

---

# Praxisprojekt

## Bachelor Elektrotechnik

### **Batteriemonitorsystem für mehrere Bleibatterie-Blöcke**

vorgelegt von: Güntürel Akinci

eingereicht bei: Prof. Dr. Eberhardt Waffenschmidt

Lindlar, 13.11.2023



Cologne Institute for  
Renewable Energy

**Technology**  
**Arts Sciences**  
**TH Köln**

## Erklärung

Name: Güntürel Akinci

Martikel-Nummer: 11082180

### Erklärung zum eigenständigen Verfassen

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit selbständig verfasst habe. Ich habe keine anderen außer den von mir angegeben Quellen und Hilfsmittel verwendet. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

13.11.2023

Güntürel Akinci

### Erklärung zur Veröffentlichung

Ich bin damit einverstanden, dass mein Praxisprojekt ausgeliehen werden darf. Sie darf von meinem Betreuer im Internet veröffentlicht werden.

13.11.2023

Güntürel Akinci

### Erklärung zu Bildrechten

Außer den im Folgenden genannten habe ich alle Bilder und Diagramme dieses Praxisprojekts selbst erstellt.

Die folgenden Bilder sind von anderen Autoren übernommen. Die Autoren sind in meiner Arbeit entsprechend genannt. Sie dürfen aufgrund einer öffentlichen Lizenz veröffentlicht werden.

Bild 9: Hardware Konfiguration Datenblatt "DS2438"

Bild 11: Verschiedene Entwicklungsboards

Tabelle 3: Befehlssequenz "Smart Batterie Monitor" Datenblatt „DS2438“

13.11.2023

Güntürel Akinci

**Lindlar, 13.11.2023**

---

Ort, Datum

---

Rechtsverbindliche Unterschrift

## Inhaltsverzeichnis

Erklärung.....	II
Erklärung zum eigenständigen Verfassen.....	II
Erklärung zur Veröffentlichung.....	II
Erklärung zu Bildrechten.....	II
Inhaltsverzeichnis.....	III
1 Einleitung.....	1
2 Stand der Technik.....	2
3 Abkürzungsverzeichnis.....	3
4 Hintergrund.....	4
4.1 One-Wire-BUS.....	4
4.1.1 Einsatzbereich.....	5
4.1.2 Versorgung Busmaster & Parasitärer Betrieb.....	5
4.1.3 Vor und Nachteile der 1-Wire Technologie.....	6
4.1.4 Topologie und Installation.....	7
4.1.5 Kommunikation.....	9
4.2 Master.....	14
4.2.1 Arduino.....	15
4.2.2 Ethernet-Shield.....	16
4.3 Slave.....	17
5 Inhalt.....	18
5.1 Batteriemonitoringssystem für mehrere Bleibatterie-Blöcke.....	18
5.2 Aufbau der Anlage.....	19
5.2.1 „1-Wire Batterie-Monitor“ Board V3.0.....	20
5.2.2 Ansteuerung über die IDE.....	27
5.3 Interpretation.....	29
6 Fazit.....	32
7 Verzeichnisse und Anhang.....	34
Abbildungsverzeichnis.....	35
Literaturverzeichnis.....	37
Anhang:.....	40

# 1 Einleitung

Um eine zuverlässige und sichere Energieversorgung mit Hilfe von Energiespeichern zu ermöglichen sind intelligente Überwachungssysteme für individuelle Einrichtungen unabdinglich.

Mit ihrer Hilfe können Anlagen, Versorgungsstationen und anderen Einrichtungen, die den Nutzen von Energiespeichern beziehen, mit einer grundlegenden Sicherheit ausgestattet werden. Sicherheiten die kleine und große Schäden an eigenen und anderen Komponenten verhindern sollen. Anhand von zeitnah erfassten Daten wird eine übersichtliche Überwachung der Anlage gewährleistet.

Im folgenden Projekt wird der Entwicklungsprozess für die Nutzung des *1-Wire-Bus* Systems vorgestellt, welches für eine Überwachung von Bleibatterie-Blöcken dient. Diese Überwachung ermittelt, speichert und überträgt die signifikante Werte von mehreren Energiespeichern und lässt eine zuverlässige Kontrolle und Wartung zu.

Ziel dieser Arbeit ist es mit Hilfe des 1-Wire-Busses, welches in diesem Fall als Fundament des Systems dient, eine zuverlässige Datenübertragung zu schaffen, um mit Hilfe dieser eine übersichtlichen Ausgabe der signifikanten Werte zu gewährleisten.

In Form eines Praxisprojekts werden im Projekt „PROGRESSUS“ jeweils 24 Platinen für die Datenerfassung der dazugehörigen Batterie-Blöcke entwickelt und implementiert. Dies erlaubt eine Überwachung zu realisieren die für Nutzer eine Möglichkeit bietet eine übersichtliche Kontrolle zu gewährleisten und gegebenenfalls ein Einschreiten zu ermöglichen, um Störungen und Schäden vorzubeugen.

## 2 Stand der Technik

Im Rahmen dieses Projekts hat ein Kommilitone bezüglich dieser Arbeit erste Schritte zur Realisierung des Vorhabens getätigt.

Name: Mohammed Amine

Nachname: Jazouli

Form: Praxisprojekt

Thema: Inbetriebnahme, Optimierung und Dokumentation eines Batterie Monitors auf Basis des One Wire Bus

### Meilensteine:

1. Das Layout der Schaltung von Prof. Dr. Eberhardt Waffenschmidt bearbeitet und angepasst (1Wire-Batterie-Monitor vom 12.04.2021) V2.0.
2. 24 Platinen zur Messung der Spannung und der Temperatur der jeweiligen Bleibatterie-Blöcke gebaut.
3. Die Software vom Arduino programmiert, zur Erkennung der Werte, die von den Sensoren erfasst und gesendet werden.
4. Alles miteinander verschaltet

Diese Arbeit vom Kommilitonen wurde zu Beginn als Bezugsmaterial verwendet, um eine eigene Arbeit zu ermöglichen. Die vorherigen Fehler und Ungenauigkeiten wurden berücksichtigt, um eine verbesserte Version und Ausführung zu ermöglichen.

Die Fortschritte in dieser Arbeit können auf die folgenden Unterpunkte kurz zusammengefasst werden.

1. Die Topologie der Schaltung (1Wire-Batterie-Monitor) angepasst und verbessert.
2. Das Layout für die Schaltung (1Wire-Batterie-Monitor vom 12.04.2021) von Prof. Dr. Eberhardt Waffenschmidt wurde auf eine erweiterte Version V3.0 zum Zeitpunkt 26.04.2022 angepasst.
3. 24 neue Platinen zur Datenübertragung gebaut, getestet und angeschlossen.
4. Software für den Mikrocontroller umgeschrieben angepasst und komprimiert.
5. 16 Platinen mit den Batterie-Blöcken verbunden und angeschlossen und erfolgreich getestet
6. Erfolgreiche Kommunikation aller Platinen mit den vorgesehenen Mikrocontrollern ermöglicht.
7. Erste Datenübertragung vom Mikrocontroller über das lokale Netzwerk an Hauptsteuerung programmiert.

### 3 Abkürzungsverzeichnis

VCC	Spannungsversorgung (+5V/+3,3V)
DQ	Datenbus (1-Wire BUS)
GND	Erdung (Ground)
μP	Mikrocontroller (BUS-Master)
IP	Internet Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
KNX	Kürzel: Gebäudeautomation
DALI	Digital Addressable Lighting Interface
BMS	Batteriemanagementsystem
RPB	Reset-Pulsbreiten

## 4 Hintergrund

### 4.1 One-Wire-BUS

Der Ursprung des 1-Wire-Bus Systems befindet sich drei Jahrzehnte zurück. Die Firma Dallas Semiconductor entwickelte diese und in der Firma Maxim Integrated ist sie aufgegangen [1]. Diese Technologie nutzt eine einfache Möglichkeit der Kommunikation, die aus drei wesentlichen Komponenten besteht. Für eine Kommunikationsverarbeitung wird ein Host (BUS-Master), ein Bauteil (Slave), welches die signifikanten Daten sendet und zuletzt ein Medium zur Übertragung der Daten (1-Wire BUS) verwendet. Diese kann in der kleinsten Ausführung nur „eine“ Ader, für einen Informationsaustausch beinhalten. Das Besondere daran ist, dass über diese eine Leitung auch gleichzeitig eine Spannungsversorgung, die für anliegende Bauteile benötigt werden, gewährleistet werden kann (parasitäre Versorgung) [2]. Zu Beginn wurden ausschließlich Bausteine verwendet, die nur eine Leitung verwendet haben. Durch Erweiterungen der Technologie und den wachsenden Anforderungen, werden aktuell kaum noch Module und Anwendungen genutzt, die die native 1-Wire Bausteine beinhalten. [2] Gängige Konstruktionen beinhalten 2-3 Adern (VCC, DQ und GND). Falls angeschlossene Bauteile keine eigene Spannungsversorgung verwenden, müssen diese anderweitig versorgt werden (externe Versorgung) [3]. Daher ist der Begriff „1-Wire“ irreführend und kann in diesem Zusammenhang falsch verstanden werden [4]. Die Kommunikation der Daten erfolgt seriell und bidirektional, dies ermöglicht einen Austausch von Signalen, welche in beide Richtungen zur selben Zeit empfangen und gesendet werden können [5]. Der Aufbau dieser Anwendung besteht aus dem One-Master/Many-Slaves Prinzip, welche als Sender und Empfänger fungieren. Der BUS-Master ist die Steuereinheit, er trägt der Signalisierung der einzelnen Komponenten miteinander bei. Dieser wird Anhand der Anforderungen für individuelle Anwendungen bestimmt. [6] Slaves werden als Datengeber genutzt diese haben eine Vielzahl von Einsatzmöglichkeiten. Ihre große Auswahl decken viele Anforderungen ab. Das Bussystem als Medium ist sehr leistungsfähig, zuverlässig und wegen seiner einfachen Bauweise und Energieersparnis sehr gefragt und geschätzt [7].

Im Folgenden wird eine beispielhafte Veranschaulichung der Schnittstellen zur Vereinfachung des One-Masters/Many-Slaves Prinzips gezeigt.

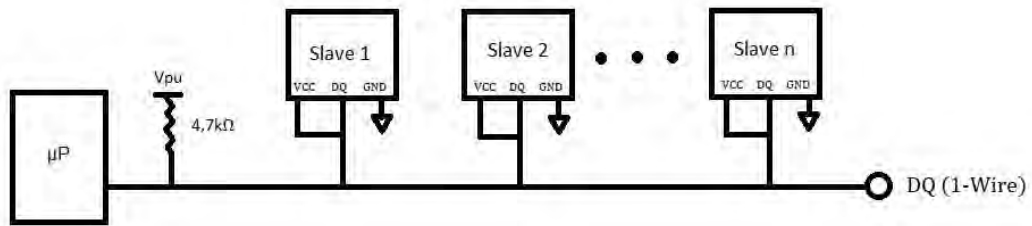


Abbildung 1: Native 1-Wire Bausteine mit Linearer Topologie Ausführung (keine GND-Ader)

In der oben gezeigten Abbildung wird die native 1-Wire Technologie verdeutlicht. Zu erkennen sind Bausteine, die in der Linien-Topologie mit Abzweigung zueinander verschaltet sind, diese greifen alle gleichzeitig auf denselben Daten-BUS. Um ein funktionierendes System zu gewährleisten ist einer der Voraussetzung, dass alle Bauteile (Slaves) mit Strom versorgt werden. Dies kann durch eine Versorgung über den Busmaster oder durch eine parasitäre Versorgung, realisiert werden. Dabei ist zu beachten, dass die Bauteile, die auf ihre individuellen Kompatibilitäten ausgelegt sind, nicht auf andere Ausrichtungen ohne weiteres genutzt werden können. Dies würde Störungen oder Funktionsbeeinträchtigungen mit sich führen.

#### 4.1.1 Einsatzbereich

Immer beliebter wird die Nutzung von 1-Wire wie z.B. in der Gebäudeautomatisierung. Hier werden Anwendungen in Form von digitalen Umweltsensoren zur Erfassung von Temperaturen, Luftfeuchtigkeit, Lichteinfall und vieles mehr verwendet. Für intelligente Haussysteme (Smart Homes) wird diese Technologie im Rahmen einer Überwachung und Ansteuerung fest implementiert. Ein weiterer Anwendungsbereich sind iButtons und RFID-Chips, welche regelmäßig bei der Personenidentifikation eine tägliche Verwendung haben [8].

#### 4.1.2 Versorgung Busmaster & Parasitärer Betrieb

##### Busmaster Betrieb:

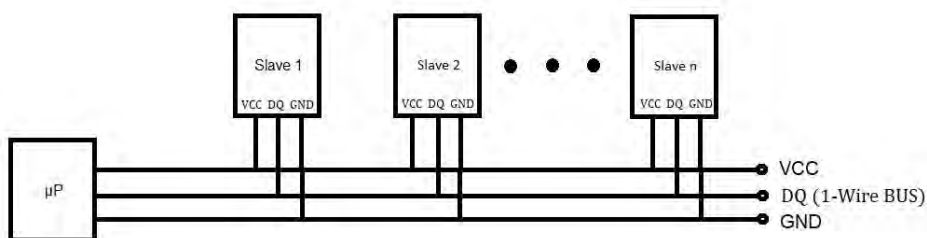


Abbildung 2: BUS-Master als Host und als Spannungsversorger

Folglich ist zu erkennen das alle Leitungen in Linie mit jedem einzelnen Slave verbunden sind. Ein BUS-Master hat in der Regel mindestens eine Versorgungsquelle, diese



kann genutzt werden, um Bauteile mit Energie zu versorgen. Abhängig der Versorgungsdimension der Slaves wäre eine Anordnung von bis zu 32 Slaves mit einer maximalen Gesamtlänge von 300m möglich. [9]

Um die Nutzung von Bauteilen mit einer höheren Stromaufnahme zu gewährleisten, ermöglicht der externe Betrieb eine zuverlässigere Versorgung. Bei noch höheren Anforderungen muss ein weiteres separates Netzteil zur Versorgung dienen. [7]

#### Parasitärer Betrieb:

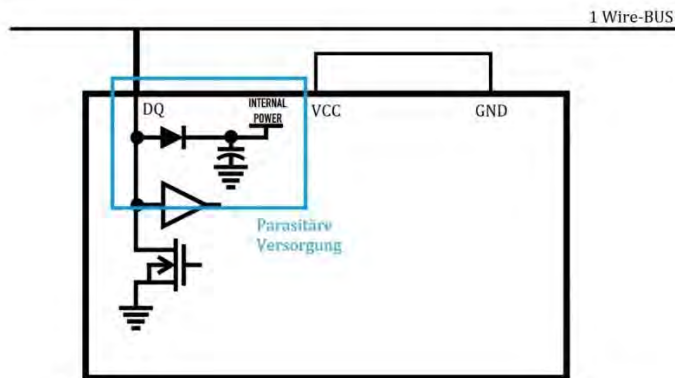


Abbildung 3: Parasitäre Versorgung

Deutlich zu erkennen ist, dass die Spannungsversorgung über die Datenleitung erfolgt. Im oben veranschaulichten Beispiel wird durch Kondensatoren, die in den Bauteilen implementiert sind, die Versorgung über die Datenleitung gewährleistet. Der BUS pendelt zwischen dem High-Pegel (3,3V oder 5V) und dem Low-Pegel (0V), in den Phasen wo der BUS auf High ist, werden die Kapazitäten der Bauteile auf die Anliegende Spannung geladen. Während das Bauteil in diesem Betriebszustand ist, müssen die Anschlüsse der Spannungsversorgung und der Erdung miteinander verbunden werden. In der Praxis werden beide Anschlüsse auf Masse gelegt, diese muss mit dem GND vom Host verbunden sein. [7]

Abhängig von der Belastung und der Versorgungsstärke des Datenbusses können nur bestimmte Bauteile, die über den parasitären Betrieb kompatibel sind, in einer begrenzten Anzahl verwendet werden. Durch die steigende Anzahl an angeschlossenen Bauteilen, steigt auch die gesamte Kapazität der Schaltung. Diese hat den Nachteil das sie bei einem zu hohen Wert einen negativen Einfluss auf die Schaltgeschwindigkeit (Timing-Protokolle) der Anordnung haben kann. [10]

#### 4.1.3 Vor und Nachteile der 1-Wire Technologie

##### Vorteile:

- Einfache Konstellation, im Vergleich zu anderen Bussystemen (KNX, DALI, etc.)
- Günstige Bauteile (Sensoren & Gateways)

- Hohe Genauigkeiten bei der Temperaturmessung
- Keine Messwertungenauigkeiten oder gar Fälschung durch hohe Leitungslängen und Störeinflüsse
- Übersichtliche und einfache Zuordnung, dank fester Adressen bei den Sensoren

#### Nachteil:

- Auslesegeschwindigkeit (relativ langsam für schnellere Anforderungen)
- Für lange Leitungslängen sind nicht alle Topologien möglich [7]

### 4.1.4 Topologie und Installation

Die einfache Bauweise ermöglicht eine Vielzahl von Verkettungen mehrerer Slaves miteinander, abhängig von den Anforderungen kann ein großes Spektrum von Sensoren in verschiedenen Verschaltungstopologien zusammengeführt und verwendet werden.

#### 4.1.4.1 Topologie

Im Wesentlichen unterscheidet man unter drei möglichen Topologien: Linie, Linie mit Abzweigen und Stern. Abhängig der Bedingungen und Erwartungen an der Anwendung, wird die Anordnung in einem der drei Formen ausgelegt.

##### Linie:

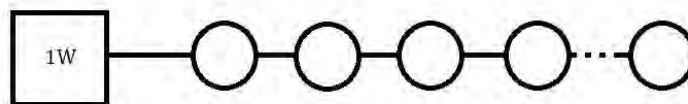


Abbildung 4: Topologie Linie

In dieser Anordnung sind alle Slaves direkt und ohne Abzweige an der Datenleitung verbunden. Diese Topologie ist die empfohlene Auslegungsform. [9]

##### Linie mit Abzweigen:

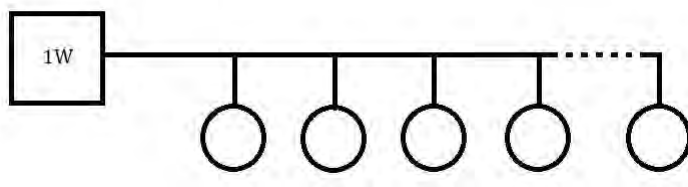


Abbildung 5: Topologie Linie mit Abzweig

Der einzige Unterschied zur vorherigen Anordnung ist dieser, dass bei jedem Slaven ein Abzweig dazugekommen ist. Dadurch verringert sich die maximale Reichweite. Bei steigender Länge der Abzweige, verringert sich /die Reichweite und die mögliche Anzahl an Bauteilen.

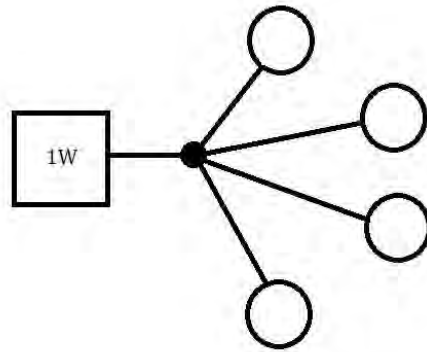
Stern:

Abbildung 6: Topologie Stern

Alle Bauteile werden an einen Punkt zusammengeführt. Dies hat den Vorteil eine lange Hauptleitung zu meiden. Gleichzeitig werden die einzelnen Abzweige pro Bauteil länger. Dies verringert die Reichweite und die Anzahl an möglichen Slaves im Gesamtgewicht deutlich. [11]

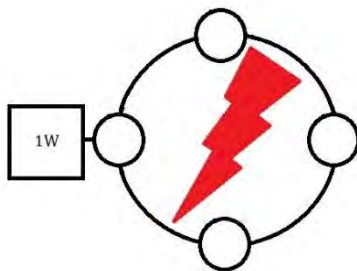
Kreis:

Abbildung 7: Topologie Ring

Diese Anordnung ist für keine Anwendung zulässig, wird der letzte Slave mit dem Hauptstrang zu Beginn verbunden funktioniert die Anordnung nicht, der BUS verliert seine Funktion.

Die optimale Topologie ist die Ausführung der Slaves in **Linie** zueinander. Dadurch wird die Reichweite und die Anzahl an möglich angeschlossenen Slaves maximiert. In der Praxis ist vor Ort eine Installation dieser Topologie ohne Abzweige meistens nicht möglich. Dennoch weist diese Form die geringste Reflexion auf und ermöglicht dadurch die höchste Zuverlässigkeit. Die ungünstigste Auslegung ist die Stern Topologie, in der Praxis ist diese jedoch die häufigste. Die Reichweite ist in dem Fall stark eingeschränkt. In gewissen Grenzen können diese Topologien gemischt werden. Zu berücksichtigen sind jedoch die Längeneempfehlungen und die maximale Anzahl der Slaves, bei der beteiligten Topologie mit den geringsten Angaben. [7]

#### 4.1.4.2 Installation

Sobald die Anforderungen von der Anordnung und die Möglichkeiten vor Ort bekannt sind kann der Radius und das Gewicht der Anlage bestimmt werden. Dies sind die wesentlichen Faktoren für den 1-Wire BUS. Der Radius wird gemessen von der entferntesten Komponente zum Master in Metern. Die Summe aller verbundenen Kabel im System bestimmt das Gewicht der Anlage. [9]

#### 4.1.4.3 Pullup-Widerstand

Zusätzlich ist ein Pullup-Widerstand für jede Anordnung unabdinglich. Dieser reduziert die Störanfälligkeit des 1-Wire BUS'es, und reguliert den Pegel der Datenleitung. Abhängig der Höhe der Betriebsspannung variiert der Widerstand, anhand folgender Formel kann dieser ausgerechnet werden.

$$R = \frac{U}{I}$$

Folglich ist diese einfache Rechnung nur für kurze Leitung zwischen Sender und Empfänger gültig. Bei langen Leitungen muss der Leitungswiderstand und daher der Spannungsabfall auf jener Verbindung berücksichtigt werden. Dazu ist die Anzahl der angeschlossenen Geräte relevant, diese haben einen Einfluss auf den tatsächlichen Widerstand. Um einen genauen Wert zu ermitteln, muss die tatsächliche Anliegende Spannung gemessen werden. [3]

#### 4.1.5 Kommunikation

Der Datentransfer zwischen Sender und Empfänger erfolgt über den 1-Wire BUS, dieser ist beim Slave auf den I/O Port ausgelegt. Grundsätzlich beinhaltet die Kommunikation die strikte Einhaltung eines vorgegebenen Timing-Protokolls. [6] Da die Verbindung seriell und bidirektional arbeitet ist eine Datenübertragung in beiden Richtungen gleichzeitig gegeben, eine weitere Leitung für das Taktsignal ist nicht von Nöten, es wird im Halbduplexverfahren übertragen. [12]

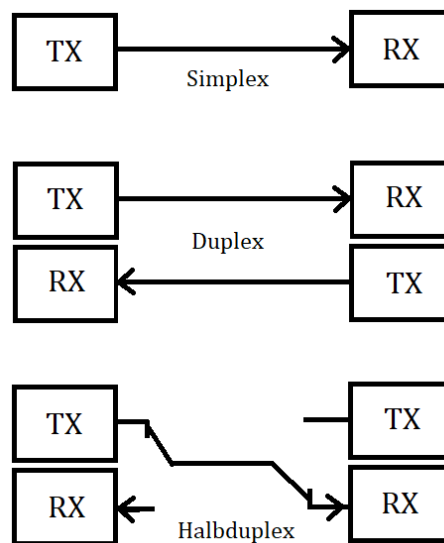


Abbildung 8: Kommunikationstechnik

Für eine Kommunikation bedarf es in der kleinsten Ausführung (Simplex) an einer Quelle (TX) und einer Senke (RX). In der praktischen Anwendung ist diese Art der Kommunikation beispielsweise in der Rundfunk- oder Fernsehübertragung vorhanden. Eine weitere Übertragungsform die eine Kommunikation in beiden Richtungen gleichzeitig ermöglicht, ist der sogenannte Gegenbetrieb. Dieser wird im Duplex oder auch Dual-Simplex genannt übertragen, für diese Form der Kommunikation sind Quelle und Senke in jeweils Sender und Empfänger nötig. Diese Übertragungsform ist gängig in Gegensprechanlagen und lokalen Netzwerken (LAN) letzteres in der „1000Base-T“ Technologie.

Falls nur ein Übertragungsmedium genutzt wird, wie es in der 1-Wire Technologie üblich ist, benötigt diese Form eine Umschaltvorrichtung, die den Wechselbetrieb auf dem 1-Wire Bus koordiniert. [13]

In diesem Betriebszustand kann zur gleichen Zeit, allerdings nur in einer Richtung kommuniziert werden. Daher die Bezeichnung „Wechselbetrieb“, der Sender und Empfänger müssen sich abwechseln, eine zeitlich verschachtelte Kommunikation wird gewährleistet. Bestes Beispiel ist der Funkverkehr mit Walkie-Talkies. [14]

#### 4.1.5.1 Hardware-Konfiguration

Im Folgenden wird die Hardware-Konfiguration einer typischen 1-Wire Kommunikation zwischen Master und Slave verdeutlicht:

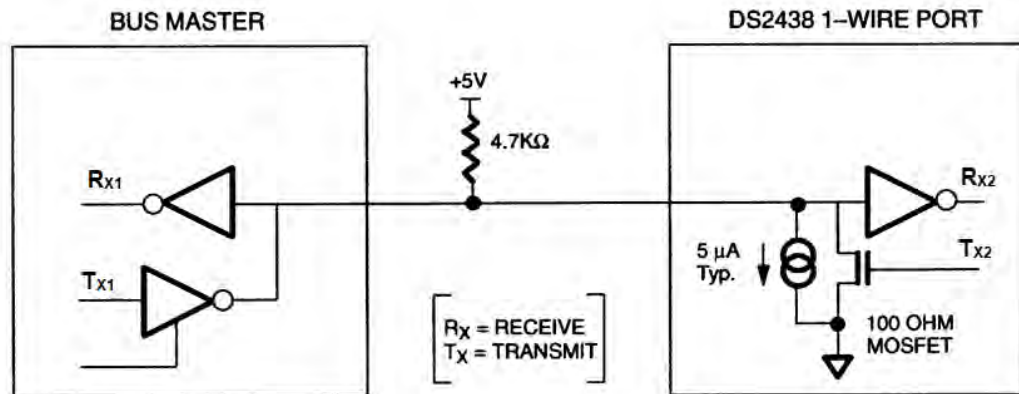


Abbildung 9: Hardware Konfiguration Datenblatt "DS2438"

In dieser Abbildung wird der „DS2348“ IC von der Firma „Maxim Integrated“ zur Veranschaulichung als Bauteil (Slave) genutzt. Deutlich wird die Form der Kommunikation aus den vorherigen Definitionen des „Wechselbetriebs (Halb-Duplex)“. Bus-Master und Slave/s haben beide jeweils eine Quelle  $T_{X1}$  &  $T_{X2}$  sowie eine Senke  $R_{X1}$  &  $R_{X2}$ . Die Umschaltung vom Senden zu einem oder mehreren Slaves gleichzeitig und das Empfangen von Daten aller Bauteile gleichzeitig an den Master und umgekehrt wird vom BUS-Master geregelt. Die Slaves passen Ihre Kommunikation an die vorgegebene Richtung des Datenflusses an, Ausnahme ist der Anwesenheitspuls. Die Anbindung der/s Slave/s am Open-Drain- oder auch der 3-State-Ausgänge ist für eine Kommunikation Voraussetzung, diese erfordern strenge Protokolle, um die Datenintegrität zu gewährleisten. [6]

#### 4.1.5.2 Signalisierung

Im Grunde besteht die Kommunikation beim 1-Wire aus insgesamt vier Operationen.

- Reset auslösen
- Logische 1 senden
- Logische 0 senden
- Ein Bit einlesen

Um das Timing-Protokoll nutzen zu können, müssen feste Strukturen definiert werden. Alle grundlegenden Operationen haben ihren festen Ablauf aus High & Low-Pegeln. Zudem wird eine bestimmte Zeitvorgabe vorausgesetzt, um signifikante Ansteuerungen und Auswertungen zu ermöglichen. [6]

Im Folgenden werden die vier Operationen im Detail veranschaulicht:

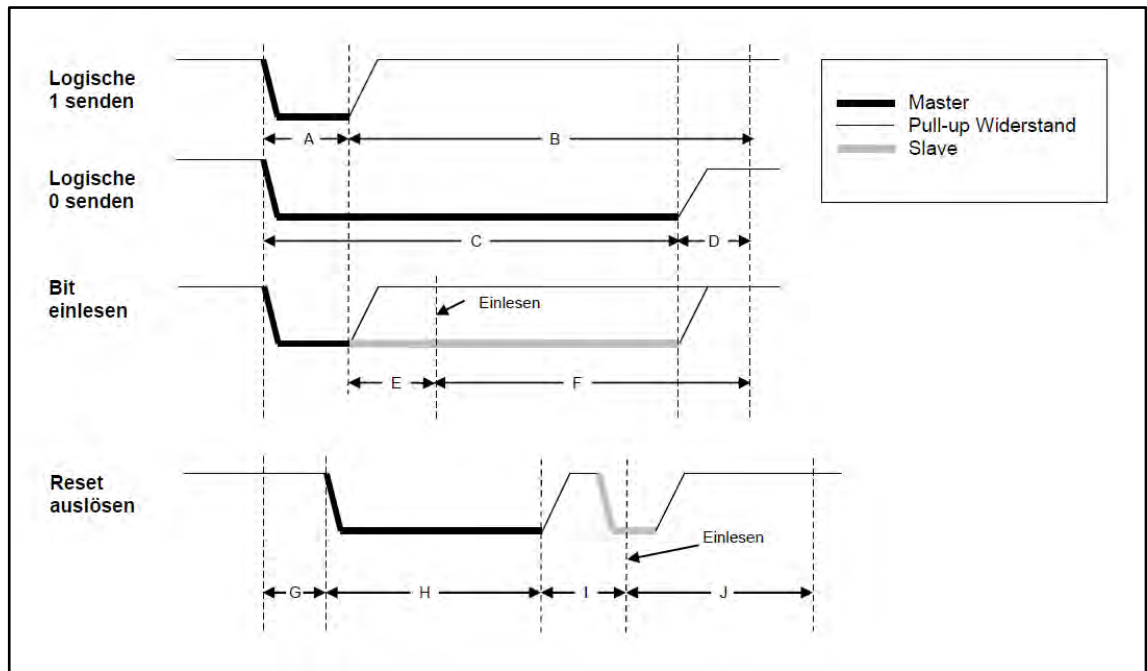


Abbildung 10: 1-Wire Verzögerungszeiten

Mit folgender Tabelle werden die zeitlichen Einteilungen der steigenden und fallenden Flanken definiert.

Tabelle 1: 1-Wire Operationen

Operation	Beschreibung	Implementierung
Logische 1 senden	Eine '1' an die 1-Wire Slaves senden	Bus auf Low '0' ziehen, Delay 'A', Bus freigeben, Delay 'B'
Logische 0 senden	Eine '0' an die 1-Wire Slaves senden	Bus auf Low '0' ziehen, Delay 'C', Bus freigeben, Delay 'D'
Bit einlesen	Ein Bit von den 1-Wire Slaves einlesen	Bus auf Low '0' ziehen, Delay 'A', Bus freigeben, Delay 'E', Bus einlesen, Delay 'F'
Reset auslösen	Die 1-Wire Slaves zurücksetzen	Delay 'G', Bus auf Low '0' ziehen, Delay 'H', Bus freigeben, Delay 'I', Bus einlesen, Delay 'J'

Für die Verzögerungen gelten folgende Werte:

Tabelle 2: 1-Wire Master Timing

Parameter	Verzögerungszeit [µs]	
	Standard	Overdrive
A	6	1,0
B	64	7,5
C	60	7,5
D	10	2,5
E	9	1,0
F	55	7
G	0	2,5
H	480	70
I	70	8,5
J	410	40

Zusätzlich zum Standardbetrieb ist der Master auch in der Lage einen „Overdrive Modus“ zu bedienen. In diesem schnelleren Betrieb wird eine höhere Datenübertragung von bis zu 142 Kbit/s erzeugt.

Anhand der Operation kommuniziert der Master mit dem/n Bauteil/en. Nach der erfolgreichen Implementierung des 1-Wire Busses wird eine Plattform für die Realisierung benötigt, diese finden sich beispielsweise in Form von Mikrocontrollern. Anhand fester vordefinierter Bibliotheken und 1-Wire Treibern werden Befehle zur Datenerfassung genutzt. In der Praxis bedeutet dies, dass die Ansteuerung des Masters ein Austausch von Signalen durch die Verwendung der definierten Verzögerungen im Zeitverlauf eine Manipulation von Bits und Befehlsworten zur Spezifizierung der Transaktion verwendet, die gelesen oder geschrieben werden. Eine typische Kommunikation zwischen Master und Slave wird am Beispiel des „DS2438“ verdeutlicht:



Tabelle 3: Befehlssequenz "Smart Batterie Monitor" Datenblatt „DS2438“

MASTER MODE	DATA (LSB FIRST)	COMMENTS
TX	Reset	Reset pulse
RX	Presence	Presence pulse
TX	CCh	Skip ROM
TX	44h	Issue Convert Temperature command, Read Slots
TX	Reset	Reset pulse
RX	Presence	Presence pulse
TX	CCh	Skip ROM
TX	B4h	Issue Convert Voltage command, Read Slots
TX	Reset	Reset pulse
RX	Presence	Presence pulse
TX	CCh	Skip ROM
TX	B8h00h	Issue Recall Memory page 00h command
TX	Reset	Reset pulse
RX	Presence	Presence pulse
TX	CCh	Skip ROM
TX	BEh00h	Issue Read SP 00h command
RX	<9 data bytes>	Read scratchpad data and CRC. This page contains temperature, voltage, and current measurements.
TX	Reset	Reset pulse
RX	Presence	Presence pulse, done

Nach Abfrage der Anwesenheit und der darauffolgenden Initialisierung, der Übermittlung der internen Kommandos und der Übermittlung der Funktionsbefehle zur Konvertierung der bauteilspezifischen Daten, wird schließlich eine Informationsabfrage vorgenommen. Im oben genannten Beispiel werden die gemessenen Werte zum Ende der Befehlskette für Temperatur, Spannung und Strom als „9 data Byte“ an den Master gesendet. Dieser Prozess ist für eine 1-Wire Übertragung üblich, eine strikte Einhaltung des Ablaufs ist Voraussetzung für die Verwirklichung. Je nach Auslegung und Eigenschaft der Slave/s werden Verschlüsselungen verwendet. Genaue Informationen bezüglich Kommunikation sind in den jeweiligen Datenblätter zu entnehmen. [6]

## 4.2 Master

Für die Wahl des richtigen Busmasters sind Anforderungen, Einschränkungen und Kompatibilitäten ausschlaggebend. Je nach Einsatzgebiet, Gewicht und Auslegung wird die Wahl der passenden Komponente entschieden. Im Grunde besitzt jeder Master einen Mikrokontroller, der zur Ansteuerung und Verwaltung von individuellen Anforderungen genutzt werden kann, im Fall vom 1-Wire, wird dieser zur Busverwaltung genutzt. Der Bereich der Busverwaltung über einen Master reicht von kleinen privaten Projekten bis hin zu professionellen Anwendungen in der Industrie. Für ersteres werden öfters so genannte „Entwicklungsboards“ genutzt. Moderne Entwicklungsboards sind leistungsfähige Nachfahren der Elektronik- Experimentierkästen. Durch die Verknüpfung mit selbst geschriebenen Programmen für individuelle Anwendungen und der Verschaltung von Zusatzelektroniken ergibt sich eine sehr leistungsfähige Kombination.

Jedoch waren Entwicklungen mit den Boards zunächst sehr unübersichtlich und in der Praxis schwierig in der Umsetzung. Dies änderte sich mit der Einführung benutzerfreundlicher Varianten wie das Arduino, Rasperry Pi und ähnlichen Einplatinen-Systemen. [15]

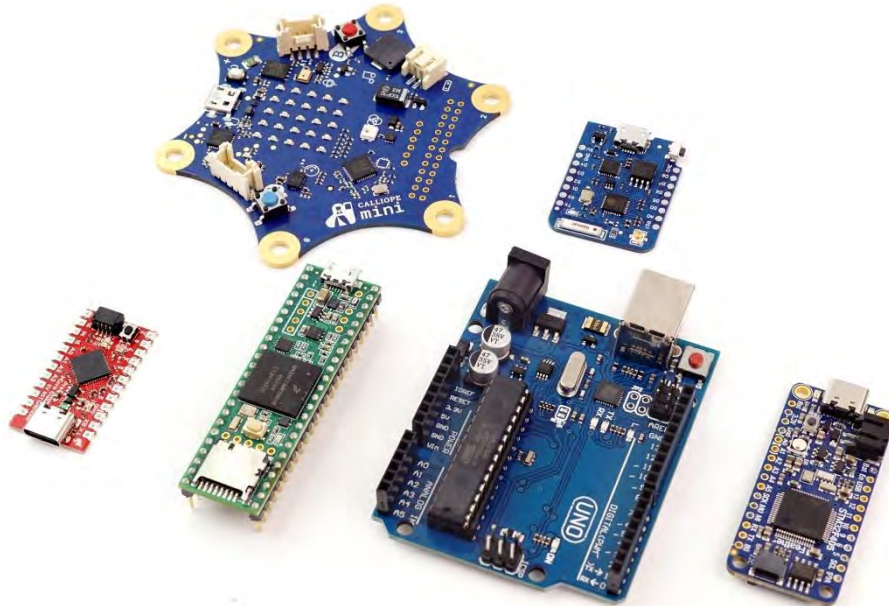


Abbildung 11: Verschiedene Entwicklungsboards

Nach einer immer größer werdenden Beliebtheit wurden diese Boards gezielt für eine massenhafte Verbreitung entworfen. Durch die hohe Stückzahl an produzierten Platinen reduzierte sich der Preis. Auch das Erlangen einer Entwicklungsumgebung wurde für die Mehrheit einfacher. Eine große Internet-Community entstand und es bildeten sich Vernetzungen von Benutzern miteinander. Daher bietet diese Möglichkeit eine zuverlässige Alternative zu professionellen, meist teuren Varianten an. [15]

#### 4.2.1 Arduino

Der Arduino hat seinen Ursprung in Italien, 2005 haben die Gründer Massimo Banzi und David Cuartielles auf Basis von Hernando Barragans entwickelten Board, ihre erste eigene Ausführung entwickelt. [16]

Bei Arduino handelt es sich um quelloffene Soft- und Hardware, diese beiden Komponenten ergeben die Physical-Computing Plattform. [17]

Die Hardware ist im Wesentlichen ein einfaches E/A-Board mit einem Mikrokontroller. Alle typischen Arduino-Boards beinhalten einen AVR-Chip. Zudem sind digitale Ein- und Ausgänge, analoge Eingänge, ein Timer sowie ein kleiner Speicher verbaut. Abhängig davon welches Board für welches Projekt genutzt wird, gibt es Unterscheidungen in der Speichergröße und Auslegung. Falls Funktionen den Anforderungen nicht genügen, gibt

es Erweiterungsplatinen (Shields) die einen größeren Nutzungsbereich ermöglichen. Damit wird eine Grundversorgung von Anbindungen und Möglichkeiten geschaffen.



Abbildung 12: Arduino Uno R3 mit Schutzkäfig

Zur Programmierung des Arduino Boards bittet der Hersteller die eigene Entwicklungsumgebung „Arduino IDE“ an. Gängige Betriebssysteme wie Windows, MacOS und Linux können mit öffentlich zugänglichen Treibern & Compilern diese Entwicklungsumgebung nutzen. Zur Programmieroberfläche werden diverse Bibliotheken eingebunden, diese ermöglichen eine Nutzung der Programmiersprachen C & C++. [18]

#### 4.2.2 Ethernet-Shield

Das Ethernet-Shield ist eine Erweiterungsplatine, mit ihrer Hilfe kann zwischen dem Arduino und dem Internet eine Verbindung ermöglicht werden. In der Grundausstattung des Arduino-Boards ist dies ohne Erweiterung nicht möglich.



Abbildung 14: Arduino "Ethernet-Shield 2"



Abbildung 13: Rückseite "Ethernet-Shield 2"

Es basiert auf den „Wiznet W5500 Ethernet Chip“ dieser bietet ein Netzwerk (IP)-Stack an, der sowohl TCP als auch UDP beherrscht. Um dies nutzen zu können muss in der IDE, die dafür vorgesehene Bibliothek, eingebunden werden. Eine Verbindung zwischen Arduino und Shield erfolgt über die Pins auf der Rückseite vom Ethernet-Shield, wie in Abbildung 13 deutlich zu erkennen ist. Über den Standard „RJ45-Ethernet-Jack“ Anschluss auf dem Shield wird über ein gängiges CAT5-Kabel die Verbindung zum Netzwerk ermöglicht. Weitere technischen Eigenschaften beinhaltet das Shield in Form von einem Steckplatz für Micro-SD Kartenleser, Reset-Controller und 5 Status LED anzeige.

### 4.3 Slave

Unter dem Begriff „Slaves“ sind Bausteine mit diversen unterschiedlichen Eigenschaften zu verstehen, die verschiedene Anforderungen realisieren und diese über den Kommunikationskanal an den vorgesehenen Datenermittler übersenden. In der 1-Wire Technologie sind Bausteine zumeist als Mess- und Kontrolleinheiten zur Nutzung vorgesehen. Es gibt unzählige Anwendungsbereiche wie Temperatur-, Spannungs-, Strom- und Luftfeuchtigkeitmessung, Wandler, Counter und diverse Speichermöglichkeiten sind hier gegeben. Diese können als reine IC's oder als iButton genutzt werden. Abhängig des Einsatzbereichs sind Kompatibilitäten im Voraus zu berücksichtigen. [9]

Jedes Bauteil besitzt eine weltweit eindeutige 16-stellige Registrierungsnummer, die auch als ROM-Code bezeichnet wird, diese ist einmalig und unveränderbar. Als Beispiel nehmen wir den „DS2438 - Batteriemonitor“ mit der einmaligen Adresse: `2603EB67020000CB`. Die letzten beiden Ziffern beziehen sich auf den 1-Wire Familien-code, die Art Zugehörigkeit des jeweiligen 1-Wire Bausteins. Die 12 vorherigen Ziffern geben die Seriennummer und die ersten beiden enthalten eine Prüfsumme (CRC) die sich aus den restlichen 14 Ziffern berechnen lassen. [8]

Im Folgenden wird ein allgemeiner Aufbau der Registrierungsnummer dargestellt:



Abbildung 15: Aufbau ROM-Adresse 64 BIT

## 5 Inhalt

### 5.1 Batteriemonitoringsystem für mehrere Bleibatterie-Blöcke

Im Rahmen eines Praxisprojekts der TH Köln wird das 1-Wire BUS Systems als Fundament in einer wissenschaftlichen Arbeit genutzt, um ein System zu kreieren welches für die Überwachung und Speicherung signifikanten Daten von zwei Dutzend Energiespeicher dient.

Das Projekt PROGRESSUS zielt im deutschen Konsortium gemeinsam mit Partner aus Industrie und Forschung auf die Entwicklung von Elektroniksystemen und Leistungselektronik für eine effiziente Energienutzung ab. [19]

Hierbei werden im Forschungsprojekt der TH Köln im Institut „Cologne Institute for Renewable Energy“ unter anderem Energiespeicher für verschiedene Anwendungen und Ausnutzungen verwendet. In einer Laborumgebung wird zum Testen von Algorithmen ein intelligentes Energiemanagement simuliert und implementiert, die eine Kompatibilität vom Netzmanagement demonstriert. [20]

Diese Speicher simulieren die Versorgung der Anlage mit der benötigten Energie, die durch Auslastungen und wieder Aufladungen im ständigen Einsatz sind. Eine intelligente Überwachung, die die aktuellen Daten ausgibt und speichert wird benötigt. Für eine übersichtliche Ausgabe wird ein Batteriemonitoringsystem entwickelt, welches der eines BMS ähnelt. Für die Werte der einzelnen Batterien werden zur Datenübertragung Sensoren auf die Oberflächen der insgesamt 24 Batterieblöcke installiert. Dieser Sensor ist der IC „DS2438 - Smart Battery Monitor“, welcher in einem dafür vorgesehenen Schaltkreis (1-Wire Batterie-Monitor) implementiert ist.

Mit seinen zahlreichen Eigenschaften ermöglicht dieser eine zuverlässige und genaue Erkennung und Übermittlung von den ermittelten Messwerten. Mit Hilfe der Sensoren werden die Batteriespannungen und die Oberflächentemperaturen der Batterieblöcke gemessen und über das 1-Wire Bussystem an die jeweiligen BUS-Master übermittelt. Diese sammeln alle Daten und übermittelt diese über das lokale Netzwerk an den zugehörigen Überwachungspunkt der gesamten Anlage.

## 5.2 Aufbau der Anlage

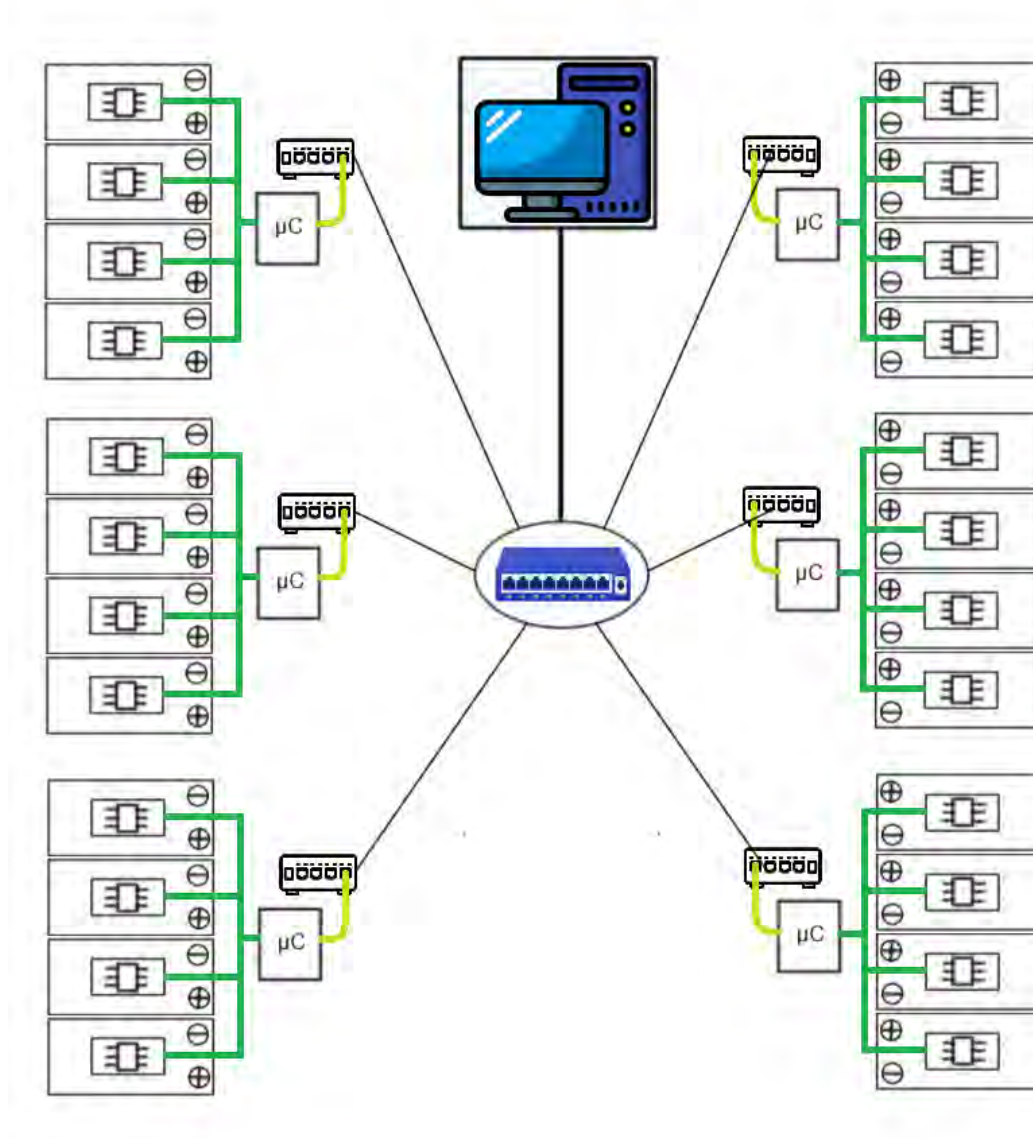


Abbildung 16: Aufbau der Anlage

Die 24 Bleibatterieblöcke werden in jeweils (6x4) Racks aufgeteilt, abhängig der Verschaltung der einzelnen Batterien zueinander können Spannungen oder Ströme erhöht oder gesenkt werden. Bei einer Betriebsspannung von ca. 12V liefern alle Batterien in Reihe geschaltet eine Höchstspannung von max. 288V. Auf jeder dieser Batterien befindet sich ein „1-Wire Batterie-Monitor“ Board, welches mit dem Sensor „DS2438“ versehen ist. Dieser sendet die erfassten Daten über den Datenbus an den BUS-Master (Arduino Uno R3). Um die Daten über das lokale Netzwerk senden zu können benötigt jeder Master eine TC/IP Schnittstelle. Mit der Erweiterungsplatine „Ethernet Shield 2“ vom



selben Hersteller sind die Hardwarebedingungen dafür gegeben. Mit der Anbindung an dem jeweiligen Switch die in jedem Rack einzeln verbaut sind, wird eine Kommunikation mit dem lokalen Netzwerk hergestellt.

### 5.2.1 „1-Wire Batterie-Monitor“ Board V3.0

Folgende Schaltung wurde für die Konstruktion der Batterieüberwachung von Prof. Dr. Eberhardt Waffenschmidt entwickelt und zum Zwecke dieser Arbeit genutzt. Es wurden nach Rücksprache mehrere Erweiterungen des Layouts vorgenommen, Rechte und Pflichten obliegen dem Urheber.

#### 5.2.1.1 Schaltungsschematik

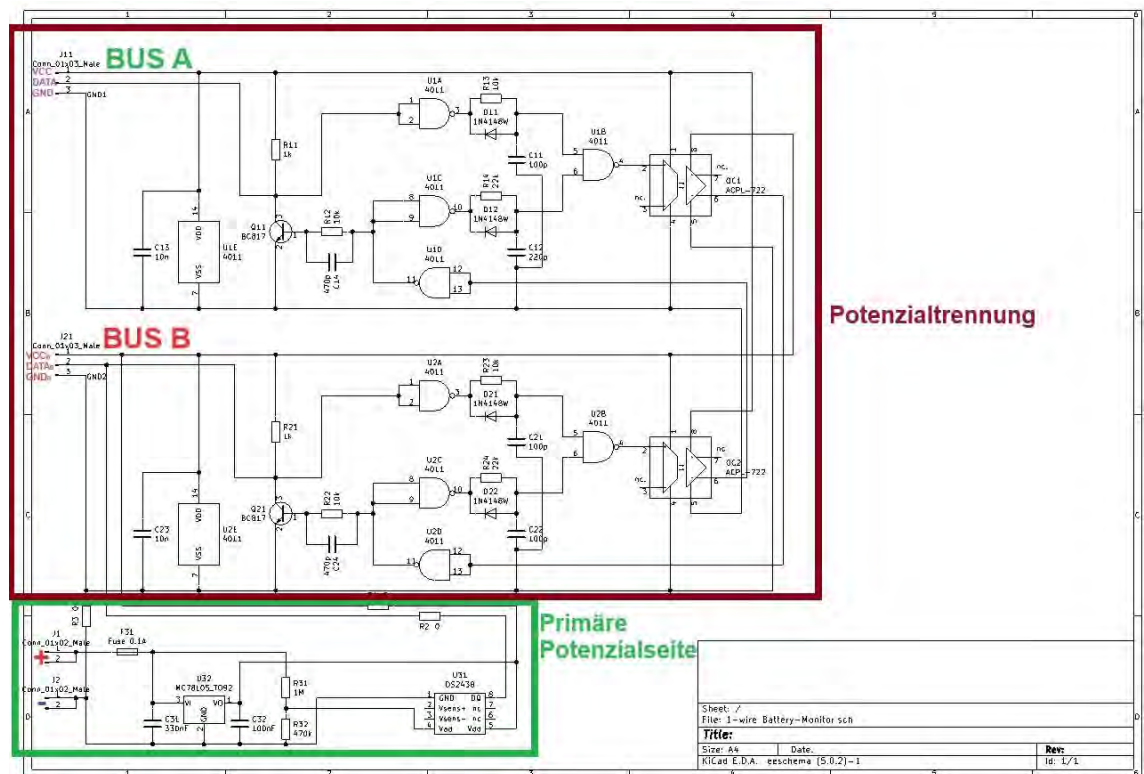


Abbildung 17: Schematik "1-Wire-Batterie-Monitor"

In der Abbildung 17 ist eine klare Unterteilung zwischen der primären Potenzielseite und der Potenzialtrennung zu erkennen. Im Folgenden werden beide Bereiche erklärt.

### Primäre Potenzielseite:

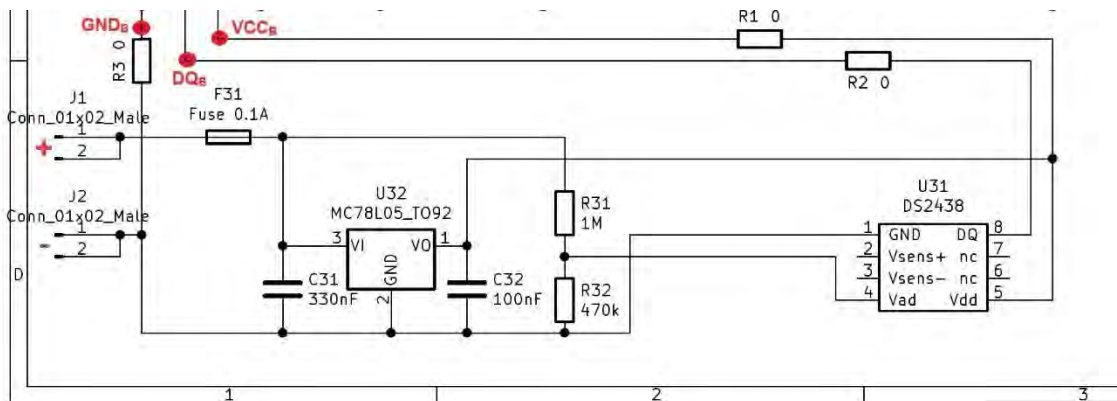


Abbildung 18: Primärer Potenzialanschluss Board

Es werden zwei Messwerte benötigt, aktuelle Batteriespannung und Temperatur der Batterieoberfläche. Der IC „DS2438“ hier in der Schematik mit dem Kürzel „U31“ dargestellt, dient als Sensor und misst beide Werte gleichzeitig. Der „U32“ ist ein linearer Spannungsregler und generiert eine Betriebsspannung von konstanten 5V ( $V_O=V_{CCB}=V_{dd}$ ) welches seine Energie aus der Batterie bezieht. Zum Schutz der Batterie vor Kurzschlüssen auf der Platine, sowie dem Schutz der Platine vor hohen Strömen von der Batterie auskommend. Wird eine Sicherung „F31“ direkt an die Klemme des positiven Batteriepoles installiert, um große Schäden an beiden Komponenten zu verhindern.

### Messung der Werte mit dem IC „DS2438“:

Die Temperatur wird mit Hilfe einer integrierten Technik welches die eigene Temperatur am IC-Gehäuse als Bezugswert nutzt, gemessen. Die gemessenen Werte werden in einem 13-Bit-Zweierkomplementformat bereitgestellt. Die Daten werden seriell über die 1-Wire-Schnittstelle an den Master gesendet. Der Messbereich liegt zwischen  $-55^{\circ}\text{C}$  bis  $+125^{\circ}\text{C}$  und wird in  $0,03125^{\circ}\text{C}$  Schritten gemessen. [21] Für die Temperaturmessung in der Praxis wird der IC im direkten Kontakt auf die Oberfläche des Batteriegehäuses mit einer kleinen Menge Wärmeleitpaste befestigt.

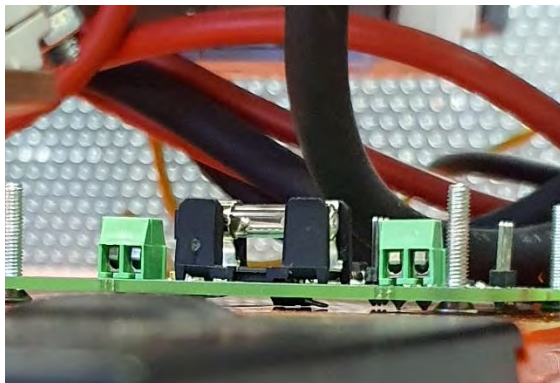


Abbildung 19: Rückseite der Platine mit dem IC DS2438 auf Batterie



Der integrierte Analog-Digitalwandler (A/D-C) hat eine Auflösung von 10 Bit und kann per Befehlsprotokoll (Convert V) eine Umwandlung der anliegenden Spannung durchführen, dieser Wert wird als Binärcode abgespeichert. Der Skalenbereich des (A/D) Wandlers ist zwischen 0V und 10V, die Versorgungsspannung am  $V_{dd}$  sollte hierbei nicht unter 2,4V liegen, da sonst ungenaue Messwerte die Folge haben. [21]

Die Spannungsmessung erfolgt über einen parallelen Abgriff der Batteriespannung über die jeweiligen Klemmen. Diese werden jeweils den aufliegenden Boards der einzelnen Batterien bei der Verbindung J1(+) & J2 (-) angelegt.

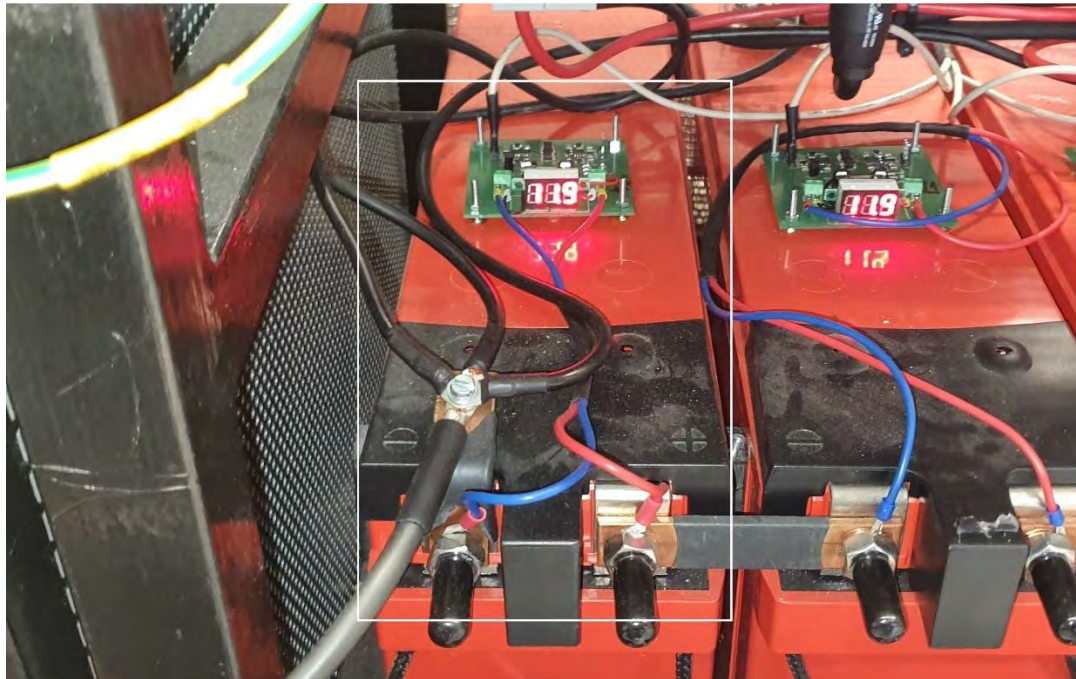


Abbildung 20: Batterie Spannungsmessung

Der aktuelle Spannungswert der Batterie wird über einen Spannungsteiler ( $R_{31}$  &  $R_{32}$ ), an den PIN „ $V_{AD}$ “ (General A/D input) angelegt. Da dieser über der maximalen von Spannung max.  $10V_{AD}$  liegt wird eine Teilung benötigt. Folgende Formel beschreibt den Faktor des Teilers:

$$U_{V_{DA}} = \frac{R_{32}}{R_{31} + R_{32}} * U_{BAT} \triangleq U_{V_{DA}} = \frac{470k\Omega}{470k\Omega + 1M\Omega} * U_{BAT} \triangleq \sim 0,3197 * U_{BAT}$$

Dieser Faktor muss später im Programmverlauf verrechnet werden, um die tatsächliche anliegende Spannung zu ermitteln.

**Potenzialtrennung:**

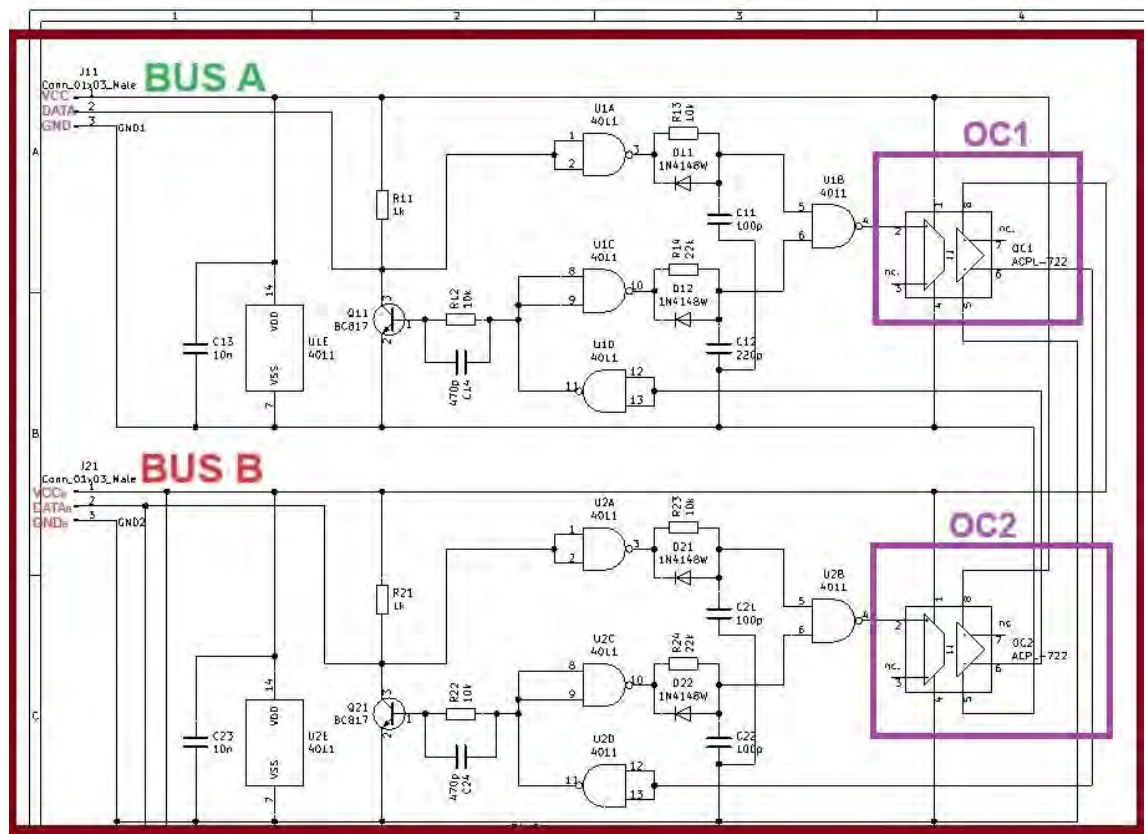


Abbildung 21: Potenzialtrennung 1-Wire-Batterie-Monitor

Über den 1 Wire-Bus werden Signale versendet, indem die angeschlossenen Geräte über ihre Open-Kollektor Ausgängen den BUS auf GND ziehen. Dieses Verhalten muss nach der Potenzialtrennung fortgesetzt werden, daher Bedarf es an einer angepassten Schaltung. Um die Funktion zu verstehen, betrachtet man wie in Abbildung 20 aufgeführt, zwei vollständige Busse (BUS A und BUS B) welche voneinander über die beiden Optokoppler (OC1 & OC2) gekoppelt sind. Ein Optokoppler hat die Eigenschaft das es Signale, über zwei voneinander galvanisch getrennten Stromkreisen, überträgt.

Im Folgenden wird eine vereinfachte Darstellung des Grundgedankens veranschaulicht:

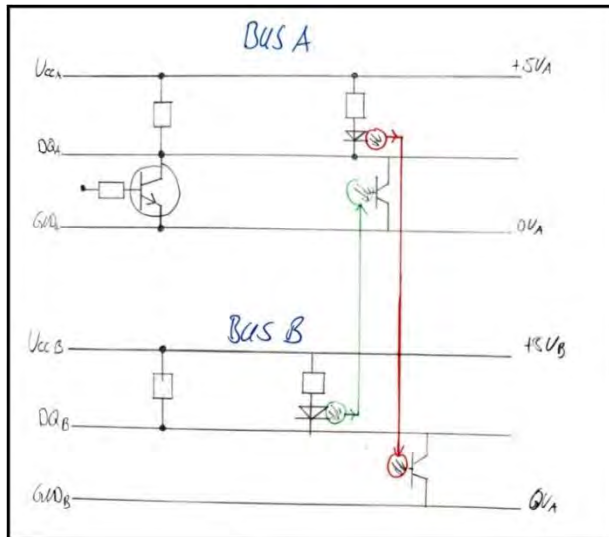


Abbildung 22: BUS Aufteilung Skizze

Diese einfache Form wird durch mehrere Faktoren nicht realisierbar sein. Unter anderem entstehen Zustände und Fehler, die nicht gewünscht sind (festgefahrener Zustand (**Death-Lock**), keine ausreichende **Versorgung** über den BUS, **Timing Probleme** etc). Für die Behebung dieser Fehler wurde für ersteres eine Logik für die Verriegelung konzipiert und dazugeschaltet, anhand dieser löst sich der „Death-Lock“ auf. Für die mangelnde Versorgung über den Datenbus werden Transistoren für die Verstärkung des LED-Stroms in den Optokopplern dazugeschaltet. Für letzteres werden Verzögerungsglieder bzw. ein Beschleunigungsglieder eingebracht.

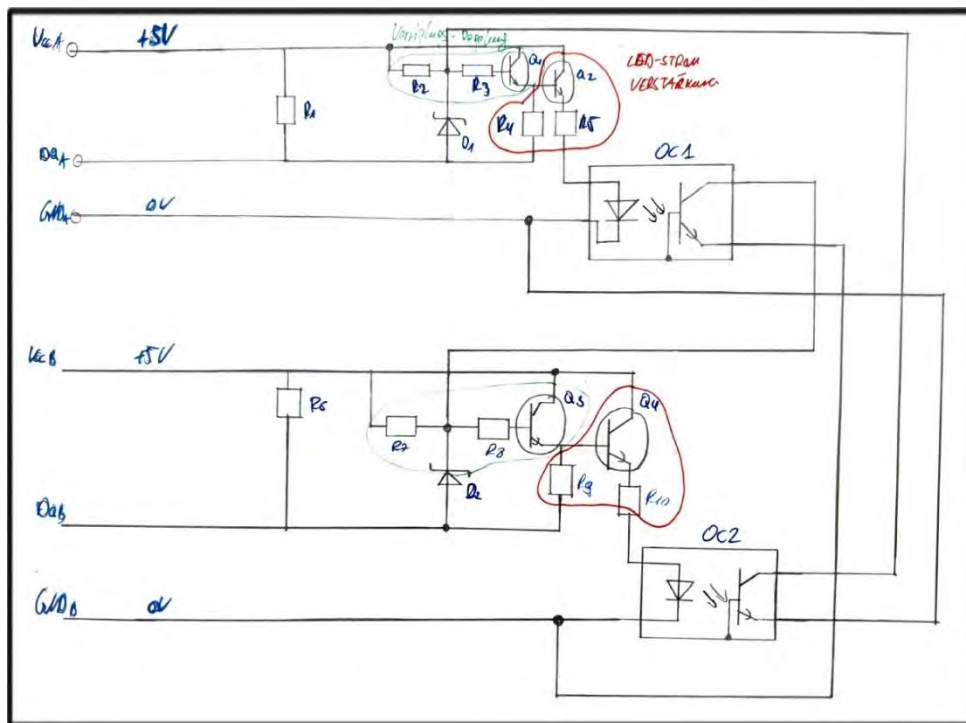


Abbildung 23: Lösung für Death Lock & Verstärkung des LED-Stroms im OC1& OC2

Nach Erweiterung mit der Logik per CMOS-NAND-Gattern und den bereits erwähnten Verzögerungsgliedern (RCD:  $C_{11}$ ,  $R_{13}$  und  $D_{11}$  sowie  $C_{12}$ ,  $R_{14}$  und  $D_{12}$ ) und dem Beschleunigungsglied (RC:  $C_{14}$  &  $R_{12}$ ), für BUS B entsprechend dieselben Bedingungen, erhalten wir die fertige Schaltung wie sie in Abbildung 20 zu sehen ist. Mit dieser Schematik kann nun das Layout realisiert werden.

### 5.2.1.2 Layout

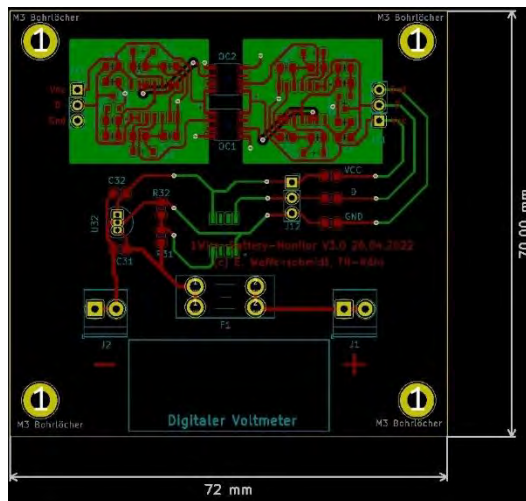


Abbildung 24: Footprint 1-Wire-Batterie-Monitor

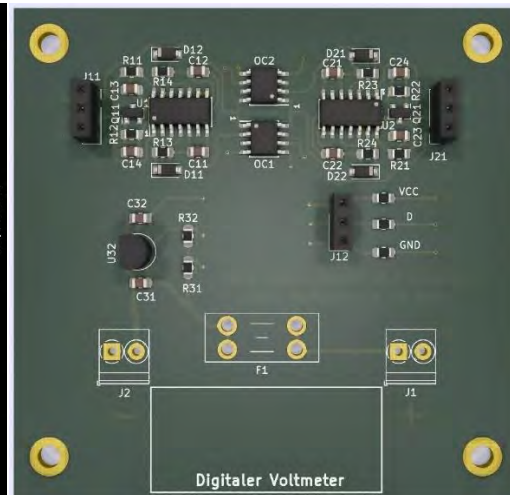


Abbildung 25: 3-D Ansicht 1-Wire-Batterie-Monitor ohne F1, J1 & J2

Nach der Erweiterung des Layouts mit dem digitalen Voltmeter, dem Abgreifpunkt bei  $J_{12}$  und anderen Veränderungen ist die aktuelle Version 3.0 für die Bestellung und Fertigung bereit.

#### Bestellung und Bestückung der Bauteile:

Nach erfolgreicher Bestellung der Bauteile und der Leiterplatten wird der PCB-Bestückungsprozess gestartet. Für eine industrielle Fertigung gibt es Unternehmen, die mit Kleinserien beauftragt werden können. Bei 24 Platinen wurde dieser Prozess ausgelassen, da es sich hierbei um eine vergleichsweise geringe Menge handelt und der preisliche Aufwand sich nicht lohnt. Für einen sauberen und genauen Arbeitsprozess wurde der sogenannte „Dispenser“ für die Lötpasten Verteilung auf den fertigen Rohlingen genutzt. Nach genauer Zuordnung der Bauteile auf der Leiterplatte wird der Prozess des SMD-Lötens im Lötöfen beendet. Nach weiterer Montage des linearen Spannungsreglers „U32“, der beiden Leiterplattenklemmen ( $J1$  &  $J2$ ), der Sicherung ( $F1$ ) und der 3x3 poligen Stiflleisten per Hand ist die Platine bereit zur weiteren Nutzung.



### Prüfung der Funktionalität:

Bevor die Boards auf die Batterien befestigt werden können, müssen diese auf Funktionalität geprüft werden. Die Primäre Potenzielseite wird mit der Batteriespannung an den Leiterplattenklemmen (J1 & J2) verbunden. Am mittleren PIN der Stiftleiste J<sub>12</sub> wird mit dem BUS-Master oder einem 1-Wire Konverter ein Abgriff zum verbauten Sensor getätigt. Nach erfolgreicher Verbindung können Daten wie ROM-Adresse, Messwerte der Batteriespannung & Temperatur mit entsprechender Software ausgegeben werden. Der zweite Test auf Funktionalität erfolgt über die Anschlüsse DQ<sub>A</sub> & DQ<sub>B</sub>. Hier wird ein Rechtecksignal angelegt und die Verzögerung zwischen beiden Punkten mit dem Oszilloskop aufgenommen und interpretiert.

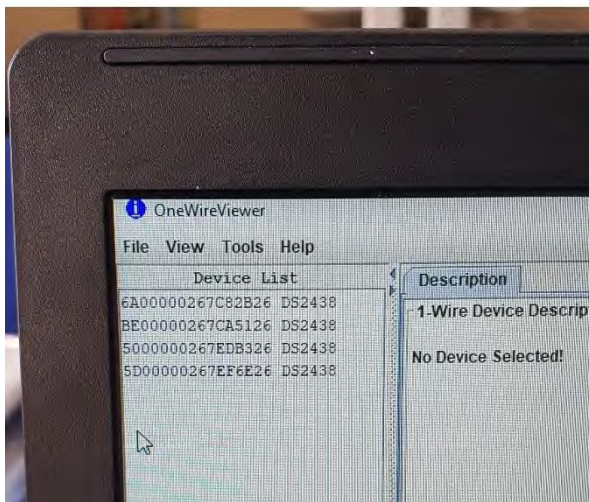


Abbildung 26: 1-Wire Viewer mit 4 Sensoren

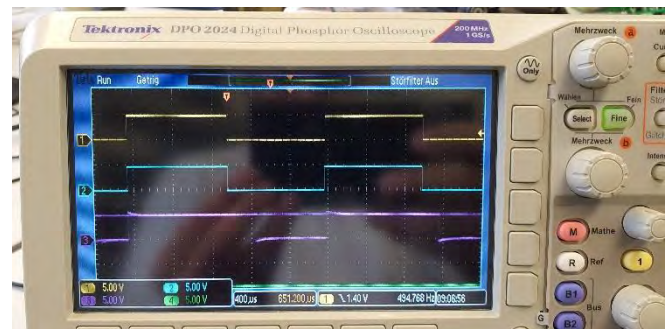


Abbildung 27: Rechteckspannung mit Delay

In Abbildung 26 ist zu erkennen das 4 Sensoren gleichzeitig vom Typ „DS2438“ über den One-Wire-Viewer erkannt und aufgelistet sind, folglich kann man die einzelnen ROM-Adressen entnehmen, diese werden von rechts nach links abgelesen. Bei der Auswahl eines der Sensoren über den One-Wire-Viewer können spezifische Messwerte getestet und ausgegeben werden. In Abbildung 26 ist das Eingangssignal am Punkt DQ<sub>A</sub> angelegt es hat die Farbe Geld, im Vergleich dazu wird das blaue Signal parallel am Punkt DQ<sub>B</sub> aufgenommen. Es ist eine Verzögerung (Delay) von knapp **0,8µs** zu erkennen. Dieser ist wegen den Gatterlaufzeiten vorhanden und kann nicht verhindert werden.

### Installation auf dem Batteriegehäuse:

Zur Befestigung der Platine auf dem Batteriegehäuse wird ein Zweikomponentenkleber verwendet. Die Köpfe der M3-Schrauben, welche die Platine an allen vier Ecken mit Muttern befestigt, wurden in den Kleber getaucht und schließlich auf das Gehäuse der Bleibatterie geklebt.

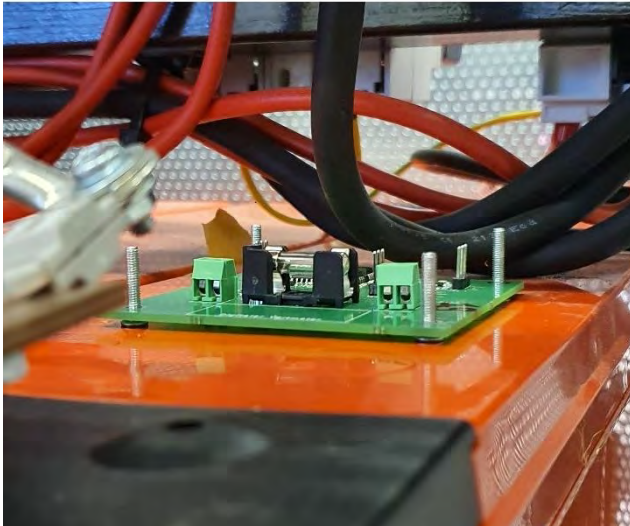


Abbildung 28: Platine auf Batteriegehäuse

Schließlich werden wie in Abbildung 29 zu erkennen ist, alle vier Platinen im jeweiligen Rack mit dem  $\mu\text{C}$  (Master) verbunden. Um dies bewerkstelligen zu können wird eine Art Sammelschiene für alle ausgehenden Leitungen (VCC, DQ und GND) konstruiert. Die Datenleitung führt zum digitalen Anschluss PIN 2 vom Arduino, dieser wird in diesem Vorhaben als 1-Wire-BUS fest definiert und genutzt. VCC und GND werden jeweils an die analogen Anschlüsse 5V und GND vom  $\mu\text{C}$  geführt.



Abbildung 29: 4 Platinen am  $\mu\text{C}$  im Rack

### 5.2.2 Ansteuerung über die IDE

Für eine Kommunikation zwischen dem Arduino und den jeweiligen Platinen ist es notwendig passende Elemente und Befehle im Sketch zu hinterlegen. Zunächst werden die Bibliotheken des 1-Wire Bussystems, des Sensors „DS2438“ selbst und der allgemeinen Bibliothek für Sensoren, die auf der 1-Wire Basis arbeiten, eingebettet. Nach dem die ROM-Adressen wie in Abbildung 26 ermittelt wurden, müssen diese im Programm fest hinterlegt sein, um eine genau Ansteuerung und Zuweisung zu gewährleisten. Mit

vordefinierten Befehlen wird so die ROM-Adresse und die Werte für Batteriespannung und Temperatur der einzelnen Sensoren abgefragt. Das Arduino bietet einen seriellen Monitor zur Veranschaulichung der Daten an. Dies kann verwendet werden, um Daten über die Konsole ausgeben zu lassen. Nach dem alle Anforderungen und Bedingungen gesetzt sind läuft die Abfrage der Messwerte in einem zeitlich festvorgegebenen Intervall ab. [22]

#### Kommunikation über das lokale Netzwerk:

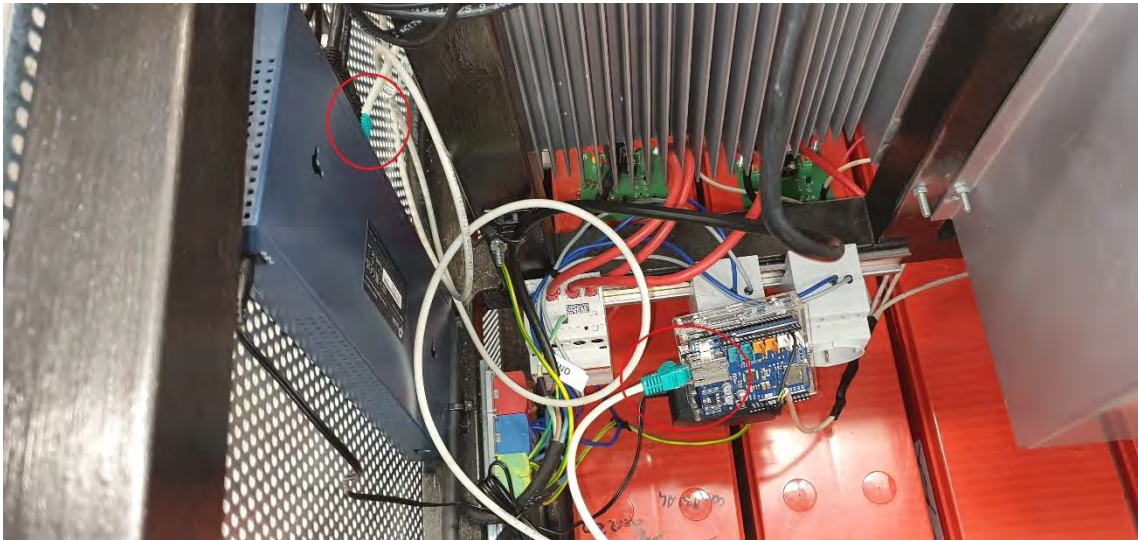


Abbildung 30: Verbindung zwischen uC und Switch

Um einen Zugriff auf die Daten des Sensors „DS2438“ über das lokale Netzwerk zu erhalten, kann das Ethernet Shield als Webserver konfiguriert werden. Nach erfolgreicher Ermittlung der Daten vom Master, der Verbindung des Shields mit dem Arduino und der Verbindung des Shields mit dem heimischen Router sind die ersten Bedingungen für die Abfrage erfüllt. Wie im vorherigen Sketch müssen Bibliotheken, Befehle und Funktionen eingebracht werden. [23]

Im selben Sketch (sketch\_dec29a) welches zur Ermittlung der Daten vom Sensor genutzt wird kann die Befehlsanordnung für das Shield mitaufgenommen werden. Schließlich wird über dem Ethernet Shield ein Server aufgespielt, über den Webbrowser konnten die ersten Daten bereits entnommen werden.



### 5.3 Interpretation

Nach den ersten fertiggestellten Platinen konnte eine erfolgreiche Verbindung mit dem Sensor erfolgen. Die ersten Daten wurden vom 1-Wire Viewer ausgegeben und analysiert. Beim Testen der Richtigkeit der gemessenen Werte wurde eine variable Spannungsquelle angeschlossen. Per direktem Fingerkontakt wurde der Sensor erwärmt gleichzeitig die Spannung erhöht und gesenkt, die steigenden bzw. fallenden Messwerte für Temperatur und Spannung wurden dabei beobachtet. Beide Werte sanken und fielen mit dem Einfluss von außen. Daher konnte die Messung als erfolgreich eingestuft werden und die Funktionalität für weitere Vorhaben war gegeben. Um die Beschaffenheit der Anlage zu testen, wurde das Konstrukt bis an seine Grenzen erweitert. Diese waren, abhängig vom Testpunkt, unterschiedlich. Das Abrufen der Daten kann parallel verlaufen. Daraufhin wurden so viele Platinen wie möglich zum Testen zusammengeschaltet. Bei Messungen im Abgreifpunkt  $J_{12}$  (ohne Kontakt zu  $DQ_B$  (BUS B)) der einzelnen Platinen gleichzeitig wurden die Daten von bis zu 15 Platinen zeitgleich empfangen. Ein Zugriff der Daten über die Potenzialtrennung erwies sich als weitaus problematischer. Die maximal zulässig verschaltete Anzahl an Platinen wurde auf 1-2 Stück pro Rack begrenzt. Nähere Informationen könnten vom Datenblatt des Sensors entnommen werden, darin heißt es das das ROM-Funktionsprotokoll für eine erfolgreiche Funktionalität Voraussetzung ist, welche jederzeit im Aufbau gegeben ist. Nach ausführlicher Analyse wird der Fehlerbereich auf die Potenzialtrennung begrenzt. Die Vermutung ist, dass die Fehlerquelle im Timing liegt. Daher wurden die Signale von den Optokopplern und den Verzögerungs- bzw. Beschleunigungsglieder ausgehend analysiert, diese stehen so wie es sein soll in Relation zueinander. Weitere Signal wie die der Busse A & B wurden auf ihre Verläufe und Beschaffenheit analysiert, es wurden keine signifikanten Abweichungen oder Unregelmäßigkeiten entdeckt.

Um die Fehlerquelle zu minimieren oder gar zu kompensieren, wird zur Sicherstellung die Kapazität beim Kondensator  $C_{12}$  verdoppelt. Dies hat den Grund das die Dimensionierung der Verzögerung (RCD- Filter  $R_{12}$ ,  $D_{12}$  und  $C_{12}$ ) zu klein gewählt ist und durch eine Erweiterung ergänzt werden muss. Dies bringt eine kleine Verbesserung mit sich, sodass sich die Menge der maximal angeschlossenen Platinen auf 2-3 Stück erhöht. Beim fortschreitenden Aufbau wird eine weitere Hürde bemerkt.

Abhängig der Zusammenschaltung des Konstrukts ist eine weitere Ungenauigkeit zu bemerken. Im Aufbau per Steckereinheit funktionierten Platinen zusammen, die wiederrum am Rack selbst mit dem dafür angefertigten Kabelbaum nicht mehr funktionieren.



Um die Ungenauigkeit der Problematik eingrenzen zu können werden die Reset-Pulsbreiten (RPB) von allen Platinen einzeln mit dem Oszilloskop gemessen und in folgender Abbildung dargestellt.

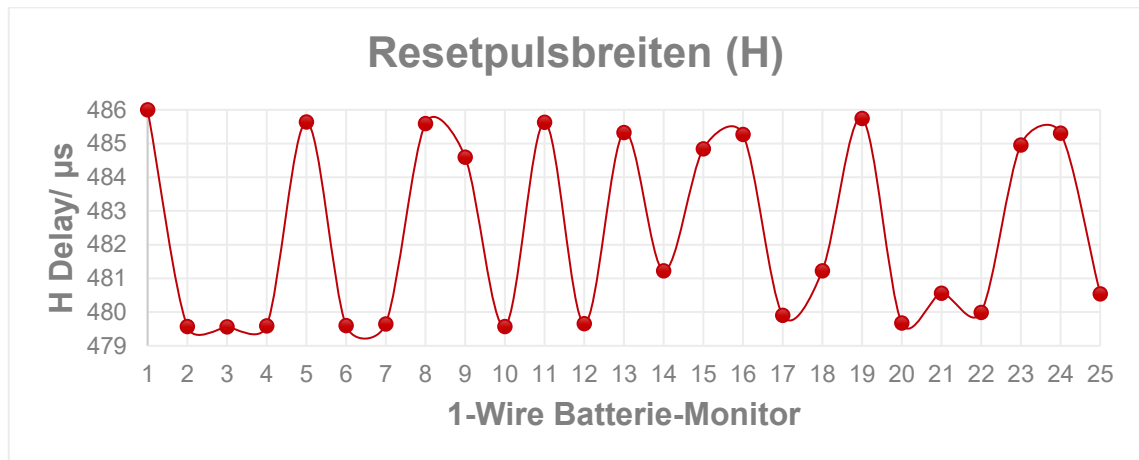


Abbildung 31: Aufnahme aller RPB (H) // Genaue Werte in der Excelltabelle

In der Tabelle 2 ist in der Zeile Delay (H) mit 480µs ein Standwert für die RPB hinterlegt. Alle Platinen mit max. +6µs (Platine 1) und min. -0,437µs (Platine 3), erfüllen die Angabe, die Abweichungen sind im Toleranzbereich von 0,8%. Um vier „1-Wire-Batterie-Monitore“ zusammen in einem Rack unterzubekommen werden Platinen ausgesucht die zueinander naheliegende RPB's aufweisen. Folgende Anordnungen ergeben sich aus der Zusammensetzung:

Tabelle 4: Verteilung der Platinen in den jeweiligen Racks

Rack:	1	2	3	4
1-Wire-Batterie-Monitor	3	1	11	17
	7	9	15	20
	12	19	16	21
	25	23	24	22

Es ist zu erkennen, dass die Anordnung nur 4 Racks beinhaltet, für Rack 5 & 6 müssen weiteren Energiespeicher, die dieser Anforderungen gerecht sind, hinzugefügt werden. Des Weiteren haben viele Boards in der Verteilung der Racks keine Zugehörigkeit erhalten. Trotz der naheliegenden RPB können die restlichen Platinen in keiner der gängigen Konstellation mit anderen Platinen zusammengeführt werden. Bei der Fehleranalyse ist folgendes Signal im Oszilloskop aufgenommen wurden.

### Fehlerabbildung Oszilloskop:

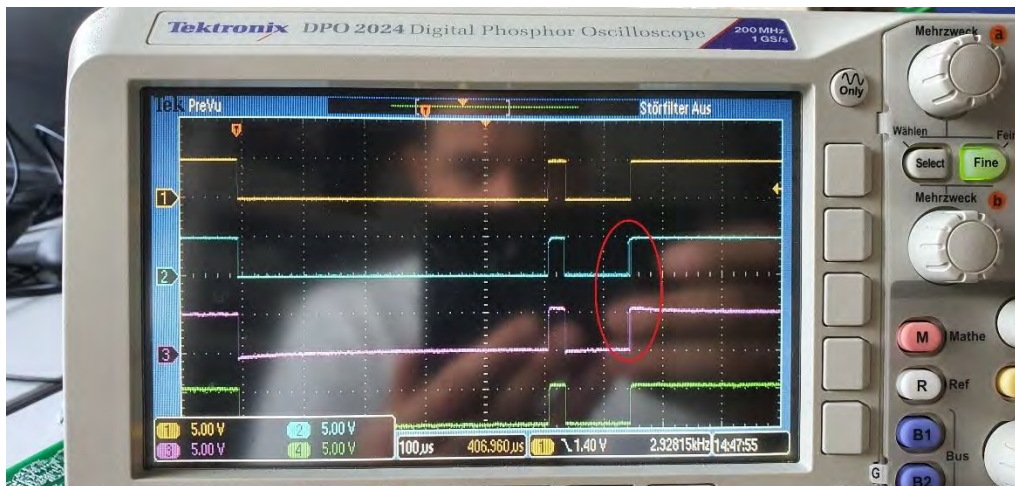


Abbildung 32: Fehleraufnahme RPB per Oszilloskop

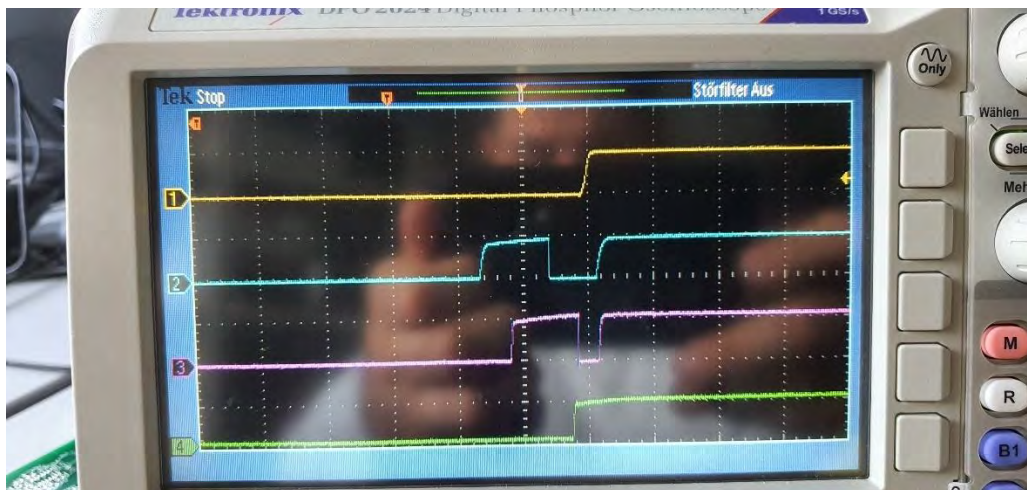


Abbildung 33: Fehleraufnahme RPB-Nahaufnahme

In beiden Abbildungen sind jeweils 4 Platinen in einem Rack angeschlossen und werden auf Funktionalität getestet. Board Blau & Board Lila weisen bei der Momentaufnahme eine Fehlfunktion auf. Nach genauer Untersuchung der beiden Signale kann in Abbildung 30 eine untypische Reaktion auf die Anwesenheitsabfrage des Masters, von beiden Sensoren auskommend, entnommen werden. Nach mehreren Messungen und Anpassungen der Bauteile, die mit anderen Platinen identisch sind, konnte der Fehler nicht eingeschränkt werden. Jedoch fällt auf, dass die neu angefertigten Platinen einen geringeren Fehleranfälligkeit aufweisen, obwohl keine Änderungen der Bauteile oder des Anfertigungsverfahren vorgenommen wurden. Somit werden die alten Platinen wegen Ihrer Ungenauigkeit und Ihrer Fehleranfälligkeit aussortiert. Platine 19 bis 24 werden nach dieser Erkenntnis für die Vollständigkeit in den jeweiligen Racks montiert. Alle vorhandenen Batterieblöcke haben somit eine funktionierende Spannungs- und Temperaturmessung, welche Ihre gemessenen Werte über den 1-Wire Bus ans lokale Netzwerk senden.

## 6 Fazit

Für eine Steigerung der Sicherheit sind intelligente Systeme zur Überwachung, Ansteuerung und Vorkehrungen unerlässlich. Diese schützen Anlagen und bilden eine zuverlässige Alternative, um Schäden und Ausfälle durch verursachte Fehlfunktionen vorzubeugen.

Um eine Form diese Überwachung zu verdeutlichen, wird in diesem Praxisprojekt die Möglichkeit mit Hilfe des 1-Wire Bussystems, die als Plattform für eine intelligente Überwachung genutzt wird, in Verbindung gebracht. Für grundlegendes Verständnis wird zunächst das 1-Wire im gesamten Betrachtet und im späteren Verlauf spezifiziert. Dies ist nötig, um die Vielseitigkeit dieses Systems zu kennen, damit gegebenenfalls weitere individuelle Anforderungen einen Nutzen daraus ziehen können. In diesem Fall wird eine Auslegung für ein laufenden Projekt als Ergänzung für Ihre Komponenten konzipiert.

Die wichtigsten Meilensteine dieser Arbeit beinhalten zunächst die Herstellung der Komponenten. Diese konnten Anhand von Hilfsmittel und Gütern erfolgreich produziert werden. Im Verlauf des Projekts wurde nach Verbesserung der Konstruktion eine Automatisierung für die Herstellung eingeführt, um eine konstante hohe Qualität zu gewährleisten. Dies hat den Vorteil, dass die Komponenten einen identischen Bezug zum gesamten Aufbau haben und später in der Verteilung eine bessere Übersicht ermöglichen. Die Zuteilung der Komponenten miteinander erweist sich im späteren Verlauf als deutlich einfacher. Des Weiteren wurde die Kommunikation der Bauteile mit einem zentralen Netzwerk verbunden, um eine Gesamtübersicht der individuellen Anlage zu ermöglichen.

Diese Arbeit zeigt zudem die Wirksamkeit des 1-Wire-Bussystems als Basis für ein Batterimonitorsystem, das die Überwachung von mehreren Bleibatterieblöcken ermöglicht. Durch die erfolgreiche Implementierung der Platinen auf die 16 Energiespeicher wird die Annahme, dass dieses System zu einer signifikanten Verbesserung der Batteriemangementeffizienz beitragen kann, bestätigt. Diese Verbesserung kommt der Optimierung der Batterielebensdauer und die Sicherheitssteigerung der gesamten Anlage zugute. Die Fähigkeit des 1-Wire-Bussystems, eine robuste und zuverlässige Kommunikation zwischen dem Master und den Sensoren zu gewährleisten, stellt einen signifikanten Vorteil gegenüber anderen Kommunikationssystemen dar. Als Resultat dieser Arbeit liegt es nahe, dass weitere Forschungen und Analysen im Bereich der Fehlereindämmung erfolgen müssen. Diese Fehler, die zu Erkennung und Zuweisung einzelner Bauteile eine Ungenauigkeit mit sich führt, müssen im weiteren Verlauf analysiert und bestenfalls beseitigt werden. Dies ist von Nöten um eine zukunftsweisende Lösung für gegeben falls größere Anwendungsbereiche zu ermöglichen. Zudem kann durch die

wirtschaftliche als auch ökologischen Vorteile des 1-Wire-Bussystems ein kostengünstiges und zuverlässiges System zur Überwachung realisiert werden.

Abschließend sei betont, dass die Ergebnisse dieser Arbeit als Grundlage für zukünftige Forschung und weiteren Entwicklungen in diesem Bereich dienen können, mit dem Ziel, die Funktionalität des 1-Wire-Bussystems als Grundlage weiter zu verbessern und auf breitere Anwendungsbereiche zu erweitern.

## 7 Verzeichnisse und Anhang

### Tabellenverzeichnis

Tabelle 1: 1-Wire Operationen .....	12
Tabelle 2: 1-Wire Master Timing.....	13
Tabelle 3: Befehlssequenz "Smart Batterie Monitor" .....	14
Tabelle 4: Verteilung der Platinen in den jeweiligen Racks.....	30

## Abbildungsverzeichnis

Abbildung 1: Native 1-Wire Bausteine mit Linearer Topologie Ausführung (keine GND-Ader).....	5
Abbildung 2: BUS-Master als Host und als Spannungsversorger .....	5
Abbildung 3: Parasitäre Versorgung.....	6
Abbildung 4: Topologie Linie .....	7
Abbildung 5: Topologie Linie mit Abzweig .....	7
Abbildung 6: Topologie Stern .....	8
Abbildung 7: Topologie Ring .....	8
Abbildung 8: Kommunikationstechnik.....	10
Abbildung 9: Hardware Konfiguration Datenblatt "DS2438" .....	11
Abbildung 10: 1-Wire Verzögerungszeiten .....	12
Abbildung 11: Verschiedene Entwicklungsboards .....	15
Abbildung 12: Arduino Uno R3 mit Schutzkäfing .....	16
Abbildung 14: Arduino "Ethernet-Shield 2" .....	16
Abbildung 13: Rückseite "Ethernet-Shield 2" .....	16
Abbildung 15: Aufbau ROM-Adresse 64 BIT .....	17
Abbildung 16: Aufbau der Anlage.....	19
Abbildung 17: Schematik "1-Wire-Batterie-Monitor".....	20
Abbildung 18: Primärer Potenzialanschluss Board .....	21
Abbildung 19: Rückseite der Platine mit dem IC DS2438 auf Batterie .....	21
Abbildung 20: Batterie Spannungsmessung.....	22
Abbildung 21: Potenzialtrennung 1-Wire-Batterie-Monitor .....	23
Abbildung 22: BUS Aufteilung Skizze.....	24
Abbildung 23: Lösung für Death Lock & Verstärkung des LED-Stroms im OC1& OC2 .....	24
Abbildung 24: Footprint 1-Wire-Batterie-Monitor.....	25
Abbildung 25: 3-D Ansicht 1-Wire-Batterie-Monitor ohne F1, J1 & J2.....	25
Abbildung 26: 1-Wire Viewer mit 4 Sensoren .....	26
Abbildung 27: Rechteckspannung mit Delay .....	26
Abbildung 28: Platine auf Batteriegehäuse.....	27
Abbildung 29: 4 Platinen am $\mu$ C im Rack .....	27

Abbildung 30: Verbindung zwischen uC und Switch .....	28
Abbildung 31: Aufnahme aller RPB (H) // Genaue Werte in der Exceltabelle.....	30
Abbildung 32: Fehleraufzeichnung RPB per Oszilloskop .....	31
Abbildung 33: Fehleraufzeichnung RPB-Nahaufnahme.....	31

## Literaturverzeichnis

- [1 K. Dembowski, „elektroniknet.de,“ 5 Mai 2021. [Online]. Available: ] <https://www.elektroniknet.de/embedded/hardware/unterschaetzte-one-wire-schnittstelle.186474.html#:~:text=Der%20One%2DWire%2DBus%20%5B,Firma%20Maxim%20Integrated%20aufgegangen%20ist.> [Zugriff am 10. August 2022].
- [2 Moritz Fuchs Elektronik, „ www.mfelabs.com,“ [Online]. Available: ] <https://www.mfelabs.com/de/info/onewire/>. [Zugriff am 10. Oktober 2022].
- [3 E. Kompendium, „Elektronik-Kompendium.de,“ [Online]. Available: ] <https://www.elektronik-kompendium.de/sites/raspberry-pi/2703071.htm>. [Zugriff am 2. Mai 2023].
- [4 K. Dembowski, „elektroniknet.com,“ 26. Mai 2021. [Online]. Available: ] <https://www.elektroniknet.de/embedded/hardware/unterschaetzte-one-wire-schnittstelle.186474.html>. [Zugriff am 10. Mai 2023].
- [5 s. V. Wikipedia-Autoren, „Wikipedia – Die freie Enzyklopädie,“ 19 April 2023. ] [Online]. Available: <https://de.wikipedia.org/wiki/Bidirektional>. [Zugriff am 1. Mai 2023].
- [6 Daniel Kampert , „Kampis Elektroecke,“ 26. November 2019. [Online]. Available: ] <https://www.kampis-elektroecke.de/2019/11/1-wire-fuer-den-avr/>. [Zugriff am 29. Juli 2023].
- [7 Elaborated Networks GmbH, „<https://shop.elabnet.de/gtc>,“ 1. Juli 2020. [Online]. ] Available: <https://shop.elabnet.de/info/1-wire>. [Zugriff am 10. Oktober 2022].
- [8 Moritz Fuchs Elektronik, „Labs MFE,“ Moritz Fuchs Elektronik, [Online]. Available: ] <https://www.mfelabs.com/de/info/onewire/#:~:text=Master%20f%C3%BCr%20Ether%20net-%20Anwendungen,alles%20digitale%20Umweltsensoren%20gro%C3%9F%20Beliebtheit.> [Zugriff am 15. Februar 2023].
- [9 Martin Fischer , „Fischer-Net,“ 11. April 2012. [Online]. Available: <https://www.fischer-net.de/hausautomation/haustechnik/1-wire/35-1-wire-bus-grundlagen.html>. [Zugriff am 30. Juli 2023].
- [1 Maxim Integrated , „Maxim Integrated,“ 19. April 2018. [Online]. Available: ] <https://www.stg-maximintegrated.com/en/products/ibutton-one-wire/one-wire.html/tab0#popupmodal>. [Zugriff am 31. Juli 2023].
- [1 Esera GmbH, „esera,“ [Online]. Available: <https://esera.de/Service-Support/1-Wire-1-Grundlagen/1-Wire-Topologie/>. [Zugriff am 01. August 2023].



[1 Wikipedia, „Wikipedia,“ 19. April 2023. [Online]. Available:  
2] <https://de.wikipedia.org/wiki/Bidirektional>. [Zugriff am 1. August 2023].

[1 Wikipedia-Autoren, „Wikipedia-Die freie Enzyklopädie,“ 6. März 2023. [Online].  
3] Available:  
[https://de.wikipedia.org/w/index.php?title=Duplex\\_\(Nachrichtentechnik\)&oldid=231536746](https://de.wikipedia.org/w/index.php?title=Duplex_(Nachrichtentechnik)&oldid=231536746). [Zugriff am 30. August 2023].

[1 J. Hohn, „Placetel,“ BroadSoft Germany GmbH, [Online]. Available:  
4] <https://www.placetel.de/ratgeber/vollduplex>. [Zugriff am 30. August 2023].

[1 Reichelt elektronik, „reichelt elektronik- The best part of your projekt,“ [Online].  
5] Available: <https://www.reichelt.de/raspberry-pi-arduino-c8241.html#:~:text=Moderne%20Entwicklerboards%20sind%20die%20leistungsfo%20C3%A4higen,aus%20einfachen%20Komponenten%20zusammensetzen%20lassen>  
.. [Zugriff am 4. September 2023].

[1 Wikipedia, „Wikipedia,“ [Online]. Available:  
6] [https://de.wikipedia.org/wiki/Arduino\\_\(Plattform\)](https://de.wikipedia.org/wiki/Arduino_(Plattform)). [Zugriff am 29. März 2023].

[1 Wikipedia-Autoren, „Wikipedia – Die freie Enzyklopädie,“ 23. August 2023. [Online].  
7] Available: [https://de.wikipedia.org/wiki/Arduino\\_\(Plattform\)](https://de.wikipedia.org/wiki/Arduino_(Plattform)). [Zugriff am 04. September 2023].

[1 L. Ngole, „Maker Wiki,“ Ruhr-Universität Bochum, 11. November 2020. [Online].  
8] Available: <https://el.rub.de/wiki/MakerWiki/index.php/Arduino>. [Zugriff am 05. September 2023].

[1 B. f. B. u. Forschung, „Elektronikforschung,“ VDI/VDE Innovation + Technik GmbH,  
9] 1. Juni 2022. [Online]. Available:  
<https://www.elektronikforschung.de/projekte/progressus>. [Zugriff am 24. September 2023].

[2 T. K. I. C. I. f. R. Energy, „TH Köln PROGRESSUS,“ 1. März 2023. [Online]. Available:  
0] [https://www.th-koeln.de/anlagen-energie-und-maschinensysteme/progressus\\_76058.php](https://www.th-koeln.de/anlagen-energie-und-maschinensysteme/progressus_76058.php). [Zugriff am 23. September 2023].

[2 M. Integrated, „Mouser electronics,“ 2005. [Online]. Available:  
1] <https://www.mouser.de/ProductDetail/Analog-Devices-Maxim-Integrated/DS2438Z+?qs=0Y9aZN%252BMVCXYzd1djJeYaw%3D%3D>. [Zugriff am 05. Oktober 2023].

[2 G. Akinci, „sketch\_dec29a,“ CD, 2023.  
2]

[2 G. Akinci, „sketch\_may29a,“ CD, 2023.  
3]

[2 A. Bohne-Lang, „Linus-Magazin,“ 1. März 2010. [Online]. Available:  
4] <https://www.linux-magazin.de/ausgaben/2010/03/auf-draht/>. [Zugriff am 2. Mai  
2023].

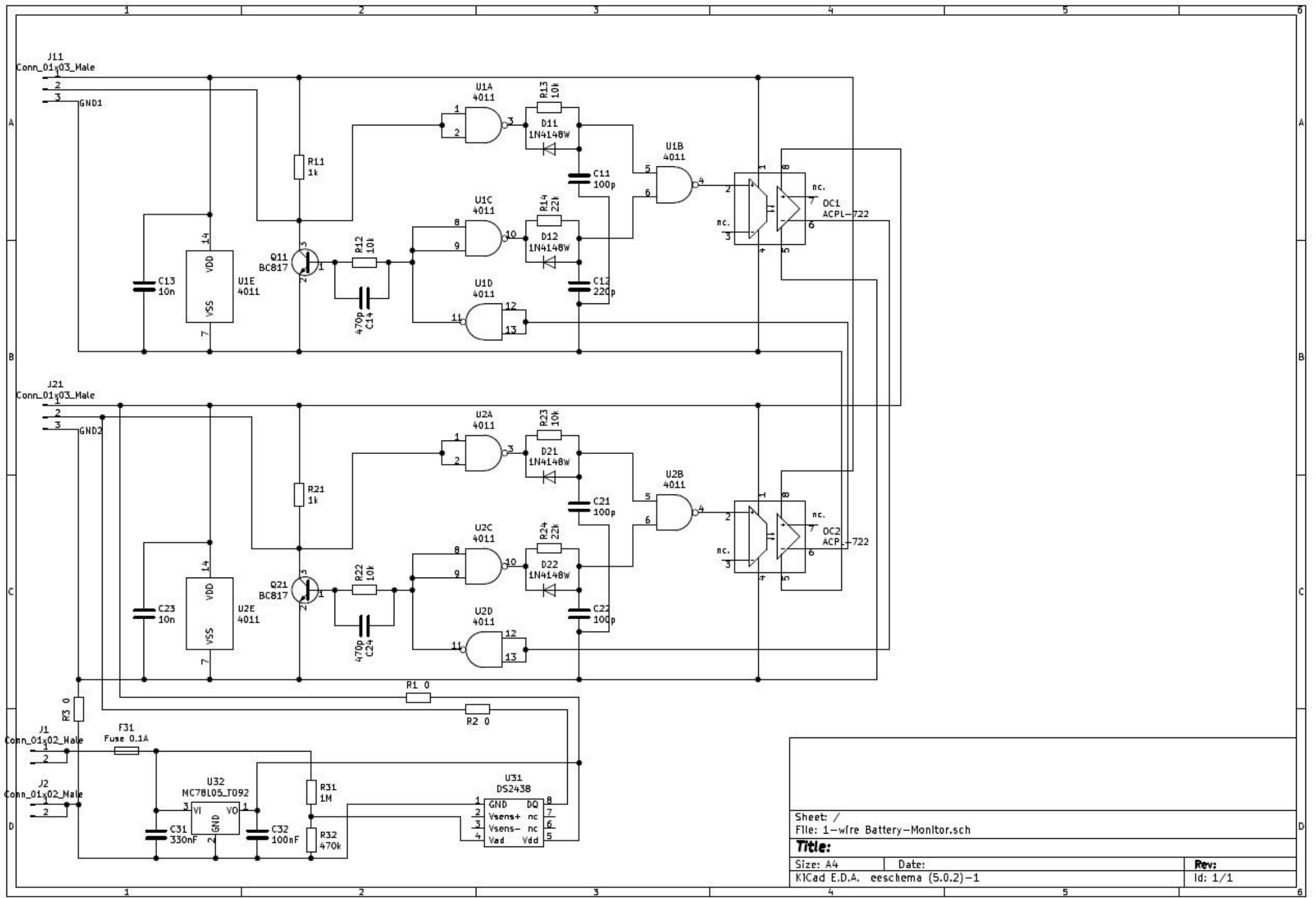
[2 Comexio GmbH, „Comexio,“ [Online]. Available:  
5] <https://www.comexio.com/grundlagen-1-wire/>. [Zugriff am 30. Juli 2023].

[2 Maxim Integrated , „Maxim Integrated,“ [Online].  
6]

[2 Esera, „Esera GmbH,“ [Online]. Available: [https://esera.de/Service-Support/1-Wire-  
7\] Grundlagen/1-Wire-Grundlagen/](https://esera.de/Service-Support/1-Wire-7-Grundlagen/1-Wire-Grundlagen/). [Zugriff am 12. August 2023].

## Anhang:

1. Schematik: „1-Wire-Batterie-Monitor“
2. CD mit den Programmcodes
3. Excel Datei für die RPB



Sheet: /  
 File: 1-wire Battery-Monitor.sch  
**Title:**  
 Size: A4      Date:  
 KiCad E.D.A. eeschema (5.0.2)-1      **Rev:**  
 Id: 1/1