromquelle (2 bis 20 mA), wobei die Energieversorgung und die Übertragung des Sensorausgangssignals über dieselbe Leitung erfolgen.

Abbildung 3.20 zeigt den prinzipiellen Aufbau einer IEPE-Messkette. Die Versorgungsspannung der Stromquelle U_S liegt im Normalfall zwischen 24 und 30 V. Bei einem empfoh-

lenen Versorgungsstrom von 4 mA, resultiert hieraus ein Energiebedarf von ca. 100 mW. Spezielle Low Power IEPE-Sensoren für batteriebetriebene Systeme können jedoch auch mit einer Stromquellenvorspannung von 9 bis 12 V und einem Strom ab 0.1 mA betrieben werden.

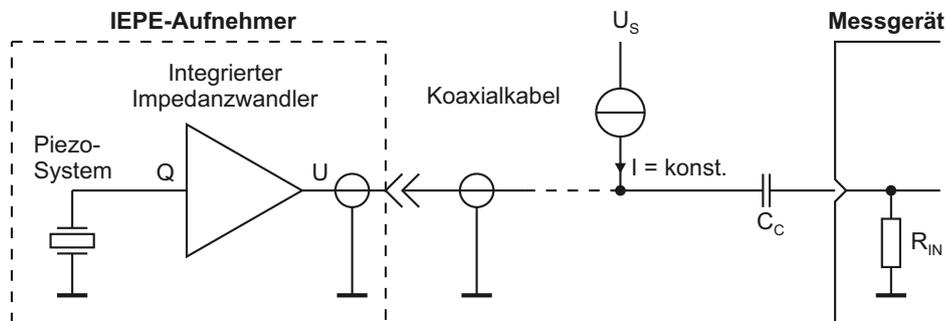


Abbildung 3.20: IEPE-Messkette nach [61]

Trotz ihrer praktischen Vorteile gibt es nach wie vor Anwendungen, die Piezo-Beschleunigungssensoren mit Ladungsausgang erfordern. Die wesentlichen Vorteile von Sensoren mit Ladungsausgang sind deren kleinere Bauform und Masse (geringere Rückwirkung auf das Messobjekt), der hohe Dynamikbereich, geringes Rauschen sowie höhere zulässige Umgebungstemperaturen [61].

3.3 Detailliertes Lösungskonzept

Entsprechend den in Abschnitt 3.2.1 dargestellten wesentlichen Eigenschaften der im 2.4 GHz-Band vorherrschenden Funkstandards wurde für das Messsystem ein Funktransceiver nach dem Standard IEEE 802.15.4 gewählt. Ausschlaggebend war hierfür im Wesentlichen die Möglichkeit der Peer-To-Peer-Kommunikation zwischen Geräten, wodurch Mesh-Netzwerke gebildet werden können, und der geringe Energiebedarf der Transceiver.

Der Einsatz von IEEE 802.15.4 erlaubt prinzipiell einerseits den Einsatz von ZigBee und anderen (proprietären) Netzwerkprotokollen und andererseits auch den Einsatz von TinyOS, das im Bereich der drahtlosen Sensornetzwerke häufig eingesetzt wird. Hinsichtlich der konkreten Implementierung der Netzwerkfunktionalität besteht daher ein hohes Mass an Flexibilität und es kann auf eine Vielzahl an Publikationen aus dem Bereich der drahtlosen Sensornetzwerke zurückgegriffen werden.

Die Entscheidung für IEEE 802.15.4 resultiert jedoch auch darin, dass nur geringe Datenmengen über das Funknetzwerk übertragen werden können. Wie in Abschnitt 2.3

dargelegt sind im Bereich der drahtlosen Sensornetzwerke unter Verwendung von IEEE 802.15.4 bei drei oder mehr Zwischenknoten lediglich Datenübertragungsraten bis zu etwa 30 kBit/s zu erreichen.

Bei einer geforderten Abtastrate von 1 kS/s (siehe Abschnitt 3.1.1) fallen bei der Übertragung von einem Messkanal 12 kBit an Messdaten pro Sekunde an. Dementsprechend könnte im Idealfall sowohl die Steuerung des Messsystems über das Funknetz erfolgen und zusätzlich ein Messkanal „live“ zur Steuereinheit übertragen werden.

3.3.1 Funktion des Messsystems

Entsprechend dem ersten Lösungsansatz (siehe Abschnitt 3.1.2), erfolgt der generelle Datenaustausch zwischen den Messknoten mittels eines Mesh-Netzwerks, wobei die Messdaten jedoch lokal am jeweiligen Knoten gespeichert werden. Konkret wird für die Kommunikation IEEE 802.15.4 in Kombination mit ZigBee zum Einsatz kommen und SD-Karten werden als Massenspeicher dienen.

Abbildung 3.21 zeigt beispielhaft ein solches Netzwerk bestehend aus verschiedenen Knotentypen. Entsprechend der ZigBee-Spezifikation wird für den Aufbau des Netzwerkes ein Koordinator benötigt, der in der Abbildung rot dargestellt ist. Die Funktion des Koordinators ist an keine zusätzlichen Hardwarevoraussetzungen gebunden, lediglich die Firmware des Knotens unterscheidet ihn von den übrigen Teilnehmern des Netzwerkes. Im Idealfall wird der Koordinator jedoch an einer Position eingesetzt, die die Versorgung durch das 24 V-Bordnetz ermöglicht.

Je nach Erfordernis kann es sich bei dem Koordinator um einen Knoten mit der vollen Funktionalität hinsichtlich der Erfassung von Sensorsignalen oder um einen reinen

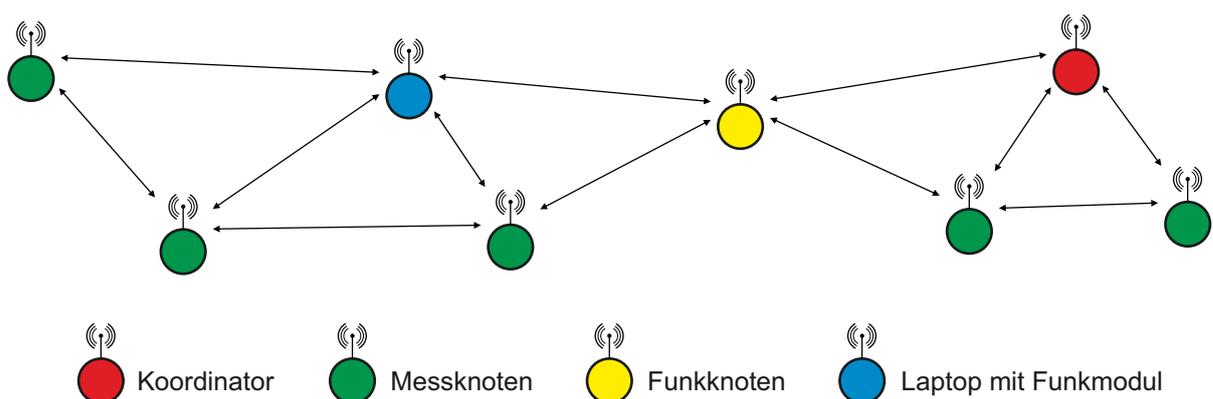


Abbildung 3.21: Knotentypen des Messsystems

Funkknoten handeln. Ein solcher Knoten ohne Messfunktion könnte entsprechend kompakt ausgeführt werden, da er lediglich aus einem Mikrocontroller mit Funk-Transceiver und Antenne sowie DC/DC-Konverter und Pufferspeicher für die Versorgung durch das Bordnetz bzw. autarker Energiequelle besteht. Ein in dieser Form ausgestatteter Knoten kann auch als Repeater eingesetzt werden, falls der Abstand zwischen zwei benachbarten Messknoten – deren Positionen im Wesentlichen durch die Wahl der Messstellen bestimmt wird – für eine direkte Kommunikation zwischen den Knoten zu groß ausfallen würde. In Abbildung 3.21 ist dieser Knotentyp gelb dargestellt.

Neben den normalen Sensorknoten (grün dargestellt) ist für die Bedienung des Systems ein Laptop mit Funkmodul vorgesehen. Dieser ist blau dargestellt und kann frei innerhalb der Strassenbahn, jedoch in Funkreichweite des Systems, positioniert werden. Die Lage dieses Knotens kann auch während des Einsatzes unter oben genannter Einschränkung frei variiert werden. Weiters muss er während der eigentlichen Aufzeichnung der Messdaten nicht zwingend vorhanden sein. Nach dem Start der Aufzeichnung kann der Laptop entfernt oder abgeschaltet werden und wird erst wieder am Ende der Messung für das Stoppen der Aufzeichnung benötigt.

Neben dem Starten und Stoppen der Aufzeichnung können über die Bedienoberfläche am Laptop die Einstellungen der Messkanäle geändert werden. Dies umfasst beispielsweise die Verstärkung, die Grenzfrequenz der Antialiasing-Filter und die Abtastrate. Darüber hinaus besteht die Möglichkeit einen Messkanal des Systems per Funk zum Laptop zu übertragen und diesen live am Bildschirm darzustellen. Dies erleichtert einerseits das Setup des Systems und ermöglicht andererseits die Betrachtung einer kritischen Messgröße während der Fahrt. Zusätzliche Funktionalität wie z.B. die Möglichkeit Triggerpegel für einzelne Kanäle zu setzen, bei deren Überschreitung dem Bediener entsprechende Rückmeldungen gegeben werden, sind natürlich ebenfalls denkbar.

Nach dem Abschluss der Aufzeichnung können die SD-Karten entnommen und die Messdaten am Laptop/PC eingelesen und analysiert oder beispielsweise als CSV-Datei exportiert werden.

Während des Betriebs muss auf Grund der langen Aufzeichnungsdauer mit einem Auseinanderdriften der lokalen Zeitbasen der einzelnen Module gerechnet werden. Je nachdem welche Anforderungen an die zeitliche Korrelation zwischen den Abtastwerten der einzelnen Messkanäle besteht, muss nach einem synchronen Start der Zeitbasen entweder die Aufzeichnungsdauer entsprechend beschränkt werden, oder eine fortlaufende Synchronisierung der Zeitbasen per Funk vorgesehen werden. Letztere Variante ist auf Grund der hieraus resultierenden höheren Praktikabilität zu präferieren.

3.3.2 Aufbau eines Messknotens

Der prinzipielle, konzeptionelle Aufbau eines Messknotens ist in Abb. 3.22 dargestellt. Auf Grund der hohen Kanalanzahl und der zu erwartenden Komplexität eines Knotens ist die Integration der gesamten Elektronik auf einer einzelnen Platine nicht praktikabel. Es bietet sich daher eine schaltungstechnische Aufteilung entsprechend der Funktion an, wobei die Grundfunktionalität (Netzwerkfunktionalität, Datenspeicher und Steuerlogik inkl. Mikrocontroller) auf einer Basisplatine vorgesehen ist. Weitere Subbaugruppen werden für die Energieversorgung des Systems, die analoge Signalaufbereitung (allgemeine Signalaufbereitung) sowie die Anbindung der Sensoren benötigt.

Die Energieversorgung leitet hierbei in der Regel vom 24 V-Bordnetz die einzelnen Versorgungsspannungen ab und stellt darüber hinaus mittel Pufferakkumulatoren sicher, dass kurzzeitige Einbrüche des Bordnetzes nicht zu Datenverlusten führen. Neben der Versorgung durch das Bordnetz ist auch die Versorgung mit Hilfe eines ausreichend dimensionierten externen Akkumulatorblocks möglich.

Die allgemeine Anpassung von Spannungssignalen an den Eingangsspannungsbereich des ADCs erfolgt durch die allgemeine Signalaufbereitung. Ziel der Entwicklung ist hier eine wählbare Verstärkung sowie eine anpassbare Tiefpassfilterung zur Vermeidung von Aliasing zu integrieren. Weiters sollten sowohl uni- als auch bipolare Signale verarbeitet werden können.

Sensoren benötigen in vielen Fällen spezielle Hilfsschaltungen wie z.B. Brückenschaltungen (Dehnmessstreifen) oder Ladungsverstärker (Piezo-Beschleunigungssensoren) um die Änderungen einer physikalischen Größe in weiterer Folge in eine proportionale Spannung umsetzen zu können. Diese Aufgabe soll im Rahmen des Messsystems für jeden einzelnen Kanal über wechselbare Sensorinterfaces erfolgen. Spezielle Hilfsspannungen, Konstantstromquellen, etc. sind ebenfalls im Sensorinterface zu integrieren.

Durch die Verwendung von Steckverbindern können die Subplatinen entsprechend der Reihenfolge in Abbildung 3.22 miteinander verbunden werden, wodurch ein kompakter Stapel entsteht, der leicht in kommerziell erhältlichen Gehäusen untergebracht werden kann. Entsprechende Steckverbinder (für Sensoren, Spannungsversorgung, Antenne, etc.) sowie LEDs für Zustandsanzeigen und eine Entnahmemöglichkeit für den Datenspeicher müssen natürlich an passender Stelle am Gehäuse vorgesehen werden.

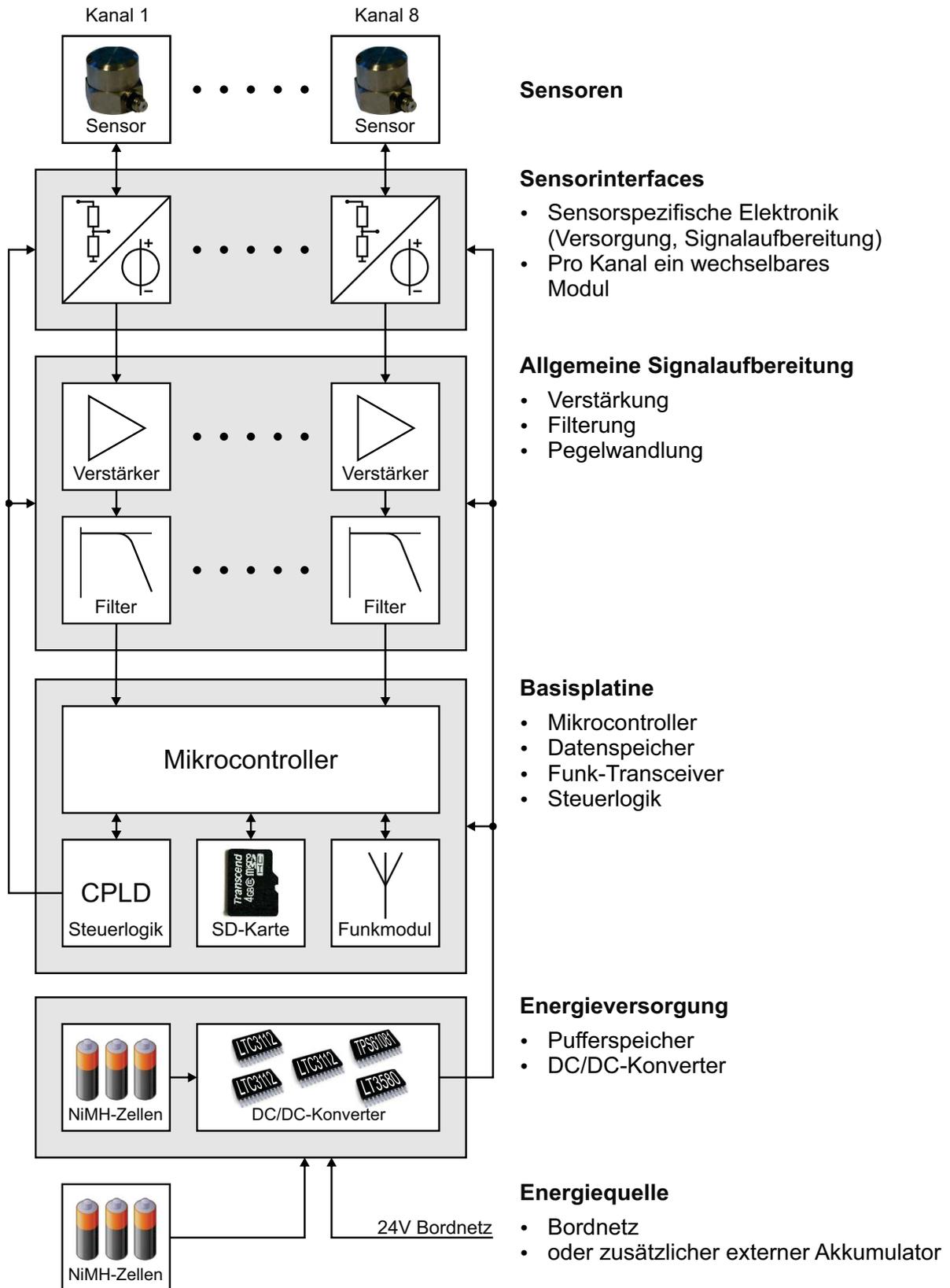


Abbildung 3.22: Schematischer Aufbau eines Messknotens

4 Das Laborsystem

Das Ziel bei der Entwicklung des Laborsystems bestand darin, ein in der Laborumgebung einfach zu handhabendes und leicht erweiterbares System zu schaffen, das hinsichtlich der Funktionalität bereits in weiten Teilen den Erfordernissen des künftigen Messsystems entspricht. Hieraus lassen sich im Wesentlichen folgende Anforderungen ableiten:

- Modularer Aufbau,
- kompakte Messknoten,
- Energieversorgung mittels AA-Standardbatterien oder Akkumulatoren und
- Beschränkung auf einen Sensorkanal pro Messknoten.

Ein modularer Aufbau ermöglicht neben einer flexiblen Anpassung einzelner Knoten an die konkreten Anforderungen hinsichtlich deren Funktionalität zusätzlich ebenfalls eine einfache Entwicklung des Systems, da sich Änderungen an einzelnen Funktionsmodulen in der Regel nur auf die betroffenen Module beschränken. Kompakte Abmessungen und eine lokale Energieversorgung mittels wechselbaren Batterien oder Akkumulatoren in der Standardbauform AA erleichtern sowohl den Einsatz im Labor als auch bei Tests in der späteren Einsatzumgebung (Straßenbahn), da keine Anbindung an eine externe Energiequelle (Labornetzteil, Bordnetz) notwendig ist und durch den Tausch von leeren Batterien oder Akkumulatoren nur eine kurzzeitige Betriebsunterbrechung entsteht. Die Verfügbarkeit von Akkumulatoren und entsprechenden Ladegeräten ermöglicht weiters den kostengünstigen und umweltschonenden Einsatz von NiMH-Zellen als Energiequelle.

Aus der kompakten Baugröße folgt schließlich ebenfalls die Beschränkung auf einen einzelnen Sensorkanal. Neben dem Platzbedarf der pro Sensorkanal benötigten Elektronik ist hierfür vor allem der Energiebedarf der Signalaufbereitung und des Sensorinterfaces ausschlaggebend. Speziell mehrere IEPE-Sensoren (siehe Abschnitt 3.2.6.1 und 4.5) würden hier entweder zu einer drastischen Reduktion der Betriebsdauer mit einem Satz Batterien führen oder eine entsprechende Kompensation durch Erhöhung der Kapazität (Anzahl der Zellen) erzwingen.

Die Entwicklung des Laborsystems erfolgte in einzelnen, nach Funktion gegliederten, Funktionsprototypen. Diese umfassen im Wesentlichen die Funktionen:

- Funk,
- Datenspeicher,
- Signalaufbereitung (Verstärker, Pegelwandler und Antialiasing-Filter),
- Sensorinterface (Energieversorgung des Sensors, Brückenschaltung, etc.),
- Energieversorgung des Systems,
- USB-Interface (Steuerung des Systems mittels Labtop) und
- Funktionsgenerator (für Testzwecke).

Mit Ausnahme des Basismoduls, das das Funkinterface und die Energieversorgung beinhaltet, ist für jede dieser Funktionen ein eigenes Modul entwickelt worden. Die Module wurden in Form von steckbaren Platinen ausgeführt, die in Abbildung 4.1 dargestellt sind. Je nach Notwendigkeit können diese mittels Steckverbindern zu Knoten mit unterschiedlicher Funktionalität assembliert werden (siehe Abbildung 4.2). Die Funktionalität

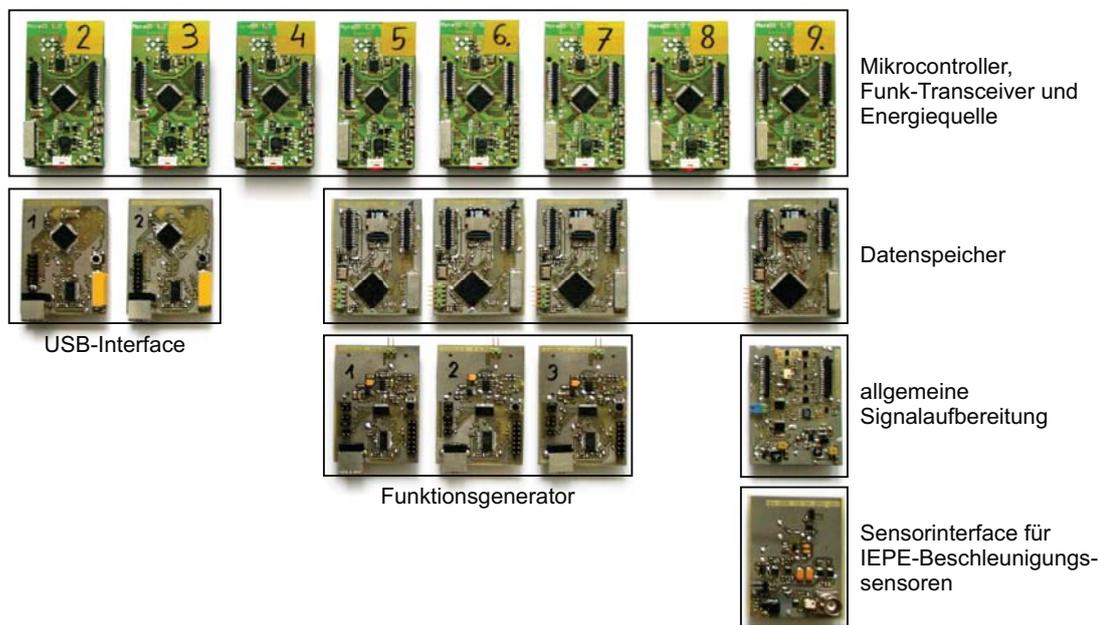


Abbildung 4.1: Module des Laborsystems



Abbildung 4.2: Assemblierte Knoten des Laborsystems

des Gesamtsystems sowie der einzelnen Module wird in den folgenden Abschnitten noch näher erläutert.

Die Steuerung des Systems erfolgt durch einen, per USB angebundenes, PC oder Laptop. Neben der eigentlichen Hard- und Firmware des Systems wird daher auf PC-Seite eine grafische Bedienoberfläche benötigt, die das Absetzen von Befehlen sowie die Darstellung von Messdaten und Rückmeldungen des Systems in einer übersichtlichen und komfortablen Art und Weise erlaubt. Auf Grund seiner Ausrichtung hinsichtlich der Mess-, Steuer- und Automatisierungstechnik und der daraus resultierenden Möglichkeit der einfachen Erstellung von GUIs (Graphical User Interfaces) wurde für die Implementierung dieser Schnittstelle LabVIEW gewählt. Die große Anzahl an bereits in LabVIEW integrierten Funktionsblöcken zur Datenanalyse (FFT, etc.) und die langjährige positive Entwicklungserfahrung gaben den Ausschlag für den Einsatz von LabVIEW.

4.1 Überblick über die Architektur und Funktion des Laborsystems

Die Grundlage des Laborsystems bildet das Basismodul (siehe Abbildung 4.3), das die grundlegende Netzwerkfunktionalität bereitstellt. Es beinhaltet die Komponenten

- Mikrocontroller (Texas Instruments MSP430F2618),
- Funk-Transceiver (Texas Instruments CC2520),
- PCB-Antenne,
- Steckverbinder für Erweiterungsmodule,
- Batteriehalter für zwei Standardzellen des Typs AA und
- DC/DC-Konverter zur Erzeugung der +3.3 V-Versorgungsspannung des Systems.

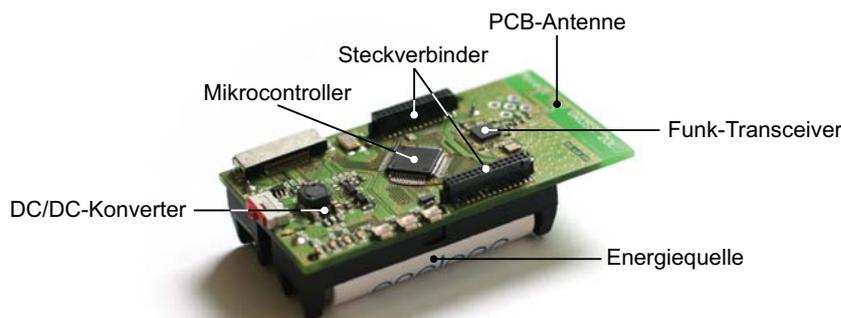


Abbildung 4.3: Basismodul des Laborsystems

Der Basisknoten entspricht im Aufbau und der Funktionalität konventionellen Knoten eines drahtlosen Sensornetzwerks, enthält jedoch im Gegensatz zu den im Jahr 2009 (Durchführung dieser Entwicklung) bekannten Knoten sowohl den IEEE 802.15.4 kompatiblen Funk-Transceiver CC2520 als auch den MSP430F2618 von Texas Instruments (TI). Im Vergleich zum CC2420 (ebenfalls IEEE 802.15.4 kompatibel), der beispielsweise in den Sensornetzwerkknoten MicaZ, Telos oder IMote2 eingesetzt wird, bietet der CC2520 ein um 8 dB höheres Link Budget. Hinsichtlich des geplanten Einsatzes in der Straßenbahn ist auf Grund der potenziell vorhandenen elektromagnetischen Störsender speziell die im Vergleich zum CC2420 um 5 dBm höhere Sendeleistung vorteilhaft.

Im WSN-Bereich werden in der Regel Low-Power Mikrocontroller von Atmel (z.B. MicaZ, Mica2 und Iris) oder Texas Instruments (z.B. Telos und TinyNode) eingesetzt. Die MSP430-Reihe von TI bietet gegenüber dem häufig verwendeten ATmega132L den Vorteil, dass der integrierte ADC (Analog-to-Digital-Converter) eine Auflösung von 12 Bit aufweist, dies ermöglicht im Vergleich zum ATmega132L eine um den Faktor 16 feinere Diskretisierung des Messsignals. Darüber hinaus handelt es sich bei der MSP430-Architektur um 16 Bit-Mikrocontroller (ATmega: 8 Bit) wodurch die Verarbeitung von 16 Bit-Variablen – und damit auch der 12 Bit breiten Abtastwerte – effizienter erfolgt.

Hinsichtlich der Programmierung bietet der Einsatz von Mikrocontroller und Funk-Transceiver vom selben Hersteller den weiteren Vorteil, dass Codebeispiele oder Funkprotokolle in Form von Software-Stacks existieren, die mit geringen Anpassungen direkt für den Mikrocontroller übernommen werden können. Die Wahl fiel aus den genannten Gründen daher auf die Kombination von MSP430F2618 und CC2520 von TI.

Hinsichtlich des Netzwerkprotokolls besteht die Möglichkeit sowohl ZigBee als auch TinyOS auf der Hardwareplattform einzusetzen. Zum Zeitpunkt der Entwicklung des

Systems war jedoch kein Treiber für den CC2520 unter TinyOS verfügbar. Auf Grund der Komplexität und des, mit einer eigenen Treiberentwicklung verbundenen, schwer abschätzbaren Zeitaufwandes, wurde von einer Eigenentwicklung Abstand genommen und stattdessen auf die ZigBee-Implementierung von TI, den Z-Stack, zurückgegriffen. Nach Abschluss der Entwicklung im Rahmen dieser Arbeit (Ende 2011), war ein solcher Treiber im TinyOS Projektarchiv [104] unter `svn/trunk/tos/chips/cc2520/` verfügbar. Der Einsatz von TinyOS auf der entwickelten Hardwareplattform ist daher eine gute Option für künftige Verbesserungen des Systems und ist speziell im Hinblick auf die unter TinyOS verfügbaren Synchronisationsprotokolle (siehe Abschnitt 2.3) und die in Abschnitt 4.3.1.3 beschriebene Problematik der synchronen Zeitbasen empfehlenswert.

Folgende Erweiterungsmodule stehen derzeit für das Laborsystem zur Verfügung:

Datenspeicher: Der Datenspeicher übernimmt die analogen Messsignale, führt die A/D-Wandlung durch und enthält den lokalen Massenspeicher. Das Modul verfügt über einen Mikrocontroller (MSP430F5438) mit integriertem ADC, einen temperaturkompensierten Quarz (Zeitbasis des Moduls) und einen Sockel für microSDHC-Karten.

Allgemeine Signalaufbereitung: Die allgemeine Signalaufbereitung dient der Anpassung des Sensorsignals an den Spannungsbereich des im Datenspeicher vorhandenen ADCs (Verstärkung und Pegelwandlung) sowie der Vermeidung von Aliasing bei der Abtastung durch eine entsprechende Tiefpassfilterung der analogen Messsignale.

IEPE-Sensorinterface: Das IEPE-Interface erlaubt die Anbindung und Versorgung von IEPE-Beschleunigungssensoren.

USB-Interface: Das USB-Interface ermöglicht den Informationsaustausch zwischen PC und Laborsystem, wodurch die Steuerung des Laborsystems mit Hilfe eines GUIs über handelsübliche PCs oder Laptops möglich ist.

Funktionsgenerator: Der Funktionsgenerator dient als portable Signalquelle für Systemtests.

Sowohl Basismodul als auch Datenspeicher sind mit Mikrocontrollern ausgestattet, die mittels RS232-Schnittstelle miteinander kommunizieren. Die Aufgabenbereiche der beiden Mikrocontroller sind strikt in Netzwerkmanagement und Kommunikation (Basismodul) sowie Datenerfassung und Aufzeichnung (Datenspeicher) getrennt. Diese Trennung

der Funktionalität war auf Grund der beschränkten Rechenleistung der Mikrocontroller, der hohen Komplexität des Funkprotokolls und den relativ großen zu verarbeitenden Datenmengen (etwa 16 kB/s) erforderlich. Abbildung 4.4 zeigt beispielhaft den Kommunikationsfluss innerhalb des Laborsystems zwischen einem Laptop mit grafischer Benutzeroberfläche (Steuerung des Systems) und einem Knoten mit angeschlossenem Sensor.

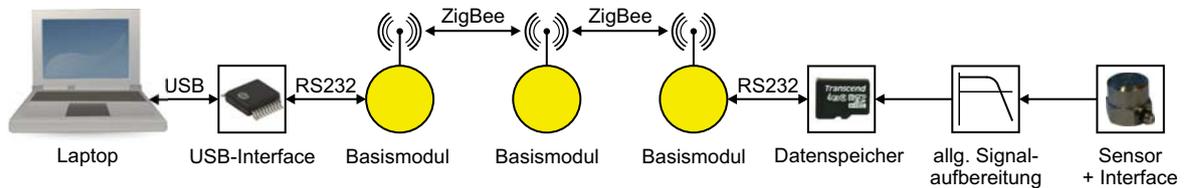


Abbildung 4.4: Kommunikationsfluss des Laborsystems

Die Messkette des Laborsystems, bestehend aus Datenspeicher, allgemeiner Signalaufbereitung und einem generischen Sensorinterface, ist in Abbildung 4.5 in Form eines vereinfachten Blockschaltbildes dargestellt. Die Steuerung der Messkette (Verstärkung, Filtereckfrequenz, etc.) erfolgt durch den Mikrocontroller des Datenspeichers.

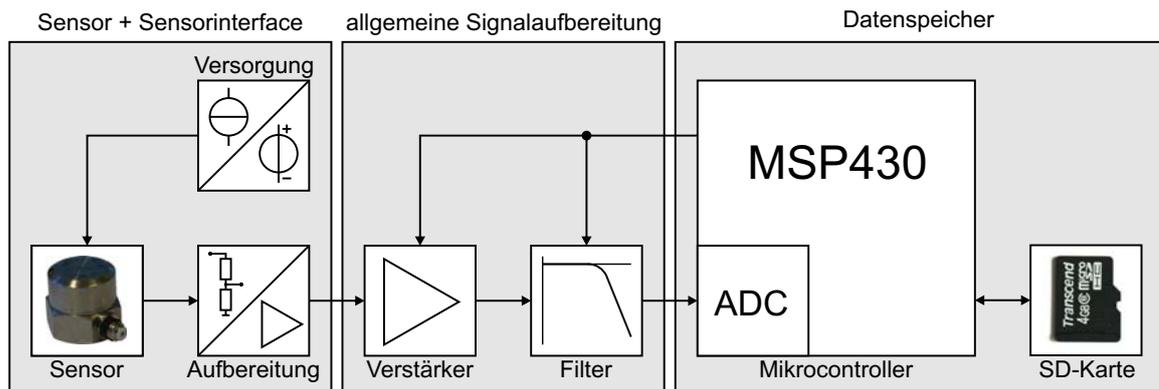


Abbildung 4.5: Messkette des Laborsystems

Auf Grund der Funktionalität des Basismoduls und der Erweiterungsmodule sind folgende Kombinationen sinnvoll:

Basismodul ohne Erweiterungsmodule: Da der Basisknoten die gesamte Netzwerkfunktionalität beinhaltet, kann dieser als Zwischenknoten zur Erhöhung der Netzwerkreichweite bzw. im Falle von ZigBee auch als Netzwerkkoordinator eingesetzt werden.

Basismodul mit USB-Interface: In Kombination mit dem USB-Interface kann das Basismodul als Gateway zwischen Laptop und Laborsystem zur Steuerung des Systems eingesetzt werden.

Basismodul mit Datenspeicher, allg. Signalaufbereitung und IEPE-Interface: Die Kombination dieser Module stellt einen funktionsfähigen Messkanal zur Beschleunigungsmessung mittels IEPE-Sensoren inklusive lokalem Datenspeicher zur Verfügung.

Basismodul mit Datenspeicher und Funktionsgenerator Mit Hilfe des Funktionsgenerators können definierte Testsignale in das Datenspeichermodul eingespeist werden. Die Kombination aus Basismodul, Datenspeicher und Funktionsgenerator emuliert einen Messknoten mit acht Sensorkanälen und lokalem Datenspeicher für Systemtests.

Bei der Assemblierung der Netzwerkknoten ist auf die Reihenfolge der Module zu achten, da diese nicht in beliebiger Reihenfolge aufeinander gesteckt werden dürfen. Sowohl der Datenspeicher als auch das USB-Interface können direkt mit dem Basismodul verbunden werden. Auf den Datenspeicher kann entweder die allgemeine Signalaufbereitung oder der Funktionsgenerator (von oben) angekoppelt werden. Etwaige Sensorinterfaces können ausschließlich auf die allgemeine Signalaufbereitung aufgesteckt werden.

Kurz zusammengefasst sind die wesentlichen technischen Merkmale des Laborsystems:

- Funknetz in Mesh-Topologie (ZigBee),
- IEEE 802.15.4 konforme Funktransceiver (2.4 GHz-ISM-Band),
- bis zu 32 GB lokaler Datenspeicher je nach verwendeter Speicherkarte (microSDHC-Standard),
- bis zu acht ADC-Kanäle mit einer Abtastrate von je 1 kS/s,
- ein analoger Eingangskanal mit über Funk konfigurierbarer allgemeiner Signalaufbereitung (Verstärkung, Offset und obere Grenzfrequenz des Antialiasing-Filters) und
- Interface für IEPE-Beschleunigungssensoren.

Eine detaillierte Beschreibung der Hard- und Software sowie die Ergebnisse eines in einer Straßenbahn am Bombardier-Betriebsgelände durchgeführten Tests des Laborsystems erfolgt im weiteren Verlauf dieses Kapitels.

4.2 Basismodul

Der grundlegende Aufbau des Basismoduls wurde bereits in Abschnitt 4.1 beschrieben (siehe auch Abbildung 4.3). Im Folgenden werden sowohl die wesentlichen Hardwarefunktionsblöcke (Funk-Transceiver und Energiequelle) sowie deren Entwicklungsstufen als auch die implementierte Firmware beschrieben. Die Pinbelegung des Basismoduls ist in Anhang D.1 dokumentiert.

4.2.1 Beschreibung der Hardware

Neben den bereits erwähnten Komponenten Mikrocontroller, Funk-Transceiver, Steckverbinder und Energieversorgung werden für den Betrieb des Basisknotens Bedien- und Anzeigeelemente, ein Uhrenquarz sowie eine Programmier- bzw. Debuggschnittstelle benötigt. Das Basismodul beinhaltet daher unter anderem die Komponenten

- Mikrocontroller MSP430F2618,
- Funk-Transceiver CC2520 inklusive PCB-Antenne und 32 MHz-Quarz (siehe Abschnitt 4.2.1.1),
- Bedienelemente (3 Taster),
- Anzeigeelemente (4 LEDs),
- 32.768 kHz-Uhrenquarz,
- Steckverbinder für Erweiterungsmodule,
- JTAG-Anschluss (Programmierung und Debuggen des MSP430) und
- Energieversorgung bestehend aus Batteriehalter und DC/DC-Konverter LTC2526L (siehe Abschnitt 4.2.1.2).

Abbildung 4.6 zeigt das Blockschaltbild des Basismoduls.

4.2.1.1 Funk-Transceiver

Ausgangspunkt für die Entwicklung des HF-Teils des Basismoduls war das CC2520EM Reference Design von TI [91], das in Abbildung 4.7 dargestellt ist. Abbildung 4.8 zeigt das, für die Antennenanbindung benötigte, Anpassungsnetzwerk des CC2520. Die Induktivitäten L_1 , L_2 und L_3 sind, wie in Abbildung 4.7 gezeigt, als Microstrip-Leiterbahnen

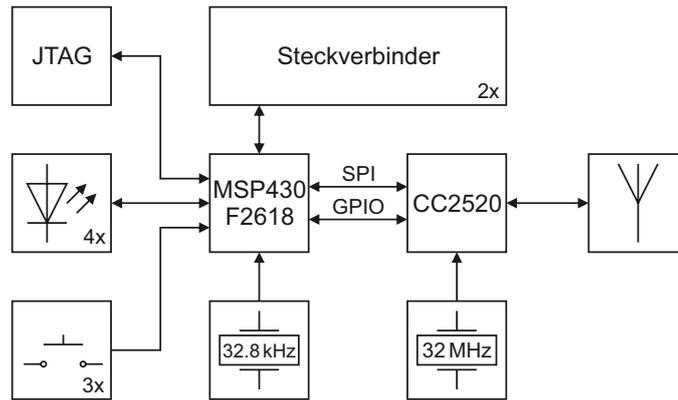


Abbildung 4.6: Blockschaltbild des Basismoduls

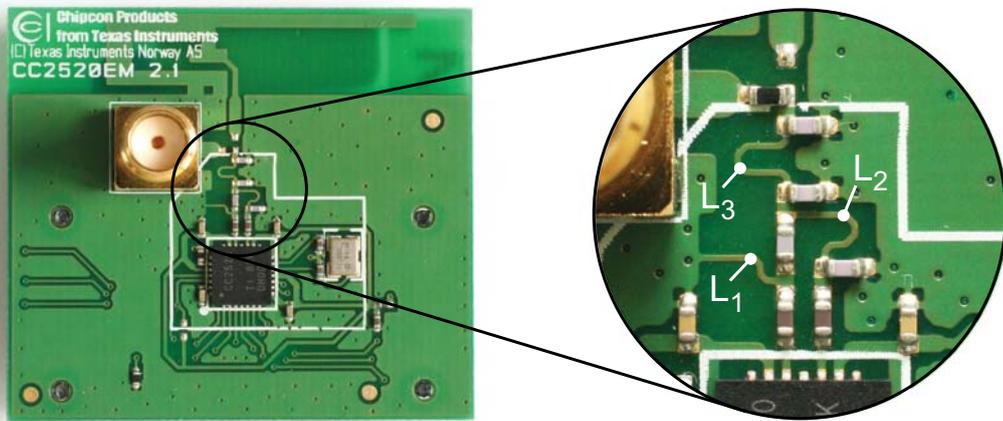


Abbildung 4.7: Reference Design CC2520EM

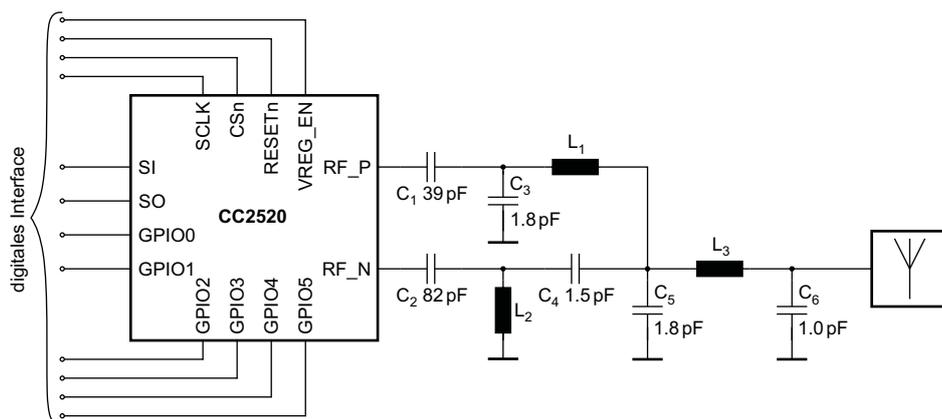


Abbildung 4.8: Anpassungsnetzwerk und digitales Interface des CC2520

ausgeführt. Konkrete Werte der Induktivitäten L_1 , L_2 und L_3 sind weder im Datenblatt des CC2520 noch in der Dokumentation des Reference Designs Werte angeführt. Das CC2520EM ist als doppelseitige Platine mit einer Dicke von 0.8 mm ausgeführt, woraus eine geringe mechanische Stabilität der Platine resultiert. Auf Grund der, während der Handhabung des Basismoduls, zu erwartenden mechanischen Belastungen – z.B. beim Aufstecken von Erweiterungsmodulen – wurde von einer Ausführung des Basismoduls in 0.8 mm FR4 Abstand genommen. Da die Induktivität von Microstrip-Leitungen unter anderem auch von deren Abstand zur Massefläche abhängt, konnte das Layout des CC2520EM nur mit einer entsprechenden Anpassung der Leiterbahngeometrie der Induktivitäten an die gewählte Leiterplattendicke von 1.5 mm übernommen werden.

4.2.1.1.1 Anpassung der Induktivitäten L_1 , L_2 und L_3

Gleichungen für die näherungsweise Berechnung der Induktivität von Microstrip-Leitungen sind beispielsweise in [36] angeführt. Der dort beschriebene Rechengang wird im Folgenden vorgestellt, wobei eine Anpassung der Gleichungen an das metrische System durchgeführt wurde. Die geometrischen Parameter h , t , w und l , entsprechend Abbildung 4.9, sind in Millimeter einzusetzen.

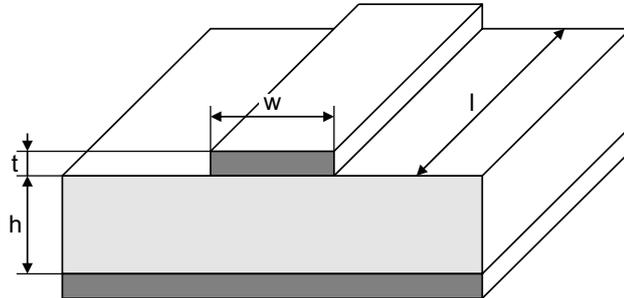


Abbildung 4.9: Microstrip-Geometrie

Die effektive relative Permittivität und effektive Leiterbahnbreite (in mm) für Microstrip-Leitungen erhält man mittels

$$\varepsilon_{r,eff} = \begin{cases} \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2} \left[\left(1 + \frac{12h}{w}\right)^{-\frac{1}{2}} + 0.04 \left(1 - \frac{w}{h}\right)^2 \right] - \frac{(\varepsilon_r - 1) \frac{t}{h}}{4.6 \sqrt{\frac{w}{h}}} & \text{für } w < h \\ \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2} \left(1 + \frac{12h}{w}\right)^{-\frac{1}{2}} - \frac{(\varepsilon_r - 1) \frac{t}{h}}{4.6 \sqrt{\frac{w}{h}}} & \text{sonst} \end{cases} \quad (4.1)$$

bzw.

$$w_{eff} = \begin{cases} w + \frac{1.25t}{\pi} \left[1 + \ln \left(\frac{4\pi w}{t} \right) \right] & \text{für } 2\pi w < h \\ w + \frac{1.25t}{\pi} \left[1 + \ln \left(\frac{2h}{t} \right) \right] & \text{sonst.} \end{cases} \quad (4.2)$$

Mit den Werten aus (4.1) und (4.2) kann in weiterer Folge die charakteristische Impedanz in Ohm

$$Z_{MS} = \begin{cases} \frac{60}{\sqrt{\varepsilon_{r,eff}}} \ln \left[\frac{8h}{w_{eff}} + \frac{w_{eff}}{4h} \right] & \text{für } w < h \\ \frac{120\pi}{\sqrt{\varepsilon_{r,eff}} \left[\frac{w_{eff}}{h} + 1.393 + 0.667 \ln \left(\frac{w_{eff}}{h} + 1.444 \right) \right]} & \text{sonst} \end{cases} \quad (4.3)$$

und die Laufzeitverzögerung in Sekunden pro Millimeter

$$P_{MS} = 3.335 \cdot 10^{-12} \sqrt{\varepsilon_{r,eff}} \quad (4.4)$$

berechnet werden. Die Induktivität der Leiterbahn in Henry erhält man schließlich aus

$$L_{MS} = P_{MS} Z_{MS} l, \quad (4.5)$$

wobei, wie aus (4.4) und (4.3) ersichtlich, die Induktivität nicht von der relativen Permittivität des Platinenmaterials abhängt.

Die Induktivitäten L_1 , L_2 und L_3 des Reference-Design sind mit einer Breite von 8 mil (1 mil entspricht $1/1000$ Zoll) ausgeführt. Mit $h = 0.8$ mm, $w = 0.203$ mm und $t = 35$ μ m erhält man mit Hilfe von (4.1) bis (4.5) eine Induktivität pro Längeneinheit von 0.634 nH/mm. In Tabelle 4.1 sind entsprechende Werte für 1.5 mm und 1.6 mm dickes Basismaterial und Leiterbahnbreiten von 0.203 mm (8 mil), 0.255 mm (10 mil) und 0.305 mm (12 mil) wiedergegeben. Zusätzlich ist der jeweilige Wert der Induktivität pro Längeneinheit auf den Wert des Referenzdesigns bezogen worden. Da mit ansteigender Platinendicke h die Induktivität steigt und durch eine alleinige Erhöhung der Leiterbahnbreite w nicht sinnvoll ausgeglichen werden kann, muss zusätzlich die Länge der Leiterbahn um den Faktor L_{ref}/L korrigiert werden.

Die Leiterbahnlängen der Induktivitäten des Referenzdesigns wurden nach einem Import der CAD-Dateien in Altium Designer bestimmt. Tabelle 4.2 enthält sowohl die Werte des Referenzdesigns als auch die korrigierten Längen für Leiterplatten mit einer Dicke von 1.5 mm und 1.6 mm. Die Leiterbahnbreite wurde für diese Platinendicken im Vergleich zum

h in mm	w in mm	t in μm	L/s in nH mm^{-1}	L/L_{ref}
0.8	0.203	35	0.634	1.000
1.5	0.203	35	0.754	1.189
	0.254	35	0.719	1.134
	0.305	35	0.691	1.090
1.6	0.203	35	0.767	1.210
	0.254	35	0.732	1.155
	0.305	35	0.703	1.110

Tabelle 4.1: Induktivität pro Millimeter für verschiedene Microstrip-Geometrien

Induktivität	l_{ref} in mm	$l_{1.5}$ in mm	$l_{1.6}$ in mm
L_1	3.396	3.116	3.063
L_2	2.687	2.466	2.424
L_3	3.780	3.261	3.409

Tabelle 4.2: Leiterbahnlänge der Induktivitäten L_1 , L_2 und L_3 bei unterschiedlichen Basismaterialdicken (l_{ref} Referenzdesign bei $w = 0.203$ mm und $h = 0.8$ mm, $l_{1.5}$ bei $w = 0.305$ mm und $h = 1.5$ mm, $l_{1.6}$ bei $w = 0.305$ mm, $h = 1.6$ mm)

Referenzdesign (0.203 mm) auf 0.305 mm erhöht. Etwaige Ursachen für Performanceunterschiede lassen sich hierdurch leicht zuordnen. Differenzen zwischen CC2520EM und der Ausführung in 0.8 mm FR4 würden auf, im Vergleich zu den Komponenten des Referenzdesign, abweichende Eigenschaften einer oder mehrerer der eingesetzten Kondensatoren (C_1 bis C_6 oder im Zuge des Fertigungsprozesses entstandene Geometrieabweichungen der Induktivitäten (L_1 bis L_3) hinweisen. Ein deutlicher Unterschied zwischen der Ausführung auf 0.8 mm und 1.5 mm wäre ein Indikator für Abweichungen der tatsächlichen Induktivitäten von den durch (4.1) bis (4.5) prognostizierten Werten, verursacht durch Variationen im Ätzprozess (Abweichung der tatsächlichen Geometrie von der Angestrebten) oder unzureichende Genauigkeit der Näherungslösung.

Die Prototypen des HF-Teils des Basismoduls sind in Abbildung 4.10 abgebildet. Die Entwicklung erfolgte in zwei Schritten, wobei im ersten Schritt das Originallayout des Referenzdesign auf 0.8 mm FR4-Basismaterial übernommen wurde und im zweiten Schritt der Übergang auf eine Platinendicke von 1.5 mm mit Anpassung der Leiterbahngeometrie von L_1 , L_2 und L_3 entsprechend Tabelle 4.2 erfolgte. Der Test auf Funktionstüchtigkeit erfolgte mittels des CC2520 Development Kits, das eine Applikation zur Bestimmung der Paketfehlerrate (PER – Packet Error Ratio) enthält.

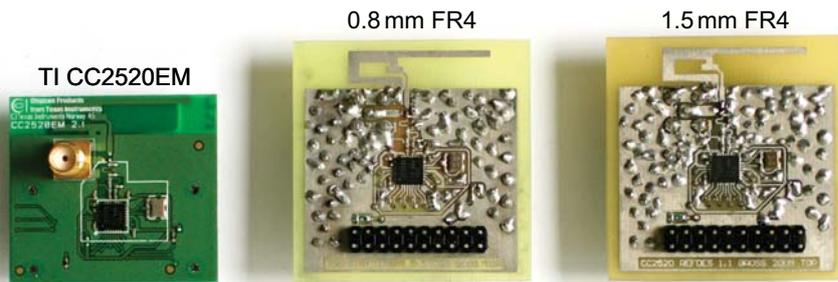


Abbildung 4.10: Reference Design CC2520EM und Prototypen des HF-Teils

4.2.1.1.2 CC2520DK Entwicklungskit

Das Development Kit (siehe Abbildung 4.11) besteht aus folgenden Baugruppen:

- CC2520EM (Referenzdesign des CC2520) inklusive Antenne (Antenova Titanis),
- CCMSP-EM430F2618 (Platine mit MSP430F2618 Mikrocontroller) und
- SmartRF05EB (Platine mit USB-Interface, LCD-Display, Bedienelementen und Batteriehaltern).



Abbildung 4.11: CC2520 Development Kit (links mit aufgestecktem Prototyp, rechts mit CC2520EM)

Das CC2520EM kann einerseits mit dem SmartRF05EB kombiniert werden, wobei der CC2520 dann per USB-Interface über den PC angesteuert wird, oder andererseits mit dem CCMSP-EM430F2618 und dem SmartRF05EB eingesetzt werden. In der letztgenannten Variante ist der MSP430F2618 mit dem digitalen Interface des CC2520 verbunden und das SmartRF05EB dient lediglich als Benutzerinterface bzw. gegebenenfalls als Energiequelle (Batteriebetrieb). Aus diesen beiden Zusammenstellungsvarianten ergeben sich für die Entwicklung des Laborsystems zwei Einsatzmöglichkeiten:

- Einsatz zur Bestimmung der Paketfehlerhäufigkeit (CC2520EM, CCMSP-EM430F2618 und SmartRF05EB) und
- Einsatz als Packetsniffer zur Aufzeichnung des Funkverkehrs (CC2520EM, SmartRF05EB und PC/Laptop).

4.2.1.1.3 Tests der Prototypen des HF-Teils

Die Überprüfung der Funktionsfähigkeit der Prototypen des HF-Teils erfolgte mit Hilfe des Entwicklungskits und der Beispielapplikation `per_test.c`, die in den CC2520 Software Examples [90] enthalten ist. Auf Grund des modularen Aufbaus des Kits kann das CC2520-Referenzdesign durch die 0.8 mm- und 1.5 mm-Prototypen ersetzt werden (siehe Abbildung 4.11), wodurch Messungen der Paketfehlerhäufigkeit ohne Anpassungen der Software durchgeführt werden konnten.

Die Messungen wurden unter folgenden Bedingungen durchgeführt:

- Distanz 20 m (mit zwei Zwischenwänden zwischen Sender und Empfänger),
- 10^5 Pakete pro Messung (100 Pakete pro Sekunde),
- nur das HF-Modul des Senders wurde variiert (Empfänger: CC2520EM Nummer 1),
- Sendeleistung -4 dBm und $+4$ dBm und
- Energieversorgung mittels USB (Sender) und externem Netzteil (Empfänger).

Die Ergebnisse der Messungen sind in Tabelle 4.3 zusammengefasst, die Werte der einzelnen Testläufe sind in den Tabellen A.2 und A.1 im Anhang aufgelistet. Am Display des Development Kits wird neben der Anzahl der fehlerfrei empfangenen Pakete ebenfalls die gemittelte Empfangsleistung der letzten 32 Pakete angezeigt, diese schwankt im Laufe der Messung deutlich wodurch der letzte angezeigte Wert nicht repräsentativ für die gesamte

Platine	P_{TX} in dBm	PER in %
CC2520EM Nr. 2	+4	6.1
	-4	25.4
CC2520EM Nr. 3	+4	8.8
	-4	51.4
FR4 0.8 mm	+4	3.4
	-4	3.9
FR4 1.5 mm	+4	0.9
	-4	4.0

Tabelle 4.3: Paketfehlerraten des CC2520EM und der entwickelten 0.8 mm- und 1.5 mm-Prototypen

Messdauer ist. Ausgewertet wurde daher lediglich die Anzahl der korrekt empfangenen Pakete.

Da bei den einzelnen Messungen nicht von einer gleichbleibenden Beeinträchtigung der Funkstrecke durch Störsender (z.B WLAN) ausgegangen werden kann, dürfen die Resultate nur als Anhaltspunkt für die Performance der getesteten Platinen angesehen werden. Bemerkenswert ist jedoch, dass die Prototypen bei einer Sendeleistung von -4 dBm eine deutlich höhere Anzahl an korrekt empfangenen Paketen aufweisen. Auf die möglichen Ursachen für die unterschiedliche Performance der CC2520EM-Platinen und des 0.8 mm-FR-Prototypen mit identischem Layout wurden bereits in Abschnitt 4.2.1.1.1 eingegangen. Für exakte Messergebnisse müssten die Tests unter kontrollierten Bedingungen durchgeführt werden. Die Funktionsfähigkeit ist auf Grund der erhaltenen Resultate jedoch mit Sicherheit gegeben.

4.2.1.2 Energieversorgung

Durch die Verwendung von zwei in Serie geschalteten NiMH-Zellen ergibt sich für die Energieversorgung des Laborsystems ein Ausgangsspannungsbereich von etwa $+1.8$ bis $+3$ V. Ein Betrieb des MSP430F2618 und des CC2520 ist entsprechend den Datenblättern [89, 94] bereits ab $+1.8$ V möglich. Da der Mikrocontroller bei $+1.8$ V jedoch lediglich mit einer Taktfrequenz von 4.15 MHz anstatt der ab $+3.3$ V möglichen 16 MHz betrieben werden kann und SD-Karten mindestens etwa $+2.7$ V benötigen [106], wurde ein DC/DC-Konverter zur Erzeugung einer stabilisierten Betriebsspannung von $+3.3$ V vorgesehen.

Gewählt wurde für diesen Zweck der LTC3526L von Linear Technology [54]. Wird der IC durch zwei Alkali/NiMH-Zellen versorgt, kann der LTC3526L bei einer Ausgangsspannung

von +3.3 V mit bis zu 200 mA belastet werden. Die Schaltfrequenz des LTC3526L liegt bei 1 MHz, wobei bei niedrigen Lasten ein Burst-Mode zur Verbesserung des Wirkungsgrades zur Verfügung steht. Bei einer Ausgangsspannung von +2.4 V liegt der Wirkungsgrad typischerweise zwischen 90 und 94 %.

Abbildung 4.12 zeigt die externe Beschaltung des LTC3526L zur Erzeugung einer Ausgangsspannung von +3.3 V. Auf die Auslegung des DC/DC-Konverters wird hier nicht näher eingegangen sondern diesbezüglich auf das Datenblatt [54] verwiesen.

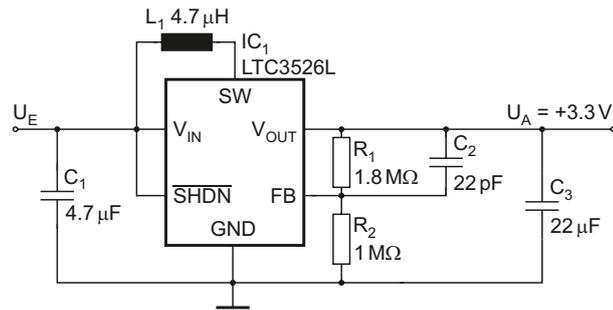


Abbildung 4.12: Schaltbild des LTC3526L

Da das zur Verfügung stehende Spannungsniveau bei zwei in Serie geschalteten NiMH-Zellen bei ca. +2.4 V liegt und diese Spannung bei einigen DC/DC-Konvertern (z.B. TPS63700 und TPS61082) kleiner als die minimal benötigte Eingangsspannung ist, werden beim Laborsystem die zusätzlich benötigten DC/DC-Konverter in Serie mit dem LTC3526L betrieben (siehe Abbildung 4.13). Die Eingangsspannung dieser Konverter liegt dadurch im Bereich der späteren Zwischenkreisspannung (vergleiche Abschnitt 5.2), wodurch im Idealfall bei beiden Systemen die gleichen Konverter eingesetzt werden können. Die DC/DC-Konverter zur Erzeugung der $\pm 2.5\text{ V}$ für die analoge Elektronik und der +24 V für die Versorgung der Konstantstromquelle des IEPE-Interfaces befinden sich auf den Platinen der allgemeinen Signalaufbereitung bzw. des IEPE-Interfaces und werden daher in den Abschnitten 4.4.4 und 4.5.1 beschrieben.

4.2.1.3 Strombedarf

Der Strombedarf der wesentlichen Komponenten des Basismoduls ist entsprechend den Datenblättern [89, 94] in Tabelle 4.4 zusammengefasst.

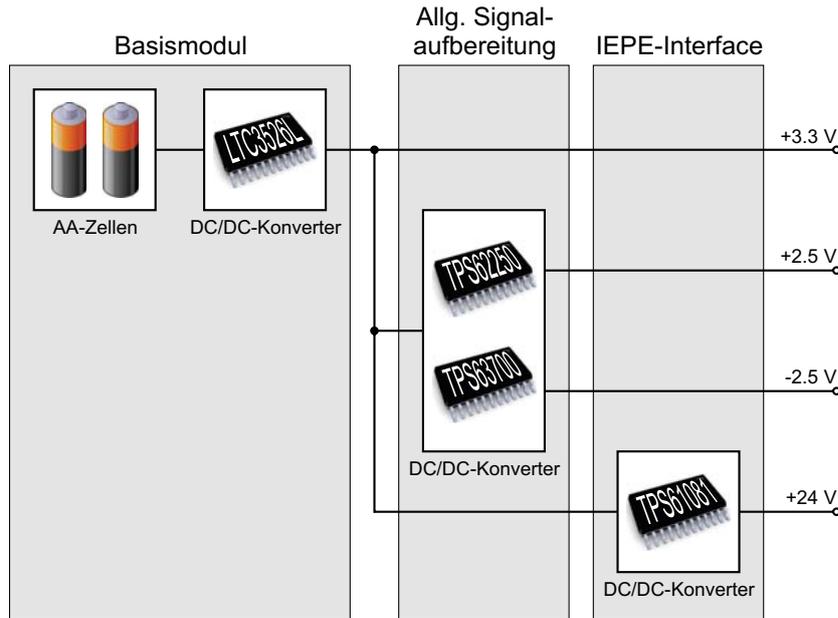


Abbildung 4.13: Energieversorgung des Laborsystems

Komponenten	Strom in mA
MSP430F2618	9.2 mA
CC2520 RX	18.5 mA
CC2520 TX	33.6 mA

Tabelle 4.4: Strombedarf MSP430F2618 und CC2520

4.2.2 Beschreibung der Firmware

Die Firmware des Basisknotens beruht auf dem Z-Stack von Texas Instruments (Version 2.3.0), wobei die Beispielapplikation SerialApp den Ausgangspunkt für die Entwicklung darstellte. Unter Verwendung der SerialApp fungiert das Development-Kit als ZigBee-Modem. Es setzt, wie in Abbildung 4.14 dargestellt, einen seriellen RS232-Datenstrom in entsprechende ZigBee-Pakete um, die per Funk zu einem zweiten Kit übertragen werden

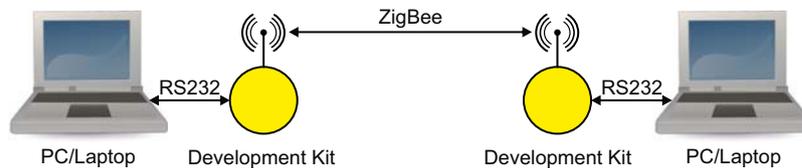


Abbildung 4.14: Beispielapplikation SerialApp von TI

und dort wiederum über die RS232-Schnittstelle an den Empfänger weitergeleitet werden. Die Kommunikation mittels der SerialApp kann bidirektional erfolgen. Eine Adressierung von verschiedenen Netzwerkknoten ist im Rahmen der SerialApp jedoch nicht möglich.

Die SerialApp musste daher um ein entsprechendes, paketbasiertes Kommunikationsprotokoll erweitert werden, das die Adressierung von einzelnen Empfängern im ZigBee-Netzwerk erlaubt. Die Adressierung der Pakete erfolgt bei ZigBee PRO über die stochastisch zum Zeitpunkt des Beitritts des Knotens zum Netzwerk zugewiesene kurze Netzwerkadresse. Da diese a priori nicht bekannt ist und erst während des Betriebs des Netzwerks dynamisch vergeben wird, muss eine Möglichkeit zur Abfrage der kurzen Adresse sowie der fix vergebenen MAC-Adresse der einzelnen Knoten vorgesehen werden. Dies muss sowohl für das lokale Basismodul als auch über das Netzwerk möglich sein. Denn ohne die Zuordnung zwischen kurzer Netzwerkadresse und MAC-Adresse wäre die gezielte Adressierung eines konkreten Knotens mittels der kurzen Netzwerkadresse nicht möglich.

Sowohl das RS232-Kommunikationsprotokoll als auch die zusätzlich implementierten Netzwerkmanagementfunktionen werden in den folgenden Abschnitten näher erläutert.

4.2.2.1 RS232-Kommunikation

Die Kommunikation zwischen Basismodul und den etwaigen Zusatzmodulen (z.B. USB-Interface und Datenspeicher) erfolgt mittels der RS232-Schnittstelle. Die Konfigurationsparameter der Schnittstelle sind in Tabelle 4.5 zusammengefasst. Die Baudrate wurde bewusst möglichst niedrig gewählt, da dem Mikrocontroller hierdurch zwischen den einzelnen übertragenen Bytes ein größeres Zeitfenster für das Auslesen des RS232-Empfangsregister zur Verfügung steht und somit größere Latenzzeiten beim Aufruf der zugehörigen Interrupt Service Routine (ISR) ohne Datenverlust zulässig sind. Die untere Grenze wird hierbei durch die maximal benötigte Übertragungsrate festgelegt. Bei der Übertragung eines Messkanals über das Funknetz sind bei einer Abtastrate von 1 kS/s und unter Vernachlässigung der Paketheader und der Stoppbytes mindestens 2 kB/s bzw. 20 kBd erforderlich.

Parameter	Wert
Baudrate	38 400
Datenbits	8
Stoppbits	1
Parity	Keine
Flusssteuerung	Aus

Tabelle 4.5: RS232-Kommunikationsparameter des Basismoduls

Für die Kommunikation wurden daher 38.4kBd gewählt, woraus ein Zeitintervall von 260.4µs zwischen zwei übertragenen Bytes resultiert.

Auf die Verwendung von Handshaking-Signalen wurde im Rahmen des Laborsystems verzichtet. Mittels der RTS- und CTS-Leitungen können miteinander kommunizierende Geräte durch Setzen des jeweiligen Ausgangs dem Gegenpart mitteilen, ob sie bereit sind Daten zu empfangen [6]. Auf Seite des Mikrocontrollers besteht die schnellste Möglichkeit zur Reaktion auf empfangene Datenbytes in der Verwendung einer ISR. Hardwareseitig ist jedoch keine Möglichkeit der Flusststeuerung mittels RTS und CTS vorgesehen. Durch eine Implementierung der Handshaking-Funktionalität in der Firmware ließe sich somit der kritische Fall, dass auf Grund zu hoher Latenzzeiten beim Aufruf der Interruptservice-routine der RS232-Schnittstelle noch nicht ausgelesene Daten im Empfangsregister durch neu empfangene Daten überschrieben werden können, nicht abdecken. Eine entsprechende Absicherung der mittels RS232 übertragenen Daten hinsichtlich der Erkennung von Übertragungsfehlern ist daher auf Paketebene vorzusehen.

Die RS232-Pakete bestehen aus dem Header, der eigentlichen Nachricht und zwei Stoppbytes. Der Header setzt sich wiederum aus einem Startbyte, der Paketlänge, dem Pakettyp und der kurzen ZigBee-Zieladresse zusammen. Mittels des Pakettyps wird festgelegt, ob das Paket für das lokale Basismodul, ein Basismodul im Funknetz oder ein mit einem Basismodul im Funknetz verbundenes Zusatzmodul bestimmt ist. Der allgemeine Paket-aufbau ist in Abbildung 4.15 dargestellt, wobei der eigentliche zu übermittelnde Inhalt der Nachricht in gelb und der Header sowie die Stoppbytes in blau hinterlegt sind. Die Bedeutung der einzelnen Bytes ist in Tabelle 4.6 zusammengefasst.



Abbildung 4.15: RS232-Paketaufbau

Die eigentliche Nachricht besteht aus mindestens zwei Bytes, der Sequenznummer und dem Nachrichtentyp. Die Sequenznummer wird nach jeder abgesetzten Nachricht um Eins erhöht und ermöglicht so auf Empfängerseite die Identifizierung von doppelt empfangenen Nachrichten. Jedem Kommando bzw. jeder daraufhin als Antwort generierten Nachricht ist ein eindeutiger Nachrichtentyp zugeordnet, der auch den weiteren Aufbau der Nachricht (MSGx) festlegt (vergleiche Abschnitt 4.2.2.2 und Anhang C).

Ein fehlerhafter Paketaufbau oder ein nicht innerhalb des Timeoutintervalls von 50 ms vollständig an das Basismodul übertragenes Paket führt zu einer Fehlermeldung des Basismoduls, die durch den Wert 0x44 des MTP-Bytes gekennzeichnet ist.

Byte	Bedeutung
STA	Startbyte: 0xCA
LEN	Länge des Pakets in Bytes (exklusive Start- und Stoppbytes)
UPT	Pakettyp (Ziel des Pakets) 0x00: Zusatzmodul des adressierten Basisknotens 0x1F: Basismodul im Netz 0xE3: lokaler Basisknoten
DAH	Highbyte der 16-Bit-Zieladresse (ZigBee-Kurzadresse)
DAL	Lowbyte der 16-Bit-Zieladresse
SEQ	Sequenznummer
MTP	Nachrichtentyp
MSGx	optionale zusätzliche Nachrichtenbytes
STO1	Stoppbyte 1: 0xD5
STO0	Stoppbyte 0: 0xF5

Tabelle 4.6: Bedeutung der einzelnen Bytes des Kommunikationsprotokolls

Am Beispiel des in Abbildung 4.16 dargestellten Netzwerks, bestehend aus einem Koordinator (Adresse 0x0000) und drei weiteren Basisknoten – wovon zwei mit Zusatzmodulen ausgerüstet sind – soll das Adressierungsschema veranschaulicht werden.

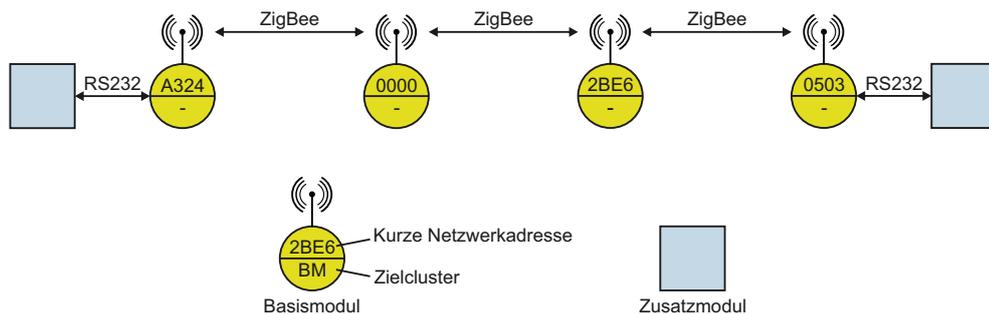


Abbildung 4.16: Kommunikation innerhalb des Laborsystems

Nachrichten eines Zusatzmoduls können drei verschiedene Empfänger betreffen:

- Ein Zusatzmodul eines anderen Knotens im Funknetz,
- ein Basismodul eines anderen Netzwerkknotens und
- das lokale Basismodul.

Im ersten Fall (siehe Abbildung 4.17) muss neben der Adresse des Zielknotens, in diesem Fall 0x0503, der Pakettyp 0x00 angegeben werden. Das Paket wird daraufhin vom lokalen Basismodul verarbeitet und exklusive Header und Stoppbyte über das Netzwerk an das

Basismodul des Zielknotens übertragen. Die Zielzuordnung zwischen Basismodul oder Zusatzmodul erfolgt über den Clustermechanismus von ZigBee. Der RS232-Schnittstelle ist Cluster 3 (UART – Universal Asynchronous Receiver Transmitter) zugeordnet (Pakettyp 0x00) und dem Basismodul (BM) Cluster 4 (Pakettyp 0x1F). Im Zielknoten übernimmt die Empfangsroutine des adressierten Clusters schließlich die Nachricht und leitet sie im Fall des Clusters 3 nach Hinzufügen von Header und Stopbytes an das angeschlossene Zusatzmodul weiter.

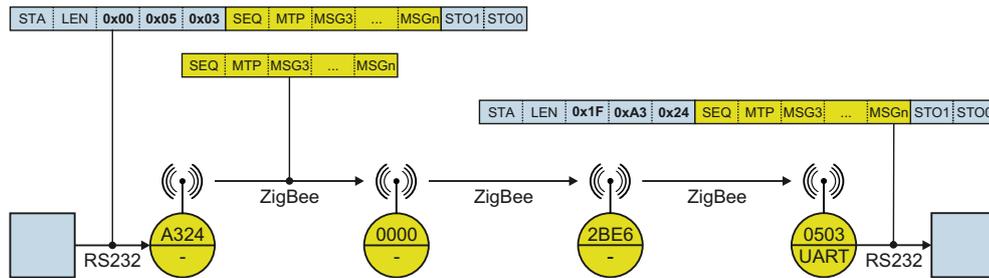


Abbildung 4.17: Beispiel – Nachricht vom lokalen Zusatzmodul an ein Zusatzmodul im Netzwerk

Eine Sonderstellung nimmt die Adresse 0xFF ein, die einen sogenannten Broadcast bewirkt und alle Knoten des Netzwerks anspricht.

Ist das Ziel der Übertragung das Basismodul des Knotens 0x0503, so muss als Pakettyp 0x1F angegeben werden. Diese Adressierung von Basismodulen im Netzwerk (siehe Abbildung 4.18) wird im Wesentlichen zur Verwaltung des Netzwerkes benötigt.

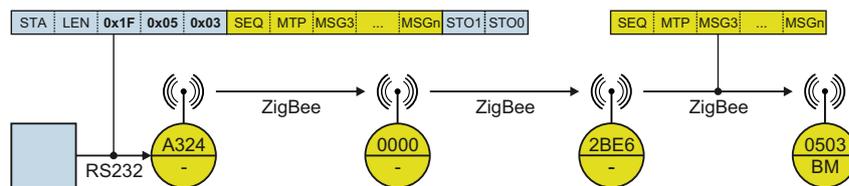


Abbildung 4.18: Beispiel – Nachricht vom lokalen Zusatzmodul an ein Basismodul im Netzwerk

Nachrichten an das lokale Basismodul werden mittels des Pakettyps 0xE3 abgesetzt (siehe Abbildung 4.19). In diesem Fall kann eine beliebige ZigBee-Adresse angegeben werden, die Verarbeitung der Nachricht im lokalen Basismodul erfolgt automatisch beim Erkennen des Pakettyps 0xE3. Das Senden von Nachrichten an das lokale Basismodul ist ebenfalls in der Regel lediglich für die Verwaltung des Netzwerkes erforderlich.

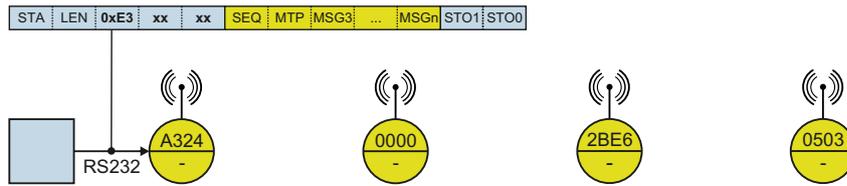


Abbildung 4.19: Beispiel – Nachricht vom lokalen Zusatzmodul an das lokale Basismodul

Per Funk empfangene Nachrichten, die über die RS232-Schnittstelle weitergeleitet werden, sind in jedem Fall als Pakettyp 0x1F gekennzeichnet. Eine Unterscheidung ob diese von einem Basisknoten (siehe Abbildung 4.20) oder von einem anderen Zusatzmodul (wie in Abbildung 4.17 – Knoten 0xA324) stammen ist anhand des Pakettyps im Header nicht möglich. Diese Zuordnung kann nur über den Nachrichteninhalt erfolgen.

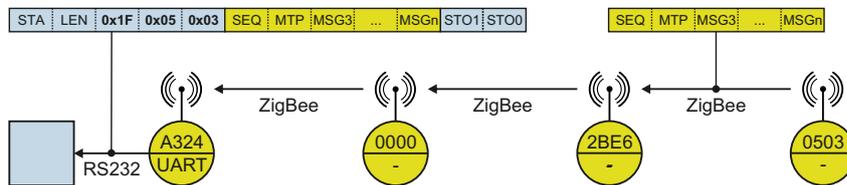


Abbildung 4.20: Beispiel – Nachricht eines Basismoduls im Netzwerk an das lokale Zusatzmodul

Nachrichten, die vom lokalen Basismodul generiert wurden, und an das lokale Zusatzmodul gerichtet sind, können anhand des Pakettyps 0xE3 identifiziert werden (siehe Abbildung 4.21).

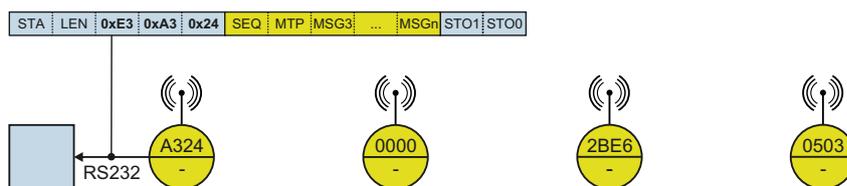


Abbildung 4.21: Beispiel – Nachricht des lokalen Basismoduls an das lokale Zusatzmodul

4.2.2.2 Netzwerkmanagementfunktionen

Die Adressierung der Knoten im ZigBee-Netzwerk erfolgt mittels einer 16-Bit-Adresse, die stochastisch beim Aufbau des Netzes vergeben wird und somit zu Beginn unbekannt ist. Um Knoten gezielt ansprechen zu können, muss sowohl diese ZigBee-Kurzadresse als

auch die jedem Knoten fix zugeordnete 64-Bit-MAC-Adresse abgefragt werden können. Hierfür wurde eine Funktion integriert, die beide Adressen auf Anfrage zurückliefert.

Der Z-Stack ermittelt zu jedem empfangenen ZigBee-Paket den zugehörigen LQI (Link Quality Indicator), RSSI (Received Signal Strength Indicator) und den Korrelationswert bei der Detektion der IEEE 802.15.4 Preamble. Der LQI-Wert wird aus der geschätzten Empfangsleistung (RSSI-Wert) in dBm mittels der Beziehung

$$LQI = 255 \frac{RSSI + 85}{91} \quad (4.6)$$

berechnet. Ein RSSI-Wert von -85 dBm entspricht daher einem LQI-Wert von 0 und $+6$ dBm entsprechen einem LQI-Wert von 255. Der LQI-Wert sollte möglichst hoch sein. Er gibt einen ersten Anhaltspunkt, ob Verbindungsprobleme zu erwarten sind. Ein Mechanismus zur Feststellung der Verbindungsqualität zwischen benachbarten Knoten wurde daher ebenfalls implementiert.

Im Folgenden werden die Befehls- und Antwortpakete dieser Funktionen näher beschrieben.

4.2.2.2.1 Anforderung der Netzwerkadresse

SEQ: 0x12

SEQ: Sequenznummer

Antwort:

- **Adressinformation**

SEQ: 0x51 SAH SAL EA7 EA6 EA5 EA4 EA3 EA2 EA1 EA0

SEQ: Sequenznummer

SAH: kurze Netzwerkadresse (Highbyte)

SAL: kurze Netzwerkadresse (Lowbyte)

EAx: MAC-Adresse (Most significant Byte: EA7)

Beispiel: Anforderung aller Netzwerkadressen im Netz (exklusive lokalem Knoten)

Die Anforderung der Netzwerkadressen aller dem Netzwerk angehörigen Knoten mit Ausnahme des lokalen Knotens kann über einen Broadcast mit dem Pakettyp 0x1F erfolgen. Dies ist in Abbildung 4.22 dargestellt. Alle adressierten Knoten antworten daraufhin mit der jeweiligen Adressinformation. Abbildung 4.23 zeigt dies exemplarisch für den Knoten 0x0503.

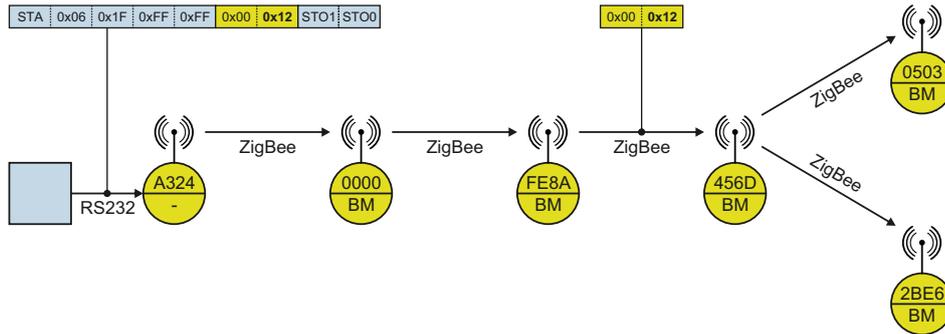


Abbildung 4.22: Anforderung der Netzwerkadresse mittels Broadcast über das Funknetz

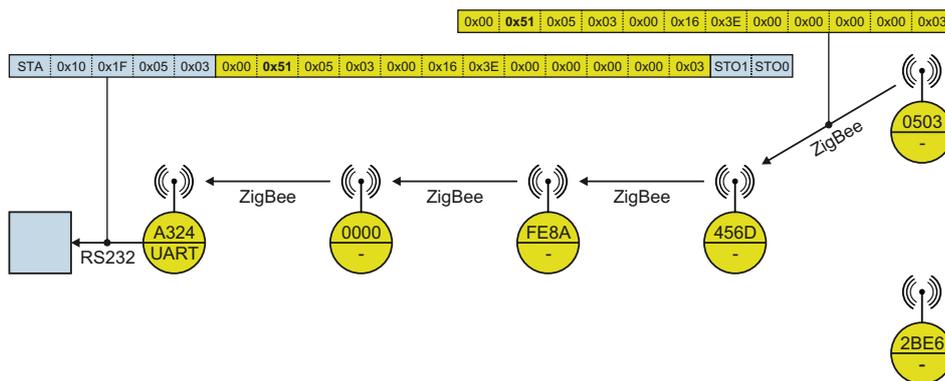


Abbildung 4.23: Übermittlung der Netzwerkadresse des Knotens 0x0503 über das Netzwerk

Beispiel: Anforderung der Adressinformation des lokalen Knotens

Abbildung 4.24 und Abbildung 4.25 zeigen sowohl das Anforderungs- als auch das Antwortpaket, die im Zuge der Feststellung der Netzwerkadresse des lokalen Knotens ausgetauscht werden.

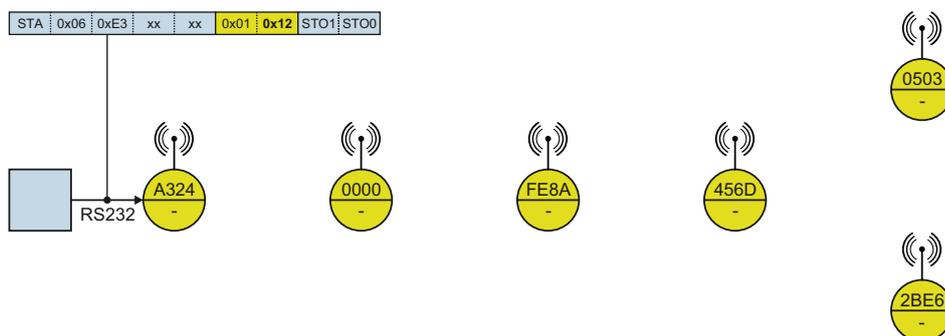


Abbildung 4.24: Anforderung der Netzwerkadresse des lokalen Knotens

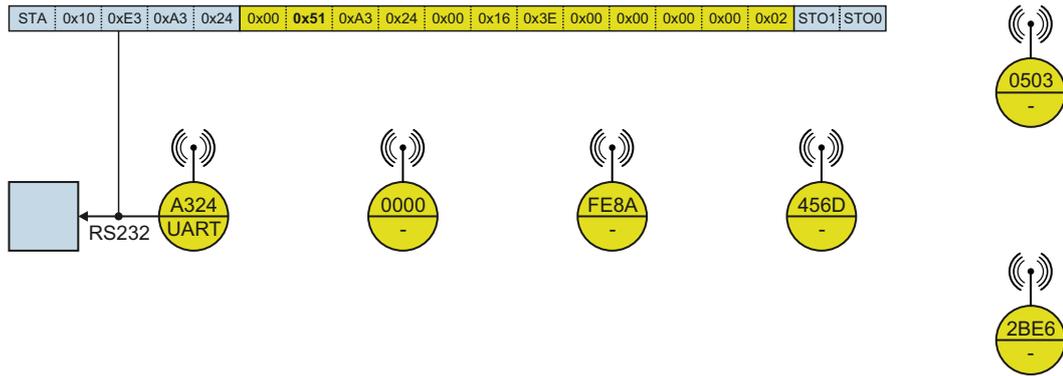


Abbildung 4.25: Übermittlung der lokalen Adressinformation

4.2.2.2.2 Feststellung der Verbindungsqualität (LQI)

SEQ: 0x24 DLY

SEQ: Sequenznummer

DLY: Zeitintervall in ms innerhalb dessen die Antworten der Nachbarknoten eingetroffen sein müssen

Antwort:

- LQI-Information

SEQ: 0x63 NUM: SAH: SAL: DAH: DAL: LSD: RSD: CSD: LDS: RDS: CDS

SEQ: Sequenznummer

NUM: Anzahl der Datensätze (hier immer 0x01)

SAH: Adresse des Quellknotens des Broadcasts (Highbyte)

SAL: Adresse des Quellknotens des Broadcasts (Lowbyte)

DAH: Zieladresse (Empfänger des Broadcasts) – Highbyte

DAL: Zieladresse (Empfänger des Broadcasts) – Lowbyte

LSD: LQI-Wert, Paketrichtung: Quelladresse → Zieladresse

RSD: RSSI-Wert, Paketrichtung: Quelladresse → Zieladresse

CSD: Korrelationswert, Paketrichtung: Quelladresse → Zieladresse

LDS: LQI-Wert, Paketrichtung: Zieladresse → Quelladresse

RDS: RSSI-Wert, Paketrichtung: Zieladresse → Quelladresse

CDS: Korrelationswert, Paketrichtung: Zieladresse → Quelladresse

Beispiel: Anforderung der LQI-Information aller Nachbarn des Knotens 0x456D

Im ersten Schritt (Abbildung 4.26) wird das Kommando zur Feststellung der LQI-Information an den Knoten 0x456D gesendet. Der Knoten 0x456D sendet daraufhin einen

Broadcast mit der Reichweite Eins aus und fordert die LQI-Information aller Knoten an, die sich in seiner direkten Funkreichweite befinden (Abbildung 4.27). Diejenigen Knoten, die diese Anforderung erhalten haben, senden die LQI-Information bezüglich des empfangenen Anforderungspakets zurück an den Knoten 0x456D. Dies ist beispielhaft in Abbildung 4.28 für den Knoten 0x0503 dargestellt. Analoges gilt für die Knoten 0xFE8A und 0x2BE6. Innerhalb des in der Anforderung zu Beginn angegebenen Zeitintervalls (im Beispiel 0.1 s) werden die von den Nachbarknoten empfangenen Pakete (zuzüglich der LQI-Information dieser Antwortpakete) an den Knoten 0xA324 weitergeleitet (Abbildung 4.29).

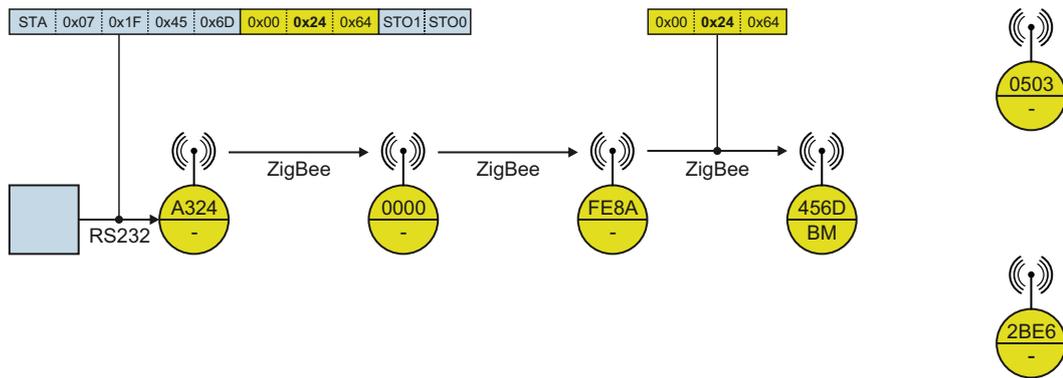


Abbildung 4.26: Bestimmung des LQI – Schritt 1 (Anforderung)

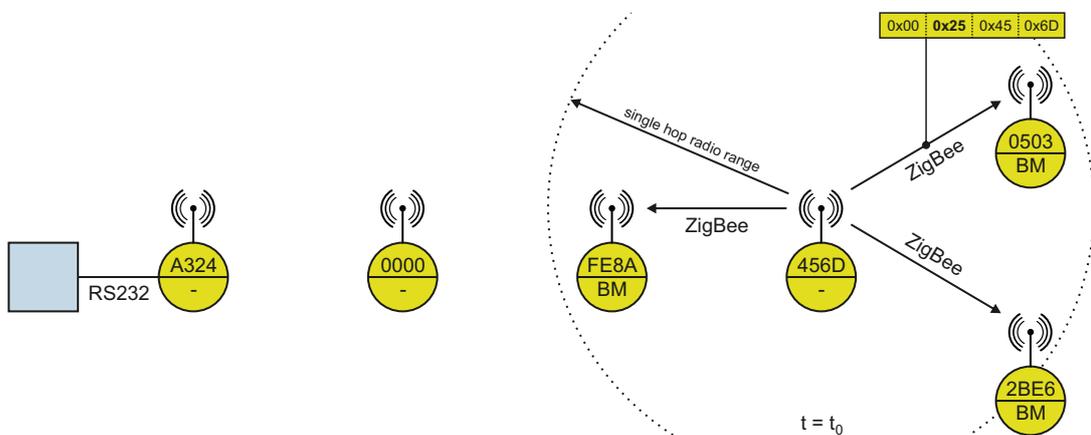


Abbildung 4.27: Bestimmung des LQI – Schritt 2 (Single Hop Broadcast)

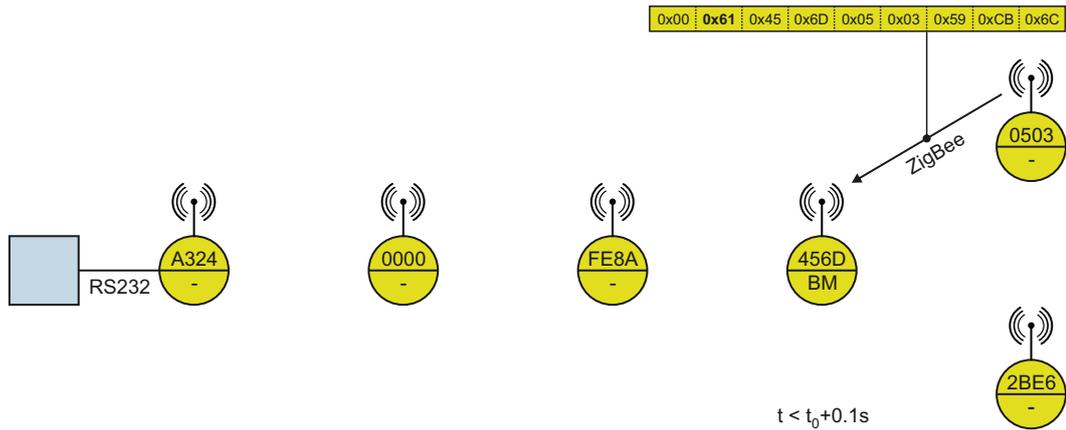


Abbildung 4.28: Bestimmung des LQI – Schritt 3 (Antwort der Nachbarknoten)

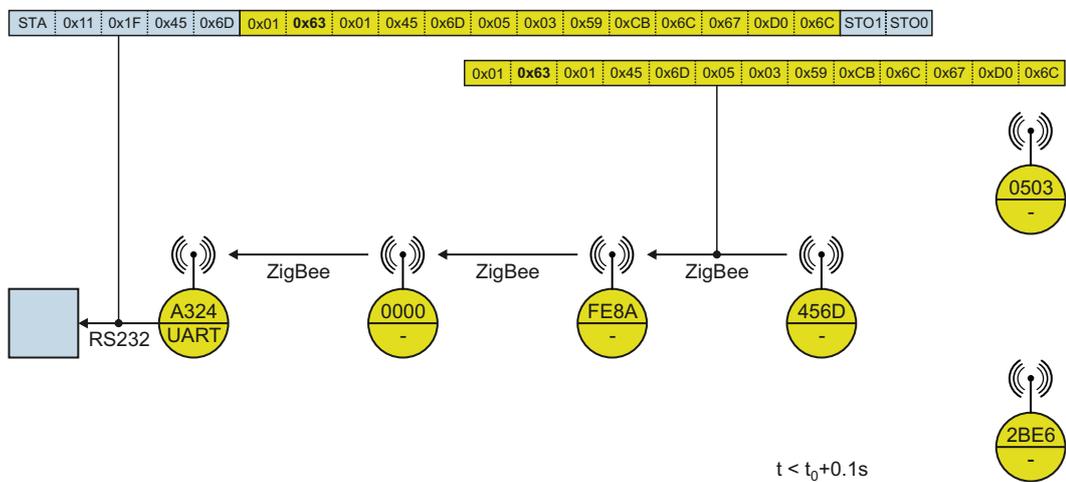


Abbildung 4.29: Bestimmung des LQI – Schritt 4 (Rückmeldung der Ergebnisse)

4.2.2.3 Z-Stack

Die Basis der Firmware des Basisknotens bildet der kostenlos von TI erhältliche Z-Stack [101] in der Version 2.3.0. Dieser ist für Mikrocontroller der MSP430-Reihe und ARM Cortex-M3 Prozessoren verfügbar, wobei jedoch beträchtliche Anteile des Z-Stacks nicht als Sourcecode vorliegen. Eine Anpassung des Z-Stacks an die eigenen Bedürfnisse ist daher leider nicht möglich. Weiters ist für die Verwendung des Z-Stacks die Embedded Workbench für MSP430 von IAR Systems zwingend erforderlich, andere Compiler werden nicht unterstützt.

Der Z-Stack sowie eigene Applikationen werden unter einem kooperativen Betriebssystem, dessen Funktionen mittels des OSAL (Operating System Abstraction Layer) APIs

(Application Programming Interfaces) zur Verfügung gestellt werden, ausgeführt. Auf eine detaillierte Beschreibung der Funktionsweise und der einzelnen Funktionen des Z-Stacks wird im Rahmen dieser Arbeit aus Platzgründen verzichtet und diesbezüglich auf die im Downloadarchiv des Z-Stacks [101] enthaltene Dokumentation verwiesen.

4.2.2.3.1 RS232-Paketverarbeitung

Die ebenfalls in den jeweiligen prozessorspezifischen Archiven [101] enthaltene Serial-App wurde entsprechend den Anforderungen um die bereits in den Abschnitten 4.2.2.1 und 4.2.2.2 beschriebene Funktionalität erweitert. Die Überprüfung der, über die RS232-Schnittstelle empfangenen, Pakete erfolgt hierbei Byte für Byte direkt in der ISR. Nach dem Empfang eines fehlerfreien Pakets wird der zugehörige Event ausgelöst und das Paket entweder im Knoten selbst verarbeitet oder über Funk an den Zielknoten weitergeleitet.

Die Signale der RS232-Schnittstelle im tatsächlichen Betrieb des Basismoduls sind in den Abbildungen 4.30 und 4.31 dargestellt. Das Basismodul ist hierbei über das in Abschnitt 4.6 beschriebene USB-Interface mit einem PC verbunden. Über die Funkschnittstelle werden ADC-Werte eines Messkanals empfangen und anschließend per RS232 an den USB-Interface-IC weitergeleitet (Signal RS-232-TX). Dies entspricht dem maximal zu erwartenden Datenaufkommen. Wie in Abbildung 4.30 zu erkennen ist, kann dieses mit den in Abschnitt 4.2.2.1 festgelegten Übertragungsparametern problemlos bewältigt werden.

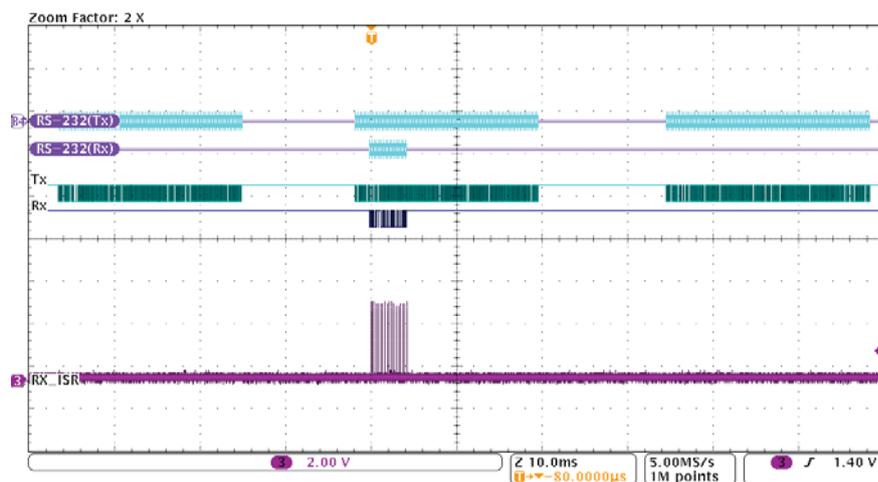


Abbildung 4.30: RS232-Signale beim Senden und Empfangen von Paketen inkl. Aktivität des RX-Anteils der ISR

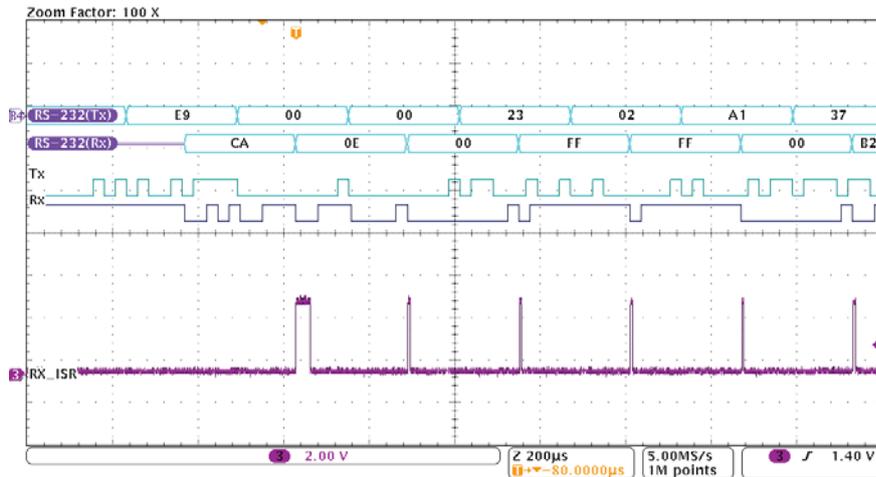


Abbildung 4.31: Detailansicht – RS232-Signale beim Senden und Empfangen von Paketen inkl. Aktivität des RX-Anteils der ISR

Zeitgleich zum Transfer des zweiten Datenpakets wurde über den PC ein Konfigurationspaket (Signal RS-232-RX) abgesetzt. Dies ist als Detailansicht in Abbildung 4.31 dargestellt. Im unteren Bereich der Abbildung ist zusätzlich die Aktivität des RX-Codes der zugehörigen ISR abgebildet. Kommunikationsprobleme auf Grund zu langer Latenzzeiten beim Aufruf der ISR treten im Betrieb nicht auf. Die maximale Ausführungsdauer des RX-Teils der ISR beträgt $33\ \mu\text{s}$. Die längere Aktivität der ISR beim ersten und letzten Byte eines Pakets begründet sich im Starten bzw. Deaktivieren eines Timers (Pakettimeout siehe 4.2.2.1) über die OSAL API.

Innerhalb der ISR wird das Paket nach Detektion des Startbytes hinsichtlich eines zulässigen Pakettyps (UPT) und dem Vorhandensein der beiden Stoppbytes (STO1 und STO0) an der durch LEN angegebenen Position überprüft. In der derzeitigen Form erfolgt somit keine Absicherung des Paketinhalts gegenüber Übertragungsfehlern. Der Verlust ganzer Bytes kann jedoch mit einer Wahrscheinlichkeit von $1 - 2^{-16} = 0.999985$ festgestellt werden.

Da große Teile des zugrundeliegenden Z-Stacks nicht als Sourcecode verfügbar sind und somit auch keine Kenntnis darüber vorliegt, über welche (maximalen) Zeitspannen Interrupts global deaktiviert werden, ist vor allem die Absicherung hinsichtlich des Verlustes einzelner Bytes unabdingbar. Auf die Absicherung des Paketinhaltes wurde im Rahmen des Laborsystems aus Gründen der einfacheren Bedienbarkeit des Systems mittels des in Abschnitt 4.6 beschriebenen Terminalprogramms bewusst verzichtet. In weiterer Folge sollte jedoch die Absicherung mittel Prüfsumme [60] oder CRC (Cyclic Redundancy Check) [68] anstatt der Verwendung fixer Stoppbytes in Betracht gezogen werden. Auf

Grund der Rückmeldungen der einzelnen Knoten auf Befehlspakete bzw. das unmittelbare Abfragen von zuvor gesetzten Parametern kann jedoch auf Systemebene eine Überprüfung hinsichtlich einer korrekten Übertragung von Befehlen bzw. der Ausführung der gewünschten Aktion erfolgen.

Ausgehend von der Tatsache, dass im Betrieb die Übertragung von ADC-Werten den überwiegenden Anteil des gesamten übertragenen Datenaufkommens ausmacht, wurden Start- und Stopbytes so gewählt, dass diese nicht in den Wertebereich der kodierten Abtastwerte (vergleiche Anhang C.8) fallen.

4.2.2.3.2 Latenzzeiten beim Senden und Empfangen

Die Latenzzeiten zwischen dem Absetzen von Nachrichten in der Z-Stack-Applikation und dem tatsächlichen Beginn der Funkübertragung sowie zwischen dem Empfang einer Nachricht durch den CC2520 und dem entsprechenden Empfangsevent im Z-Stack, sind für eine mögliche Synchronisation der Zeitbasen der Module des Messsystems von entscheidender Bedeutung. Aus diesem Grund wurden diesbezüglich Messungen durchgeführt.

Unmittelbar vor dem Absetzen eines Broadcasts im Koordinator und nach dem Auftreten des Empfangsevents im zweiten Knoten wurde jeweils ein Ausgangspin (GPIO – General Purpose Input/Output) der Mikrocontroller kurzzeitig auf High gesetzt. Der genaue Sende- und Empfangszeitpunkt der Funknachricht kann über das SFD-Signal (Start of Frame Delimiter, siehe [89]) auf CC2520_GPIO4 bestimmt werden und wurde gemeinsam mit dem GPIO-Signalen der MSP430s mittels eines Oszilloskops aufgezeichnet. Die betrachteten Ereignisse sowie die Bezeichnung der einzelnen gemessenen Zeitintervalle sind in Abbildung 4.32 dargestellt. Da es sich bei der abgesetzten Nachricht um einen

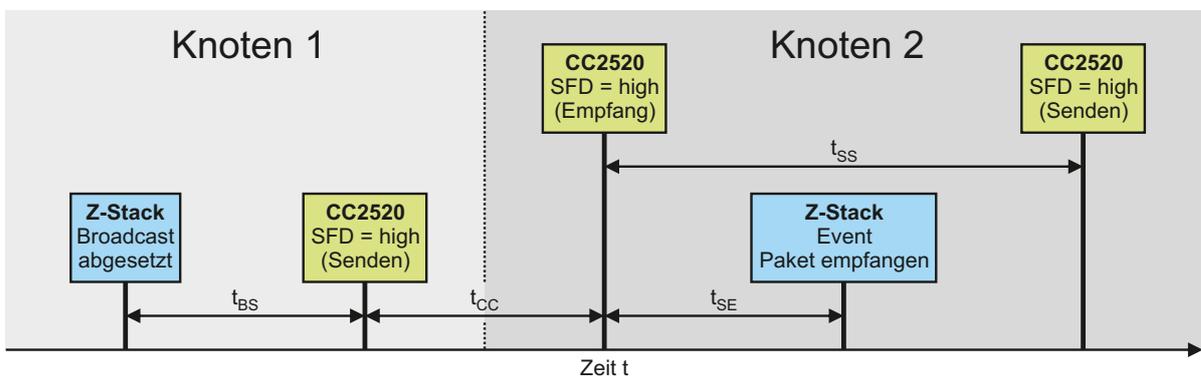


Abbildung 4.32: Zeitintervalle beim Senden und Empfangen von Nachrichten

Broadcast handelte, erfolgt eine erneute Aussendung der Nachricht durch den zweiten Knoten ohne ein Zutun der Benutzerapplikation. Die Mittelwerte und Standardabweichungen der gemessenen Zeitintervalle (siehe Tabelle A.3 im Anhang) sind in Tabelle 4.7 zusammengefasst.

Intervall	\bar{t} in s	σ_t in s
t_{BS}	$1.753 \cdot 10^{-3}$	$0.672 \cdot 10^{-3}$
t_{CC}	$3.344 \cdot 10^{-6}$	$41.30 \cdot 10^{-9}$
t_{SE}	$2.276 \cdot 10^{-3}$	$19.65 \cdot 10^{-6}$
t_{SS}	$3.498 \cdot 10^{-3}$	$0.624 \cdot 10^{-3}$

Tabelle 4.7: Zeitintervalle beim Senden und Empfangen von Nachrichten

Auffällig ist hierbei die hohe Standardabweichung der mit dem Absenden bzw. Weiterleiten der Nachricht assoziierten t_{BS} und t_{SS} Werte. Für ein Synchronisierungssignal, das durch den Knoten 1 als Broadcast zum Zeitpunkt $t = 0$ abgesendet wurde, ergibt sich für den Knoten n ($n \geq 2$) nach $n - 2$ Zwischenknoten der Empfangszeitpunkt zum Zeitpunkt

$$t_E(n) = t_{BS} + (n - 2)(t_{CC} + t_{SS}) + t_{CC} + t_{SE}. \quad (4.7)$$

Die Standardabweichung von $t_{E,n}$ erhält man mittels des Gauß'schen Fehlerfortpflanzungsgesetzes für abhängige Größen $y = f(x_1, \dots, x_n)$ [85]

$$\sigma_y = \sqrt{\sum_{j=1}^n \left(\left. \frac{\partial f(x_1, \dots, x_n)}{\partial x_j} \right|_{\bar{x}_j} \sigma_j \right)^2}. \quad (4.8)$$

Es folgt daher für die Standardabweichung $\sigma_{t,E}(n)$ des Empfangszeitpunkts am n -ten Knoten

$$\sigma_{t,E}(n) = \sqrt{\sigma_{t,BS}^2 + (n - 1)^2 \sigma_{t,CC}^2 + (n - 2)^2 \sigma_{t,SS}^2 + \sigma_{t,SE}^2}. \quad (4.9)$$

Mit den Werten aus Tabelle 4.7 erhält man für den zweiten bis sechsten Knoten die in Tabelle 4.8 angeführten Werte. Am fünften Knoten trifft eine abgesendete Nachricht im Mittel also nach 14.5 ms ein, wobei die Standardabweichung 2.0 ms beträgt. 68.3% aller Nachrichten würden daher innerhalb von 12.5 bis 16.5 ms nach deren Übergabe an den Z-Stack im Sender bei der Applikation im Empfänger ankommen. Vor allem auf Grund der hohen Schwankungsbreite ist eine Synchronisierung der Zeitbasen der einzelnen Knoten mit dem Ziel einer synchronen Abtastung der Messsignale mit 1 kS s^{-1} per Funk mit dem Z-Stack nicht praktikabel.

n	$\bar{t}_E(n)$ in ms	$\sigma_{t,E}(n)$ in ms
2	4.032	0.672
3	7.534	0.917
4	11.04	1.418
5	14.54	1.989
6	18.04	2.585

Tabelle 4.8: Empfangszeitpunkte von Nachrichten am n -ten Knoten

4.3 Datenspeicher

Das Datenspeichermodul besteht im Wesentlichen aus einem Mikrocontroller, der die Ausgangsspannung U_{CO} der allgemeinen Signalaufbereitung mittels des integrierten ADCs (12 Bit Auflösung) abtastet und verarbeitet sowie einem Massenspeicher zur nichtflüchtigen Speicherung der Sensorsignale. Die Verarbeitung der Abtastwerte umfasst die strukturierte Speicherung der Sensordaten aller Kanäle mittels eines hierfür ausgelegten Dateiformats, die Übergabe der Sensordaten eines Kanals an das Basismodul zwecks Übertragung per Funk, sowie abhängig vom konkreten Hardwareaufbau, die Verwaltung des Dateisystems vom Massenspeicher.

Weiters übernimmt der Mikrocontroller des Datenspeichers die Ansteuerung der allgemeinen Signalaufbereitung sowie der Sensorinterfaces. Die Kommunikation zwischen Datenspeicher und Basismodul erfolgt mittels RS232-Schnittstelle (D_TX und D_RX) und der in Abschnitt 4.2.2.1 definierten Datenpakete. Als Massenspeicher wurden für das Laborsystem microSDHC-Karten auf Grund ihrer geringen Baugröße gewählt. Der schematische Aufbau des Datenspeichers sowie die Signalflüsse zwischen Datenspeicher und Basismodul sowie der allgemeinen Signalaufbereitung sind in Abbildung 4.33 dargestellt. Die Pinbelegungen sind in Anhang D.2 dokumentiert.

4.3.1 Beschreibung der Hardware

Die Anbindung einer SD-Karte an den Mikrocontroller kann einerseits direkt über die SPI-Schnittstelle erfolgen, wobei der Mikrocontroller in diesem Fall für die Verwaltung des Dateisystems verantwortlich ist, oder andererseits über einen zwischengeschalteten FAT-Treiber-Baustein (FAT – File Allocation Table) erfolgen. Im zweiten Fall erfolgt die Verwaltung des Dateisystems durch den Treiberbaustein und im Mikrocontroller muss lediglich das Kommunikationsprotokoll des Treibers implementiert werden. Die Kommunikation mit dem Treiber-IC erfolgt dabei in der Regel mittels der RS232- oder SPI-

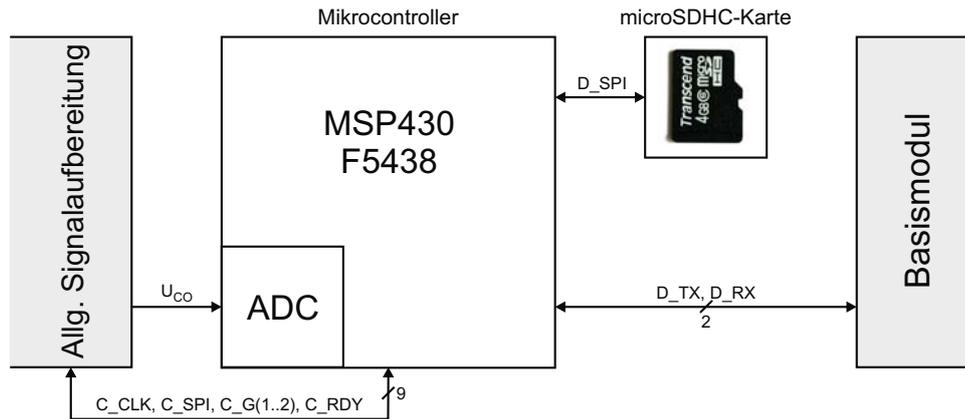


Abbildung 4.33: Signalverbindungen zwischen Datenspeichermodul und Basismodul sowie der allgemeinen Signalaufbereitung

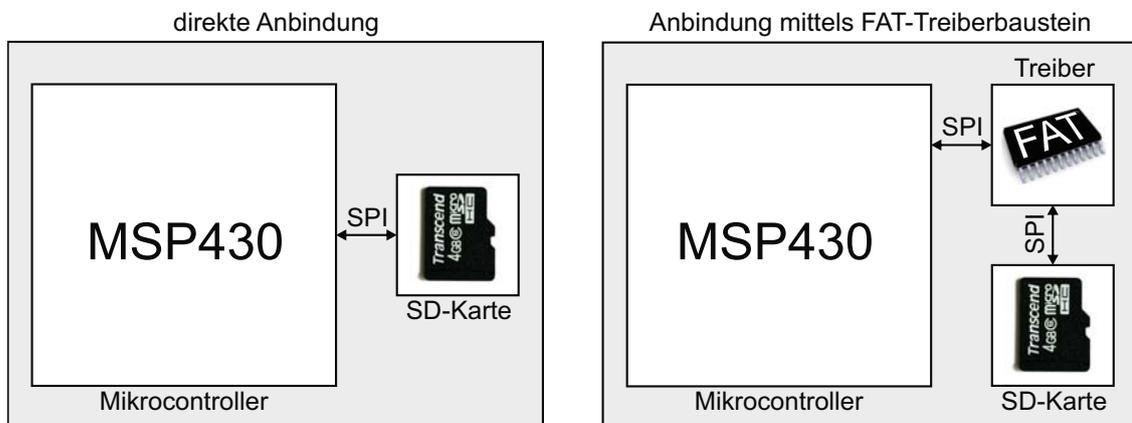


Abbildung 4.34: Anbindungsmöglichkeiten von SD-Karten

Schnittstelle. Beide Möglichkeiten der Anbindung von SD-Karten sind in Abbildung 4.34 dargestellt.

Im Rahmen dieses Projekts wurde zuerst die Anbindung von SD-Karten mittels eines FAT-Treiber-ICs untersucht, wobei diese später zu Gunsten der direkten Anbindung aufgegeben wurde. Ausschlaggebend hierfür war die Einstellung der Produktion des verwendeten ICs. Die Verwaltung des Dateisystems erfolgt in der aktuellen Ausführung des Moduls durch einen kommerziellen FAT32-Firmwaretreiber. Die mit beiden Varianten erzielbaren Datentransferraten beim Schreiben sowie die wesentlichen Einflussfaktoren auf die erzielbare Performance sind in den Abschnitten 4.3.1.1 und 4.3.1.2 näher beschrieben.

Für den Datenspeicher wurde der MSP430-Mikrocontroller mit der höchsten zulässigen Taktfrequenz (18 MHz) sowie dem größten RAM (16 kB) gewählt. Dies war zum Ent-

wicklungszeitpunkt des Moduls der neu verfügbare MSP430F5438, wobei dieser – wie sich später herausstellte – einige gravierende Fehler im Embedded Emulation Module aufwies. Diese wurden jedoch erst nach Fertigstellung der Module im Erratasheet [92] veröffentlicht. Eine effiziente Entwicklung der Software unter Verwendung der Debug-Funktionalität war mit dem MSP430F5438 nicht möglich, da beispielsweise im Beisein von häufig auftretenden Interrupts die Breakpoints nicht eingehalten wurden. Abhilfe schaffte hier der MSP430F5438A, der etwas später in Einzelstücken als Samples erhältlich war und über ein voll funktionsfähiges Embedded Emulation Module verfügt. Da das Nachfolgemodell pincompatibel zum MSP430F5438 ist, wurde in weiterer Folge für die Entwicklung der Firmware ein Datenspeichermodul mit dem MSP430F5438A ausgestattet. Die mittels dieses Moduls entwickelte Firmware ist mit geringen Modifikationen auf beiden Prozessoren lauffähig. Der Geschwindigkeitsvorteil des MSP430F5438A, der mit 25 MHz getaktet werden kann, wurde aus Kompatibilitätsgründen jedoch nicht ausgenutzt. Dies ist mit geringem Aufwand im Zuge einer weiteren Entwicklung des eigentlichen Messsystems möglich und empfehlenswert.

Ein Blockschaltbild des Datenspeichermoduls in der derzeit aktuellen Hardwareausführung mit direkter Ansteuerung der Speicherkarte durch den Mikrocontroller ist in Abbildung 4.35 dargestellt. Neben dem Mikrocontroller sind auf dem Modul ein Uhrenquarz für den Betrieb der Echtzeituhr des Mikrocontrollers sowie ein temperaturkompensierter 10 MHz-Quarz zur Erzeugung der Zeitbasis des ADCs vorhanden (Details siehe Abschnitt 4.3.1.3). Weiters sind vier GPIOs auf einer Stiftleiste herausgeführt worden und es sind zwei LEDs zur Anzeige des Modulzustandes sowie Steckverbinder für die Modulankoppe-

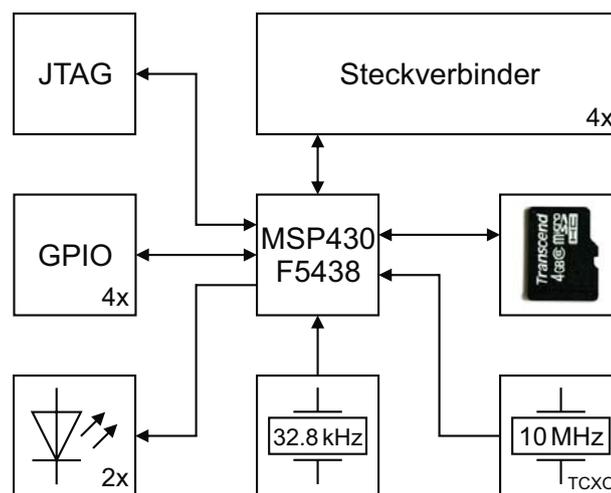


Abbildung 4.35: Blockschaltbild des Datenspeichermoduls

lung (Basismodul und allgemeine Signalverarbeitung) und ein JTAG-Anschluss vorhanden. Die in Abschnitt 4.3.1.3 beschriebene Synchronisierung der Module erfolgt über einen der herausgeführten GPIOs.

4.3.1.1 FAT32-Treiber-IC – DOSonChip

Zum Zeitpunkt des Beginns der Entwicklung des Datenspeichermoduls waren zwei Treiber-ICs zur Ansteuerung von SD-Karten am Markt verfügbar. Die wesentlichen Eckdaten beider Bausteine sind in Tabelle 4.9 zusammengefasst. Bei acht Kanälen mit einer Abtastrate von jeweils 1 kS/s pro Kanal und einer Auflösung von 12 Bit ist unter Vernachlässigung des Dateioverheads (Zeitmarken, etc.) eine minimale Schreibrate von 16 kB/s erforderlich. Entsprechend Tabelle 4.9 verfügen beide ICs über mittlere Schreibraten, die diesen Wert deutlich überschreiten. Da das ursprüngliche und damals noch aktuelle Konzept eine durchgehende Versorgung des Systems mittels Akkumulator vorsah wurde der DOSonChip-IC auf Grund seines deutlich geringeren Strombedarfs gewählt.

Baustein	Schreibrate in kB/s	Strombedarf in mA
DOSonChip	30 ¹	5
uALFAT	60	38

¹ Keine Herstellerangabe verfügbar – Benutzerangabe im SparkFun-Forum [84]

Tabelle 4.9: Treiberbausteine für SD- bzw. SDHC-Karten

4.3.1.1.1 DOSonChip Performance

Die unter Verwendung des DOSonChip-Treibers mit dem MSP430F5438 erzielten mittleren Schreibgeschwindigkeiten sind in Tabelle 4.10 zusammengefasst. Bei den Messungen wurden jeweils Datenblöcke einer bestimmten Größe unmittelbar aufeinander folgend an den IC übergeben und die Zeit, die für die Übertragung von 10 MB benötigt wurde, mittels eines Timers bestimmt. Die angegebenen Schreibraten sind Mittelwerte, die aus zehn Einzelmessungen (siehe Tabelle A.4 im Anhang) berechnet wurden. Als Datenträger diente eine 4 GB, Klasse 6, microSDHC-Karte von Transcend. Die Größe der Zuordnungseinheiten betrug 32 kB. Die Versuche haben gezeigt, dass die Datenübertragungsraten mit der Größe der Datenblöcke zunehmen, wobei die Größe dieser Blöcke durch die Größe des im Mikrocontroller zur Verfügung stehenden Arbeitsspeichers (16 kB) begrenzt ist.

Blockgröße in B	Schreibrate in kB/s
512	61.8
1024	83.0
2048	99.1
4096	108.7
6144	111.2

Tabelle 4.10: Mittlere Schreibgeschwindigkeit des DOSonChip-ICs in Abhängigkeit von der übergebenen Blockgröße (Transcend 4 GB microSDHC-Karte, FAT32 mit 32 kB großen Zuordnungseinheiten)

Hinsichtlich der eigentlichen Anwendung als Datenspeicher im Rahmen eines Messsystems hat sich gezeigt, dass im vorliegenden Fall die mittlere Schreibrate keine zuverlässige Kenngröße darstellt, die eine Prognose bezüglich einer zu erwartenden lückenlosen Aufzeichnung einer gegebenen und in Echtzeit anfallenden Datenmenge erlaubt. Ursache hierfür ist der in Relation zur pro Sekunde anfallenden Datenmenge (16 kB) geringe Arbeitsspeicher (16 kB) des MSP430F5438. Da die Verarbeitungszeit einzelner, an den Treiber-IC übergebener, Datenblöcke starken Schwankungen unterliegt, muss im Mikrocontroller ein ausreichend groß dimensionierter Puffer vorgesehen werden um einen Datenverlust zu vermeiden. Je geringer die Differenz zwischen der erreichbaren mittleren Schreibrate und der in Echtzeit pro Sekunde anfallende Datenmenge ausfällt, desto mehr RAM muss im Mikrocontroller zur Zwischenspeicherung von Daten zur Verfügung stehen, um eine lückenlose Aufzeichnung sicherzustellen.

Nachdem der Umfang, der pro Sekunde anfallenden Sensordaten etwa 15 % der maximalen erreichten mittleren Schreibrate beträgt, ist nicht davon auszugehen, dass im konkreten Fall mit einem Verlust von Messdaten zu rechnen ist.

Für die Aufzeichnung der Messdaten wurde entsprechend den bisherigen Erkenntnissen ein mit 12.8 kB relativ großer Bereich MSP430F5438-RAMs in Form eines Ringpuffers zur Zwischenspeicherung der Messdaten vorgesehen. Die im Ringpuffer enthaltenen Daten werden in 2 kB großen Blöcken entsprechend Abbildung 4.36 an den Treiber-IC übergeben und schließlich durch diesen auf der Speicherkarte abgelegt.

Auf Grund der hohen Komplexität des aus dem MSP430, dem Treiber-IC, sowie dem Controller der Speicherkarte bestehenden Systems und der Tatsache, dass lediglich die Firmware des MSP430 – und damit auch deren zeitliches Verhalten – bekannt ist, kann eine Bestimmung der in Echtzeit erzielbaren Schreibgeschwindigkeit nur durch Messungen erfolgen.

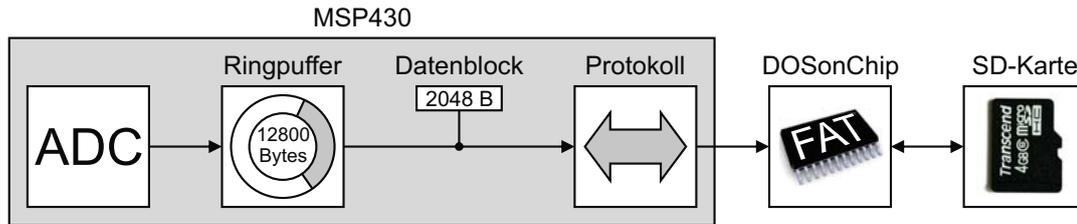


Abbildung 4.36: Aufzeichnung von Messdaten mit dem DOSonChip-Treiber-IC

Hierfür wurden bekannte Daten entsprechend Abbildung 4.36 an den Treiber-IC übergeben. Die Generierung der zu speichernden Daten erfolgte hierbei jedoch abweichend von Abbildung 4.36 zeitgesteuert durch eine Zählvariable. Zur Bestimmung der maximalen, ohne Pufferüberlauf, erzielbaren Aufzeichnungsrate wurde die Datenrate in 5 kB/s-Schritten variiert und jeweils der maximale Füllstand des Ringpuffers festgestellt. Jede Einzelmessung dauerte zehn Minuten, wobei bei jeder Datenrate zehn unmittelbar aufeinander folgende Einzelmessungen durchgeführt wurden und jeweils der maximale Füllstand der zehn Einzelmessungen als Endergebnis herangezogen wurde. Die Ergebnisse der Einzelmessungen sind im Anhang in den Tabellen A.5 bis A.7 dokumentiert.

Die, in dieser Form ermittelten, maximalen Echtzeit-Schreibraten von drei microSDHC-Karten verschiedener Hersteller sind in Tabelle 4.11 wiedergegeben. Die Ergebnisse zeigen, dass die tatsächlich erzielbare maximale Schreibrate in erheblichem Maße von der eingesetzten Karte abhängig ist, wobei die angegebene Geschwindigkeitsklasse, zumindest im SPI-Modus, keinen Schluss auf die erzielte Performance zulässt.

Hersteller	Kapazität in GB	Klasse	Schreibrate in kB/s
Kingston	4	4	45
SanDisk	4	4	15
Transcend	4	6	35

Tabelle 4.11: Echtzeit-Schreibrate des DOSonChip-Treibers mit verschiedenen microSDHC-Karten (FAT32 mit 32 kB großen Zuordnungseinheiten)

4.3.1.2 FAT32-Firmwaretreiber

Nach Bekanntwerden der Tatsache, dass der DOSonChip in naher Zukunft nicht mehr erhältlich sein würde, musste eine alternative Lösung gefunden werden. Um in weiterer Folge eine analoge Situation mit ähnlichen Treiber-ICs zu vermeiden, wurde beschlossen eine reine Firmwarelösung zu wählen. Neben der Unabhängigkeit vom Hersteller besteht

bei dieser Lösung im Anfall die Möglichkeit den Treiber, entsprechend den konkreten Bedürfnissen, zu modifizieren. Darüber hinaus ist mit dieser Maßnahme eine Reduktion der Kosten und der Anzahl der Komponenten verbunden.

Die Ansteuerung und Verwaltung der SD-Karten erfolgt durch den FAT32-Firmwaretreiber der Firma Embedded-Code (<http://www.embedded-code.com>). Es handelt sich hierbei um einen kommerziellen Treiber, der den Zugriff auf FAT32-formatierte SDHC-Karten ermöglicht.

4.3.1.2.1 Performance

Das Schreiben der Messdaten erfolgt erneut mittels eines Ringpuffers, der die Daten in 512 B-Blöcken an den Treiber übergibt (siehe Abbildung 4.37). Die Bestimmung der Echtzeit-Performance des Treibers mit verschiedenen Karten erfolgte analog zu Abschnitt 4.3.1.1.1. Die erzielten Schreibgeschwindigkeiten sind in Tabelle 4.12 zusammengefasst.

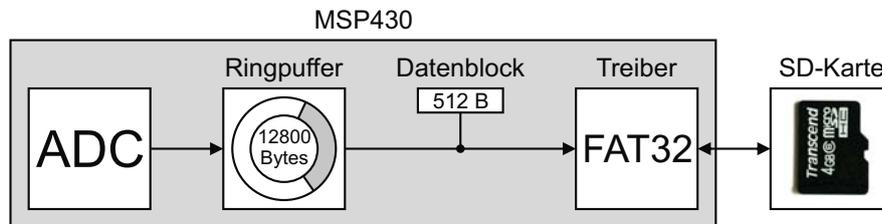


Abbildung 4.37: Aufzeichnung von Messdaten mit dem FAT32-Firmwaretreiber

Hersteller	Kapazität in GB	Klasse	Schreibrate in kB/s
Kingston	4	4	k.A. ¹
SanDisk	4	4	20
Transcend	4	6	30

¹ Keine Angabe – Karte nicht kompatibel mit FAT32-Treiber

Tabelle 4.12: Echtzeit-Schreibrate des FAT32-Firmwaretreibers mit verschiedenen microSDHC-Karten (FAT32 mit 32 kB großen Zuordnungseinheiten)

Beim Schreiben auf die microSDHC-Karte der Firma Kingston traten Probleme bei der Verwaltung des Inhaltsverzeichnisses auf, weshalb an dieser Stelle kein gültiger Wert bestimmt werden konnte. Die Ursache für die beobachtete Inkompatibilität wurde nicht näher bestimmt. Die beiden übrigen Karten konnten problemlos beschrieben werden, wobei die Karte von SanDisk einen um 5 kB/s höheren und die Karte von Transcend einen

um den selben Betrag niedrigeren Wert als bei den Messungen mit dem DOSonChip-Treiber-IC (vergleiche Tabelle 4.11) erzielte.

Für die konkrete Anwendung im Messsystem bietet die Transcend-Karte mit einer Echtzeit-Schreibgeschwindigkeit von 30 kB/s die größte Reserve und wurde daher für alle weiteren Tests des Laborsystems eingesetzt. Die Zwischenspeicherung mittels Ringpuffer wurde in weiterer Folge entsprechend Abbildung 4.37 beibehalten.

4.3.1.2.2 Prototyp

Abbildung 4.38 zeigt den Prototypen des Datenspeichermoduls. Es handelt sich hierbei um den letzten, an die Gegebenheiten des Laborsystems angepassten, Hardwareentwurf mit dem DOSonChip-Treiber, wobei dieser Bauteil mittels Lackdrähten überbrückt wurde um die direkte Anbindung der Speicherkarte an den Mikrocontroller zu erreichen. Der Schaltungsaufbau entspricht somit Abbildung 4.35.



Abbildung 4.38: Prototyp des Datenspeichers

4.3.1.3 Zeitbasis

Die Zeitbasis eines Mikrocontrollers wird in der Regel von Quarzen oder Oszillatoren abgeleitet. Abhängig von den konkreten Anforderungen hinsichtlich der Genauigkeit der Zeitbasis können hierfür entweder interne Taktgeneratoren oder externe Bausteine herangezogen werden. Bei hohen Anforderungen ist die Verwendung externer Oszillatoren in der Regel unumgänglich.

Da im konkreten Anwendungsfall Langzeitmessungen über eine Dauer von 20 bis 30 h durchgeführt werden sollen, wobei eine zeitliche Zuordnung der einzelnen Samples auch

zwischen unterschiedlichen Modulen gewährleistet sein soll, ist eine genaue Zeitbasis zwingend erforderlich. Insbesondere, da auf Grund der Ergebnisse in Abschnitt 4.2.2.3.2 eine fortlaufende Synchronisierung der Zeitbasen mit ausreichender Genauigkeit durch einen Austausch von ZigBee-Funknachrichten in der derzeitigen Ausführung des Systems nicht praktikabel ist. Eine mögliche Abhilfe ist in Abschnitt 5.3 ausgeführt.

Oszillatoren beruhen in der Regel auf Schwingquarzen deren Eigenfrequenz jedoch von den Umgebungsbedingungen (Temperatur, Luftfeuchtigkeit und Luftdruck) mitbestimmt wird [2]. Die Frequenzstabilität wird in der Regel in parts per million (ppm) angegeben, wobei 1 ppm $1.0 \cdot 10^{-4} \%$ entspricht. Konkret dürfte bei einer angenommenen Messdauer von 20 h und einem maximalen Fehler der Zeitbasis von 1 ms beim Ablauf der Aufzeichnungsdauer ein maximaler Fehler von 0.014 ppm nicht überschritten werden. Bei einer geplanten Abtastrate von einem 1 kS/s würde der angenommene Wert von 1 ms exakt einem Abtastintervall entsprechen und somit in einer guten zeitlichen Korrelation der Abtastwerte im Gesamtsystem resultieren. Die hier angenommenen Werte sind jedoch – wie sich noch zeigen wird – gleichbedeutend mit sehr hohen Anforderungen an die Stabilität der Zeitbasis.

In Tabelle 4.13 sind die wesentlichen Kennwerte dreier quarzbasierter Oszillatoren zusammengefasst. Es handelt sich hierbei um einen beheizten Quarzoszillator (OCXO), der die höchste Frequenzstabilität aufweist, einen temperaturkompensierten Quarzoszillator (TCXO) und einen gewöhnlichen Quarzoszillator (XO). Die hohe Genauigkeit eines beheizten Quarzoszillators ist jedoch mit einem um nahezu zwei Größenordnungen höheren Energiebedarf verbunden, der daraus resultiert, dass der Quarz mittels einer Heizung auf einem konstanten, über der Umgebungstemperatur liegenden, Wert gehalten wird. Weiters ist der erhöhte Platzbedarf und der wesentlich höhere Preis des OCXO zu beachten. Für den Einsatz im Rahmen eines batteriebetriebenen Messsystems ist der Einsatz

Oszillator	Typ	Stabilität in ppm		LxBxH	Leistung	Preis
		Temp.	Alterung ¹	in mm	in mW	in €
Abracon – AOCJY	OCXO	$\pm 0.03^2$	± 0.1	25x22x13	1400	142
IQD – CFPT-126	TCXO	$\pm 0.5^3$	± 1.0	7x5x2	10	21
Euroquarz – XO91	XO	$\pm 25^3$	± 5.0	7x5x2	50	5

¹ im ersten Jahr

² im Bereich von -40 bis $+75$ °C

³ im Bereich von -40 bis $+85$ °C

Tabelle 4.13: Vergleich verschiedener quarzbasierter Oszillatortypen

von beheizten Quarzoszillatoren auf Grund ihrer hohen elektrischen Verlustleistung nicht praktikabel. Für das Laborsystem wurde daher der Baustein CFPT126 von IQD gewählt. Bei einer ausschließlichen Versorgung des Messsystems durch das Bordnetz des Fahrzeuges wäre der Einsatz eines OCXOs jedoch eine sinnvolle Option.

Die Übersicht in Tabelle 4.13 zeigt, dass speziell unter Berücksichtigung der Alterung Langzeitmessungen im Bereich von 20 h mit hohen Anforderungen an die Synchronität der Zeitbasen nur mit periodischer Synchronisierung im Betrieb bzw. im Fall der beheizten Oszillatoren zumindest mit einem gleichzeitigen Start der Zeitbasen vor der Messung durchgeführt werden können. Messungen, die hinsichtlich der zeitlichen Korrelation der Messwerte kritisch sind, wie z.B. die Berechnung von Übertragungsfunktionen, müssen daher ohne diese Maßnahmen auf kurze Messdauern beschränkt werden. Die hierbei maximal zulässigen Messintervalle sind abhängig von der Stabilität des verwendeten Oszillators und den Anforderungen an zeitliche Zuordenbarkeit der einzelnen Messwerte.

Abbildung 4.39 zeigt anhand der Zeitbasen A und B einerseits die Synchronisierung der Zeitbasen sowie im Fall der Zeitbasis B ebenfalls die Korrektur der Geschwindigkeit. Beide Zeitbasen starten gleichzeitig bei $t = 0\text{s}$ und laufen mit unterschiedlichen Geschwindigkeiten α_A und α_B . Für den jeweiligen Wert gilt ohne Eingriffe von außen

$$\hat{t}_A(t) = \alpha_A t \quad (4.10)$$

$$\hat{t}_B(t) = \alpha_B t. \quad (4.11)$$

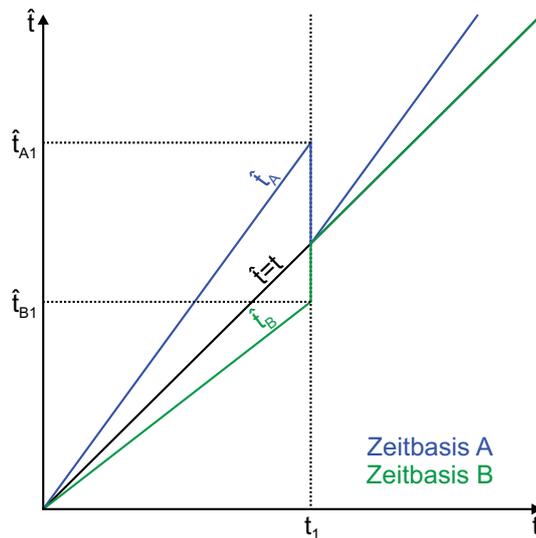


Abbildung 4.39: Möglichkeiten zur Synchronisierung von Zeitbasen

Zum Zeitpunkt der Synchronisierung bei $t = t_I$ weisen die Zeitbasen die Werte \hat{t}_{A1} bzw. \hat{t}_{B1} auf. Der Wert der Zeitbasis A mit \hat{t}_A wird nun gleich t_I gesetzt und taktet mit gleichbleibender Geschwindigkeit weiter, wodurch erneut eine Abweichung entsteht. Die Kenntnis des Wertes der Zeitbasis zum Zeitpunkt der Synchronisierung erlaubt jedoch ebenfalls eine Korrektur der Geschwindigkeit, wie am Beispiel der Zeitbasis B gezeigt. Mittels der bekannten Werte \hat{t}_{B1} und t_I kann die Geschwindigkeit mittels

$$\hat{\alpha}_A = \frac{\hat{t}_{B1}}{t_I} \quad (4.12)$$

geschätzt und damit eine Anpassung der Geschwindigkeit vorgenommen werden. Man erhält in diesem Fall für den Wert der Zeitbasis B

$$\hat{t}_B(t) = \begin{cases} \alpha_B t & \text{für } t < t_I \\ t_I + \hat{\alpha}_B^{-1} \alpha_B (t - t_I) & \text{sonst,} \end{cases} \quad (4.13)$$

wobei im Idealfall $\hat{\alpha}_B^{-1} \alpha_B = 1$ gilt und ab dem Zeitpunkt t_I die Zeitbasis B den exakten Wert t widerspiegelt. In der Praxis ist eine exakte Übereinstimmung jedoch auf Grund von Messunsicherheiten und den unvermeidlichen Einflüssen der Umgebung auf den Oszillator selbst nicht zu erreichen.

Mit dem realisierten Laborsystem ist aus bereits erwähnten Gründen eine fortlaufende Synchronisierung der Zeitbasen per ZigBee-Nachrichten nicht durchführbar. Alternative Quellen für Synchronisierungssignale stellen beispielsweise GPS- (Weltweit) oder DCF77-Empfänger (Westeuropa) dar. Ein ständiger Empfang der jeweiligen Funksignale in ausreichender Stärke und Qualität ist jedoch nicht gewährleistet (Funkabschattung auf Grund der Einbausituation des Moduls, Tunneldurchfahrt, etc.), weshalb diese Empfänger nicht als primäre Lösung für eine fortlaufende Synchronisierung herangezogen werden können.

Für den Datenspeicher des Laborsystems wurde der temperaturkompensierte Oszillator CFPT-126, der Firma IQD, (siehe Tabelle 4.13) mit einer Taktfrequenz von 10 MHz als Basis gewählt. Ausschlaggebend hierfür war der geringere Strombedarf sowie die geringe Baugröße des Bauteils. Für die Ableitung der Zeitbasis mit einer kleinsten Einheit von 1 ms werden nominell 2500 steigende Flanken eines vom ursprünglichen Ausgangssignal des Oszillators abgeleiteten Taktsignals mit einer Frequenz von 2.5 MHz herangezogen.

Um die Gangunterschiede der einzelnen Oszillatoren zu bestimmen wurden Vergleichsmessungen durchgeführt bei denen die Zeitbasen der Datenspeichermodule 1 bis 4 über ein Start- und Stoppsignal gleichzeitig aktiviert und deaktiviert wurden. Die Messungen erfolgten über einen Zeitraum von 4 h und wurden jeweils vier Mal wiederholt, wobei das

Aktivierungsintervalls durch das Modul 0 gesteuert wurde. Die gemittelten Endstände der einzelnen Zeitbasen inklusive der Standardabweichung sind in Tabelle 4.14 zusammengefasst. Die Ergebnisse der Einzelmessungen sind in Tabelle A.10 im Anhang dokumentiert.

Modul	\bar{t}_E				$\sigma_{t,E}$
	min	s	ms	μ s	μ s
0	240	0	0	6.40	0.00
1	239	59	996	219.20	49.26
2	239	59	994	547.36	309.26
3	239	59	993	604.80	57.02
4	239	59	993	155.68	94.93

Tabelle 4.14: Mittlere Endwerte t_E der Zeitbasen der Module 0 bis 4 nach 4 h Laufzeit

Mit den Werten aus Tabelle 4.14 wurde in weiterer Folge eine Geschwindigkeitsanpassung der einzelnen Zeitbasen durchgeführt, wobei die Zeitbasis des Moduls 0 als Referenz herangezogen wurde. Es handelt sich hierbei also nicht um eine Kalibrierung und Justage der Zeitbasen bezüglich einer hochgenauen Referenz sondern lediglich um eine Minimierung der Gangunterschiede zwischen den einzelnen Modulen. Eine Kalibrierung und Justage der Zeitbasen könnte in weiterer Folge jedoch mit Hilfe des Zeitsignals eines GPS-Empfängers wie z.B. dem LEA-6T Modul (Genauigkeit laut Homepage 15 ns) der Firma ublox (<http://www.u-blox.com>) oder unter Verwendung eines DC77-Empfängers durchgeführt werden. Im Rahmen des Laborsystems wurde dies auf Grund des damit verbundenen Aufwandes nicht realisiert.

Die Anpassung der Geschwindigkeit der Zeitbasen erfolgte ähnlich der Verwendung von Schalttagen im Gregorianischen Kalender durch eine Reduktion oder Erhöhung des nominellen Timerendstandes n_E um Eins für jeweils einen Timerdurchlauf in fixen zeitlichen Abständen. Ob eine Reduktion oder Erhöhung vorgenommen werden muss, ist davon abhängig, ob die Zeitbasis eine höhere oder niedrigere Geschwindigkeit als die Referenz aufweist. Die Anzahl der nominellen Timerdurchläufe n_K zwischen diesen Schaltdurchläufen ist wiederum vom absoluten Wert der Abweichung abhängig. Diese Vorgehensweise ist anhand des Timer-Zählerstandes n_Z in Abbildung 4.40 dargestellt.

Der Wert n_K wurde mittels

$$n_K = \left\lfloor \frac{n_F}{|\Delta n_F| n_E} + \frac{1}{2} \right\rfloor - 1 \quad (4.14)$$

bestimmt, wobei n_F für die Anzahl der am betrachteten Modul beobachteten Flanken des

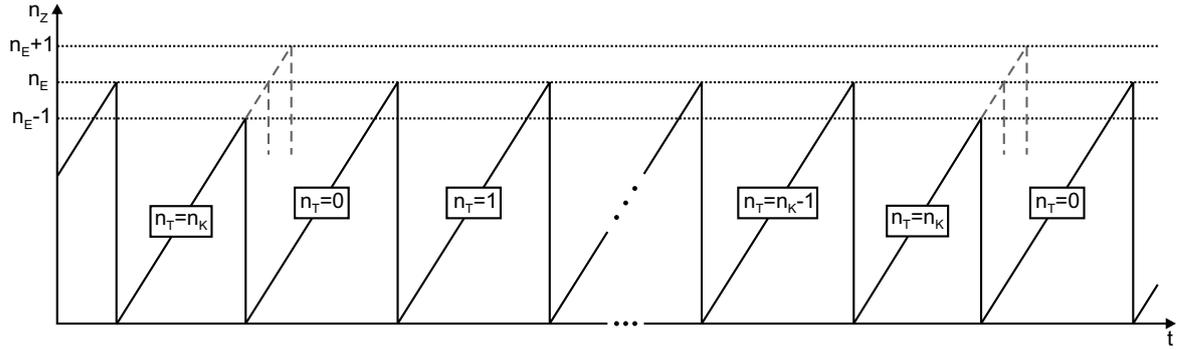


Abbildung 4.40: Geschwindigkeitskorrektur der Zeitbasis

Timertaktes und

$$\Delta n_F = n_F - n_{F,ref} \quad (4.15)$$

für die Abweichung in Relation zum entsprechenden Wert der Referenzzeitbasis $n_{F,ref}$ steht.

Mit den Werten aus Tabelle 4.14 erhält man mit (4.14) die in Tabelle 4.15 angegebenen Korrekturwerte für die Geschwindigkeitsanpassung der Zeitbasen. Die Zählerendstände der Module 1 bis 4 sind entsprechend den Ergebnissen jeweils alle $n_K + 1$ Millisekunden um den Wert Δn_E korrigiert worden. Da die Zeitbasen der Module jeweils langsamer als die Referenz laufen, müssen die Nominalwerte nach unten korrigiert werden um einen schnelleren Lauf der Zeitbasen zu erzielen.

Modul	n_F	Δn_F	n_K	Δn_E
0	36 000 000 016	0	–	–
1	35 999 990 548	–9468	1520	–1
2	35 999 986 368	–13 648	1054	–1
3	35 999 984 012	–16 004	899	–1
4	35 999 982 889	–17 127	840	–1

Tabelle 4.15: Korrekturwerte zur Geschwindigkeitsanpassung der Zeitbasen

Eine erneute Bestimmung der Endwerte der Zeitbasen nach 4 h im Vergleich zur Referenz ergab nach fünf Messungen die in Tabelle 4.16 zusammengefassten Mittelwerte und Standardabweichungen. Die Resultate der Einzelmessungen sind wiederum im Anhang (siehe Tabelle A.11) dokumentiert. Die Auswertung zeigt eine deutliche Verringerung des Gangunterschiedes, wobei die Abweichung maximal 0.015 ppm beträgt.

Bemerkenswert ist die Tatsache, dass die erhaltenen Zeitstände trotz durchgeführter Korrektur und einer Auf- bzw. Abrundung der Werte n_K auf ganze Zahlen erneut unter

Modul	\bar{t}_E				$\sigma_{t,E}$
	min	s	ms	μ s	μ s
1	240	59	0	6.40	0.00
1	239	59	999	996.72	31.78
2	239	59	999	793.52	61.23
3	239	59	999	990.16	14.68
4	239	59	999	943.28	36.93

Tabelle 4.16: Werte der Zeitbasen t_E der Module 0 bis 4 nach 4 h Laufzeit mit Geschwindigkeitsanpassung

denen der Referenz liegen. Bei einer vollständigen Korrektur wären hier Abweichungen in beide Richtungen zu erwarten.

Präzise Messungen im Bereich von Mikrosekunden setzen eine hochgenaue und äußerst stabile Referenz voraus, die im Rahmen dieser Messungen nicht vorlag. Hier diente lediglich ein gleichwertiges Modul als Referenz über dessen Verhalten und Genauigkeit keine ausreichenden Informationen zur Verfügung stehen. Die Ergebnisse sind somit lediglich bezüglich der Abweichungen der Zeitbasen untereinander aussagekräftig und können nicht zur Charakterisierung der Oszillatoren bezüglich Stabilität, Alterungsverhalten, etc. herangezogen werden. Speziell die Alterung sowie die Stabilität in Abhängigkeit von den Umgebungsbedingungen sind jedoch bei einem Einsatz des Systems ohne kontinuierliche Synchronisation der Zeitbasen per Funk wesentliche Kriterien zur Abschätzung der kurz- und langfristig erzielbaren Genauigkeit hinsichtlich der zeitlichen Zuordenbarkeit der Abtastwerte im Gesamtsystem. Diesbezügliche Untersuchungen unter Verwendung einer hochgenauen zeitlichen Referenz sowie unter kontrollierten Umgebungsbedingungen (Klimakammer) wären daher je nach geplanter Ausführung des eigentlichen Messsystems empfehlenswert.

Im Rahmen des Laborsystems ist in weiterer Folge ein synchroner Start der Zeitbasen mittels eines kabelgebundenen Signals vorgesehen. Die hierfür benötigte Verbindung ist jedoch nur vor Beginn der Messung erforderlich und kann danach getrennt werden. Dies stellt einen synchronen Start der Zeitbasen vor Beginn der Messung sicher, bei hohen Anforderungen an die zeitliche Korrelation der einzelnen Abtastwerte ist die zulässige Messdauer jedoch begrenzt. Speziell, wenn die einzelnen Knoten unterschiedlichen Temperaturverläufen unterliegen.

4.3.1.4 Strombedarf der SD-Karte

Der Strombedarf des Gesamtmoduls setzt sich im Wesentlichen aus den Anteilen des Mikrocontrollers, des Oszillators und der Speicherkarte zusammen. Mit Ausnahme der Speicherkarte können die einzelnen Anteile mittels des jeweiligen Datenblattes in ausreichender Genauigkeit abgeschätzt werden. Im Datenblatt der Transcend microSDHC Class 6 Card Serie sind diesbezüglich jedoch lediglich über einen Zeitraum von 1 s gemittelte Werte angegeben, die keinen Rückschluss auf die tatsächlich auftretenden Spitzenströme zulassen. Es wurde daher der Strombedarf der verwendeten Karte mittels Messung bestimmt. Abbildung 4.41 zeigt das Ergebnis dieser Messung bei einer Schreibgeschwindigkeit von 90 kB/s. Es treten Spitzenströme bis etwa 200 mA auf, wobei der mittlere Strom jedoch lediglich bei 22 mA liegt. Dies ist bei der Auslegung der Energieversorgung des Messsystems entsprechend zu berücksichtigen.

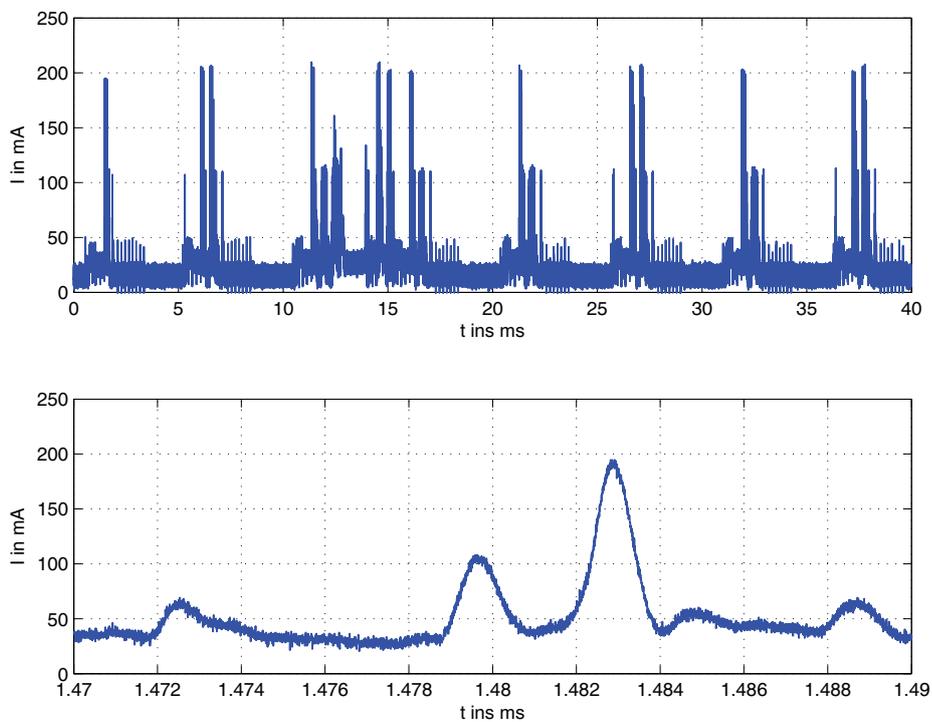


Abbildung 4.41: Strombedarf der Transcend 4 GB microSD-Karte bei einer mittleren Schreibgeschwindigkeit von 90 kB/s

4.3.2 Beschreibung der Firmware

Die Firmware des Datenspeichers ist in Form eines Zustandsautomaten (state machine) entwickelt worden, wodurch eine einfache Erweiterung möglich ist. Die folgenden Abschnitte enthalten eine kurze Beschreibung des implementierten Zustandsautomaten sowie die Dokumentation des verwendeten Dateiformats.

4.3.2.1 Endlicher Zustandsautomat

Abbildung 4.42 zeigt das vereinfachte Zustandsdiagramm des Zustandsautomaten, wobei hier automatische Zustandsänderungen, durch Kommandos verursachte Zustandsänderungen und Änderungen die nur auf Grund von Fehlern auftreten, farblich kodiert dargestellt sind. Die einzelnen Zustände sind im Anhang B und die Kommandocodes in Anhang C näher beschrieben.

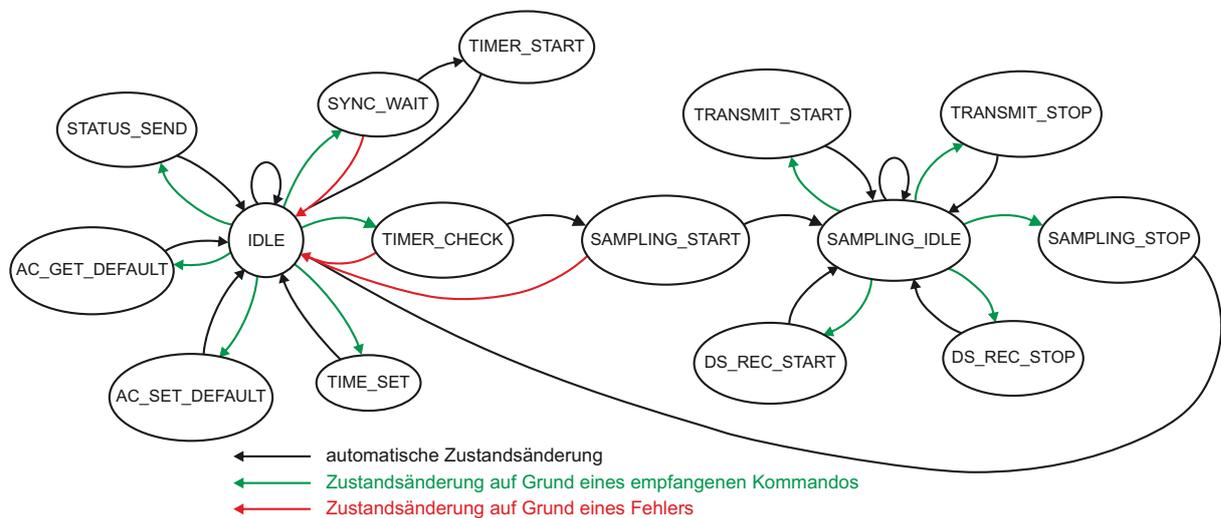


Abbildung 4.42: Zustandsautomat des Datenspeichermoduls

4.3.2.2 Verarbeitung der Abtastwerte

Bei der Erstellung der Firmware stellte sich die Verarbeitung der Abtastwerte als kritisch heraus, da hier die Daten sowohl auf der SD-Karte abgelegt als auch gleichzeitig die Abtastwerte eines Messkanals (Funkübertragung) an das angeschlossene Basismodul übergeben werden müssen. Darüber hinaus muss das Datenspeichermodul sicher auf Kommandos reagieren, die vom Basismodul weitergeleitet werden.

Ziel der Entwicklung war es, die Interrupt-Service-Routinen so kurz wie möglich zu halten um die Latenzzeiten bei Interrupts zu minimieren. Abbildung 4.43 zeigt schematisch die prinzipiellen Vorgänge, die bei der Verarbeitung der Abtastwerte ablaufen. Die Abtastwerte werden in der ADC-ISR einerseits im ADC-Puffer und andererseits im C2T-Puffer abgelegt, sofern die Übertragung eines Messkanals über die Funkschnittstelle aktiviert ist. Der ADC-Puffer ist als Ringpuffer ausgelegt und übergibt die enthaltenen Daten jeweils in 512 Bit-Blöcken an den FAT-Treiber, der in der Schleife des Hauptprogramms aufgerufen wird. Die per Funk zu übertragenden Abtastwerte eines Kanals werden in zwei linearen Arrays, dem C2T-Puffer abgelegt. Sobald eines der beiden Arrays vollständig gefüllt ist, werden Header und Stopbytes hinzugefügt und die Daten an den TX-Puffer der RS232-Schnittstelle übergeben. Von dort werden die Daten Byte für Byte, gesteuert durch den Interrupt der RS232-Schnittstelle, an das Basismodul übergeben.

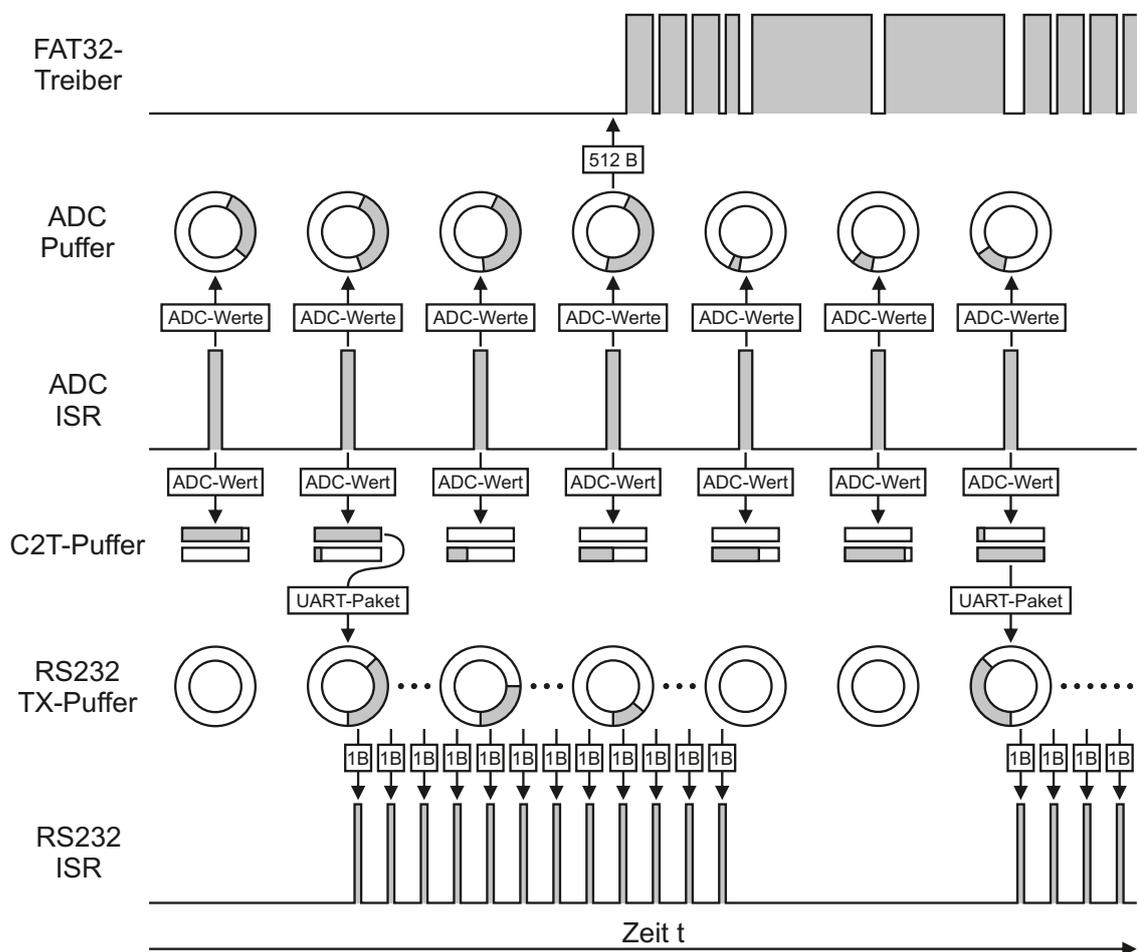


Abbildung 4.43: Verarbeitung der Abtastwerte

Die Ausführungsdauer sowie die Priorität der beteiligten Interrupts ist in Tabelle 4.17 zusammengestellt. Die Laufzeit der ADC-ISR ist hierbei davon abhängig, ob Daten lediglich aufgezeichnet oder gleichzeitig ebenfalls an das Basismodul übertragen werden müssen. Die Aufbereitung der zu übertragenden Datenpakete und der Transfer in den Sendepuffer der RS232-Schnittstelle sind in diesem Fall für die längere Ausführungszeit verantwortlich.

Interrupt	t_A in μs	t_I in μs	P	Bemerkung
Zeitbasis (TB0)	8.0	1000	59	
ADC	95.0	1000	55	Aufzeichnung auf SD-Karte
	214.0	1000	55	Aufzeichnung und Funkübertragung
RS232 (UCA1)	8.0	260	46	Empfang eines Bytes
	7.8	260	46	Absenden eines Bytes

Tabelle 4.17: Maximale Ausführungsdauer t_A , Zeitintervall t_I zwischen den Aufrufen und Priorität P der Interrupt-Service-Routinen des Datenspeichers

Interrupts mit höherer Priorität haben Vorrang vor Interrupts mit niedriger Priorität, wobei laufende ISR jedoch nicht zu Gunsten eines neu aufgetretenen Interrupts mit höherer Priorität unterbrochen werden. Von den eingesetzten Interrupts besitzt die Zeitbasis die höchste Priorität. Da zwischen dem Auftreten dieses Interrupts jeweils eine Millisekunde liegt und der Prozessor, falls eine andere ISR zum Zeitpunkt des Auftretens des TB0-Interrupts bereits ausgeführt wird, maximal $214\ \mu\text{s}$ belegt ist, kann die ISR in jedem Fall vor dem nächsten Auftreten des nächsten Interrupts ausgeführt werden. Ähnliches gilt für den ADC-Interrupt. Der Aufruf der zugehörigen ISR kann maximal $16\ \mu\text{s}$ verspätet erfolgen. Dies ist der Fall, wenn die RS232-ISR ($8\ \mu\text{s}$) bereits ausgeführt wird und die Timer-ISR ($8\ \mu\text{s}$) mit höherer Priorität ebenfalls noch zuvor abläuft. Wesentlich kritischer ist der Fall beim Empfang von Daten über die RS232-Schnittstelle, da alle $260\ \mu\text{s}$ ein neues Byte eingelesen werden muss. Die Ausführung der zugehörigen ISR darf daher maximal $260\ \mu\text{s}$ verzögert werden, ansonsten werden nicht ausgelesene Daten im Empfangsregister überschrieben und Daten gehen somit verloren. Entsprechend Tabelle 4.17 beträgt die maximale Latenz $229.8\ \mu\text{s}$ und liegt damit knapp unterhalb dieses Wertes. Die Latenz setzt sich dabei aus den Ausführungsdauern der RS232-ISR (Senden), ADC-ISR (Aufzeichnung und Funkübertragung) und der ISR der Zeitbasis zusammen.

Auf Grund der angeführten Ausführungszeiten kann davon ausgegangen werden, dass in keinem Fall Daten, sowohl ADC-Werte als auch per RS232 empfangene Daten, verloren gehen oder die Zeitbasis aussetzt. In der Praxis konnten erwartungsgemäß diesbezügliche

Probleme nicht beobachtet werden. Bei etwaigen Erweiterungen der Firmware – speziell der Interrupt-Service-Routinen – ist das Timing der ISR jedoch erneut kritisch zu prüfen.

4.3.2.3 Dateiformat

Neben den Abtastwerten werden zusätzlich Zeitmarken abgespeichert. Diese geben die, seit dem Start der Zeitbasis, verstrichene Zeit an. Hierdurch können Aufnahmelücken erkannt werden und die zeitliche Zuordnung der Abtastwerte ist in jedem Fall garantiert. Ein einzelnes Datenpaket besteht aus einer Zeitmarke (T-Bytes) gefolgt von zehn Abtastwerten pro aufgezeichnetem ADC-Kanal (D-Bytes). Abbildung 4.44 zeigt die prinzipielle Struktur des Datenformats. Die Zeitmarke besteht hierbei aus vier Bytes, in denen die Minuten (MIN) und Millisekunden (MS) seit dem Start der Zeitbasis entsprechend Abbildung 4.45 kodiert sind. Die Abtastwerte des 12-Bit-ADCs sind in Form von jeweils zwei Bytes (D-HBYTE und D-LBYTE) abgelegt (siehe Abbildung 4.46). Zeitmarken und Abtastwerte können anhand des höchstwertigen Bits (bei Zeitmarken gleich Eins) unterschieden werden.



Abbildung 4.44: Struktur der gespeicherten Daten

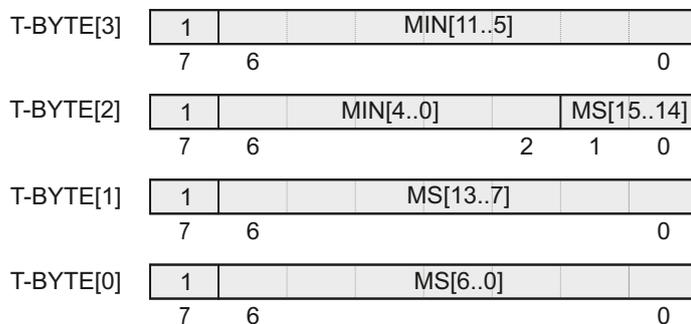


Abbildung 4.45: Aufbau der Zeitmarken

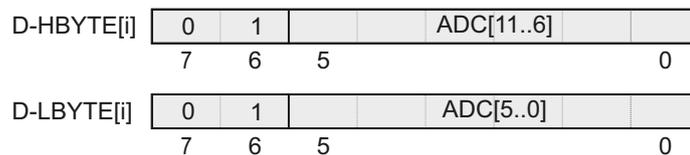


Abbildung 4.46: Kodierung der Abtastwerte

4.3.2.4 Kommunikationsbeispiel

Der Ablauf einer Messung und der Aufbau der am Laptop/PC abgesetzten bzw. empfangenen Pakete soll im Folgenden anhand eines Beispiels verdeutlicht werden. Das Beispiel geht von dem in Abbildung 4.47 dargestellten System aus. Zwei der dargestellten Basismodule (0x3389 und 0x77CE) verfügen über Datenspeicher, die in der Praxis Sensordaten aufzeichnen würden. In Abbildung 4.47 wurde jedoch auf die Darstellung der Sensorinterfaces und Sensoren verzichtet.

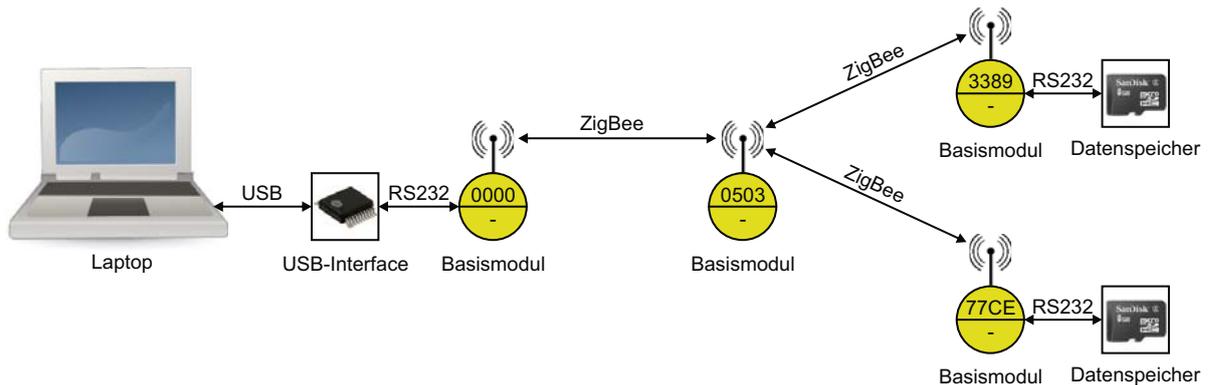


Abbildung 4.47: Kommunikationsbeispiel – Minimalsystem

Die dargestellten Datenpakete entsprechen den, vom Laptop an das USB-Interface des Basismodul 0x0000 übertragenen (TX), Befehlen bzw. den vom Basismodul an den Laptop übertragenen Antworten (RX). Sowohl Header als auch Stoppbytes sind in den Beispieldaten daher enthalten, wobei die Start- und Stoppbytes als STA bzw. STO1 und STO2 dargestellt sind. Die Befehle werden in der Regel als Broadcasts an den RS232-Cluster versendet, da hierdurch alle Knoten mit Datenspeicher mittels eines Datenpakets angesprochen werden können. Lediglich bei der Übertragung eines ADC-Kanals muss das Basismodul direkt adressiert werden. Da im System zwei Datenspeicher vorhanden sind, sind im Normalfall pro, per Broadcast, abgesetztem Befehl immer zwei Antwortpakete (eines pro Datenspeicher) zu erwarten.

Standardadresse festlegen (Broadcast)

TX:	STA	0x09	0x00	0xFF	0xFF	0x00	0x81	0x00	0x00	0x00	STO1	STO0
RX:	STA	0x08	0x1F	0x33	0x89	0x00	0xC1	0x81	0x00	STO1	STO0	
RX:	STA	0x08	0x1F	0x77	0xCE	0x00	0xC1	0x81	0x00	STO1	STO0	

Systemzeit einstellen (Broadcast) – Do 5.8.2010, 9:32 Uhr

TX:	STA	0x0E	0x00	0xFF	0xFF	0x01	0x84	0x00	0x20	0x09	0x04	0x05	0x08	0x07	0xDA	STO1	STO0	
RX:	STA	0x08	0x1F	0x33	0x89	0x01	0xC1	0x84	0x01	STO1	STO0							
RX:	STA	0x08	0x1F	0x77	0xCE	0x01	0xC1	0x84	0x01	STO1	STO0							

Zeitbasen synchronisieren (Broadcast)

TX:	STA	0x06	0x00	0xFF	0xFF	0x02	0x91	STO1	STO0									
RX:	STA	0x06	0x1F	0x33	0x89	0x02	0xD1	STO1	STO0									
RX:	STA	0x06	0x1F	0x77	0xCE	0x02	0xD1	STO1	STO0									

Kabelgebundener Synchronisierungsimpuls

RX:	STA	0x06	0x1F	0x33	0x89	0x03	0xD2	STO1	STO0									
RX:	STA	0x06	0x1F	0x77	0xCE	0x03	0xD2	STO1	STO0									

Abtastung starten (Broadcast)

TX:	STA	0x06	0x00	0xFF	0xFF	0x03	0xA1	STO1	STO0									
RX:	STA	0x06	0x1F	0x33	0x89	0x04	0xE1	STO1	STO0									
RX:	STA	0x06	0x1F	0x77	0xCE	0x04	0xE1	STO1	STO0									

Aufnahme starten (Broadcast)

TX:	STA	0x06	0x00	0xFF	0xFF	0x04	0xA3	STO1	STO0									
RX:	STA	0x06	0x1F	0x33	0x89	0x05	0xE4	STO1	STO0									
RX:	STA	0x06	0x1F	0x77	0xCE	0x05	0xE4	STO1	STO0									

Funkübertragung des Kanals 1 des Knotens 0x3389 starten

TX:	STA	0x07	0x00	0x33	0x89	0x05	0xA5	0x01	STO1	STO0									
RX:	STA	0x50	0x1F	0x33	0x89	0x06	0xE9	0x01	0x00	0x23	0x01	DBH	DBL	DBH	DBL	STO1	STO0
RX:	STA	0x50	0x1F	0x33	0x89	0x07	0xE9	0x02	0x00	0x23	0x01	DBH	DBL	DBH	DBL	STO1	STO0

RX: weitere Datenpakete bis die Übertragung gestoppt wird.

Funkübertragung des Kanals 1 des Knotens 0x3389 stoppen

TX:	STA	0x06	0x00	0x33	0x89	0x06	0xA6	STO1	STO0									
RX:	STA	0x50	0x1F	0x33	0x89	0x08	0xEB	STO1	STO0									

Aufnahme stoppen (Broadcast)

TX:	STA	0x06	0x00	0xFF	0xFF	0x07	0xA4	STO1	STO0									
RX:	STA	0x06	0x1F	0x33	0x89	0x09	0xE8	STO1	STO0									
RX:	STA	0x06	0x1F	0x77	0xCE	0x06	0xE8	STO1	STO0									

4.4 Allgemeine Signalaufbereitung

Die allgemeine Signalaufbereitung dient der Anpassung des Ausgangssignals des Sensorinterfaces an den Eingangsbereich des ADCs. Im Fall des MSP430F5438 können mit der integrierten Referenzspannungsquelle Signale im Bereich von 0 bis 2.5 V mit einer Auflösung von 12 Bit abgetastet werden. Hieraus ergeben sich Quantisierungsstufen in der Höhe von 0.61 mV. Um trotz der fix vorgegebenen Diskretisierungsstufen Sensorsignale mit geringer Amplitude wertmäßig zufriedenstellend auflösen zu können, müssen diese verstärkt werden. Die allgemeine Signalaufbereitung beinhaltet aus diesem Grund, wie in Abbildung 4.48 dargestellt, den invertierenden Verstärker LTC6910 von Linear Technology dessen Verstärkung mittels des 3 Bit breiten Signals `C_G` gewählt werden kann. Der LTC6910 ist in drei Versionen mit unterschiedlichen Verstärkungsstufen erhältlich (siehe Tabelle 4.18). Die derzeitige Version des Moduls verwendet den LTC6910-2, wobei die drei Varianten jedoch pincompatibel sind und bei Bedarf ebenfalls der LTC6910-1 oder der LTC6910-3 bestückt werden können.

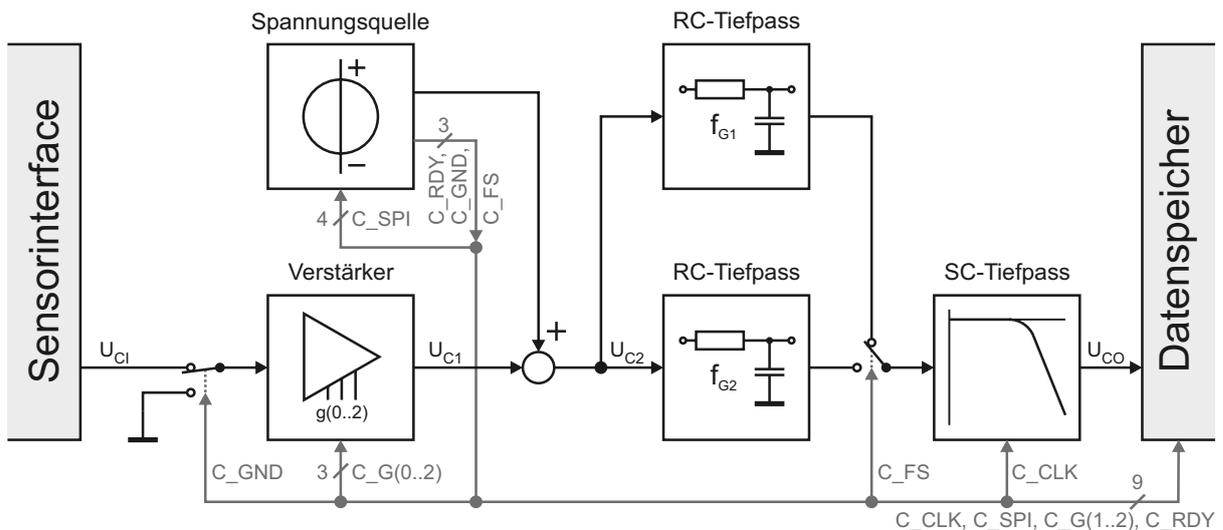


Abbildung 4.48: Blockschaltbild der allgemeinen Signalaufbereitung

Sensorsignale können sowohl in bipolarer als auch in unipolarer Form vorliegen. Da der ADC des MSP430F5438 lediglich unipolare Signale im Bereich von 0 bis 2.5 V verarbeiten kann, müssen bipolare Signale durch eine Pegelanpassung in diesen Bereich transformiert werden. Dies wird im Fall der analogen Signalaufbereitung durch die Addition einer mittels SPI-Interface (`C_SPI`) steuerbaren Offsetspannung (maximal ± 1.5 V) erreicht. Um die absolute Lage des Bezugspotentials (0 V) zu überprüfen kann die Signalmasse mittels

Steuersignal C_G			Verstärkung		
Bit 2	Bit 1	Bit 0	6910-1	6910-2	6910-3
0	0	0	0	0	0
0	0	1	-1	-1	-1
0	1	0	-2	-2	-2
0	1	1	-5	-4	-3
1	0	0	-10	-8	-4
1	0	1	-20	-16	-5
1	1	0	-50	-32	-6
1	1	1	-100	-64	-7

Tabelle 4.18: Verstärkungen des LTC6010-1, LTC6910-2 und LTC6910-3 laut [56]

eines Analogschalters durch das Steuersignal C_GND ebenfalls als Eingangssignal für die allgemeine Signalaufbereitung gewählt werden.

Neben der Anpassung des Sensorsignals an den Spannungsbereich des ADCs muss zusätzlich eine Anpassung im Frequenzbereich erfolgen um Aliasing zu vermeiden. Dies geschieht durch eine Tiefpassfilterung des Signals, wobei für das Ausgangssignal des Filters das Abtasttheorem von Nyquist und Shannon erfüllt sein muss. Eine Anpassung der oberen Grenzfrequenz des Antialiasing-Filters wird durch die Verwendung eines Switched-Capacitor-Filters (SC-Filter) erreicht, dessen Eckfrequenz durch die Frequenz des Taktsignales C_CLK bestimmt wird. Da SC-Filter zeitdiskret arbeiten benötigen diese vorgeschaltete analoge Filter, die in der Regel als RC-Tiefpässe ausgeführt werden. Wie in Abbildung 4.48 dargestellt, werden im Rahmen der allgemeinen Signalaufbereitung zwei mittels C_FS umschaltbare Tiefpassfilter eingesetzt um zwei Abtastfrequenzen ohne Eingriffe in die Hardware (Anpassung von Widerständen bzw. Kondensatoren) zu unterstützen.

4.4.1 Eingangsschaltkreis und Verstärker

Abbildung 4.49 zeigt den Eingangsschaltkreis der analogen Signalaufbereitung inklusive LTC6910. Da die Eingangsimpedanz des LTC6910 von der gewählten Verstärkung abhängt und laut Datenblatt [56] im Bereich von 1 bis 10 k Ω liegt, wurde dem LTC6910 ein Spannungsfollower vorgeschaltet. Der hierfür eingesetzte OPA2738 besitzt Rail-to-Rail Ein- und Ausgänge, die einen Spannungshub erlauben, der betragsmäßig 15 mV unterhalb der jeweiligen Versorgungsspannung liegt [99]. Weist die positive Versorgungsspannung dementsprechend ein Wert von mindestens 2.515 V auf, so kann mit diesem Operations-

verstärker der gesamte unipolare Eingangsbereich des ADCs abgedeckt werden. Im Fall des LTC6910 ist eine asymmetrische Versorgung mit -2.5 und $+3.3$ V erforderlich um den gesamten unipolaren Bereich von 0 bis 2.5 V nützen zu können.

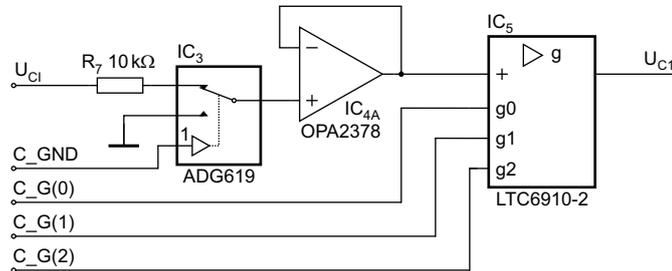


Abbildung 4.49: Schaltbild des Verstärkers LTC6910

Die Umschaltung zwischen dem Sensorsignal ($C_GND = \text{high}$) und der Signalmasse ($C_GND = \text{low}$) erfolgt durch den Analogschalter ADG619. R_7 dient der Strombegrenzung bei Überspannungen am analogen Eingang des Moduls.

4.4.2 Pegelanpassung

Das Schaltbild der Pegelanpassung ist in Abbildung 4.50 dargestellt. Sie erfolgt mittels des, durch IC_{8B} , R_{10} , R_{11} , R_{12} und R_{13} , gebildeten Subtrahierers. Zum Ausgleich der ne-

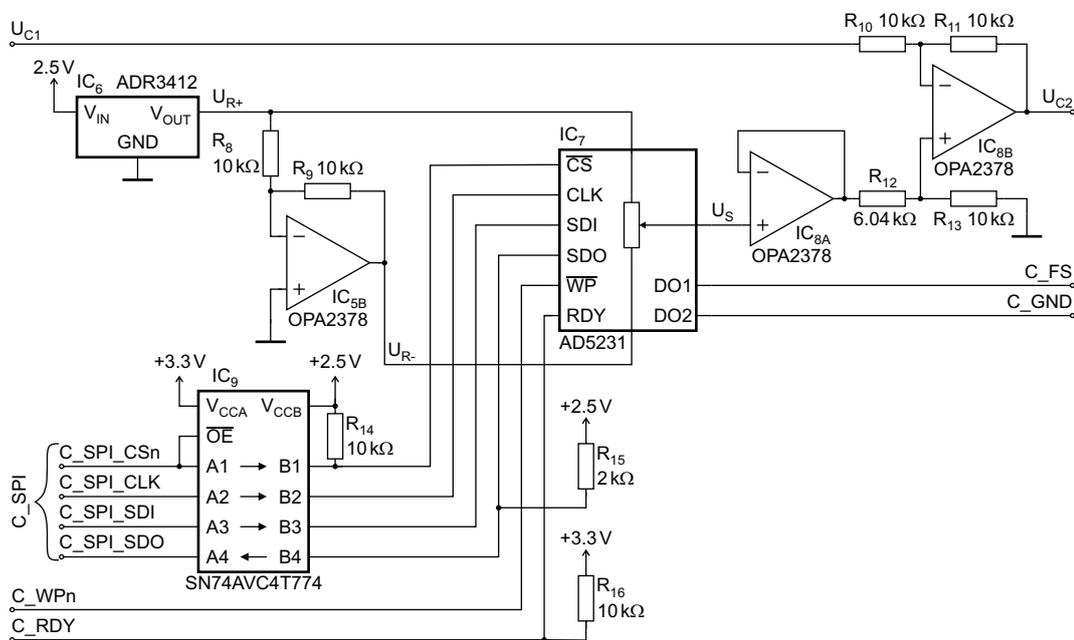


Abbildung 4.50: Schaltbild der Pegelanpassung

gativen Verstärkung des LTC6910 wurde das Sensorsignal an den invertierenden Eingang gelegt. Für die Ausgangsspannung der Schaltung erhält man nach kurzer Rechnung

$$U_{C2} = -\frac{R_{11}}{R_{10}}U_{C1} + \frac{R_{13}}{R_{12} + R_{13}} \left(1 + \frac{R_{11}}{R_{10}}\right) U_S \quad (4.16)$$

Die Widerstände wurden so gewählt, dass mit einer Referenzspannung von $U_{R+} = 1.2\text{ V}$ (ADR3412) ein Offsetbereich von $\pm 1.5\text{ V}$ abgedeckt werden kann. Für U_{C2} erhält man dementsprechend

$$U_{C2} = -U_{C1} + 1.247 U_S. \quad (4.17)$$

In weiterer Folge gilt für die Ausgangsspannung U_{SO} des Moduls mit der Verstärkung A_{PGA} des LTC1069 im Passbandbereich des Antialiasingfilters

$$U_{SO} = A_{PGA}U_{SI} + 1.247 U_S. \quad (4.18)$$

Die Spannung U_S wird unter Verwendung des digitalen Potentiometers AD5231 generiert. Dieses wird mittels SPI-Schnittstelle (C_SPI) angesteuert und erlaubt die Verstellung des elektronischen Gleitkontakts in 1024 Schritten. Der Offset kann daher unter Berücksichtigung von (4.18) in Schritten von 2.9 mV variiert werden. Da die Versorgung des AD5231 mit $\pm 2.5\text{ V}$ erfolgt und dessen digitales Interface folglich auf 2.5 V-CMOS -Pegel ausgelegt ist, wird zur Ansteuerung durch den MSP430F5438 (3.3 V-LVTTL) der Pegelwandler IC_9 benötigt.

Um mehrere Bausteine über die selbe SPI-Schnittstelle ansprechen zu können, muss der Ausgang C_SPI_SDO des Pegelwandlers in den hochohmigen Zustand geschaltet werden können. Dies wird durch die Verbindung von S_SPI_CS_n mit dem Output-Enable-Signal des Pegelwandlers erreicht.

Mit Hilfe des Signals C_RDY signalisiert der AD5231, die erfolgte Ausführung von bestimmten Kommandos (Details siehe [3]). Es handelt sich hierbei um einen Open-Drain-Ausgang, weshalb kein Pegelwandler erforderlich ist. Weiters kann mittels C_WP_n der Flash-Speicher des ICs vor einem Überschreiben der gespeicherten Werte geschützt werden.

Zusätzlich zur eigentlichen Potentiometerfunktionalität besitzt der AD5231 die digitalen Ausgänge DO1 und DO2, die mittels SPI-Interface auf high oder low gesetzt werden können. Durch die Verwendung dieser Ausgänge zur Ansteuerung der vorhandenen Analogschalter (Signale C_GND und C_FS) können pro Messkanal zwei Verbindungsleitungen zwischen Datenspeicher und allgemeiner Signalaufbereitung eingespart werden,

Auf Grund des mit $35\text{ k}\Omega$ relativ geringen Eingangswiderstandes und des geringen maximalen Ausgangsstroms von $+8$ bzw. -1 mA wurden dem LTC1069-6 jeweils ein Spannungsfolger vor- und nachgeschaltet. Weiters wurde ein durch R_{19} und C_{14} gebildetes Glättungsfilter nachgeschaltet um die Amplituden der durch die sprunghafte Änderung des Ausgangssignals verursachten höherfrequenten Signalanteile zu reduzieren.

Hinsichtlich der Auslegung des SC-Filters zur Vermeidung bzw. Reduktion von Aliasing ergeben sich bei einer Abtastrate von $f_s = 1\text{ kHz}$ des ADCs folgende Anforderungen: Um eine möglichst geringe Phasenverschiebung im relevanten Frequenzbereich des Messsignals von 0 bis 100 Hz zu erreichen sollte die Grenzfrequenz des Filters möglichst hoch gewählt werden. Die obere Grenze von $f_{g,SC}$ wird jedoch durch die Nyquistfrequenz f_{Ny} vorgegeben, wobei im Idealfall bei $f_s/2$ die Dämpfung mindestens dem Signal-Rausch-Verhältnis des eingesetzten ADCs entspricht [65]. Theoretisch sollte die Dämpfung laut [65] bei 12 Bit-ADCs zumindest 74 dB betragen. Da in erster Linie die (Starrkörper-) Schwingungen der Wagenkästen gemessen werden sollen, ist auf Grund der Tiefpasswirkung des mechanischen Systems nicht davon auszugehen, dass bei Frequenzen oberhalb von f_{Ny} Signalanteile mit Spitzen-Spitzen-Werten im Bereich des vollen Messbereichs auftreten werden. Entsprechend dem Datenblatt des LTC1069-6 [52] gilt für die Verstärkung im Übergangsbereich des Filters

$$A_{SC}(1.3f_{g,SC}) = -42\text{ dB} \quad (4.22)$$

bzw.

$$A_{SC}(2.1f_{g,SC}) = -70\text{ dB}. \quad (4.23)$$

Unter Berücksichtigung der bereits genannten Gesichtspunkte wurde $f_{g,SC} = 375\text{ Hz}$ gewählt. Hieraus ergibt sich ein maximaler, durch Aliasing verursachten, systematischer Fehler von 0.8% (bezogen auf den Skalenendwert des gewählten Messbereichs) bei einer möglichst geringen Phasenverschiebung im relevanten Frequenzbereich.

Für die Frequenz des Taktsignales f_{C_CLK} bei der maximalen Abtastrate $f_{s,1} = 1\text{ kHz}$ erhält man aus (4.19)

$$f_{C_CLK} = 50 \cdot 375\text{ Hz} = 18.75\text{ kHz}. \quad (4.24)$$

Der zugehörige, durch R_{17} und C_{12} gebildete, RC-Tiefpass besitzt eine obere Grenzfrequenz von

$$f_{g_RC,1} = \frac{1}{2\pi R_{17} C_{12}} = 1.17\text{ kHz}. \quad (4.25)$$

Da die Abtastrate des SC-Filters nach (4.20) bei 37.5 kHz liegt, werden ab einer Fre-

quenz von 37 kHz erstmals Signale in den Bereich von 0 bis 500 Hz abgebildet. Ein Signal mit einer Frequenz von 37 kHz würde bei 500 Hz erscheinen. Die Dämpfung durch den RC-Tiefpass beträgt bei 37 kHz nach (3.2) 30 dB. Auf Grund des Einsatzbereiches des Messsystems ist nicht mit nennenswerten Signalanteilen jenseits dieser Frequenz zu rechnen, weshalb eine Dämpfung von lediglich 30 dB in diesem Fall ausreichend ist.

Der zweite RC-Tiefpass (R_{18} und C_{13}) wurde für eine Abtastrate von 100 Hz ausgelegt. Die wesentlichen Kennwerte der Filter sowie die zugehörigen Charakteristika bzw. Werte der Ansteuersignale sind in Tabelle 4.19 zusammengefasst.

Abtastrate	f_{C_CLK} in kHz	$f_{g,SC}$ in Hz	$f_{g,RC}$ in Hz	Signal C_F
1 kHz	18.750	375.0	1.170	1
100 Hz	1.875	37.5	0.117	0

Tabelle 4.19: Filtergrenzfrequenzen und Steuersignale für die Abtastraten 1 kHz und 100 Hz

4.4.4 Spannungsversorgung der analogen Elektronik

Die, zusätzlich zu der bereits durch das Basismodul zur Verfügung gestellten Betriebsspannung von +3.3 V, zur Versorgung der analogen Elektronik benötigten Spannungen von ± 2.5 V werden durch die DC/DC-Konverter TPS62250 und TPS63700 von Texas Instruments generiert. Da diese dem LTC3526L nachgeschaltet sind, ist eine Soft-Start-Funktion zur Begrenzung des Spitzenstromes beim Start der Konverter erforderlich. Sowohl der TPS62250 als auch der TPS63700 verfügen über eine integrierte Soft-Start-Funktion. Weiters können sie mit einer konstanten Schaltfrequenz betrieben werden, wodurch die Ausgangswelligkeit auch bei kleinen Lasten gering gehalten wird. Die Schaltbilder der beiden Konverter sind in den Abbildungen 4.52 und 4.53 dargestellt.

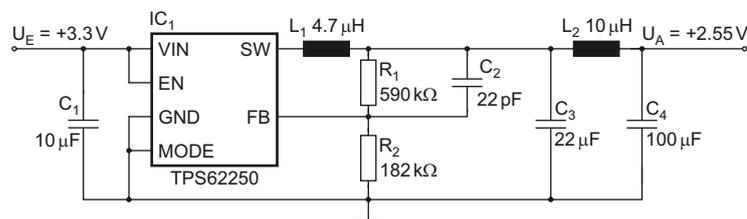


Abbildung 4.52: Schaltbild des TPS62250

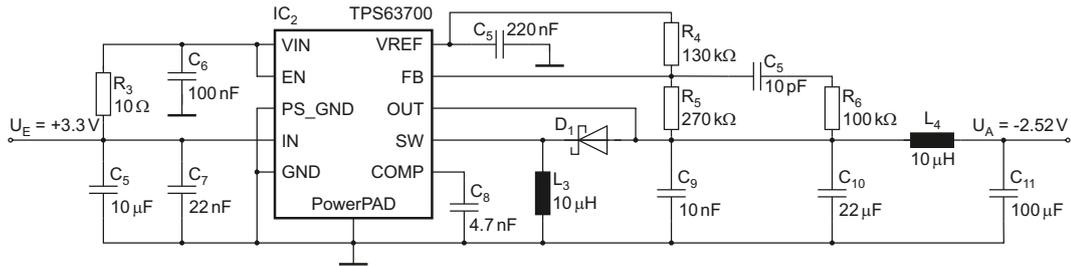


Abbildung 4.53: Schaltbild des TPS63700

Die Ausgangsspannung des TPS62250 wurde mittels des Spannungsteilers R_1 , R_2 auf 2.55 V festgelegt. Diese liegt 50 mV oberhalb der maximalen Spannung des Messsignales, wodurch eine Übersteuerung des OPA2378 (vergleiche Abschnitt 4.4.1) vermieden wird.

Zur Reduktion der Ausgangswelligkeit der DC/DC-Konverter wurden die Kondensatoren am Ausgang (C_3 , C_{10}) jeweils mittels einer Induktivität (L_2 , L_4) und eines weiteren Kondensators (C_4 , C_{11}) zu einem π -Netzwerk erweitert. Wie Abbildung 4.54 zeigt, wird hierdurch speziell beim TPS63700 eine deutliche Reduktion der Spannungsspitzen erreicht.

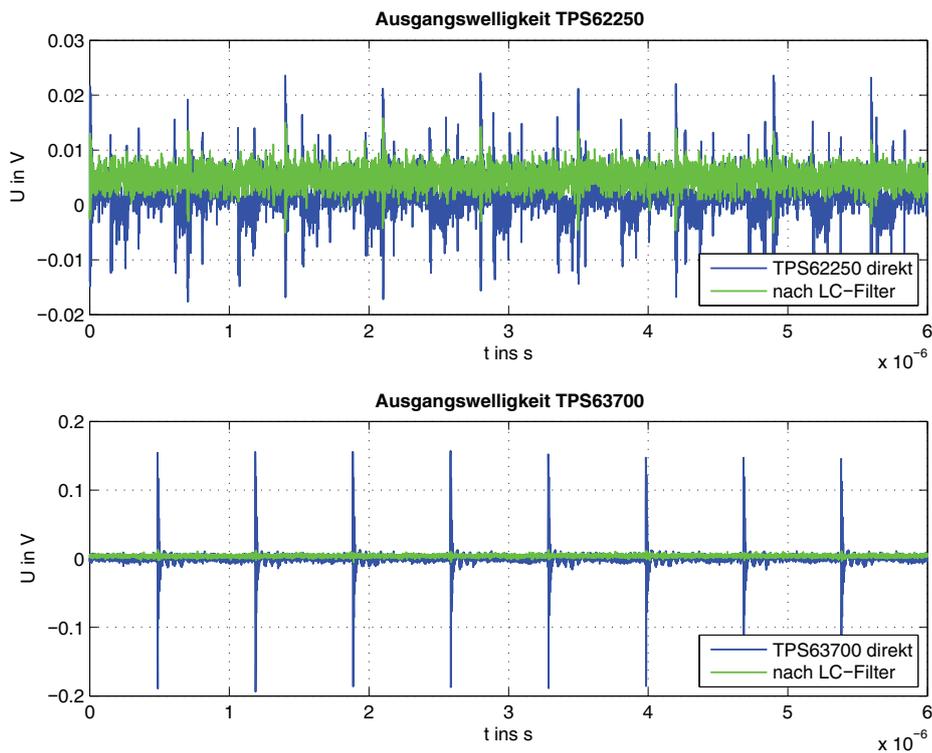


Abbildung 4.54: Ausgangswelligkeit des TPS62250 und TPS63700 (AC-Einkopplung, normale Betriebsbelastung durch die analoge Elektronik des Moduls)

4.4.5 Funktionstests

Abbildung 4.55 zeigt die Ein- und Ausgangsspannungen der allgemeinen Signalaufbereitung bei einem sinusförmigen Eingangssignal mit 10 Hz. Es wurde sowohl die uni- wie auch die bipolare Konfiguration des Moduls bei einer Verstärkung von 1 bzw. 64 getestet. Bei einer Verstärkung von 1 wurde eine Eingangsamplitude von 1.25 V gewählt. Dies entspricht einer Vollaussteuerung der Signalkette und darf zu keiner Begrenzung des Signals führen. Kritisch ist in dieser Hinsicht vor allem die unipolare Konfiguration im Bereich der maximalen Ausgangsspannung von 2.5 V. Bei einer Verstärkung von 64 wurde eine Amplitude von 10 mV gewählt, die der minimalen Ausgangsamplitude des verwendeten Funktionsgenerators Agilent 33220A entspricht. Wie in Abbildung 4.55 zu erkennen ist, stellt das Modul die geforderte Funktionalität ohne Übersteuerung zur Verfügung.

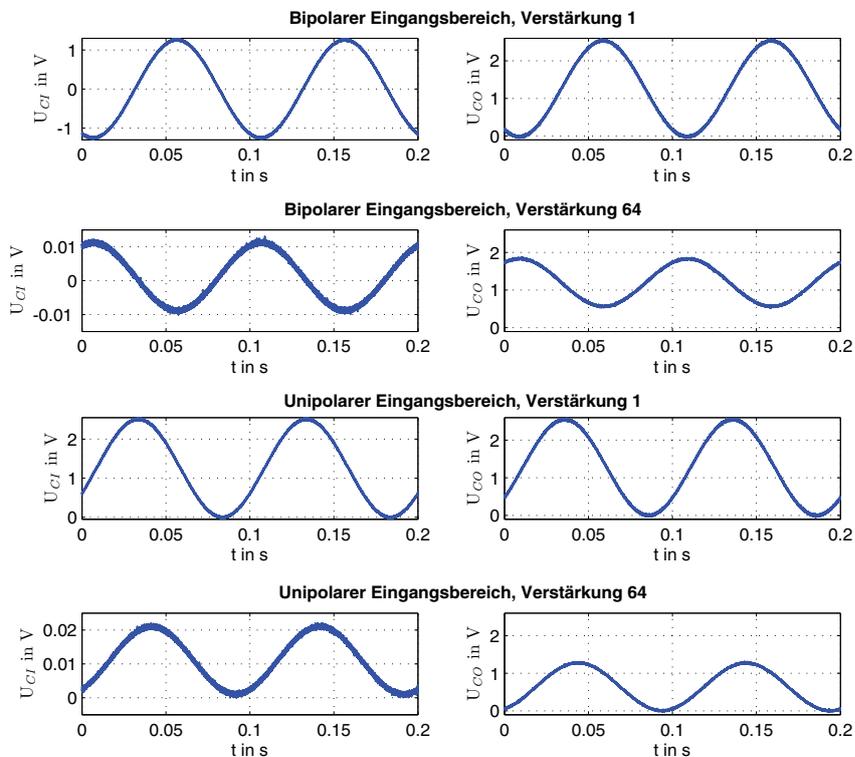


Abbildung 4.55: Ein- und Ausgangsspannungen der allgemeinen Signalaufbereitung in uni- und bipolarer Konfiguration ($f_{g,SC} = 375$ Hz)

Die Frequenzgänge des Moduls wurden mit Hilfe des Signalanalysators Agilent 35670A bestimmt und sind in den Abbildungen 4.56 und 4.57 dargestellt. Es wurden die Konfigurationen für die Samplingraten 1 kS/s ($f_{g,SC} = 375$ Hz) und 100 S/s ($f_{g,SC} = 37,5$ Hz) entsprechend Tabelle 4.19 bei verschiedenen Verstärkungen getestet. Der Phasengang ist

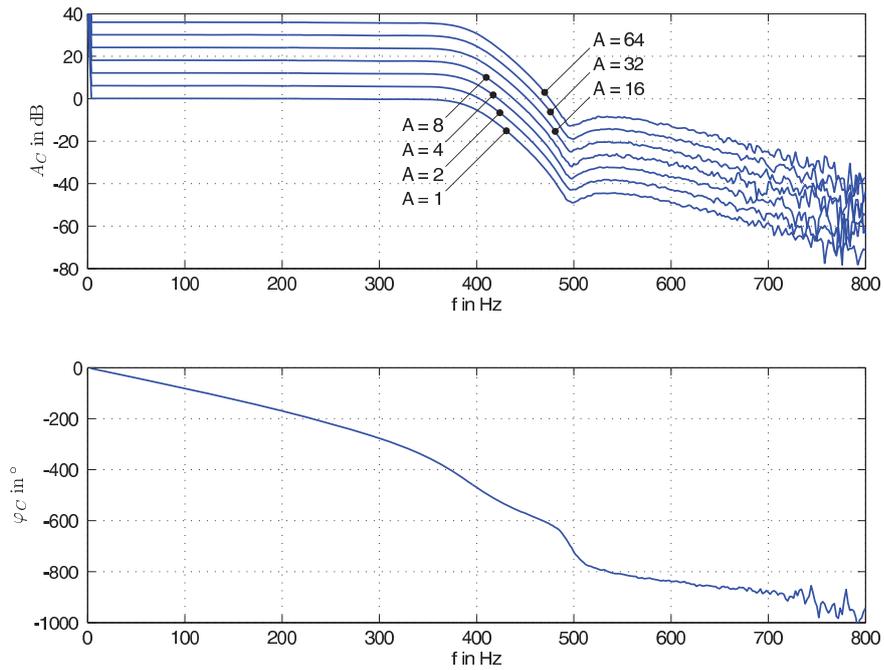


Abbildung 4.56: Frequenzgänge der allgemeinen Signalaufbereitung bei verschiedenen Verstärkungen für $f_{g,SC} = 375$ Hz (bipolare Konfiguration)

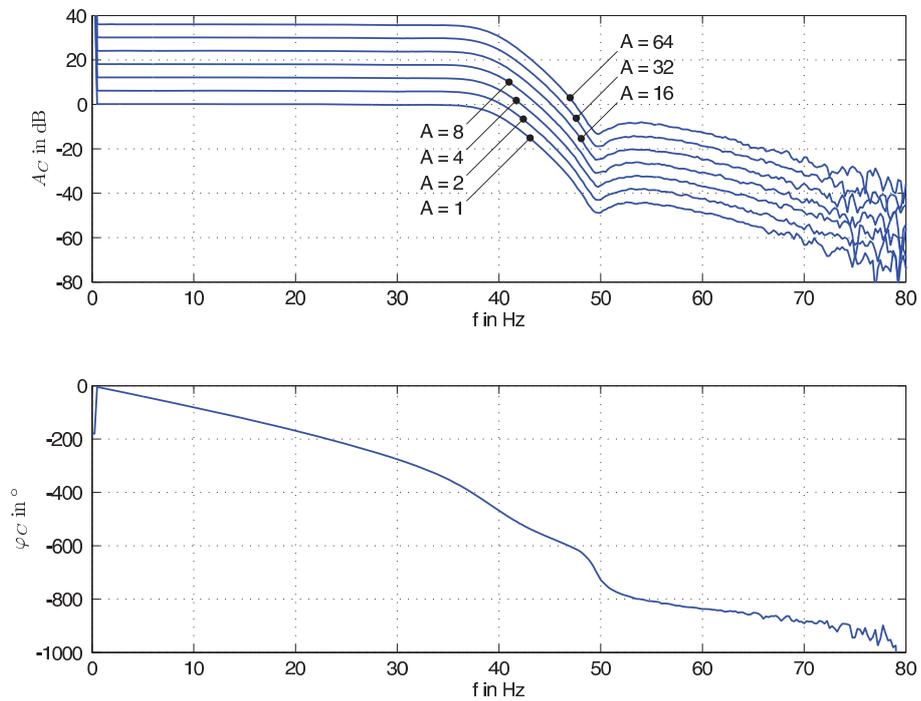


Abbildung 4.57: Frequenzgänge der allgemeinen Signalaufbereitung bei verschiedenen Verstärkungen für $f_{g,SC} = 37.5$ Hz (bipolare Konfiguration)

im betrachteten Frequenzbereich unabhängig von der gewählten Verstärkung, weshalb dieser bei einer Verstärkung von 1 repräsentativ für alle Verstärkungen abgebildet wurde. Da die Messung von Beschleunigungen mittels des in Abschnitt 4.5 beschriebenen IEPE-Interfaces mit bipolarer Ausgangsspannung die vorherrschende Messaufgabe darstellt, wurde für die Messungen die bipolare Konfiguration gewählt. Der additive Gleichanteil, der durch den Pegelwandler des Moduls erzeugt wird, führt im Bereich von 0 Hz zu den in den Abbildungen 4.56 und 4.57 erkennbaren Spitzen bzw. zu Problemen bei der Bestimmung des Phasenganges in diesem Bereich. Eine Dämpfung um 44 dB relativ zur Verstärkung im Durchlassbereich des Filters ist entsprechend den Messergebnissen ab 500 Hz bzw. 50 Hz sichergestellt. Dies entspricht den Auslegungskriterien in Abschnitt 4.4.3.

4.4.6 Strombedarf

Der Strombedarf der allgemeinen Signalaufbereitung, abhängig vom Betriebszustand, ist in Tabelle 4.20 dargestellt. Der Wert im normalen Betrieb wurde bei $f_{C_CLK} = 18.7$ kHz mit Hilfe von drei Multimetern (Amprobe 37XR-A) bestimmt. Beim Lesen bzw. Schreiben des Flash-Speichers vom digitalen Potentiometer AD5231 werden zusätzlich zum Ruhestrom 9 mA bzw. 40 mA benötigt. Das Setzen der Ausgänge DO1 (C_FS) und DO2 (C_GND) erfordert einen Schreib-/Lesezugriff auf das Flash des AD5231, weshalb der zusätzliche Strombedarf in der Regel nur bei einer Änderung der Konfiguration der allgemeinen Signalaufbereitung auftritt.

Betriebszustand	+3.3 V	+2.5 V	-2.5 V
normaler Betrieb	7.68 mA	1.57 mA	-5.17 mA
Flash – Schreiben	–	40 mA	-40 mA
Flash – Lesen	–	9 mA	-9 mA

Tabelle 4.20: Strombedarf der allgemeinen Signalaufbereitung

4.4.7 Entwicklungsgeschichte

Die im Zuge der Entwicklung entstandenen Prototypen der allgemeinen Signalaufbereitung sind in Abbildung 4.58 dargestellt. Ausgangspunkt war die links dargestellte „Versuchsplatine LTC6910 und LTC1069“, die bereits die wesentliche Funktionalität hinsichtlich Verstärkung und Filterung für bipolare Eingangssignale bereitstellt. Hierauf folgte die

Anpassung an das Format des Laborsystems, wobei das Design ebenfalls um die DC/DC-Konverter (TPS62205 und LM2611) zur Generierung der zusätzlichen Versorgungsspannungen erweitert wurde (V1.0). Auf Grund der nicht vorhandenen Soft-Start-Funktion gelangte der LM2611 nicht in den normalen Betriebszustand und musste daher ersetzt werden. Im Zuge des Neuentwurfs wurde der TPS62205 ebenfalls ersetzt, da er wegen des Power-Save-Mode bei geringer Last, eine erhöhte Ausgangswelligkeit aufwies.

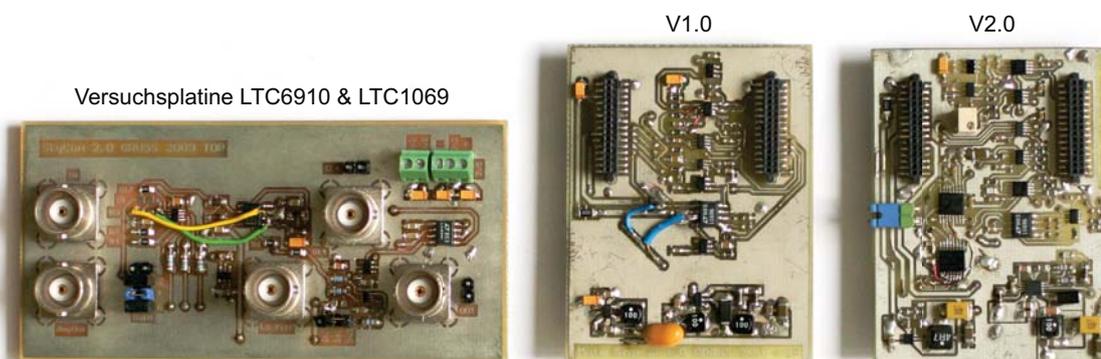


Abbildung 4.58: Prototypen der allgemeinen Signalaufbereitung

Für die Version 2.0 wurden die in Abschnitt 4.4.4 beschriebenen DC/DC-Konverter TPS62250 und TPS63700 verwendet. Weiters wurde die Signalaufbereitung durch die Verwendung des digitalen Potentiometers AD5231 um die Möglichkeit der Verarbeitung von unipolaren Signalen erweitert. Die Version V2.0 stellt den aktuellen Stand der Elektronik gemäß den Abschnitten 4.4.1 bis 4.4.4 dar.

4.5 IEPE-Sensorinterface

Die grundlegende Funktion von IEPE-Beschleunigungssensoren wurde bereits in Abschnitt 3.2.6.1 erläutert, an dieser Stelle wird in weiterer Folge auf die schaltungstechnische Realisierung des IEPE-Interfaces des Laborsystems eingegangen. Der prinzipielle Aufbau des Interfaces ist in Abbildung 4.59 in Form eines Blockschaltbildes dargestellt.

Die Versorgung der Konstantstromquelle, die der Energieversorgung des Sensors dient, erfolgt mit einer Gleichspannung von +24 V. Diese Spannung wird mittels eines DC/DC-Konverters aus der +3.3 V-Versorgungsspannung des Basismoduls generiert. Da der Arbeitspunkt der Sensors bei etwa 13 V liegt und je nach Sensor bei Vollaussteuerung Amplituden im Bereich von ca. 8 V auftreten, wird zur Abdeckung des gesamten Messbereichs der Sensoren mittels des verwendeten ADCs ein, bei Bedarf mittels S_D zuschaltbarer,

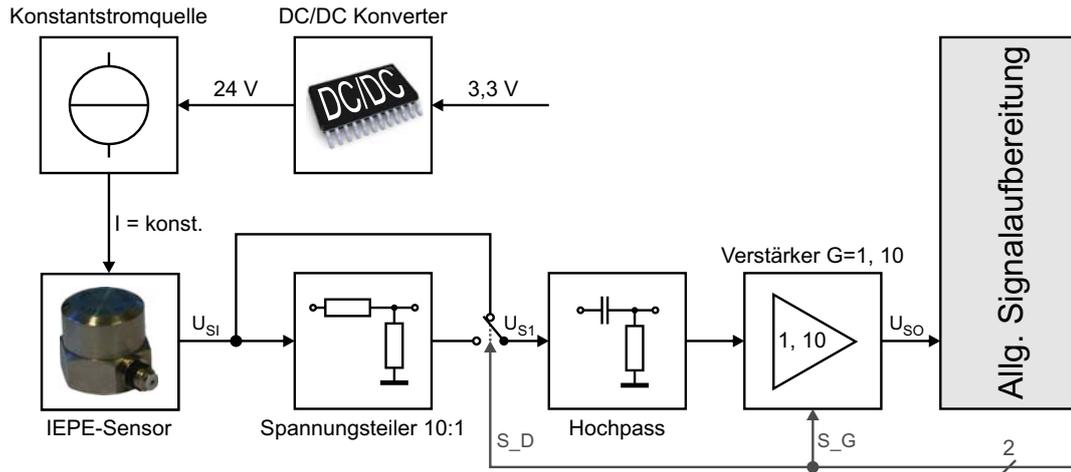


Abbildung 4.59: Blockschaltbild des IEPE-Interfaces

10:1-Spannungsteiler benötigt. Anschließend wird durch einen Hochpass der Gleichanteil des Sensorsignals eliminiert. Über einen driftkompensierten Verstärker, dessen Verstärkung mittels Steuersignal S_G zwischen den Werten 1 und 10 umgeschaltet werden kann, ist eine weitere Anpassungsmöglichkeit der Ausgangspegel des IEPE-Interfaces gegeben. Tabelle 4.21 fasst die, mit Hilfe von S_D und S_G , wählbaren Verstärkungen des Moduls zusammen. Auf die in Abbildung 4.59 dargestellten Funktionsblöcke wird im weiteren Verlauf dieses Abschnitts noch näher eingegangen.

S_D	S_G	Verstärkung
0	0	1
0	1	10
1	0	0.1
1	1	1

Tabelle 4.21: Verstärkung des IEPE-Interfaces

4.5.1 DC/DC-Konverter

Die Versorgungsspannung der Konstantstromquelle erfolgt mittels des DC/DC-Konverters TPS61081 [100], dessen Schaltbild in Abbildung 4.60 dargestellt ist. Der TPS61081 ist für Eingangsspannungen von 2.5 bis 6 V ausgelegt und ist in der Lage Ausgangsspannungen bis zu 27 V zu erzeugen. Die integrierten Schutzfunktionen umfassen eine Begrenzung des Stroms durch L_1 auf 1.3 A, die Trennung von Primär und Sekundärseite mittel eines integrierten MOSFETs im Fall eines sekundärseitigen Kurzschlusses, einen sekundärseitigen

Überspannungsschutz sowie den Schutz vor thermischer Überlastung. Weiters kann der Konverter mit einer fixen Schaltfrequenz von 600 kHz oder 1.2 MHz betrieben werden und verfügt über eine Soft-Start-Funktion.

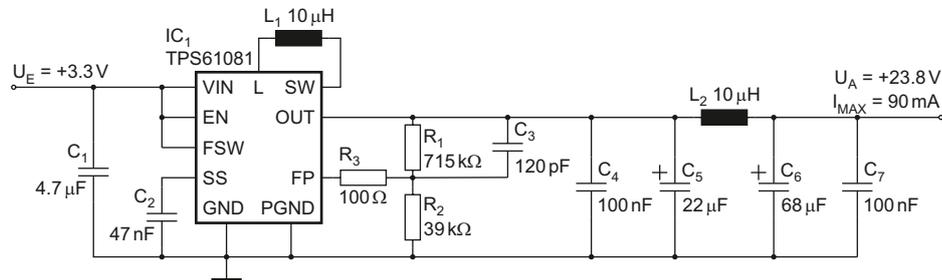


Abbildung 4.60: Schaltbild des TPS61081

Abbildung 4.61 zeigt die Ausgangswelligkeit des TPS61081 bei einer Eingangsspannung von +3.3 V, einer Schaltfrequenz von 1.2 MHz und einem Laststrom von 4 mA. Die Wirksamkeit des dem DC/DC-Konverter nachgeschalteten LC-Tiefpasses ist anhand der Reduktion der Schaltspitzen deutlich erkennbar.

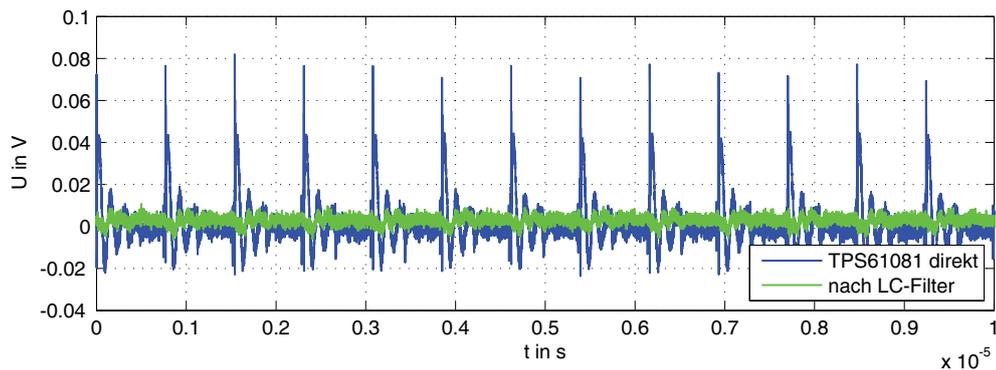


Abbildung 4.61: Ausgangswelligkeit des TPS61081 bei 4 mA Laststrom (AC-Einkopplung)

4.5.2 Stromquelle LT3092

Die Konstantstromquelle des IEPE-Interfaces beruht auf dem LT3092 von Linear Technology. Entsprechend dem Schaltbild in Abbildung 4.62 kann mittels des Trimmers R_6 der Strom im Bereich von 0.5 bis 10 mA eingestellt werden. Der LT3092 wird direkt mittels des TPS61081 mit einer Spannung von +23.8 V versorgt. Bei Bedarf kann diese nach [50] bis

zu +27 V betragen, wobei jedoch die thermische Belastung in Folge der Verluste zu berücksichtigen ist. Der IC verfügt sowohl über einen thermischen Überlastschutz als auch über eine Strombegrenzungsfunktion. Details können den Datenblatt [50] entnommen werden.

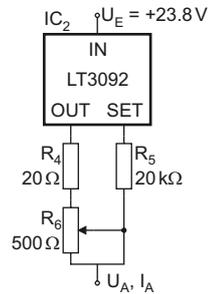


Abbildung 4.62: Schaltbild der Stromquelle

4.5.3 Spannungsteiler

Der Spannungsteiler wurde unter Verwendung des Analogschalters DG470 von Vishay Siliconix und des Operationsverstärkers OPA2244 von Texas Instruments entsprechend Abbildung 4.63 ausgeführt. Beide ICs werden durch den TPS61081 versorgt und können laut den Datenblättern [98,108] mit Versorgungsspannungen bis zu +44 V (DG470) bzw. +36 V (OPA2244) betrieben werden.

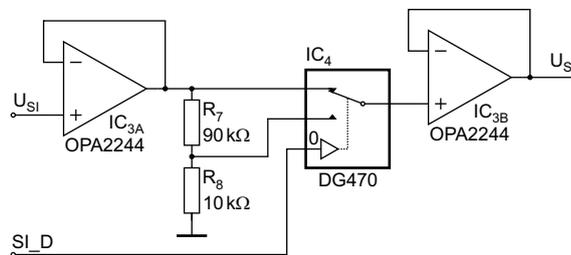


Abbildung 4.63: Schaltbild des Spannungsteilers

4.5.4 Hochpass und Verstärker

Den Ausgangspunkt der Entwicklung des Hochpasses und des Verstärkers des IEPE-Interfaces stellte ein bereits zuvor am Institut entwickeltes Interface für IEPE-Beschleunigungssensoren dar, dessen Aufbau in Abbildung 4.64 in Form eines Blockschaltbildes dargestellt ist.

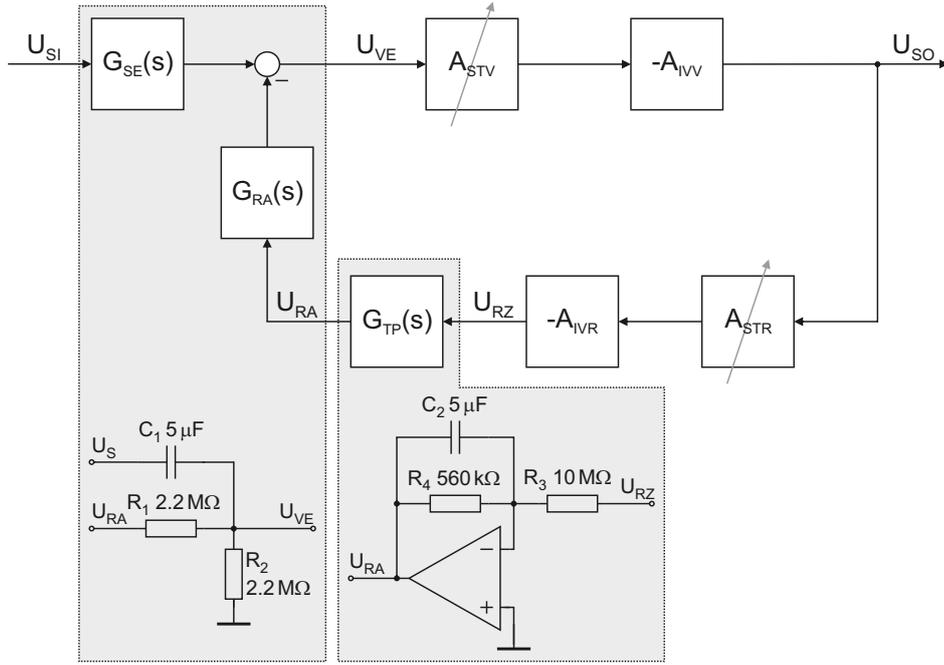


Abbildung 4.64: Blockschaltbild des IEPE-Interfaces mit variabler Verstärkung – Ausgangspunkt der Entwicklung

Die Schaltung besteht im Wesentlichen aus variablen Spannungsteilern mit den Verstärkungen $A_{STV} < 1$ und $A_{STR} < 1$ sowie invertierenden Verstärkern mit den fixen Verstärkungen $-A_{IVV} < 0$ und $-A_{IVR} < 0$. Im Frequenzbereich bis 1 kHz können diese als Blöcke mit frequenzunabhängigem Übertragungsverhalten betrachtet werden. Jeweils ein Spannungsteiler und ein invertierender Verstärker befinden sich im Vorwärts- und Rückkopplungs-Zweig der Schaltung. Im Rückkopplungszweig befindet sich darüber hinaus ein Tiefpass mit der Übertragungsfunktion

$$G_{TP}(s) = -\frac{R_4}{R_3 + R_3 R_4 C_2 s}. \quad (4.26)$$

Die Einkopplung des Sensorsignals und der Ausgangsspannung des Rückkopplungszweigs erfolgt mittels einer Kombination von zwei Widerständen (R_{RA} , R_2) und einem aus zwei antiseriell geschalteten Tantalkondensatoren gebildeten bipolaren $5 \mu\text{F}$ -Kondensator (C_1). Für das Übertragungsverhalten dieser Teilschaltung erhält man nach kurzer Rechnung

$$U_{VE}(s) = G_{SE}(s)U_S(s) + G_{RA}(s)U_{RA}(s) \quad (4.27)$$

$$= \frac{R_1 R_2 C_1 s}{R_1 + R_2 + R_1 R_2 C_1 s} U_S(s) + \frac{R_2}{R_1 + R_2 + R_1 R_2 C_1 s} U_{RA}(s) \quad (4.28)$$

Mit (4.26) und (4.28) wird deutlich, dass auf Grund des fehlenden I-Anteils bei dieser Schaltung eine eventuelle Offsetdrift nicht vollständig kompensiert wird.

Im ersten Entwicklungsschritt wurde zunächst der prinzipielle Aufbau des Vorwärtszweiges der Schaltung beibehalten, wobei jedoch – abweichend von der Ausgangsschaltung – für die geplanten Betriebsspannungen von $\pm 2.5\text{ V}$ geeignete Operationsverstärker (OPA2378 von Texas Instruments) und Analogschalter (ADG619 von Analog Devices) gewählt werden mussten. Abbildung 4.65 zeigt das zugehörige Schaltbild für die Prototypen 1 und 2.

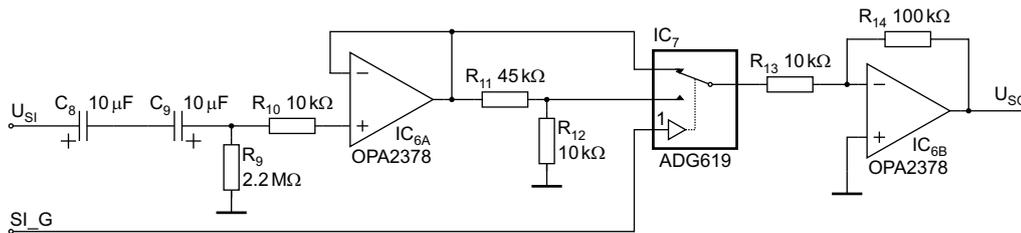


Abbildung 4.65: Schaltbild des Hochpasses und Verstärkers für die Prototypen 1 & 2

Messungen am Prototypen ergaben folgende Offsetspannungen in mV:

$$U_{SO} = \{-142, -119, -108, -105, -102, -92, -93, -91, -90, -88\}.$$

Ursache für diesen Offset sind in erster Linie die Leckströme des Operationsverstärkers und der Tantalkondensatoren. Vernachlässigt man die mit maximal $70\ \mu\text{V}$ sehr geringe Offsetspannung des OPA2378 wäre ein Leckstrom von $4.1\ \text{nA}$ nötig um eine Offsetspannung von $90\ \text{mV}$ am Ausgang der Schaltung hervorzurufen. Der Leckstrom des Operationsverstärkers beträgt laut Datenblatt $\pm 670\ \text{pA}$ und kann daher nicht den alleinigen Verursacher des beobachteten Offsets darstellen. Hinsichtlich des tatsächlich vorhandenen Leckstromes der Tantalkondensatoren ist im Datenblatt des T491D106K035AT von KEMET lediglich eine obere Schranke von $1\ \text{nA}\ \mu\text{F}^{-1}\ \text{V}^{-1}$ angegeben. Dies würde für den hier verwendeten Kondensator mit $10\ \mu\text{F}$ und einer maximalen Spannung von $35\ \text{V}$ einen Leckstrom von $3.5\ \mu\text{A}$ bei $35\ \text{V}$ ergeben. Nach [24] sind jedoch typischerweise $0.1\ \text{nA}\ \mu\text{F}^{-1}\ \text{V}^{-1}$ zu erwarten, woraus ein Leckstrom von $35\ \text{nA}$ bei einer Spannung von $35\ \text{V}$ folgt. Auf Grund der nichtlinearen Abhängigkeit des Leckstromes von Spannung, Temperatur und nicht zuletzt dem Aufbau, Typ und Hersteller des Kondensators ist hier keine nähere Eingrenzung des Leckstroms möglich. Der beobachtete Offset am Ausgang der Schaltung ist jedoch mit den im Datenblatt angegebenen bzw. abgeschätzten Leckströmen des Operationsverstärkers und der Kondensatoren vereinbar.

Da dieser Offset in Relation zum Eingangsspannungsbereich des ADCs bereits 4% beträgt und darüber hinaus unter anderem von der Temperatur abhängig ist, ist eine Kompensation des Offsets bzw. des Offsetdrifts erforderlich. Dies wurde, wie in Abbildung 4.66 dargestellt, beim Prototyp 3 durch die Verwendung eines Integrators im Rückführungszweig sowie der Schließung des Rückkopplungszweiges mittels eines Subtrahierers erreicht. Dies entspricht dem in Abschnitt 3.2.5 beschriebenen, durch Rückkopplung gebildeten, Hochpass, wobei dem ursprünglichen, in Abbildung 3.17 dargestellten, System hier ein zusätzlicher RC-Hochpass vorgeschaltet ist.

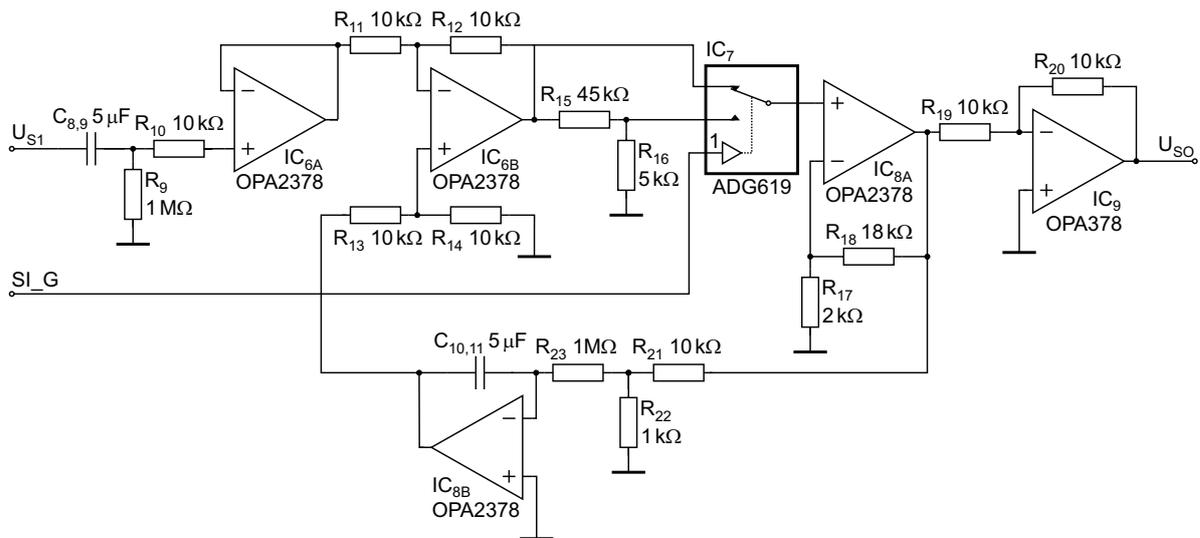


Abbildung 4.66: Schaltbild des Hochpasses und Verstärkers – Prototyp 3

Bei den, entsprechend Abbildung 4.66, gewählten Werten erhält man eine untere Grenzfrequenz von 32 mHz für den mittels $C_{8,9}$ und R_9 gebildeten RC-Hochpass sowie eine untere Grenzfrequenz von 2.9 mHz (Verstärkung 1) bzw. 29 mHz (Verstärkung 10) für den Verstärker mit Rückkopplung. SPICE-Simulationen mittels Tina-TI ergaben eine untere Grenzfrequenz der gesamten Schaltung von 32 mHz bei einer Verstärkung von 1 bzw. 47 mHz bei einer Verstärkung von 10. Die angestrebte untere Grenzfrequenz von 0.1 Hz konnte daher zumindest um den Faktor 2 unterschritten werden.

Abbildung 4.67 zeigt die Ausgangsspannungen des Prototypen 2 (Schaltbild entsprechend Abbildung 4.65) und des Prototypen 3 (Schaltbild entsprechend Abbildung 4.66) bei einer kurzen Temperaturerhöhung durch Anblasen der Schaltung für 10 s mit 40 °C warmer Luft. Sowohl der deutlich reduzierte Offset der Ausgangsspannung als auch der wesentlich geringere und kürzere temperaturbedingte Spannungsausgang sind hier deutlich erkennbar.

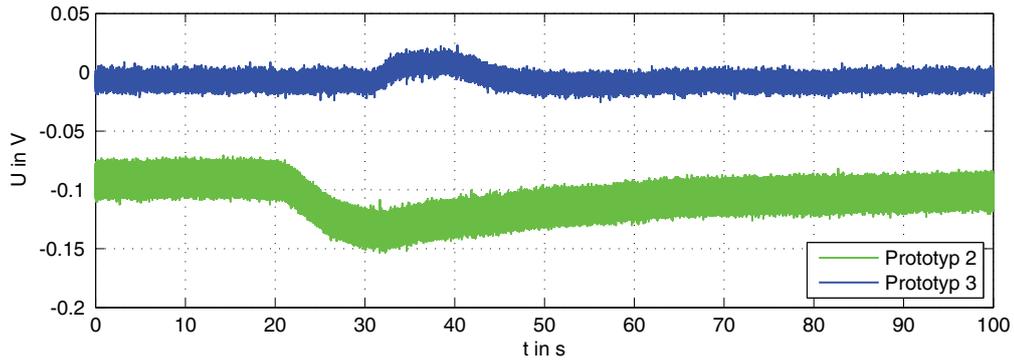


Abbildung 4.67: Ausgangsspannungen der Versionen 2.0 und 3.0 des IEPE-Interfaces bei Temperaturänderung

4.5.5 Frequenzgänge des IEPE-Sensorinterfaces

Die abschließende Überprüfung, ob die entwickelte Schaltung im relevanten Frequenzbereich die gewünschte Funktionalität tatsächlich aufweist, erfolgte in zwei Schritten. Im ersten Schritt wurde der Frequenzgang des in Abbildung 4.66 dargestellten Schaltungsteils überprüft und danach die der gesamten Schaltung mit angeschlossenem Sensor. Als Referenz für den letzten Schritt diente eine Messkette von Brüel & Kjær. Die im Zuge dieser Messungen verwendeten Geräte sind in Tabelle 4.22 zusammengefasst. Die Kalibrierdaten der verwendeten Sensoren sind im Anhang in den Abbildungen F.3 und F.1 aufgelistet.

Gerät	Hersteller	Typenbezeichnung	Seriennummer
IEPE-Sensor	MMF	KS 77C.10	08026
Impedanzmesskopf	Brüel & Kjær	8001	1492165
Ladungsverstärker	Brüel & Kjær	Nexus 2692	2078245
Oszilloskop	Tektronix	MSO 4054	C001028
Signalanalysator	Agilent	35670A	MY42507855
Kalibriershaker	Brüel & Kjær	4294	1670988
Funktionsgenerator	Agilent	33220A	MY44049491
Rauschgenerator	MDMT	–	–
Leistungsverstärker	Brüel & Kjær	2706	638608
Shaker	Brüel & Kjær	4809	630915

Tabelle 4.22: Im Rahmen der Messungen verwendete Geräte

4.5.5.1 IEPE-Interface exklusive Sensor

Der Versuchsaufbau zur Bestimmung des Hochpass- und Verstärker-Frequenzganges, entsprechend Abbildung 4.66, ist in Abbildung 4.68 dargestellt. Als Testsignal diente rosa Rauschen. Da die Schaltung für die Verarbeitung von unipolaren Signalen ausgelegt ist, musste das bipolare Rauschsignal mittels eines auf einem Steckbrett aufgebauten Pegelwandlers in den zulässigen Eingangsbereich angehoben werden. Während der Messungen wurde die Konstantstromquelle des IEPE-Interfaces elektrisch von der Eingangsbuchse abgetrennt. Die Bestimmung des Übertragungsverhaltens erfolgte mit einem Signalanalysator von Agilent Technologies. Abbildung 4.69 zeigt die Ergebnisse dieser Messungen. Der Amplitudengang weist im betrachteten Frequenzbereich oberhalb der unteren Grenzfrequenz das gewünschte frequenzunabhängige Verhalten auf, wobei die Verstärkungen mit den in Tabelle 4.21 angegebenen Werten übereinstimmen. Die Phasenverschiebung im betrachteten Bereich ist vernachlässigbar gering, wobei der Phasengang bei $S_D = 1$ und $S_G = 1$ auf Grund numerischer Effekte bei der Auswertung einen Wert von 360° annimmt. Die tatsächliche Phasenverschiebung beträgt hier ebenfalls 0° .

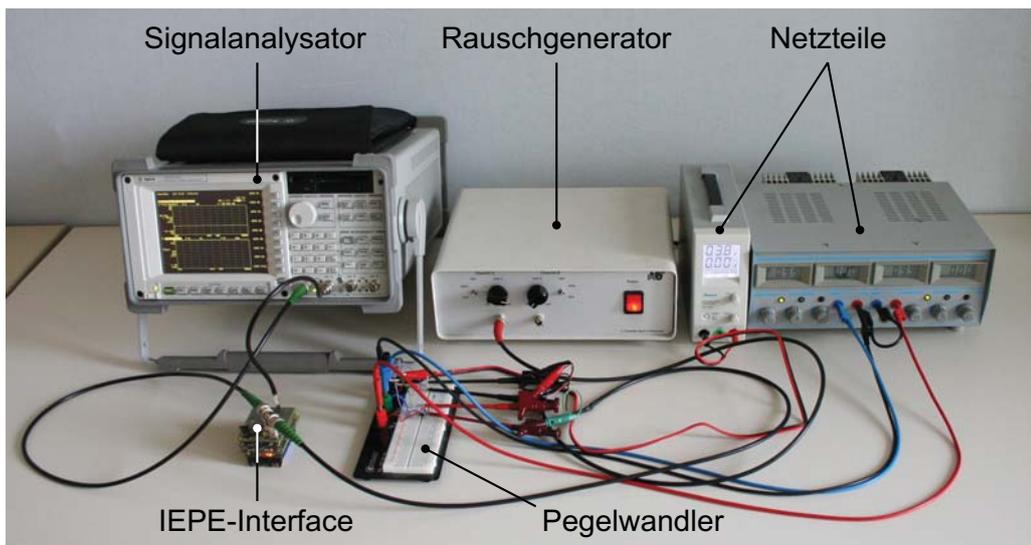


Abbildung 4.68: Versuchsaufbau zur Bestimmung der Frequenzgänge des IEPE-Interfaces exklusive Sensor

Die Bestimmung der unteren Grenzfrequenzen ist auf Grund der sehr geringen Werte mittels des beschriebenen Messaufbaus nicht praktikabel. Hierfür wurde mittels eines Funktionsgenerators ein sinusförmiges Eingangssignal generiert und die Amplitudenabschwächung im Vergleich zum Durchlassbereich des Hochpasses mit Hilfe eines Oszilloskops bestimmt. Konkret dienten die Verstärkungen bei 10 Hz als Referenz für die Über-

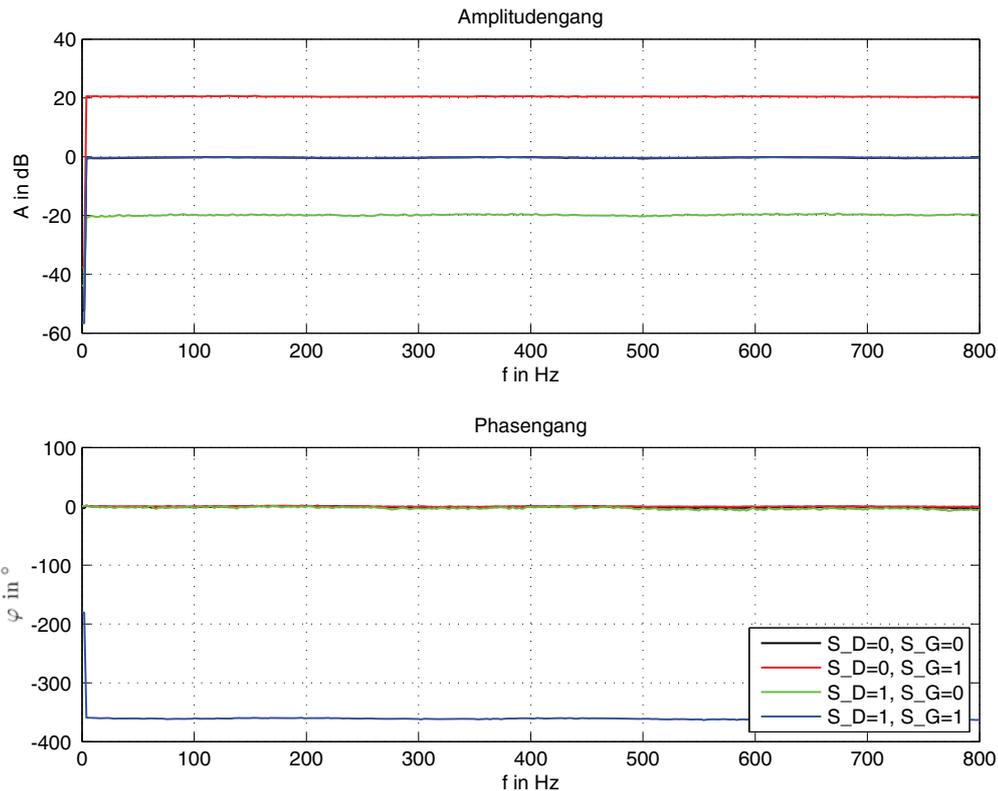


Abbildung 4.69: Frequenzgänge des IEPE-Sensorinterfaces exklusive Sensor

prüfung der unteren Grenzfrequenz. Die Ergebnisse dieser Messungen (siehe Tabelle 4.23) bestätigten die in Abschnitt 4.5.4 prognostizierten Werte für die unteren Grenzfrequenzen der Schaltung.

S_D	S_G	Verstärkung	Frequenz in Hz	A/A_{ref} in dB
0	0	1.0	0.030	-3.2
0	1	10.0	0.045	-3.0
1	0	0.1	0.030	-3.2
1	1	1.0	0.045	-3.0

Tabelle 4.23: Bestimmung der unteren Grenzfrequenzen des IEPE-Interfaces

4.5.5.2 IEPE-Interface inklusive Sensor

Die Überprüfung der Funktionalität der gesamten Schaltung mit dem Sensor KS77C-10 der Firma Metra Mess- und Frequenztechnik erfolgte mit Hilfe des in Abbildung 4.70 dargestellten Versuchsaufbaus. Das Übertragungsverhalten wurde relativ zu einer Refe-

renzmesskette von Brüel & Kjær bestimmt. Die Erregung der miteinander verschraubten Sensoren erfolgte mittels eines Shakers, der wiederum mit rosa Rauschen angesteuert wurde.

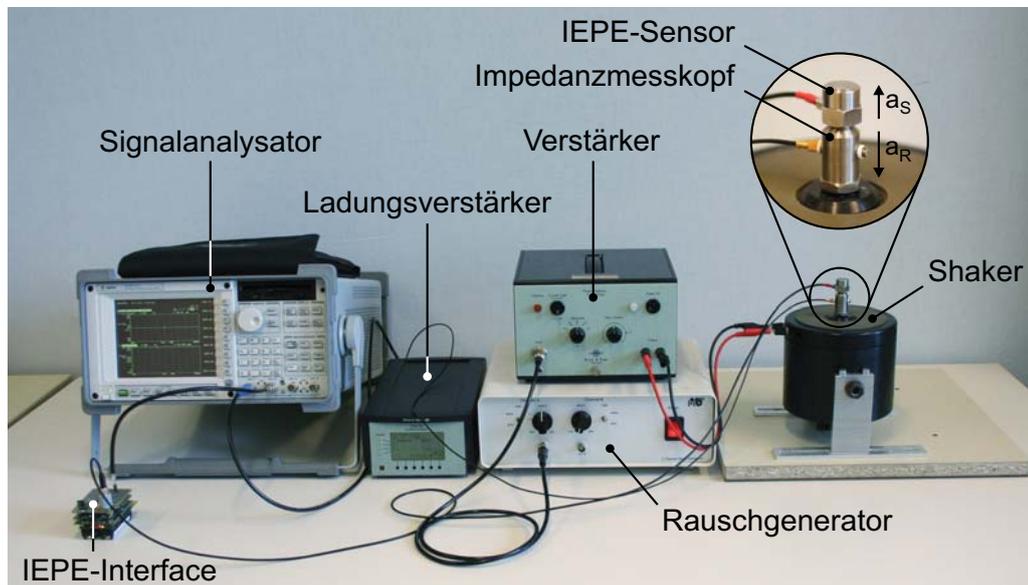


Abbildung 4.70: Versuchsaufbau zur Bestimmung der Frequenzgänge des IEPE-Interfaces inklusive Sensor

Temperaturspannungen im Gehäuse des Impedanzmesskopfs Brüel & Kjær Typ 8001, die bereits in geschlossenen Räumen durch kaum wahrnehmbare Luftbewegungen induziert werden, verursachen bei ruhendem Sensor einen nicht vernachlässigbaren Offsetdrift. Abbildung 4.71 oben zeigt exemplarisch den Zeitverlauf des Ausgangssignal der Referenzmesskette in einem Büroraum am Institut für Mechanik und Mechatronik bei unbewegtem Beschleunigungsaufnehmer. Die Ausgangsspannung wurde für die Darstellung bereits in die entsprechende physikalische Einheit der Messgröße umgerechnet. Das Ausgangssignal ist bei einer unteren Grenzfrequenz von 0.1 bzw. 1 Hz, die im Setup des Ladungsverstärkers Nexus 2692 gewählt werden kann, dargestellt. Wie deutlich ersichtlich ist, kann der niederfrequente, durch inhomogene Temperaturverteilungen im Gehäuse des Sensors verursachte, Signalanteil durch eine Hochpassfilterung des Signals bei einer gewählten unteren Grenzfrequenz von 1 Hz wirkungsvoll unterdrückt werden. Die Messung des Frequenzganges erfolgte daher mit einem dahingehend konfigurierten Ladungsverstärker.

Zur Untermauerung der vermuteten Ursache des Offsetdrifts wurde eine 350 °C warme Lötspitze für 5 s in einem Abstand von 1 cm zum Impedanzmesskopf gehalten und der Verlauf der Ausgangsspannung der Referenzmesskette (untere Grenzfrequenz 0.1 Hz) auf-

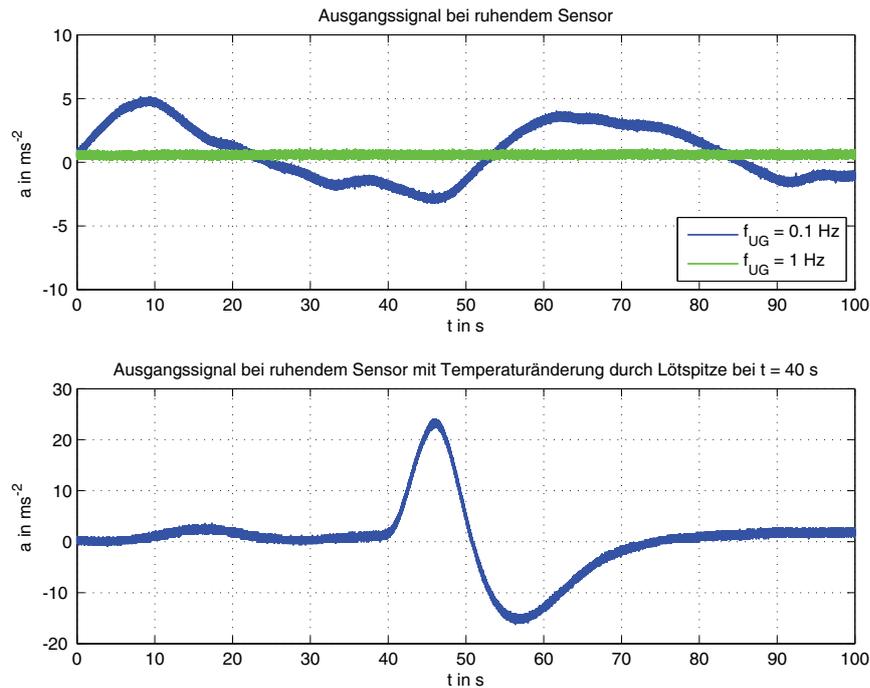


Abbildung 4.71: Ausgangssignal der Referenzmesskette auf Grund von temperturbedingten mechanischen Spannungen

gezeichnet. Das Ergebnis des Versuchs ist in Abbildung 4.71 unten dargestellt und zeigt einen deutlichen Einfluss der berührungslos von der Lötspitze auf den Sensor übertragenen Wärme auf das Ausgangssignal der Messkette.

Die Ergebnisse der eigentlichen Messung des Übertragungsverhaltens des IEPE-Interfaces in Relation zur Referenzmesskette sind in Abbildung 4.72 dargestellt. Die Empfindlichkeit der Referenzmesskette betrug bei den Messungen $10 \text{ mV}/(\text{m}/\text{s}^{-2})$. Mit der Empfindlichkeit des IEPE-Sensors $E_{IEPE} = 1.039 \text{ mV}/(\text{m}/\text{s}^{-2})$ (siehe Abbildung F.3) und den Verstärkungen 0.1, 1 und 10 des Interfaces erhält man in Relation zur Referenzmesskette im Durchlassbereich die Verstärkungen -39.7 , -19.7 und 0.3 dB . Diese konnten durch die Messungen bestätigt werden. Werte des Amplituden- und Phasengangs unterhalb von 1 Hz sind auf Grund des dahingehend im Ladungsverstärker konfigurierten Hochpasses nicht aussagekräftig. Die gemessene Phasenverschiebung von $\pm 180^\circ$ resultiert aus den, wie in Abbildung 4.68 ersichtlich, zueinander entgegengesetzt ausgerichteten Achsen der Beschleunigungssensoren.

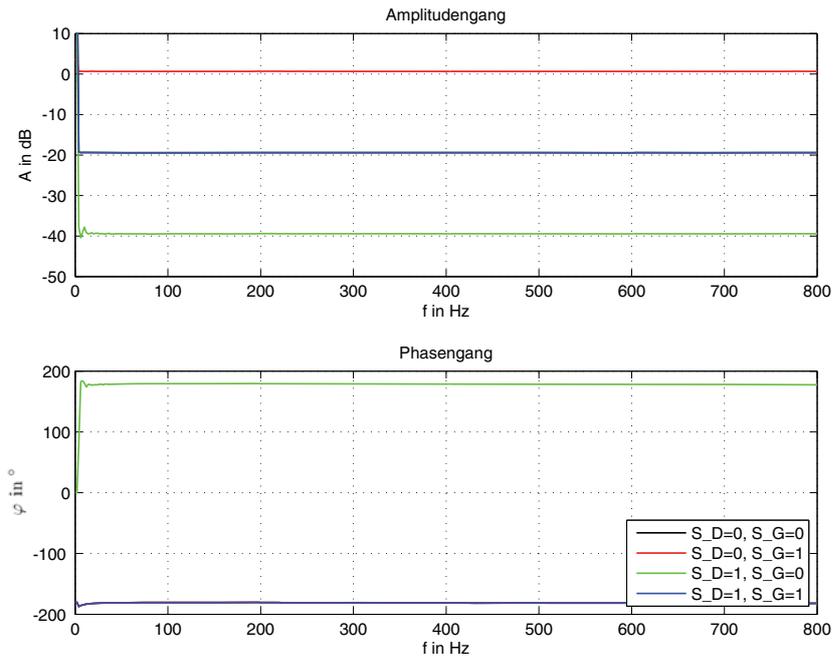


Abbildung 4.72: Frequenzgänge des IEPE-Sensorinterfaces inklusive Sensor KS77C.10 von MMF (Referenz: Messkette von Brüel & Kjær)

4.5.6 Strombedarf

Der Strombedarf der einzelnen Funktionsblöcke des IEPE-Interfaces ist in Tabelle 4.24 zusammengefasst. Anhand der Zahlen ist ersichtlich, dass der wesentliche Anteil der Leistung zur Versorgung des IEPE-Sensors benötigt wird. Durch die Verwendung von Low-Power-IEPE-Sensoren, wie sie z.B. von Metra Mess- und Frequenztechnik angeboten werden, ließe sich der Strombedarf des Moduls bzw. in weiterer Folge des gesamten Messsystems deutlich reduzieren. Typischerweise werden solche Sensoren mit Strömen ab 0.1 mA bei Spannungen von 9 bis 12 V betrieben. Die Versorgung derartiger Sensoren ist mittels des derzeit eingesetzten DC/DC-Konverters TSP61081 prinzipiell möglich, erfordert jedoch eine entsprechende Anpassung dieses Schaltungsteils in Form einer Modifikation von speziell ausgelegten Widerstands- und Kondensatorwerten.

Verbraucher	+24 V	+2.5 V	-2.5 V
Stromquelle	4 mA	–	–
Spannungsteiler	0.32 mA	–	–
Verstärker	–	0.54 mA	-0.54 mA

Tabelle 4.24: Strombedarf der Teilschaltungen des IEPE-Interfaces

4.5.7 Entwicklungsstufen

Im Zuge der Entwicklung des IEPE-Interfaces entstanden drei Prototypen, die in Abbildung 4.73 dargestellt sind. Die Energieversorgung des Sensors erfolgte im Rahmen des ersten Prototyps mittels des TPS61040 von Texas Instruments, wobei bei dieser Version des Interfaces noch keine zusätzliche Filterung der Versorgungsspannung vorgesehen war. Da die Schaltfrequenz des TPS61040 vom Laststrom abhängig ist, tritt bei niedrigen Lasten – wie im konkreten Anwendungsfall – eine erhöhte Welligkeit der Ausgangsspannung auf. Als Gegenmaßnahme wurde für den zweiten Prototypen daher der TPS61081 mit fixer Schaltfrequenz in Kombination mit einem zusätzlichen LC-Filter implementiert. Der Hochpass- und Verstärkerteil des Interfaces entspricht bei diesen beiden Prototypen Abbildung 4.65. Die aktuelle Ausführung des Prototyp 3 ist in Abbildung 4.66 dargestellt. Die übrige Elektronik ist bei allen Prototypen identisch und entspricht den hier dokumentierten Schaltbildern.

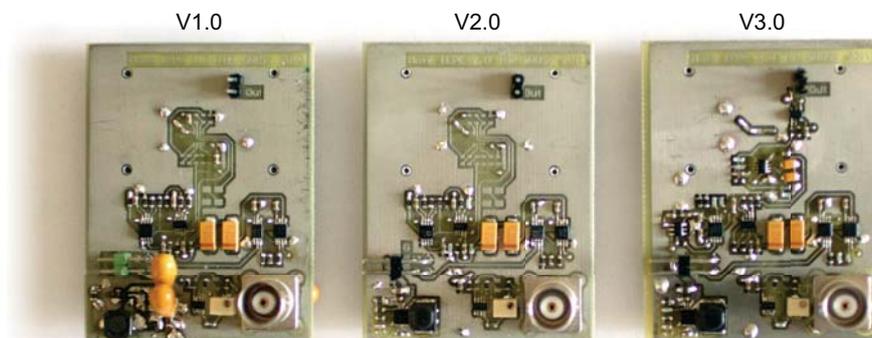


Abbildung 4.73: Prototypen des IEPE-Interfaces

4.5.8 IEPE-Interface V3.1

Bei näherer Betrachtung von Abbildung 4.66 ist ersichtlich, dass bei dieser Version die Verstärkung 1 im Vorwärtszweig durch einen 10:1-Spannungsteiler und einen nachgeschalteten Verstärker mit der Verstärkung 10 erzielt wird. Diese ungünstige Konfiguration kann unter Einsparung des durch R_{15} und R_{16} gebildeten Spannungsteilers, wie in Abbildung 4.74 dargestellt, vermieden werden. Eine diesbezügliche Layoutänderung des IEPE-Interfaces ist bisher aus Zeitgründen nicht erfolgt, diese sollte jedoch spätestens bei einer Verwirklichung des Messsystems in Betracht gezogen werden.

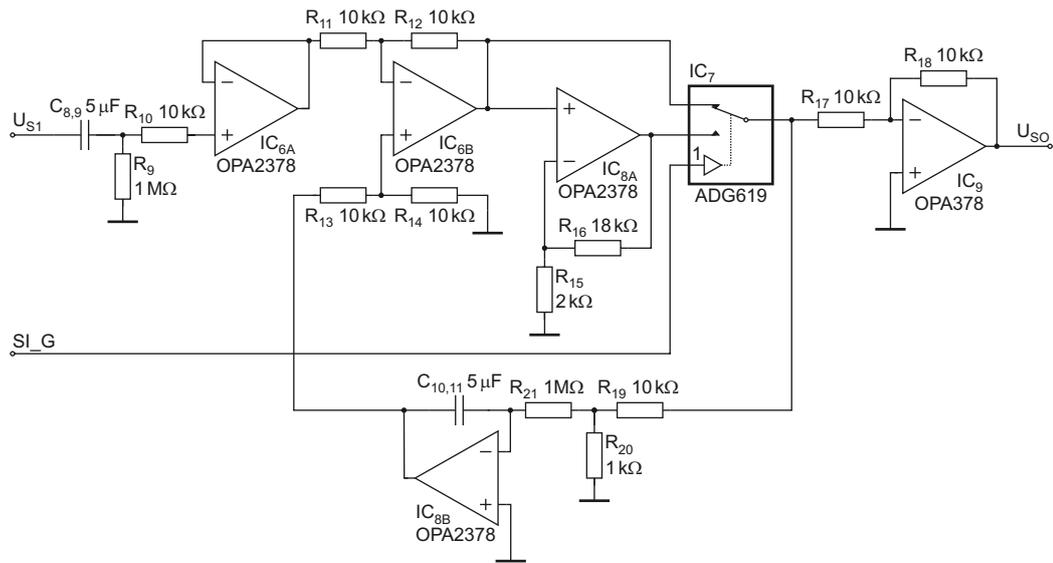


Abbildung 4.74: Schaltbild des Hochpasses und Verstärkers – IEPE-Interface V3.1

4.6 USB-Interface

Die Steuerung des Messsystems soll auf einfache und komfortable Weise über ein PC-basiertes GUI erfolgen. Die hierfür notwendige Anbindung des Messsystems an den PC oder Laptop erfolgt mittels USB. Der MSP430F2618 verfügt lediglich über die Standardschnittstellen RS232, SPI (Serial Peripheral Interface) und I2C (Inter Integrated Circuit), weshalb eine direkte Anbindung an einen modernen PC/Laptop nicht möglich ist. Es wird daher ein USB-Interface benötigt, das den Datenaustausch zwischen dem USB und einer der genannten Schnittstellen ermöglicht. Auf dem Markt ist eine Vielzahl von ICs mit dieser Funktionalität verfügbar, wobei der FT232R von Future Technology Devices Incorporated (FTDI) eine sehr einfache Möglichkeit zur Herstellung dieser Verbindung zwischen einem PC/Laptop und einem Basismodul, entsprechend Abbildung 4.75, bietet. Nach Installation des FT232R-Gerätetreibers steht am PC/Laptop eine virtuelle RS232-Schnittstelle zur Verfügung, auf die z.B. mittels eines Terminal- oder LabVIEW-Programms zugegriffen werden kann.

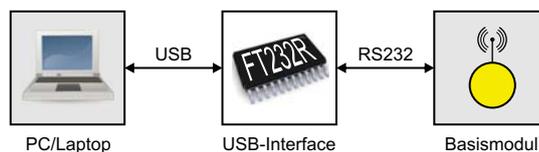


Abbildung 4.75: Anbindung des Basismoduls an einen PC/Laptop

Ein Foto sowie ein Blockschaltbild des USB-Interfaces sind in den Abbildungen 4.76 und 4.77 dargestellt. Das Modul besteht im Wesentlichen aus dem FT232R sowie einem Mikrocontroller (MSP430F2618), der über Steckverbinder mit dem Basismodul verbunden wird. Die Kommunikation zwischen dem MSP430 des USB-Interfaces und dem FT232R erfolgt mittels RS232-Interface. Neben den Receive- und Transmit-Leitungen FT_RXD und FT_TXD verfügt die Schnittstelle des FT232R (siehe Abbildung 4.78) ebenfalls über die Handshaking-Leitungen FT_RTSn und FT_CTSn sowie die Leitungen FT_RIn, und FT_CBUS0 bis FT_CBUS4. FT_RIn kann bei entsprechender Konfiguration des FT232R dazu benutzt werden um den USB-Host-Controller vom Suspend-Zustand in den aktiven Zustand zu schalten. Abhängig von der Konfiguration des FT232R können die FT_BUS-Signale verschiedene Statussignale ausgeben oder als GPIOs verwendet werden. Diesbezügliche Details sind dem Datenblatt [25] zu entnehmen.

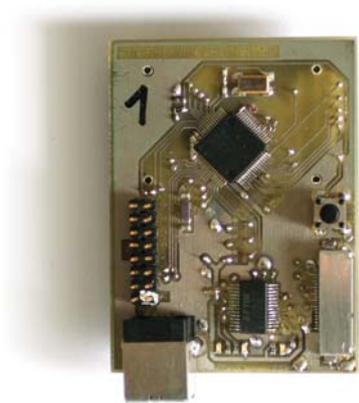


Abbildung 4.76: Foto des USB-Interfaces

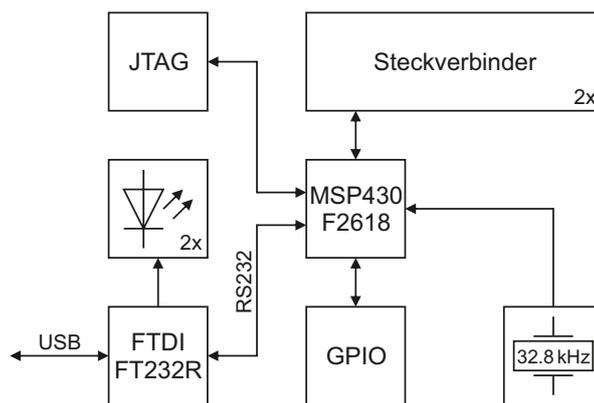


Abbildung 4.77: Blockschaltbild des USB-Interfaces

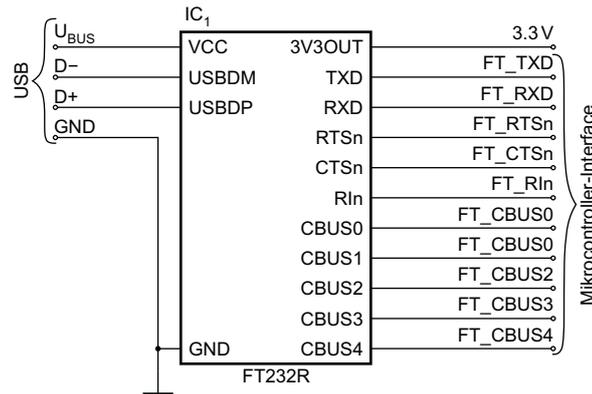


Abbildung 4.78: Schaltbild des FT232R

Die Signale FT_CBUS0 und FT_CBUS1 sind im Auslieferungszustand des FT232R so konfiguriert, dass diese die Busaktivität anzeigen. Die beiden Pins sind daher mit jeweils einer LED verbunden um das Senden und Empfangen von Daten sichtbar zu machen. Der MSP430 ist so programmiert, dass FT_RXD und FT_TXD direkt an das Basismodul weitergeleitet werden. Bei Bedarf kann jedoch beispielsweise eine Zwischenspeicherung der Daten oder eine Weiterleitung der Daten an das Basismodul mittels SPI-Interface durch eine entsprechende Änderung der Firmware erreicht werden. Ein kompletter Port des MSP430 ist über eine Stiftleiste zugänglich, wobei diese IOs jedoch nicht in Verwendung sind. Der vorhandene Uhrenquarz ist derzeit für die Funktion des Moduls ebenfalls nicht erforderlich. Wie in Abbildung 4.78 ersichtlich, stellt der FT232R eine, von U_{BUS} abgeleitete, +3.3 V-Versorgungsspannung bei einem Strom von bis zu 50 mA zur Verfügung. Das gesamte Submodul wird daher über USB mit Energie versorgt. Ein angeschlossenes Basismodul muss jedoch nach wie vor über eine eigene Energiequelle (Batterien oder Akkumulatoren) verfügen.

Die Bedienung des Laborsystems erfolgte während der gesamten Entwicklung mittels des Terminalprogramms Docklight (Screenshot siehe Abbildung 4.79), das die Definition von RS232-Paketen erlaubt. Diese können bei Bedarf mittels einer zugehörigen Schaltfläche durch einen Mausklick gesendet werden. Weiters bietet Docklight die Möglichkeit den gesamten Datenverkehr in einer Log-Datei zu speichern. Diese Funktion war insbesondere bei den Tests des Laborsystems vorteilhaft.

Die Entwicklung des LabVIEW-basierten GUIs konnte im Rahmen der für das Projekt zur Verfügung stehenden Zeit nicht mehr abgeschlossen werden. Ein Screenshot der Benutzeroberfläche für erste Vortests ist in Abbildung 4.80 dargestellt.

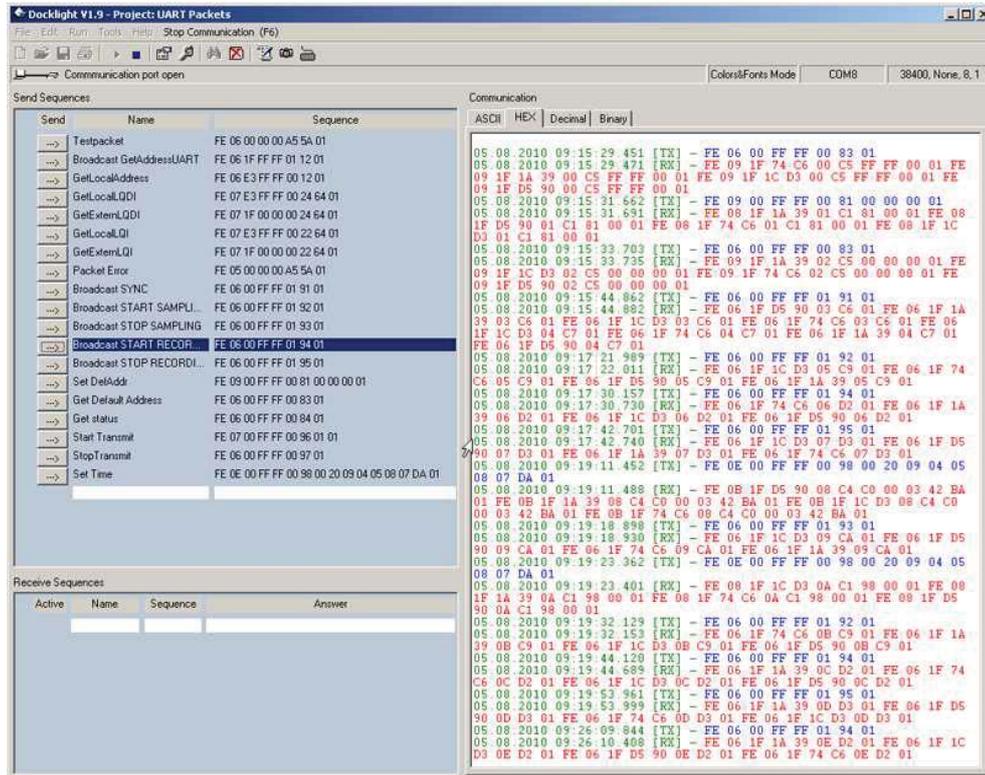


Abbildung 4.79: Terminalprogramm Docklight

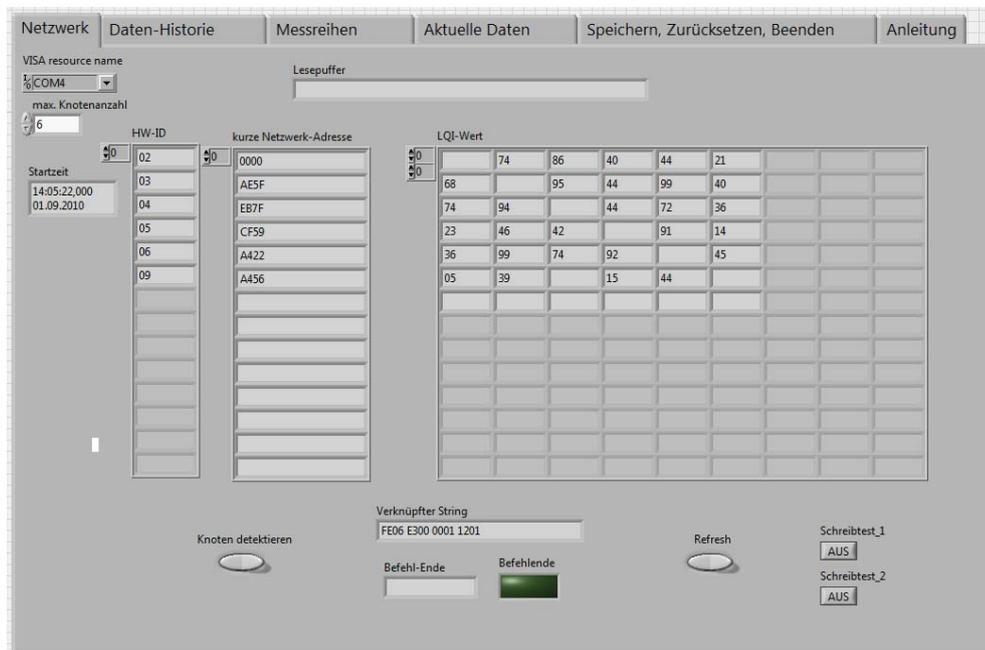


Abbildung 4.80: LabVIEW-GUIs

4.7 Funktionsgenerator (DDS-Modul)

Der Funktionsgenerator dient zur Erzeugung genau definierter Signale (Signalform, Frequenz, Amplitude und Offset), die die Beurteilung der Qualität der vom Datenspeicher aufgezeichneten Daten erlauben. Unter Verwendung des DDS-Funktionsgenerators (Direct Digital Synthesis) AD9833 von Analog Devices wurde ein Modul entwickelt, das auf den Datenspeicher aufgesteckt werden kann und als Signalquelle für eine Testfahrt (siehe Abschnitt 4.8) diene.

Der AD9833 deckt einen Frequenzbereich von 0 bis 12.5 MHz ab und kann die Signalformen Sinus und Dreieck erzeugen. Die Vorgabe der Signalparameter erfolgt mittels SPI-Interface, weshalb auf dem Modul ein Mikrocontroller für die Konfiguration der Signalparameter integriert wurde. Auf Grund der auf 0.65 V limitierten Ausgangsspannung des AD9833 wurde diesem ein Verstärker nachgeschaltet, um den Eingangsbereich des im Datenspeicher (vergleiche Abschnitt 4.3) enthaltenen ADCs möglichst gut abzudecken. Ein Blockschaltbild der Signalgenerierung ist in Abbildung 4.81 dargestellt.

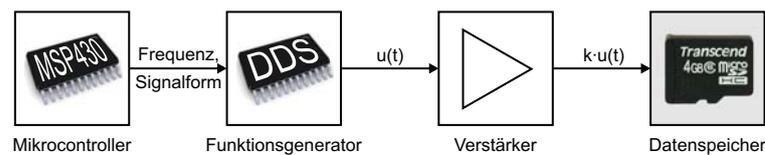


Abbildung 4.81: Blockschaltbild der Signalgenerierung des Funktionsgenerators

Ein Foto sowie ein Blockschaltbild des DDS-Moduls sind in Abbildung 4.82 bzw. 4.83 dargestellt. Die Konfiguration des AD9833 erfolgt mittels eines MSP430F2132. Dieser überprüft nach einem Reset des Prozessors (z.B. durch Betätigung des Reset-Tasters), welche der Konfigurationsjumper gesetzt wurden und sendet die entsprechenden Datenpakete zur Festlegung von Frequenz und Signalform über die SPI-Schnittstelle an den AD9833. Der ebenfalls vorhandene FT232R (Beschreibung siehe Abschnitt 4.6) kann nach einer diesbezüglichen Erweiterung der Firmware des MSP430 ebenfalls zur Konfiguration der Signalparameter herangezogen werden. Da dies im Rahmen dieses Projekts jedoch nicht erforderlich war, bleibt diese Option ungenützt. Zum Betrieb der internen Logik benötigt der AD9833 ein externes Taktsignal mit einer Frequenz von 25 MHz, das durch einen Oszillator erzeugt wird. Entsprechend Abbildung 4.84 wird der Ausgangspegel des AD9833 mittels eines nichtinvertierenden Verstärkers mit $A = 3.35$ auf eine maximale Spannung von 2.18 V angehoben. Zusätzlich zum verstärkten Ausgangssignal U_{FO} des AD9833 wird mittels eines Spannungsteilers, gefolgt von einem Spannungsfolger auch eine

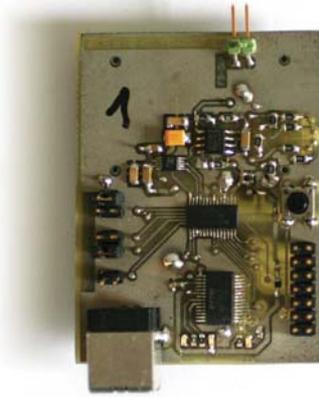


Abbildung 4.82: Foto des DDS-Moduls

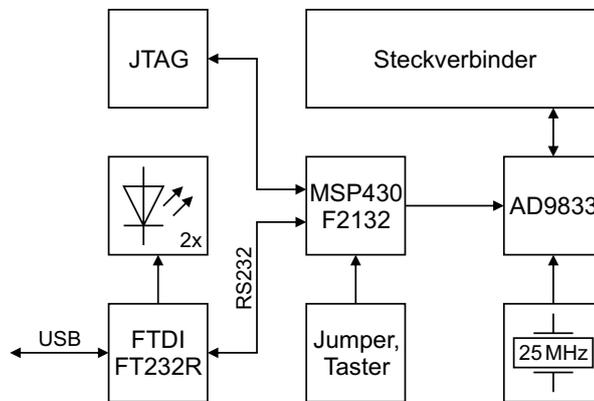


Abbildung 4.83: Blockschaltbild der Komponenten des DDS-Moduls

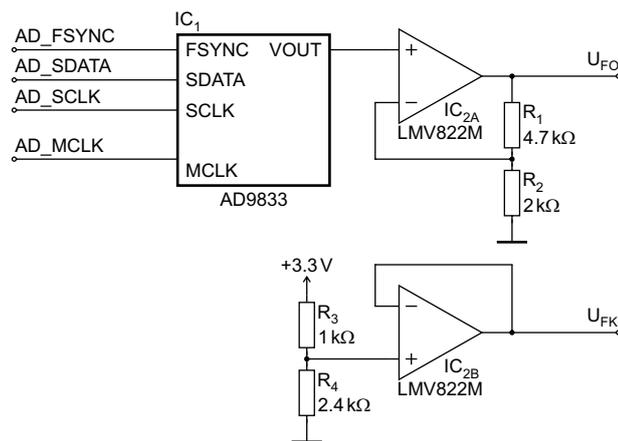


Abbildung 4.84: Schaltbild der Signalerzeugung

konstante Spannung $U_{FK} = 2.33 \text{ V}$ für den Qualitätstest der Signalaufzeichnung erzeugt. Die Spannungen U_{FO} und U_{FK} werden schließlich über die Steckverbinder zum Datenspeichermodul weitergeleitet.

Die Ausgangsspannungen U_{FO} des DDS-Moduls für alle implementierten Kombinationsmöglichkeiten hinsichtlich Signalform und Frequenz sind in Abbildung 4.85 dargestellt.

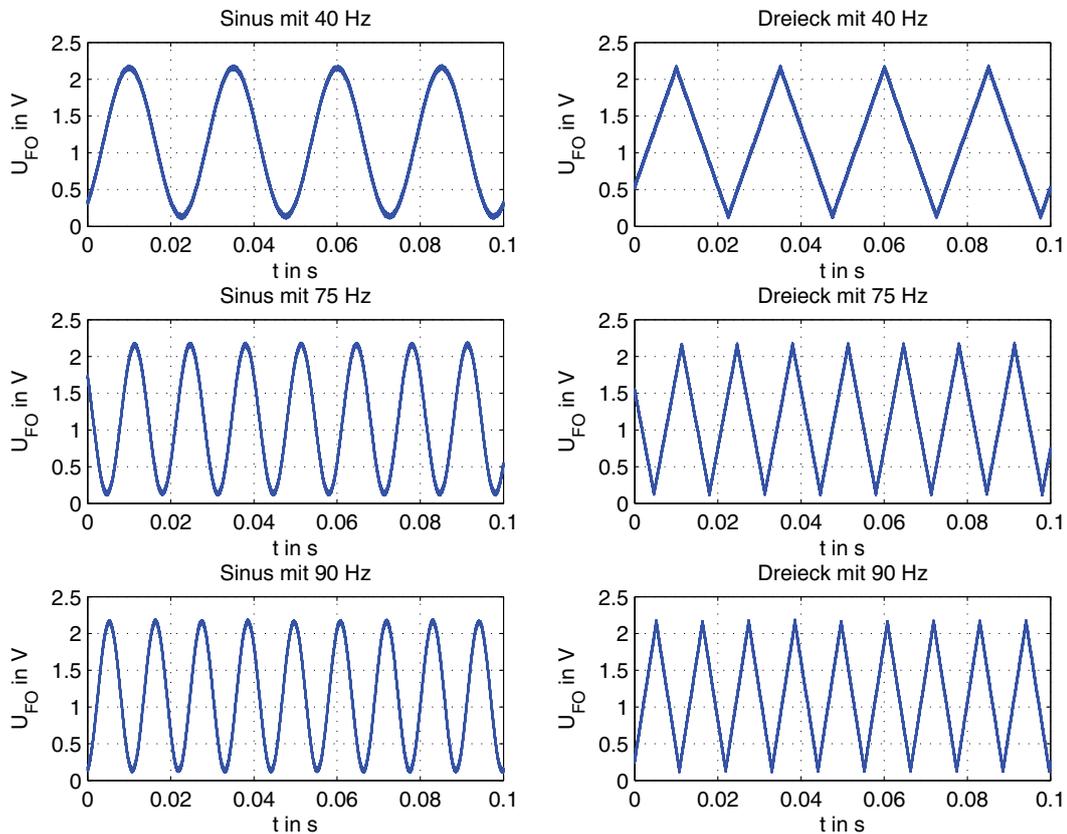


Abbildung 4.85: Ausgangsspannungen U_{FO} des DDS-Moduls

4.8 Testfahrt

Am 2.9.2010 fanden auf der Teststrecke am BTA-Werksgelände in Wien (siehe Abbildung 4.86) mit einer Flexity Outlook C für Palermo Störsicherheitstests mit den Basis- und Datenspeichermodulen des Laborsystems statt. Die Knoten wurden in einem Abstand von etwa 5.6 m – dies entspricht der mittleren Distanz laut Default-Messstellenplan der Firma Bombardier – entlang des Fahrzeuges verteilt. Bedient wurde das System mittels eines, mit dem Koordinator (Knoten 2) per USB verbundenen, Laptops. Die Steuerbefehle wurden mit Hilfe des Terminalprogramms Docklight abgesetzt und die empfangenen Daten in einer

Log-Datei gespeichert. Vier Knoten waren mit Datenspeichern ausgerüstet (Knoten 3, 4, 6 und 9), wobei drei Knoten (Knoten 3, 6 und 9) zusätzlich mit einem Funktionsgenerator (DDS-Modul) bestückt waren. Die Funktionsgeneratoren erzeugten Dreieckssignale mit unterschiedlichen Frequenzen. Die Konfiguration bzw. der Aufbau der einzelnen Knoten kann Tabelle 4.25 entnommen werden.



Abbildung 4.86: BTA-Werksgelände 1220 Wien, Hermann Gebauer Straße 5 (Datenquelle: Stadt Wien – data.wien.gv.at)

Parameter↓\Knoten→	2	3	4	5	6	9
ZigBee Device Type	Koord.	Router	Router	Router	Router	Router
Funkkanal	15	15	15	15	15	15
USB-Modul (Nr.)	2	–	–	–	–	–
Datenspeicher (Nr.)	–	1	4	–	2	3
DDS-Modul (Nr.)	–	1	–	–	2	3
Signalform	–	Dreieck	–	–	Dreieck	Dreieck
Frequenz in Hz	–	90	–	–	75	40

Tabelle 4.25: Konfiguration der Knoten während der Tests

Die Messungen sollten Aufschlüsse über die Robustheit des Funknetzes hinsichtlich elektro-magnetischer Störungen in der späteren Betriebsumgebung (Straßenbahn) geben. Insbesondere die Einflüsse von

- elektromagnetischen Störungen bei maximalen Beschleunigungen (Starten und Stoppen),
- Störungen durch Zusatzaggregate (z.B. Klimaanlage) und
- den Betrieb von Bluetoothsendern innerhalb der Straßenbahn

sollten untersucht werden. Hierfür wurden mehrere Messungen sowohl im Stillstand als auch mit Fahrtanteilen durchgeführt, die im Folgenden näher beschrieben werden.

4.8.1 Messungen im Stillstand

Um Referenzwerte für die Beurteilung der Auswirkungen der verschiedenen potentiellen Störquellen zu erhalten wurden zu Beginn Messungen im ruhenden Zug ohne aktive Zusatzaggregate oder Sender durchgeführt. Die Positionen der Systemknoten und deren jeweiligen Netzwerkadressen sind in Abbildung 4.87 dargestellt. Die Tests umfassten die Feststellung der Verbindungsqualität der Knoten untereinander, die Live-Übertragung von Messkanälen über Funk, sowie die Aufzeichnung der Messdaten auf microSD-Karten. Als Signalquellen für die Messungen dienten die Funktionsgeneratormodule.

4.8.1.1 Verbindungsqualität

Zur Feststellung der Verbindungsqualität wurde ausgehend von jedem Knoten der vom ZigBee-Stack gelieferte LQI-Wert der Nachbarknoten zehn Mal abgefragt. Die Ergebnisse sind in Abbildung 4.88 zusammengefasst (Zeile = Quellknoten – SRC, Spalte = Antwortknoten – DST). Neben den LQI-Werten (oberer Wert – je höher der Wert, desto besser ist die Verbindungsqualität) ist ebenfalls der Prozentsatz der erhaltenen Antworten (70 % = 7 Antworten von 10 Anfragen) angegeben.

Die Ergebnisse zeigen, dass eine gute Verbindungsqualität nicht nur zwischen unmittelbar benachbarten Knoten sondern auch unter Auslassung eines Zwischenknotens (z.B. von Knoten 2 zu Knoten 3) vorhanden war. In dieser Konfiguration hätte der Ausfall eines Knotens keinen Einfluss auf die Funktion des verbleibenden Systems.

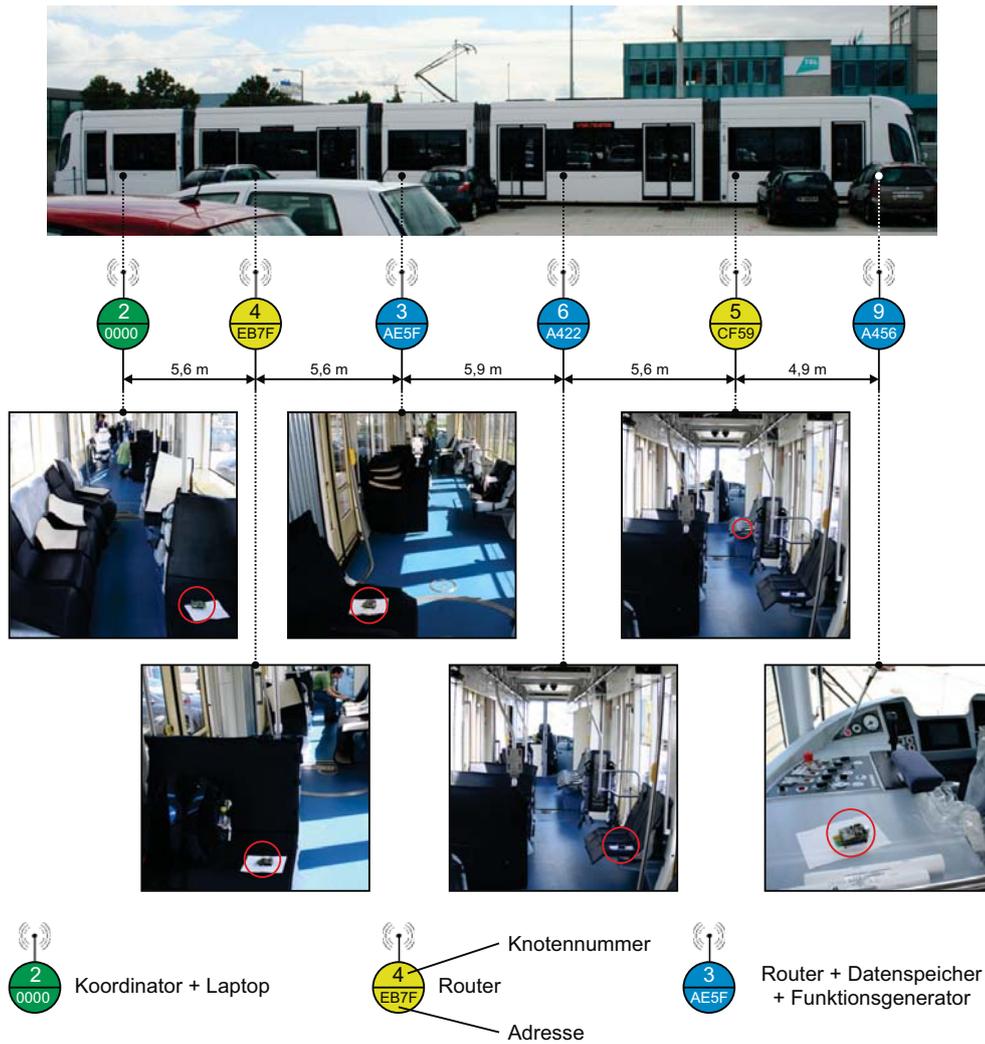


Abbildung 4.87: Position der Systemknoten (Messung im Stillstand)

DST ↓ SRC	2	4	3	6	5	9
2		86 100%	74 100%	44 100%	40 100%	21 100%
4	74 80%		94 80%	72 100%	44 100%	36 50%
3	68 100%	95 90%		99 90%	44 100%	40 40%
6	36 80%	74 70%	99 90%		92 90%	45 90%
5	23 90%	42 70%	46 80%	91 80%		14 40%
9	15 70%	- 0%	39 90%	44 100%	15 100%	

Abbildung 4.88: Verbindungsqualität zwischen den einzelnen Knoten

4.8.1.2 Übertragung eines ADC-Kanals per Funk

Abbildung 4.89 zeigt die per Funk übertragenen Signale der Knoten 3, 6 und 9. Zum Vergleich ist zusätzlich das mittels Oszilloskop (Tektronix MSO 4054) abgetastete Signal dargestellt.

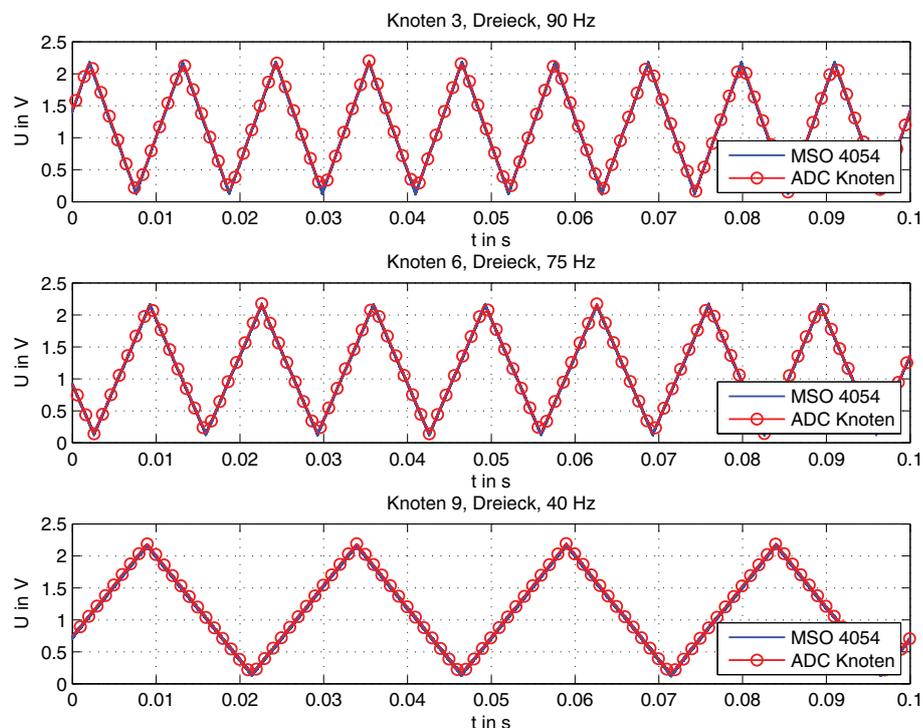


Abbildung 4.89: Per Funk übertragene ADC-Kanäle der Knoten 3, 6 und 9

Die Auswertung der Log-Files hat die in Tabelle 4.26 zusammengefassten Werte hinsichtlich der verlorenen Funkpakete ergeben.

Ergebnis↓\Knoten→	3	6	9
Übertragungsdauer in s	280	308	281
Paketverlust in %	0	0	0.11

Tabelle 4.26: Paketverlust im Stillstand

Trotz einer Messdauer von über vier Minuten traten bei den Übertragungen von den Knoten 3 und 6 keine Paketverluste auf. Lediglich beim weitest entfernten Knoten 9 betrug, auf Grund einer neuen Routenwahl im Netzwerk, der Paketverlust 0.11 %.

4.8.1.3 Lokal aufgezeichnete Messdaten

Drei ADC-Kanäle (ADC0, ADC3 und ADC6) des Datenspeicher-Moduls zeichneten das Dreiecksignal des Funktionsgenerators auf, zwei (ADC1 und ADC5) einen konstanten Analogwert (Quelle: Spannungsteiler) und drei weitere (ADC2, ADC4 und ADC7) ein ebenfalls konstantes Signal mit 0 V (Signalmasse). Ein Ausschnitt dieser Aufzeichnung des Knotens 9 (Datenspeichermodul DM03) ist in Abbildung 4.90 dargestellt.

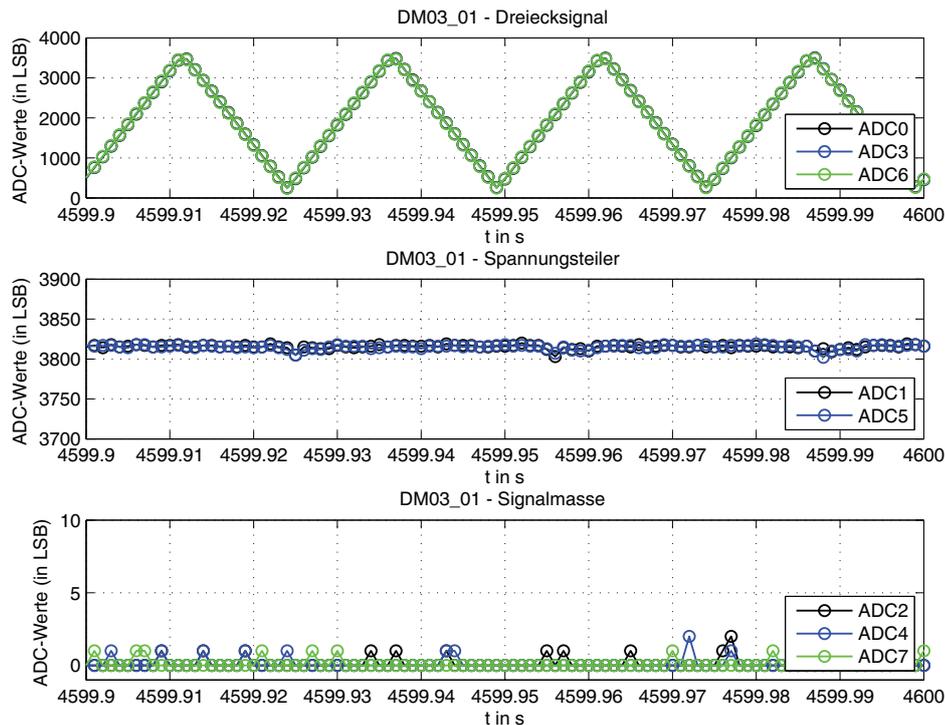


Abbildung 4.90: Im Stillstand aufgezeichnete Messdaten des Knotens 9 (Ausschnitt)

Die Aufzeichnungsdauer betrug jeweils 19 min. Die Auswertung der mit dem Signal abgespeicherten Zeitmarken ergab keine Lücken bzw. Ausfälle.

Die Ergebnisse der Digitalisierung zeigt eine sehr hohe Qualität. Bei vorgegebenen Eingangswerten liegen die Variationen bei typischen ein bis zwei LSB (siehe Abbildung 4.90).

4.8.2 Messungen mit Fahrtanteilen

Die Knoten 5 und 6 wurden für diese Messungen im Bereich hinter den oberen, aufklappbaren Abdeckungen (Vouten) mit Klebeband fixiert um möglichst realistische Ergebnisse zu erzielen (siehe Abbildung 4.91).

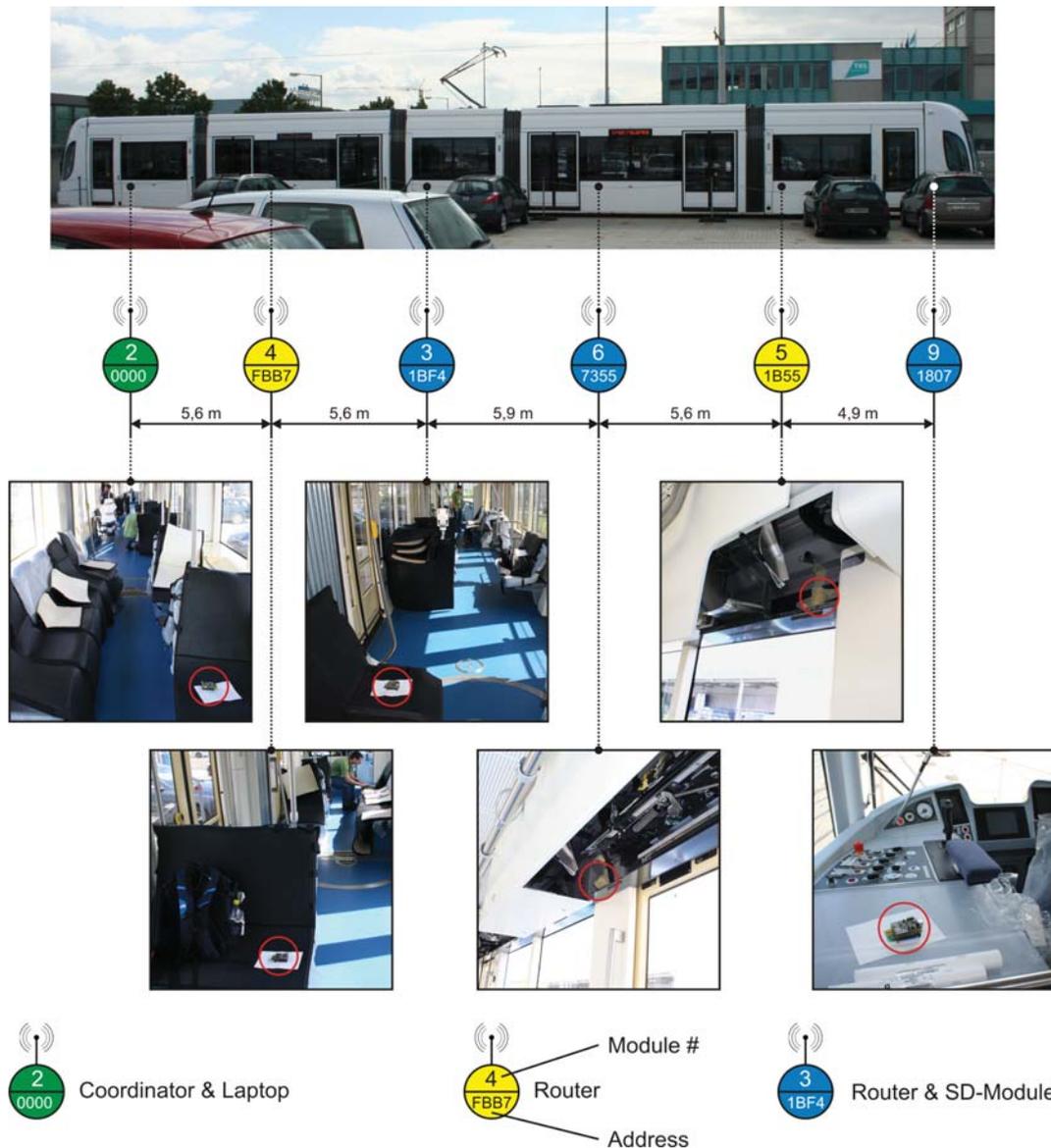


Abbildung 4.91: Position der Systemknoten (Messung mit Fahrtanteilen)

Bei den Fahrten wurde speziell auf eine möglichst hohe Startbeschleunigung und Bremsverzögerung Wert gelegt, da hierbei die größten elektromagnetischen Störungen zu erwarten waren.

Zum Test der Robustheit des Systems hinsichtlich des Einflusses von Bluetoothsendern wurden zusätzlich im Bereich des Knotens 6 Fotos von einem Handy auf einen PDA übertragen (siehe Abbildung 4.92).

Es wurden drei Messungen mit Fahrtanteilen und verschiedenen potentiellen Störquellen durchgeführt:



Abbildung 4.92: Bluetoothübertragung in der Nähe des Knotens 6

Messung 1 Zwei gefahrene Längen ohne Klimaanlage

Messung 2 Zwei Längen ohne Klimaanlage und zwei Längen mit Klimaanlage

Messung 3 Zwei Längen mit Klimaanlage und Bluetooth

Bei allen Messungen wurden die mitprotokollierten Live-Daten des Moduls 9, das den größten Abstand zum Steuerknoten 2 aufwies, sowie die aufgezeichneten Daten aller mit Funktionsgeneratoren ausgestatteten Knoten ausgewertet.

4.8.2.1 Übertragung eines ADC-Kanals per Funk

Die Auswertung der Messungen 1 bis 3 ergab für die Übertragung des ADC-Kanals vom Knoten 9 die in Tabelle 4.27 zusammengefassten Paketverluste.

Ergebnis↓\Messung→	1	2	3
Übertragungsdauer in s	548	512	134
Paketverlust in %	0.44	0	0.18

Tabelle 4.27: Paketverlust während der Fahrt

4.8.2.2 Aufgezeichnete Messdaten

Die Analyse der, auf microSD-Karten aufgezeichneten, Messdaten ergab bei allen drei Messungen weder Lücken noch Ausfälle. Abbildung 4.93 zeigt einen Ausschnitt der vom Knoten 9 während der Messung 3 aufgezeichneten Daten.

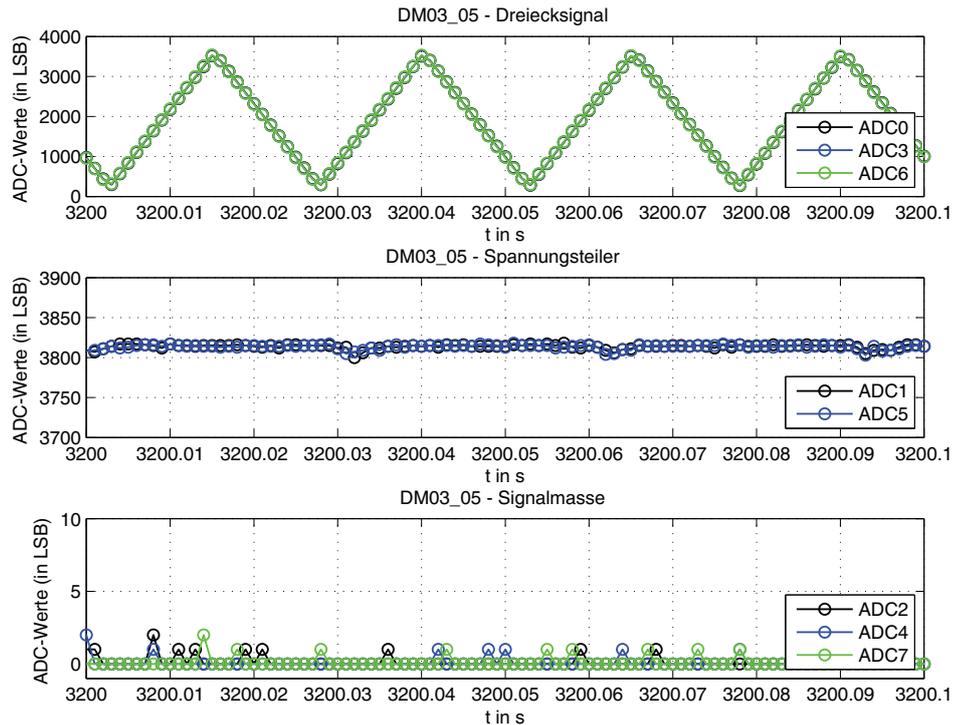


Abbildung 4.93: Während der Fahrt aufgezeichnete Messdaten des Knotens 9 (Ausschnitt – Messung 3)

4.8.3 Zusammenfassung

Zusammengefasst ergeben sich folgende Resultate für die durchgeführten Tests:

- Keine Lücken oder Ausfälle bei der Aufzeichnung auf die SD-Karten.
- Gute Verbindungsqualität bei Entfernungen bis ca. 11 m zwischen zwei Knoten.
- Maximal 0.44 % der Pakete gingen während der Übertragung eines Funkkanals verloren.

Die Analyse der Log-Dateien und der per Funk übertragenen Messdaten zeigt, dass in realer Betriebsumgebung durch elektromagnetische Störungen oder Funkübertragungen im selben Frequenzbereich (z.B. WLAN, Bluetooth) kurze Lücken auftreten können. Dies hat jedoch keinerlei Einfluss auf die am Knoten aufgezeichneten Daten. Die Funktion des Basismoduls und des Datenspeichers konnte damit unter realen Umgebungsbedingungen nachgewiesen werden.

5 Mesh-DAQ

Die, im Rahmen des Laborsystems (siehe Kapitel 4), beschriebenen Schaltungen können im Wesentlichen für das geplante Messsystem – im Folgenden als Mesh-DAQ bezeichnet – unverändert übernommen werden. Auf Grund der Begrenzung des Laborsystems auf lediglich einen Messkanal und der Versorgung mittels AA-Zellen sind jedoch eine zusätzliche Steuerlogik sowie eine abweichende Energieversorgung vorzusehen ist. In weiterer Folge wird daher in Abschnitt 5.1 ein Konzept zur Verwirklichung der Steuerlogik mittels eines CPLDs und in Abschnitt 5.2 der Entwurf eines entsprechenden Energieversorgungssystems vorgestellt. Weiters wird kurz auf eine Möglichkeit zu Kompensation der RX-/TX-Latenzzeiten des Z-Stacks (Abschnitt 5.3) eingegangen.

5.1 Steuerlogik

Um eine möglichst hohe Flexibilität und Skalierbarkeit hinsichtlich der ausgeführten Kanalanzahl der allgemeinen Signalaufbereitung zu erzielen, wurde für die Generierung der Steuersignale eine Kombination aus peripheren Logikbausteinen, die für jeden einzelnen Kanal vorzusehen sind, und einem zentralen Complex Programmable Logic Device (CPLD, Xilinx Coolrunner 2) gewählt. Die Grundstruktur der geplanten Logik ist in Abbildung 5.1 in Form eines Blockschaltbildes dargestellt. Diese Abbildung zeigt die externen Blöcke sowie die zugehörigen Steuersignale für den generischen Messkanal i . Entsprechend der tatsächlich ausgeführten Anzahl an Kanälen sind diese mehrfach vorhanden. Zur Speicherung von zusätzlichen Informationen wie z.B. Kalibrierdaten, Typ des Sensorinterfaces, etc. sind für die allgemeine Signalaufbereitung, sowie auf jedem steckbaren Sensorinterfacemodul, Flash-Speicher vorgesehen.

Das Interface zwischen Mikrocontroller und CPLD besteht einerseits aus einer Konfigurationsschnittstelle bestehend aus Daten- und Adressbus inklusive Taktsignal sowie zugehörigen Steuerleitungen und andererseits aus Steuer- und Kommunikationssignalen, die entsprechend der momentanen Konfiguration des CPLDs mittels der implementierten Logik zum jeweiligen Zielbaustein des selektierten Kanals weitergeleitet werden. Die

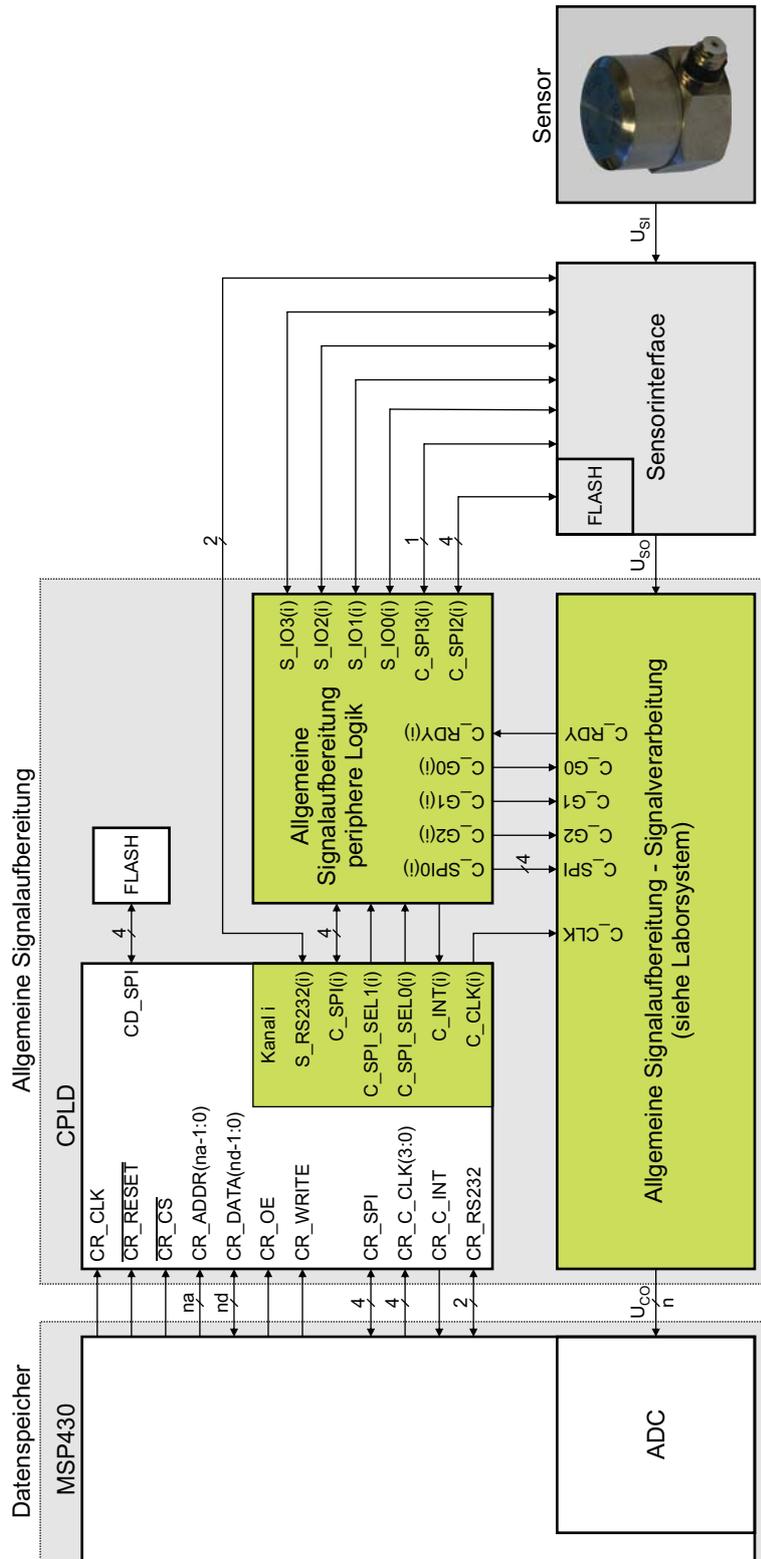


Abbildung 5.1: Blockschaltbild der Steuerlogik (dargestellt für einen generischen Messkanal i)

Signal	Beschreibung
CR_CLK	Taktsignal der synchronen Logik ($f_{CR_CLK} = 25$ MHz)
CR_RESETh	Asynchroner Reset der CPLD-Logik (active-low)
CR_CS _n	Chip-Select (active-low)
CR_ADDR	Adressbus
CR_DATA	Datenbus (bidirektionaler Bus)
CR_OE	Output Enable zur Aktivierung der Ausgangstreiber des Datenbusses (CR_OE = low → hochohmiger Zustand)
CR_WRITE	Write-Signal (Übernahme der Daten in den Konfigurationsspeicher bei CR_WRITE = low)

Tabelle 5.1: Signale der Konfigurationsschnittstelle des CPLDs

Signal	Beschreibung
CR_SPI	SPI-Bus bestehend aus den Signalen CR_SPI_CLK, CR_SPI_CS _n , CR_SPI_MOSI und CR_SPI_MISO
CR_C_CLK	Taktsignale zur Ansteuerung der SC-Filter (das Taktsignal für jeden einzelnen Kanal kann mittels der CPLD-Logik ausgewählt werden)
CR_C_INT	Interrupt-Leitung des Port-Extenders (siehe Abschnitt 5.1.1)
CR_RS232	RS232-Interface bestehend aus CR_TXD und CR_RXD (kann optional, durch Änderung der CPLD-Firmware, auch als GPIO verwendet werden)

Tabelle 5.2: Steuer- und Kommunikationssignale

Signale dieses Interfaces sind in den Tabellen 5.1 und 5.2 zusammengefasst.

Mit Hilfe der CPLD-internen Logik werden abhängig vom aktuellen Konfigurationszustand die Steuerausgänge C_SPI_SEL_x der Kanäle gesetzt bzw. die Kommunikationsleitungen C_SPI, S_RS232 und C_INT sowie die Taktsignale der SC-Filter C_CLK entsprechend weitergeleitet. Der SPI-Bus CR_SPI kann bei entsprechender Konfiguration des CPLDs ebenfalls zum Ansprechen des seriellen Flash-Speichers der allgemeinen Signalaufbereitung verwendet werden. Eine detaillierte Beschreibung der CPLD-Logik bietet Abschnitt 5.1.2.

Durch die periphere Logik werden jeweils pro Kanal mit Hilfe eines Port-Extenders mit SPI-Schnittstelle zusätzliche IOs zur Ansteuerung der analogen Signalaufbereitung entsprechend Abschnitt 4.4 sowie eines Sensorinterfaces (S_IO0 bis S_IO3) zur Verfügung gestellt. Weiters stehen der SPI-Bus inklusive zweier Chip-Select-Leitungen (C_SPI2 und C_SPI3) und die serielle Schnittstelle S_RS232 zur Kommunikation mit dem Sensorinterface zur Verfügung, wobei C_SPI2 für das Auslesen der auf dem Flash-Speicher des Sensorinterfaces vorhandenen Informationen vorgesehen ist. Somit stehen ein SPI-

und RS232-Interface zur Kommunikation mit, auf zukünftigen Sensorinterfaces eventuell vorhandenen, ICs zur Verfügung. Mittels RS232-Kommunikation könnte beispielsweise ein GPS-Modul angesprochen werden und die GPS-Positionsbestimmung, demzufolge mittels eines steckbaren Moduls, ausgeführt werden.

5.1.1 Periphere Logik

Obwohl sich die periphere Logik hinsichtlich des vom Mikrocontroller ausgehenden Signalfusses nach der CPLD-Logik befindet, wird die Beschreibung der peripheren Logik an dieser Stelle vorgezogen, da für die Entwicklung der CPLD-Firmware die Kenntnis der vorhandenen Hardware zwingend erforderlich ist und dies somit auch dem eigentlichen Entwicklungsprozess entspricht.

Abbildung 5.2 zeigt das vereinfachte Schaltbild der peripheren Logik eines generischen Kanals (grün hinterlegt) sowie des seriellen Flash-Speichers M25P80, der für die Kalibrierdaten aller Kanäle des Signalaufbereitungsmoduls vorgesehen ist und daher nur ein einziges Mal erforderlich ist.

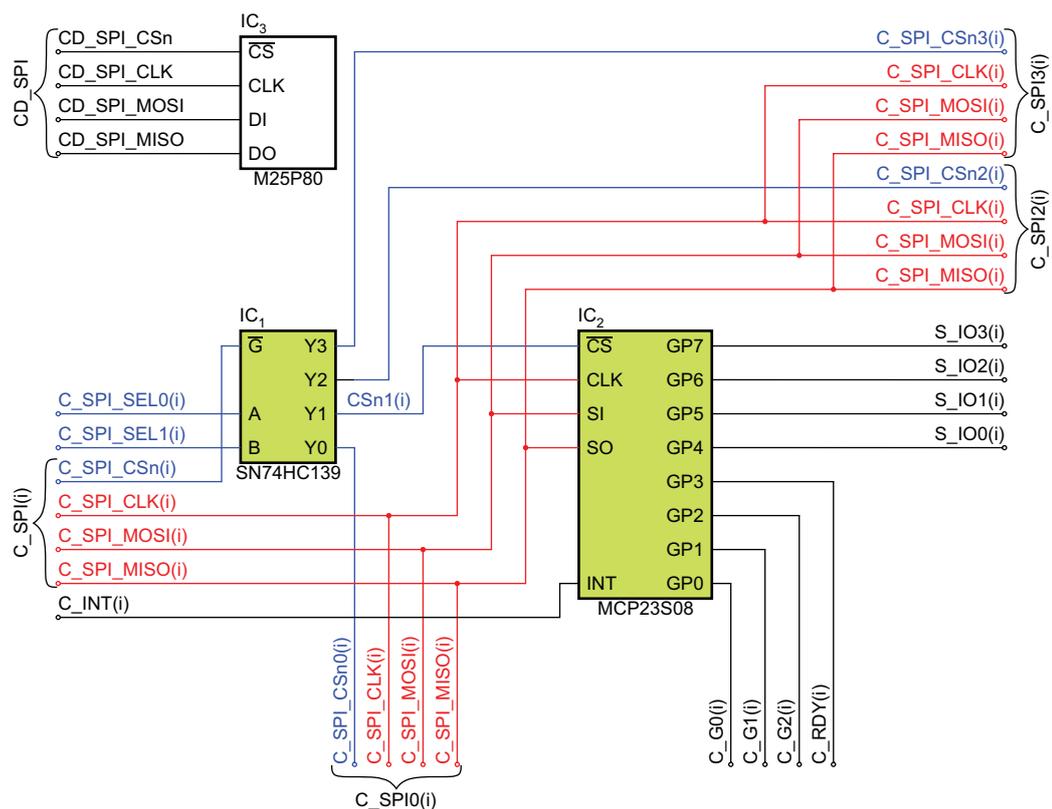


Abbildung 5.2: Schaltbild der peripheren Logik eines generischen Kanals i

Die einzelnen SPI-Anschlüsse des Moduls C_SPI0 bis C_SPI3 bestehen aus den, durch die CPLD-Logik entsprechend der momentanen Konfiguration weitergeleiteten, Signalen C_SPI_CLK und C_SPI_MOSI, den Ausgangssignalen des jeweils angeschlossenen Bausteins C_SPI_MISO, sowie dem, mittels des Demultiplexers SN74HC139 aus den Signalen C_SPI_CS_n, C_SPI_SEL0 und C_SPI_SEL1 generierten, Chip-Select-Signalen C_SPI_CS_{n0} bis C_SPI_CS_{n3}. Die Ausführung unter Verwendung eines diskreten Logikbausteins ist in diesem Fall nicht zwingend erforderlich, reduziert allerdings die benötigte Anzahl an Signalleitungen von vier (C_SPI_CS_{n0} bis C_SPI_CS_{n3}) auf drei (C_SPI_CS_n, C_SPI_SEL0 und C_SPI_SEL1) und kann bei einer hohen Kanalanzahl auf Grund der beschränkten Anzahl von zur Verfügung stehenden CPLD-IOs notwendig sein. Der SN74HC139 besitzt zwei identische Kanäle, wobei in Abbildung 5.2 lediglich ein Kanal dargestellt ist. Durch eine entsprechende Verschaltung dieser beiden Kanäle können durch die Verwendung von vier Steuersignalen (C_SPI_SEL0 bis C_SPI_SEL3) bei Bedarf sieben Chip-Select-Signale generiert werden. Bei der derzeit geplanten Ausführung mit vier, über die SPI-Schnittstelle anzusprechenden ICs wird der Demultiplexer jedoch sinnvollerweise direkt in die CPLD-Logik zu verlagern sein.

Da zur Ansteuerung der analogen Signalverarbeitung drei Aus- und ein Eingang (C_G0, C_G1, C_G2 sowie C_RDY) und zumindest zwei weitere IOs für das IEPE-Interface (siehe Abschnitt 4.5) benötigt werden, wurde der Port-Extender MCP23S08 mit SPI-Interface vorgesehen, womit durch die Verwendung des ohnehin benötigten SPI-Interfaces eine deutliche Reduktion der benötigten CPLD-IOs und der notwendigen Leiterbahnen gegeben ist. Mittels C_INT kann bei Bedarf durch die Verwendung der entsprechenden Funktionen im Port-Extender und im Mikrocontroller über Interrupts auf Zustandsänderungen der digitalen Eingänge des Port-Extendens reagiert werden. Angesteuert wird der Port-Extender mittels C_SPI1.

In Summe ergeben sich pro Kanal für die Ansteuerung des Sensorinterfaces mit der beschriebenen Logik folgende Anschlussmöglichkeiten:

- Zwei SPI-Interfaces, wobei eines für den Anschluss eines Flash-Speichers auf dem Sensorinterface vorgesehen ist,
- ein RS232-Interface und
- vier IOs.

Je nach den konkreten Anforderungen kann bei einer ausreichenden Anzahl an freien CPLD-IOs und abhängig von den verfügbaren CPLD-Ressourcen bei Bedarf ein Teil oder die gesamte periphere Logik in das CPLD selbst verlagert werden.

5.1.2 Zentrale CPLD-Logik

Abbildung 5.3 zeigt ein vereinfachtes Prinzipschaltbild der CPLD-Logik. Die Logik des CPLDs besteht im Wesentlichen aus einem Konfigurationsspeicher, der aus einzelnen Registern besteht. Es kann sowohl schreibend als auch lesend auf diesen Speicher zugegriffen werden. Ist ENABLE high ($CR_WRITE = \text{high}$ und $CR_CSn = \text{low}$) so wird CR_DATA an der entsprechenden Adresse CR_ADDR gespeichert. Die aktuelle Konfiguration kann durch die Verwendung von Tri-State-Ausgängen bei Aktivierung der Ausgangstreiber mittels $CR_OE = \text{high}$ (bei $CR_CSn = \text{low}$) über die Datenleitungen CR_DATA ausgelesen werden.

Die SPI- und RS232-Kommunikationsleitungen des MSP430-CPLD-Interfaces werden mittels Demultiplexer (CR_SPI_CSn , CR_SPI_CLK , CR_SPI_MOSI und CR_TXD) bzw. Multiplexer (CR_SPI_MISO und CR_RXD), die im Prinzipschaltbild zur Erhaltung der Übersichtlichkeit der Grafik zusammengefasst als Demultiplexer dargestellt sind, entsprechend den Werten im Konfigurationsregister (CD_SPI_SEL , SPI_CH und $RS232_CH$) weitergeleitet. Zusätzlich wird das C_INT -Signal des mittels SPI_CH gewählten Kanals an den Mikrocontroller weitergegeben. Die Registerinhalte SPI_SEL0 und SPI_SEL1 dienen der Generierung der Chip-Select-Signale des SPI-Busses durch die periphere Logik (siehe Abschnitt 5.1.1) und werden ebenfalls mittels eines Demultiplexers zum jeweils gewählten Kanal weitergeleitet. Die Ausgangspegel der nicht aktiven bzw. momentan nicht selektierten Ausgänge befinden sich im Zustand low, wobei die CSn -Leitungen hier eine Ausnahme bilden, da es sich um active-low-Signale handelt. Sie befinden sich demzufolge im high-Zustand.

Mittels CLK_SEL kann für jeden Kanal eines der vier CR_C_CLK -Signale als Taktsignal für den entsprechenden SC-Filter der analogen Signalaufbereitung gewählt werden.

5.1.2.1 Funktionsbelegung des Konfigurationsspeichers

Die derzeitige Implementierung der CPLD-Logik besitzt acht Bit breite Datenworte bei einem Adressbereich von $0x0$ bis $0x5$. Es sind jedoch nur die Addressleitungen $CR_ADDR(0)$ bis $CR_ADDR(3)$ in Verwendung, $CR_ADDR(4)$ ist in der aktuellen Version der Firmware ohne Funktion. Die Belegung sowie die Beschreibung der Funktionen der einzelnen Register sind in den Tabellen 5.3 und 5.4 zusammengefasst. Eine Erweiterung der CPLD-Logik von derzeit vier auf beispielsweise acht unterstützte Kanäle ist mit geringem Aufwand möglich.

Die Konfiguration der SPI- bzw. RS232-Multiplexer und -Demultiplexer erfolgt mittels

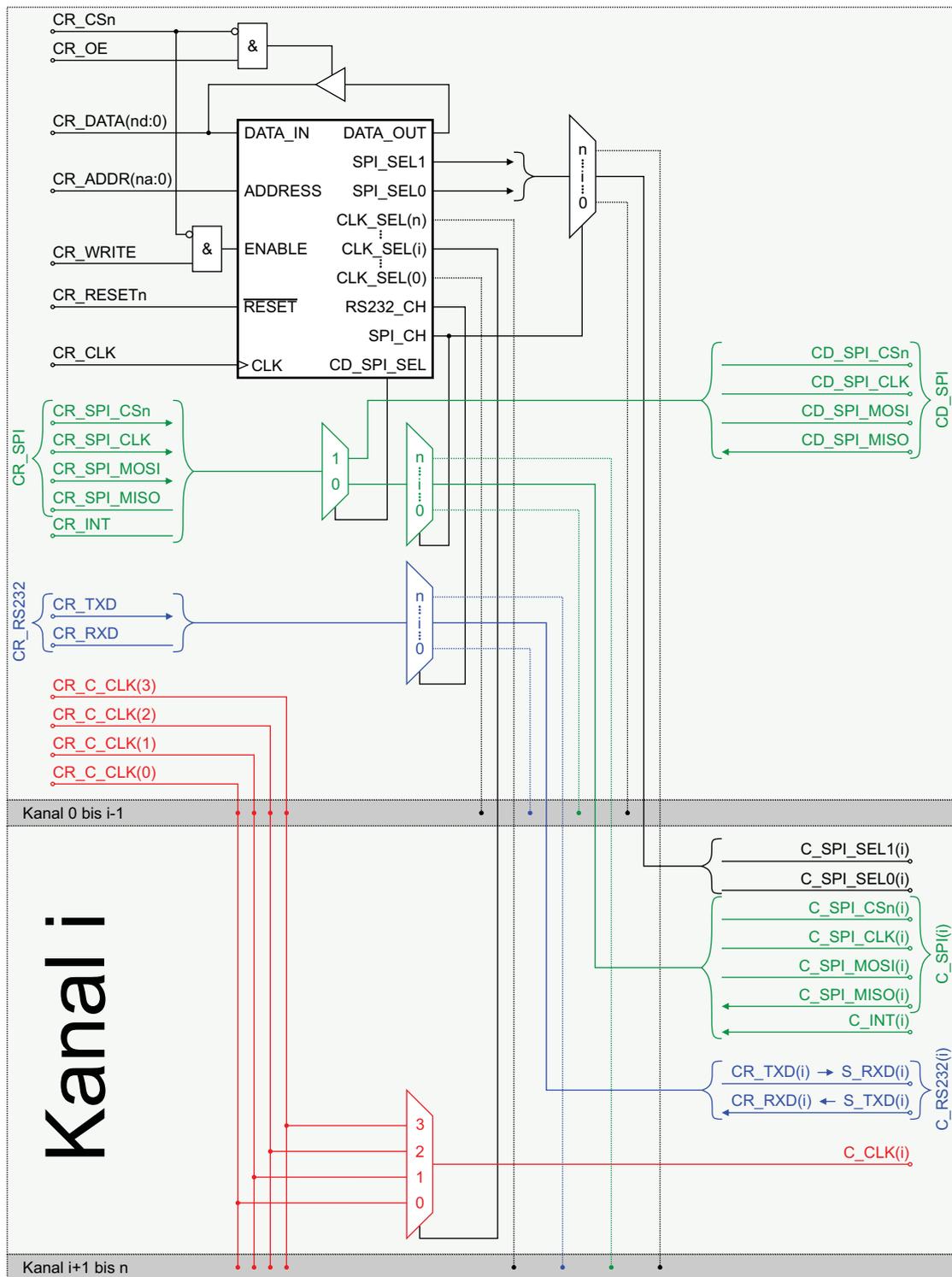


Abbildung 5.3: Vereinfachtes Prinzipschaltbild der CPLD-Logik

Addr.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0	SC1	SC0	SS1	SS0	–	–	–	CDSS
0x1	CEN(0)	CS1(0)	CS0(0)	–	–	–	–	–
0x2	CEN(1)	CS1(1)	CS0(1)	–	–	–	–	–
0x3	CEN(2)	CS1(2)	CS0(2)	–	–	–	–	–
0x4	CEN(3)	CS1(3)	CS0(3)	–	–	–	–	–
0x5	RSEN	RSC1	RSC0	–	–	–	–	–

Tabelle 5.3: Registerbelegung des Konfigurationsspeichers

Register	Signal	Beschreibung
SCx	SPI_CH	Wählt den Kanal für die SPI-Kommunikation (aktiver Kanal $i = SC1 \cdot 2 + SC0$)
SSx	SPI_SELx	Wählt den aktiven Baustein mittels Chip-Select (aktiver Kanal $i = SS1 \cdot 2 + SS0$)
CDSS	CD_SPI_SEL	Bei CDSS = 0b1 wird das mit CD_SPI verbundene serielle Flash angesprochen
CEN(i)	–	Aktiviert (CEN(i) = 0b1) bzw. deaktiviert das C_CLK- Signal des jeweiligen Kanals
CSx(i)	CLK_SEL(i)	Wählt das auszugebende CR_C_CLK Signal aus (aktives Taktsignal $k(i) = CS1(i) \cdot 2 + CS0(i)$)
RSEN	–	Aktiviert die RS232-Ausgänge des CPLDs
RSCx	–	Wählt den Kanal für die RS232-Kommunikation (aktiver Kanal $i = RSC1 \cdot 2 + RSC0$)

Tabelle 5.4: Beschreibung der einzelnen Register

der Register bei Adresse 0x0 bzw. 0x5. Die Adressen 0x1 bis 0x4 betreffen jeweils die Konfiguration der C_CLK(i) des in Klammer angeführten Kanals (0x1 betrifft Kanal 0, 0x2 Kanal 1, usw.). Zusammengehörige Register wie z.B. SC1 und SC0 werden in Tabelle 5.4 gemeinsam unter der Bezeichnung SCx behandelt. Die Ziffer nach der jeweiligen Buchstabenkombination gibt die Wertigkeit des Bits an, wobei 0 das LSB kennzeichnet. Bei identischem Aufbau der Register, wie im Fall der Konfiguration der C_CLK-Signale durch CEN und CS1 und CS2 für die Kanäle 0 bis 3 wird lediglich der generische Fall CEN(i) bzw. CSx(i) beschrieben. In der Spalte Signal wird zusätzlich die Signalbezeichnung entsprechend Abbildung 5.3 angeführt.

5.1.2.2 Simulation der CPLD-Logik

Die Firmware des CPLDs wurde in VHDL mittels der Xilinx ISE Design Suite 11.3 erstellt, wobei auf den Code selbst aus Platzgründen an dieser Stelle nicht näher eingegangen

werden kann. Zur Überprüfung der implementierten Funktionalität wurde ebenfalls eine entsprechende Test-Bench erstellt. Auszugsweise sind zur Illustration der Funktionsweise der Logik in den Abbildungen 5.4, 5.5 und 5.6 die Signalverläufe beim Schreiben und Lesen des Konfigurationsspeichers, beim Umschalten zwischen SPI-Kanälen sowie beim Umschalten zwischen CR_C_CLK-Signalen dokumentiert.

Vor dem ersten Schreiben in den Konfigurationsspeicher (siehe Abbildung 5.4) befinden sich sowohl die Ausgangstreiber des CPLDs als auch die simulierten Ausgänge des MSP430 im hochohmigen Zustand, gekennzeichnet durch den Buchstaben Z. Bei $t = 1.76 \mu\text{s}$ wird extern erstmals ein Datenwort vorgegeben und CR_WRITE auf high gesetzt. Entsprechend der angelegten Adresse 0b0001 wird der Datenwert 0b11100000 mit der nächsten steigenden Flanke des Taktsignals in den Konfigurationsspeicher data_reg[5:0] übernommen. In weiterer Folge werden drei zusätzliche Datenworte in den Speicher geschrieben und danach die vier soeben gespeicherten Werte durch Anlegen der entsprechenden Adressen mittels CR_OE = high (CR_WRITE = low) wieder ausgelesen, wobei die mit dem Datenbus verbundenen IOs des MSP430 in diesem Fall in den hochohmigen Zustand geschaltet werden müssen. Darüber hinaus ist bei allen Zugriffen auf den Konfigurationsspeicher CS_N = low zu setzen.

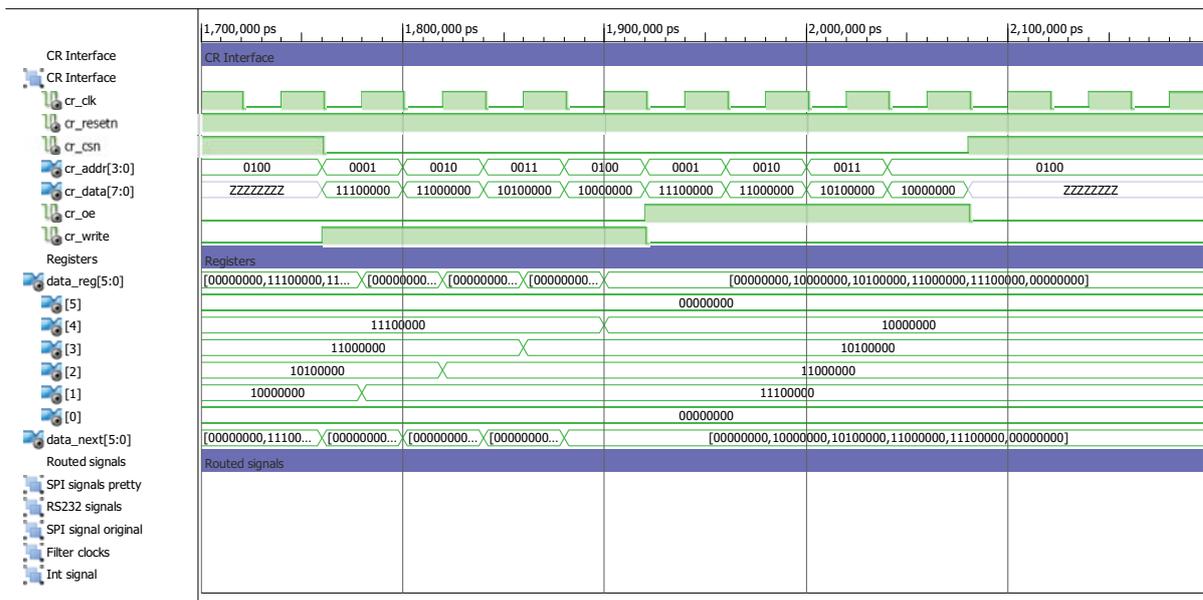


Abbildung 5.4: Simulation der CPLD-Logik – Schreiben und Lesen des Konfigurationsspeichers

Abbildung 5.5 zeigt das Umschalten zwischen aktiven SPI-Kanälen. Die SPI-Interfaces $C_SPI(0)$ und $C_SPI(1)$, entsprechend Abbildung 5.3, sind in den Simulationsergebnissen unter der Bezeichnung c_spi_0 bzw. c_spi_1 aufgeführt. Bis $t = 3\ \mu\text{s}$ ist $C_SPI(0)$ aktiv, danach wird durch Schreiben von $0b01010000$ an Adresse $0b0000$ $C_SPI(1)$ ausgewählt. Die Kurvenverläufe innerhalb der rot eingerahmten Bereiche von $C_SPI(0)$ und $C_SPI(1)$ sind gemäß der angestrebten Funktionalität identisch mit den zeitgleichen Abschnitten von CR_SPI .

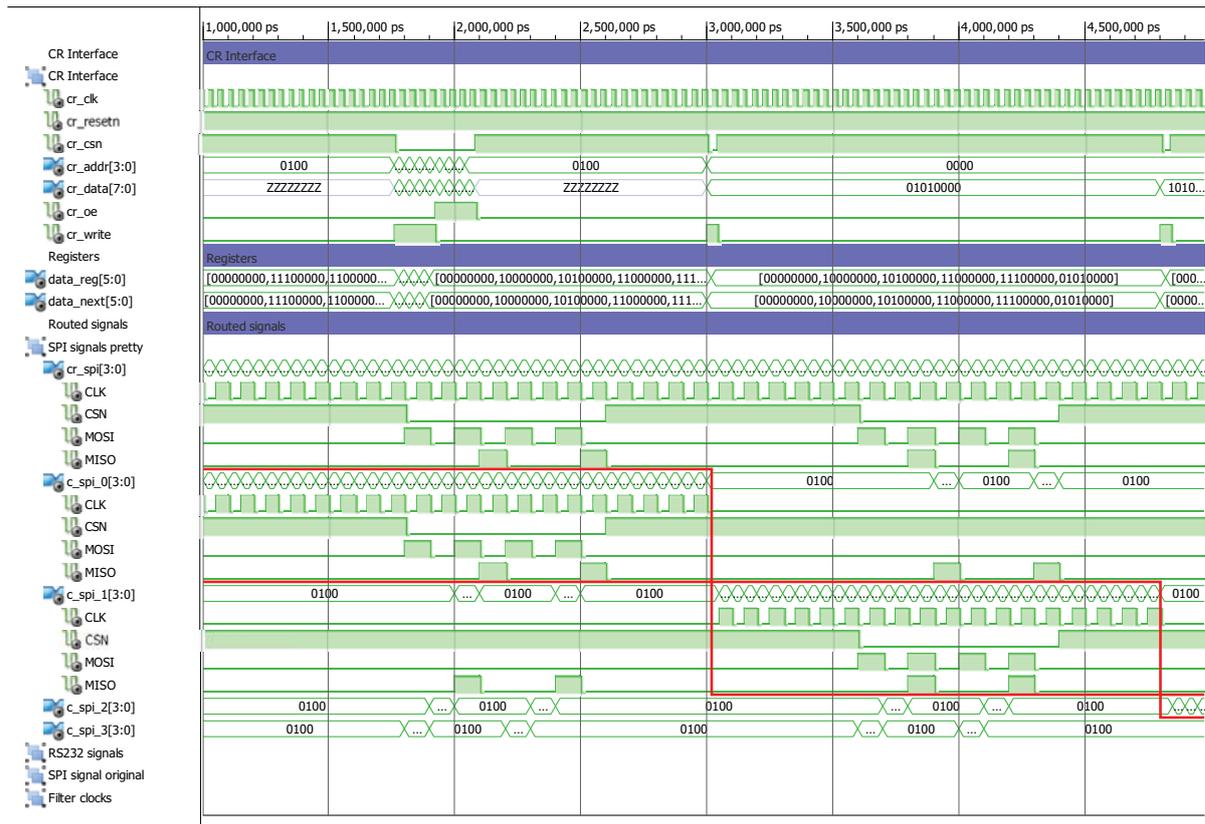


Abbildung 5.5: Simulation der CPLD-Logik – Umschalten des SPI-Kanals

Die Signalverläufe beim Umschalten zwischen verschiedenen CR_C_CLKs ist in Abbildung 5.6 dargestellt. Bei $t = 1.7\ \mu\text{s}$ erfolgt die Neukonfiguration durch Überschreiben der bisherigen Konfigurationsdaten an den Adressen $0b0001$ bis $0b0100$ mit neuen Werten, entsprechend Abbildung 5.4. Der jeweilige Umschaltzeitpunkt der einzelnen C_CLKs ist durch einen senkrechten roten Strich gekennzeichnet.

Wie die Abbildungen 5.4 bis 5.6 zeigen, konnte die gewünschte Funktionalität in der Simulation nachgewiesen werden.

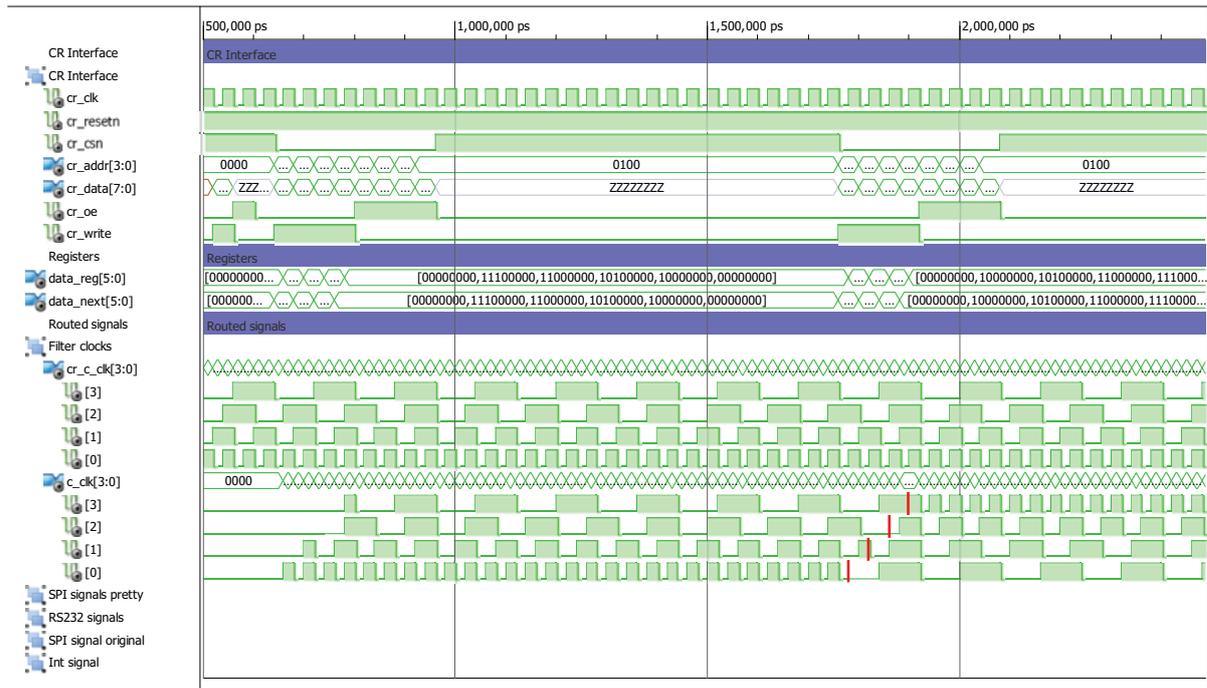


Abbildung 5.6: Simulation der CPLD-Logik – Umschalten der C_CLKs

5.1.2.3 CPLD-Timing

Entsprechend dem Timing-Report der ISE Design Suite ergeben sich die in Tabelle 5.5 angeführten maximalen Werte für die Input-Setup-Time $t_{IS,max}$, die Output-Valid-Time $t_{OV,max}$ sowie die Pad-To-Pad-Delay-Time $t_{PD,max}$. Die Werte $t_{IS,max}$ und $t_{OV,max}$ beziehen sich hierbei, wie in Abbildung 5.7 dargestellt, auf die steigende Flanke von CR_CLK.

Bezeichnung	Wert in ns
$t_{IS,max}$	4.1
t_{IH}	0.0
$t_{OV,max}$	10.6
$t_{PD,max}$	9.2

Tabelle 5.5: Timing-Kennwerte der CPLD-Logik

Die Zeiten t_{IS} und t_{IH} geben bei Eingangssignalen das Zeitfenster an, in dem diese konstant gehalten werden müssen, um eine korrekte Übernahme des jeweiligen Wertes durch den Baustein zu gewährleisten. Die Zeiten t_{OV} und t_{OH} beziehen sich auf Ausgangssignale und geben einerseits die Zeit an, die, bezogen auf die Ausgabeflanke des Taktsignals, vergeht bis der Ausgang den gültigen Wert annimmt (t_{OV}) und andererseits die Zeit, die

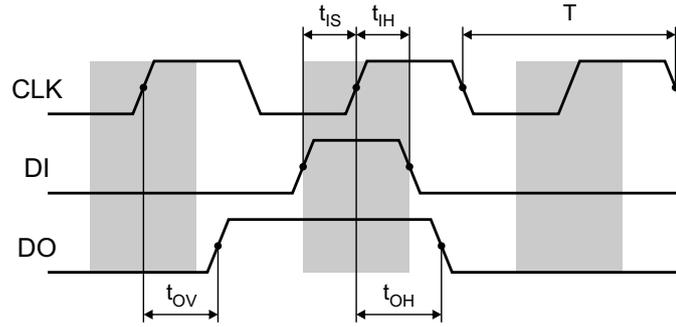


Abbildung 5.7: Setup- und Hold-Zeiten für die Eingangssignale DI bzw. DO in Relation zum Taktsignal CLK

das Ausgangssignal nach der zugehörigen Übernahmeflanke des Taktes den gültigen Wert beibehält (t_{OH}).

Hinsichtlich der Bestimmung der maximal zulässigen SPI-Taktfrequenz ist darüber hinaus die Signallaufzeit $t_{PD,max}$ zwischen Eingangs- und Ausgangspin relevant. Zu beachten ist hierbei, dass die angegebenen Zeiten von der Pinbelegung des CPLDs abhängig sind und bei jeder Änderung der Firmware, auch wenn diese lediglich die Pinbelegung betrifft, erneut zu überprüfen sind.

Tabelle 5.6 fasst die Timing-Kennwerte der mittels SPI-Bus verbundenen Bausteine inklusive der sich im Signalweg befindenden Logik zusammen. Die Werte der MOSI- bzw. MISO-Signale werden bei den angeführten Bausteinen bei der steigenden Flanke an den Eingängen übernommen, die Ausgänge ändern bei der fallenden Flanke des SPI-Takts ihren Zustand. Demzufolge beziehen sich die Werte für t_{IS} und t_{IH} auf die steigende und die Angaben bezüglich t_{OV} und t_{OH} in diesem Fall auf die fallende Flanke. Weitere Details sind in den Datenblättern [3, 64, 71, 95] dokumentiert.

Wie aus Tabelle 5.6 ersichtlich, ist der Signalpfad zwischen MSP430 und AD5231 kritisch hinsichtlich der maximal zulässigen Taktfrequenz des SPI-Busses. Das am CLK-Eingang des Bausteins anliegende Signal erfährt einerseits durch das CPLD und den zusätzlichen Pegelwandler in Relation zum ursprünglichen CLK-Signal des MSP430 die längste Verzögerung und andererseits weist der Baustein die längste Output-Valid-Time $t_{OV,3}$ auf. Für das MISO-Signal am zugehörigen Eingang des Mikrocontrollers erhält man daher für den AD5231 den folgenden Zeitversatz in Relation zur fallenden Flanke des ursprünglichen Taktsignals:

$$t_{OV,MISO} = t_{PP,CLK} + t_{PP,SN} + t_{OV,3} + t_{PP,SN} + t_{PP,MISO} = 73.2 \text{ ns.} \quad (5.1)$$

Bauteil	Bezeichnung	Wert in ns	Bemerkung
MSP430F5438	t_{IS}	25.0	$PMMCOREV = 3, V_{CC} = 3.0\text{ V}$
	t_{IH}	0.0	$PMMCOREV = 3, V_{CC} = 3.0\text{ V}$
	t_{OV}	20.0	$PMMCOREV = 3, V_{CC} = 3.0\text{ V}$
	t_{OH}	–	keine Angabe im Datenblatt
MCP23S08	$t_{IS,1}$	10.0	$V_{CC} = 2.7\text{ bis }5.5\text{ V}$
	$t_{IH,1}$	10.0	$V_{CC} = 2.7\text{ bis }5.5\text{ V}$
	$t_{OV,1}$	45.0	$V_{CC} = 2.7\text{ bis }5.5\text{ V}$
	$t_{OH,1}$	0.0	$V_{CC} = 2.7\text{ bis }5.5\text{ V}$
	$T_{CLK,min,1}$	100.0	$V_{CC} = 2.7\text{ bis }5.5\text{ V}$
M25P80	$t_{IS,2}$	5.0	$V_{CC} = 2.7\text{ bis }3.6\text{ V, Grade 3}$
	$t_{IH,2}$	5.0	$V_{CC} = 2.7\text{ bis }3.6\text{ V, Grade 3}$
	$t_{OV,2}$	15.0	$V_{CC} = 2.7\text{ bis }3.6\text{ V, Grade 3}$
	$t_{OH,2}$	0.0	$V_{CC} = 2.7\text{ bis }3.6\text{ V, Grade 3}$
	$T_{CLK,min,2}$	50.0	$V_{CC} = 2.7\text{ bis }3.6\text{ V, Grade 3}$
AD5231	$t_{IS,3}$	5.0	$C_L \leq 20\text{ pF}$
	$t_{IH,3}$	5.0	$C_L \leq 20\text{ pF}$
	$t_{OV,3}$	50.0	$C_L \leq 20\text{ pF}$
	$t_{OH,3}$	0.0	$C_L \leq 20\text{ pF}$
	$T_{CLK,min,3}$	20.0	$C_L \leq 20\text{ pF}$
SN74AVC4T774	$t_{PP,SN}$	2.6	$V_{CCA} = 3.3\text{ V, }V_{CCB} = 2.5\text{ V}$
XC2C384	$t_{PP,MOSI}$	8.8	Speed Grade –7
	$t_{PP,CLK}$	8.8	Speed Grade –7
	$t_{PP,MISO}$	9.2	Speed Grade –7

Tabelle 5.6: Timing-Kennwerte der Bausteine

Unter Berücksichtigung von t_{IS} erhält man damit die minimal zulässige Periodendauer des Taktsignals

$$T_{CLK,min} = 2(t_{OV,MISO} + t_{IS}) = 196.4\text{ ns.} \quad (5.2)$$

Die aus den Input-Setup- und Output-Valid-Zeiten resultierende minimale Periodendauer $T_{CLK,min}$ des Taktsignals überschreitet die in Tabelle 5.6 enthaltenen Werte $T_{CLK,min,i}$ der jeweiligen Bausteine und schränkt die mögliche Taktfrequenz $f_{CLK,max}$ daher nach oben ein. Die maximale Taktfrequenz beträgt folglich

$$f_{CLK,max} = 1/T_{CLK,min} = 5.09\text{ MHz.} \quad (5.3)$$

5.1.3 Prototyp der Steuerlogik

Die Funktionstests der Steuerlogik erfolgten mittels eines, aus zwei Platinen bestehenden, Funktionsprototypen. Sowohl die CPLD-Platine als auch die zugehörige Erweiterungsplatine sind in Abbildung 5.8 dargestellt. Ein Basismodul des Laborsystems (siehe Kapitel 4) sowie dessen Zusatzmodule mit Ausnahme des Datenspeichers können, wie in Abbildung 5.9 dargestellt, mit der Erweiterungsplatine verbunden werden. Somit kann in weiterer Folge die Firmware des Messsystem bereits vor der Fertigstellung der eigentlichen Hardware des Messsystems entwickelt und getestet werden.

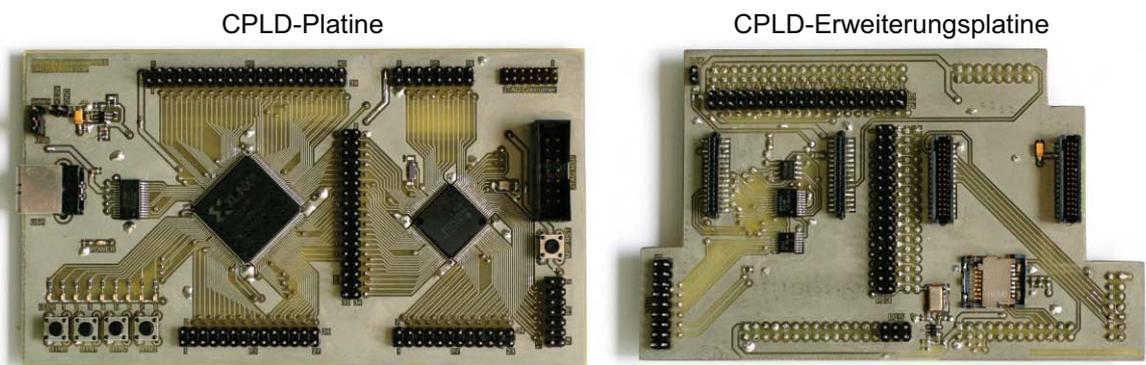


Abbildung 5.8: Funktionsprototyp der Steuerlogik bestehend aus CPLD-Platine und der zugehörigen Erweiterungsplatine

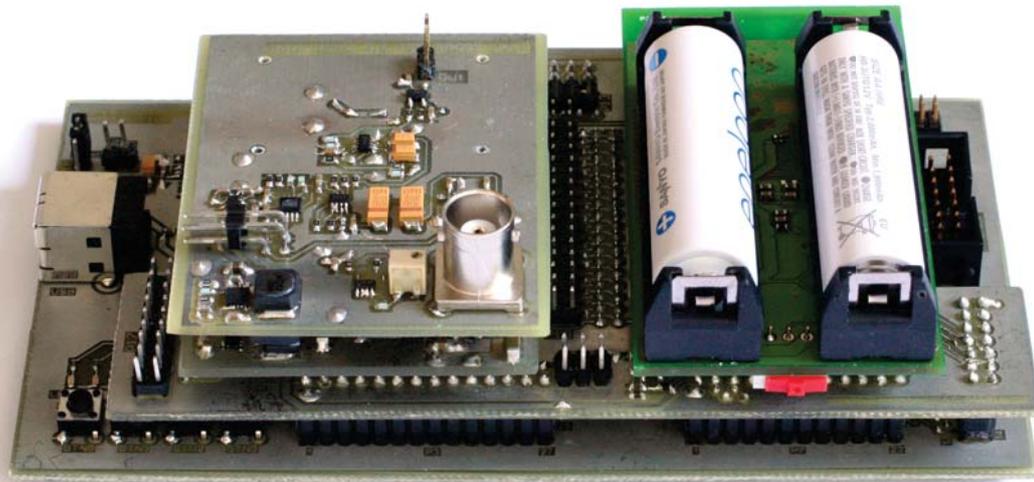


Abbildung 5.9: Funktionsprototyp der Steuerlogik inklusive der Module des Laborsystems

Abbildung 5.10 zeigt ein Blockschaltbild der beiden Platinen des Funktionsprototypen der CPLD-Logik. Die Basisplatte verfügt über einen MSP430F5438A, der mit einem Xilinx-CPLD (Coolrunner-II X2C384) verbunden ist. Alle, nicht durch Bauteile (Schalter, LEDs und FT232R) belegten, Ein- und Ausgänge des CPLDs sind über Stiftleisten zugänglich. Analoges gilt für den MSP430. Zur Kommunikation zwischen MSP430 und dem CPLD stehen 32 Signalleitungen zur Verfügung, die ebenfalls über eine Stiftleiste zugänglich sind. Details hinsichtlich der Verbindungen zwischen Mikrocontroller und CPLD sowie der Pinbelegungen der Stift- und Buchsenleisten sind im Anhang E zu finden.

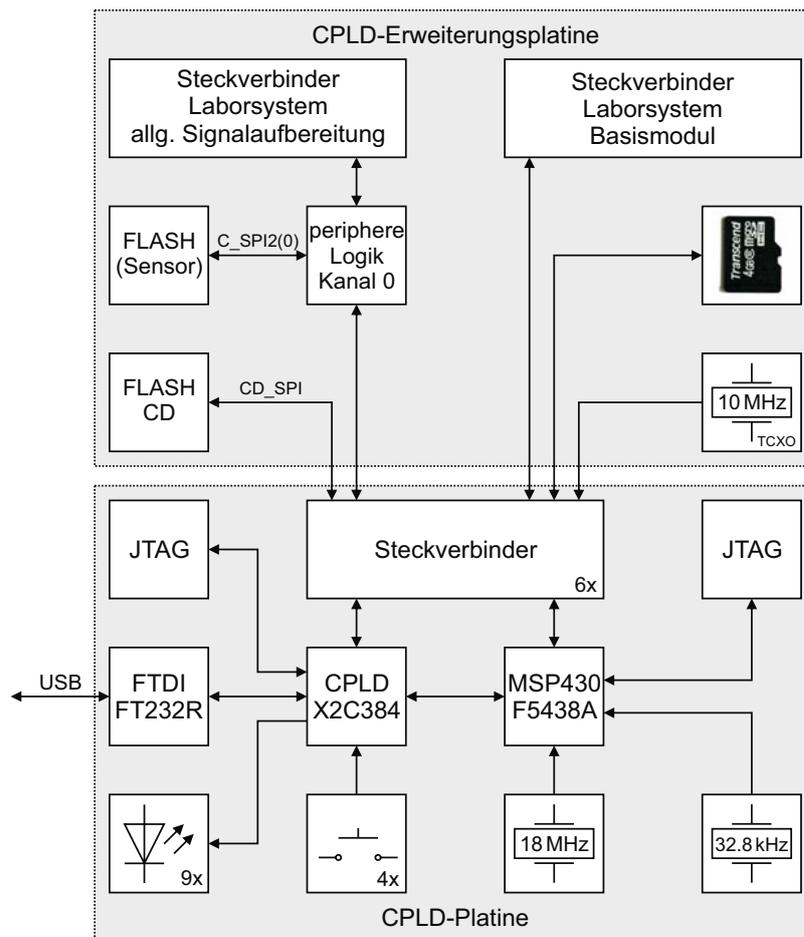


Abbildung 5.10: Blockdiagramm des Funktionsprototypen der Steuerlogik

Die Erweiterungsplatine enthält neben einem Sockel für microSD-Karten einen temperaturkompensierten Quarz und stellt hierdurch in Kombination mit dem Mikrocontroller der Basisplatte die Funktionalität des Datenspeichermoduls (siehe Abschnitt 4.3) zur Verfügung. Entsprechende Steckverbinder zum Aufstecken eines Basismoduls sind ebenfalls vorgesehen.

Neben der Funktionalität des Datenspeichers sind auf der Erweiterungsplatine die periphere Logik (siehe Abschnitt 5.1.1) eines Messkanals, sowie zwei serielle Flash-Speicher vorhanden. Mittels zweier Steckverbinder kann darüber hinaus die allgemeine Signalaufbereitung inklusive eines Sensor-Interfaces mit der Erweiterungsplatine verbunden werden. In Kombination mit den Modulen des Laborsystems steht daher ein Messknoten mit einem Messkanal inklusive Steuerlogik zur Verfügung.

5.1.4 Funktionstests

Auf Grund der umfangreichen Funktionalität der Hardware kann an dieser Stelle nicht auf alle diesbezüglichen Details inklusive der durchgeführten Tests eingegangen werden. Die Funktion aller verwendeten Bausteine sowie der Steuerlogik des CPLDs wurde mittels speziell erstellten Testroutinen (MSP430-Firmware) überprüft und konnte bestätigt werden. Eine Anpassung bzw. Erweiterung der Firmware des Laborsystems an die Hardware des vorliegenden Prototypen der Steuerlogik ist jedoch zum jetzigen Zeitpunkt noch nicht erfolgt.

Die Signalverläufe beim Schreiben und Lesen des Konfigurationsspeichers des CPLDs sind in Abbildung 5.11 dargestellt. Ein Schreib- bzw. Lesezyklus erstreckt sich über mehrere Zyklen des Mikrocontrollertaktes (CLK), der auch als Takt für das CPLD dient. Da die Steuersignale durch den Mikrocontroller sequentiell gesetzt werden, ist darauf zu achten, dass während CR_WRITE high gesetzt ist, die restlichen Signale konstant gehalten wer-

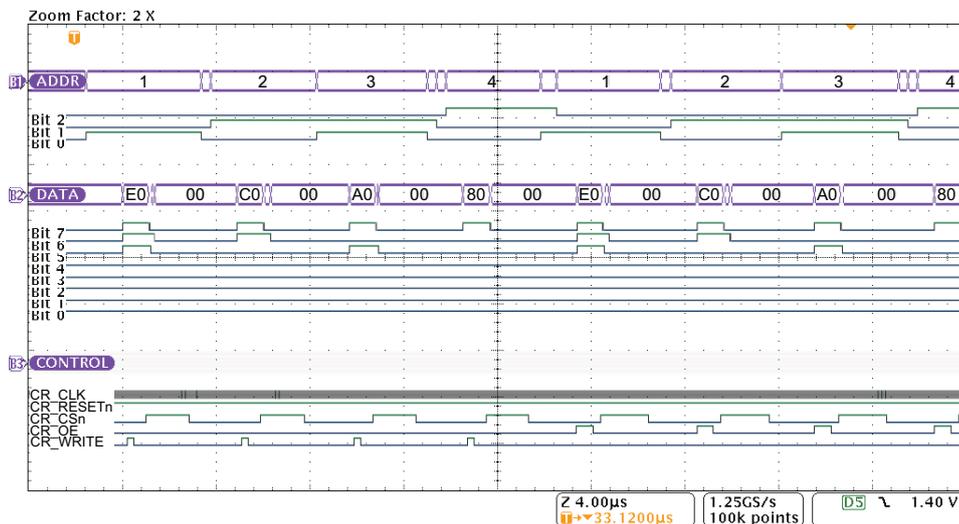


Abbildung 5.11: Signalverläufe beim Schreiben und Lesen des Konfigurationsspeichers

den und deren Werte gültig sind. Beim Lesezugriff auf den Konfigurationsspeicher muss sichergestellt werden, dass die mit den Datenleitungen verbundenen Pins des Mikrocontrollers als Eingänge konfiguriert wurden bevor CR_OE high gesetzt wird. Hinsichtlich der Einhaltung dieser Bedingungen ist zu überprüfen ob etwaig aktivierte Optimierungsalgorithmen des Compilers zu keiner Verletzung dieser Vorbedingung führen. Im Rahmen der Tests war die Compiler-Optimierung deaktiviert.

Angaben bezüglich der Timingwerte der Ein- und Ausgänge bei Verwendung als GPIOs sind im Datenblatt [93] nicht angeführt. Die Messungen im Rahmen der Versuche haben gezeigt, dass die verwendeten Ausgänge spätestens 4 ns nach der fallenden Flanke des Prozessortaktes gültig sind, womit keine Verletzung der Timinganforderungen des CPLDs (siehe Tabelle 5.5) gegeben ist. Auf Grund der unterschiedlichen Funktionalität der Mikrocontroller-Pins und dem daraus resultierenden, unterschiedlichen Aufbau der jeweiligen Ausgangslogik, kann hier nicht von einer generellen Gültigkeit ausgegangen werden. Eine Überprüfung hinsichtlich der Einhaltung der Timingvorgaben sollte daher bei einer geänderten Zuordnung zwischen CPLD-Signalen und MSP430-Pins erneut durchgeführt werden.

Abbildung 5.12 zeigt die SPI-Signale des MSP430 (CR_SPI) sowie die korrespondierenden Signale des Kanals 0, beim Zugriff auf den MCP23S08 und den AD5231. Die Auswahl des aktiven SPI-Slaves erfolgt mittels C_SPI_CS_n1 bzw. C_SPI_CS_n0, durch Low-Setzen von CR_SPI_CS_n in Kombination mit einer Änderung der Konfigurations-

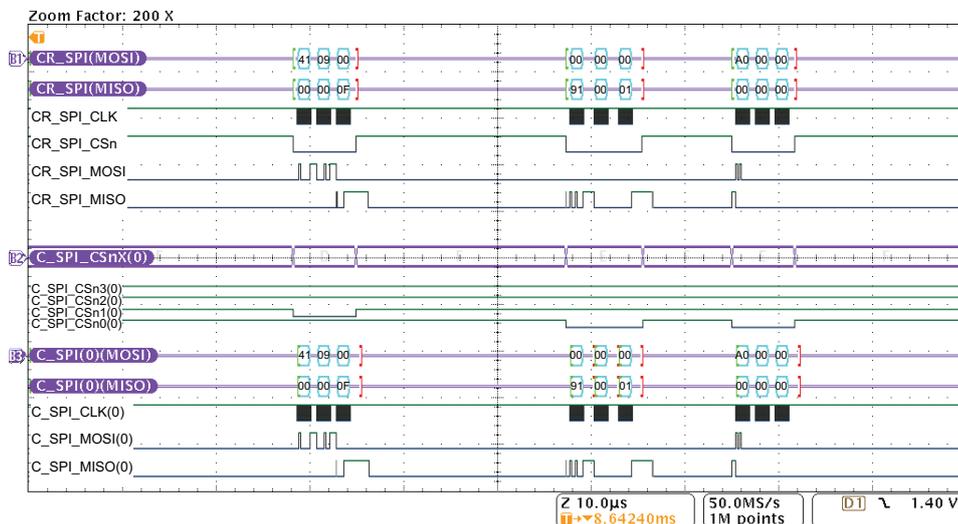


Abbildung 5.12: SPI-Kommunikation zwischen MSP430 und MCP23S08 (C_SPI1) bzw. AD5231 (C_SPI0)

register im CPLD. Eine Darstellung aller beteiligten Signale (Adress-, Datenbus, etc.) war wegen der beschränkten Anzahl der digitalen und analogen Eingänge (16 + 4) des MSO4054 nicht möglich.

Die Funktion der C_CLK-Demultiplexer ist in Abbildung 5.13 veranschaulicht. Zwei C_CLKs mit Frequenzen von 18.75 kHz und 1.875 kHz werden durch die CPLD-Logik entsprechend der jeweiligen Konfiguration zu den einzelnen Kanälen weitergeleitet. Die CR_C_CLK(0) mit 18.75 kHz dient hierbei jeweils für etwa 2 ms als Quelle für das Taktsignal (SC_CLK(i)) jedes Kanals. In der verbleibenden Zeit wurde CR_C_CLK(1) als Quelle gewählt. Die Adress-, Daten- und Steuerleitungen des CPLD-Interfaces konnten aus bereits erwähnten Gründen nicht dargestellt werden.

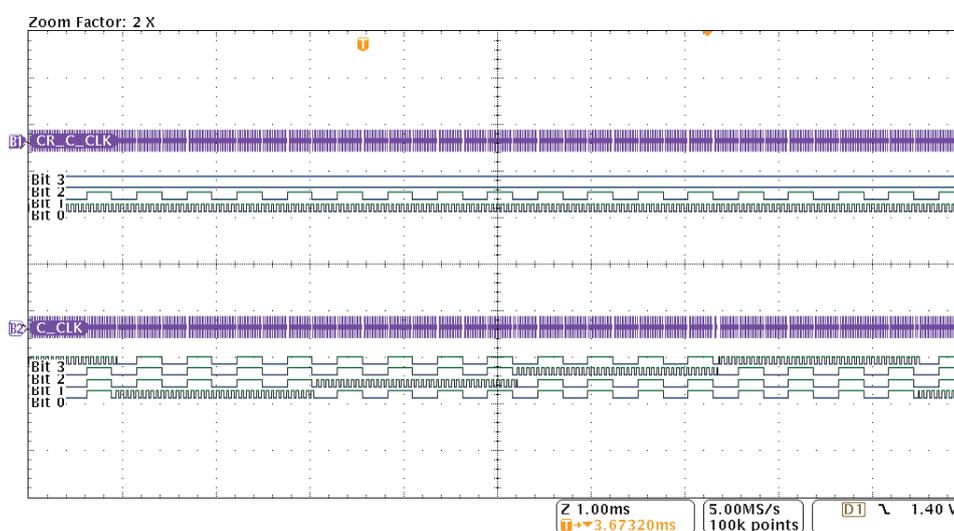


Abbildung 5.13: Umschaltung der Quelle der Taktsignale CSC_CLK(0) (Bit0) bis SC_CLK(3) (Bit3)

5.1.5 Strombedarf

Das verwendete CPLD benötigt Versorgungsspannungen in der Höhe von +1.8V und +3.3V. Im Rahmen des Funktionsprototypen werden +1.8V mittels des linearen Spannungsreglers ADP3331 (Analog Devices) von der vorhandenen +3.3V-Versorgungsspannung abgeleitet. Der Strombedarf des CPLDs inklusive des ADP3331 beträgt im Betrieb 4.1 mA.

5.2 Energieversorgung

Die Energieversorgung des Messknotens (siehe Abbildung 5.14) erfolgt im Normalfall über das Bordnetz der Straßenbahn, wobei die Nennspannung $+24\text{ V}$ beträgt und eine Toleranz von -30% bzw. $+25\%$ festgelegt ist. Das Energieversorgungssystem des Messsystems muss daher für Eingangsspannungen von $+16.8$ bis $+30\text{ V}$ dimensioniert werden.

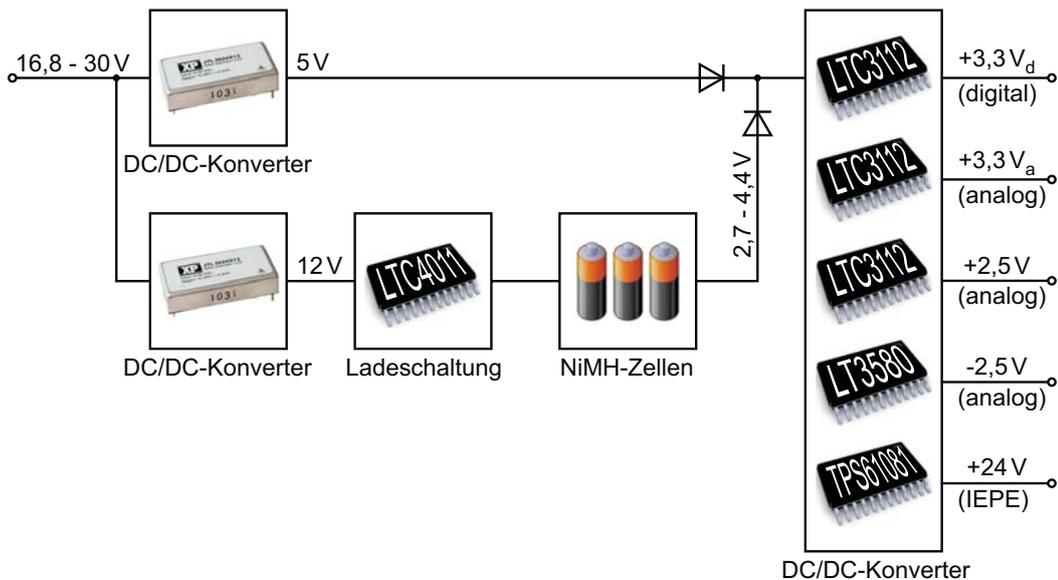


Abbildung 5.14: Energieversorgung des Messknotens

Die digitalen und analogen Komponenten des Systems erfordern stabilisierte Spannungen von $+3.3$, $+2.5$, -2.5 und $+24\text{ V}$. Diese werden mittels DC/DC-Konvertern von einer Zwischenkreisspannung abgeleitet. Im Normalfall – der Versorgung des Systems durch das Bordnetz – beträgt diese $+5\text{ V}$. Bei kurzen Spannungseinbrüchen übernimmt ein Pufferakkumulator bestehend aus drei in Serie geschalteten NiMH-Zellen die Energieversorgung des Systems. Bei einer Entladerate von 1 C liegt die Zellenspannung etwa im Bereich von 0.9 bis 1.45 V (vergleiche Abbildung 3.13). Die sekundären DC/DC-Konverter müssen daher für Eingangsspannungen von $+2.7$ bis $+5\text{ V}$ ausgelegt sein.

Beim Laden von NiMH-Zellen werden je nach Zelle und Ladestrom unterschiedlich hohe Spannungen benötigt. Abbildung 3.14 zeigt beispielhaft den Verlauf der Ladespannung einer Zelle bei einer Laderate von 0.5 C . Da bei $+5\text{ V}$ Zwischenkreisspannung pro Zelle lediglich maximal 1.67 V zur Verfügung stehen würden und Lade ICs, wie der hier verwendete LTC4011, pro Zelle etwa 2 V benötigen, kann die Ladeschaltung nicht mit der gewählten Zwischenkreisspannung betrieben werden.

Eine direkte Versorgung des LTC4011 durch das 24 V-Bordnetz wäre prinzipiell möglich, ist auf Grund der fehlenden galvanischen Trennung jedoch nicht vorteilhaft. Eine entsprechende Erhöhung der Zwischenkreisspannung auf beispielsweise +12 V würde eine drastische Einschränkung der einsetzbaren DC/DC-Konverter mit sich bringen, da der Eingangsspannungsbereich von +2.7 bis +12 V nur von einer geringen Anzahl der verfügbaren IC-Typen unterstützt wird.

Um eine möglichst große Flexibilität speziell auch hinsichtlich späterer Erweiterungen durch Sensorinterfaces und damit eventuell benötigter DC/DC-Konverter zu gewährleisten, wurde die Zwischenkreisspannung auf +5 V begrenzt. Die Versorgung der Ladeschaltung mit +12 V erfolgt in weiterer Folge über einen eigenen DC/DC-Konverter mit galvanisch isolierter Sekundärspannung.

Sowohl die digitale Elektronik als auch einige Bauteile im Signalweg der Sensorsignale benötigen eine Versorgungsspannung von +3.3 V. Um eine Beeinflussung der analogen Signale durch Schwankungen der Versorgungsspannung, die durch den etwaigen Betrieb von digitalen Komponenten über das selbe Versorgungsnetz verursacht werden, zu minimieren, ist für die Bauteile im Signalweg der Sensorsignale ein separater DC/DC-Konverter vorgesehen. Die jeweiligen Netze werden im Folgenden durch $+3.3V_d$ (digitale Elektronik) bzw. $+3.3V_a$ (analoge Signalverarbeitung) gekennzeichnet.

In den weiteren Abschnitten ist neben einer Abschätzung des Strombedarfs für das Gesamtsystem sowohl die Beschreibung der, für die Verwendung im Messsystem ausgelegten, DC/DC-Konverter als auch die Dokumentation der Ladeschaltung inklusive Power-Path™Controller ausgeführt.

5.2.1 Abschätzung des Energiebedarfs eines Knotens

Der Strombedarf der einzelnen Komponenten des Messsystems (siehe Tabellen 4.4, 4.20 und 4.24 sowie Abschnitte 4.3.1.4 und 5.1.5) ist in Tabelle 5.7 zusammengefasst.

Ein System mit vier Messkanälen, das mit IEPE-Sensoren und entsprechenden Interfaces ausgerüstet ist, würde somit im normalen Betrieb etwa die in Tabelle 5.8 angegebenen Ströme benötigen. Hierbei wurde angenommen, dass sich der Funk-Transceiver jeweils 50 % der Zeit im Sende- bzw. Empfangsmodus befindet. Etwaige kurzzeitige Erhöhungen des Strombedarfs, die z.B. in Folge der Konfiguration des AD5231 auftreten, sind hier jedoch nicht berücksichtigt. Um bei zukünftigen Erweiterungen des Systems die Wahrscheinlichkeit eines notwendigen Redesigns der Energieversorgung gering zu halten, sollte diese ausreichende Reserven hinsichtlich der Ausgangsleistung aufweisen. Die zulässigen

Komponente	+3.3 V _d	+3.3 V _a	+2.5 V	-2.5 V	+24 V
Basismodul					
MSP430F2618	9.2	–	–	–	–
CC2520 RX	18.5	–	–	–	–
CC2520 TX	33.6	–	–	–	–
Datenspeicher					
MSP430F2618	5.6	–	–	–	–
SD-Karte	22.0	–	–	–	–
Oszillator	3.0	–	–	–	–
Allg. Signalaufbereitung					
Steuerlogik (CPLD)	4.1	–	–	–	–
Schreiben M25P80 (ser. Flash)	15.0	–	–	–	–
Analogteil (pro Kanal)	–	7.7	1.6	-5.2	–
Konfiguration AD5231	–	–	40.0	-40.0	–
IEPE-Interface (pro Kanal)					
Stromquelle	–	–	–	–	4.0
Spannungsteiler	–	–	–	–	0.3
Verstärker	–	–	0.5	-0.5	–

Tabelle 5.7: Strombedarf der einzelnen Module in mA

Versorgungsspannung	Strom in mA	Leistung in mW
+3.3 V _d	70.0	231
+3.3 V _a	30.8	102
+2.5 V	8.6	21
-2.5 V	-23.0	57
+24 V	17.3	415

Tabelle 5.8: Strombedarf des Messsystems im normalen Betrieb

Ausgangsströme der, in den weiteren Abschnitten beschriebenen, DC/DC-Konverter überschreiten daher die aus den bisherigen Entwicklungsschritten bekannten Anforderungen deutlich.

5.2.2 DC/DC-Konverter

Auf die Auslegung der einzelnen DC/DC-Konverter kann an dieser Stelle aus Platzgründen nicht näher eingegangen werden, diesbezüglich wird auf die jeweiligen Datenblätter [51, 53, 100] verwiesen. Neben einer kurzen Zusammenfassung der Eigenschaften der

Konverter sind in den folgenden Abschnitten die zugehörigen Schaltbilder und die, durch Messung der Ein- und Ausgangsspannungen bzw. -Ströme (vergleiche Abbildung 5.15), ermittelten Wirkungsgrade (inklusive Diagrammen der Konverter-Ausgangswelligkeiten) wiedergegeben. Die Wirkungsgrade wurden sowohl für den im Normalfall zu erwartenden Wert der Zwischenkreisspannung (+5 V) bei Versorgung durch das Bordnetz als auch für die im Akkumulatorbetrieb in der Regel bzw. bei nahezu vollständig entladene Akkumulator vorherrschenden Zwischenkreisspannungen von +3.6 V bzw. +2.7 V bestimmt. Die konkreten Messwerte, inklusive der daraus bestimmten Wirkungsgrade, sind im Anhang A dokumentiert. Die Zeitverläufe der Ausgangsspannungen der DC/DC-Konverter wurden jeweils unmittelbar am Ausgang des Konverters und nach dem zugehörigen Ausgangsfilter für Lasten im Bereich des, entsprechend Tabelle 5.8, zu erwartenden Laststromes gemessen.

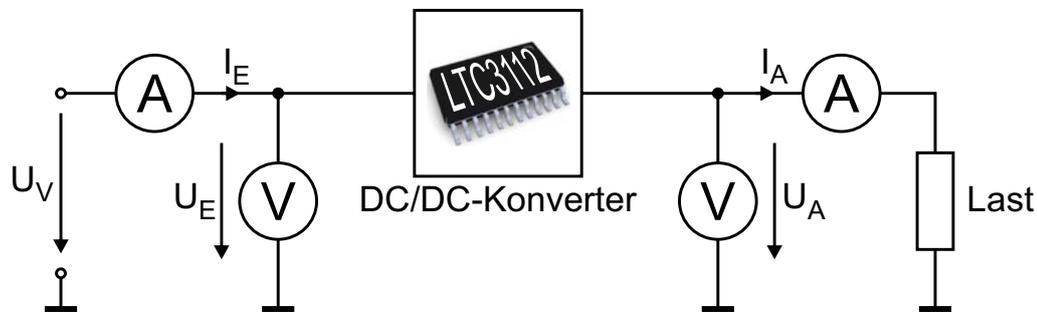


Abbildung 5.15: Messung des Wirkungsgrades

5.2.2.1 LTC3112

Der LTC3112 von Linear Technology [53] ist für Eingangsspannungen von +2.7 bis +15 V ausgelegt und kann Ausgangsspannungen generieren, die sowohl unter als auch über der Eingangsspannung liegen. Konkret kann mit diesem IC ein Ausgangsspannungsbereich von +2.5 bis +14 V abgedeckt werden. Der LTC3112 kann folglich für die Bereitstellung der +3.3 V und der +2.5 V-Versorgungsspannungen verwendet werden. Zur Reduktion der Ausgangswelligkeit des Konverters kann dieser mit einer konstanten Schaltfrequenz von 750 kHz betrieben werden. Weiters weist der LTC3112 eine integrierte Strombegrenzung, einen Schutz vor Überhitzung sowie eine Soft-Start-Funktion auf.

Das Schaltbild sowie die Prototypen für eine Ausgangsspannung von +3.3 und +2.5 V sind in den Abbildungen 5.16 und 5.17 dargestellt. Beide Prototypen arbeiten mit einer konstanten Schaltfrequenz von 750 kHz. Um eine Ausgangsspannung von +2.5 V zu er-

zeugen sind die Widerstände R_1 , R_3 und R_4 abweichend von den Werten in Abbildung 5.16 entsprechend Tabelle 5.9 zu bestücken.

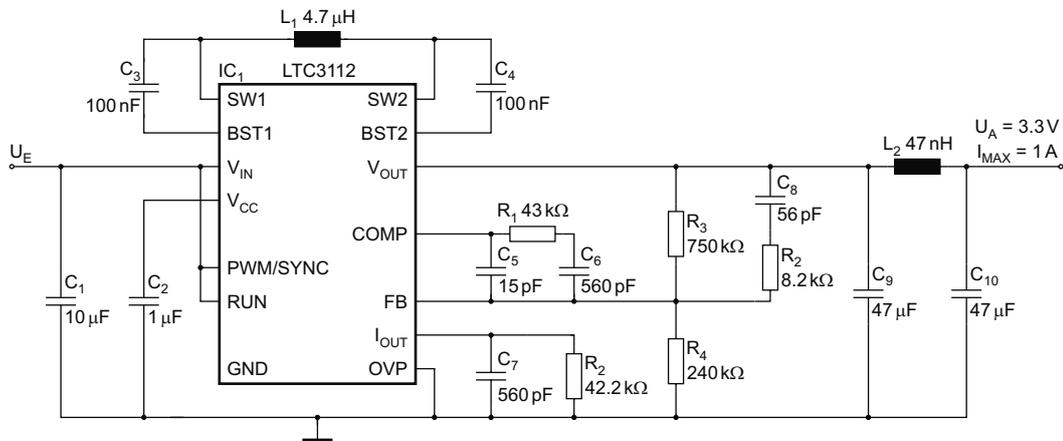


Abbildung 5.16: Schaltbild des LTC3112 – Ausgangsspannung +3.3V

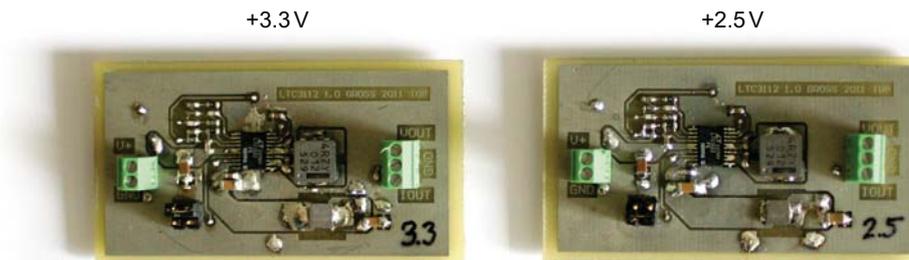
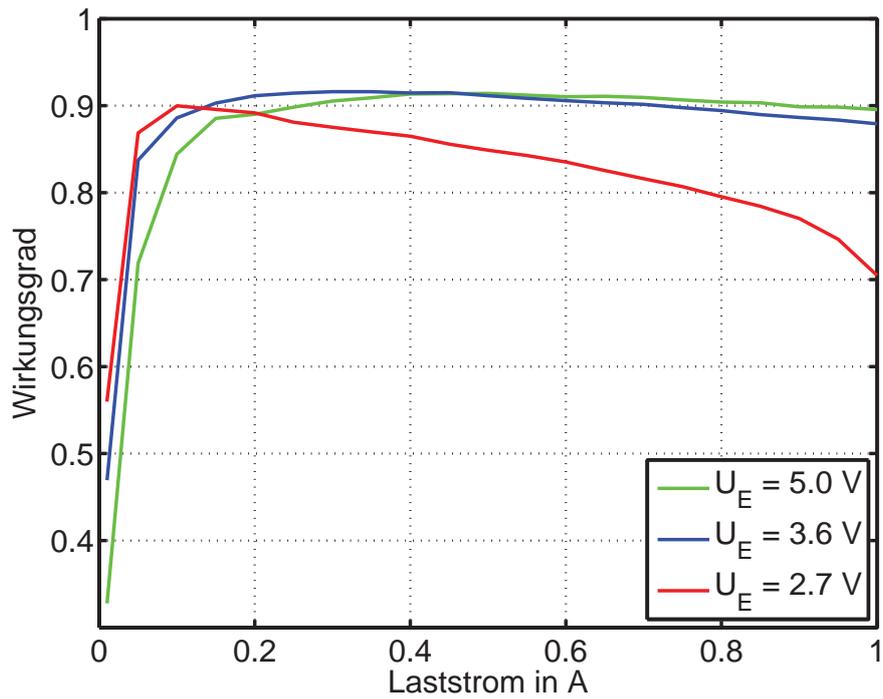
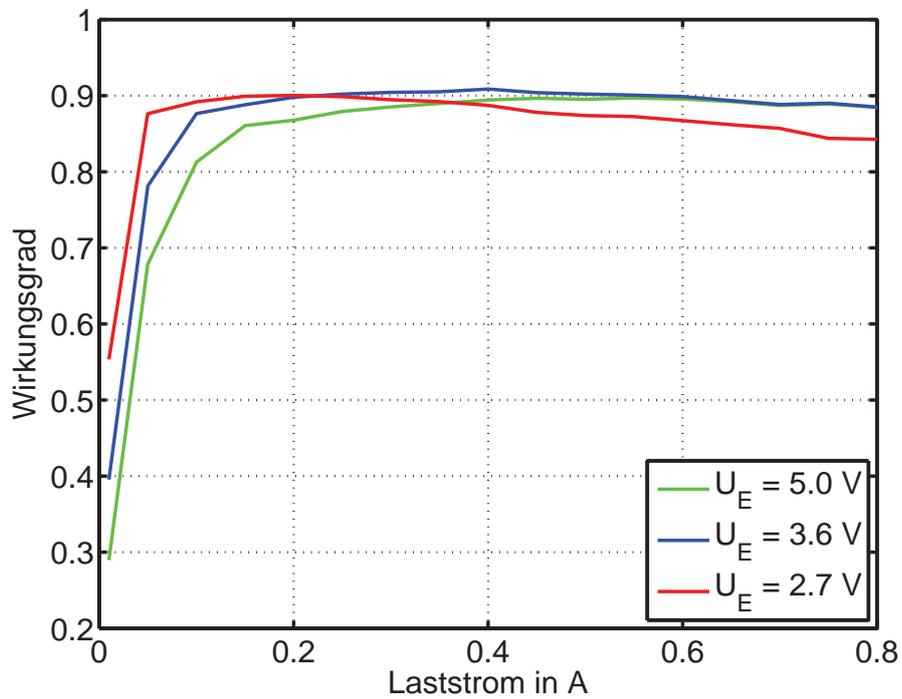


Abbildung 5.17: LTC3112 Prototypen

Widerstand	Wert in kΩ
R_1	39
R_3	680
R_4	309

Tabelle 5.9: LTC3112 – Widerstandswerte für $U_A = +2.5V$

Der Wirkungsgrad des LTC3112 bei den gewählten Ausgangsspannungen ist in den Abbildungen 5.18 und 5.19 in Abhängigkeit vom Ausgangsstrom und der Eingangsspannung dargestellt. Weiters sind die jeweiligen Ausgangswelligkeiten des LTC3112 für den typischen Lastfall (vergleiche Tabelle 5.8) in den Abbildungen 5.20 und 5.21 wiedergegeben. Die Eingangsspannung U_E betrug jeweils +5V.

Abbildung 5.18: Wirkungsgrad des LTC3112 bei $U_A = +3.3\text{ V}$ Abbildung 5.19: Wirkungsgrad des LTC3112 bei $U_A = +2.5\text{ V}$

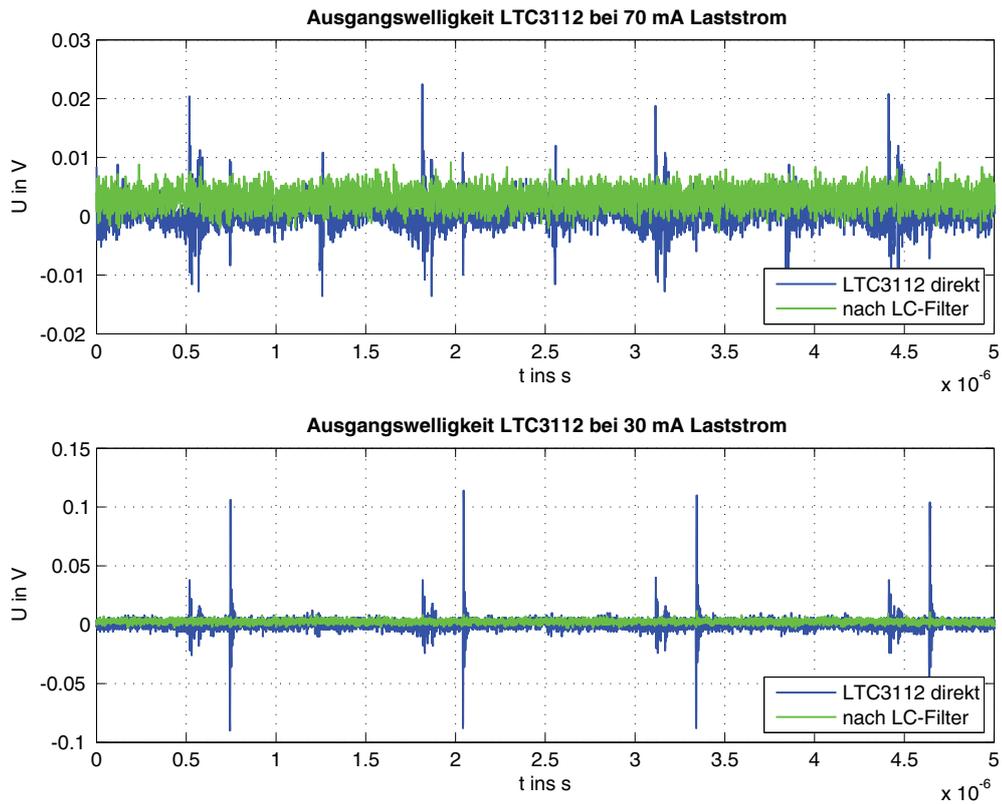


Abbildung 5.20: Ausgangswelligkeit des LTC3112 bei $U_E = +5\text{ V}$ und $U_A = +3.3\text{ V}$ (AC-Einkopplung)

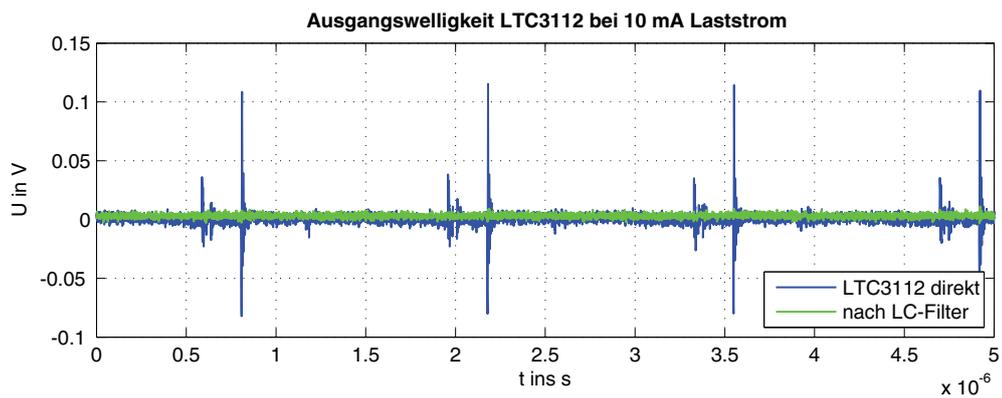


Abbildung 5.21: Ausgangswelligkeit des LTC3112 bei $U_E = +5\text{ V}$ und $U_A = +2.5\text{ V}$ (AC-Einkopplung)

5.2.2.2 LT3580

Der LT3580 [51] von Linear Technology ist für Eingangsspannungen von +2.5 bis +32 V ausgelegt und kann sowohl negative Ausgangsspannungen als auch Spannungen die oberhalb der Eingangsspannung liegen erzeugen, wobei der Strom durch die integrierten MOSFETs jedoch auf 2 A beschränkt ist. Der Konverter arbeitet mit einer fixen Schaltfrequenz, die mittels eines Widerstandes innerhalb des Bereiches von 200 kHz bis 2.5 MHz gewählt werden kann. Neben einem Schutz vor Überhitzung verfügt der LT3580 ebenfalls über eine Soft-Start-Funktion.

Das Schaltbild des LT3580 sowie ein Foto des Prototypen sind in den Abbildungen 5.22 und 5.23 dargestellt. Die gewählte Schaltfrequenz des Prototypen beträgt 1 MHz. Mittels Messungen wurden sowohl der Wirkungsgrad als auch die Ausgangswelligkeit für den typischen Lastfall (vergleiche Tabelle 5.8) bei einer Eingangsspannung von +5 V bestimmt. Die Ergebnisse sind in den Abbildungen 5.24 und 5.25 dargestellt.

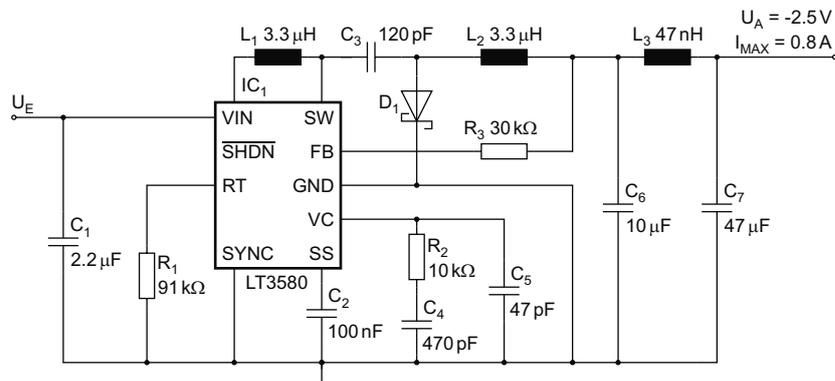


Abbildung 5.22: Schaltbild des LT3580

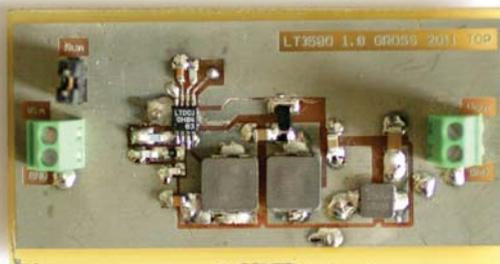
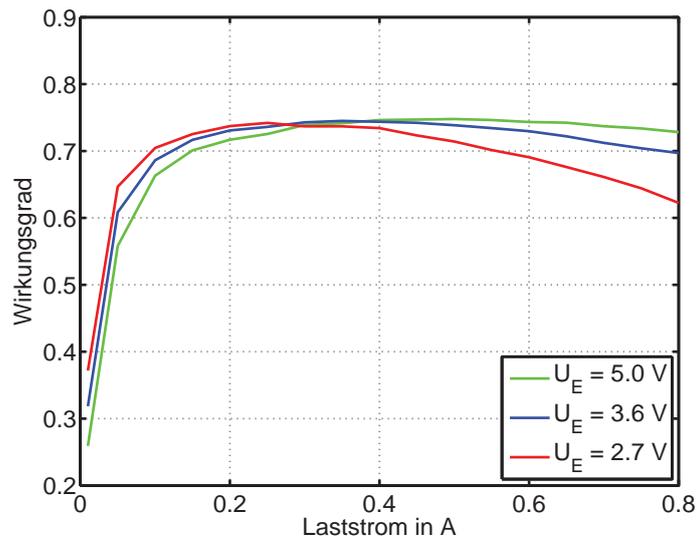
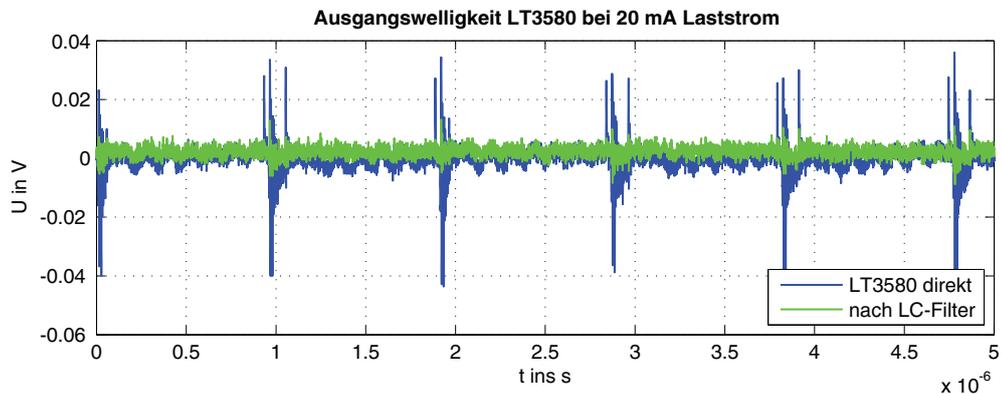
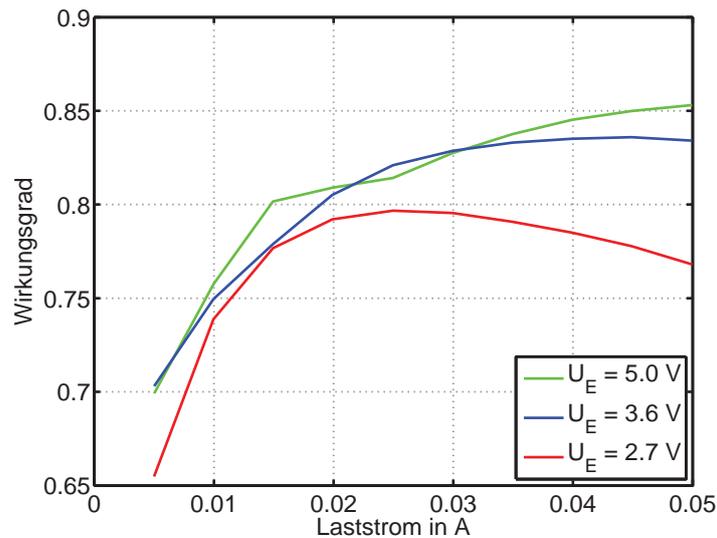
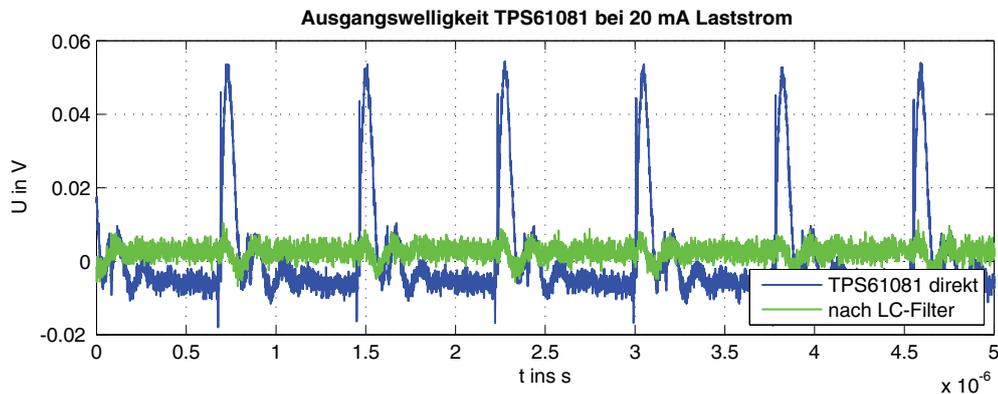


Abbildung 5.23: LT3580 Prototyp

Abbildung 5.24: Wirkungsgrad des LT3580 bei $U_A = -2.5\text{ V}$ Abbildung 5.25: Ausgangswelligkeit des LT3580 bei $U_E = +5\text{ V}$ und $U_A = -2.5\text{ V}$ (AC-Einkopplung)

5.2.2.3 TPS61081

Der TPS61081 wurde bereits in Rahmen des IEPE-Interfaces (siehe Abschnitt 4.5.1) eingesetzt, weshalb auf eine nähere Beschreibung an dieser Stelle verzichtet werden kann. Der Wirkungsgrad des DC/DC-Konverters bei einer Schaltfrequenz von 1.2 MHz sowie die Ausgangswelligkeit für den typischen Lastfall (vergleiche Tabelle 5.8) sind in Abbildung 5.26 bzw. 5.27 dargestellt.

Abbildung 5.26: Wirkungsgrad des TPS61081 bei $U_A = 24\text{ V}$ Abbildung 5.27: Ausgangswelligkeit des TPS61081 bei $U_E = +5\text{ V}$ und $U_A = +24\text{ V}$ (AC-Einkopplung)

5.2.3 Ladeschaltung LTC4011 und PowerPath™ Controller LTC4412

Auf dem Markt sind eine Reihe von linearen und schaltenden Lade-ICs für NiCd- bzw. NiMH-Akkumulatoren verfügbar. Die bei linearen Ladeschaltungen unvermeidbar hohen Verluste, die in Form von Wärme abgeführt werden müssen, würden unmittelbar zu einer Temperaturerhöhung im Gehäuse des Messsystems führen. Da die Eigenschaften nahezu aller elektronischen Bauteile von der Temperatur abhängen, ist speziell bei der analogen Verarbeitung von Messsignalen eine konstante Temperatur anzustreben. Unregelmäßige

Temperaturerhöhungen, die z.B. durch das Nachladen von Akkumulatoren verursacht werden, sollten daher so gering wie möglich ausfallen. Aus diesem Grund ist in Hinsicht auf die Qualität der aufgezeichneten Messdaten die Verwendung einer linearen Ladeschaltung nicht vorteilhaft. Die Wahl fiel daher auf den schaltenden Laderegler LTC4011, der für das Laden von bis zu 16 in Serie geschalteten NiCd- oder NiMH-Zellen eingesetzt werden kann. Abbildung 5.28 zeigt das Schaltbild des LTC4011.

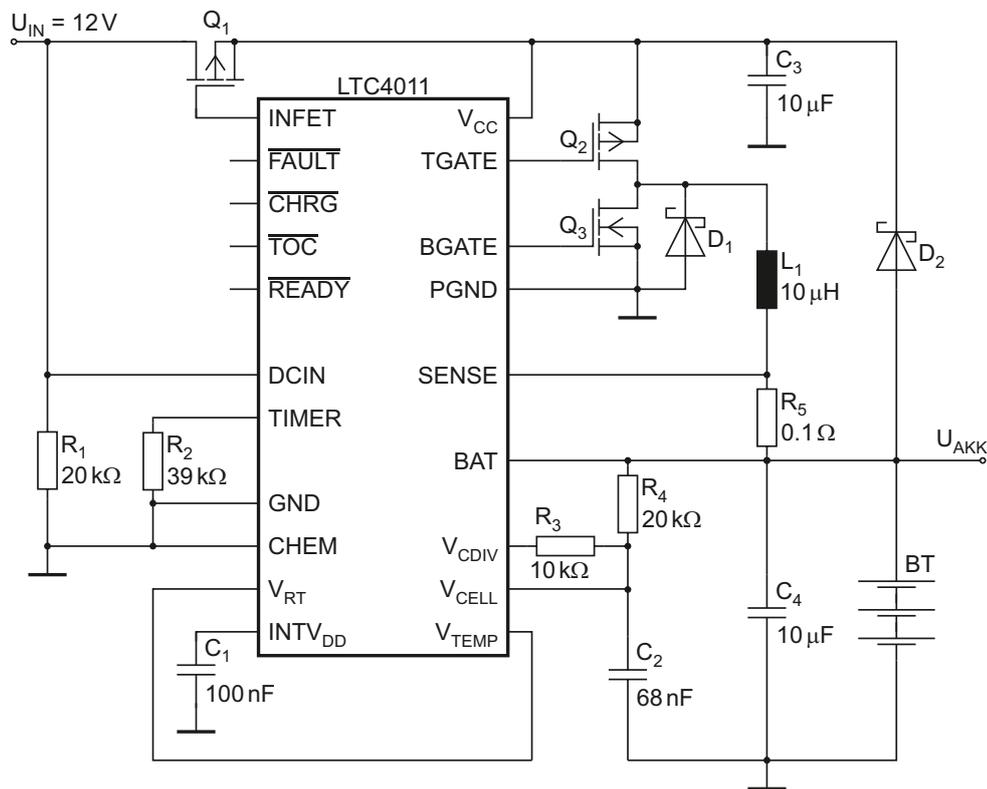


Abbildung 5.28: Schaltbild des Ladereglers LTC4011

Die Schaltung ist für das Laden von drei in Serie geschalteten NiMH-Zellen mit einer Kapazität von 800 mA h bei einer Eingangsspannung von +6.3 bis +34 V ausgelegt. Der Ladestrom beträgt 1 A, wobei der eigentliche Ladezyklus entweder bei Erfüllung des $-\Delta V$ -Kriteriums oder nach einer fixen Zeit beendet wird. Die Abschaltung nach einer fixen Zeit wird über Timer-Pin (siehe Abbildung 5.28) des LTC4011 konfiguriert und dient hierbei als reine Schutzfunktion vor Überladung. Im Normalfall wird das Ende des Ladezyklus mittels des $-\Delta V$ -Kriteriums bestimmt. Nach dem Ende des Hauptladezyklus erfolgt eine zeitlich begrenzte Nachladung der Akkumulatoren mit einem Zehntel des nominellen Ladestroms um die vollständige Ladung der Akkumulatoren sicherzustellen (Top-Off Charge). Sinkt die mittlere Zellenspannung der angeschlossenen Akkumulatoren

unter 1.325 V, so erfolgt eine automatische Ladung der Zellen. Nähere Details hinsichtlich der Eigenschaften des LTC4011 können dem Datenblatt [55] entnommen werden.

Ein mit dieser Schaltung durchgeführter Ladezyklus eines, aus drei in Serie geschalteten NiMH-Zellen bestehenden, Akkumulatorpacks der Firma Emmerich mit einer Gesamtkapazität von 800 mA h ist in Abbildung 5.29 dargestellt.

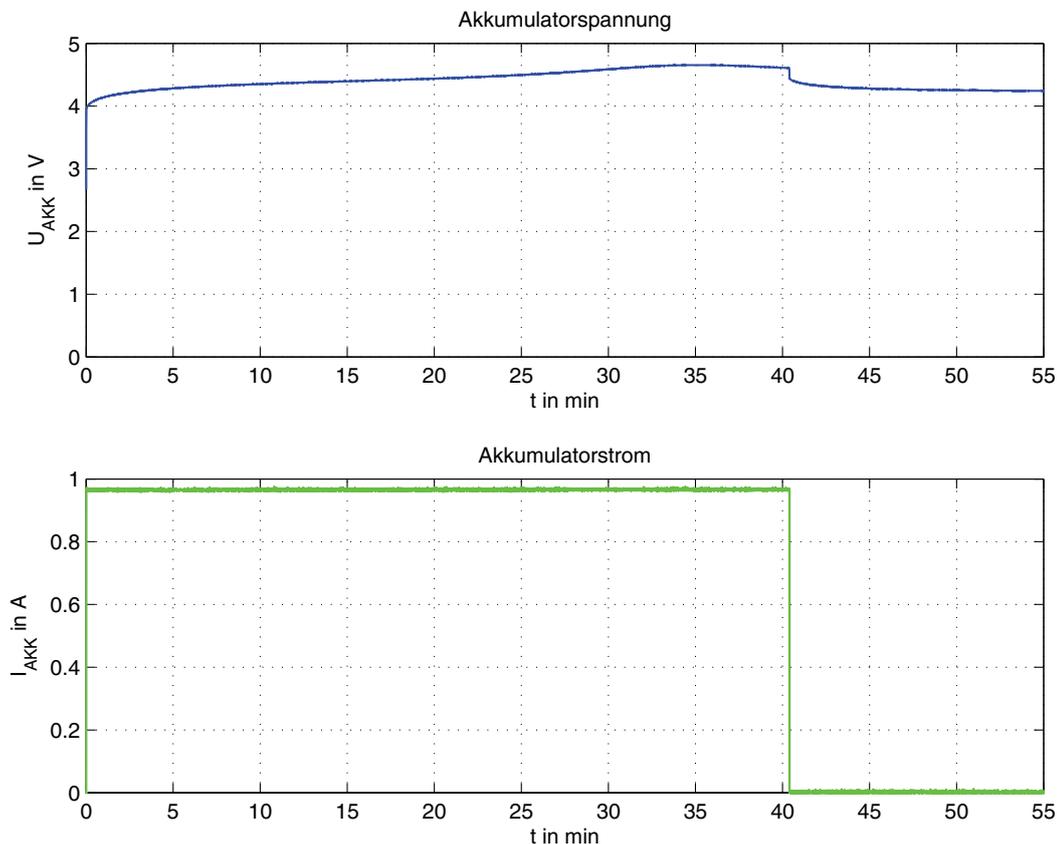


Abbildung 5.29: Ladezyklus des LTC4011 von drei in Serie geschalteten NiMH-Zellen

Die Umschaltung zwischen den Energiequellen Bordnetz und Akkumulator kann mittels des PowerTMPath-Controllers LTC4412 unter Verwendung von MOSFETs (siehe Abbildung 5.30) verlustarm durchgeführt werden. Für die Bereitstellung der Versorgungsspannung für die Ladeschaltung sowie die Ableitung der +5 V-Zwischenkreisspannung vom 24 V-Bordnetz wurden für den Prototypen die DC/DC-Konverter JTL3024S12 (Ausgangsspannung +12 V, 30 W) und JTH1524S05 (Ausgangsspannung +5 V, 15 W) der Firma XP Power gewählt. Der zulässige Eingangsspannungsbereich beider Konverter beträgt +9 bis +36 V, womit der Spannungsbereich des Bordnetzes problemlos abgedeckt werden kann. Hinsichtlich der zulässigen Ausgangsströme bieten beide Konverter ebenfalls eine deutliche Reserve nach oben.

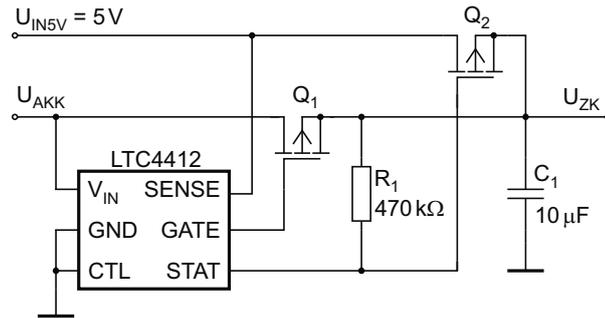


Abbildung 5.30: Schaltbild des PowerPath™Controllers LTC4412

Ein Foto dieses Prototypen zeigt Abbildung 5.31. Weiters sind in Abbildung 5.32 die Verläufe der Ausgangsspannung des JTH1524S05, der Akkumulatorspannung sowie der resultierenden Zwischenkreisspannung bei einem Ausfall der Bordspannung dargestellt. Bei einem Wegfall der Versorgungsspannung wird die Zwischenkreisspannung durch den Akkumulator gestützt. Sobald die Versorgungsspannung wieder einsetzt erfolgt automatisch die Ladung des Pufferakkumulators. Dieses Verhalten ist anhand des ebenfalls dargestellten Akkumulatorstromes deutlich nachvollziehbar. Als Last diente im Rahmen dieser Messungen der +3.3 V-Prototyp des LTC3112 bei einem Ausgangsstrom von 0.8 A.

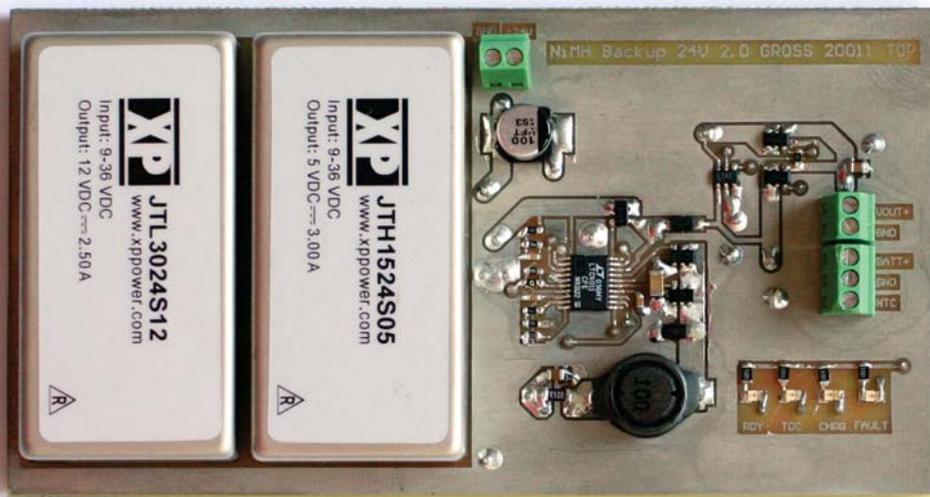


Abbildung 5.31: Foto der Ladeschaltung inklusive LTC4412

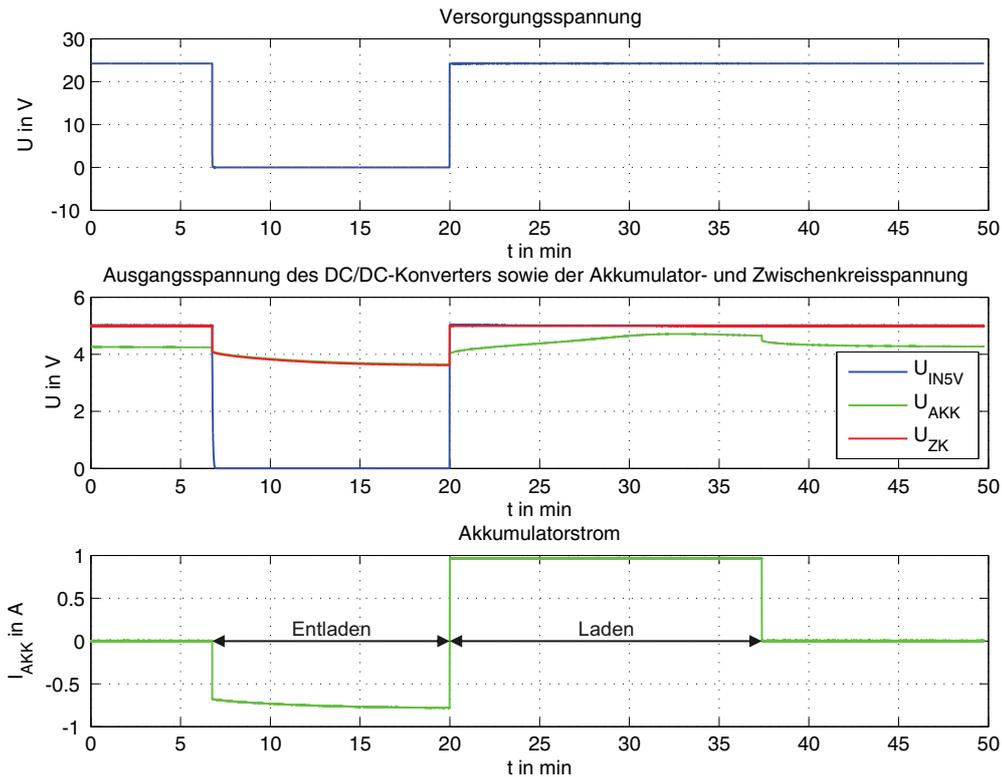


Abbildung 5.32: Spannungs- und Stromverläufe bei einem Ausfall der Versorgungsspannung

5.3 RX-/TX-Jitter-Kompensation des Z-Stack

Analog zur, in Abschnitt 4.2.2.3.2 beschriebenen, Messung der Latenzzeiten kann, durch Beobachtung der zwischen dem Mikrocontroller des Basismoduls und dem Funk-Transceiver CC2520 ausgetauschten Daten mittels eines zusätzlichen Mikrocontrollers oder programmierbaren Logikbausteins entsprechend Abbildung 5.33, der exakte Sendezeitpunkt von Nachrichten festgestellt werden. Da der Aufbau, der über die SPI-Schnittstelle zwischen den beiden ICs ausgetauschten Datenpakete bekannt ist, können die vom Datenspeichermodul abgesetzten Nachrichten eindeutig identifiziert werden.

Dies ist insofern notwendig, da vom Basismodul unabhängig vom Datenspeichermodul Funk-Broadcasts automatisch weitergeleitet werden und Antworten auf Netzwerkmanagementanfragen ebenfalls automatisch abgesendet werden. Es kann somit der Fall eintreten, dass unmittelbar nach oder während der Übermittlung einer Synchronisierungsmeldung vom Datenspeichermodul an das Basismodul (RS232Schnittstelle), eine solche automatisch generierte Nachricht an den CC2520 weitergegeben wird. Wird nach der Ge-

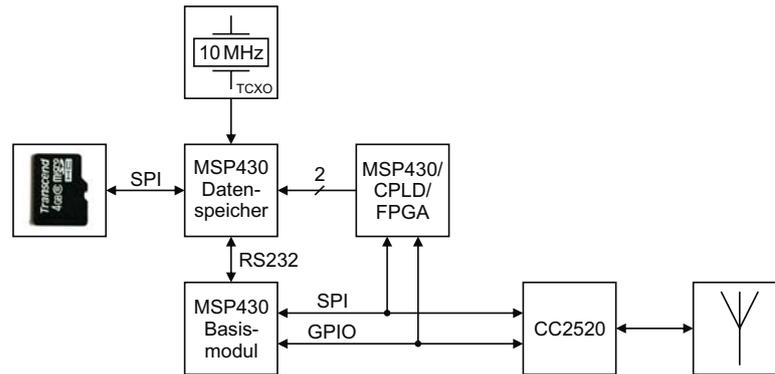


Abbildung 5.33: Hardwaremodifikation des Datenspeichers (Variante 1)

nerierung einer Synchronisierungsnachricht durch das Datenspeichermodul lediglich der Zeitpunkt der Übertragung der nächsten Funknachricht an den Funk-Transceiver detektiert, so könnten hier die Sendezeitpunkte von Nachrichten, die nicht der Synchronisation dienen, registriert werden und somit falsche Korrekturwerte bestimmt werden.

Dieser tatsächliche Sende- und Empfangszeitpunkt könnte beispielsweise über zwei interruptfähige Eingänge dem Datenspeicher-Mikrocontrollers mitgeteilt werden, wobei hier die maximale Latenzzeit bis zum Aufruf der entsprechenden Interrupt-Service-Routine für die erzielbare Genauigkeit ausschlaggebend ist.

Weiters wäre es möglich die gesamte Verwaltung der Zeitbasis in den externen Mikrocontroller bzw. ein CPLD/FPGA auszulagern, wodurch die Problematik der Latenzzeiten bei der Detektion der Sende- und Empfangszeitpunkte entfallen würde. Es wäre hierbei jedoch eine komplexere Schnittstelle zwischen dem Mikrocontroller des Datenspeichers und dem Baustein, der die Zeitbasis verwaltet, erforderlich. Diese Variante ist in Abbildung 5.34 dargestellt.

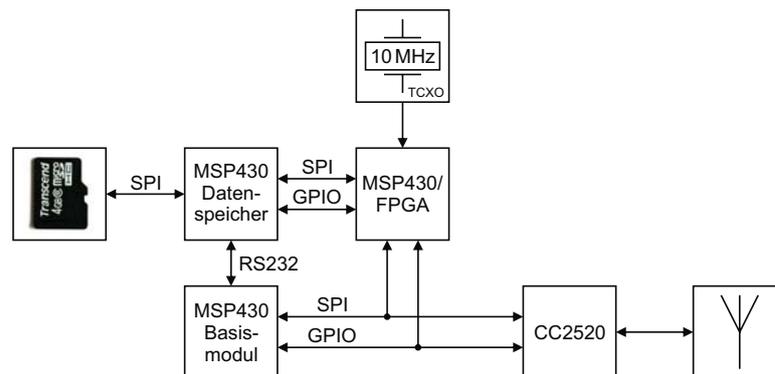


Abbildung 5.34: Hardwaremodifikation des Datenspeichers (Variante 2)

Der prinzipielle Ablauf zur Bestimmung des Gangunterschiedes $\Delta t_{SE} = t_{1E} - t_{1S}$ der Zeitbasen zweier Messknoten ist in Abbildung 5.35 dargestellt. Wie bereits in Abschnitt 4.2.2.3.2 erläutert, ist die Ausführungszeit dieses Schrittes mit einer vergleichsweise hohen Standardabweichung behaftet (vergleiche Tabelle 4.7). Dieser Umstand ist in Abbildung 5.35 durch den grau hinterlegten Bereich markiert. Da in Relation zum Verarbeitungsprozess durch den Z-Stack die übrigen Abläufe nur einen geringen Einfluss auf die bei der Bestimmung des Gangunterschiedes Δt_{SE} erzielbare Genauigkeit haben, wurden die zugehörigen Ausführungszeiten bei der Darstellung der Vorgänge in Abbildung 5.35 vernachlässigt und folglich als waagrechte Strecken zwischen Output und Input dargestellt.

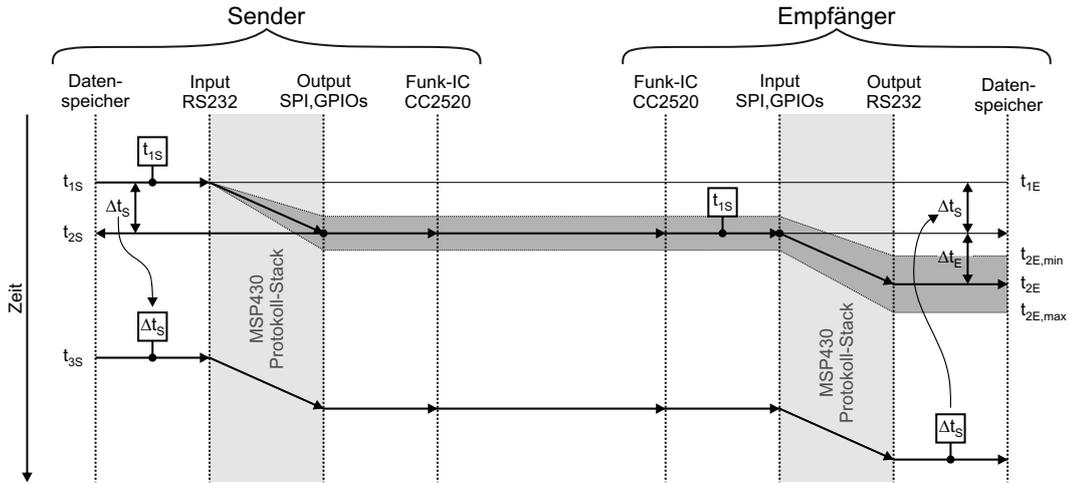


Abbildung 5.35: RX-/TX-Jitter-Kompensation

Zum Zeitpunkt t_{S1} wird durch den Mikrocontroller des Datenspeichers vom Sendeknoten der aktuelle Stand der Zeitbasis t_{S1} als Datenpaket an das angeschlossene Basismodul weitergeleitet. Durch den Z-Stack wird anschließend ein entsprechendes Datenpaket generiert, das dem Funk-Transceiver übergeben wird. Dieses wird durch den CC2520 per Funk an den Empfängerknoten übertragen, wobei hier analoge Vorgänge in umgekehrter Reihenfolge ablaufen. Die Information t_{1S} steht folglich dem Mikrocontroller des Datenspeichers zur Verfügung, wobei die seit dem Absenden der Information verstrichene Zeit $t_{1S} - t_{2E}$ zu diesem Zeitpunkt noch unbekannt ist. Lediglich Δt_E kann beim Empfängerknoten direkt bestimmt werden. Die Information hinsichtlich des Intervalls Δt_S steht jedoch im Sendeknoten zur Verfügung und kann zu einem späteren Zeitpunkt t_{3S} per Funk an den Empfängerknoten übertragen werden. Somit kann nun mittels

$$\Delta t_{SE} = t_{2E} - \Delta t_E - \Delta t_S - t_{1S} \quad (5.4)$$

der Gangunterschied bestimmt und die Zeitbasis des Empfängers korrigiert werden.

Mit der beschriebenen Methode können mit Ausnahme der Verarbeitungszeiten in den Funktransceivern und der eigentlichen Laufzeit des Funksignals prinzipiell alle anderen Verzögerungen erfasst werden. Es ist jedoch anzumerken, dass die hierbei erzielbare Präzision natürlich wesentlich von der Genauigkeit der für die Bestimmung des Gangunterschiedes gemessenen Intervalle Δt_E und Δt_S abhängt.

6 Zusammenfassung

Im Rahmen dieser Arbeit erfolgte die Entwicklung von Hard- und Softwarekomponenten eines Messsystems für die betriebliche, messtechnische Online-Überwachung von Schienenfahrzeugen. Die einzelnen Funktionsmuster wurden im Zuge der Entwicklung unter Laborbedingungen getestet und erfüllten die an sie gestellten funktionalen Anforderungen. Weiters wurde ein erfolgreicher Test der Netzwerkfunktionalität und der zuverlässigen Datenspeicherung unter realen Bedingungen im Rahmen einer Messfahrt am Betriebsgelände der Firma Bombardier in Wien durchgeführt.

Mit Abschluss des Entwicklungsprojekts stehen somit die wesentlichen Schaltungen für die Implementierung des Messsystems zur Verfügung. Zusätzlicher Entwicklungsbedarf besteht allerdings noch hinsichtlich der Einbindung von GPS-Empfängern und einer Schnittstelle zum Fahrzeugbus. Schaltungstechnische Maßnahmen zur Erfüllung der einschlägigen Normen (elektromagnetische Verträglichkeit, etc.) sind in weiterer Folge ebenfalls noch vorzusehen.

Die Software des Laborsystems bietet ebenfalls bereits die grundlegende Funktionalität. Für einen Einsatz des Systems in der Praxis muss die Bedienung des Systems allerdings benutzerfreundlicher ausgeführt werden. Dies umfasst neben der Entwicklung einer grafischen Benutzeroberfläche ebenfalls die Implementierung von zusätzlichen Funktionen in der Firmware der Knoten. Die Speicherung von Kalibrierdaten in den Flash-Speichern der Sensorinterfaces, sowie das Abspeichern dieser Daten zusammen mit den eigentlichen Messdaten auf den Speicherkarten wäre beispielsweise eine solche Funktion.

In Summe hat sich das, im Rahmen des Laborsystems umgesetzte und getestete, Lösungskonzept bewährt und der Einsatz der Technologie zur betrieblichen, messtechnischen Online-Überwachung von Schienenfahrzeugen erscheint in hohem Maße praktikabel.

Das im Rahmen dieser Arbeit entwickelte und erprobte Konzept ist jedoch auf Grund seiner Flexibilität und Leistungsfähigkeit nicht auf den Einsatz in Schienenfahrzeugen beschränkt und kann auf Grund des Verzichts von Kabelverbindungen bei einer Vielzahl von messtechnischen Aufgabenstellungen vorteilhaft eingesetzt werden. Speziell wenn die Aufgabenstellung eine hohe Anzahl von Sensoren erforderlich macht und die Verwendung von

Kabelverbindungen aus technischer (z.B. Messungen an bewegten Komponenten) oder betrieblicher Sicht (z.B. Behinderung von Personen oder bei Gefahr von Vandalismus) nicht möglich oder erwünscht ist, können konventionelle Messsystemen oft nicht mehr bzw. nur mit Kompromissen eingesetzt werden. Darüber hinaus kann in vielen Fällen eine schnellere Installation durch den Wegfall der Verkabelung einen wesentlichen Kostenfaktor darstellen. Speziell wenn eine solche Messung zur Analyse eines konkreten Problems zeitnah erfolgen muss. In all diesen Fällen kann ein, nach dem vorgestellten Konzept, ausgeführtes Messsystem die Lösung der messtechnischen Aufgabenstellung entweder überhaupt erst ermöglichen oder zumindest aus der Sicht des Anwenders die bessere und/oder billigere Lösung darstellen.

A Messdaten

Platine	P_{TX} in dBm	RX Pakete	P_{RX} in dBm
Nr. 2	+4	94 230	-100
	+4	93 835	-99
	+4	93 044	-99
	+4	94 852	-98
	+4	93 761	-99
	-4	88 622	-104
	-4	87 475	-103
	-4	51 651	-104
	-4	62 738	-103
	-4	82 516	-103
Nr. 3	+4	88 724	-100
	+4	90 016	-102
	+4	92 081	-102
	+4	92 450	-102
	+4	92 668	-103
	-4	27 769	-103
	-4	67 765	-103
	-4	1697	-105
	-4	70 647	-103
	-4	75 348	-103

Tabelle A.1: Ergebnisse der PER-Messungen der CC2520EM-Platinen Nummer 2 und 3 (P_{TX} : gewählte Sendeleistung; RX Pakete: Anzahl der empfangenen Pakete; P_{RX} : angezeigte mittlere Empfangsleistung der letzten 32 Pakete)

Ausführung	P_{TX} in dBm	RX Pakete	P_{RX} in dBm
FR4 0.8 mm	+4	95 779	-87
	+4	94 690	-87
	+4	99 112	-94
	+4	97 783	-94
	+4	95 568	-96
	-4	95 922	-93
	-4	95 823	-93
	-4	95 993	-93
	-4	96 035	-94
	-4	96 741	-94
FR4 1.5 mm	+4	99 140	-73
	+4	99 113	-73
	+4	99 129	-73
	+4	99 142	-74
	+4	99 139	-74
	-4	96 489	-81
	-4	96 519	-81
	-4	96 388	-81
	-4	95 464	-82
	-4	95 317	-81

Tabelle A.2: Ergebnisse der PER-Messungen der HF-Prototypen (P_{TX} : gewählte Sendeleistung; RX Pakete: Anzahl der empfangenen Pakete; P_{RX} : angezeigte mittlere Empfangsleistung der letzten 32 Pakete)

t_{BS} in ms	t_{CC} in μs	t_{SE} in ms	t_{SS} in ms
1.126	3.358	2.269	2.768
1.766	3.274	2.299	2.795
1.154	3.428	2.315	3.118
1.152	3.406	2.268	3.403
1.471	3.382	2.266	4.364
1.767	3.262	2.261	3.380
2.407	3.334	2.258	4.036
1.150	3.366	2.277	2.777
1.127	3.290	2.298	3.740
1.148	3.336	2.270	4.048
1.767	3.380	2.274	2.769
3.066	3.310	2.261	4.360
3.047	3.352	2.310	3.113
1.767	3.340	2.250	2.729
2.427	3.366	2.237	3.376
1.470	3.332	2.304	3.438
1.127	3.364	2.278	4.057
1.148	3.292	2.257	4.351
1.126	3.312	2.255	2.750
2.747	3.352	2.289	3.412
2.427	3.342	2.287	2.766
1.447	3.298	2.272	4.046
1.147	3.372	2.267	3.082
2.086	3.388	2.289	4.388
2.747	3.376	2.289	4.388

Tabelle A.3: Gemessene Zeitintervalle beim Senden und Empfangen von Nachrichten mittels Z-Stack

512 B	1024 B	2048 B	4096 B	6144 B
167.337	123.368	103.560	94.299	91.163
165.360	123.566	103.198	94.294	91.056
164.735	124.524	103.061	94.129	91.501
166.188	123.303	103.043	94.114	90.882
165.394	123.657	103.226	94.304	91.164
166.491	123.256	103.693	94.138	90.890
165.196	123.261	103.384	94.470	101.175
165.286	123.240	103.043	94.135	91.054
165.718	123.097	103.242	94.146	91.174
165.702	123.102	104.003	94.177	91.344

Tabelle A.4: Zeiten, die für das Schreiben von 10 MB mittels DOSonChip bei verschiedenen Blockgrößen von 512 B bis 6144 B benötigt werden. Datenspeicher: Transcend 4 GB microSDHC-Karte (FAT32 mit 32 kB großen Zuordnungseinheiten).

5 kB/s	10kB/s	15 kB/s	20 kB/s	25 kB/s	30 kB/s	35 kB/s	40 kB/s	45 kB/s	50 kB/s
2048	2590	3908	5224	6589	7922	9348	10790	12260	12800
2048	2048	2048	2048	2048	2048	4714	2048	2232	2610
2048	2048	2048	2048	2048	2048	3278	2048	2358	2397
2048	2048	2048	2048	2048	2048	3278	2048	2238	2643
2048	2048	2048	2048	2048	2048	3276	2048	2048	2601
2048	2048	2048	2048	2048	2048	3836	2048	2230	2598
2048	2048	2048	2048	2048	2048	3276	2048	2232	2602
2048	2048	2048	2048	2048	2048	3276	2048	2234	2644
2048	2048	2048	2048	2048	2048	3358	2048	2234	2486
2048	2048	2048	2048	2048	2048	3278	2048	2226	2604

Tabelle A.5: Maximale Pufferfüllstände (Puffergröße 12800) bei Verwendung des DOnChip-Treiber-ICs für verschiedenen Echtzeitdatenraten. Datenspeicher: Kingston 4GB microSDHC-Karte (FAT32 mit 32kB großen Zuordnungseinheiten)

5 kB/s	10 kB/s	15 kB/s	20 kB/s
2048	2048	5110	2048
2048	2048	2048	12 800
2048	2048	9754	12 800
2048	5672	5616	12 800
2048	3710	8548	12 800
2048	3462	8516	12 140
2048	5014	5594	7476
2048	3702	8548	12 116
2048	4992	5566	12 242
2048	3744	8540	7530

Tabelle A.6: Maximale Pufferfüllstände (Puffergröße 12800) bei Verwendung des DOSonChip-Treiber-ICs für verschiedenen Echtzeitdatenraten. Datenspeicher: SanDisk 4 GB microSDHC-Karte (FAT32 mit 32 kB großen Zuordnungseinheiten)

5 kB/s	10 kB/s	15 kB/s	20 kB/s	25 kB/s	30 kB/s	35 kB/s	40 kB/s
2048	2140	3232	5014	5406	6672	7452	12 800
2048	2048	2048	2768	3911	5348	6578	8097
2048	2048	2048	2048	3998	5284	6536	8030
2048	2048	2048	2704	2048	5296	6616	7910
2048	2048	2048	2690	4056	5266	6588	6968
2048	2048	2048	2780	4048	5204	6564	7946
2048	2048	2048	2048	3934	5225	6514	7856
2048	2048	2048	2722	3932	5402	6648	7932
2048	2048	2048	2774	3996	5314	6646	7982
2048	2048	2048	2692	3916	5226	6584	8139

Tabelle A.7: Maximale Pufferfüllstände (Puffergröße 12800) bei Verwendung des DOSonChip-Treiber-ICs für verschiedenen Echtzeitdatenraten. Datenspeicher: Transcend 4 GB microSDHC-Karte (FAT32 mit 32 kB großen Zuordnungseinheiten)

5 kB/s	10 kB/s	15 kB/s	20 kB/s	25 kB/s
759	1083	1305	1547	9287
761	1531	6887	9583	12 800
771	1029	4501	7435	10 581
801	1589	4495	7529	10 871
765	1599	4443	7521	10 721
793	1573	4465	7601	10 605
795	1589	4425	7645	10 661
797	1731	4423	7467	10 699
759	1607	4463	7595	11 179
793	1579	4459	7567	10 617

Tabelle A.8: Maximale Pufferfüllstände (Puffergröße 12800) bei Verwendung des FAT32-Firmware Treibers für verschiedenen Echtzeitdatenraten. Datenspeicher: SanDisk 4 GB microSDHC-Karte (FAT32 mit 32 kB großen Zuordnungseinheiten)

5 kB/s	10 kB/s	15 kB/s	20 kB/s	25 kB/s	30 kB/s	35 kB/s
1023	3029	3089	4111	5767	6963	12 800
1015	2005	2995	4019	5047	6931	7603
1015	1995	3059	4777	5833	6935	7319
1013	2015	3679	4045	5553	6121	7595
629	2027	3299	4035	5181	6097	8023
627	2011	2991	4013	5087	6405	7705
1017	2013	3285	4023	5001	6095	7063
1017	2581	3667	4771	5715	6811	7341
1013	1993	2967	4835	5515	6111	8015
1017	1993	3009	4761	5977	6999	7695

Tabelle A.9: Maximale Pufferfüllstände (Puffergröße 12800) bei Verwendung des FAT32-Firmware Treibers für verschiedenen Echtzeitdatenraten. Datenspeicher: Transcend 4 GB microSDHC-Karte (FAT32 mit 32 kB großen Zuordnungseinheiten)

Modul	n_F	t_E in s
0	36 000 000 016	14 400.000 006
	36 000 000 016	14 400.000 006
	36 000 000 016	14 400.000 006
	36 000 000 016	14 400.000 006
	36 000 000 016	14 400.000 006
1	35 999 990 559	14 399.996 224
	35 999 990 554	14 399.996 222
	35 999 990 743	14 399.996 297
	35 999 990 440	14 399.996 176
	35 999 990 444	14 399.996 178
2	35 999 987 103	14 399.994 841
	35 999 985 375	14 399.994 150
	35 999 985 827	14 399.994 331
	35 999 987 123	14 399.994 849
	35 999 986 414	14 399.994 566
3	35 999 983 973	14 399.993 589
	35 999 983 858	14 399.993 543
	35 999 984 968	14 399.993 627
	35 999 984 228	14 399.993 691
	35 999 983 933	14 399.993 573
4	35 999 983 208	14 399.993 283
	35 999 982 893	14 399.993 157
	35 999 982 864	14 399.993 146
	35 999 982 939	14 399.993 176
	35 999 982 542	14 399.993 017

Tabelle A.10: Anzahl der Taktflanken n_F und Endstand der Zeitbasis t_E der Datenspeichermodule 0 bis 4 nach 4 h – ohne Geschwindigkeitskorrektur

Modul	n_F	t_E in s
0	36 000 000 016	14 400.000 006
	36 000 000 016	14 400.000 006
	36 000 000 016	14 400.000 006
	36 000 000 016	14 400.000 006
	36 000 000 016	14 400.000 006
1	36 000 000 035	14 400.000 014
	36 000 000 107	14 400.000 043
	35 999 999 908	14 399.999 963
	35 999 999 967	14 399.999 987
	35 999 999 942	14 399.999 977
2	35 999 999 787	14 399.999 787
	35 999 999 753	14 399.999 753
	35 999 999 887	14 399.999 887
	35 999 999 812	14 399.999 812
	35 999 999 728	14 399.999 728
3	35 999 999 978	14 399.999 991
	36 000 000 026	14 400.000 010
	35 999 999 969	14 399.999 988
	35 999 999 981	14 399.999 992
	35 999 999 923	14 399.999 969
4	35 999 999 831	14 399.999 932
	35 999 999 904	14 399.999 962
	35 999 999 720	14 399.999 888
	35 999 999 968	14 399.999 987
	35 999 999 868	14 399.999 947

Tabelle A.11: Anzahl der Taktflanken n_F und Endstand der Zeitbasis t_E der Datenspeichermodule 0 bis 4 nach 4 h – mit Geschwindigkeitskorrektur

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
5.000	20.32	3.300	10.09	0.102	0.033	0.328
5.000	45.96	3.300	50.06	0.230	0.165	0.719
5.000	78.17	3.300	100.0	0.391	0.330	0.844
5.000	111.8	3.300	150.0	0.559	0.495	0.886
5.000	148.3	3.300	200.0	0.742	0.660	0.890
5.000	183.8	3.300	250.2	0.919	0.826	0.898
5.000	218.7	3.300	300.0	1.094	0.990	0.905
5.000	254.4	3.300	350.4	1.272	1.156	0.909
5.000	289	3.300	400.0	1.445	1.320	0.913
5.000	325	3.300	450.0	1.625	1.485	0.914
5.000	361	3.300	500.0	1.805	1.650	0.914
5.000	398	3.300	550	1.990	1.815	0.912
5.000	435	3.300	600	2.175	1.980	0.910
5.000	471	3.300	650	2.355	2.145	0.911
5.000	508	3.300	700	2.540	2.310	0.909
5.000	546	3.300	750	2.730	2.475	0.907
5.000	508	3.300	800	2.920	2.640	0.904
5.000	621	3.300	850	3.105	2.805	0.903
5.000	661	3.300	900	3.305	2.970	0.899
5.000	698	3.300	950	3.490	3.135	0.898
5.000	737	3.300	1000	3.685	3.300	0.896

Tabelle A.12: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = +3.3\text{ V}$ und $U_E = +5.0\text{ V}$

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
3.602	19.69	3.300	10.09	0.071	0.033	0.469
3.599	54.81	3.300	50.06	0.197	0.165	0.837
3.599	103.5	3.300	100.0	0.372	0.330	0.886
3.602	152.2	3.300	150.0	0.548	0.495	0.903
3.601	201.1	3.300	200.0	0.724	0.660	0.911
3.600	250.8	3.300	250.2	0.903	0.826	0.914
3.599	300.3	3.300	300.0	1.080	0.990	0.916
3.601	350.5	3.300	350.4	1.260	1.160	0.916
3.599	401	3.300	400.0	1.440	1.320	0.915
3.599	451	3.300	450.0	1.620	1.490	0.915
3.599	503	3.300	500.0	1.810	1.650	0.911
3.600	555	3.300	550	2.000	1.820	0.908
3.601	607	3.300	600	2.190	1.980	0.906
3.603	659	3.300	650	2.370	2.150	0.903
3.604	711	3.300	700	2.560	2.310	0.901
3.599	766	3.300	750	2.760	2.480	0.898
3.600	820	3.300	800	2.950	2.640	0.894
3.599	876	3.300	850	3.150	2.810	0.890
3.599	931	3.300	900	3.350	2.970	0.886
3.599	986	3.300	950	3.550	3.140	0.883
3.599	1043	3.300	1000	3.750	3.300	0.879

Tabelle A.13: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = +3.3V$ und $U_E = +3.6V$

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
2.703	21.90	3.300	10.04	0.059	0.033	0.560
2.700	70.54	3.300	50.14	0.190	0.165	0.869
2.699	136.0	3.300	100.1	0.367	0.330	0.900
2.703	204.3	3.300	149.9	0.552	0.495	0.896
2.703	273.8	3.300	200.0	0.740	0.660	0.892
2.699	347	3.300	250.0	0.937	0.825	0.881
2.700	419	3.300	300.0	1.131	0.990	0.875
2.699	492	3.300	350.0	1.328	1.155	0.870
2.701	565	3.300	400.0	1.526	1.320	0.865
2.703	642	3.300	450.0	1.735	1.485	0.856
2.700	720	3.300	500.0	1.944	1.650	0.849
2.699	798	3.300	550	2.154	1.815	0.843
2.700	878	3.300	600	2.371	1.980	0.835
2.701	962	3.300	650	2.598	2.145	0.826
2.701	1048	3.300	700	2.831	2.310	0.816
2.700	1136	3.300	750	3.067	2.475	0.807
2.699	1230	3.300	800	3.320	2.640	0.795
2.703	1323	3.300	850	3.576	2.805	0.784
2.700	1428	3.300	900	3.856	2.970	0.770
2.699	1556	3.300	950	4.200	3.135	0.746
2.700	1734	3.300	1000	4.682	3.300	0.705

Tabelle A.14: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = +3.3V$ und $U_E = +2.7V$

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
5.000	17.92	2.560	10.14	0.090	0.026	0.290
5.000	37.74	2.560	50.01	0.189	0.128	0.678
5.000	63.07	2.560	100.1	0.315	0.256	0.813
5.000	89.31	2.560	150.1	0.447	0.384	0.860
5.000	118.0	2.560	200.0	0.590	0.512	0.868
5.000	145.8	2.560	250.4	0.729	0.641	0.879
5.000	173.6	2.560	300.1	0.868	0.768	0.885
5.000	201.4	2.560	350.0	1.007	0.896	0.890
5.000	229	2.560	400.0	1.145	1.024	0.894
5.000	257	2.560	450.0	1.285	1.152	0.896
5.000	286	2.560	500.1	1.430	1.280	0.895
5.000	314	2.560	550	1.570	1.408	0.897
5.000	343	2.560	600	1.715	1.536	0.896
5.000	373	2.560	650	1.865	1.664	0.892
5.000	404	2.560	700	2.020	1.792	0.887
5.000	432	2.560	750	2.160	1.920	0.889
5.000	463	2.560	800	2.315	2.048	0.885

Tabelle A.15: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = +2.5\text{ V}$ und $U_E = +5.0\text{ V}$

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
3.604	18.14	2.560	10.10	0.065	0.026	0.395
3.601	45.53	2.560	50.05	0.164	0.128	0.781
3.604	81.13	2.560	100.1	0.292	0.256	0.876
3.600	120.2	2.560	150.1	0.433	0.384	0.888
3.602	158.3	2.560	200.0	0.570	0.512	0.898
3.602	197.3	2.560	250.4	0.711	0.641	0.902
3.603	235.8	2.560	300.1	0.850	0.768	0.904
3.600	275	2.560	350.0	0.990	0.896	0.905
3.589	314	2.560	400.0	1.127	1.024	0.909
3.600	354	2.560	450.0	1.274	1.152	0.904
3.602	394	2.560	500.1	1.419	1.280	0.902
3.602	434	2.560	550	1.563	1.408	0.901
3.598	475	2.560	600	1.709	1.536	0.899
3.602	517	2.560	650	1.862	1.664	0.894
3.602	560	2.560	700	2.017	1.792	0.888
3.601	599	2.560	750	2.157	1.920	0.890
3.599	643	2.560	800	2.314	2.048	0.885

Tabelle A.16: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = +2.5\text{ V}$ und $U_E = +3.6\text{ V}$

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
2.702	17.36	2.560	10.14	0.047	0.026	0.553
2.703	54.03	2.560	49.99	0.146	0.128	0.876
2.703	106.1	2.560	99.9	0.287	0.256	0.892
2.703	158.1	2.560	150.1	0.427	0.384	0.899
2.699	210.7	2.560	200.0	0.569	0.512	0.900
2.699	264.3	2.560	250.4	0.713	0.641	0.899
2.699	318.2	2.560	300.1	0.859	0.768	0.895
2.699	372.1	2.560	350.0	1.004	0.896	0.892
2.703	427	2.560	400.0	1.154	1.024	0.887
2.700	486	2.560	450.0	1.312	1.152	0.878
2.703	542	2.560	500.1	1.465	1.280	0.874
2.703	597	2.560	550	1.614	1.408	0.873
2.700	656	2.560	600	1.771	1.536	0.867
2.700	715	2.560	650	1.931	1.664	0.862
2.702	774	2.560	700	2.091	1.792	0.857
2.699	843	2.560	750	2.275	1.920	0.844
2.701	900	2.560	800	2.431	2.048	0.842

Tabelle A.17: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = +2.5\text{ V}$ und $U_E = +2.7\text{ V}$

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
5.000	19.60	-2.500	10.16	0.098	0.025	0.259
5.000	44.85	-2.500	50.06	0.224	0.125	0.558
5.000	75.48	-2.500	100.1	0.377	0.250	0.663
5.000	107.0	-2.500	150.0	0.535	0.375	0.701
5.000	139.5	-2.500	200.0	0.698	0.500	0.717
5.000	172.5	-2.500	250.3	0.863	0.626	0.726
5.000	203.1	-2.500	300.1	1.016	0.750	0.739
5.000	236	-2.500	350.0	1.180	0.875	0.742
5.000	268	-2.500	400.0	1.340	1.000	0.746
5.000	300	-2.490	450.0	1.500	1.121	0.747
5.000	333	-2.490	500.0	1.665	1.245	0.748
5.000	367	-2.490	550	1.835	1.370	0.746
5.000	402	-2.490	600	2.010	1.494	0.743
5.000	436	-2.490	650	2.180	1.619	0.742
5.000	473	-2.490	700	2.365	1.743	0.737
5.000	509	-2.490	750	2.545	1.868	0.734
5.000	547	-2.490	800	2.735	1.992	0.728

Tabelle A.18: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = -2.5\text{ V}$ und $U_E = +5.0\text{ V}$

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
3.601	22.14	-2.500	10.16	0.080	0.025	0.319
3.599	57.14	-2.500	50.06	0.206	0.125	0.609
3.600	101.3	-2.500	100.1	0.365	0.250	0.686
3.604	145.2	-2.500	150.0	0.523	0.375	0.717
3.599	190.1	-2.500	200.0	0.684	0.500	0.731
3.600	235.2	-2.490	250.3	0.847	0.623	0.736
3.603	279.2	-2.490	300.1	1.006	0.747	0.743
3.599	325	-2.490	350	1.170	0.872	0.745
3.600	372	-2.490	400	1.339	0.996	0.744
3.603	419	-2.490	450	1.510	1.121	0.742
3.602	468	-2.490	500	1.686	1.245	0.739
3.601	518	-2.490	550	1.865	1.370	0.734
3.599	569	-2.490	600	2.048	1.494	0.730
3.604	622	-2.490	650	2.242	1.619	0.722
3.599	680	-2.490	700	2.447	1.743	0.712
3.599	737	-2.490	750	2.652	1.868	0.704
3.600	794	-2.490	800	2.858	1.992	0.697

Tabelle A.19: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = -2.5\text{ V}$ und $U_E = +3.6\text{ V}$

U_E/V	I_E/mA	U_A/V	I_A/mA	P_E/W	P_A/W	η
2.700	25.30	-2.5	10.16	0.068	0.025	0.372
2.701	71.64	-2.5	50.06	0.193	0.125	0.647
2.699	131.6	-2.5	100.1	0.355	0.250	0.705
2.699	190.8	-2.49	150.0	0.515	0.374	0.725
2.700	250.2	-2.49	200.0	0.676	0.498	0.737
2.699	311.2	-2.49	250.3	0.840	0.623	0.742
2.703	375	-2.49	300.0	1.014	0.747	0.737
2.700	438	-2.49	350.0	1.183	0.872	0.737
2.702	502	-2.49	400.0	1.356	0.996	0.734
2.699	574	-2.49	450.0	1.549	1.121	0.723
2.703	645	-2.49	500.0	1.743	1.245	0.714
2.701	720	-2.48	550	1.945	1.364	0.701
2.703	797	-2.48	600	2.154	1.488	0.691
2.703	882	-2.48	650	2.384	1.612	0.676
2.699	965	-2.46	700	2.605	1.722	0.661
2.701	1043	-2.42	750	2.817	1.815	0.644
2.699	1124	-2.36	800	3.034	1.888	0.622

Tabelle A.20: Lastabhängiger Wirkungsgrad des LTC3112 bei $U_A = -2.5\text{ V}$ und $U_E = +2.7\text{ V}$

B Funktionen des Zustandsautomaten

Im Folgenden sind die einzelnen Funktionen des Zustandsautomaten für die Datenspeicherung (siehe Abschnitt 4.3.2.1) näher beschrieben.

IDLE

Das Modul befindet sich im Wartezustand. Mögliche Aktionen im Zustand IDLE:

- Systemzeit einstellen
(IDLE → TIME_SET)
- Standard-Antwortadresse festlegen
(IDLE → AC_SET_DEFAULT)
- Standard-Antwortadresse auslesen
(IDLE → AC_GET_DEFAULT)
- Status des Moduls abfragen
(IDLE → STATUS_SEND)
- Synchronisation der Zeitbasis
(IDLE → TIMER_START)
- Abtastung starten
(IDLE → TIMER_CHECK → SAMPLING_START → SAMPLING_IDLE)

TIME_SET

Das Modul übernimmt die übergebene Systemzeit in das RTC-Modul (Real Time Clock) des Mikrocontrollers. Danach wechselt es erneut in den Zustand IDLE.

AC_SET_DEFAULT

Das Modul übernimmt die übergebene Adresse (inkl. ZigBee-Cluster) als Standard-Antwortadresse für den Versand von Fehlermeldungen bzw. die Übertragung des Live-

Datenstroms. Danach wechselt es erneut in den Zustand IDLE.

AC_GET_DEFAULT

Das Modul gibt die eingestellte Standardadresse zurück und wechselt danach erneut in den Zustand IDLE.

STATUS_SEND

Das Modul gibt den momentanen Status zurück und wechselt danach erneut in den Zustand IDLE.

SYNC_WAIT

Das Modul wartet auf den kabelgebundenen Synchronisationsimpuls. Wird vor dem Impuls ein neues Kommando empfangen, so wird die Synchronisation mit einer Fehlermeldung abgebrochen. Im Normalfall wird nach dem empfangenen Synchronisationssignal die Zeitbasis gestartet (Wechsel in den Zustand TIMER_START).

TIMER_START

Das Modul startet die interne Zeitbasis und wechselt danach in den Zustand IDLE zurück.

TIMER_CHECK

Beim Starten der Abtastung wird zunächst überprüft ob die Zeitbasis aktiv ist. Ist dies nicht der Fall, wird eine Fehlermeldung ausgegeben und es erfolgt der Rücksprung in den Zustand IDLE. Bei aktiver Zeitbasis erfolgt der Übergang in den Zustand SAMPLING_START.

SAMPLING_START

Das Modul konfiguriert in diesem Zustand den ADC des Mikrocontrollers. Bei fehlerhaften Konfigurationsdaten erfolgt die Ausgabe einer Fehlermeldung und der Rücksprung in den Zustand IDLE. Bei erfolgreicher Konfiguration des ADC wechselt das Modul in den Zustand SAMPLING_IDLE.

SAMPLING_IDLE

Im Zustand SAMPLING_IDLE ist die Abtastung aktiv und es können folgende Aktionen ausgeführt werden:

- Starten der Funkübertragung eines ADC-Kanals
(SAMPLING_IDLE → TRANSMIT_START)
- Stoppen der Funkübertragung eines ADC-Kanals
(SAMPLING_IDLE → TRANSMIT_STOP)
- Starten der Aufzeichnung auf die SD-Karte
(SAMPLING_IDLE → DS_REC_START)
- Stoppen der Aufzeichnung auf die SD-Karte
(SAMPLING_IDLE → DS_REC_STOP)
- Stoppen der Abtastung
(SAMPLING_IDLE → SAMPLING_STOP → IDLE)

TRANSMIT_START

Das Modul startet die Übertragung des übergebenen ADC-Kanals an die Standard-Antwortadresse und kehrt danach in den Zustand SAMPLING_IDLE zurück.

TRANSMIT_STOP

Das Modul stoppt die Übertragung des ADC-Kanals und kehrt danach in den Zustand SAMPLING_IDLE zurück.

DS_REC_START

Das Modul startet die Aufzeichnung aller konfigurierten ADC-Kanäle und kehrt danach in den Zustand SAMPLING_IDLE zurück.

DS_REC_STOP

Das Modul stoppt die Aufzeichnung der ADC-Kanäle und kehrt danach in den Zustand SAMPLING_IDLE zurück.

SAMPLING_STOP

Stoppt die Abtastung und wechselt in den Zustand IDLE, falls die Aufzeichnung auf SD-Karte und die Funkübertragung deaktiviert sind. Falls nicht, k das Modul in den Zustand SAMPLING_IDLE zurück und setzt eine Statusmeldung ab.

C Kommandocodes des Datenspeichermoduls

Im Folgenden werden die implementierten Befehlscodes (siehe Abschnitt 4.3.2) inklusive der zugehörigen Rückmeldungen des Datenspeichermoduls erläutert.

C.1 Systemzeit einstellen

SEQ	0x84	SEC	MIN	HRS	DOW	DAY	MON	YRH	YRL
-----	------	-----	-----	-----	-----	-----	-----	-----	-----

SEQ: Sequenznummer

SEC: Sekunden

MIN: Minuten

HRS: Stunden

DOW: Wochentag (Montag = 1)

DAY: Tag

MON: Monat

YRH: Jahr – Highbyte

YRL: Jahr – Lowbyte

Antwort:

- **Erfolgsmeldung**

SEQ	0xC1	0x84	CSN
-----	------	------	-----

CSN: Sequenznummer des empfangenen Befehlspakets

C.2 Standard-Antwortadresse festlegen

SEQ	0x81	DAH	DAL	DCL
-----	------	-----	-----	-----

SEQ: Sequenznummer

DAH: Adresse – Highbyte

DAL: Adresse – Lowbyte

DCL: Cluster

Antwort:

- **Erfolgsmeldung**

SEQ | 0xC1 | 0x81 | CSN

CSN: Sequenznummer des empfangenen Befehlspakets

C.3 Standard-Antwortadresse abfragen

SEQ | 0x82

SEQ: Sequenznummer

Antwort:

- **Standardadresse**

SEQ | 0xC5 | DAH | DAL | DCL

DAH: Adresse – Highbyte

DAL: Adresse – Lowbyte

DCL: Cluster

C.4 Status des Moduls abfragen

SEQ | 0x83

SEQ: Sequenznummer

Antwort:

- **Modulstatus**

SEQ | 0xC4 | SFL | MIH | MIL | MSH | MSL

SFL: Statusflags

TIM | SAM | REC | CHT | BOV | ADE | - | -

TIM: Status der Zeitbasis (1: Zeitbasis aktiv)

SAM: Status der Abtatsung (1: ADC aktiv)

REC: Status der Aufzeichnung (1: Aufzeichnung aktiv)

CHT: Status der Funkübertragung (1: ADC-Kanal wird übertragen)

BOV: Status des Aufnahmebuffers (1: Overflow)

ADE: ADC-Konfigurationsfehler (1: Fehler)

MIH: Minuten seit dem Start der Zeitbasis – Highbyte

MIL: Minuten seit dem Start der Zeitbasis – Lowbyte

MSH: Millisekunden seit dem Start der Zeitbasis – Highbyte

MSL: Millisekunden seit dem Start der Zeitbasis – Lowbyte

C.5 Synchronisierung der ADC-Zeitbasis

SEQ : 0x91

SEQ: Sequenznummer

Antworten:

- Modul wartet auf Synchronisierungssignal

SEQ : 0xD1

- Zeitbasis wurde synchronisiert

SEQ : 0xD2

- Synchronisierung abgebrochen (Fehlermeldung)

SEQ : 0xD3

C.6 Zeitbasis des ADCs per Funk starten

SEQ : 0x92

SEQ: Sequenznummer

Antworten:

- Zeitbasis wurde synchronisiert

SEQ : 0xD2

C.7 Abtastung starten

SEQ : 0xA1

SEQ: Sequenznummer

Antworten:

- Abtastung aktiv

SEQ : 0xE1

- Fehler bei der Konfiguration des ADCs

SEQ : 0xE2

- Zeitbasis wurde nicht gestartet (Fehlermeldung)

SEQ : 0xD3

C.8 Funkübertragung eines ADC-Kanals starten

SEQ | 0xA5 | CHN

SEQ: Sequenznummer

CHN: Nummer des zu übertragenden Kanals

Antworten:

- ADC-Werte (wird kontinuierlich gesendet)

SEQ | 0xE9 | CHN | CLK | SNO | BNO | DBH | DBL | ... | ... | DBH | DBL

CHN: Nummer des übertragenen Kanals

CLK: Divisionsfaktor der Sample-Clock

SNO: Anzahl der enthaltenen Abtastwerte

BNO: Puffer Nummer 1 oder 2

DBH: Abtastwert – Highbyte

DBL: Abtastwert – Lowbyte

DBH	1	0	ADC(11..6)	
	7	6	5	0

DBL	0	0	ADC(5..0)	
	7	6	5	0

- Abtastung wurde nicht gestartet (Fehlermeldung)

SEQ | 0xE3

- Ungültige Kanalnummer (Fehlermeldung)

SEQ | 0xEA

C.9 Analoge Signalaufbereitung konfigurieren

SEQ | 0xB2 | CHN | UBI | CLK | GAI | GND | RCS | RDH | RDL

SEQ: Sequenznummer

CHN: Nummer des Kanals

UBI: Uni- oder bipolarer Eingangsbereich

0x00: Vorgabe des RDAC-Werts mittels RDH und RDL

0x01: Unipolar Bereich

0x02: Bipolar Bereich

CLK: Wahl des Taktsignals des SC-Filters

0x00: Frequenz 18.7 kHz (Abtastrate 1 kS/s)

0x01: Frequenz 1.87 kHz (Abtastrate 100 S/s)

GAI: Wahl der Verstärkung

$$(GAI = C_G(2) \cdot 2^2 + C_G(1) \cdot 2^1 + C_G(0) \cdot 2^0)$$

GND: Auswahl zwischen dem Sensorsignal und dem Referenzpegel GND

0x00: Sensorsignal

0x01: Referenzpegel GND

RCS: Wahl des RC-Filters

0x00: Obere Grenzfrequenz 1.17 kHz (Abtastrate 1 kS/s)

0x01: Obere Grenzfrequenz 117 Hz (Abtastrate 100 S/s)

RDH: RDAC-Wert – Highbyte

RDL: RDAC-Wert – Lowbyte

Antworten:

- Konfiguration übernommen

SEQ | 0xC1 | 0xB2 | CSN

- Konfiguration fehlgeschlagen (Fehlermeldung)

SEQ | 0xF4

C.10 Konfiguration der analogen Signalaufbereitung auslesen

SEQ | 0xB3

SEQ: Sequenznummer

Antwort:

- Konfigurationsdaten

SEQ | 0xF3 | CHN | UBI | CLK | GAI | GND | RCS | RDH | RDL

Details siehe C.9 Analoge Signalaufbereitung konfigurieren

C.11 Funkübertragung eines ADC-Kanals stoppen

SEQ | 0xA6

SEQ: Sequenznummer

Antwort:

- Übertragung gestoppt

SEQ | 0xEB

C.12 Aufzeichnung auf SD-Karte starten

SEQ: 0xA3

SEQ: Sequenznummer

Antwort:

- Aufzeichnung läuft

SEQ: 0xE4

- Abtastung wurde nicht gestartet (Fehlermeldung)

SEQ: 0xE3

- Initialisierung der SD-Karte fehlgeschlagen (Fehlermeldung)

SEQ: 0xE5

- Datei konnte nicht erstellt werden (Fehlermeldung)

SEQ: 0xE6

C.13 Aufzeichnung stoppen

SEQ: 0xA4

SEQ: Sequenznummer

Antwort:

- Aufzeichnung gestoppt

SEQ: 0xE8

C.14 Abtastung stoppen

SEQ: 0xA2

SEQ: Sequenznummer

Antwort:

- Abtastung gestoppt

SEQ: 0xE3

- Statusmeldung (Fehlermeldung – Aufzeichnung und/oder Funkübertragung noch im Gang)

SEQ: 0xC4 SFL MIH MIL MSH MSL

Details siehe Abschnitt C.4 Status des Moduls abfragen

D Pinbelegungen der Module des Laborsystems

Im Folgenden sind in den Abschnitten D.1 bis D.6 die Pinbelegungen der Stecker und Buchsen sowie der Mikrocontroller der einzelnen Module des Laborsystems beschrieben. In Abschnitt D.7 sind darüber hinaus die Verbindungen zwischen den Modulen in Tabellenform dokumentiert. Folgende Abkürzungen werden in weiterer Folge verwendet:

E/A Ein- oder Ausgangsignal des Moduls (A → Ausgang, E → Eingang)

S/B Stecker- bzw. Buchsenpin

k.F. (derzeit) keine Funktion

D.1 Basismodul



Abbildung D.1: Bedienelemente, Stecker und Buchsen des Basismoduls

MSP430F2618	Bauteil	Bemerkung
XIN, XOUT	X1	32.768 kHz Uhrenquarz (MSP430-RTC)
P3.7 GPIO	CC2520	CC2520_CS _n
P5.1 UCB1MOSI	CC2520	CC2520_SI
P5.2 UCB1MISO	CC2520	CC2520_SO
P5.3 UCB1CLK	CC2520	CC2520_SCLK
P1.3 GPIO	CC2520	CC2520_GPIO0
P1.5 GPIO	CC2520	CC2520_GPIO1
P1.6 GPIO	CC2520	CC2520_GPIO2
P1.1 GPIO	CC2520	CC2520_GPIO3
P1.2 GPIO	CC2520	CC2520_GPIO4
P1.7 GPIO	CC2520	CC2520_GPIO5
P1.0 GPIO	CC2520	CC2520_VREG_EN
P5.7 GPIO	CC2520	CC2520_RESE _{Tn}
P3.6 GPIO	D2	LED – GPIO = low → LED an
P5.0 GPIO	D3	LED – GPIO = low → LED an
P3.0 GPIO	D4	LED – GPIO = low → LED an
P2.7 GPIO	D5	LED – GPIO = low → LED an
P2.1 GPIO	S3	Taster – Taster gedrückt → GPIO = low
P2.0 GPIO	S4	Taster – Taster gedrückt → GPIO = low
P1.4 GPIO	S5	Taster – Taster gedrückt → GPIO = low

Tabelle D.1: Basismodul – Interne Verbindungen des MSP430F2618

Pin J1	MSP430	E/A	Bemerkung
1	–	–	GND
2	–	–	GND
3	–	–	nicht belegt
4	–	–	nicht belegt
5	–	A	+3.3 V
6	–	A	+3.3 V
7	–	–	nicht belegt
8	P5.1	A	CC2520_SI
9	P5.2	A	CC2520_SO
10	P5.3	A	CC2520_SCLK
11	P1.0	A	CC2520_VREG_EN
12	P3.7	A	CC2520_CS _n
13	P4.0	E/A	GPIO/TB0
14	P4.1	E/A	GPIO/TB1
15	P4.2	E/A	GPIO/TB2
16	P4.3	E/A	GPIO/TB3
17	P4.4	E/A	GPIO/TB4
18	P4.5	E/A	GPIO/TB5
19	P4.6	E/A	GPIO/TB6
20	P4.7	E/A	GPIO/TBCLK
21	P5.4	E/A	GPIO/MCLK
22	RST _n	E	Reset des MSP430F2618
23	P5.5	E/A	GPIO/SMCLK
24	P6.0	E/A	GPIO/ADC0
25	P5.6	E/A	GPIO/ACLK
26	P6.1	E/A	GPIO/ADC1
27	P5.7	A	CC2520_RESE _{Tn}
28	P6.2	E/A	GPIO/ADC2
29	–	A	Batteriespannung
30	–	–	GND

Tabelle D.2: Basismodul – Pinbelegung Buchse J1

Pin J2	MSP430	E/A	Bemerkung
1	P1.2	A	CC2520_GPIO4
2	–	–	GND
3	P3.5	E/A	GPIO/UCA0RXD/UCA0MISO
4	P1.1	A	CC2520_GPIO3
5	P1.3	A	CC2520_GPIO0
6	P3.4	E/A	GPIO/UCA0TXD/UCA0MOSI
7	P3.2	E/A	GPIO/UCB0SOMI/UCB0SCL
8	P3.3	E/A	GPIO/UCB0UCLK/UCA0STE
9	P3.0	A	D4 (LED)
10	P3.1	E/A	GPIO/UCB0MOSI/UCB0SDA
11	P2.6	E/A	GPIO/ADC12CLKDMAE0/CA7
12	P2.7	A	D5 (LED)
13	P2.4	E/A	GPIO/CA1/TA2
14	P2.5	E/A	GPIO/ROSC/CA5
15	P2.2	E/A	GPIO/CAOUT/TA0/CA4
16	P2.3	E/A	GPIO/CA0/TA1
17	P6.7	E/A	GPIO/ADC7/DAC1
18	P2.1	A	S3 (Taster)
19	P6.6	E/A	GPIO/ADC6/DAC0
20	P2.0	A	S4
21	P6.5	E/A	GPIO/ADC5/DAC1
22	P1.7	A	CC2520_GPIO5
23	P6.4	E/A	GPIO/ADC4
24	P1.6	A	CC2520_GPIO2
25	P6.3	E/A	GPIO/ADC3
26	P1.5	A	CC2520_GPIO1
27	P1.4	A	S5 (Taster)
28	P3.6	A	D2 (LED)
29	–	GND	
30	P5.0	A	D3 (LED)

Tabelle D.3: Basismodul – Pinbelegung Buchse J2

D.2 Datenspeicher

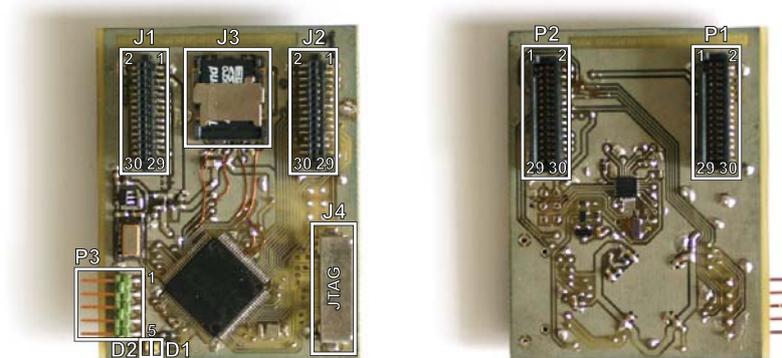


Abbildung D.2: Bedienelemente, Stecker und Buchsen des Datenspeichermoduls

MSP430F5438	Bauteil	Bemerkung
XIN, XOUT	X1	32.768 kHz Uhrenquarz (MSP430-RTC)
P4.7 TB0CLK	X2	10 MHz TXCO (Zeitbasis der Datenerfassung)
P3.0 GPIO	J3	microSD-Sockel Pin 2 – D_CS _n
P3.1 UCB0MOSI	J3	microSD-Sockel Pin 3 – D_SI
P3.2 UCB0MISO	J3	microSD-Sockel Pin 7 – D_SO
P3.3 UVB0CLK	J3	microSD-Sockel Pin 5 – D_SCLK
P9.1 GPIO	D1	LED – GPIO = low → LED an
P9.0 GPIO	D2	LED – GPIO = low → LED an

Tabelle D.4: Datenspeicher – Interne Verbindungen des MSP430F5438

Pin P1	MSP430F5438	E/A	Bemerkung
1	–	–	GND
2	–	–	GND
5	–	E	+3.3 V-Versorgungsspannung des Moduls
6	–	E	+3.3 V-Versorgungsspannung des Moduls
30	–	–	GND

Tabelle D.5: Datenspeicher – Pinbelegung Stecker P1 (nur belegte Anschlüsse)

Pin P2	MSP430	E/A	Bemerkung
2	–	–	GND
3	P5.6	E/A	GPIO/UCA1TXD/UCA1MOSI (D_TX)
6	P5.7	E/A	GPIO/UCA1RXD/UCA1MISO (D_RX)
8	P3.6	E/A	GPIO/UCA1CLK (k.F.)
14	P5.5	E/A	GPIO/UCA1STE (k.F.)
15	RSTn	E	Reset des MSP430F5438
16	P1.0	E/A	GPIO/TA0CLK/ACLK (k.F.)
29	–	–	GND

Tabelle D.6: Datenspeicher – Pinbelegung Stecker P2 (nur belegte Anschlüsse)

Pin J1	MSP430	E/A	Bemerkung
1	–	–	GND
2	–	–	GND
5	–	A	+3.3 V-Versorgungsspannung des Moduls
6	–	A	+3.3 V-Versorgungsspannung des Moduls
15	P1.4	E/A	GPIO/TA0.3
17	P1.6	E/A	GPIO/SMCLK
19	P1.7	E/A	GPIO
21	P2.0	E/A	GPOTA1CLK/MCLK
22	P10.4	E/A	GPIO/UCA3TXD/UCA3MOSI
23	P1.3	E/A	GPIO/TA0.2
24	P10.0	E/A	GPIO/UCA3CLK
25	P10.5	E/A	GPIO/UCA3RSD/UCA3MISO
26	P9.7	E/A	GPIO
30	–	–	GND

Tabelle D.7: Datenspeicher – Pinbelegung Buchse J1 (nur belegte Anschlüsse)

Pin J2	MSP430	E/A	Bemerkung
2	–	–	GND
12	P7.7	E/A	GPIO/ADC15
18	P7.6	E/A	GPIO/ADC14
19	P6.3	E/A	GPIO/ADC3
20	P7.5	E/A	GPIO/ADC13
21	P6.4	E/A	GPIO/ADC4
23	P6.5	E/A	GPIO/ADC5
25	P6.6	E/A	GPIO/ADC6
28	P7.4	E/A	GPIO/ADC12
29	–	–	GND
30	P6.7	E/A	GPIO/ADC7

Tabelle D.8: Datenspeicher – Pinbelegung Buchse J2 (nur belegte Anschlüsse)

D.3 Allgemeine Signalaufbereitung

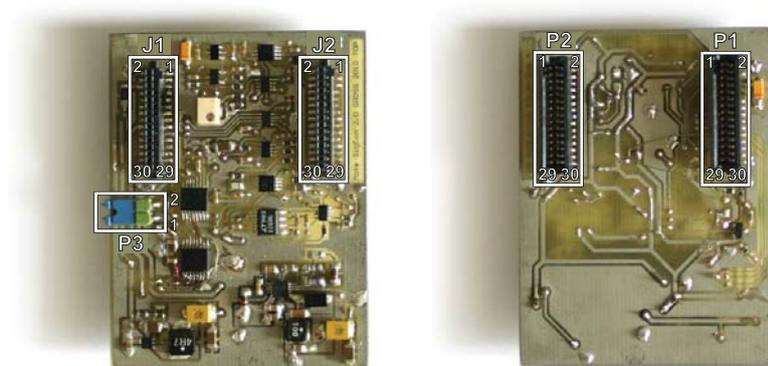


Abbildung D.3: Bedienelemente, Stecker und Buchsen der allgemeinen Signalaufbereitung

Pin P1	Netz	E/A	Bemerkung
1	GND	–	Signalmasse
2	GND	–	Signalmasse
5	+3.3 V	E	+3.3 V-Versorgungsspannung des Moduls
6	+3.3 V	E	+3.3 V-Versorgungsspannung des Moduls
15	C_G0	E	Verstärkung des LTC6910 – Bit 0 (LSB)
17	C_G1	E	Verstärkung des LTC6910 – Bit 1
19	C_G2	E	Verstärkung des LTC6910 – Bit 2 (MSB)
21	C_RDY	A	Ready-Signal des AD5231
22	C_SPI_SDI	E	Serieller Dateneingang des AD5231
23	C_CLK	E	Taktsignal des SC-Filters LTC1069-6
24	C_SPI_CLK	E	SPI-Taktsignal des AD5231
25	C_SPI_SDO	E	Serieller Datenausgang des AD5231
26	C_SPI_CS _n	E	Chip-Select-Signal des AD5231
30	GND	–	Signalmasse

Tabelle D.9: Allgemeine Signalaufbereitung – Pinbelegung Stecker P1 (nur belegte Anschlüsse)

Pin P2	Netz	E/A	Bemerkung
2	GND	–	Signalmasse
19	U _{CO}	A	Analoges Ausgangssignal
29	GND	–	Signalmasse

Tabelle D.10: Allgemeine Signalaufbereitung – Pinbelegung Stecker P2 (nur belegte Anschlüsse)

Pin J1	Netz	E/A	Bemerkung
1	GND	–	Signalmasse
2	GND	–	Signalmasse
5	+3.3 V	A	+3.3 V-Versorgungsspannung des Moduls
6	+3.3 V	A	+3.3 V-Versorgungsspannung des Moduls
17	+2.5 V	A	+2.5 V-Versorgungsspannung des Moduls
18	+2.5 V	A	+2.5 V-Versorgungsspannung des Moduls
19	–2.5 V	A	–2.5 V-Versorgungsspannung des Moduls
20	–2.5 V	A	–2.5 V-Versorgungsspannung des Moduls
30	GND	–	Signalmasse

Tabelle D.11: Allgemeine Signalaufbereitung – Pinbelegung Buchse J1 (nur belegte Anschlüsse)

Pin J2	Netz	E/A	Bemerkung
2	GND	–	Masse
6	U _{CI}	E	Analoges Eingangssignal
29	GND	–	Masse

Tabelle D.12: Allgemeine Signalaufbereitung – Pinbelegung Buchse J2 (nur belegte Anschlüsse)

Pin P3	Netz	E/A	Bemerkung
1	+2.5 V	A	+2.5 V-Versorgungsspannung des Moduls
2	C_WPn	E	Write-Protect-Signal des AD5231 (Jumper vorhanden → C_WPn = 1)

Tabelle D.13: Allgemeine Signalaufbereitung – Jumper P3

D.4 IEPE-Interface

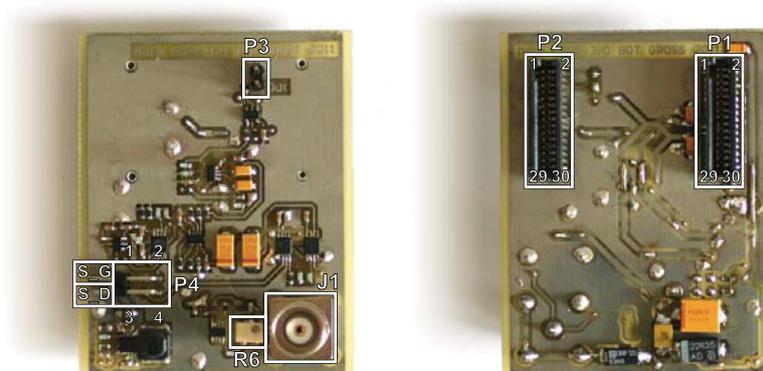


Abbildung D.4: Bedienelemente, Stecker und Buchsen des IEPE-Interfaces

Pin P1	Netz	E/A	Bemerkung
1	GND	–	Masse
2	GND	–	Masse
5	+3.3 V	E	+3.3 V-Versorgungsspannung des Moduls
6	+3.3 V	E	+3.3 V-Versorgungsspannung des Moduls
17	+2.5 V	E	+2.5 V-Versorgungsspannung des Moduls
18	+2.5 V	E	+2.5 V-Versorgungsspannung des Moduls
19	–2.5 V	E	–2.5 V-Versorgungsspannung des Moduls
20	–2.5 V	E	–2.5 V-Versorgungsspannung des Moduls
30	GND	–	Masse

Tabelle D.14: IEPE-Interface – Pinbelegung Stecker P1 (nur belegte Anschlüsse)

Pin P2	Netz	E/A	Bemerkung
2	GND	–	Masse
6	U_{SO}	A	Analoges Ausgangssignal
29	GND	–	Masse

Tabelle D.15: IEPE-Interface – Pinbelegung Stecker P2 (nur belegte Anschlüsse)

Pin P3	Netz	E/A	Bemerkung
1	U_{SO}	A	Analoges Ausgangssignal
2	GND	–	Masse

Tabelle D.16: IEPE-Interface – Pinbelegung Stiftleiste P3

Pin P4	Netz	E/A	Bemerkung
1	S_G	E	Steuersignal des Spannungsteilers (Jumper über Pin 1 und 2 $\rightarrow S_G = 0 \rightarrow A_G = 1$)
2	GND	–	Masse
3	S_D	E	Steuersignal des Verstärkers (Jumper über Pin 3 und 4 $\rightarrow S_D = 0 \rightarrow A_D = 1$)
4	GND	–	Masse

Tabelle D.17: IEPE-Interface – Pinbelegung Jumper P4

Element	Bemerkung
J1	Sensoranschlussbuchse (U_{SI})
R6	Potentiometer zum Einstellen des Sensor-Versorgungsstromes

Tabelle D.18: IEPE-Interface – Weitere Bedienelemente und Buchsen

D.5 USB-Interface

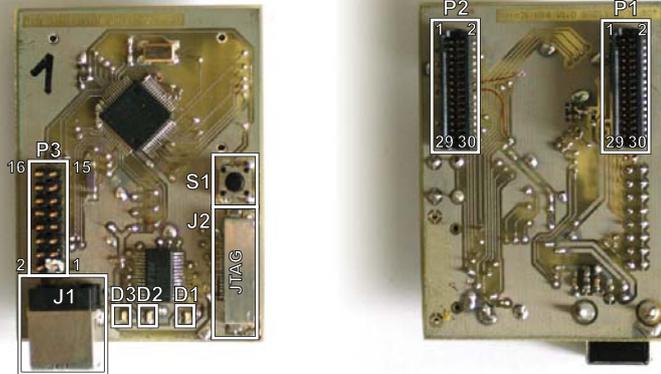


Abbildung D.5: Bedien- und Anzeigeelemente sowie Stecker und Buchsen des USB-Interfaces

MSP430F2618	Bauteil	Signal/Bemerkung
XIN, XOUT	X1	32.768 kHz Uhrenquarz (MSP430-RTC)
P3.7 UCA1RXD	FT232R	FT_TXD
P3.6 UCA1TXD	FT232R	FT_RXD
P1.5 GPIO	FT232R	FT_RTSn
P1.6 GPIO	FT232R	FT_CTSn
P1.7 GPIO	FT232R	FT_RIn
P1.0 GPIO	FT232R	FT_CBUS0
P1.1 GPIO	FT232R	FT_CBUS1
P1.2 GPIO	FT232R	FT_CBUS2
P1.3 GPIO	FT232R	FT_CBUS3
P1.4 GPIO	FT232R	FT_CBUS4

Tabelle D.19: USB-Interface – Interne Verbindungen des MSP430

Pin P1	MSP430	E/A	Bemerkung
1	–	–	GND
2	–	–	GND
30	–	–	GND

Tabelle D.20: USB-Interface – Pinbelegung Stecker P1 (nur belegte Anschlüsse)

Pin P2	MSP430	E/A	Bemerkung
2	–	–	GND
3	P3.4	E/A	GPIO/UCA0TXD/UCA0MOSI (U_TX)
6	P3.5	E/A	GPIO/UCA0RXD/UCA0MISO (U_RX)
7	P3.2	E/A	GPIO/UCB0MISO/UCB0SCL (k.F.)
8	P3.3	E/A	GPIO/UCB0CLK/UCA0STE (k.F.)
10	P3.1	E/A	GPIO/UCB0MOSI/UCB0SDA (k.F.)
11	P3.0	E/A	GPIO/UCB0STE/UCA0CLK (k.F.)
13	P2.4	E/A	GPIO/CA1/TA2 (k.F.)
14	P5.4	E/A	GPIO/MCLK (k.F.)
15	P2.2	E/A	GPIO/CAOUT/TA0/CA4 (k.F.)
16	P2.5	E/A	GPIO/ROSC/CA5 (k.F.)
25	RSTn	E	Reset des MSP430F2618
29	–	–	GND

Tabelle D.21: USB-Interface – Pinbelegung Stecker P2 (nur belegte Anschlüsse)

Pin P3	MSP430	E/A	Bemerkung
1	P6.7	E/A	GPIO/ADC7
2	–	–	GND
3	P6.6	E/A	GPIO/ADC6
4	–	–	GND
5	P6.5	E/A	GPIO/ADC5
6	–	–	GND
7	P6.4	E/A	GPIO/ADC4
8	–	–	GND
9	P6.3	E/A	GPIO/ADC3
10	–	–	GND
11	P6.2	E/A	GPIO/ADC2
12	–	–	GND
13	P6.1	E/A	GPIO/ADC1
14	–	–	GND
15	P6.0	E/A	GPIO/ADC0
16	–	–	GND

Tabelle D.22: USB-Interface – Pinbelegung Stiftleiste P3

Element	Bemerkung
D1	Anzeige ob Betriebsspannung vorhanden
D2	TX-Aktivität am USB (FT_UCBUS0)
D3	RX-Aktivität am USB (FT_UCBUS1)
S1	Reset des Moduls
J1	USB-Buchse
J2	JTAG-Anschluss

Tabelle D.23: USB-Interface – Bedien- und Anzeigeelemente sowie weitere Stecker bzw. Buchsen

D.6 Funktionsgenerator

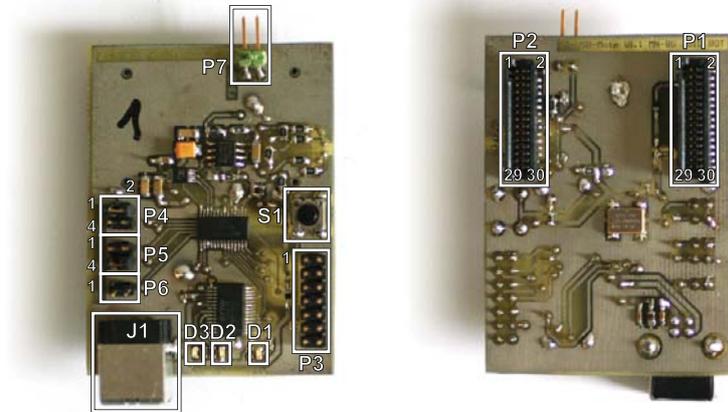


Abbildung D.6: Bedien- und Anzeigeelemente sowie Stecker und Buchsen des Funktionsgenerators

MSP430F2132	Bauteil	Signal/Bemerkung
P3.3	UCB0CLK	AD9833 AD_SCLK
P3.1	UCB0MOSI	AD9833 AD_SDATA
P3.0	GPIO	AD9833 AD_FSYNC
P3.5	UCA0RXD	FT232R FT_TXD
P3.4	UCA0TXD	FT232R FT_RXD
P1.1	GPIO	FT232R FT_RTSn
P1.2	GPIO	FT232R FT_CTSn
P1.3	GPIO	FT232R FT_RIn

Tabelle D.24: Funktionsgenerator – Interne Verbindungen des MSP430F2132

Pin P1	MSP430	E/A	Bemerkung
1	–	–	GND
2	–	–	GND
5	+3.3 V	A	+3.3 V-Versorgungsspannung des Moduls
6	+3.3 V	A	+3.3 V-Versorgungsspannung des Moduls
30	–	–	GND

Tabelle D.25: Funktionsgenerator – Pinbelegung Stecker P1 (nur belegte Anschlüsse)

Pin P2	E/A	Signal
2	–	GND
18	–	GND
19	A	U _{FO}
20	A	U _{FO}
21	A	U _{FK}
23	–	GND
25	A	U _{FO}
28	A	U _{FK}
29	–	GND
30	–	GND

Tabelle D.26: Funktionsgenerator – Pinbelegung Stecker P2 (nur belegte Anschlüsse)

Pin	MSP430	E/A	Bemerkung
P4.1	–	–	GND
P4.2	P2.0	E	F_FORM0 (Jumper über P4.1 und P4.2 → F_FORM0 = 0)
P4.3	–	–	GND
P4.4	P2.1	E	F_FORM1 (Jumper über P4.3 und P4.4 → F_FORM1 = 0)
P5.1	–	–	GND
P5.2	P2.6	E	F_FREQ0 (Jumper über P5.1 und P5.2 → F_FREQ0 = 0)
P5.3	–	–	GND
P5.4	P2.7	E	F_FREQ0 (Jumper über P5.3 und P5.4 → F_FREQ1 = 0)
P6.1	–	–	GND
P6.2	P2.5	E	F_MODE (Jumper über P6.1 und P6.2 → F_MODE = 0)

Tabelle D.27: Funktionsgenerator – Pinbelegung Stiftheisten P4, P5, P6

Signal	Bit0	Bit1	Bemerkung
F_FORM	1	1	Signalform Sinus
	0	1	Signalform Dreieck
F_FREQ	1	1	Signalfrequenz 40 Hz
	0	1	Signalfrequenz 75 Hz
	1	0	Signalfrequenz 90 Hz
F_MODE	1	–	Konfiguration des AD9833 entsprechend F_FORM und F_FREQ
	0	–	Konfiguration des AD9833 über USB (nicht implementiert)

Tabelle D.28: Funktionsgenerator – Festlegung von Signalform und Frequenz mittels P4, P5 und P6

Element	Bemerkung
D1	Anzeige ob Betriebsspannung vorhanden
D2	TX-Aktivität am USB (FT_UCBUS0)
D3	RX-Aktivität am USB (FT_UCBUS1)
S1	Reset des Moduls
J1	USB-Buchse
P3	JTAG-Anschluss

Tabelle D.29: Funktionsgenerator – Bedien- und Anzeigeelemente sowie weitere Stecker bzw. Buchsen

D.7 Verbindungstabellen

Signal	Datenspeicher				Basismodul			Bemerkung
	E/A	MSP430F5438		S/B	S/B	MSP430F2618		
D_TX	A	P5.6	UCA1TXD	P2.3	J2.3	P3.5	UCA0RXD	UART
D_RX	E	P5.7	UCA1RXD	P2.6	J2.6	P3.4	UCA0TXD	UART
D_RSTn	E	RSTn	–	P2.15	J2.15	P2.2	GPIO	Modulreset
–	–	P3.6	–	P2.8	J2.8	P3.3	–	k.F.
–	–	P5.5	–	P2.14	J2.14	P2.5	–	k.F.
–	–	P1.0	–	P2.16	J2.16	P2.3	–	k.F.

Tabelle D.30: Signalverbindungen zwischen Datenspeicher und Basismodul

Allg. Signalaufbereitung		Datenspeicher				Bemerkung
Signal	E/A	S/B	S/B	MSP430F5438		
C_G0	E	P1.15	J1.15	P1.4	GPIO	Verstärkung (LSB)
C_G1	E	P1.17	J1.17	P1.6	GPIO	Verstärkung
C_G2	E	P1.19	J1.19	P1.7	GPIO	Verstärkung (MSB)
C_RDY	A	P1.21	J1.21	P2.0	GPIO	Ready-Signal
C_SPI_SDI	E	P1.22	J1.22	P10.4	UCA3MOSI	Serieller Dateneingang
C_CLK	E	P1.23	J1.23	P1.3	TA0.2	Takt SC-Filter
C_SPI_CLK	E	P1.24	J1.24	P10.0	UCA3CLK	SPI-Takt
C_SPI_SDO	E	P1.25	J1.25	P10.5	UCA3MISO	Serieller Datenausgang
C_SPI_CS _n	E	P1.26	J1.26	P9.7	GPIO	Chip-Select-Signal
U _{CO}	A	P2.19	J2.19	P6.3	ADC3	Analoges Ausgangssignal

Tabelle D.31: Signalverbindungen zwischen allgemeiner Signalaufbereitung und dem Datenspeicher

IEPE-Interface		Allg. Signalaufbereitung		Bemerkung
Signal	E/A	S/B	Stecker/Buchse	
U _{SO}	A	P2.6	J2.6	Analoger Ausgang IEPE-Interface

Tabelle D.32: Signalverbindungen zwischen IEPE-Interface und allgemeiner Signalaufbereitung

USB-Interface				Basismodul				Bemerkung
Signal	E/A	MSP430F2618		S/B	S/B	MSP430F2618		
U_TX	A	P3.4	UCA0TXD	P2.3	J2.3	P3.5	UCA0RXD	UART
U_RX	E	P3.5	UCA0RXD	P2.6	J2.6	P3.4	UCA0TXD	UART
U_RSTn	E	RSTn	–	P2.25	J2.25	P6.3	GPIO	Modulreset
–	–	P3.2	–	P2.7	J2.7	P3.2	–	k.F.
–	–	P3.3	–	P2.8	J2.8	P3.3	–	k.F.
–	–	P3.1	–	P2.10	J2.10	P3.1	–	k.F.
–	–	P3.0	–	P2.11	J2.11	P2.6	–	k.F.
–	–	P2.4	–	P2.13	J2.13	P2.4	–	k.F.
–	–	P5.4	–	P2.14	J2.14	P2.5	–	k.F.
–	–	P2.2	–	P2.15	J2.15	P2.2	–	k.F.
–	–	P2.5	–	P2.16	J2.16	P2.3	–	k.F.

Tabelle D.33: Signalverbindungen zwischen Datenspeicher und Basismodul

Funktionsgenerator			Datenspeicher				Bemerkung
Signal	E/A	S/B	S/B	MSP430F5438			
GND	A	P2.18	J2.18	P7.6	ADC14		Testsignal
U _{FO}	A	P2.19	J2.19	P6.3	ADC3		Testsignal
U _{FO}	A	P2.20	J2.20	P7.5	ADC13		Testsignal
U _{FK}	A	P2.21	J2.21	P6.4	ADC4		Testsignal
GND	A	P2.23	J2.23	P6.5	ADC5		Testsignal
U _{FO}	A	P2.25	J2.25	P6.6	ADC6		Testsignal
U _{FK}	A	P2.28	J2.28	P7.4	ADC12		Testsignal
GND	A	P2.14	J2.30	P6.7	ADC7		Testsignal

Tabelle D.34: Signalverbindungen zwischen Funktionsgenerator und Datenspeicher

E Pinbelegung der Steuerlogik-Prototypen

Die hier verwendeten Signalbezeichnungen korrespondieren mit den Signalbezeichnungen der peripheren Logik bzw. der zentralen CPLD-Logik, wie sie in den Abschnitten 5.1.1 und 5.1.2 beschrieben sind.

Folgende Abkürzungen werden in weiterer Folge verwendet:

E/A Ein- oder Ausgangssignal des Moduls (A → Ausgang, E → Eingang).

S/B Stecker- bzw. Buchsenpin.

SF Spezialfunktion eines Pins des CPLDs oder Mikrocontrollers, die im konkreten Fall genutzt wird.

k.F. (derzeit) keine Funktion.

E.1 CPLD-Platine

Die Energieversorgung des Moduls kann einerseits über den USB-Anschluss J4 oder die Stiftleiste P13 (Pin 1: +3.3V; Pin 2: GND) erfolgen. Für die Versorgung mittels USB sind auf P12 die Pins 2 und 3 mittels Jumper zu verbinden, bei einer Versorgung über P13 die Pins 1 und 2.

Im Folgenden sind die Pinbelegungen der übrigen Stiftleisten bzw. die Signalverbindungen zwischen CPLD und Mikrocontroller in Tabellenform dokumentiert.

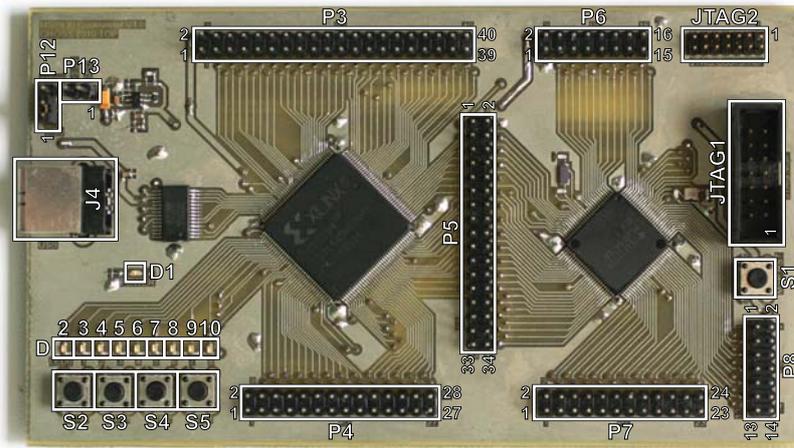


Abbildung E.1: Bedienelemente, Stecker und Buchsen der CPLD-Platine

CPLD Pin	E/A	Bauteil	Bemerkung
P110	A	D2	LED
P111	A	D3	LED
P112	A	D4	LED
P113	A	D5	LED
P114	A	D6	LED
P115	A	D7	LED
P116	A	D8	LED
P117	A	D9	LED
P118	A	D10	LED
P119	E	S2	Taster (Pullup im CPLD benötigt)
P120	E	S3	Taster (Pullup im CPLD benötigt)
P121	E	S4	Taster (Pullup im CPLD benötigt)
P124	E	S5	Taster (Pullup im CPLD benötigt)
P107	E	FT232R	FT_TXD
P105	A	FT232R	FT_RXD
P106	E	FT232R	FT_RTSn
P103	A	FT232R	FT_CTSn
P104	A	FT232R	FT_RIn
P98	E	FT232R	FT_CBUS0 (TXLEDn)
P97	E	FT232R	FT_CBUS1 (RXLEDn)
P101	E	FT232R	FT_CBUS2 (TXDEN)
P100	E	FT232R	FT_CBUS3 (PWRENn)
P102	E	FT232R	FT_CBUS4 (SLEEPn)

Tabelle E.1: CPLD-Platine – Verbindungen des CPLDs mit Bedienelementen und FTDI232R

CPLD			MSP430			Stecker	Verwendung
Pin	SF	E/A	Pin	SF	E/A		
P38	GCK2	E	P2.0	MCLK	A	P5.11	CR_CLK
P143	GSR	E	P4.6	–	A	P5.33	CR_RESETh
P39	–	E	P2.1	–	A	P5.10	CR_ADDR(0)
P35	–	E	P2.2	–	A	P5.12	CR_ADDR(1)
P34	–	E	P2.3	–	A	P5.13	CR_ADDR(2)
P33	–	E	P2.4	–	A	P5.14	CR_ADDR(3)
P46	–	E/A	P1.0	–	E/A	P5.3	CR_DATA(0)
P45	–	E/A	P1.1	–	E/A	P5.4	CR_DATA(1)
P44	–	E/A	P1.2	–	E/A	P5.5	CR_DATA(2)
P43	–	E/A	P1.3	–	E/A	P5.6	CR_DATA(3)
P42	–	E/A	P1.4	–	E/A	P5.7	CR_DATA(4)
P41	–	E/A	P1.5	–	E/A	P5.8	CR_DATA(5)
P32	–	E/A	P1.6	–	E/A	P5.15	CR_DATA(6)
P40	–	E/A	P1.7	–	E/A	P5.9	CR_DATA(7)
P30	–	E	P2.6	–	A	P5.17	CR_CS _n
P28	–	E	P2.7	–	A	P5.18	CR_OE
P6	–	E	P4.1	–	A	P5.28	CR_WRITE
P23	–	E	P3.3	UCB0CLK	A	P5.22	CR_SPI_CLK
P20	–	E	P3.6	–	A	P5.25	CR_SPI_CS _n
P25	–	E	P3.1	UCB0MOSI	A	P5.20	CR_SPI_MOSI
P24	–	A	P3.2	UCB0MISO	E	P5.21	CR_SPI_MISO
P5	–	E	P4.2	TB0.2	A	P5.29	CR_C_CLK(0)
P4	–	E	P4.3	TB0.3	A	P5.30	CR_C_CLK(1)
P3	–	E	P4.4	TB0.4	A	P5.31	CR_C_CLK(2)
P2	–	E	P4.5	TB0.5	A	P5.32	CR_C_CLK(3)
P31	–	E	P2.5	–	A	P5.16	CR_C_INT
P21	–	E	P3.5	UCA0RXD	A	P5.24	CR_TXD
P22	–	A	P3.4	UCA0TXD	E	P5.23	CR_RXD
P142	–	–	P4.7	TB0CLK	E	P5.34	10 MHz-TCO
P7	–	–	P4.0	–	–	P5.27	k.F.
P19	–	–	P3.7	–	–	P5.26	k.F.
P26	–	–	P3.0	–	–	P5.19	k.F.

Tabelle E.2: CPLD-Platine – Signalverbindungen zwischen CPLD und MSP430 (Pinbelegung P5)

Pin P3	CPLD	E/A	Verwendung
1	–	–	GND
2	–	–	+3.3 V
3	P96	A	C_SPI_CLK(1)
4	P95	A	C_SPI_CLK(0)
5	P94	A	C_SPI_MOSI(1)
6	P92	A	C_SPI_MOSI(0)
7	P91	E	C_SPI_MISO(1)
8	P88	E	C_SPI_MISO(0)
9	P87	A	C_SPI_CS _n (1)
10	P86	A	C_SPI_CS _n (0)
11	P85	A	C_SPI_SEL0(1)
12	P83	A	C_SPI_SEL0(0)
13	P82	A	C_SPI_SEL1(1)
14	P81	A	C_SPI_SEL1(0)
15	P80	A	C_SPI_CLK(2)
16	P79	A	CD_SPI_CS _n
17	P78	A	C_SPI_MOSI(2)
18	P77	A	CD_SPI_CLK
19	P76	E	C_SPI_MISO(2)
20	P75	A	CD_SPI_MOSI
21	P74	A	C_SPI_CS _n (2)
22	P71	E	CD_SPI_MISO
23	P70	A	C_SPI_SEL0(2)
24	P69	A	C_CLK(0)
25	P68	A	C_SPI_SEL1(2)
26	P66	E	C_INT(3)
27	P64	A	C_SPI_CLK(3)
28	P61	E	C_INT(2)
29	P60	A	C_SPI_MOSI(3)
30	P59	E	C_INT(1)
31	P58	E	C_SPI_MISO(3)
32	P57	E	C_INT(0)
33	P56	A	C_SPI_CS _n (3)
34	P54	–	k.F.
35	P53	A	C_SPI_SEL0(3)
36	P52	A	C_CLK(1)
37	P51	A	C_SPI_SEL1(3)
38	P50	A	C_CLK(2)
39	P49	–	k.F.
40	P48	A	C_CLK(3)

Tabelle E.3: CPLD-Platine – Pinbelegung Stiftleiste P3

Pin P4	CPLD	E/A	Verwendung
1	–	–	GND
2	–	–	+3.3 V
3	P126	A	S_RS232_RXD(0)
4	P125	E	S_RS232_TXD(0)
5	P129	A	S_RS232_RXD(1)
6	P128	E	S_RS232_TXD(1)
7	P131	A	S_RS232_RXD(2)
8	P130	E	S_RS232_TXD(2)
9	P133	A	S_RS232_RXD(3)
10	P132	E	S_RS232_TXD(3)
11	P135	–	k.F.
12	P134	–	k.F.
13	P137	–	k.F.
14	P136	–	k.F.
15	P139	–	k.F.
16	P138	–	k.F.
17	P9	–	k.F.
18	P140	–	k.F.
19	P11	–	k.F.
20	P10	–	k.F.
21	P13	–	k.F.
22	P12	–	k.F.
23	P15	–	k.F.
24	P14	–	k.F.
25	P17	–	k.F.
26	P16	–	k.F.
27	–	–	–
28	P18	–	k.F.

Tabelle E.4: CPLD-Platine – Pinbelegung Stiftleiste P4

Pin P6	E/A	MSP430F5438A		Verwendung
		Pin	SF	
1	–	–	–	GND
2	–	–	–	+3.3 V
3	–	P5.1	–	k.F.
4	–	P5.0	–	k.F.
5	–	P7.7	–	k.F.
6	–	P7.6	–	k.F.
7	–	P7.5	–	k.F.
8	–	P7.4	–	k.F.
9	–	P6.7	–	k.F.
10	–	P6.6	–	k.F.
11	–	P6.5	–	k.F.
12	–	P6.4	–	k.F.
13	–	P6.3	–	k.F.
14	–	P6.2	–	k.F.
15	–	P6.1	–	k.F.
16	E	P6.0	ADC0	U_{CO}

Tabelle E.5: MSP430-CPLD-Platine – Pinbelegung Stiftleiste P6

Pin P7	E/A	MSP430F5438A		Verwendung
		Pin	Spezialfunktion	
1	–	–	–	GND
2	–	–	–	+3.3 V
3	–	P5.4	–	k.F.
4	–	P5.5	–	k.F.
5	–	P5.6	–	k.F.
6	–	P5.7	–	k.F.
7	–	P7.2	–	k.F.
8	–	P7.3	–	k.F.
9	–	P8.0	–	k.F.
10	–	P8.1	–	k.F.
11	–	P8.2	–	k.F.
12	–	P8.3	–	k.F.
13	–	P8.4	–	k.F.
14	–	P8.5	–	k.F.
15	–	P8.6	–	k.F.
16	–	P8.7	–	k.F.
17	–	P9.0	–	
18	A	P9.1	UCB2MOSI	SD-Karte D_SI
19	E	P9.2	UCB2MISO	SD-Karte D_SO
20	A	P9.3	UCB2CLK	SD-Karte S_SCLK
21	–	P9.4	–	k.F.
22	–	P9.5	–	k.F.
23	A	P9.6	–	SD-Karte D_CS _n
24	–	P9.7	–	k.F.

Tabelle E.6: MSP430-CPLD-Platine – Pinbelegung Stiftleiste P7

Pin P8	E/A	MSP430F5438A		Verwendung
		Pin	Spezialfunktion	
1	–	–	–	GND
2	–	–	–	+3.3 V
3	–	P11.1	–	k.F.
4	–	P11.2	–	k.F.
5	–	P10.7	–	k.F.
6	–	P11.0	–	k.F.
7	–	P10.5	UCA3RXD	Kommunikation Basismodul – D_RX
8	–	P10.6	–	k.F.
9	–	P10.3	–	k.F.
10	A	P10.4	UCA3TXD	Kommunikation Basismodul – D_TX
11	–	P10.1	–	k.F.
12	–	P10.2	–	k.F.
13	–	–	–	k.F.
14	–	P10.0	–	k.F.

Tabelle E.7: MSP430-CPLD-Platine – Pinbelegung Stiftleiste P8

Element	Bemerkung
D1	Anzeige ob Betriebsspannung vorhanden
S1	Reset des MSP430
JTAG1	JTAG-Anschluss des MSP430
JTAG2	JTAG-Anschluss des CPLDs

Tabelle E.8: CPLD-Platine – Bedien- und Anzeigeelemente sowie weitere Stecker bzw. Buchsen

E.2 CPLD-Erweiterungsplatine

Auf Grund der hohen Anzahl an Steckverbindern wird im Folgenden lediglich die Belegung der von oben zugänglichen Stiftleisten (siehe Abbildung E.2) beschrieben. Da es sich bei der Platine um eine Erweiterungsplatine zur CPLD-Platine handelt, und die Buchsenleisten an der Unterseite im aufgesteckten Zustand nicht zugänglich sind, kann an dieser Stelle auf eine Dokumentation der einzelnen Funktionen dieser Pins verzichtet werden.

Auf der Erweiterungsplatine ist die periphere Logik eines Kanals (Kanal 0) entsprechend der Dokumentation in Abschnitt 5.1.1 implementiert. Weiters sind zwei Flash-Speicher vorhanden, die durch eine entsprechende Konfiguration des Konfigurationsspeichers im CPLD (siehe Tabellen 5.3 und 5.4) mittels CD_SPI bzw. C_SPI2 (siehe Abbildung 5.2) angesprochen werden. Durch setzen des Jumpers P11 kann der Inhalt dieser Speicherbau-

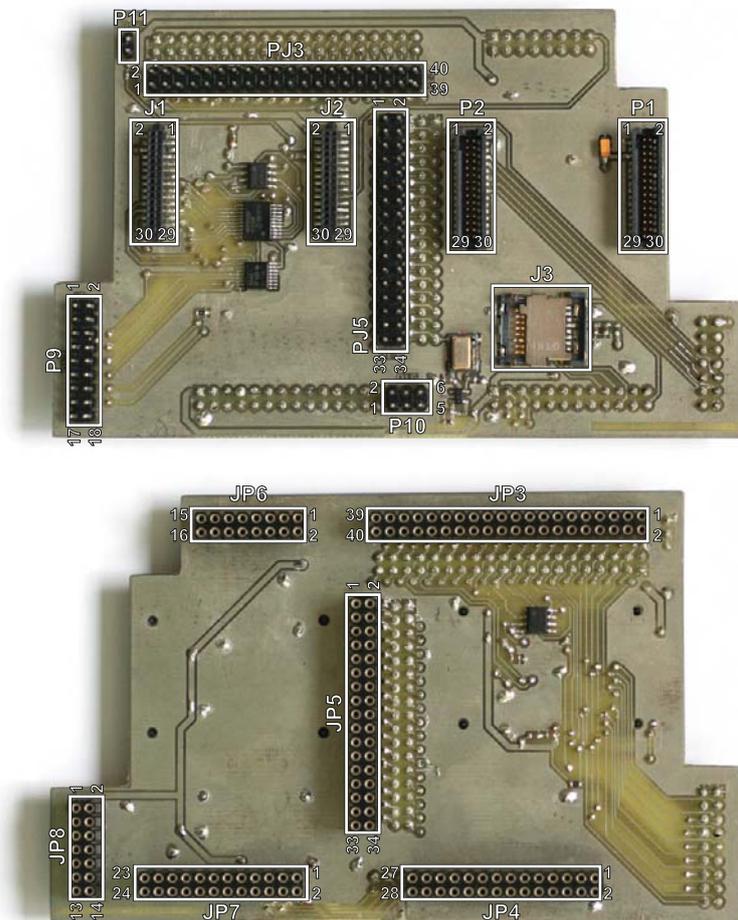


Abbildung E.2: CPLD-Datenspeicher-Platine

steine vor Veränderungen geschützt werden.

Die Funktionalität des Datenspeichers (siehe Abschnitt 4.3) wird durch einen 10 MHz-Oszillator, der mit dem P4.7 des MSP430 verbunden ist, und den SD-Kartensockel J3 (siehe Tabelle E.13) ermöglicht.

Die Versorgung der gesamten Elektronik kann entweder über die CPLD-Platine (USB-Bus J4 bzw. Stiftleiste P13) oder über ein aufgestecktes Basismodul (Batterien) erfolgen. Bei einer Versorgung über ein Basismodul darf auf der CPLD-Platine kein Jumper auf P12 gesetzt sein.

Die Belegung der Stiftleisten PJ3 und PJ5 ist identisch zu P3 und P5 in Abschnitt E.1. Im folgenden sind die Pinbelegungen aller weiteren Stiftleisten auf der Oberseite der Erweiterungsplatine in Tabellenform dokumentiert.

Pin P1	MSP430F5438	E/A	Bemerkung
1	–	–	GND
2	–	–	GND
5	–	E/A	+3.3 V-Versorgungsspannung
6	–	E/A	+3.3 V-Versorgungsspannung
30	–	–	GND

Tabelle E.9: CPLD-Erweiterungsplatine – Pinbelegung Stecker P1 (Verbindung zum Basismodul – nur belegte Anschlüsse)

Pin P2	MSP430	E/A	Verwendung
2	–	–	GND
3	P10.5	E	D_TX (RS232-Kommunikation)
6	P10.4	A	D_RX (RS232-Kommunikation)
8	P10.3	–	k.F.
14	P10.2	–	k.F.
16	P10.1	–	k.F.
29	–	–	GND

Tabelle E.10: CPLD-Erweiterungsplatine – Pinbelegung Stecker P2 (Verbindung zum Basismodul – nur belegte Anschlüsse)

Pin J1	E/A	Verwendung
1	A	GND
2	A	GND
5	A	+3.3 V-Versorgungsspannung
6	A	+3.3 V-Versorgungsspannung
15	E/A	C_G0(0)
17	E/A	C_G1(0)
19	E/A	C_G2(0)
21	E/A	C_RDY(0)
22	E/A	C_SPI_MOSI(0)
23	E/A	C_CLK(0)
24	E/A	C_SPI_CLK(0)
25	E/A	C_SPI_MISO(0)
26	E/A	C_SPI_CSn0(0)
30	A	GND

Tabelle E.11: CPLD-Erweiterungsplatine – Pinbelegung Buchse J1 (Verbindung zur allgemeinen Signalaufbereitung – nur belegte Anschlüsse)

Pin J2	MSP430	E/A	Verwendung
2	–	A	GND
19	P6.0	E	U_{CO} (Ausgangssignal der allg. Signalaufbereitung)
29	–	A	GND

Tabelle E.12: CPLD-Erweiterungsplatine – Pinbelegung Buchse J2 (Verbindung zur allgemeinen Signalaufbereitung – nur belegte Anschlüsse)

Pin J3	MSP430F5438A	E/A	Verwendung
2	P9.6	A	D_CSn
3	P9.1	A	D_SI
4	–	A	+3.3 V
5	P9.3	A	D_CLK
6	–	A	GND
7	P9.2	E	D_SO

Tabelle E.13: CPLD-Erweiterungsplatine – Pinbelegung Sockel SD-Karte J3 (nur belegte Pins)

Pin P9	CPLD	E/A	Verwendung
1	–	GND	
2	–	GND	
3	A	C_G0(0)	
4	–	k.F.	
5	A	C_G1(0)	
6	A	C_SPI_CS _n 0(0)	
7	A	C_G2(0)	
8	A	C_SPI_CS _n 1(0)	
9	E	C_RDY(0)	
10	A	C_SPI_CS _n 2(0)	
11	E/A	S_IO0(0)	
12	A	C_SPI_CS _n 3(0)	
13	E/A	S_IO1(0)	
14	A	C_SPI_CLK(0)	
15	E/A	S_IO2(0)	
16	A	C_SPI_MOSI(0)	
17	E/A	S_IO3(0)	
18	E	C_SPI_MISO(0)	

Tabelle E.14: CPLD-Erweiterungsplatine – Pinbelegung Stiftleiste P9 (Signale der peripheren Logik für den Sensor kanal 0)

Pin P10	MSP430F5438A	E/A	Verwendung
1	–	–	+3.3 V
2	–	–	GND
3	P5.6	–	k.F.
4	P5.7	–	k.F.
5	P5.4	–	k.F.
6	P5.5	–	k.F.

Tabelle E.15: CPLD-Erweiterungsplatine – Pinbelegung Stiftleiste P10

F Kalibrierdatenblätter der Sensoren

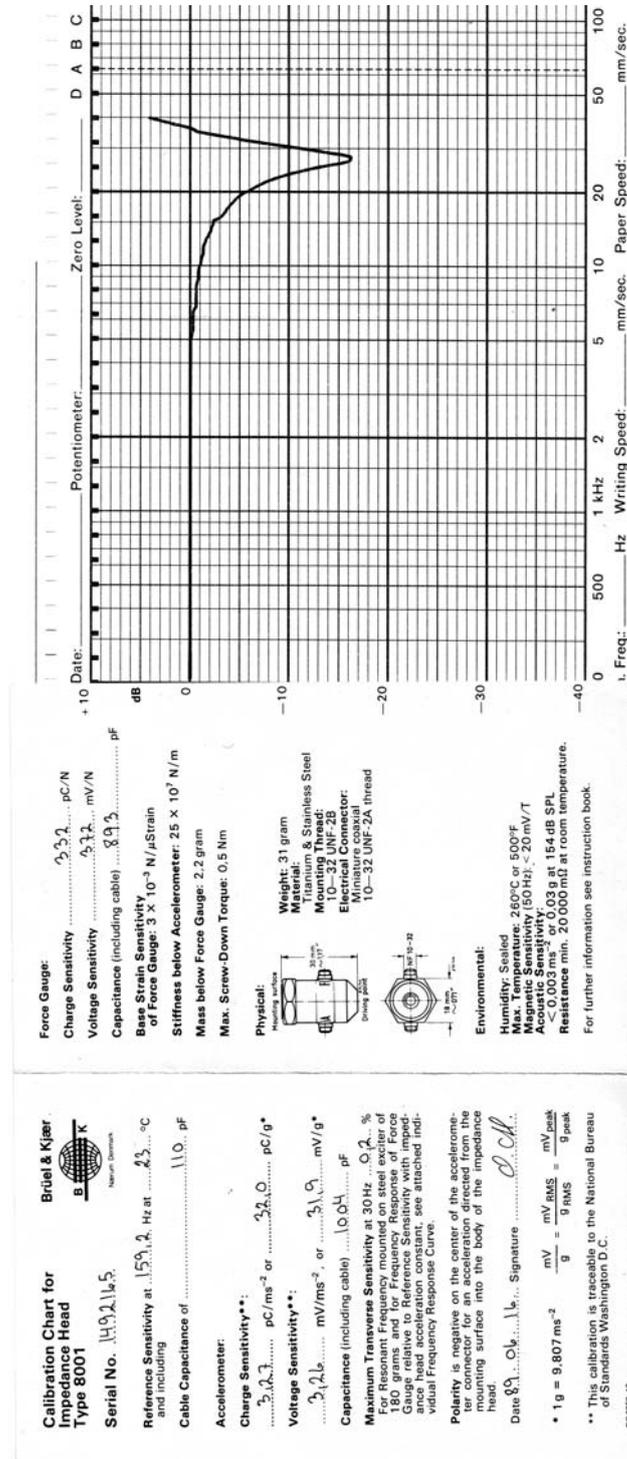


Abbildung F.1: Kalibrierdatenblatt des Impedanzmesskopfs Brüel & Kjær Typ 8001

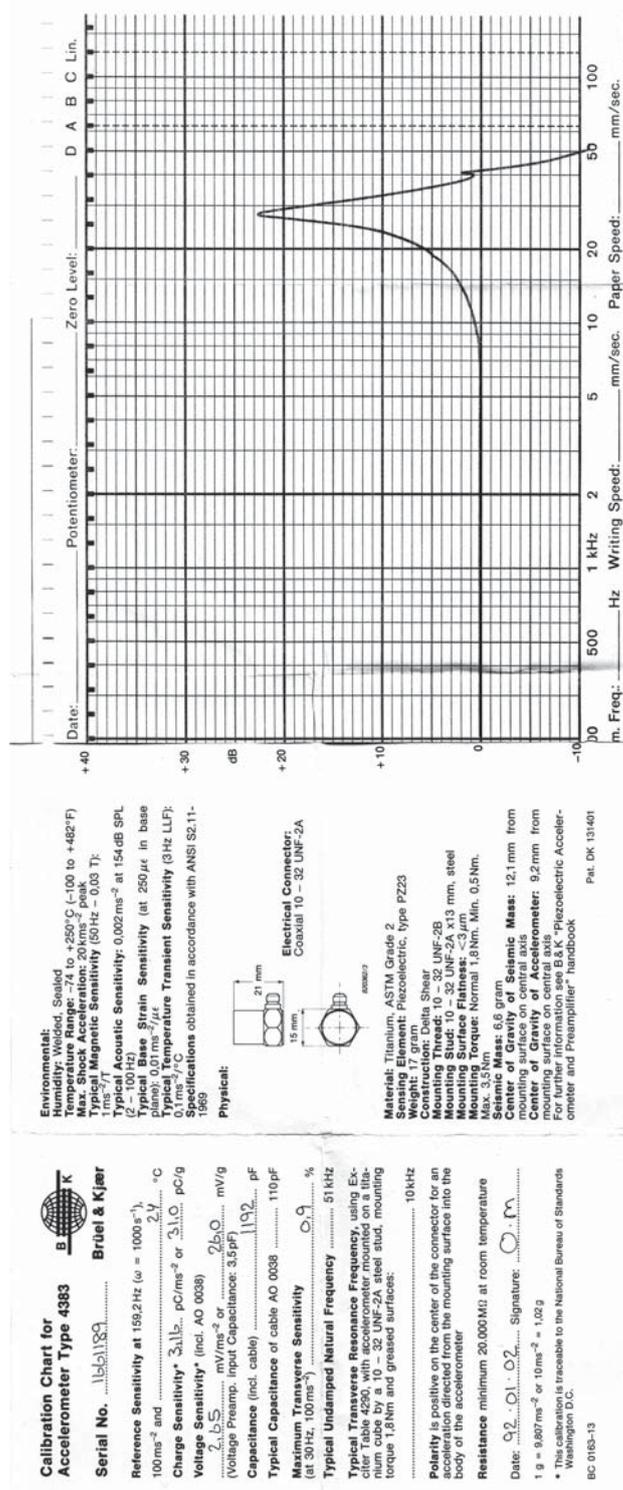


Abbildung F.2: Kalibrierdatenblatt des Piezo-Beschleunigungssensors Brüel & Kjær Typ 4383

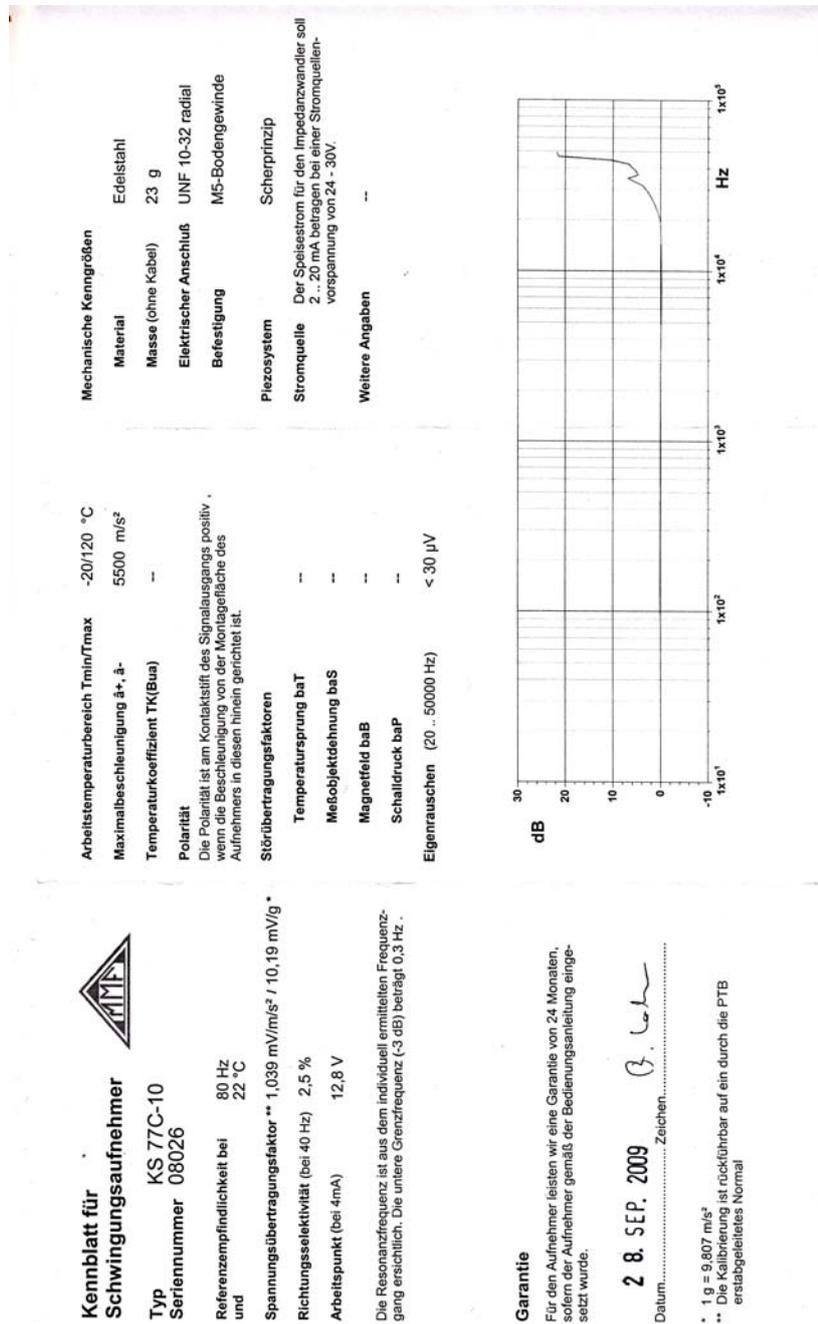


Abbildung F.3: Kalibrierdatenblatt des IEPE-Beschleunigungssensors MMF KS77C-10

Abkürzungsverzeichnis

A/D-Wandlung	Analog-Digital-Wandlung
ADC	Analog-to-Digital-Converter
API	Application Programming Interface
BM	Basismodul
CAD	Computer Aided Design
CF	CompactFlash
CPLD	Complex Programmable Logic Device
CRC	Cyclic Redundancy Check
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
CSV	Comma-Separated Values
CTS	Clear To Send
DC	Direct Current
DDS	Direct Digital Synthesis
FAT	File Allocation Table
FDD	Frequency Domain Division
FFD	Full Function Device
FFT	Fast Fourier Transform
GPIO	General Purpose Input/Output
GUI	Graphical User Interface
HF	Hochfrequenz
I2C	Inter Integrated Circuit
IC	Integrated Circuit
IEEE	Institute of Electrical and Electronics Engineers
IEPE	Integrated Electronics Piezo Electric
IP	Internet Protocol
ISM	Industrial, Scientific and Medical
ISO	International Organization for Standardization
ISR	Interrupt Service Routine

ITU	International Telecommunication Union
JTAG	Joint Test Action Group
LED	Light-Emitting Diode
LQI	Link Quality Indicator
MAC	Media Access Control
MEMS	Micro-Electro-Mechanical Systems
NiMH	Nickel-Metallhydrid
OCXO	Oven Controlled Crystal Oscillator
OQPSK	Offset Quadrature Phase-Shift Keying
OSAL	Operating System Abstraction Layer
OSI	Open Systems Interconnection
OUI	Organizational Unique Identifiers
PAN	Personal Area Networks
PC	Personal Computer
PCB	Printed Circuit Board
PER	Packet Error Ratio
PHY	Physical (Layer)
ppm	Parts Per Million
RAM	Random Access Memory
RFD	Reduced Function Device
RFID	Radio Frequency Identification
RSSI	Received Signal Strength Indicator
RTL	Register Transfer Level
RTS	Request To Send
RX	Receive
SC	Switched Capacitor
SD	Secure Digital (Card)
SDHC	Secure Digital High Capacity (Card)
SFD	Start of Frame Delimiter
SPI	Serial Peripheral Interface
TCP	Transmission Control Protocol
TCXO	Temperature Compensated Crystal Oscillator
TDD	Time Domain Division
TI	Texas Instruments
TX	Transmit

UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
VHDL	Very high speed integrated circuit Hardware Description Language
WPAN	Wireless Personal Area Network
XO	Crystal Oscillator

Abbildungsverzeichnis

2.1	Aufbau eines Oberflächenwellensensors	4
3.1	Prinzipieller Aufbau eines Systemknotens	16
3.2	Mesh-Topologie	16
3.3	Lokale Verfügbarkeit der ISM-Bänder	18
3.4	OSI-Modell mit IEEE-Erweiterung [18]	20
3.5	Bluetooth Pico- und Scatternets	22
3.6	IEEE 802.15.4 Funkkanäle im 2.4 GHz-Band	23
3.7	Stern- und Peer-to-Peer-Topologie	24
3.8	IEEE 802.15.4 Schichten	24
3.9	Aufbau des IEEE 802.15.4 MAC-Datenrahmens	25
3.10	ZigBee Protokollarchitektur	28
3.11	Beispiel eines ZigBee-Netzwerkes	30
3.12	Ragone Diagramm	31
3.13	Entladespannung – Eneloop HR-3UTG	34
3.14	Ladespannung – Eneloop HR-3UTG	34
3.15	Aufbau einer Flash-Speicherkarte	35
3.16	Passive und aktive Hoch- und Tiefpässe 1. Ordnung	37
3.17	Hochpass mittels negativem Feedback	38
3.18	Invertierender Integrator	39
3.19	Schematischer Aufbau eines Piezo-Beschleunigungssensors	40
3.20	IEPE-Messkette	41
3.21	Knotentypen des Messsystems	42
3.22	Schematischer Aufbau eines Messknotens	45
4.1	Module des Laborsystems	47
4.2	Assemblierte Knoten des Laborsystems	48
4.3	Basismodul des Laborsystems	49

4.4	Kommunikationsfluss des Laborsystems	51
4.5	Messkette des Laborsystems	51
4.6	Blockschaltbild des Basismoduls	54
4.7	Reference Design CC2520EM	54
4.8	Anpassungsnetzwerk und digitales Interface des CC2520	54
4.9	Microstrip-Geometrie	55
4.10	Reference Design CC2520EM und Prototypen des HF-Teils	58
4.11	CC2520 Development Kit	58
4.12	Schaltbild des LTC3526L	61
4.13	Energieversorgung des Laborsystems	62
4.14	Beispielapplikation SerialApp von TI	62
4.15	RS232-Paketaufbau	64
4.16	Kommunikation innerhalb des Laborsystems	65
4.17	Beispiel – Nachricht vom lokalen Zusatzmodul an ein Zusatzmodul im Netz- werk	66
4.18	Beispiel – Nachricht vom lokalen Zusatzmodul an ein Basismodul im Netz- werk	66
4.19	Beispiel – Nachricht vom lokalen Zusatzmodul an das lokale Basismodul . .	67
4.20	Beispiel – Nachricht eines Basismoduls im Netzwerk an das lokale Zusatz- modul	67
4.21	Beispiel – Nachricht des lokalen Basismoduls an das lokale Zusatzmodul . .	67
4.22	Anforderung der Netzwerkadresse mittels Broadcast über das Funknetz . .	69
4.23	Übermittlung der Netzwerkadresse des Knotens 0x0503 über das Netzwerk	69
4.24	Anforderung der Netzwerkadresse des lokalen Knotens	69
4.25	Übermittlung der lokalen Adressinformation	70
4.26	Bestimmung des LQI – Schritt 1 (Anforderung)	71
4.27	Bestimmung des LQI – Schritt 2 (Single Hop Broadcast)	71
4.28	Bestimmung des LQI – Schritt 3 (Antwort der Nachbarknoten)	72
4.29	Bestimmung des LQI – Schritt 4 (Rückmeldung der Ergebnisse)	72
4.30	RS232-Signale beim Senden und Empfangen von Paketen	73
4.31	Detailansicht – RS232-Signale beim Senden und Empfangen von Paketen .	74
4.32	Zeitintervalle beim Senden und Empfangen von Nachrichten	75
4.33	Signalverbindungen zwischen Datenspeichermodul und Basismodul sowie der allgemeinen Signalaufbereitung	78
4.34	Anbindungsmöglichkeiten von SD-Karten	78

4.35	Blockschaltbild des Datenspeichermoduls	79
4.36	Aufzeichnung von Messdaten mit dem DOSonChip-Treiber-IC	82
4.37	Aufzeichnung von Messdaten mit dem FAT32-Firmwaretreiber	83
4.38	Prototyp des Datenspeichers	84
4.39	Möglichkeiten zur Synchronisierung von Zeitbasen	86
4.40	Geschwindigkeitskorrektur der Zeitbasis	89
4.41	Strombedarf der Transcend 4 GB microSD-Karte	91
4.42	Zustandsautomat des Datenspeichermoduls	92
4.43	Verarbeitung der Abtastwerte	93
4.44	Struktur der gespeicherten Daten	95
4.45	Aufbau der Zeitmarken	95
4.46	Kodierung der Abtastwerte	95
4.47	Kommunikationsbeispiel – Minimalsystem	96
4.48	Blockschaltbild der allgemeinen Signalaufbereitung	98
4.49	Schaltbild des Verstärkers LTC6910	100
4.50	Schaltbild der Pegelanpassung	100
4.51	Schaltbild des Antialiasing-Filters	102
4.52	Schaltbild des TPS62250	104
4.53	Schaltbild des TPS63700	105
4.54	Ausgangswelligkeit des TPS62250 und TPS63700	105
4.55	Ein- und Ausgangsspannungen der allgemeinen Signalaufbereitung	106
4.56	Frequenzgänge der allgemeinen Signalaufbereitung für $f_{g,SC} = 375 \text{ Hz}$	107
4.57	Frequenzgänge der allgemeinen Signalaufbereitung für $f_{g,SC} = 37.5 \text{ Hz}$	107
4.58	Prototypen der allgemeinen Signalaufbereitung	109
4.59	Blockschaltbild des IEPE-Interfaces	110
4.60	Schaltbild des TPS61081	111
4.61	Ausgangswelligkeit des TPS61081 bei 4 mA Laststrom	111
4.62	Schaltbild der Stromquelle	112
4.63	Schaltbild des Spannungsteilers	112
4.64	Blockschaltbild des IEPE-Interfaces mit variabler Verstärkung – Ausgangs- punkt der Entwicklung	113
4.65	Schaltbild des Hochpasses und Verstärkers für die Prototypen 1 & 2	114
4.66	Schaltbild des Hochpasses und Verstärkers – Prototyp 3	115
4.67	Ausgangsspannungen der Versionen 2.0 und 3.0 des IEPE-Interfaces bei Temperaturänderung	116

4.68	Versuchsaufbau zur Bestimmung der Frequenzgänge des IEPE-Interfaces exklusive Sensor	117
4.69	Frequenzgänge des IEPE-Sensorinterfaces exklusive Sensor	118
4.70	Versuchsaufbau zur Bestimmung der Frequenzgänge des IEPE-Interfaces inklusive Sensor	119
4.71	Ausgangssignal der Referenzmesskette auf Grund von temperturbedingten mechanischen Spannungen	120
4.72	Frequenzgänge des IEPE-Sensorinterfaces inklusive Sensor	121
4.73	Prototypen des IEPE-Interfaces	122
4.74	Schaltbild des Hochpasses und Verstärkers – IEPE-Interface V3.1	123
4.75	Anbindung des Basismoduls an einen PC/Laptop	123
4.76	Foto des USB-Interfaces	124
4.77	Blockschaltbild des USB-Interfaces	124
4.78	Schaltbild des FT232R	125
4.79	Terminalprogramm Docklight	126
4.80	LabVIEW-GUIs	126
4.81	Blockschaltbild der Signalgenerierung des Funktionsgenerators	127
4.82	Foto des DDS-Moduls	128
4.83	Blockschaltbild der Komponenten des DDS-Moduls	128
4.84	Schaltbild der Signalerzeugung	128
4.85	Ausgangsspannungen U_{FO} des DDS-Moduls	129
4.86	BTA-Werksgelände 1220 Wien, Hermann Gebauer Straße 5	130
4.87	Position der Systemknoten (Messung im Stillstand)	132
4.88	Verbindungsqualität zwischen den einzelnen Knoten	132
4.89	Per Funk übertragene ADC-Kanäle der Knoten 3, 6 und 9	133
4.90	Im Stillstand aufgezeichnete Messdaten des Knotens 9	134
4.91	Position der Systemknoten (Messung mit Fahrtanteilen)	135
4.92	Bluetoothübertragung in der Nähe des Knotens 6	136
4.93	Während der Fahrt aufgezeichnete Messdaten des Knotens 9	137
5.1	Blockschaltbild der Steuerlogik	139
5.2	Schaltbild der peripheren Logik	141
5.3	Vereinfachtes Prinzipschaltbild der CPLD-Logik	144
5.4	Simulation der CPLD-Logik – Schreiben und Lesen des Konfigurationsspei- chers	146

5.5	Simulation der CPLD-Logik – Umschalten des SPI-Kanals	147
5.6	Simulation der CPLD-Logik – Umschalten der C_CLKs	148
5.7	Setup- und Hold-Zeiten	149
5.8	Funktionsprototyp der Steuerlogik bestehend aus CPLD-Platine und der zugehörigen Erweiterungsplatine	151
5.9	Funktionsprototyp der Steuerlogik inklusive der Module des Laborsystems	151
5.10	Blockdiagramm des Funktionsprototypen der Steuerlogik	152
5.11	Signalverläufe beim Schreiben und Lesen des Konfigurationsspeichers . . .	153
5.12	SPI-Kommunikation zwischen MSP430 und MCP23S08 bzw. AD5231 . . .	154
5.13	Umschaltung der Quelle der Taktsignale	155
5.14	Energieversorgung des Messknotens	156
5.15	Messung des Wirkungsgrades	159
5.16	Schaltbild des LTC3112 – Ausgangsspannung +3.3 V	160
5.17	LTC3112 Prototypen	160
5.18	Wirkungsgrad des LTC3112 bei $U_A = +3.3\text{ V}$	161
5.19	Wirkungsgrad des LTC3112 bei $U_A = +2.5\text{ V}$	161
5.20	Ausgangswelligkeit des LTC3112 bei $U_E = +5\text{ V}$ und $U_A = +3.3\text{ V}$	162
5.21	Ausgangswelligkeit des LTC3112 bei $U_E = +5\text{ V}$ und $U_A = +2.5\text{ V}$	162
5.22	Schaltbild des LT3580	163
5.23	LT3580 Prototyp	163
5.24	Wirkungsgrad des LT3580 bei $U_A = -2.5\text{ V}$	164
5.25	Ausgangswelligkeit des LT3580 bei $U_E = +5\text{ V}$ und $U_A = -2.5\text{ V}$	164
5.26	Wirkungsgrad des TPS61081 bei $U_A = 24\text{ V}$	165
5.27	Ausgangswelligkeit des TPS61081 bei $U_E = +5\text{ V}$ und $U_A = +24\text{ V}$	165
5.28	Schaltbild des Ladereglers LTC4011	166
5.29	Ladezyklus des LTC4011 von drei in Serie geschalteten NiMh-Zellen	167
5.30	Schaltbild des PowerPath™Controllers LTC4412	168
5.31	Foto der Ladeschaltung inklusive LTC4412	168
5.32	Spannungs- und Stromverläufe bei einem Ausfall der Versorgungsspannung	169
5.33	Hardwaremodifikation des Datenspeichers (Variante 1)	170
5.34	Hardwaremodifikation des Datenspeichers (Variante 2)	170
5.35	RX-/TX-Jitter-Kompensation	171
D.1	Bedienelemente, Stecker und Buchsen des Basismoduls	200
D.2	Bedienelemente, Stecker und Buchsen des Datenspeichermoduls	203

D.3	Bedienelemente, Stecker und Buchsen der allgemeinen Signalaufbereitung	205
D.4	Bedienelemente, Stecker und Buchsen des IEPE-Interfaces	207
D.5	Bedien- und Anzeigeelemente sowie Stecker und Buchsen des USB-Interfaces	209
D.6	Bedien- und Anzeigeelementes sowie Stecker und Buchsen des Funktionsgenerators	212
E.1	Bedienelemente, Stecker und Buchsen der CPLD-Platine	218
E.2	CPLD-Datenspeicher-Platine	225
F.1	Kalibrierdatenblatt des Impedanzmesskopfs Brüel & Kjær Typ 8001	230
F.2	Kalibrierdatenblatt des Piezo-Beschleunigungssensors Brüel & Kjær Typ 4383	231
F.3	Kalibrierdatenblatt des IEPE-Beschleunigungssensors MMF KS77C-10	232

Tabellenverzeichnis

3.1	Typische Eckdaten von Funkmodulen – IEEE802.11bg, Bluetooth und IEEE 802.15.4	27
4.1	Induktivität pro Millimeter für verschiedene Microstrip-Geometrien	57
4.2	Leiterbahnlänge der Induktivitäten des Anpassungsnetzwerkes	57
4.3	Paketfehlerraten des CC2520EM und der HF-Prototypen	60
4.4	Strombedarf MSP430F2618 und CC2520	62
4.5	RS232-Kommunikationsparameter des Basismoduls	63
4.6	Bedeutung der einzelnen Bytes des Kommunikationsprotokolls	65
4.7	Zeitintervalle beim Senden und Empfangen von Nachrichten	76
4.8	Empfangszeitpunkte von Nachrichten am n-ten Knoten	77
4.9	Treiberbausteine für SD- bzw. SDHC-Karten	80
4.10	Mittlere Schreibgeschwindigkeit des DOSonChip-Treibers	81
4.11	Echtzeit-Schreibrate des DOSonChip-Treibers	82
4.12	Echtzeit-Schreibrate des FAT32-Firmwaretreibers	83
4.13	Vergleich verschiedener quarzbasierter Oszillatortypen	85
4.14	Mittlere Endwerte t_E der Zeitbasen der Module 0 bis 4 nach 4 h Laufzeit	88
4.15	Korrekturwerte zur Geschwindigkeitsanpassung der Zeitbasen	89
4.16	Werte der Zeitbasen t_E der Module 0 bis 4 nach 4 h Laufzeit mit Geschwindigkeitsanpassung	90
4.17	Timing der Interrupt-Service-Routinen des Datenspeichers	94
4.18	Verstärkungen des LTC6010-1, LTC6910-2 und LTC6910-3 laut [56]	99
4.19	Grenzfrequenzen und Steuersignale für die Abtastraten 1 kHz und 100 Hz	104
4.20	Strombedarf der allgemeinen Signalaufbereitung	108
4.21	Verstärkung des IEPE-Interfaces	110
4.22	Im Rahmen der Messungen verwendete Geräte	116
4.23	Bestimmung der unteren Grenzfrequenzen des IEPE-Interfaces	118
4.24	Strombedarf der Teilschaltungen des IEPE-Interfaces	121

4.25 Konfiguration der Knoten während der Tests	130
4.26 Paketverlust im Stillstand	133
4.27 Paketverlust während der Fahrt	136
5.1 Signale der Konfigurationsschnittstelle des CPLDs	140
5.2 Steuer- und Kommunikationssignale	140
5.3 Registerbelegung des Konfigurationsspeichers	145
5.4 Beschreibung der einzelnen Register	145
5.5 Timing-Kennwerte der CPLD-Logik	148
5.6 Timing-Kennwerte der Bausteine	150
5.7 Strombedarf der einzelnen Module	158
5.8 Strombedarf des Messsystems im normalen Betrieb	158
5.9 LTC3112 – Widerstandswerte für $U_A = +2.5\text{ V}$	160
A.1 Ergebnisse der PER-Messungen der CC2520EM-Platinen	175
A.2 Ergebnisse der PER-Messungen der HF-Prototypen	176
A.3 Gemessene Zeitintervalle beim Senden und Empfangen von Nachrichten mittels Z-Stack	177
A.4 DOsonChip – Zum Schreiben von 10 MB benötigte Zeit	178
A.5 DOsonChip – Maximale Pufferfüllstände bei der Kingston-Speicherkarte	179
A.6 DOsonChip – Maximale Pufferfüllstände bei der SanDisk-Speicherkarte	180
A.7 DOsonChip – Maximale Pufferfüllstände bei der Transcend-Speicherkarte	180
A.8 FAT32-Firmware Treiber – Maximale Pufferfüllstände bei der SanDisk- Speicherkarte	181
A.9 FAT32-Firmware Treiber – Maximale Pufferfüllstände bei der Transcend- Speicherkarte	181
A.10 Anzahl der Taktflanken n_F und Endstand der Zeitbasis t_E der Datenspei- chermodule 0 bis 4 nach 4 h – ohne Geschwindigkeitskorrektur	182
A.11 Anzahl der Taktflanken n_F und Endstand der Zeitbasis t_E der Datenspei- chermodule 0 bis 4 nach 4 h – mit Geschwindigkeitskorrektur	183
A.12 Wirkungsgrad des LTC3112 bei $U_A = +3.3\text{ V}$ und $U_E = +5.0\text{ V}$	184
A.13 Wirkungsgrad des LTC3112 bei $U_A = +3.3\text{ V}$ und $U_E = +3.6\text{ V}$	185
A.14 Wirkungsgrad des LTC3112 bei $U_A = +3.3\text{ V}$ und $U_E = +2.7\text{ V}$	186
A.15 Wirkungsgrad des LTC3112 bei $U_A = +2.5\text{ V}$ und $U_E = +5.0\text{ V}$	187
A.16 Wirkungsgrad des LTC3112 bei $U_A = +2.5\text{ V}$ und $U_E = +3.6\text{ V}$	187
A.17 Wirkungsgrad des LTC3112 bei $U_A = +2.5\text{ V}$ und $U_E = +2.7\text{ V}$	188

A.18 Wirkungsgrad des LTC3112 bei $U_A = -2.5\text{ V}$ und $U_E = +5.0\text{ V}$	188
A.19 Wirkungsgrad des LTC3112 bei $U_A = -2.5\text{ V}$ und $U_E = +3.6\text{ V}$	189
A.20 Wirkungsgrad des LTC3112 bei $U_A = -2.5\text{ V}$ und $U_E = +2.7\text{ V}$	189
D.1 Basismodul – Interne Verbindungen des MSP430F2618	200
D.2 Basismodul – Pinbelegung Buchse J1	201
D.3 Basismodul – Pinbelegung Buchse J2	202
D.4 Datenspeicher – Interne Verbindungen des MSP430F5438	203
D.5 Datenspeicher – Pinbelegung Stecker P1	203
D.6 Datenspeicher – Pinbelegung Stecker P2	204
D.7 Datenspeicher – Pinbelegung Buchse J1	204
D.8 Datenspeicher – Pinbelegung Buchse J2	204
D.9 Allgemeine Signalaufbereitung – Pinbelegung Stecker P1	205
D.10 Allgemeine Signalaufbereitung – Pinbelegung Stecker P2	206
D.11 Allgemeine Signalaufbereitung – Pinbelegung Buchse J1	206
D.12 Allgemeine Signalaufbereitung – Pinbelegung Buchse J2	206
D.13 Allgemeine Signalaufbereitung – Jumper P3	206
D.14 IEPE-Interface – Pinbelegung Stecker P1	207
D.15 IEPE-Interface – Pinbelegung Stecker P2	207
D.16 IEPE-Interface – Pinbelegung Stiftleiste P3	208
D.17 IEPE-Interface – Pinbelegung Jumper P4	208
D.18 IEPE-Interface – Weitere Bedienelemente und Buchsen	208
D.19 USB-Interface – Interne Verbindungen des MSP430	209
D.20 USB-Interface – Pinbelegung Stecker P1	209
D.21 USB-Interface – Pinbelegung Stecker P2	210
D.22 USB-Interface – Pinbelegung Stiftleiste P3	210
D.23 USB-Interface – Bedien- und Anzeigeelemente sowie weitere Stecker bzw. Buchsen	211
D.24 Funktionsgenerator – Interne Verbindungen des MSP430F2132	212
D.25 Funktionsgenerator – Pinbelegung Stecker P1	212
D.26 Funktionsgenerator – Pinbelegung Stecker P2	213
D.27 Funktionsgenerator – Pinbelegung Stiftleisten P4, P5, P6	213
D.28 Funktionsgenerator – Festlegung von Signalform und Frequenz mittels P4, P5 und P6	213

D.29 Funktionsgenerator – Bedien- und Anzeigeelemente sowie weitere Stecker bzw. Buchsen	214
D.30 Signalverbindungen zwischen Datenspeicher und Basismodul	215
D.31 Signalverbindungen zwischen allgemeiner Signalaufbereitung und dem Da- tenspeicher	215
D.32 Signalverbindungen zwischen IEPE-Interface und allgemeiner Signalauf- bereitung	215
D.33 Signalverbindungen zwischen Datenspeicher und Basismodul	216
D.34 Signalverbindungen zwischen Funktionsgenerator und Datenspeicher	216
E.1 CPLD-Platine – Verbindungen des CPLDs mit Bedienelementen und FTDI232R	
	218
E.2 CPLD-Platine – Signalverbindungen zwischen CPLD und MSP430	219
E.3 CPLD-Platine – Pinbelegung Stiftleiste P3	220
E.4 CPLD-Platine – Pinbelegung Stiftleiste P4	221
E.5 MSP430-CPLD-Platine – Pinbelegung Stiftleiste P6	222
E.6 MSP430-CPLD-Platine – Pinbelegung Stiftleiste P7	223
E.7 MSP430-CPLD-Platine – Pinbelegung Stiftleiste P8	224
E.8 CPLD-Platine – Bedien- und Anzeigeelemente sowie weitere Stecker bzw. Buchsen	224
E.9 CPLD-Erweiterungsplatine – Pinbelegung Stecker P1	226
E.10 CPLD-Erweiterungsplatine – Pinbelegung Stecker P2	226
E.11 CPLD-Erweiterungsplatine – Pinbelegung Buchse J1	227
E.12 CPLD-Erweiterungsplatine – Pinbelegung Buchse J2	227
E.13 CPLD-Erweiterungsplatine – Pinbelegung Sockel SD-Karte J3	227
E.14 CPLD-Erweiterungsplatine – Pinbelegung Stiftleiste P9	228
E.15 CPLD-Erweiterungsplatine – Pinbelegung Stiftleiste P10	228

Literaturverzeichnis

- [1] Alippi, C. and C. Galperti: *An Adaptive System for Optimal Solar Energy Harvesting in Wireless Sensor Network Nodes*. IEEE Transactions on Circuits and Systems I: Regular Papers, 55(6):1742–1750, July 2008.
- [2] Allan, David W., James Barnes, Franco Cordara, Michael Garvey, William Hanson, Robert Kinsman, Jack Kusters, Robert Smythe, and Fred L. Walls: *Precision Oscillators: Dependence of Frequency on Temperature, Humidity and Pressure*. In *IEEE Frequency Control Symposium*, 1992.
- [3] Analog Devices: *AD5321 Datasheet*.
- [4] Anastasi, G., A. Falchi, A. Passarella, M. Conti, and E. Gregori: *Performance Measurements of Motes Sensor Networks*. In *Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '04*, pages 174–181, 2004, ISBN 1-58113-953-5.
- [5] Association, SD Card: *SD Specification Part 1 – Physical Layer Simplified Specification Version 4.0*.
- [6] Axelson, Jan: *Serial Port Complete: COM Ports, USB Virtual COM Ports, and Ports for Embedded Systems*. Lakeview Research LLC, 2007.
- [7] Ballantine, D. S., R. M. White, S. J. Martin, A. J. Rico, E. T. Zellers, G. C. Frye, and H. Wohltjen: *Acoustic Wave Sensors – Theory, Design, and Physico-Chemical Applications*. Academic Press, 1997.
- [8] Banerjee, A., A. Mitra, W. Najjar, D. Zeinalipour-Yazti, V. Kalogeraki, and D. Gunopulos: *RISE – Co-S: High Performance Sensor Storage and Co-Processing Architecture*. In *Second Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, pages 1–12, September 2005.

- [9] Baronti, Paolo, Prashant Pillai, Vince W. C. Chook, Stefano Chessa, Alberto Gotta, and Y. Fun Hu: *Wireless Sensor Networks: A Survey on the State of the Art and the 802.15.4 and ZigBee Standards*. Computer Communications, 30(7):1655–1695, May 2007.
- [10] Beitelschmidt, M., M. Harter, G. Striegler und I. Sauermann: *Die Dresdner Messstrassenbahn – Konzept, Architektur, Komponenten*. Eisenbahntechnische Rundschau, Heft Dezember, 2009.
- [11] Bluetooth SIG: *Specification of the Bluetooth System – Wireless Connections Made Easy*, 2009.
- [12] Buratti, Chiara, Andrea Conti, Davide Dardari, and Roberto Verdone: *An Overview on Wireless Sensor Networks Technology and Evolution*. Sensors, 9(9):6869–6896, 2009, ISSN 1424-8220.
- [13] Chu, Pong P.: *RTL Hardware Design Using VHDL*. Wiley-Interscience, 2006.
- [14] Chu, Pong P.: *FPGA Prototyping by VHDL Examples*. Wiley-Interscience, 2008.
- [15] Davies, John H.: *MSP430 Microcontroller Basics*. Newnes, 2008.
- [16] Dogan, Ibrahim: *SD Card Projects Using the PIC Microcontroller*. Elsevier, 2010.
- [17] Eady, Fred: *Hands-On ZigBee: Implementing 802.15.4 with Microcontrollers*. Newnes, 2007.
- [18] Elahi, Ata and Adam Gschwendner: *ZigBee – Wireless Sensor and Control Network*. Prentice Hall, 2009.
- [19] European Telecommunications Standard Institute: *ETSI EN 300 328*, 2012.
- [20] Faludi, Robert: *Building Wireless Sensor Networks*. O’Reilly, 2011.
- [21] Farahani, Shahin: *ZigBee Wireless Networks and Transceivers*. Newnes, 2008.
- [22] Farshchi, S., P.H. Nuyujukian, A. Pesterev, I. Mody, and J.W. Judy: *A TinyOS-Based Wireless Neural Sensing, Archiving, and Hosting System*. In *2nd International IEEE EMBS Conference on Neural Engineering*, pages 671–674, March 2005.
- [23] Finkenzeller, Klaus: *RFID Handbook*. Wiley, 2010.

- [24] Franklin, R. W.: *Analysis of Solid Tantalum Capacitor Leakage Current*. Technical report, AVX Corporation.
- [25] Future Technology Devices Incorporated: *FT232R Datasheet*.
- [26] Gay, David, Matt Welsh, Philip Levis, Eric Brewer Robert von Behren, and David Culler: *The nesC Language: A Holistic Approach to Networked Embedded Systems*. In *In Proceedings of Programming Language Design and Implementation*, pages 1–11, 2003.
- [27] Gay, Nicolas and Wolf-Joachim Fischer: *Ultra-low-power RFID-Based Sensor Mote*. In *IEEE SENSORS Conference*, 2010.
- [28] Gislason, Drew: *ZigBee Wireless Networking*. Newnes, 2007.
- [29] Healy, M., T. Newe, and E. Lewis: *Wireless Sensor Node Hardware: A Review*. In *Sensors, 2008 IEEE*, pages 621–624, October 2008.
- [30] Hribšek, Marija F., Dejan V. Tošić, and Miroslav R. Radosavljević: *Surface Acoustic Wave Sensors in Mechanical Engineering*. *FME Transactions*, 38:11–18, 2010.
- [31] IEEE: *IEEE Std. 802.11-2012 – IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*.
- [32] IEEE: *IEEE Std. 802.11g-2003 – IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications – Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz Band*.
- [33] IEEE: *IEEE Std. 802.11n-2009 – IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications – Amendment 5: Enhancements for Higher Throughput*.
- [34] IEEE: *IEEE Std. 802.15.4-2006 – IEEE Standard for Information Technology – Local and Metropolitan Area Networks – Specific Requirements – Part 15.4: Wireless*

- Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs).*
- [35] IEEE: *IEEE Std. 802.15.4-2011 – IEEE Standard for Local and Metropolitan Area Networks – Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs).*
- [36] Johnson, Howard and Martin Graham: *High-Speed Digital Design.* Prentice Hall, 1993.
- [37] Johnson, M., M. Healy, P. van de Ven, M.J. Hayes, J. Nelson, T. Newe, and E. Lewis: *A Comparative Review of Wireless Sensor Network Mote Technologies.* In *Sensors, 2009 IEEE*, pages 1439–1442, Oct. 2009.
- [38] Jossen, Andreas und Wolfgang Weydanz: *Moderne Akkumulatoren richtig einsetzen.* Inge Reichardt Verlag, 2006.
- [39] Juang, Philo, Hidekazu Oki, Yong Wang, Margaret Martonosi, Li Shiuan Peh, and Daniel Rubenstein: *Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet.* In *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS-X*, pages 96–107, 2002, ISBN 1-58113-574-2.
- [40] Jung, Walt (editor): *Op Amp Applications Handbook.* Newnes, 2006.
- [41] Karl, Holger and Andreas Willig: *Protocols and Architectures for Wireless Sensor Networks.* John Wiley & Sons, 2007.
- [42] Kim, Sukun, Rodrigo Fonseca, Prabal Dutta, Arsalan Tavakoli, David Culler, Philip Levis, Scott Shenker, and Ion Stoica: *Flush: A Reliable Bulk Transport Protocol for Multihop Wireless Networks.* In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems, SenSys 2007*, pages 351–365, 2007, ISBN 978-1-59593-763-6.
- [43] Kim, Sukun, Shamim Pakzad, David Culler, James Demmel, Gregory Fennes, Steven Glaser, and Martin Turon: *Health Monitoring of Civil Infrastructures Using Wireless Sensor Networks.* In *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, pages 254–263. ACM Press, 2007.
- [44] Kozlovski, N. Y. and D. C. Malocha: *SAW Passive Wireless Multi Sensor System.* In *IEEE International Ultrasonics Symposium*, 2009.

- [45] Lüders, Christian: *Lokale Funknetze*. Vogel Buchverlag, 2007.
- [46] Lee, Pius, Hwee-Pink Tan, and Mingding Han: *Demo: A Solar-powered Wireless Parking Guidance System for Outdoor Car Parks*. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, SenSys 2011, pages 423–424, 2011, ISBN 978-1-4503-0718-5.
- [47] Levis, Philip: *TinyOS Programming*. Cambridge University Press, 2011.
- [48] Lieschneegg, M., B. Lechner, A. Fuchs, and O. Mariani: *Versatile Sensor Platform for Autonomous Sensing in Automotive Applications*. *International Journal on Smart Sensing and Intelligent Systems*, 4(3):496–507, 2011.
- [49] Linden, David and Thomas B. Reddy: *Handbook of Batteries*. McGraw–Hill, 2002.
- [50] Linear Technology: *LT3092 Datasheet*.
- [51] Linear Technology: *LT3580 Datasheet*.
- [52] Linear Technology: *LTC1069-6 Datasheet*.
- [53] Linear Technology: *LTC3112 Datasheet*.
- [54] Linear Technology: *LTC3526L/LTC3526LB Datasheet*.
- [55] Linear Technology: *LTC4011 Datasheet*.
- [56] Linear Technology: *LTC6910-1/LTC6910-2/LTC6910-3 Datasheet*.
- [57] Malesci, Umberto and Samuel Madden: *A Measurement-based Analysis of the Interaction between Network Layers in TinyOS*. In *Proceedings of the European Workshop for Wireless Sensor Networks*, pages 292–309, 2006.
- [58] Malocha, D.C., J. Pavlina, D. Gallagher, N. Kozlovski, B. Fischer, and N. Saldanha D. Puccio: *Orthogonal Frequency Coded SAW Sensors and RFID Design Principles*. In *IEEE International Frequency Control Symposium*, 2008.
- [59] Maxfield, Clive: *The Design Warrior’s Guide to FPGAs*. Newnes, 2004.
- [60] Maxino, Theresa C. and Philip J. Coopman: *The Effectiveness of Checksums for Embedded Control Networks*. *IEEE Transactions on Dependable and Secure Computing*, 6(1):59–72, 2009.

- [61] Metra Mess- und Frequenztechnik Radebeul: *Applikationsschrift AN4 – IEPE-Standard*, 2008.
- [62] Meyer, J., R. Bischoff, G. Feltrin, and M. Motavalli: *Wireless Sensor Networks for Long-term Structural Health Monitoring*. *Journal Smart Structures and Systems*, 6(3):263–275, 2010.
- [63] Micheloni, Rino, Luca Crippa, and Alessia Marelli: *Inside NAND Flash Memories*. Springer, 2010.
- [64] Microchip Technology: *MCP23008/MCP23S08 Datasheet*.
- [65] Microchip Technology: *AN699 – Anti-Aliasing, Analog Filters for Data Acquisition Systems*, 1999.
- [66] Microstrain: *V-Link User Manual Version 4.0.1*, 2010.
- [67] Möckel, Rico, Alexander Spröwitz, Jérôme Maye, and Auke Jan Ijsspeert: *An easy to use bluetooth scatternet protocol for fast data exchange in wireless sensor networks and autonomous robots*. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2801 – 2806, 2007.
- [68] Moon, Todd K.: *Error Correction Coding*. Wiley-Interscience, 2005.
- [69] Musiani, Daniele, Kaisen Lin, and Tajana Simunic Rosing: *Active Sensing Platform for Wireless Structural Health Monitoring*. In *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, IPSN 2007, pages 390–399, 2007, ISBN 978-1-59593-638-7.
- [70] Nejikovsky, B. and E. Keller: *Wireless Communications Based System to Monitor Performance of Rail Vehicles*. In *Proceedings of the 2000 ASME/IEEE Joint Railroad Conference*, pages 111–124, 2000.
- [71] Numonyx: *M25P80 Datasheet*.
- [72] Österlind, Fredrik and Adam Dunkels: *Approaching the Maximum 802.15.4 Multi-Hop Throughput*. In *Proceedings of the Fifth ACM Workshop on Embedded Networked Sensors (HotEmNets)*, June 2008.
- [73] Pohl, Alfred: *A Review of Wireless SAW Sensors*. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 47(2):317–332, March 2000.

- [74] Reindl, L., G. Scholl und F. Schmidt: *Funksensorik und Identifikation mit OFW-Sensoren*. VDI Berichte, 1530:799–810, 2000.
- [75] Reindl, Leonhard M.: *Wireless passive sensors: Basic principles and performances*. In *IEEE Sensors Conference*, 2008.
- [76] Sample, Alanson P., Daniel J. Yeager, Pauline S. Powledge, Alexander V. Mamishev, and Joshua R. Smith: *Design of an RFID-Based Battery-Free Programmable Sensing Platform*. *IEEE Transactions on Instrumentation and Measurement*, 57(11):2608–2615, 2008.
- [77] Scholl, G., C. Korden, E. Riha, C.C.W. Ruppel, U. Wolff, G. Riha, L. Reindl, and R. Weigel: *SAW-Based Radio Sensor Systems for Short-Range Applications*. *IEEE Microwave Magazine*, 4:68–76, 2003.
- [78] Selavo, L., A. Wood, Q. Cao, T. Sookoor, H. Liu, A. Srinivasan, Y. Wu, W. Kang, J. Stankovic, D. Young, *et al.*: *Luster: Wireless Sensor Network for Environmental Research*. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems*, pages 103–116, 2007.
- [79] Selavo, L., G. Zhou, and J.A. Stankovic: *SeeMote: In-Situ Visualization and Logging Device for Wireless Sensor Networks*. In *3rd International Conference on Broadband Communications, Networks and Systems*, pages 1–9. IEEE, 2006.
- [80] Sikora, Axel: *Compatibility of IEEE 802.15.4 (ZigBee) with IEEE 802.11 (WLAN), Bluetooth, and Microwave Ovens in 2.4 GHz ISM-Band*. Technical report, Steinbeis Transfer Centre for Embedded Design and Networking, 2004.
- [81] Simjee, F. and P.H. Chou: *Everlast: Long-Life, Supercapacitor-Operated Wireless Sensor Node*. In *Proceedings of the 2006 International Symposium on Low Power Electronics and Design (ISLPED)*, pages 197–202, October 2006.
- [82] Sivrikaya, Fikret and Bülent Yener: *Time Synchronization in Sensor Networks: A Survey*. *IEEE Network*, 18:45–50, 2004.
- [83] Somov, Andrey: *Wildfire Safety with Wireless Sensor Networks*. *ICST Transactions on Ambient Systems*, 11(10–12):1–11, 2011.
- [84] SparkFun-Forum. <http://forum.sparkfun.com/viewtopic.php?f=14&t=5065&start=15>.

- [85] Springer, Helmut und Johann Wassermann: *Skriptum zur Vorlesung Mess- und Schwingungstechnik*, 2007.
- [86] Sundararaman, Bharath, Ugo Buy, and Ajay D. Kshemkalyani: *Clock Synchronization for Wireless Sensor Networks: A Survey*. *Ad Hoc Networks* (Elsevier), 3:281–323, 2005.
- [87] Szewczyk, Robert, Alan Mainwaring, Joseph Polastre, John Anderson, and David Culler: *An Analysis of a Large Scale Habitat Monitoring Application*. In *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, pages 214–226, 2004, ISBN 1-58113-879-2.
- [88] Tas, N.C., C. Sastry, and Z. Song: *IEEE 802.15.4 Throughput Analysis under IEEE 802.11 Interference*. In *International Symposium on Innovations and Real Time Applications of Distributed Sensor Networks*, 2007.
- [89] Texas Instruments: *CC2520 Datasheet*.
- [90] Texas Instruments: *CC2520 Software Examples (Rev. B)*.
- [91] Texas Instruments: *CC2520EM Reference Design*.
- [92] Texas Instruments: *MSP320F543x, MSP430F541x Device Erratasheet*.
- [93] Texas Instruments: *MSP320F543xA, MSP320F5431A Datasheet*.
- [94] Texas Instruments: *MSP430F241x, MSP430F261x Datasheet*.
- [95] Texas Instruments: *MSP430F543x, MSP430F541x Datasheet*.
- [96] Texas Instruments: *MSP430x1xx Family User's Guide (Rev. F)*.
- [97] Texas Instruments: *MSP430x5xx and MSP430x6xx Family User's Guide (Rev. J)*.
- [98] Texas Instruments: *OPA244, OPA2244, OPA4244 Datasheet*.
- [99] Texas Instruments: *OPA378/OPA2378 Datasheet*.
- [100] Texas Instruments: *TPS61081 Datasheet*.
- [101] Texas Instruments: *Z-Stack*. <http://www.ti.com/tool/z-stack>.
- [102] Texas Instruments: *RF Basics – Low Power Wireless*, 2007.

- [103] Tietze, Ulrich und Christoph Schenk: *Halbleiter-Schaltungstechnik*. Springer, 2009.
- [104] TinyOS Alliance: *TinyOS Projektarchiv*. <http://code.google.com/p/tinyos-main/>.
- [105] Tolle, Gilman, Joseph Polastre, Robert Szewczyk, David Culler, Neil Turner, Kevin Tu, Stephen Burgess, Todd Dawson, Phil Buonadonna, David Gay, and Wei Hong: *A Macroscopic in the Redwoods*. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, pages 51–63, 2005, ISBN 1-59593-054-X.
- [106] Transcend: *microSDHC Card Series Datasheet*.
- [107] Union International Telecommunication: *The Radio Regulations*, 2008.
- [108] Vishay Siliconix: *DG469, DG470 Datasheet*.
- [109] Werner-Allen, G., K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh: *Deploying a Wireless Sensor Network on an Active Volcano*. Internet Computing, IEEE, 10(2):18–25, March–April 2006.
- [110] Wilson, W. C., D. C. Malocha, N. Kozlovski, D. R. Gallagher, B. Fischer, J. Pavlina, N. Saldanha, D. Puccio, and G. M. Atkinson: *Orthogonal Frequency Coded SAW Sensors for Aerospace SHM Applications*. IEEE Sensors Journal, 9:1546–1556, 2009.
- [111] Wissenwasser, J., M. Vellekoop, and R. Heer: *Highly Sensitive Passive Radio Frequency Identification Based Sensor Systems*. AIP – Review of Scientific Instruments, 81:025106 1–5, 2010.
- [112] Xu, Ning, Sumit Rangwala, Krishna Kant Chintalapudi, Deepak Ganesan, Alan Broad, Ramesh Govindan, and Deborah Estrin: *A Wireless Sensor Network for Structural Monitoring*. In *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, pages 13–24, 2004, ISBN 1-58113-879-2.
- [113] Yick, Jennifer, Biswanath Mukherjee, and Dipak Ghosal: *Wireless Sensor Network Survey*. Computer Networks, 52(12):2292–2330, 2008.
- [114] Zimmermann, Hubert: *OSI Reference Model – The ISO Model of Architecture for Open Systems Interconnection*. IEEE Transactions on Communications, 28:425–431, 1980.