

# **Entwicklung und Konstruktion eines Mikrocontroller gesteuerten Lithium-Ionen Batterieladegerätes und Monitoringsystems**

Diplomarbeit



Institut für Elektronik  
Technische Universität Graz

Betreuer

Ass.-Prof. Dipl.-Ing. Dr.techn. Gunter Winkler

Vorgelegt von Antonio Jonjic

**Institutsleiter: Univ.-Prof. Dipl.-Ing. Dr.techn. Wolfgang Bösch**

A - 8010 Graz, Inffeldgasse 12  
Telefon: (+43 316) 873 7521  
Telefax: (+43 316) 873 8020  
<http://www.ife.tugraz.at/>  
<http://www.tugraz.at>

Graz / April - 2012



## **Danksagung**

*Ich möchte mich an dieser Stelle beim Herrn Ass.-Prof. Dipl.-Ing. Dr. techn. Gunter Winkler für die Aufstellung eines wegweisenden Konzeptes für das Mikrocontroller gesteuerte Batterieladegeräte und Monitoringsystem sowie seine kompetenten fachlichen Ratschläge und Betreuung bedanken. Ohne seine Hilfeleistung wäre das Projekt nicht realisierbar gewesen.*

*Herrn Dipl.-Ing. Dr. techn. Mario Auer möchte ich für seine Unterstützung beim lösen von Problemen der Firmware des Geräte herzlich danken. Seine umfangreichen Kompetenzen und die guten Lösungsvorschläge waren sehr hilfreich.*

*Ebenfalls bedanken möchte ich mich beim Dipl.-Ing. Dr. techn. Christian Josef, Kreiner, sein herausragendes Wissen im Bereich der Firmware-Entwicklung hat mir sehr geholfen eine stabile und verlässliche Software für das Gerät zu entwerfen.*

*Ein Dank gebührt auch Herrn Ing. Issam Al-Abassy für die Möglichkeit die er mir gegeben hat, in der Firma AccuPower die Diplomarbeit zu realisieren.*

*Aus tiefstem Herzen und in besonderer Weise möchte ich meiner Mutter danken, die mich in meiner Absicht studieren zu gehen immer bestärkte, und mich während meines Studiums in jeder Hinsicht unterstützt hat. Ihr möchte ich meine Diplomarbeit widmen, ohne Sie hätte ich diese wundervolle Zeit niemals erleben können.*

Graz, April 2012

Antonio Jonjic

## **Kurzfassung**

Die Wirtschaft unserer komplexen Gesellschaftsform ist in großem Maße von ausreichend zur Verfügung stehender fossiler Energie abhängig. Diese Abhängigkeit wird durch schnell steigende Rohölpreise immer schmerzlicher und verlangt nach neuen Energieformen und einem rationalem Energieverbrauch. Die Mobilität ist der Kern dieser Problematik, hier ist auf lange Sicht der Umstieg auf Elektrofahrzeuge der offensichtlichste Lösungsweg. Der Umstieg auf diese neue Technologie verlangt neue und bessere chemische Energiespeicher, sowie eine effizientere Ladung und Entladung dieser. In diesem Sinne ist es sinnvoll eine genaue Kontrolle und Übersicht über den Lade und Entladevorgang von chemischen Energiespeichern zu haben. Diese Arbeit beschäftigt sich mit der Entwicklung und Konstruktion eines Mikrocontroller gesteuerten Ladegerätes für Lithium-Ionen Batterien, welches mit einer Tastatur und graphischen Display ausgestattet ist und mit diesen Elementen eine bessere Übersicht und Kontrolle des Ladevorganges ermöglicht. Der eingesetzte Messbaustein MAX11068 ermöglicht durch integrierte Schalter die Balancierung der Zellen eines Akkupacks und garantiert damit eine gleichmäßige und schonende Aufladung aller Zellen. Der verwendete Strommessbaustein DS2764 ermöglicht eine hochpräzise Strommessung von 0,625mA LSB. Die Firmware des Gerätes ist implementiert als eine „Statemachine“ die sich in einer sogenannten „Superloop“ (while Schleife) befindet. Zustandsänderungen der „Statemachine“ ergeben sich indem nach dem Polling-Prinzip Zustandsvariablen abgefragt werden.

Schlüsselwörter:

Elektrofahrzeuge, Mikrocontroller, Tastatur, graphisches Display, Ladegerät, Lithium-Ionen Batterie, MAX11068, DS2764, Firmware, Statemachine, Polling-Prinzip

## **Abstract**

The economy of our complex form of society is largely dependent sufficient available fossile fuels. This dependence becomes more painful by rapidly rising crude oil prices and requires new forms of energy and rational energy consumption. The mobility is the core of this problem, here is in long term the transition to electrical vehicles the most obvious solution. Upgrading to this new technology requires new and better chemical energy storage, and a more efficient charge and discharge of it. In this case, it makes sense to have an accurate control and visibility of the loading and unloading of chemical energy storage systems. This work deals with the development and construction of a microcontroller-controlled charger for Lithium-Ion batteries, which is equipped with a keyboard and graphic display. These elements enable a better overview and control of the charging process. The measuring ic MAX11068 allows the balancing of the battery cells of a pack by internal switches. These ensures a smooth and gentle charging of all cells. The current measuring ic DS2764 provides a highly accurate current measurement of 0.625mA LSB. The firmware of the device is implemented as a "State Machine" which is in a so-called "Superloop" (while loop) located. Changes of state of the "State Machine" results by checking the state variables, using the polling principle.

Keyword:

Electric vehicles, microcontrollers, keyboard, graphic display, battery charger, lithium-ion battery, MAX11068, DS2764, firmware, state machine, polling principle

Deutsche Fassung:

Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008  
Genehmigung des Senates am 1.12.2008

## **EIDESSTATTLICHE ERKLÄRUNG**

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am .....

.....

(Unterschrift)

Englische Fassung:

## **STATUTORY DECLARATION**

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

.....

Date

.....

(signature)

# Inhaltsverzeichnis

<b>1 Zeichenerklärung .....</b>	<b>10</b>
<b>2 Einleitung .....</b>	<b>14</b>
2.1 Speicherung elektrischer Energie .....	14
2.2 Zielsetzungen der Diplomarbeit.....	15
<b>3 Grundlagen.....</b>	<b>16</b>
3.1 Die Volta Säule .....	16
3.2 Grundlagen chemischer Speichersysteme .....	17
3.2.1 Die elektrolytische Zelle .....	19
3.3 Die verschiedenen chemischen Speichersysteme.....	22
3.3.1 Primärbatterien .....	22
3.3.2 Sekundärbatterien.....	24
3.4 Das Ragone-Diagramm .....	31
<b>4 Die Entladung einer Batterie: .....</b>	<b>33</b>
4.1 Einleitung.....	33
4.2 Die Blei-Säure-Batterie .....	35
4.2.1 Entlade-Charakteristik .....	35
4.2.2 Selbstentladung.....	36
4.3 Die Nickel-Cadmium-Batterie .....	36
4.3.1 Entlade-Charakteristik .....	37
4.3.2 Selbstentladung.....	38
4.4 Nickel-Metall-Hydrid-Batterie .....	39
4.4.1 Entlade-Charakteristik .....	39
4.4.2 Selbstentladung.....	40
4.5 Die Lithium/Lithium-Ionen –Batterie.....	41
4.5.1 Entlade-Charakteristik .....	42
4.5.2 Selbstentladung.....	43
4.5.3 Entlade-Charakteristik .....	43
4.5.4 Selbstentladung.....	43
4.6 Zusammenfassung der einzelnen Batteriesysteme .....	44
<b>5 Die Ladung einer Batterie.....</b>	<b>46</b>
5.1 Ladung der Blei-Säure-Zellen.....	48
5.1.1 Konstantspannungs-Methode.....	48

5.1.2	Konstantspannung ohne Strombegrenzung.....	48
5.1.3	Konstantspannung mit Strombegrenzung .....	50
5.1.4	Erhaltungsladung.....	52
5.1.5	Gemischte Methode.....	52
5.2	Lademethode von geschlossenen NiCd-Batterien .....	53
5.2.1	Konstantstromladung.....	54
5.2.2	Semikonstantstrom-Ladung .....	54
5.2.3	Erhaltungsladung.....	54
5.3	Lademethode von Nickel-Metall-Hydrid –Batterien.....	55
5.3.1	Semikonstantstrom .....	56
5.3.2	Konstantstrom.....	56
5.3.3	Abschaltkriterien bei NiMH-Systemen .....	57
5.4	Lademethoden von Lithium-Ionen-Systemen .....	58
<b>6</b>	<b>Das Batterie Management System .....</b>	<b>60</b>
6.1	Batterie Pack Architektur .....	60
<b>7</b>	<b>Balancierung .....</b>	<b>64</b>
7.1.1	Passives Balancieren.....	64
7.1.2	Aktives Balancieren .....	66
7.2	Balancierungs Bausteine (IC's).....	68
7.2.1	Anforderungen an den Balancierungs-Baustein.....	69
7.3	Die Auswahl des Balancierungsbausteines.....	72
7.3.1	Der Baustein MAX 11068 im Detail .....	75
7.4	Schaltungstechnische Veränderungen am Evaluation Board des MAX11068....	97
<b>8</b>	<b>Der Strommessbaustein DS2764.....</b>	<b>99</b>
8.1	Allgemeines .....	99
8.2	Die Strommessung.....	100
8.3	Die Strommengenmessung .....	100
8.4	Blockschaltbilde des DS2764 Bausteines .....	102
8.5	Anwendungsbeispiel des DS2764 .....	102
8.6	Beschaltung des Bausteines DS2764 in der Schaltung.....	103
8.7	Abbildung des Blockschaltbildes der Schaltung mit den System-Massen.....	104
<b>9</b>	<b>Der „Charge Pump“- Baustein ICL7660 .....</b>	<b>105</b>
9.1	Allgemeines .....	105
9.2	Die Funktionsbeschreibung.....	105

9.3	Anwendungsbeispiel des Bausteines ICL7660.....	107
<b>10</b>	<b>Das FTDI-Modul (UM232R mit FT232R-Chip) .....</b>	<b>108</b>
10.1	Allgemeines .....	108
10.2	Schaltung.....	108
10.3	Die USART-Schnittstelle .....	109
10.3.1	Allgemeines .....	109
10.3.2	Beschreibung der Hardware.....	109
<b>11</b>	<b>JTAG-Interface.....</b>	<b>111</b>
11.1	Warum JTAG .....	112
11.2	Funktionsweise.....	112
11.2.1	Das „Test Access Port“ .....	113
11.2.2	Der TAP-Controller .....	114
11.2.3	Die Register eines JTAG-fähigen Systems.....	115
<b>12</b>	<b>Die Steuerung des Ladegerätes.....</b>	<b>116</b>
12.1	Prinzipschaltbild der Steuerausgänge .....	117
12.2	Das passive Tiefpassfilter 3.Ordnung.....	117
12.3	LC Filter, T-Schaltung.....	117
<b>13</b>	<b>Blockschaltbild des Gesamtsystems.....</b>	<b>119</b>
13.1	Abbildung des Schaltplanes der Messplatine.....	120
13.2	Abbildung des Layouts der Messplatine .....	121
13.3	Abbildung des Schaltplanes der Steuerungsplatine.....	122
13.4	Abbildung des Layouts der Steuerungsplatine.....	123
<b>14</b>	<b>Der ARM-Prozessor .....</b>	<b>124</b>
14.1	Einleitung.....	124
14.2	Eigenschaften .....	124
14.3	Besonderheiten des Befehlssatzes.....	125
14.4	Blockschaltbild des STM32F103.....	126
<b>15</b>	<b>Softwaredokumentation .....</b>	<b>127</b>
15.1	Einbinden der STM32F10x Standard Peripheral Library .....	127
15.2	Die Programmstruktur der Firmware .....	129
15.3	Auszug aus der „main.c“ Datei .....	130
15.4	Auszug aus der „i2c_MAX11068.c“ Datei.....	135
15.5	Auszug aus der „i2c_DS2764.c“ Datei.....	138
15.6	Auszug aus der „pwm.c“ Datei .....	141

15.7	Auszug aus der „timer.c“ Datei.....	143
15.8	Auszug aus der „balancierung.c“ Datei.....	145
<b>16</b>	<b>Zusammenfassung und Ergebnisse .....</b>	<b>148</b>
<b>17</b>	<b>Abbildungsverzeichnis .....</b>	<b>152</b>
<b>18</b>	<b>Tabellenverzeichnis.....</b>	<b>154</b>
<b>19</b>	<b>Literaturverzeichnis.....</b>	<b>155</b>

## 1 Zeichenerklärung

### Symbolverzeichnis

<b>C</b>	Kapazität
<b>C<sub>F</sub></b>	Entstörkapazität
<b>D</b>	Diode allgemeine
<b>d</b>	duty cycle, Tasteverhältnis
<b>f</b>	Frequenz allgemein
<b>GND</b>	Ground
<b>GND<sub>U</sub></b>	pegelverschobenes Ground
<b>GND<sub>L</sub></b>	pegelverschobenes Ground
<b>DCIN</b>	Versorgungsspannung
<b>I</b>	Strom allgemein, Effektivwert
<b>I<sub>DIS</sub></b>	Entladestrom
<b>I<sub>BALANCE</sub></b>	Balancierungsstrom
<b>L</b>	Induktivität allgemein
<b>R</b>	Widerstand allgemein
<b>R<sub>load</sub></b>	Lastwiderstand
<b>F<sub>1</sub></b>	Schmelzsicherung
<b>R<sub>S</sub></b>	Schuntwiderstand
<b>R<sub>SW</sub></b>	Transistor On-Widerstand
<b>R<sub>EQ</sub></b>	Balancierungswiderstand
<b>R<sub>GATE</sub></b>	Transistor Gatewiderstand
<b>R<sub>BIAS1</sub></b>	Balancierungswiderstand
<b>S<sub>1</sub></b>	Schalter
<b>T</b>	Transistor, MOSFET

<b>V<sub>CC</sub></b>	Versorgungsspannung
<b>V<sub>Cell</sub></b>	Zellenspannung
<b>V<sub>+</sub></b>	positive Spannung
<b>V<sub>i</sub></b>	input Voltage, Eingangsspannung
<b>V<sub>OUT</sub></b>	output Voltage
<b>V<sub>DDU</sub></b>	pegelverschobene Versorgungsspannung
<b>V<sub>DDL</sub></b>	pegelverschobene Versorgungsspannung
<b>P<sub>V</sub></b>	Verlustleistung
<b>P<sub>MAX</sub></b>	Maximale Verlustleistung
<b>f</b>	Frequenz
<b>T</b>	Periodendauer
<b>t<sub>ein</sub></b>	Einschaltzeit
<b>t<sub>aus</sub></b>	Ausschaltzeit
<b>t</b>	Zeit allgemein
<b>t<sub>d</sub></b>	delay time, Verriegelungszeit
<b>t<sub>ovd</sub></b>	overvoltage detection
<b>t<sub>ocd</sub></b>	overcurrent detection
<b>t<sub>scd</sub></b>	short circuit
<b>t<sub>uvd</sub></b>	undervoltage detection
<b>(<math>\frac{\Delta T}{\Delta t}</math>)</b>	Temperaturanstieg
<b>U</b>	Spannung allgemein, Effektivwert
<b>W</b>	Energie [J]
<b>h</b>	1 Stunde
<b>SCL</b>	Taktsignal
<b>SDA</b>	Datensignal

<b><math>I_{s1}</math></b>	Schuntwiderstand Messeingang
<b><math>I_{s2}</math></b>	Schuntwiderstand Messeingang
<b><math>G(s)</math></b>	Transferfunktion

## Abkürzungsverzeichnis

<b>BJT</b>	Bipolar Transistor
<b>DC</b>	Direct Current, Gleichstrom
<b>EAGLE<sup>®1</sup></b>	Einfach anzuwendender grafischer Layout-Editor
<b>IC</b>	Integrated Circuit, integrierte Schaltung
<b>LED</b>	Light Emitting Diode, Leuchtdiode
<b>PWM</b>	puls width modulation, Pulsweitenmodulation
<b>SOC</b>	State of charge
<b>Keil<sup>®2</sup></b>	Fimrwareentwicklungsumgebung
<b>μC</b>	Mikrocontroller
<b>OPTO</b>	Optokoppler
<b>MCU</b>	Microcontroller Unit
<b>ARM</b>	Advanced RISC Machines
<b>JTAG</b>	Join Test Action Group (Programmierschnittstelle)
<b>I<sup>2</sup>C</b>	Serieller Datenbus für integrierte Baustein
<b>USART</b>	Universal Synchronous/Asynchronous Receiver Transmitter
<b>Li-Ion</b>	Lithium Ionen
<b>NimH</b>	Nickel Metall Hydrid

---

<sup>1</sup> <http://www.cadsoft.de/>

<sup>2</sup> <http://www.keil.com>

## 2 Einleitung

Zu den mit Abstand wichtigsten Erfindungen der letzten 400 Jahre gehört die Elektrizität. Zur Überraschung vieler gibt es die Elektrizität tatsächlich schon so lange. Sie wurde aber von der Menschheit erst im späten 19. Jahrhundert nutz voll eingesetzt. Zu den ältesten Methoden Elektrizität zu gewinnen, war, statische Ladung zu verursachen. Alessandro Volta (1745-1827) erfand die „elektrische Pistole“ und gilt als ein Begründer des Zeitalters der Elektrizität. Dabei handelte sich um einen elektrischen Leiter, der in eine mit Methangas gefüllte Kanne platziert wurde. Wurde ein elektrischer Strom durch den Draht geschickt explodierte die Kanne. Es gibt viele Möglichkeiten elektrische Energie zu erzeugen. Sie gehört zu einer Energieform die sowohl emissionsarm aber auch emissionslos erzeugt werden kann, wie dass der Fall bei erneuerbaren Energiequellen ist. Diese haben den Vorteil dass Sie sich selbst erneuern oder deren Nutzung nicht zur Erschöpfung der Quelle beiträgt. Im Mobilitätsbereich wird die elektrische Energie zur Zeit vor allem im Bereich der Triebfahrzeuge verwendet. Bei Zügen im Fernverkehr oder Regionalverkehr, im innerstädtischen Verkehr bei Straßenbahnen und U-Bahnen. Diese Art der Nutzung der elektrischen Energie im Mobilitätsbereich hat schon lange Einzug im Alltag erhalten. Obwohl das erste Automobil auf der Welt ein elektrisch betriebenes war, hat der elektrische Antrieb bei Personenkraftfahrzeugen noch nicht seinen Weg gefunden. Dies liegt einerseits an der geschichtlichen Entwicklung der Antriebstechnologien, durch die Erfindung der Verbrennungsmotoren, andererseits liegt es vor allem an der Möglichkeit qualitativ und quantitativ elektrische Energie zu speichern.

### 2.1 Speicherung elektrischer Energie

Bis heute werden vor allem chemische Speicher benutzt, um elektrische Energie in einer komfortablen Form zu speichern. Die Geschichte der chemische Speicher reicht lange in die Vergangenheit.

Durch die Eroberung des Verbrennungsmotors wurde die Technologie der chemischen Speicherung von elektrischer Energie lange vernachlässigt beziehungsweise konzentrierte Sie sich hauptsächlich auf den Bereich kleinere Energiemengen wie zum Beispiel Batterien für die Unterhaltungsindustrie, oder Batterien für mittlere Energiemengen im Bereich der Automobilität und Inselversorgungssystemen von nachrichtentechnischen Systemen. Durch die ökonomische Entwicklung auf der Welt, die rasant steigende Nachfrage nach Rohöl als Schmierstoff der Weltwirtschaft, wird die Entwicklung von Antriebstechnologien mit anderen Energieträgern immer interessanter. Als Beispiel die Wasserstofftechnologie, als alternativer Brennstoff und natürlich der Elektromotor mit chemischem Energiespeicher. Neben dem reinen Wasserstoffantrieb und Elektroantrieb ist die Hybridtechnologie die vielversprechendste Antriebstechnologie.

Zu den bekanntesten Speichertypen von chemischen Energiespeichern gehören wohl die Batterien und Akkumulatoren. Chemische Energiespeicher sind sehr weit verbreitet und finden ihre Anwendung täglich in unserem Alltag. Zu den ältesten Batterietechnologien gehört die Blei-Säure Batterie, der Blei-Säure Batterie folgte die Nickel-Cadmium Technologie, als Ersatz für die giftige Nickel-Cadmium Technologie wurde die NiMH Batterie entwickelt. Zu den neuesten und effizientesten Batterietechnologien gehört die Lithium-Ionen Batterie. Diese Technologie weist die höchste Energiedichte auf, ist aber von der Ladetechnik sehr viel aufwendiger. Die Lithium-Ionen Batterien benötigen eine Schutzelektronik.

## **2.2 Zielsetzungen der Diplomarbeit**

Im Rahmen dieser Diplomarbeit soll ein Lithium-Ionen Akkuladegerät entwickelt werden, welches die Möglichkeit bietet über eine Tastatureingabe und Computerschnittstelle die Ladeeinstellungen vorzunehmen.

### 3 Grundlagen

#### 3.1 Die Volta Säule

<sup>3</sup>Als Erfinder der ersten Batterie gilt Alessandro Volta (\* 18. Februar 1745 in Como, Italien; † 5. März 1827 in Camnago bei Como) der 1800 die Entdeckung machte das zwischen zwei in Elektrolyt eingetauchten metallischen Elektroden ein kontinuierlicher Stromfluss möglich ist. Die Voltasche Säule wurde von ihm 1799 entwickelt, diese funktioniert nach dem Prinzip eines galvanischen Systems. Damit war die Volta-Säule die erste elektrochemische Stromquelle und Vorläufer der heutigen Batterie. Volta stapelte dabei abwechselnd Zink- und Kupferplatten übereinander, diese Platten wurden mit einer Salzsäure getränkten Pappe getrennt. An der negativen Zinkanode kommt es zur Oxidation (es werden Elektronen abgegeben) und durch die Reduktion von Sauerstoff durch das entstehende Kupferoxid an der positiven Kupferkathode bildet sich eine elektrochemische Stromquelle. Durch die unterschiedliche Konsistenz lösen sich die Zinkelemente rascher auf und geben dabei jeweils zwei Elektronen ab. Somit entsteht ein Elektronenüberschuss in der Zinkelektrode und das Zink wird zur negativen Elektrode. An der Kupferkathode kommt es zu einer Reduktion, dies nimmt Elektronen auf. Am Ende der Säule kann eine Spannung abgegriffen werden, pro Kupfer-Zink Kombination entsteht eine Spannung von 1,1 V. Umso mehr Säulen hintereinander geschaltet werden umso höher ist die Spannung.

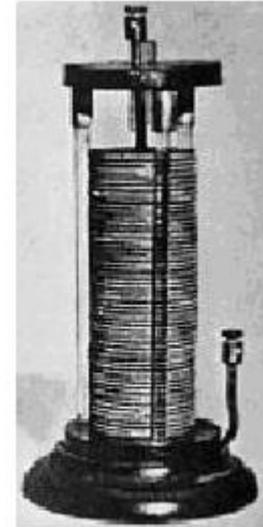


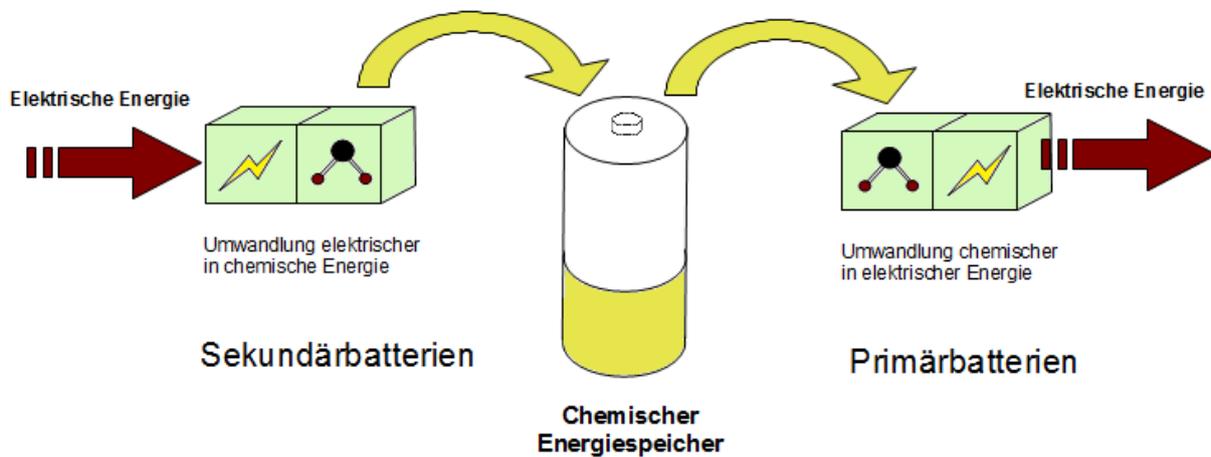
Abbildung 1: Die Volta Säule

---

<sup>3</sup> <http://www.gsg-physik.net>

## 3.2 Grundlagen chemischer Speichersysteme

<sup>4</sup>Elektrochemische Speicher können grundsätzlich in zwei Gruppen aufgeteilt werden in Primärsysteme und Sekundärsysteme. Primärsysteme sind Batterien die nicht wieder aufladbar sind, weil der chemische Prozess bei der Entladung nicht reversibel ist. Sekundärsysteme sind Batterien die wiederaufladbar sind, weil die elektrische Energie wieder in chemische Energie umgewandelt werden kann. Die Aufgabe eines chemischen Speichers besteht darin die in chemischer Form gespeicherte Energie als elektrische Energie wieder abzugeben.



**Abbildung 2: Funktionsweise eines chemischen Energiespeichers**

Jede Batterie ist im Grunde eine galvanische Zelle, diese geht zurück auf den italienischen Arzt Luigi Galvani. Er entdeckte zufällig bei einem Versuch an einem Froschschenkel dass dieser Muskelzuckungen auslöste als er ihn mit seinem aus verschiedenen Metall zusammengesetzten Instrument berührte. Das rührte daher dass sich ein Redox-System bildete das als galvanisches Element eine Spannung aufbaute, und diese einen Stromfluss verursachte. <sup>5</sup>In Abbildung 3 ist das Prinzip einer galvanischen Zelle abgebildet. Die galvanische Zelle ist eine elektrochemische Zelle die eine elektrische Spannung erzeugt, weil in ihr eine spontane chemische Reaktion abläuft.

<sup>4</sup> Batteries in a portable world, Buchman Isidor

<sup>5</sup> Batterien und Ladekonzepte, Halaczek, Thaddäus Leonhard, Radecke, Hans Dieter

Das bedeutet chemische Energie wird in elektrisch Energie umgewandelt. Damit ist die galvanische Zelle die Grundlage bzw. in ihrer Funktionsweise das Grundprinzip einer jeden Batterie. Die spontane chemische Reaktion die da abläuft ist eine Redoxreaktion. Die Redoxreaktion besteht aus zwei Teilreaktionen nämlich einer Oxidationsreaktion und einer

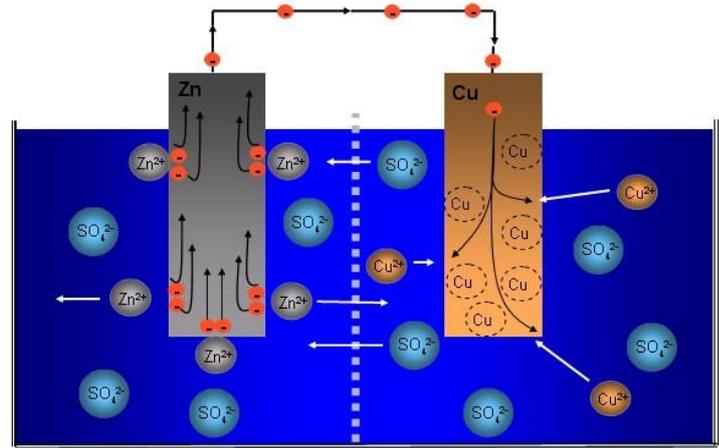


Abbildung 3: Darstellung eines galvanischen Zelle mit Zinn und Kupfer als Reaktionspartner

Reduktionsreaktion. Die Oxidation eines Stoffes ist mit einer Elektronenabgabe verbunden, die Reduktion eines Stoffes ist mit einer Elektronenaufnahme verbunden. Es kommt also bei der Redoxreaktion zu einer Elektronen Übertragung, oder einem Fluss von Elektronen. Ziel ist es diesen Elektronenfluss extern über einen Verbraucher verlaufen zu lassen. Diese beiden Teilreaktionen müssen räumlich voneinander getrennt sein. Das geschieht durch das Daniell-Element, dieses ist aufgebaut aus zwei Halbzellen die durch eine semipermeable Membran voneinander getrennt sind. In der linken Halbzelle befindet sich eine Zinksulfatlösung, in der rechten eine Kupfersulfatlösung. In der Zinksulfatlösung ist ein Stab aus Zinkmetall eingetaucht, in der Kupfersulfatlösung ist ein Kupferstab eingetaucht. Diese beiden Metallstäbe sind elektrisch leitend miteinander verbunden. Es wird nun ein Teil des Zinkmetalls gelöst und es bilden sich Zink-Kationen, im Zinkstab bleiben nun Elektronen zurück. Es bildet sich eine sogenannte Doppelschicht entlang der Metallstabgrenze mit einem Potential. An dem Kupfermetallstab läuft der selber chemische Prozess ab, es bleiben jedoch wesentlich weniger Elektronen im Kupferstab zurück der Kupfer das edlere Metall ist. Da Zink das unedlere Metall ist (eher bereit Elektronen abzugeben) wird der Zinkmetallstab negativ geladen sein (Anode) und der Kupfermetallstab wird positive geladen (Kathode) sein. Da zwischen den beiden Elektroden eine Potentialdifferenz herrscht kommt es zu einem Ausgleichvorgang bzw. Elektronenfluss von der Anode zu Kathode wenn sie elektrisch miteinander verbunden sind. Die zusätzlich Elektronen in der Kupferelektrode führen dazu das die Kupfer-Ionen in der Lösung reduziert werden.

An der Kupferkathode scheidet sich noch mehr elementares Kupfer ab, der Kupfersulfatlösung werden Kationen entzogen. In der Zinksulfatlösung passiert es genau umgekehrt, hier wird immer mehr elementares Zink gelöst und Lösung bekommt immer mehr Kationen. Durch die halbdurchlässige Membran können nun Sulfat-Anionen durch diffundieren und es kommt zu einem Ladungsausgleich mit den positiven Ladungen der Zinksulfatlösung. Das heißt Außen fließen die Elektronen und in der Zelle bzw. zwischen den beiden Halbzellen fließen die Sulfat-Anionen. Die wichtigsten Teilchen sollen hier zusammenfassend nochmals aufgeführt werden, die auf den Weg von der Anode zur Kathode vorkommen.

- *Elementares Zink*
- *Zinksulfatlösung*
- *Halbdurchlässige Membrane*
- *Kupfersulfatlösung*
- *Elementares Kupfer*

### 3.2.1 Die elektrolytische Zelle

Eine Zelle bei welcher ein Strom eingepreßt wird, also bei welcher ein Strom in die Gegenrichtung getrieben wird, heißt „elektrolytische Zelle“. Diese soll hier anhand einer Blei-Säure Batterie-Zelle erklärt werden. Als erstes betrachten wir die galvanische Zelle, die wir vorab schon anhand einer Zink-Eisen-Zelle beschrieben haben.

Die Blei-Säure-Zelle ist aufgebaut wie folgt:

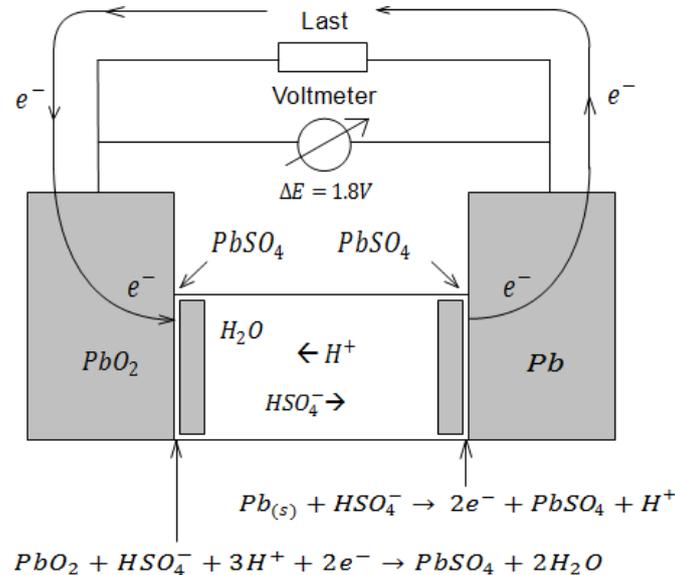
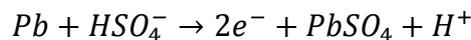


Abbildung 4: Darstellung einer galvanischen Zelle ausgeführt als Blei-Säure Zelle

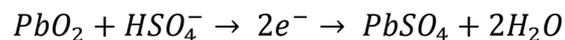
Wie bereits erwähnt ist eine chemische Stromquelle im wesentlichen ein Stromgenerator der durch zwei chemische Reaktionen seine Wirkung hat. Dabei laufen folgende chemischen Teilreaktionen ab:

Da sich die Elektronen nicht im Elektrolyt bewegen können und Ionen nicht in der Elektrode, finden die chemischen Reaktionen an den Grenzflächen der Elektroden statt. Diese Reaktionen sind folgende:

Auf der **Anode**, der negativen oxydierenden Elektrode lautet:



Auf der **Kathode**, der positiven, reduzierenden Elektrode lautet:



Die Elektronen fließen hier von der Pb-Elektrode durch den elektrischen Leiter zu PbO<sub>2</sub>-Elektrode. Eine Zelle die auf dieser Weise arbeitet nennt man „galvanische Zelle“. Wenn wir nun anstatt der Last eine Stromquelle in den elektrischen Kreis der Zelle einbauen, wird ein Strom in die Gegenrichtung getrieben. Die Funktion der Elektroden hat sich jetzt umgekehrt.

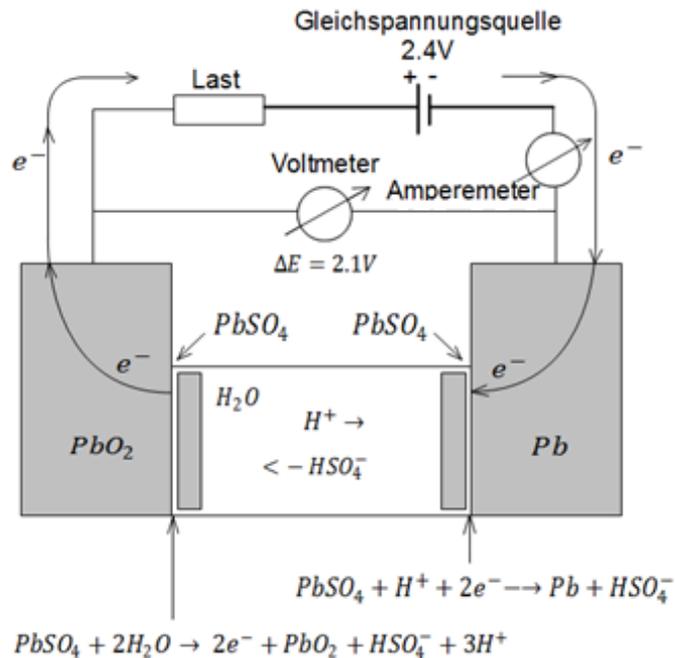
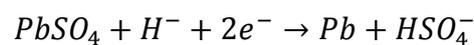


Abbildung 5: Darstellung einer elektrolytischen Zelle ausgeführt als Blei-Säure Zelle

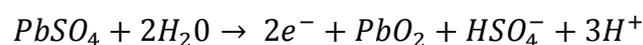
Das bedeutet nicht dass die Anode jetzt zur Kathode und umgekehrt geworden ist. Die Bezeichnung der Elektroden wurde nämlich für die galvanische Zelle definiert und ändert sich deswegen nicht.

Hier laufen jetzt folgende chemische Reaktionen ab:

An der **Pb-Elektrode**:



An der **PbO<sub>2</sub>-Elektrode** reagiert:



Bei diesen Reaktionen wird das Produkt der Elektrodenreaktionen in ihr Ausgangsprodukt umgewandelt.

### 3.3 Die verschiedenen chemischen Speichersysteme

#### 3.3.1 Primärbatterien

##### 3.3.1.1 Die Zink-Kohle Batterie

<sup>6</sup>Die Zink-Kohle Batterie gehört zu den Primärelementen, das heißt sie ist nicht wiederaufladbar. Die negative Elektrode besteht aus einer zu einem Becher geformten Zinkelektrode und der positive Pol besteht aus Mangandioxidpulver mit einer Kohlelektrode als zentrische elektrische Zuleitung. Als Elektrolyt wird eine 20%ige Ammoniumchloridlösung eingesetzt. Bei der Entladung der Zink-Kohle Batterie oxidieren die Zinkatome zu Zinkionen und geben dabei pro Zinkatom zwei Elektronen ab. Dadurch wird die Elektrode negativ. Das heißt das unedle Metall Zink löst sich im Elektrolyt selbst auf, das bedeutet die Zinkelektrode gibt Zink-Ionen ab und lädt sich negativ auf. Auf der Zinkelektrode kommt es nun zu einem

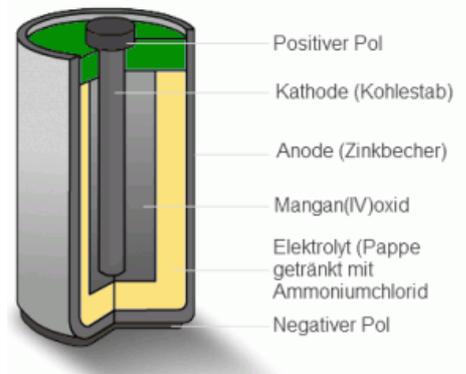


Abbildung 7: Querschnitt einer Zink-Kohle Batterie

Elektronen Überschuss, solange der Stromkreis nicht leitend mit dem Kohlestift verbunden ist können die „überschüssigen Elektronen nicht abfließen. Diese „überschüssigen“ Elektronen verhindern die Bildung weiterer Zink-Ionen und es stellt sich ein Gleichgewichtszustand ein. Diese im Elektrolyt gelösten Zink-Ionen laden den leitend verbundenen Kohlestab positiv auf. Zwischen den beiden Elektroden bildet sich eine Spannung von 1,5V aus. Wird der Stromkreis nun leitend verbunden fließen die Elektronen über den Verbraucher von der Zinkelektrode zu der Kohlelektrode und ziehen auch die positiven Zink-Ionen zur Kohlelektrode. Dieser Vorgang dauert so lange an bis das Elektrolyt keine Zink-Ionen mehr aufnehmen kann. Ein Hauptproblem der Zink-Kohle

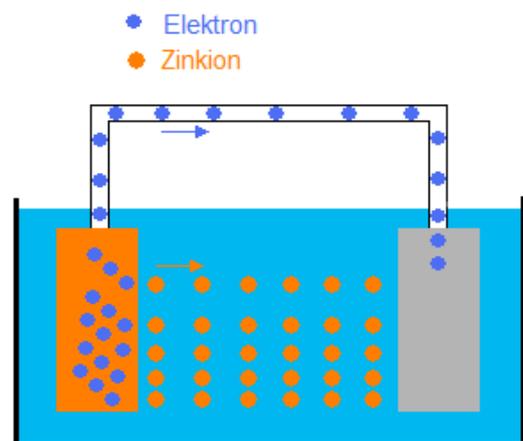


Abbildung 6: Querschnitt einer Zink-Kohle Batterie

Batterien ist das sie oft nicht Auslaufsicher sind, weshalb sie nicht in Geräten verwendet werden sollten in denen sie länger verbleiben.

<sup>6</sup> Batteries in a portable world, Buchman Isidor

### 3.3.1.2 Alkali Mangan Batterie

Die Alkali Mangan Zelle gehört zu Primärbatterien und ist einer der wichtigsten chemischen Energiespeicher. Aufgrund der vielen Vorteile gegenüber der Zink-Kohle Batterie, wie höhere Kapazität, bessere Belastbarkeit und längere Lagerfähigkeit hat sie die Zink-Kohle Batterie aus vielen Anwendungen verdrängt. Die Alkali Mangan Batterie ist aufgebaut aus einer Anode aus Zink-Pulver, einer Kathode

aus Mangan-Dioxid und Graphit, einer Metallhülse und einem sogenanntem Nagel aus Messing oder Kupfer. Genau wie bei der Zink-Kohle Batterie wird die elektrische Energie geliefert durch die Oxidation von Zink und die Reduktion von Mangandioxid. Die freigesetzten Elektronen wandern über den geschlossenen Stromkreis von der Anode zur Kathode, während OH<sup>-</sup>-Ionen zum Ladungsausgleich durch das Elektrolyt von der Kathode zur Anode wandern. Obwohl die Alkali Mangan Batterie zu den Primärbatterien zählt die nicht wiederaufladbar sind, kann sie begrenzt wiederaufgeladen werden. Es gibt spezielle Versionen

der Alkali Mangan Batterie die sogenannte RAM-Zellen (Rechargeable Alkaline Manganese) diese können zu den wiederaufladbaren Sekundärbatterien gezählt werden. Die RAM-Zellen werden speziell in Bereichen eingesetzt wo es um lange Lagerfähigkeit geht, was ihren Einsatz auf sehr spezielle Anwendungen begrenzt, wie zum Beispiel in Geräten für den Bergbau.

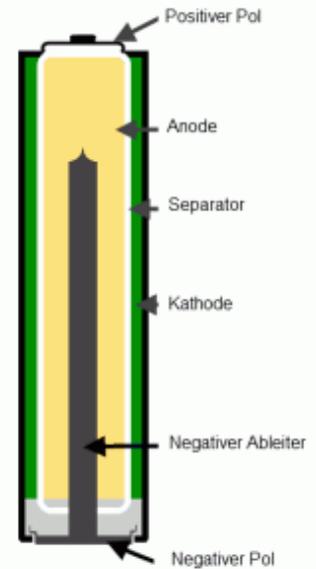


Abbildung 8: Querschnitt einer Alkali-Mangan Batterie

### 3.3.1.3 Die Knopfzelle

Die erste Knopfzelle wurde schon 1883 erfunden. Sie wurde anfänglich hauptsächlich im militärischen und im Bereich der Raumfahrt eingesetzt. Wiederaufladbare Knopfzellen besitzen eine Membran die als zusätzlicher

Separator für eine längere Lebensdauer dient. Abhängig vom Elektrodenmaterial unterscheidet man Silberoxid, Quecksilberoxid und Lithiumzelle. Knopfzellen werden in Geräten eingesetzt die einen geringen Strombedarf haben, wie z.B. Armbanduhren, Taschenrechner etc.

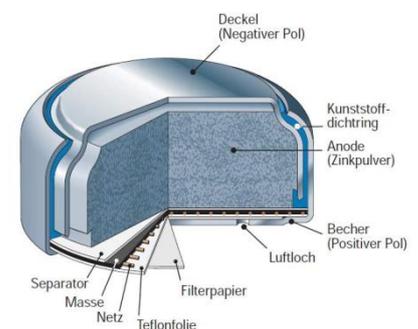


Abbildung 9: Querschnitt einer Knopfzelle

### 3.3.2 Sekundärbatterien

#### 3.3.2.1 Die Bleibatterie

Die Bleibatterie gehört zu der ältesten wiederaufladbaren Batterie. Sie wurde vor über 100 Jahren erfunden und gehört heute noch zu den am häufigsten eingesetzten Batterietypen. Ihre Erfindung geht zurück auf den deutschen Physiker Jose Sinsteden und dem Franzosen Gaston Plante zurück. Das Prinzip der Bleibatterie entdeckte Sinsteden in dem er zwei Bleiplatten mit einem Abstand in eine verdünnte Schwefelsäure einlegte. Er legte an die Bleiplatten eine Spannung an und ließ einen Ladestrom fließen, nach dem Ladevorgang konnte die Zelle wieder entladen werden. Gaston Plante hat durch eine spiralenförmige Anordnung der Bleiplatten den Bleiakкумуляtor erheblich weiterentwickelt. Camille Faure entwickelte 1880 die pastierte Platte, diese bestand aus einer Beschichtung aus Bleipulver (Bleioxid) und Schwefel.

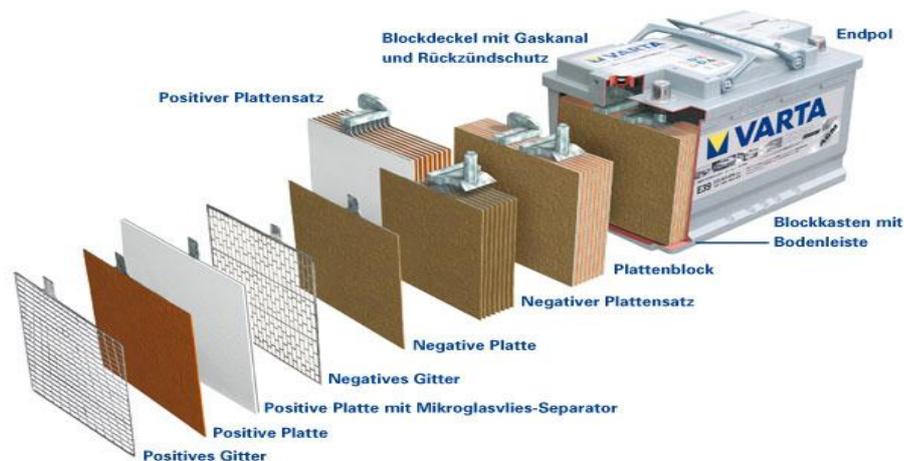


Abbildung 10: Querschnitt einer Blei-Säure Batterie

Diese Methode der Elektrodenherstellung wird heutzutage noch angewendet. Die Bleibatterie zeichnet sich durch eine sehr geringe Selbstentladung aus (40% pro Jahr), und es gibt keinen Memory-Effekt. Allerdings sind Bleibatterien nicht geeignet für Schnellladungen, typische Ladezeiten sind 8 bis 16 Stunden, sie darf auch nicht im entladenen Zustand gelagert werden da dies eine Sulfatierung zur Folge hat. Abhängig

vom Tiefentladung und Umgebungstemperatur hat die Bleibatterie eine Lebenserwartung von 200 bis 300 Lade-/Entlade –Zyklen. Durch ihre geringe Energiedichte von 0,11 MJ/kg ist die Bleibatterie ungeeignet für den Einsatz in mobilen Geräten.

### 3.3.2.2 Die Nickel-Cadmium Batterie

Die Nickel-Cadmium Batterie wurde 1899 vom Schweden Waldmar Junger erfunden. Die Nickel-Cadmium Batterie gehört zu den alkalischen Batteriesystemen. Zur gleichen Zeit arbeitete Thomas Alva Edison an der Entwicklung des Nickel-Eisen-Akkumulators, allerdings hat die Nickel-Cadmium Kombination eine etwas höhere Energiedichte. Im Jahr 1908 entwickelte Edison die Röhrenelektrode was der Batterie eine höhere mechanische Festigkeit gibt. Die Elektroden bestehen aus zwei Platten wo die positive Elektrode mit Nickel-oxidhydroxid beladen ist und die negative Elektrode mit fein verteiltem Cadmium. Als Elektrolyt wird eine 20% Kaliumhydroxid-Lösung verwendet. Diese Kombination liefert eine Spannung von 1,3 V. Bei der Nickel-Cadmium Batterie passiert eine Oxidation des metallischen Cadmiums und gleichzeitig kommt es zu einer Reduktion von Nickel Oxid-Hydrat in einem alkalischen Elektrolyt. Den Elektroden wird Graphitpulver zugegeben um die Leitfähigkeit zu verbessern. Bei Überladung der Batterie wird an der positiven Elektrode Sauerstoff und an der negative Elektrode Wasserstoff produziert. Um Explosionen zu verhindern wird die negative Cadmiumelektrode überdimensioniert und dient als Entladereserve. Die Nickel-Cadmium Batterie zeichnet sich vor allem durch Ihre Hochstromfähigkeit aus beim Laden sowie Entladen. Bei entsprechender Bauform ist eine Schnellladung innerhalb von 10 Minuten möglich. Die Nickel-Cadmium Batterie ist sehr robust was die Überladung und Tiefentladung betrifft, kältefest und hat eine hohe Lebensdauer. Vor allem ist das Temperaturverhalten der Batterie gegenüber der anderen Technologie hervorzuheben. So besitzt die Nickel-Cadmium Batterie bei  $-40^{\circ}\text{C}$  mit Faserstrukturplatten-Technik noch immer 50% ihrer Nominale Kapazität bei Raumtemperatur. Allerdings bevorzugt die Nickel-Cadmium Batterie eine Schnellladung gegenüber einer Langsamladung. Die spezifische Energiedichte der Nickel-Cadmium Batterie beträgt  $40 \text{ Wh/kg}$  was die Batterie im unteren Kapazitätsbereich ansiedelt.

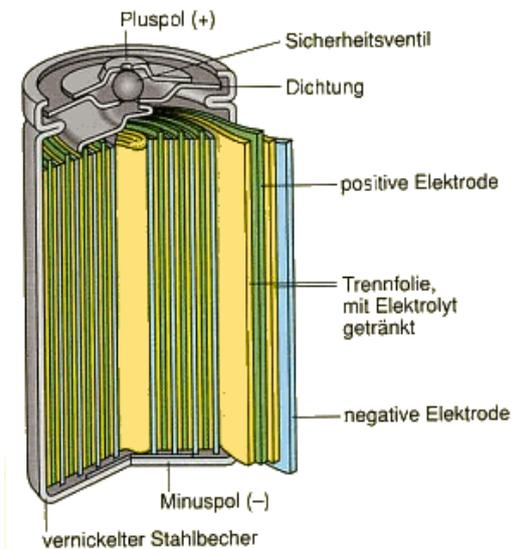


Abbildung 11: Querschnitt einer Nickel-Cadmium Batterie

### 3.3.2.3 Die Nicke-Metall-Hydrid Batterie

Die Entwicklung an der Nickel-Metall-Hydrid Batterie begannen mit dem Fortschritt in der Raumfahrt in den 1970 er Jahren. Die Nickel-Metall-Hydrid Batterie ist dabei eine Weiterentwicklung der Nickel-Wasserstoff Batterie. Dabei wurde die Cadmium Elektrode durch eine Wasserstoffelektrode ersetzt. Die Nickel-Hydrogen-Batterien verlangen ein Hochdruckstahlgehäuse und sind deshalb entsprechend teuer. Die negative Elektrode der Nickel-Metall-Hydrid Batterie besteht aus einer Seltenerd-Nickellegierung, während die positive Elektrode im geladenen Zustand aus Nickel-Oxyhydrat und im entladenen Zustand aus Nickelhydroxid besteht. Als Elektrolyt in der Nickel-Metall-Hydrid Batterie wird eine Kalilauge verwendet. Es werden bei der Ladung der Batterie an der positiven Elektrode negative Hydroxid-Ionen verbraucht und an der negative Elektrode

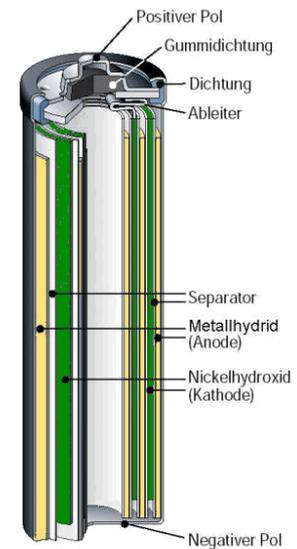


Abbildung 12: Querschnitt einer Nickel-Metall-Hydrid Batterie

produziert. Aufgrund der ähnlichen Nennzellenspannung von 1,2V konnte die Nickel-Cadmium Batterie direkt von der Nickel-Metall-Hydrid Batterie ausgetauscht werden. Dabei hat die Nickel-Metall-Hydrid Batterie den Vorteil einer höheren Energiedichte und dem Fehlen des giftigen Cadmiums. Die spezifische Energiedichte der Nickel-Metall-Hydrid Batterie beträgt 80 Wh/kg und ist fast doppel so groß wie die der Nickel-Cadmium Batterie. Allerdings ist die Nickel-Metall-Hydrid Batterie weniger robust und dauerhaft sowie mehr Temperaturabhängig als die Nickel-Cadmium Batterie. Besonders die Selbstentladung der Nickel-Metall-Hydrid Batterie ist eine große Schwachstelle. Bei einer Lagerung von 4 Wochen ohne Gebrauch, verliert der Nickel-Metall-Hydrid Batterie ca 20-50% ihrer Ladung. Die Nickel-Metall-Hydrid Batterie wird vor allem als Ersatz für die giftige Nickel-Cadmium Batterie eingesetzt. Die Nickel-Metall-Hydrid Batterie wird nach dem  $-\Delta U$  Kriterium geladen, das bedeutet wenn die Batterie vollgeladen ist beginnt ihre Zellenspannung leicht zu sinken.

### 3.3.2.4 Die Lithium/Lithium-Ionen Batterie

<sup>7</sup>Lithium Systeme lassen sich aufgrund der charakteristischen Merkmale in fünf Kategorien klassifizieren.

- **Zellen mit flüssigem organischem Elektrolyten:**
  - Festkörperkathode: Einlagerungsverbindung
  - Anode: metallisches Lithium
  - Elektrolyt: flüssig, organisch
  
- **Zellen mit festem Elektrolyten:**
  - Festkörperkathode: Einlagerungsverbindung
  - Anode: metallisches Lithium
  - Elektrolyt: festes Polymer
  
- **Lithium-Ionen-Zellen:**
  - Festkörperkathode: Einlagerungsverbindung
  - Anode: Einlagerungsverbindung
  - Elektrolyt: flüssig, oder festem Polymer
  
- **Zellen mit anorganischem Elektrolyten:**
  - Flüssige Kathode: gleichzeitig Lösungsmittel des Elektrolyten
  - Anode: metallisches Lithium
  
- **Lithium-Legierungs-Zellen:**
  - Kathoden: verschiedene Kathoden
  - Anode: Lithium-Legierung
  - Elektrolyt: flüssig, organisch

---

<sup>7</sup> Batterien und Ladekonzepte, Halaczek, Thaddäus Leonhard, Radecke, Hans Dieter

<sup>8</sup>Die ersten Entwicklungen an der Lithium Batterie begannen 1912 unter G.N. Lewis. Die erste kommerzielle Lithium Batterie wurde in den 70er Jahren hergestellt. Allerdings schlugen mehrere Versuche fehl eine wiederaufladbare Lithium Batterie herzustellen aufgrund von Sicherheitsproblemen. Da das metallische Lithium sehr instabil ist, ging die Entwicklung in Richtung einer nicht metallischen Lithium Batterie, unter Verwendung von Lithium-Ionen. Diese Batterie hat zwar eine etwas kleinere Energiedichte als metallisches Lithium, ist aber eine sicherer Substanz wenn gewisse Vorkehrungen beim Laden und Entladen getroffen werden. Die SONY® Corporation brachte 1991 die erste kommerzielle Lithium-Ionen Batterie auf dem Markt. Das aktive Material der negativen Elektrode einer Lithium- Ionen Batterie kann aus

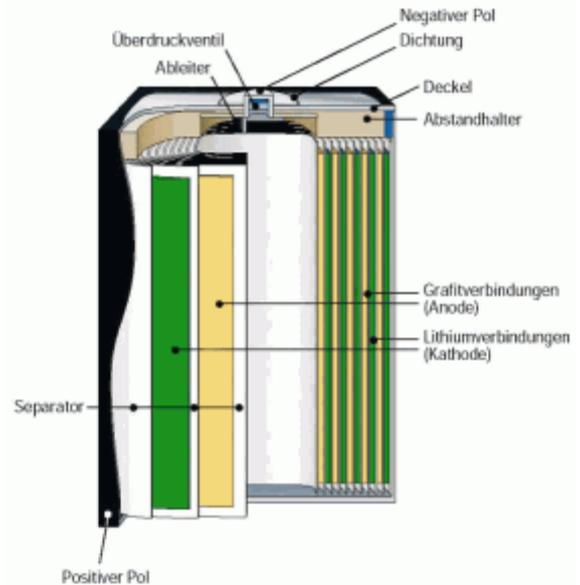


Abbildung 13: Querschnitt einer Lithium-Ionen Batterie

Graphitoder aus Nanokristallinem, amorphem Siuzium aufgebaut sein. Die positive Elektrode enthält Lithium-Metalloxide, vom Spinell-Typ wie  $\text{LiCoO}_2$  (Lithium cobaltdioxid),  $\text{LiNiO}_2$  und  $\text{LiMn}_2\text{O}_4$  oder  $\text{Li}_4\text{Ti}_5\text{O}_{11}$  und  $\text{LiFePO}_4$  (Lithium-Eisen-Phosphaat Batterie). Das Elektrolyt muss wasserfrei sein, damit es nicht mit dem Lithium reagiert. Dafür werden aprotische Lösungsmittel verwendet wie Ethylencarbonat, Propylencarbonat, Dimethylcarbonat, Diethylcarbonat oder 1,2-Dimethoxyethan gelöste Lithiumsalze wie  $\text{LiPF}_6$ . Eine konventionelle Lithium-Ionen Batterie liefert eine Spannung von 3.6 V, fast dreimal so viel wie eine NiMH Batterie, was dein Einsatz von nur einer Zelle in mobilen Geräte möglich macht. Die spezifische Energiedichte ist relativ hoch und beträgt  $100\text{Wh/kg}$ . Wenn man auf die Entladespannung von minimal 3V achtet, um die Lebensdauer zu verlängert, dann sinkt die spezifische Energiedichte auf  $60\text{--}70\text{Wh/kg}$ . Die Kapazität einer Lithium-Ionen Batterie verringert sich mit der Zeit von selbst. Dabei gilt die Faustregel dass eine Lithium-Ionen Batterie nach drei Jahren 50% ihrer Kapazität eingebüßt hat, was weniger als die Hälfte gegenüber einer Nickel-Cadmium Batterie ist. Eine Entladung der Batterie unter 40% sollte vermieden werden, da es bei tiefen Zyklen zu größeren Kapazitätsverlusten kommt aufgrund von irreversiblen Reaktionen auf der Elektrode. Eine Lithium-Ionen Batterie benötig nur

<sup>8</sup> Batteries in a portable world, Buchman Isidor

wenig Wartungsaufwand, ein Vorteil gegenüber anderen Chemien. Da kein Memory-Effektauftritt sind keine regelmäßigen Lade/Entlade-Zyklen nötig. Trotz vieler Vorteile hat die Lithium-Ionen Batterie den Nachteil dass sie sehr empfindlich ist was eine Schutzschaltung unbedingt nötig macht. Diese Schutzschaltung begrenzt die Spitzenspannung beim Laden und beim Entladen verhindert sie ein zu tiefes absinken der Zellenspannung. Der maximale Lade- und Entladestrom liegt zwischen 1C und 2C. Das Laden einer Lithium-Ionen Batterie erfolgt nach dem IU-Verfahren. (CC/CV Methode, Constant Current/Constant Voltage). Zuerst wird der Ladestrom bei z.B. 1C konstant gehalten, nach Erreichen der maximalen Ladeschlussspannung wird die Spannung konstant gehalten und die Batterie nimmt den Strom auf den sie kann. Die Lithium-Ionen Batterie gehört zu der zukunftsweisenden Technologie und wird von den Herstellern in ihren Eigenschaften laufend verbessert. Die wirtschaftlichste Lithium-Ionen Batterie ist die in der zylindrischen Bauform vom Typ 18650 (18 mm Durchmesser und 65 mm Länge).



Abbildung 14: Darstellung von Lithium-Ionen Zellen im gängigem Gehäuseformat 18650

### 3.4 Das Ragone-Diagramm

Je nach Anwendung und Betriebsbedingungen gibt es einen entsprechenden Batterietypen der am besten geeignet ist. Die einzelnen Batterietypen unterscheiden sich dabei in ihrer spezifischen Energiedichte und spezifischen Leistungsdichte. Dabei haben Batterietypen mit einer hohen Energiedichte eine geringe Leistungsdichte und die Batterietypen mit einer hohen Leistungsdichte eine geringe Energiedichte. Das Ragone-Diagramm stellt genau diesen Zusammenhang der Leistungsdichte gegenüber der Energiedichte dar. Im Ragone-Diagramm wird die spezifische gravimetrische Leistung  $W/kg$  über die spezifische gravimetrische Energie  $Wh/kg$  aufgezeichnet. Somit können über das Ragone-Diagramm die einzelnen Eckdaten der verschiedenen Batterietypen schnell und einfach erfasst werden.

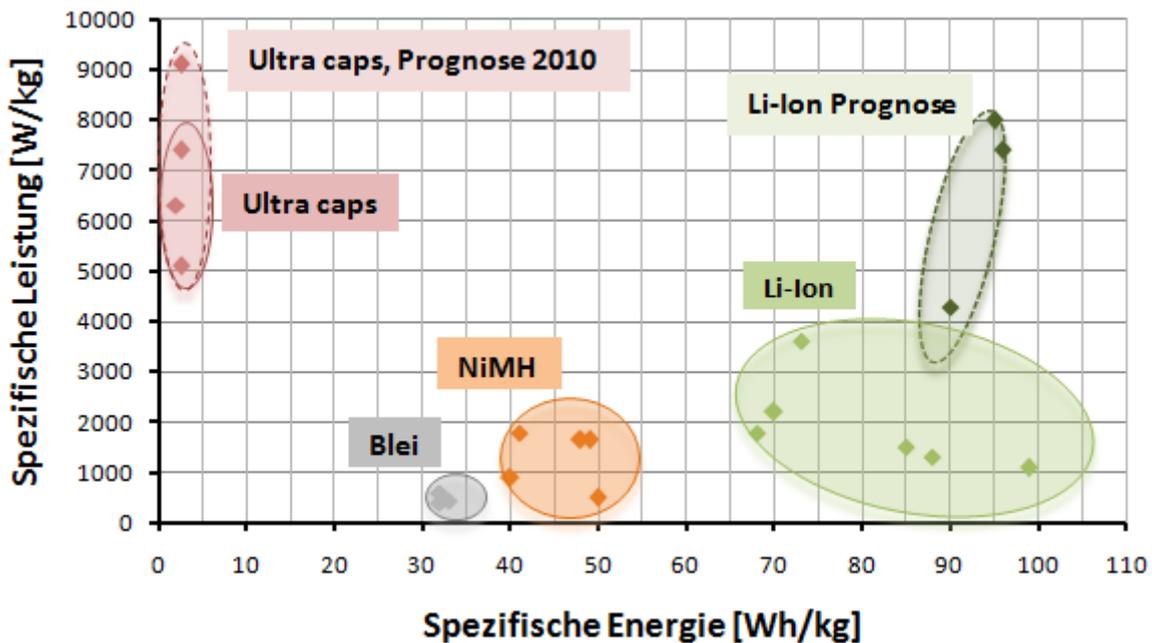


Abbildung 15: Die Gegenüberstellung verschiedener chemischer Speichersysteme im Ragone Diagramm

Beginnend mit den „Ultra Caps“ (Super Kondensatoren) erkennt man dass diese die höchste Spezifische Leistungsdichte haben, allerdings ist die Energiemenge die sie liefern können sehr gering, sie eignen sich für Anwendungen wo ein hoher Strom für kurze Zeit gefordert wird. Die Bleibatterie hat von allen Batterietypen die niedrigste spezifische Energiedichte diese ist im Bereich  $30Wh/kg$ . Die nächst höhere Energiedichte besitzt die Nickel-Metall-Hydrid Batterie, diese hat eine etwas höhere Streuung was die spezifische

Energiedichte betrifft diese reicht theoretisch von 40 bis 50 Wh/kg, auch die spezifische Leistungsdichte ist mit ungefähr 2000W/kg fast doppelt so groß wie bei der Bleibatterie. Die Lithium-Ionen Batterie ist absoluter Spitzenreiter was die spezifische Energiedichte betrifft diese hat einen Maximalwert von 100Wh/kg. Eine große Streuung ergibt sich bei der spezifischen Leistungsdichte, abhängig von verwendetem Material für die positive Elektrode kann mit der Lithium-Eisen-Phosphat Batterie eine spezifische Leistungsdichte von über 3000W/kg erreicht werden.

## 4 Die Entladung einer Batterie:

### 4.1 Einleitung

<sup>9</sup>Das natürliche Ziel einer Entladung einer Batterie ist es so viel Energie wie möglich aus der Zelle rauszuholen, die vorher rein geladen wurde. Der Lade-Wirkungsgrad ist ein Maß für dieses Verhältnis. Dieser gibt das Verhältnis zwischen entnehmbarer Energie zu geladener Energie. Wenn die Energiemenge in Amper-Stunden angegeben wird, spricht man vom **Ah-Wirkungsgrad**, auch Stromwirkungsgrad. Wenn die Energiemenge in Watt-Stunden angegeben wird, spricht man vom **Wh-Wirkungsgrad** oder energetischem Wirkungsgrad. Der reziproke Wert des Ah-Wirkungsgrad heißt Stromladefaktor:  $\lambda = \eta_{Ah}^{-1}$ . Die Batteriekapazität oder die entnehmbare Ladung, ist stark vom Entladestrom und der Batterietemperatur abhängig. Entladung mit einem großen Strom verursacht einen großen Anfangsspannungsabfall.

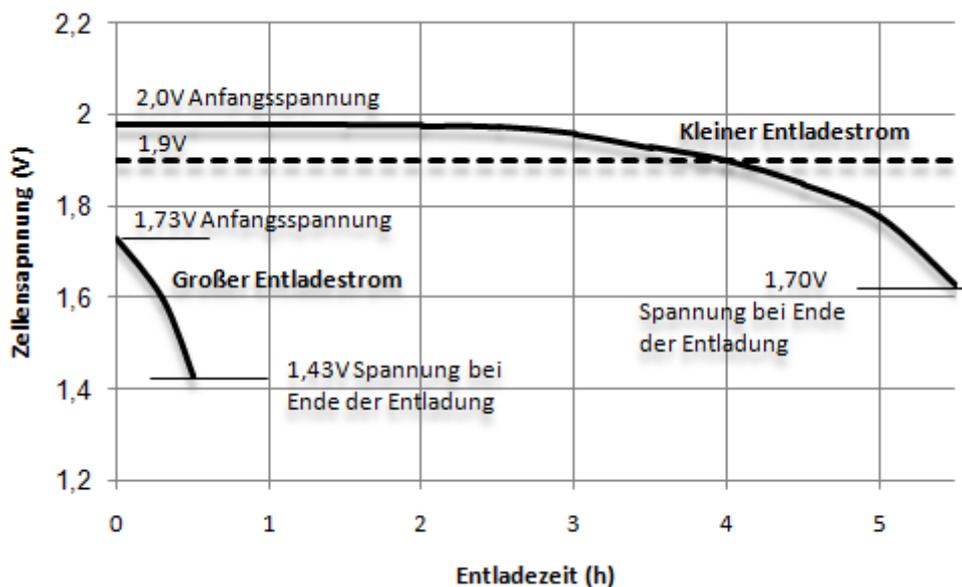


Abbildung 16: Die Entlade-Charakteristik für zwei unterschiedliche Stromstärken einer Blei-Zelle

Deshalb wird die Entladung, um möglichst viel Ladung zu entnehmen, zu einer höheren Entladeschlussspannung durchgeführt. Bei der Temperaturabhängigkeit der Batterie spielen zwei Effekte eine Rolle, einerseits steigen die ohmschen Widerstände bei

<sup>9</sup> Batterien und Ladekonzepte, Halaczek, Thaddäus Leonhard, Radecke, Hans Dieter

steigender Temperatur, andererseits steigt das Diffusionstempo der chemischen Reaktionen, was wiederum sich insgesamt positive auf die Batteriekapazität auswirkt. Bei sehr tiefen Temperaturen unter 0° C (-20° C) sinkt die Batteriekapazität ab.

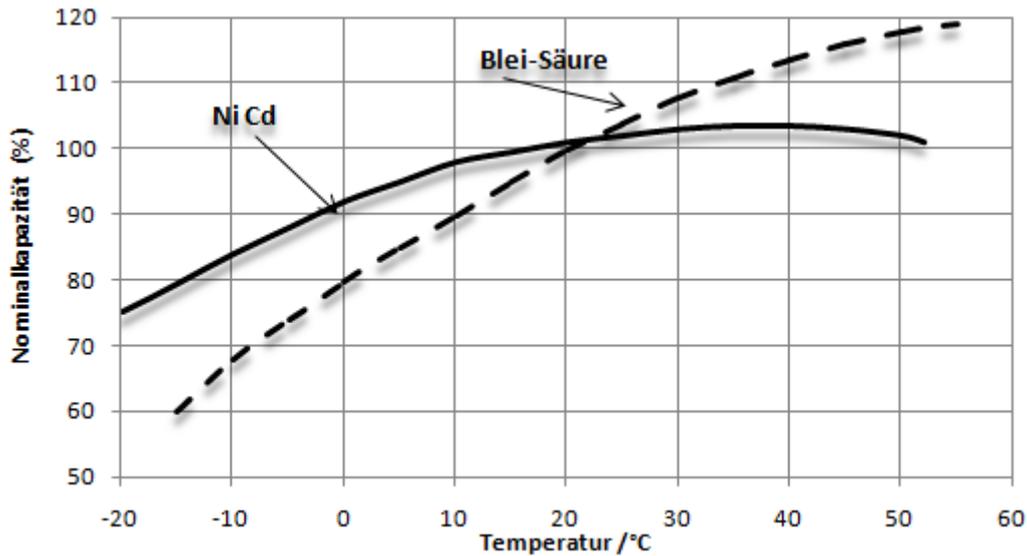
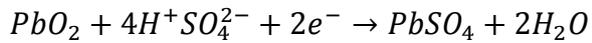


Abbildung 17: Die Thermische Abhängigkeit der Kapazität einer Blei und NiCd-Zelle

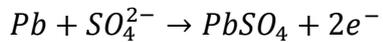
Die Temperaturabhängigkeit der Kapazität kann durch eine empirische Formel beschrieben werden:  $C_{20} = \frac{C}{1 + \delta(\theta - 20)}$ . Die Batteriekapazität berechnet sich aus dem entnommenen Strom und Entladezeit, bzw. die entnehmbare Ladung unter konstanter Entladestromstärke definiert die in der Zelle vorhandene Energiemenge:  $C = I * t|_{U_s[Ah]}$  I ist dabei die konstante Entladestromstärke [A] und t die Entladezeit bei der Entladeschlussspannung, C ist die Batteriekapazität in Ampere-Stunden [Ah].

## 4.2 Die Blei-Säure-Batterie

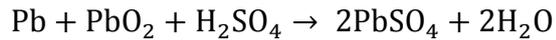
Positive Elektrode:



Negative Elektrode:



Gesamtreaktion:



Aus der Gleichung ist ersichtlich dass sich die beiden Elektroden während der Entladung in Bleisulfat verwandeln. Diese Reaktion verläuft unter Verbrauch von Schwefelsäure und Produktion von Wasser.

### 4.2.1 Entlade-Charakteristik

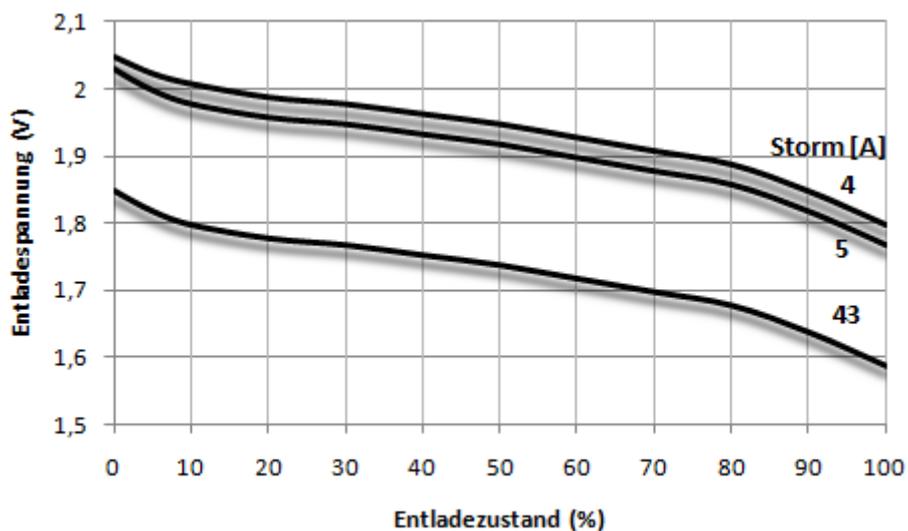


Abbildung 18: Die Entlade-Charakteristik eines Bleie-Akkus bei unterschiedlichen Belastungen

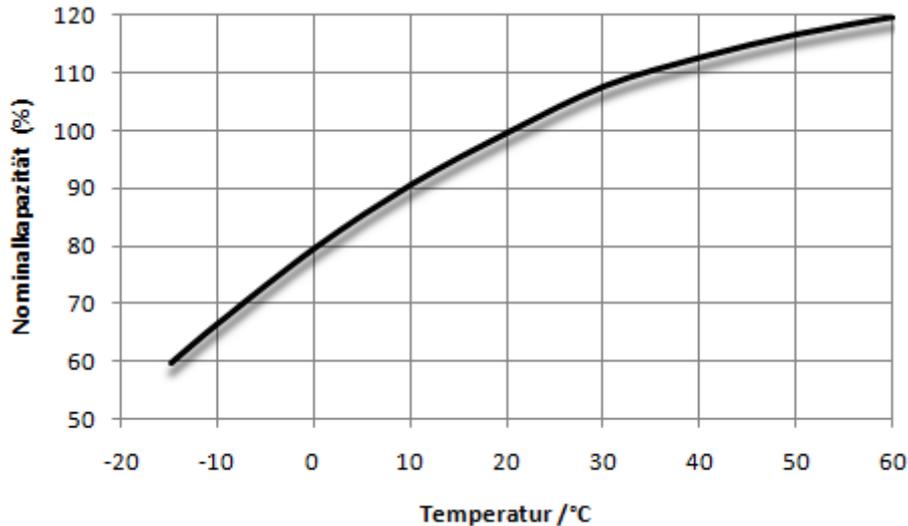


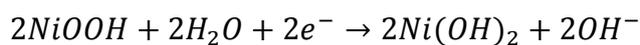
Abbildung 19: Die Abhängigkeit der Kapazität der Zelle von der Temperatur

### 4.2.2 Selbtsentladung

Die Selbstentladung ist ein thermodynamischer Effekt der die Zerlegung des Elektrolyten unter Emission von Wasserstoff bewirkt. Dieser Effekt lässt sich nicht vermeiden da er mit der Zellenreaktion zusammenhängt.

### 4.3 Die Nickel-Cadmium-Batterie

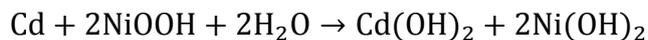
Positive Elektrode:



Negative Elektrode:



Gesamtreaktion:



Während der Lade/Entladezyklen verändern die aktiven Materialien ihre chemische Zusammensetzung ohne den physikalischen Zustand zu beeinflussen. Es ergeben sich also keine physikalischen Veränderungen am Zustand der Zelle, die vom Ladezustand abhängen würden, da sich die Konzentration des Elektrolyten fast nicht verändert. Da die aktiven Materialien von beiden Elektroden im Elektrolyt kaum löslich sind weder im

Aufgeladenen noch im Entladenen Zustand, führt das zu einem sehr stabilen Verhalten der Batterie über längere Zeiträume.

### 4.3.1 Entlade-Charakteristik

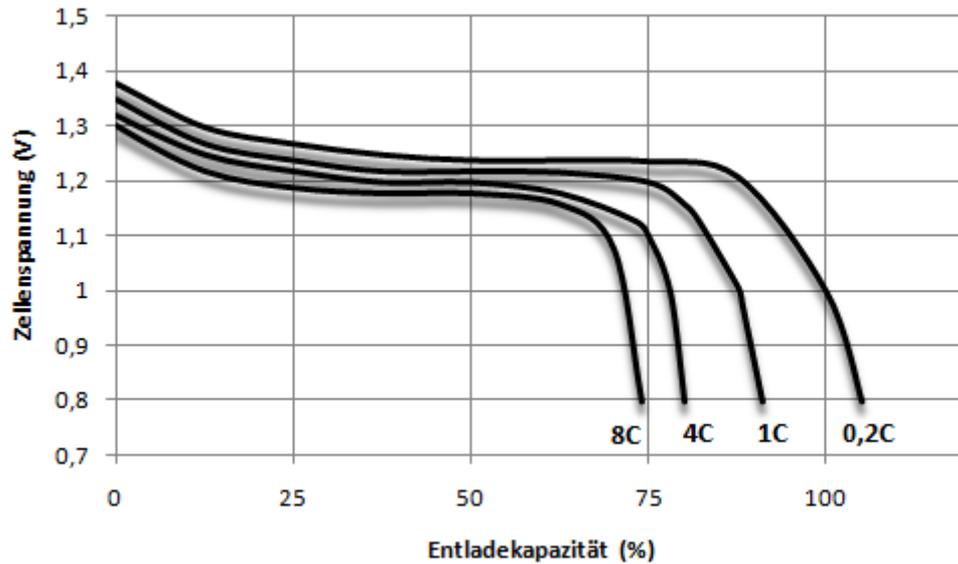


Abbildung 20: Die Entlade-Charakteristik unter verschiedenen Belastungen

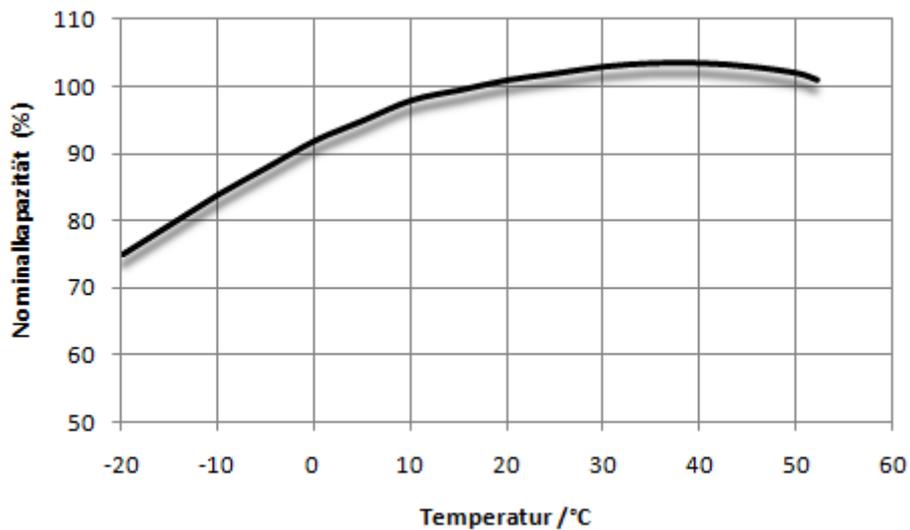


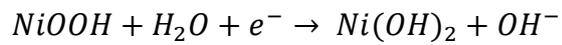
Abbildung 21: Die Temperaturabhängigkeit der Kapazität einer NiCd-Zelle

### 4.3.2 Selbstentladung

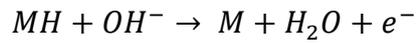
Die Selbstentladung tritt gleich nach der Ladung ein. Es gibt zwei Ursachen dafür. Einerseits kommt es zum Zerfall des Nickelhydrids unter Emission von Sauerstoff, und Selbstentladungsströme, ausgelöst durch Ionenmigration. Die Selbstentladung hängt von Parametern ab wie: Temperatur, Elektrodenabstand, Separatormaterial, Elektrolyt und dessen elektrische Leitfähigkeit. Bei niedrigeren Temperaturen sinken die Diffusionskoeffizienten, deshalb sinkt auch die Selbstentladungsrate. Während längere Lagerung verliert die NiCd-Zelle an Ladung und auch an Kapazität (Fähigkeit zur Ladungsaufnahme).

## 4.4 Nickel-Metall-Hydrid-Batterie

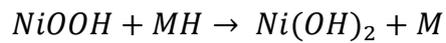
Positive Elektrode:



Negative Elektrode:



Gesamtreaktion:



### 4.4.1 Entlade-Charakteristik

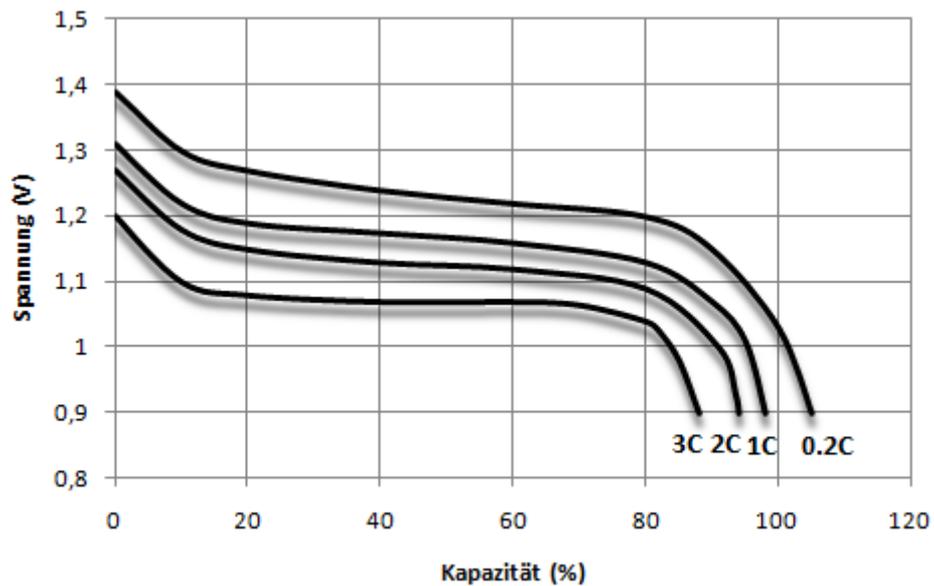


Abbildung 22: Die Entlade-Charakteristik unter verschiedenen Belastungen

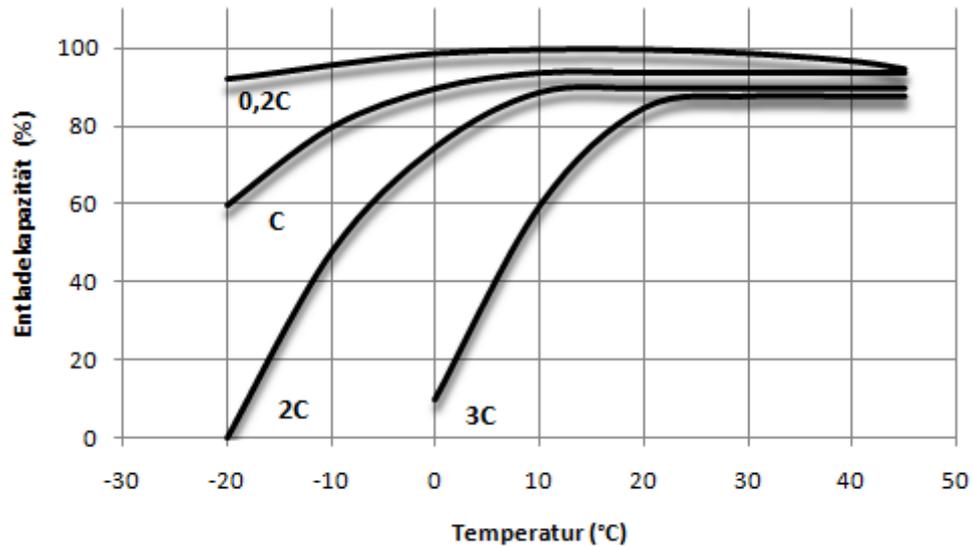


Abbildung 23: Die Abhängigkeit der Kapazität von der Temperatur und Belastung

#### 4.4.2 Selbstentladung

Die Selbstentladung hat zwei Ursachen. Die erste hat ihre Ursache in der Wechselwirkung des Wasserstoffes mit der positiven Elektrode, dies führt zur Entladung. Die zweite Ursache ist die chemische Aufspaltung des Elektrodenmaterials. Das führt zu Verlust der vorhandenen Ladung und beeinträchtigt die Kapazität, sprich die Fähigkeit Ladung zu speichern. Dieser Effekt ist allerdings reversibel, es müssen nur einige Ladezyklen durchgeführt werden um die Nominalkapazität wieder zu erreichen. Die Temperatur wirkt sich derart auf die Selbstentladung aus, das bei niedrigerer Temperatur sich die Selbstentladung verringert weil sich die Prozessgeschwindigkeit der chemischen Reaktion verlangsamt.

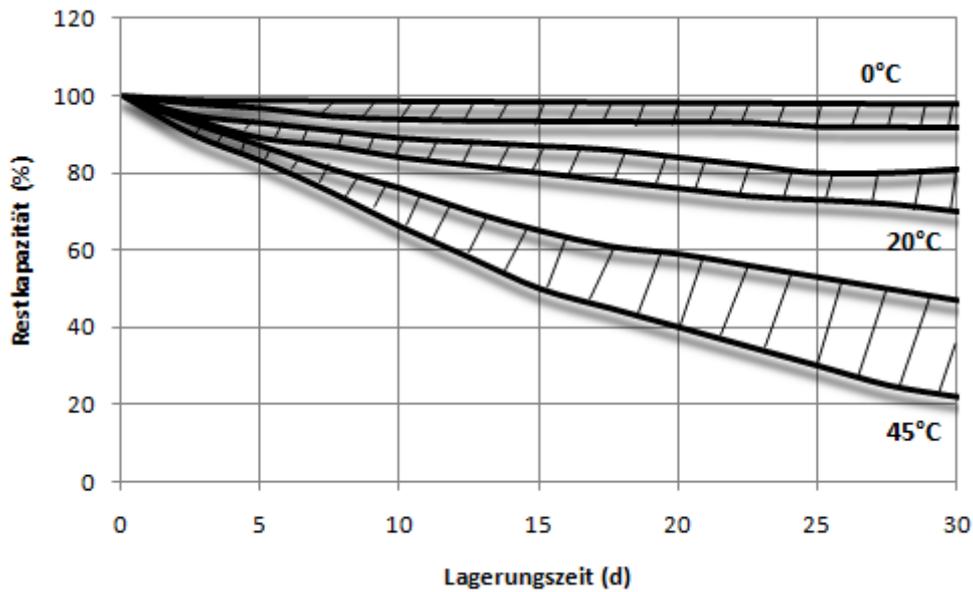
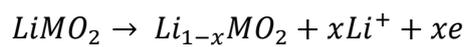


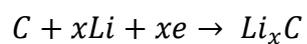
Abbildung 24: Die Selbstentladung der NiMH-Zelle

## 4.5 Die Lithium/Lithium-Ionen -Batterie

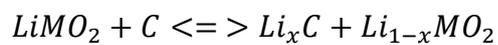
Positive Elektrode:



Negative Elektrode:



Gesamtreaktion:



### 4.5.1 Entlade-Charakteristik

Kohlenstoff/Lithium-Nickeloxid-Zelle,  $LiNiO_2$ :

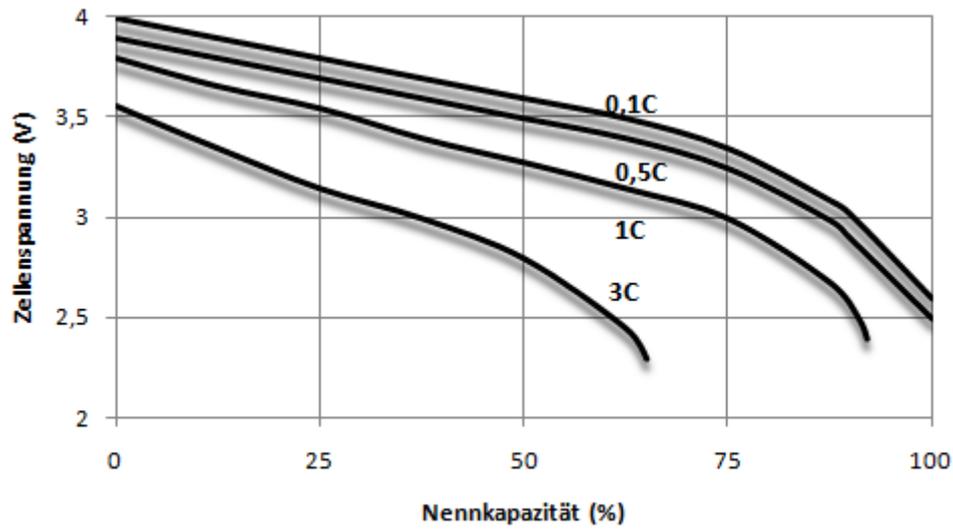


Abbildung 25: Die Entlade-Charakteristik einer Petrolkoks/ $LiNiO_2$ -Zelle

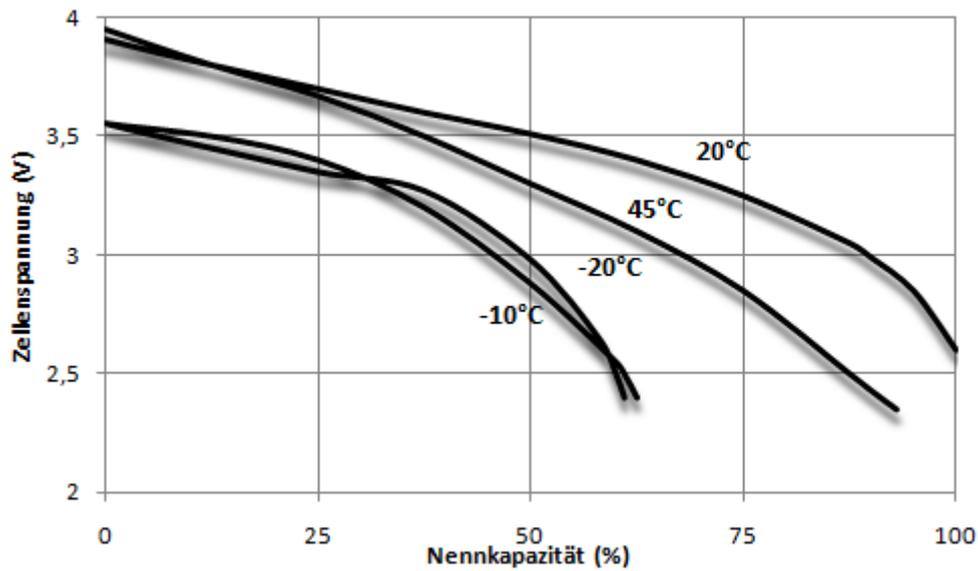


Abbildung 26: Die Selbstentlade-Charakteristik einer C/ $LiNiO_2$ -Zelle

## 4.5.2 Selbstentladung

$\text{LiNiO}_2$  hat eine niedrigere Selbstentladung als  $\text{LiCoO}_2$  und zwar beträgt diese 1 bis 3% pro Monat.

## 4.5.3 Entlade-Charakteristik

Kohlenstoff/Lithium-Manganoxid-Zelle mit festem Polymer-Elektrolyt,  $\text{LiMn}_2\text{O}_4$ :

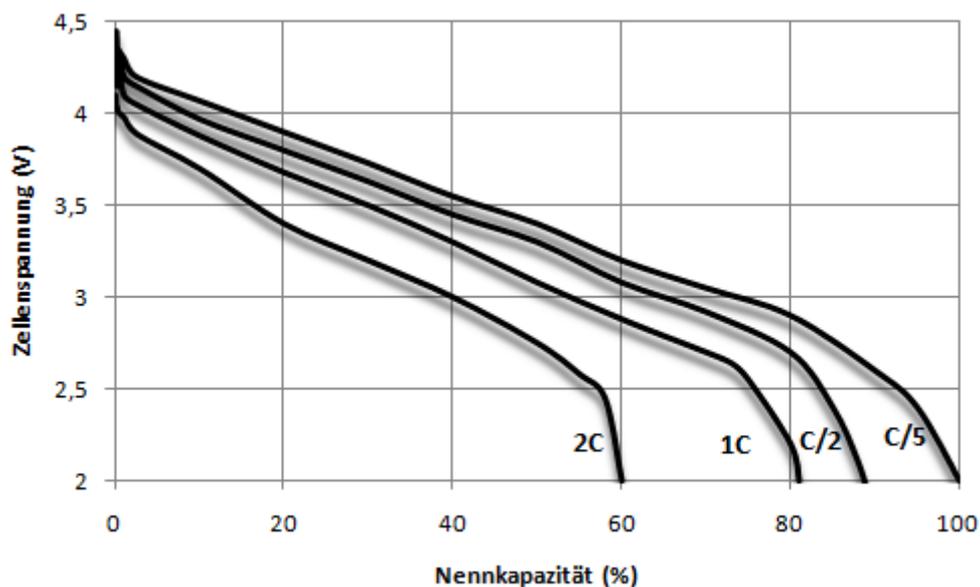


Abbildung 27: Die Entlade-Charakteristik der  $\text{LiMn}_2\text{O}_4$ -Zelle

## 4.5.4 Selbstentladung

Bei Zellen mit metallischem Lithium entwickelt das Lithium eine sogenannte Passivierungsschicht an der Grenzfläche zum Elektrolyten, diese Passivierungsschicht bewirkt dass sich der Selbstentladungsprozess verringert. Dadurch wird die Selbstentladungsrate solcher Zellen auf 1% pro Monat reduziert. Bei Lithium-Ionen-Zellen bildet sich diese Passivierungsschicht nicht aus, dadurch beträgt die Selbstentladungsrate dieser Zellen 10% pro Monat. Der Selbstentladungsprozess einer Lithium-Ionen-Zelle verläuft teilweise lokal d.h. ohne den Transfer der Ladungen zwischen den Elektroden.

Eine solche Reaktion wird als lokale Redox-Reaktion bezeichnet:  $Li_{1-x}MO_2 + yLi^+ + ye \rightarrow Li_{1-x} + yMO_2$ . Diese wird durch die Zerlegung des Elektrolyten verursacht.

#### 4.6 Zusammenfassung der einzelnen Batteriesysteme

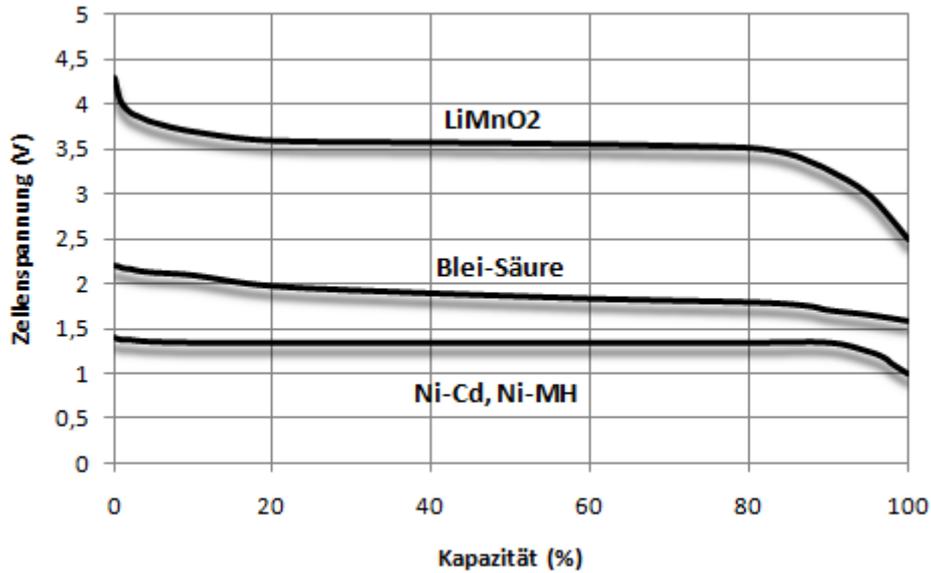


Abbildung 28: Die Entlade-Charakteristik der verschiedenen Speichersysteme

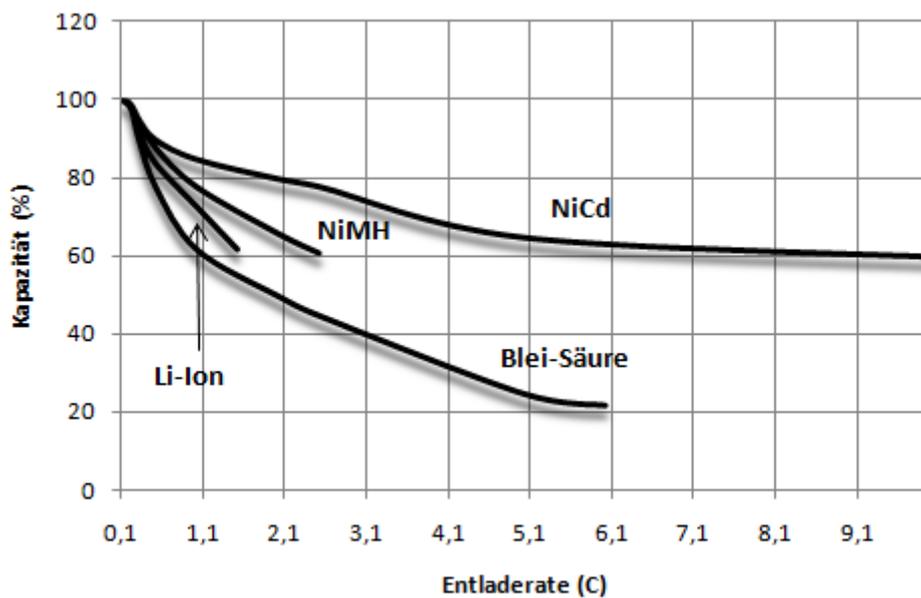


Abbildung 29: Die Gegenüberstellung der verschiedenen Speichersysteme

<sup>10</sup>Tabelle 1: Gegenüberstellung der verschiedenen Speichersysteme

System	Blei-Säure	NiCd	NiMH	Li-Ion
<b>Batterietyp</b>	Geschlos.	Geschlos.	Geschlos.	Geschlos.
<b>Anode</b>	Pb	Cd	MH	$\text{Li}_x\text{C}_6$
<b>Kathode</b>	$\text{PbO}_2$	NiOOH	NiOOH	$\text{LiCoO}_2$
<b>Nom. Zellenspannung</b>	2,0	1,2	1,2	3,7(3,5)
<b>Elektrolyt</b>	Flüssig, anorganisch	Flüssig, anorganisch	Flüssig, anorganisch	Flüssig,(Fest) organisch
<b>Arbeitsbereich [V]</b>	2,0-1,6	1,25-1,0	1,25-1,0	4,1-2,7
<b>Grenzwert der Entlade- schlußspannung</b>	1,6	0,8	0,9	2,5
<b>Max. Entladestrom [C]</b>	10	20	2-3	2(3)
<b>Arbeitstemperatur [°C]</b>	-40-+55	-40-+70	-20-+50	-20-+60
<b>Energiedichte</b>				
- gravim. Wh/kg	30(40)	40(60)	50	100(125)
- volum. Wh/l	90(100)	105(110)	175	260(250)
<b>Selbstentladung, bei 20°C</b>				
in %NK/Monat	8	15	25	10
<b>Zyklusfestigkeit bei 100%</b>				
Entladetiefe	200-300	300-700	300-600	500(1000)
<b>Nennentladerate [C]</b>	0,1-0,05	0,2	0,2	0,2

<sup>10</sup> Batteries in a portable world, Buchman Isidor

## 5 Die Ladung einer Batterie

Rein physikalisch bedeutet die Ladung einer Batterie nichts anderes als die Rückführung der bei den Arbeitszyklen entstehende Zellenreaktionsprodukte in ihre hochenergetischen Zustände durch Energiezufuhr aus einer externen Quelle“. Ein effektiver Aufladeprozess ist von essentieller Bedeutung für jede Akku-Batterie, da davon die Kapazität und die Brauchbarkeit der Batterie abhängig ist. Das Paradigma des Aufladeprozesses lautet dabei dass die Batterie vollständig aufgeladen werden muss, und dass die Überladung maximal eingeschränkt werden muss. Der Ladeprozess wird hauptsächlich durch die Parameter Ladestrom, Ladespannung und Temperatur der Zelle bestimmt. Beim Laden muss also Rücksicht genommen werden auf chemische Prozesse wie Diffusion und Kristallisation. Man unterscheidet folgende Ladeverfahren die sich entweder selbständig ergeben oder durch die Lademethode kompliziert zusammenschaltet werden.

### 1. Hauptladung

### 2. Ergänzungsladung

### 3. Erhaltungsladung

- a. Pufferladung
- b. Ausgleichsladung
- c. Formierungsladung

Hierbei sind die ersten beiden Lademethoden zwei Teile einer einzigen Lademethode. Dabei liefert der erste Teil der Zelle 75-90% ihrer Kapazität innerhalb weniger Minuten. Nach dieser Phase wird der hohe Strom abgeschaltet um eine Überladung zu verhindern und durch einen kleinen Strom ersetzt, diese Ladephase heißt Ergänzungsladung. Unmittelbar nach dem Ende des Ladeprozesses beginnt die Selbstentladung, um diese Selbstentladung zu verhindern wird die Batterie mit einem Strom geladen der das ausgleicht, diese Ladephase nennt man Erhaltungsladungsphase. Die Pufferladung wird für Batterien eingesetzt die ständig in Bereitschaft sind, d.h. das die Batterie ständig an einem Ladegerät hängt und gleichzeitig an einer Last. Bei mehreren in Serie geschalteten Zellen verwendet man die Ausgleichsladung, welche die unterschiedlichen Zellenspannungen ausgleichen soll, damit einige Zellen nicht Überladen werden. Werden Batterien länger gelagert ist eine Erfrischungsladung notwendig. Diese ist vor

allem von Lagertemperatur abhängig und wird bei manchen Systemen als verkürzte Hauptladung durchgeführt.

## 5.1 Ladung der Blei-Säure-Zellen

### 5.1.1 Konstantspannungs-Methode

Diese Methode als Hauptladung oder Pufferladung funktioniert optimal zwischen 2,3 und 2,4 V.

### 5.1.2 Konstantspannung ohne Strombegrenzung

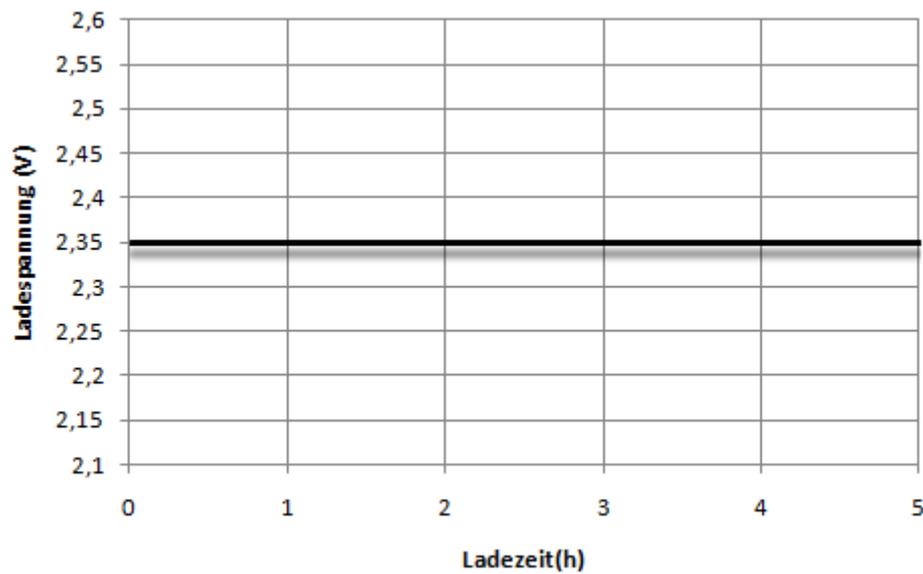


Abbildung 30: Charakteristiken der Konstantspannungs-Methode

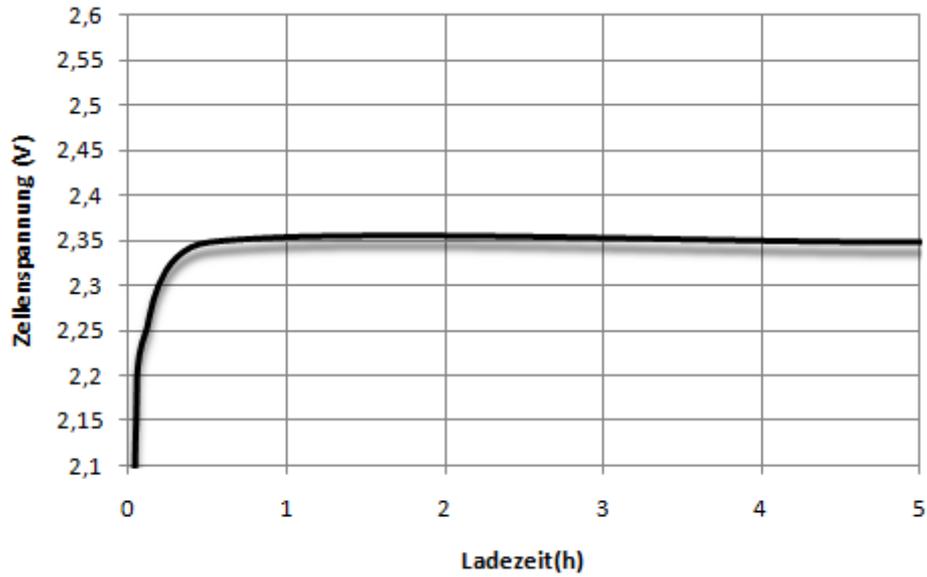


Abbildung 31: Charakteristiken der Konstantspannungs-Methode

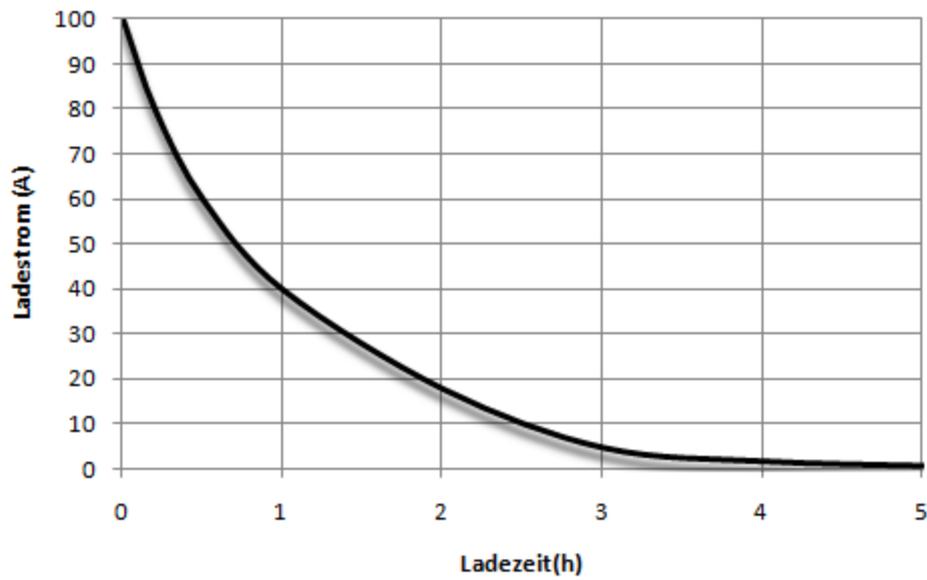


Abbildung 32: Charakteristiken der Konstantspannungs-Methode

Diese Verfahren ist sehr einfach, weil es nur eine Spannungsbegrenzung gibt. Die Anfangsgröße von Strom und Spannung müssen vor dem Ladeprozess in den Herstellerangaben kontrolliert werden und dürfen diese natürlich nicht übersteigen. Die Spannung der Batterie steigt während der Ladung langsam an bis der vorgegebene Wert erreicht ist. Der Strom bleibt am Anfang der Ladung konstant und beginnt dann nach einer Zeit asymptotisch auf einen kleineren Wert abzusinken.

### 5.1.3 Konstantspannung mit Strombegrenzung

Die Ladephasen sind gleich wie vorher beschrieben, allerdings wird der Höchstwert des Stromes begrenzt. Der Strom bleibt länger konstant, sinkt aber dann auch asymptotisch ab. Bei Pufferladung hängt die Batterie ständig am Ladegerät und der Last, es besteht eine permanente Überladungsgefahr. Der Pufferstrom muss minimiert werden, das erreicht man durch eine Spannungsstabilisierung mit Werten zwischen 2,3 und 2,45V.

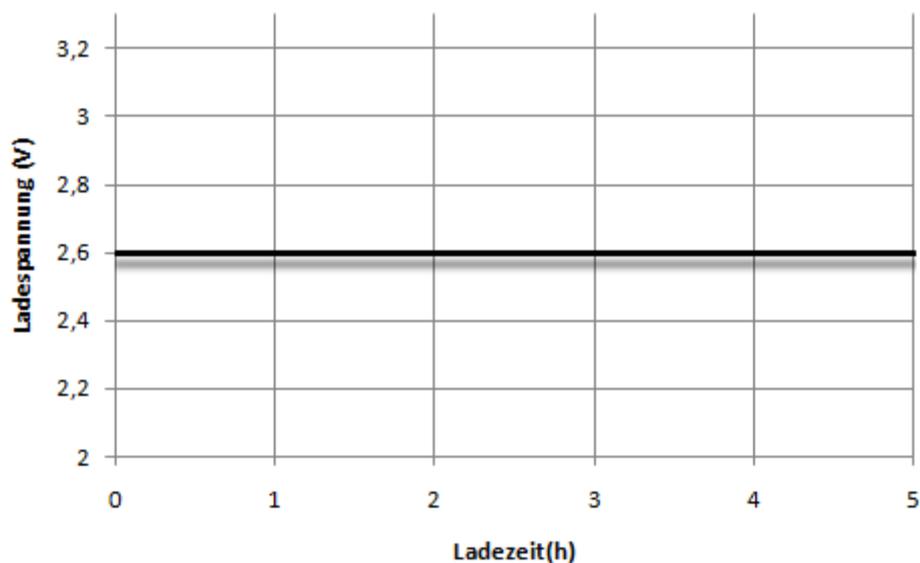


Abbildung 33: Charakteristiken der Konstantspannungs-Methode mit Strombegrenzung

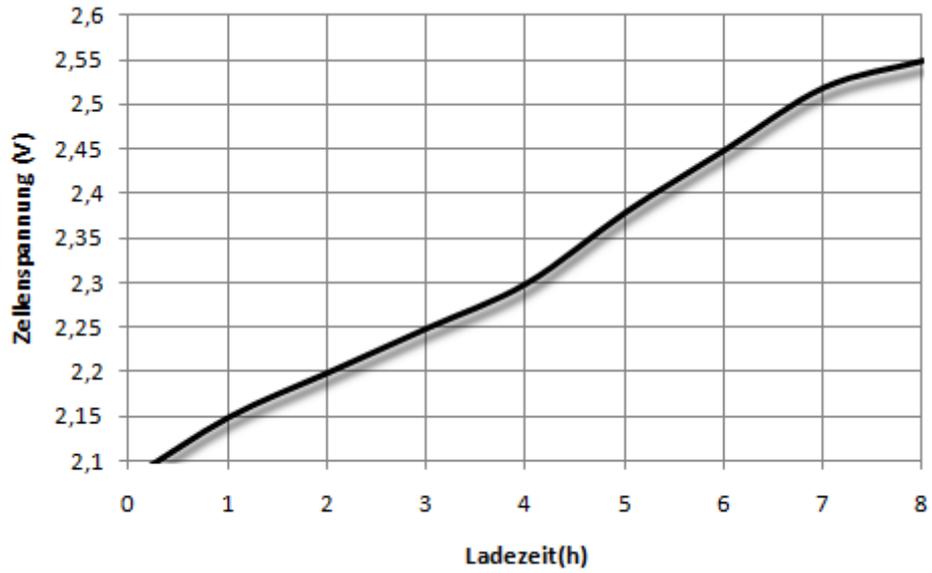


Abbildung 34: Charakteristiken der Konstantspannungs-Methode

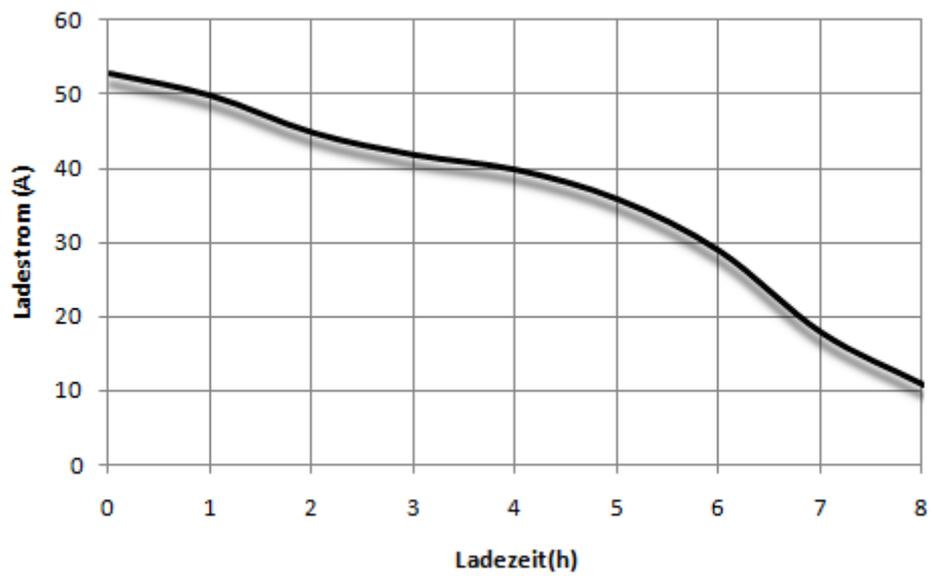


Abbildung 35: Charakteristiken der Konstantspannungs-Methode mit Strombegrenzung

## 5.1.4 Erhaltungsladung

Man kann die Konstantstromladung auch bequem für die Erhaltungsladung einsetzen, dabei wird der Ladestrom auf maximal  $C/500$  bei  $25^\circ\text{C}$  begrenzt.

## 5.1.5 Gemischte Methode

### 5.1.5.1 Impulsladung

Bei diesem Ladeverfahren ist ein Spannungsanstieg auf  $2,6\text{V}$  zulässig, wobei eine Strombegrenzung stattfinden muss. Zuerst kommt eine Konstantstromladung bis die Batterie  $2,35\text{V}$  erreicht, dann Konstantspannungsladung mit einer Spannung von  $2,35\text{V}$  bis der Strom  $0,015C$  erreicht, gefolgt von einer Stromimpulsladung mit einer Impulsamplitude von  $0,02C$  und verkürzter Impulslänge ohne Spannungsbegrenzung. Die Spannungsamplitude erreicht dann im Impuls  $2,6\text{V}$ . Diese letzte Phase dient als Ausgleichladung (dauert bis ca. 48 Stunden).

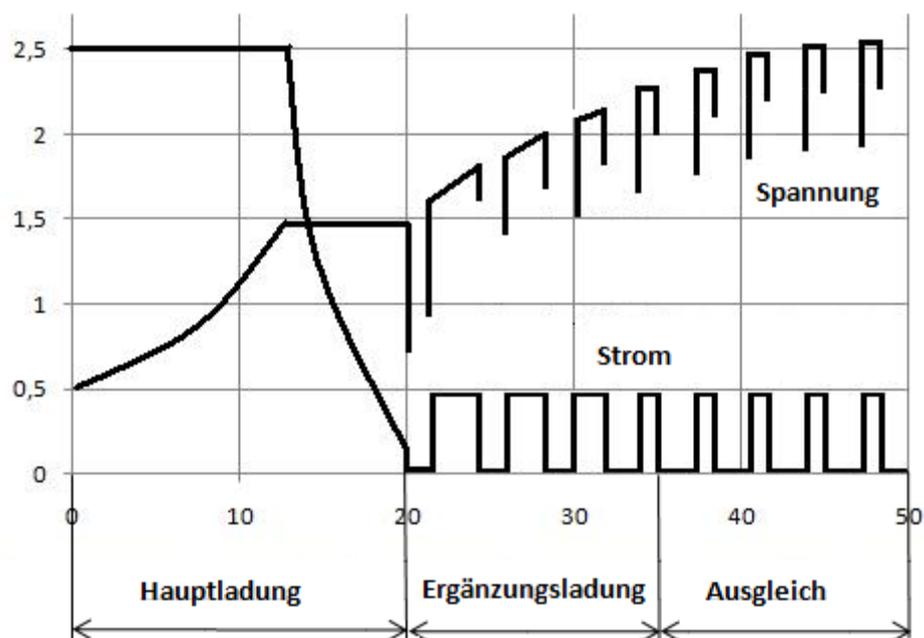


Abbildung 36: Impulsladung mit erhöhter Spannung

## 5.2 Lademethode von geschlossenen NiCd-Batterien

Die Überladung der NiCd-Batterie ist wesentlich unkritischer als bei Blei-Säure Systemen. Das liegt daran das der Gasabsorptionsmechanismus viel effektiver ist als bei Blei-Zellen.

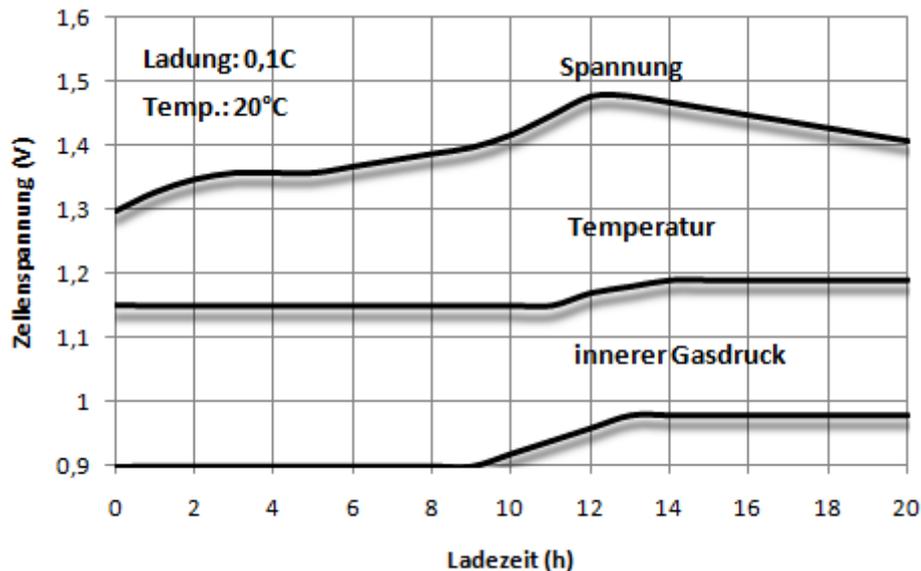


Abbildung 37: Ladekurve eines NiCd-Akkus bei einem Strom von 0,1C

Die ersten 60% der Kapazität werden schon bei niedrigen Spannungen erreicht und das unabhängig vom Ladestrom. Ein Teil des Stromes wird für den inneren Sauerstoff-Zyklus benötigt, dies führt zu einer Sauerstoffemission, diese verstärkt sich da die Stromakzeptanz der Elektrode während des weiteren Verlaufes der Ladung sinkt. Dadurch steigt der Druck in der Zelle an. Den Ladestrom kann man in einen akzeptierten Strom, der asymptotisch absinkt und die Batterie weiter auflädt, und einen nicht akzeptierten der für den internen Sauerstoff-Zyklus benötigt wird und die Gasung verursacht einteilen.

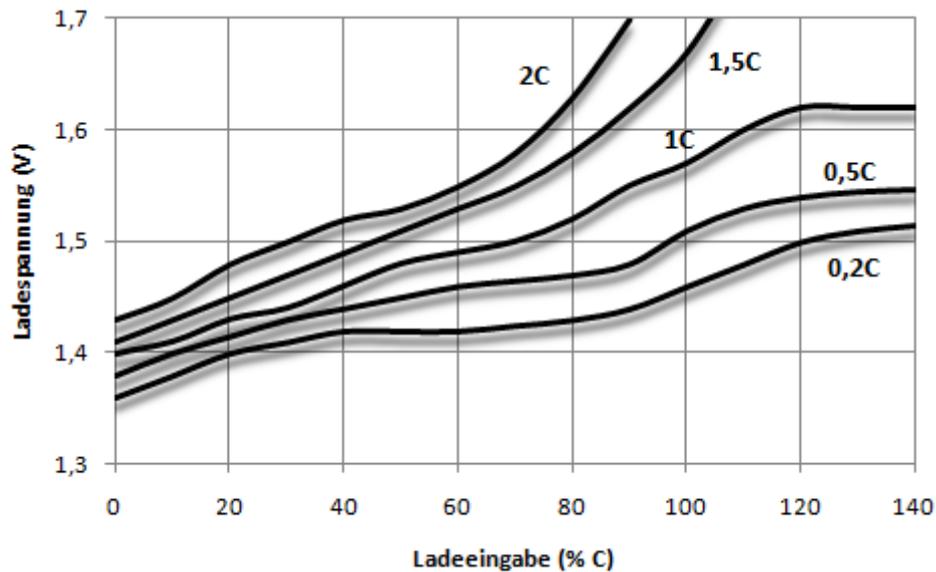


Abbildung 38: Ladecharakteristik bei verschiedenen Ladeströmen

### 5.2.1 Konstantstromladung

Das ist die empfohlene Lademethode für die NiCd und NiMH-Batterien. Der Batterieparameter der angepasst wird ist der Ladestrom. Wenn man die Batterie mit 0,1C lädt (14 bis 16 Stunden Ladung) ist keine zusätzliche Kontrolle der Temperatur notwendig (bei 0,3C ist eine Temperaturkontrolle notwendig).

### 5.2.2 Semikonstantstrom-Ladung

Bei dieser Methode wird der Strom durch einen Widerstand in Serie mit der Batterie begrenzt. Dabei muss der Widerstand den dreifachen Spannungsabfall aufweisen, als der Wert der Batteriespannung ist. Es wird empfohlen mit einem niedrigeren Strom als 0,1C zu laden.

### 5.2.3 Erhaltungsladung

Es handelt sich um eine gewöhnliche Konstantstrom-Ladung mit einem Strom im Bereich von 0,05 bis 0,02C der die Selbstentladung ausgleicht.

### 5.3 Lademethode von Nickel-Metall-Hydrid -Batterien

NiMH-Systeme und NiCd-Systeme sind sehr ähnlich. Beide haben ähnliche Leerlaufspannungen von 1,2 bis 1,35 V und ähnliche Spannungen bei Belastung von 1,2V und ähnliche Entladeschlussspannungen von 1V. Trotzdem dürfen Ladegeräte für NiCd-Batterien nicht für NiMH-Batterien eingesetzt werden, weil es beim Ladeverhalten große Unterschiede gibt. Der wichtigste ist das NiMH-Batterien sehr empfindlich gegenüber Überladung sind, weil die Zellen mit größerem Druck als NiCd-Systeme arbeiten. Das NiMH-System hat auch eine geringere Toleranz gegenüber der Erhaltungsladung (geringe Stromstärke von C/300).

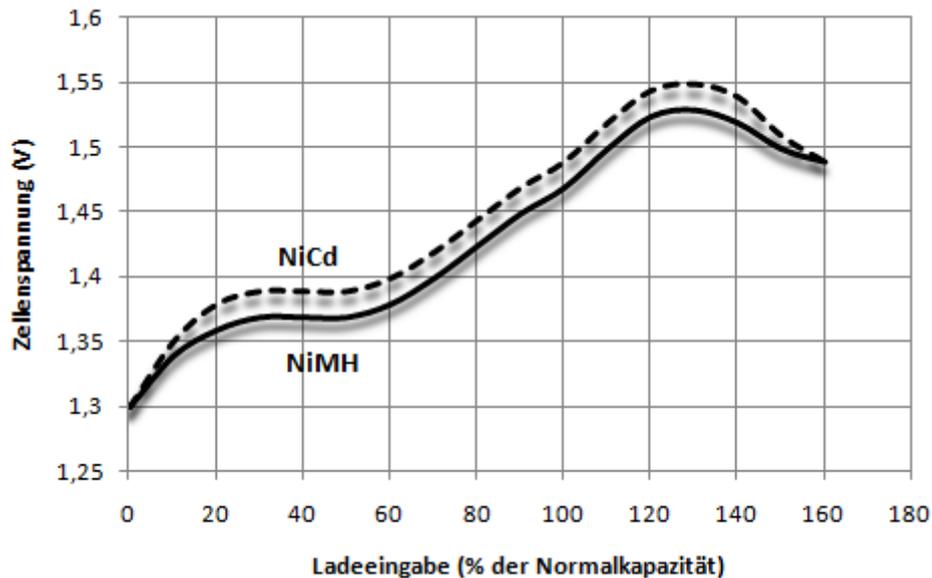


Abbildung 39: Vergleich der Ladekurven der NiCd und NiMH Batterien

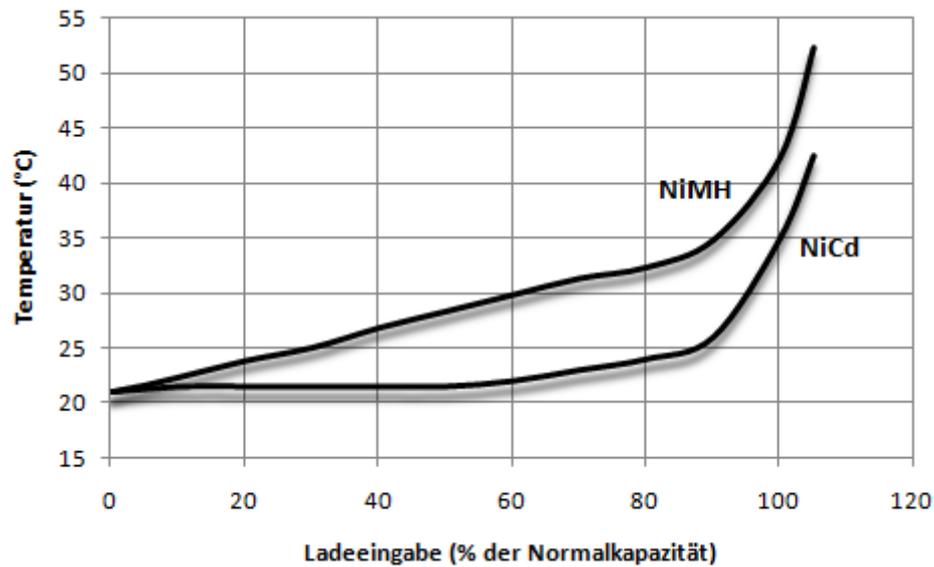


Abbildung 40: Vergleich der internen Temperatur während des Ladens einer NiCd und NiMH-Zelle

### 5.3.1 Semikonstantstrom

Entspricht der gleichen Lademethode wie für das NiCd-System. Der Stromwert entspricht anfänglich 0,1C. Beim NiMH-System liegt die Ladedauer bei 12 bis 15 Stunden.

### 5.3.2 Konstantstrom

Entspricht gleich wie beim NiCd-System. Der Ladestrom liegt im Bereich zwischen 0,1C und 0,3C. Eine Ladeschlusskontrolle ist unbedingt notwendig.

Bild einfügen:

### 5.3.3 Abschaltkriterien bei NiMH-Systemen

Alle NiMH-Systemen und schnellladefähige NiCd-Systeme müssen wegen der Sicherheit über Abschaltkriterien terminiert werden. Dabei genügt oft nicht nur ein Abschaltkriterium sondern es muss z.B. neben der Temperatur auch über das Zeitkriterium terminierbar sein. Denn manchmal sind unter bestimmten Bedingungen bestimmte Kriterien nicht zutreffend, z.B. kann es bei der NiCd-Batterie zu einem Pseudospannungsabfall  $U_s$  kommen, und die Ladung wird nach 2-10min abgebrochen.

#### 5.3.3.1 Einige Abschaltkriterien für NiMH und NiCd-Systeme

- Timer: Funktioniert nach dem Ladungszählerprinzip

$$\text{Ladezeit} * \text{Ladestrom} = \text{Ladefaktor} * \text{Nennkapazität}$$

- Spannungsabfall:  $-\Delta V$ , bei NiCd beträgt dieser 10-20mV und bei NiMH beträgt er 5-10mV pro Zelle.
- Spannungsplateau: (zero  $\Delta V$ ), Beim Erreichen der Steigung 0 der Spannung, wird die Ladung abgebrochen.

Temperatur: Übersteigt die Zellentemperatur einen bestimmten Wert ist das Kriterium erreicht. Bei NiCd 45-70°C bei NiMH 50-55°C.

- Temperaturänderung: Es wird der Temperaturanstieg von der Anfangstemperatur des Ladens an gemessen.

Temperaturanstieg:  $(\frac{\Delta T}{\Delta t})$ , Es wird die Geschwindigkeit des Temperaturanstieges gemessen. Dieser Parameter reagiert sehr empfindlich auf Überladung, weshalb diese Methode für das Schnellladeverfahren von Batterien eingesetzt wird. Für NiMH und NiCd liegen die vorgegebenen Werte bei 1°C/min.

## 5.4 Lademethoden von Lithium-Ionen-Systemen

Aufgrund der Tatsache dass hier Lithium-Ionen anstatt des metallischen Lithiums verwendet wird sind hier einige Gefahren eliminiert. Allerdings ist das System weiterhin sehr empfindlich. Da die meisten Systeme mit organischem Elektrolyt arbeiten, kann eine Überhitzung der Zelle zu einem Brand führen. Das System ist sehr empfindlich gegenüber Tiefentladung und Überladung, beides führt zu einer erheblichen Verkürzung der Lebensdauer der Zelle, oder sogar zur Beschädigung.

Tabelle 2: Gegenüberstellung der Lebensdauer bei verschiedener Entladeschlussspannung

Ladespannung	Kapazität	Lebensdauer
4,1V	1200mAh	800 Zyklen
4,2	1300mAh	400 Zyklen

Man sieht anhand der Tabelle wie empfindlich das System gegen die Spannung ist. Deswegen kommt hier nur die Lademethode mit Konstantspannung und Strombegrenzung in Frage. Aufgrund der Problematik der Lithium-Ionen –Systeme kümmert sich der Hersteller selbst um die Sicherheit, indem der eine Schutzschaltung einbaut, welche das Batterie-Pack vor Überladung und Tiefentladung schützt. Ladeverfahren für Lithium-Ionen Zellen sieht folgendermaßen aus: Die Lademethode ist Konstantspannung mit Strombegrenzung. Die Temperatur ist auf 40°C begrenzt. Die Ladespannung beträgt ziemlich genau 4,1V und darf maximal um +/-0,05V abweichen. Der Ladestrom beträgt typischerweise 0,5C und bei Schnelllade-Batterien 1-2C. Vor Beginn der Ladung wird ein Batteriespannungstest durchgeführt, wenn die Spannung außerhalb bestimmter Grenzen ist, gibt es eine Fehlermeldung. Liegt die Entladeschlussspannung unter dem zulässigem Bereich, wird zuerst mit einer sehr kleinem Strom von z.B. 10mA geladen. Wenn die Spannung allerdings größer oder gleich der Entladeschlussspannung ist, dann wird mit Konstantspannung geladen mit Stromkontrolle.

Es wird allgemein ein Timer als Zusatzkontrolle eingesetzt. Es sind Pufferladebetrieb und Erhaltungsladung möglich.

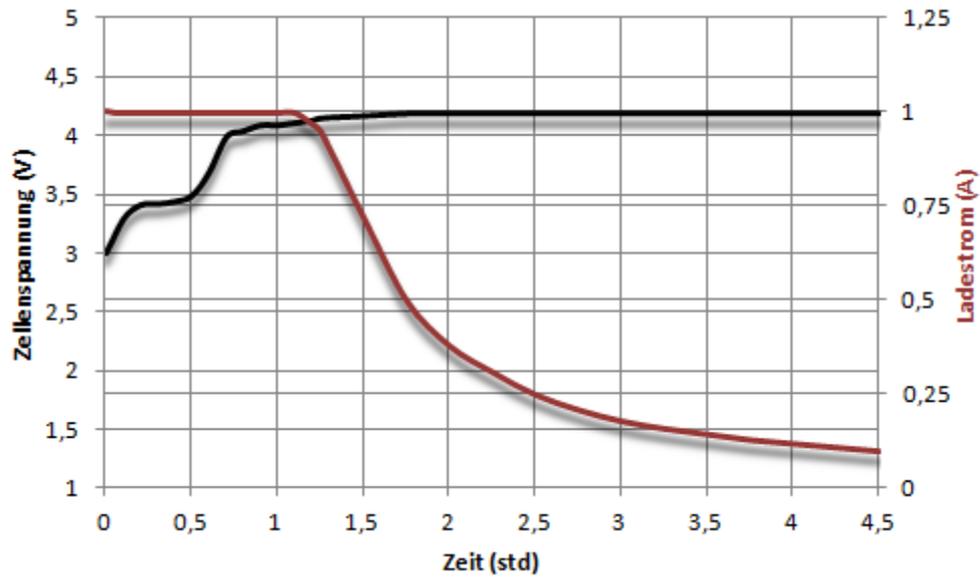


Abbildung 41: Ladekurve von Strom und Spannung einer Lithium-Zelle

## 6 Das Batterie Management System

<sup>11</sup>Beim „Batterie Management System“ (BMS) handelt es sich um ein Steuergerät zur Überwachung und Regelung eines nachladbaren Batteriesystems. Ein BMS in einem Elektrofahrzeug überwacht dabei die Zellenspannung, die Pack-Stromstärke und die Temperatur. Das BMS ist dabei zusammengesetzt aus zwei Komponenten. Die erste Komponente ist der Master-Controller des Systems. Diese behandelt auch den Ladezustand, Gesundheitszustand und „fault“-Management Funktionen des Akkus. Die zweite Komponente ist die Daten-Monitoring-Funktion, welche Informationen sammelt über die Bedingungen der Batterie-Zellen, nimmt Messungen von Spannung, Strom und Temperatur vor. Der Slave-Monitor-Controller ist direkt verbunden mit dem Batterie „Stack“ (Batteriemodul). Der Slave-Monitor-Controller misst die Zellenspannung und die Modul Temperatur, und übernimmt die Kontrolle der Zell-Ladungsausgleich Funktion deren Aufgabe darin besteht alle Zellen auf den gleichen Ladungszustand hin zu bringen. Die Slave-Monitor-Controller dienen auch zur Bekanntgabe von Alarmbedingungen wie Zellen Über/Unterspannung, offene Stromkreise und im Falle von Li+ Batterien-Chemien übertemperatur Situationen. Der Slave-Monitor-Controller wird vom Master-Controller verwaltet. Der Master-Controller orchestriert alle Daten-Akquisitionen und Zellen-Balancierungs Aufgaben in den Slaves. Der Master-Controller misst auch den Pack-Strom zusammen mit der Spannung, somit kann der Gesundheitszustand des Akkus bestimmt werden kann. Der Pack-Strom wird dabei über einen „low-value“ shunt Widerstand oder einen Hall-Sensor gemessen.

### 6.1 Batterie Pack Architektur

Akku-Packs sind in einer modularen Art und Weise aufgebaut, um mehrere verschiedene Konfigurationen und eine schnelle und flexible Montage zu ermöglichen. Durch die Optimierung des bau und reparatur Prozesses werden die Kosten gesenkt. Die Definition eines Akku-Packs ist ein System bestehend aus einem oder mehreren Batteriemodulen,

---

<sup>11</sup> MAXIM MAX11068 Datenblatt

angeschlossen entweder in einer Reihen oder Matrix-Konfiguration um eine Hochspannungsquelle zu erzeugen. Transport oder Batterie-Backup-System Anwendungen verwenden in der Regel viele in Reihe geschaltete Batteriemodule, um Spannung von mehreren hundert Volt zu erzeugen. Diese Spannung kann dann invertiert und transformiert werden um Sie für die Spannungsebene der entsprechenden Last anzupassen. Ein Batterie-Modul besteht aus einer Serie von Zellen, konfiguriert als ein Subsystem, das mit anderen Modulen kombiniert werden kann zum Aufbau eines Hochspannungs-Packs. Beim MAX11068 wird die minimale Zellenanzahl begrenzt durch die Erforderlichen 6.0V für den Regler, während die maximale Zellenanzahl 12 beträgt. Die 6.0V Mindestanforderung grenzt die Konfiguration auf mindestens 2 Lithium-Ionen und sechs NiMH Zellen ein. Akku-Packs in Transport-Anwendungen können aus verschiedenen Akku-Technologien aufgebaut sein (Lithium-Ionen, NiMH oder Blei-Säure), darüber hinaus beinhalten Sie auch ein „Battery Management System“ BMS und mehrere Sicherheits-Features. In Hybrid-Elektrofahrzeugen, Plug-in Hybrid-Elektrofahrzeugen, Elektrofahrzeuge oder Brennstoffzellen-Fahrzeuge kann die Zellenanzahl von 36 bis 200 vom Typ Lithium-Ionen und 200 bis 500 Zellen vom Typ NiMH betragen. Es gibt grundsätzlich zwei Akku-Pack-Management Architekturen, die mit dem MAX11068 realisiert werden können:

- „Distributed“ Modul Kommunikation
- „SMBus-laddered“ Modul Kommunikation

Ein „Distributed“ Modul-System setzt eine Punkt-zu-Punkt-Verbindung ein, dabei geht von jedem Batteriemodul eine Verbindung zurück zu einem Master-Mikrocontroller. Da die Batterie Module von einem Hochspannungs-Batterie-Stack aus arbeiten, muss eine galvanische Isolierung bei der Kommunikation mit dem Master-Mikrocontroller vorgenommen werden. Ein „SMBus-laddered“ Modul-System setzt einen seriellen Kommunikations-Bus ein der durch jedes Batterie Modul reist und an einem Punkt erfolgt dann der Einstieg in das System durch den Master-Mikrocontroller. Das „SMBus-Ladder“-Verfahren senkt die Kosten und erfordert eine einzige galvanische Trennung zwischen den Hochspannungs-Batterie-Stack und dem Master-Mikrocontroller.

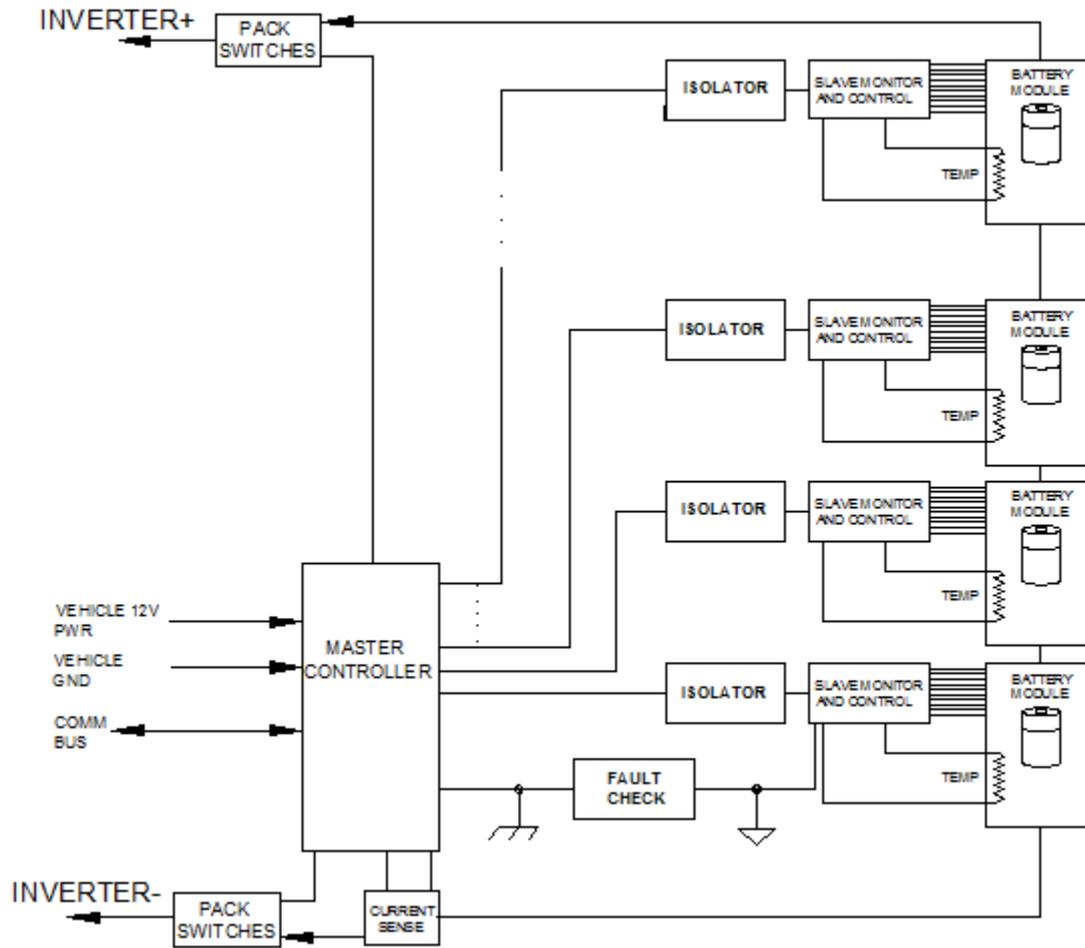


Abbildung 42: Verteiltes Kommunikationsmodell einer Battery-Pack Architektur

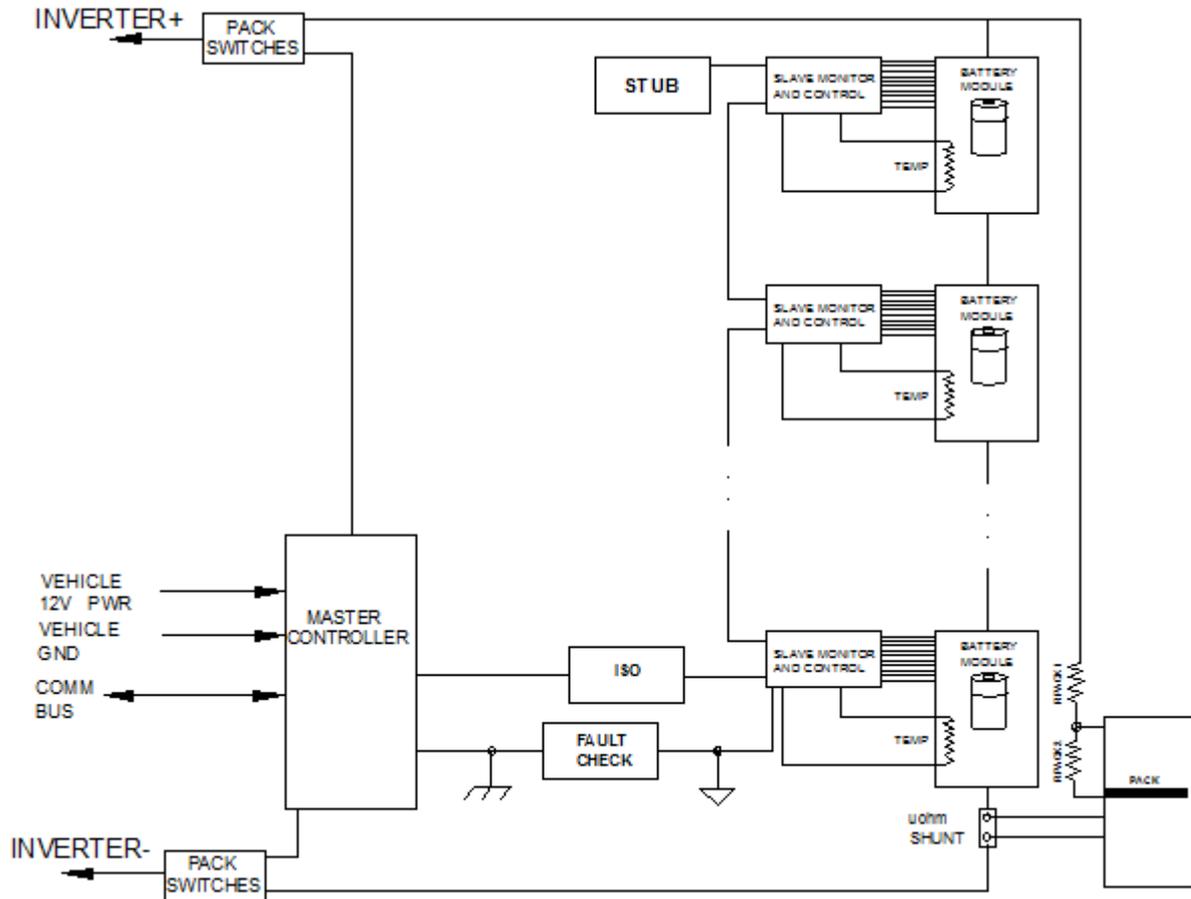


Abbildung 43: SMBus Kommunikationsmodell einer Battery-Pack Architektur

## 7 Balancierung

<sup>12</sup>Balancierung dient zum ausgleichen von Zellenspannungen von Lithium Akkupacks die mindestens 2 Zellen in Serie geschaltet haben, da beim Ladevorgang nicht automatisch gewährleistet werden kann das alle Zellen gleichmäßig aufgeladen werden. Mit der Balancierung der Lithium Akkupacks Zellen wird ein positiver Einfluss auf die Lebensdauer, die Energiemenge und die Leistung der Zellen genommen. Die Balancierung der Zellen kann dabei „nur“ während des Ladevorganges oder auch während des Betriebes vorgenommen werden.

Es gibt grundsätzlich zwei Arten von Balancierung: Aktives Balancing und Passives Balancing.

### 7.1.1 Passives Balancieren

Passives Balancing bedeutet, dass die Zelle die eine zu hohe Spannung aufweist über einen Widerstand entladen wird, und so auf die anderen Zellen abgeglichen wird.

Dabei gibt es zwei Verfahren:

- **Balancieren über Spannungsteiler**

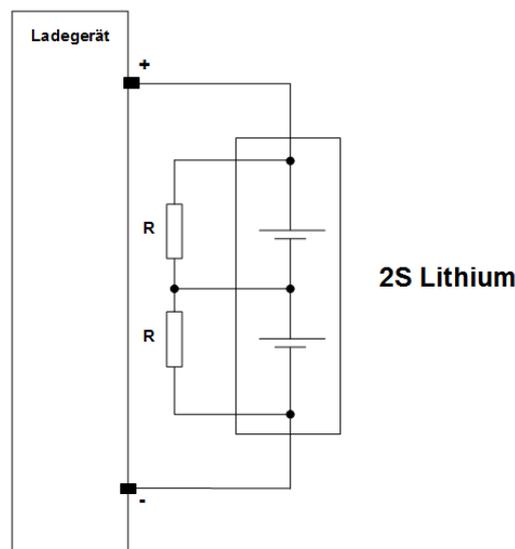


Abbildung 44: Balancierung mit ohmschen Widerständen

<sup>12</sup> [http:// www.rc-drift-szene.de](http://www.rc-drift-szene.de)

Das ist die einfachste Form eine Balancierung durchzuführen, jedoch ist hier das Problem das man wesentlich mehr Leistung reinsteckt, als im Endeffekt in den Akku geladen wird, da durchgehend Leistung über die Widerstände „verbrannt“ wird.

- **Balancieren durch gezieltes ein/aus schalten eines Schalters**

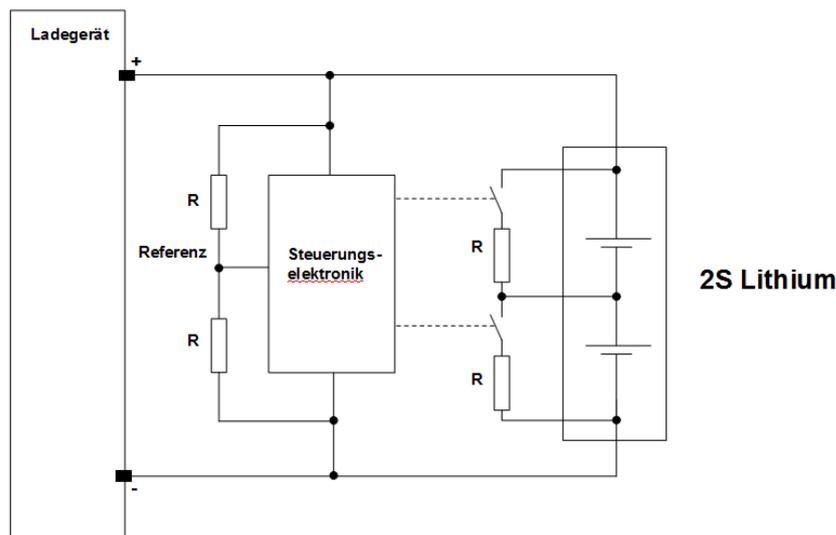


Abbildung 45: Balancierung mit Schalter

Bei dieser Form der Balancierung wird das Entladen der Zellen gezielt gesteuert, es werden nur jene Zellen über den Widerstand entladen die eine höher Spannung gegenüber den anderen aufweisen. Es wird die gesamte Spannung des Akkupacks über einen Spannungsteiler halbiert (wenn 3S dann gedrittelt, usw.) Anschließend werden die Spannungen der einzelnen Zellen mit dieser halbierten Spannung (Referenzspannung) verglichen. Wenn eine Spannung der Zelle höher ist, wird der Schalter geschlossen und die Zellen entlädt sich bis zur Referenzspannung, ist die Zellenspannung kleiner passiert nichts. Die Höhe des Balancing –Stromes ist dabei eine entscheidende Größe, dieser bewegt sich im Bereich von 100mA bis 300mA. Als Schalter werden MOSFET's verwendet.

## 7.1.2 Aktives Balancieren

Beim Aktiven-Balancing wird die Energie nicht über einen Widerstand nutzlos „verbrannt“ sondern es wird versucht die überflüssige Energie die aus einer Zelle entnommen wurde in eine andere Zelle reinzustecken.

Dabei gibt es zwei Verfahren:

- **Geschaltete Kapazität**

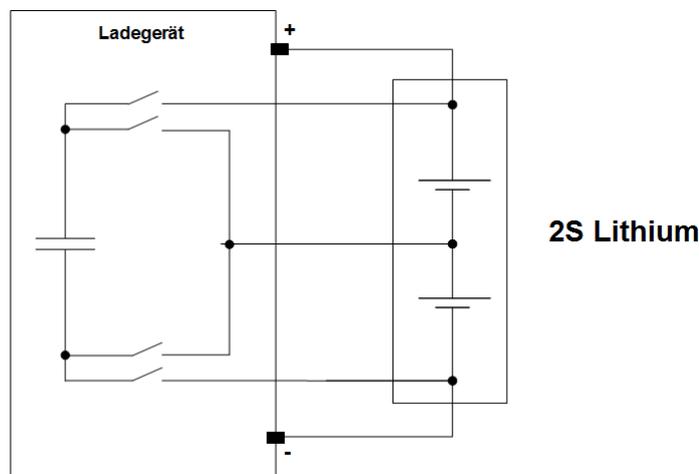


Abbildung 46: Balancierung mit geschalteter Kapazität

Bei dieser Form der Balancierung wird ein Kondensator über die Zelle mit der höheren Spannung aufgeladen. Anschließend wird der Kondensator auf die Zelle mit der niedrigsten Spannung geschaltet und diese Zelle wird dann aufgeladen. Bei diesem Verfahren ist entscheidend der Lade/Entladestrom des Kondensator bzw. Lithium-Zelle und wie lange man den Kondensator bzw. die Lithium-Zelle lädt/entlädet. Dieses Balancierungs-Verfahren ist nur bedingt besser, da es am effizientesten funktioniert je mehr sich die Spannungen der Einzelzellen unterscheiden, weil da der maximale Energiefluss gewährleistet wird. Dies lässt sich gut anhand der folgenden SOC („State of the charge“) Kennlinie für Lithium-Zellen erkennen. Man sieht das bei diesem Balancierungs-Verfahren im Bereich wo die Kennlinie sehr flach ist, nur wenig Energie transportiert werden kann.

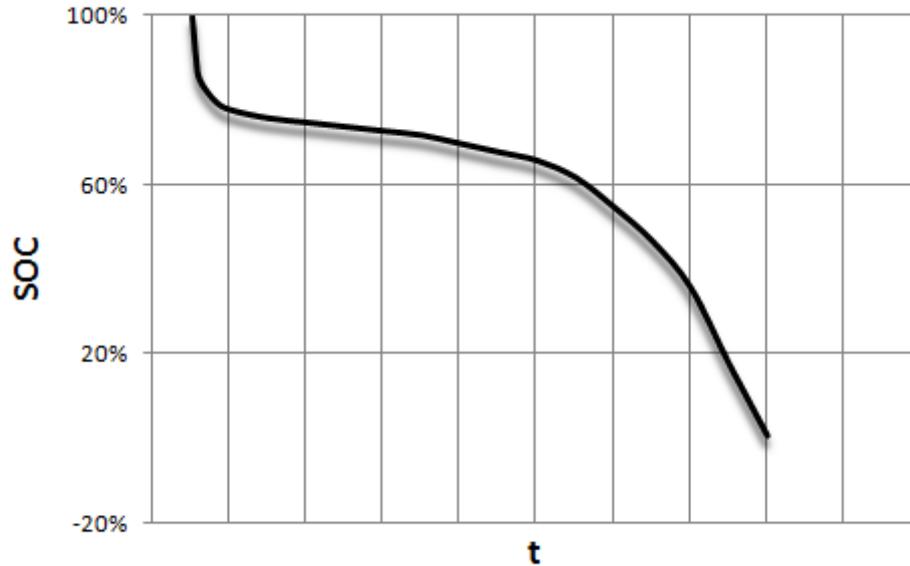


Abbildung 47: "State of charge"-Kennlinie bei Balancierung mit geschalteter Kapazität

- **Geschaltete Induktivität**

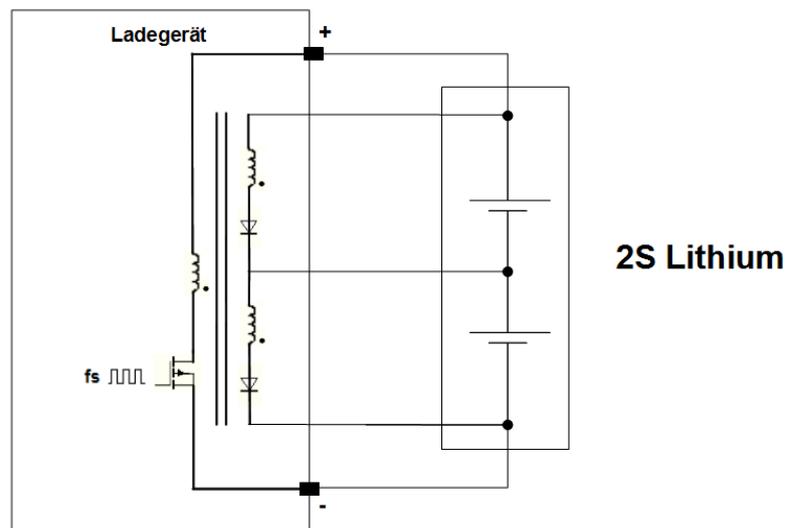


Abbildung 48: Balancierung mit geschalteter Induktivität

Dieses Balancierungs-Verfahren beruht auf dem transformatorischen Prinzip. Es wird eine Gleichspannung mit einer fixen Frequenz gekettet und erzeugt so ein magnetisches Feld welches eine Spannung in den sekundär Wicklungen induziert. Diese Sekundärseitige Spannung teilt sich über alle Sekundärspulen gleichmäßig auf und es werden alle Zellen geladen bzw. diejenigen mit höhere Zellenspannung entladen und so

die Energie in die anderen Zellen gesteckt. Der Wirkungsgrad bei diesem Verfahren ist sehr hoch, weshalb es auch dort eingesetzt wird, wo eine Balancierung über den Ladevorgang hinaus nötig ist. Umso höher die Schaltfrequenz ist desto kleiner kann der Transformator sein.

## 7.2 Balancierungs Bausteine (IC's)

<sup>13</sup>Heutzutage benötigen eine Vielzahl von technischen Anwendungen im Bereich der Elektroantriebstechnik und Batterieversorgungssystemen integrierte Bausteine zur Lösung von technischen Herausforderungen beim Betreiben von Aukkupacksystemen. Viele Halbleiterhersteller bieten heute in diesem Bereich des „Power Managmentes“ integrierte Lösungen an. Es gibt eine Vielzahl von Bausteinen dessen Hardware spezielle für das Messen und Balancieren von Lithium Akkupacks ausgelegt ist. Diese Bausteine sind im gemeinen so aufgebaut dass sie über eine Logische Einheit zur Erfassung der Zellenspannungen verfügen d.h. einen eingebauten „Analog Digital“-Konverter, eingebaute Schalter parallel zu jeder Zelle sowie eine Schnittstelle mit der man zu einem Mikrocontroller kommunizieren kann verfügen, der wiederum die nötige Software programmiert hat, mit dem gewünschten Balancierungs-Algorithmus. Die Bausteine unterscheiden sich im wesentlich durch die minimale und maximale Anzahl der anschließbaren Akkuzellen, sowie dem verwendbaren chemischen System, da es Bausteine gibt die nur Lithium Zellen verarbeiten könne und solche die sowohl Lithium als auch NimH Zellen messen können. Unterschiede gibt es auch bei der Versorgungsspannung, da manche Bausteine nur vom Akkupack versorgt werden können, anderen dagegen könne an eine externe Versorgungsspannung angeschlossenwerden. Dies hat was mit der internen Struktur der Bausteine zu tun, beziehungsweise muss darauf geachtet werden ob und wann die internen parasitären Diodenstrukturen leitend werden können. Bei der Suche nach dem geeignetem Baustein standen folgende Modelle der einzelnen Hersteller zur Verfügung. Der interne Aufbau eines BalancierungsBausteines mit seinen entsprechenden Hardware Komponenten soll hier anhand des **MAXIM11068** beschrieben werden.

---

<sup>13</sup> MAXIM MAX11068 Datenblatt

## 7.2.1 Anforderungen an den Balancierungs-Baustein

- Er muss mindestens 1 NiMH Zelle und maximal 10 Lithium Zellen messen können
- Er muss extern versorgt werden können (unabhängig von der Akkupack-Spannung)

### 7.2.1.1 LTC6802

Ist ein mess und monitoring Baustein mit einem internen 12-bit ADC, eine 3,075V Referenzspannung, einem hochspannungs Multiplexer, einer SPI-Schnittstelle sowie internen Schalttransistoren. Dieser Baustein zeichnet sich dadurch aus das er 12 Zellen auf einmal messen kann, das hat den Vorteil dass nur ein Baustein benötigt wird. Allerdings ist dieser Baustein in der Lage nur Lithium Zellen zu messen, da aber auch NimH Zellen gefordert sind ist dieser Baustein nicht geeignet gewesen. Zwei weitere negative Aspekte sind dass er nur vom Akkupack versorgt werden kann und dass er mindestens 4 Lithium Zellen zum funktionieren benötigt.

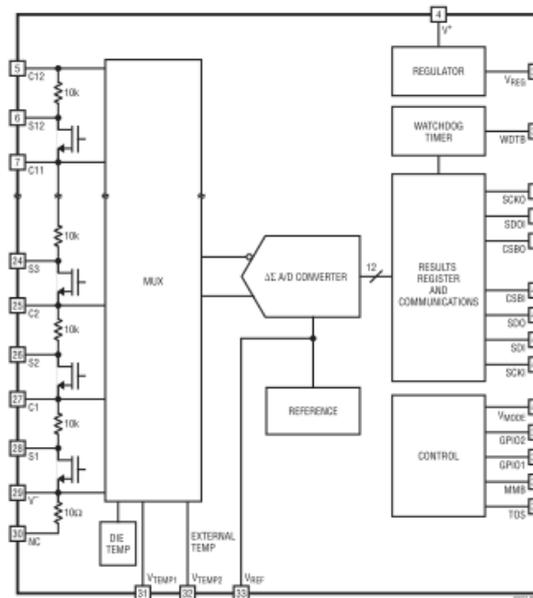


Abbildung 49: Blockschaltbild des LTC6802 Bausteines

### 7.2.1.2 LTC6803

Dieser Baustein ist im Grunde ähnlich aufgebaut wie der LTC6802, er hat einen 12-bit ADC, eine 3,075V Referenzspannung, einen hochspannungs Multiplexer, eine SPI Schnittstelle sowie interne Schalttransistoren. Allerdings hat dieser Baustein den Vorteil dass er auch NimH Zellen messen kann. Desweiteren ist er in der Lage durch eine externe Spannung versorgt zu werden und funktioniert deshalb auch mit nur einer Lithium oder NiMH Zelle. Leider waren die Lieferzeiten für diesen Baustein zu lange.

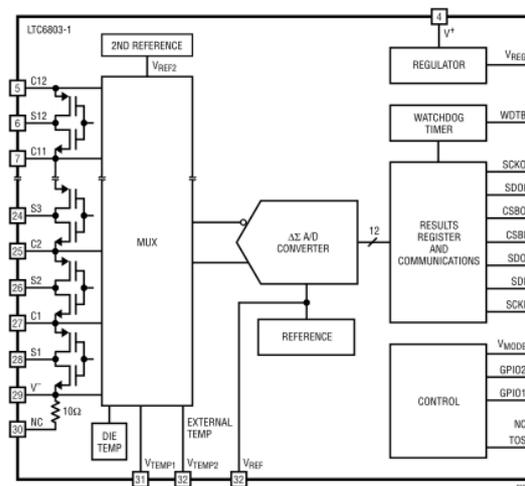


Abbildung 50: Blockschaltbild des LTC6803 Bausteines

### 7.2.1.3 ISL9216

Ist ein monitoring Baustein der allerdings keinen internen AD-Wandler besitzt und als Schnittstelle einen I<sup>2</sup>C Bus verwendet. Er kann bis zu 12 Lithium Zellen messen allerdings benötigt man einen zusätzlichen Baustein ISL9216 um mehr als 7 Lithium Zellen messen zu können, benötigt aber mindestens 8 Lithium Zellen um zu funktionieren. Der Vorteil bei diesem Baustein ist dass er eine Überstromschutzfunktion hat. Auch dieser Baustein zeigte sich ungeeignet da er nur vom Akkupack versorgt werden kann.

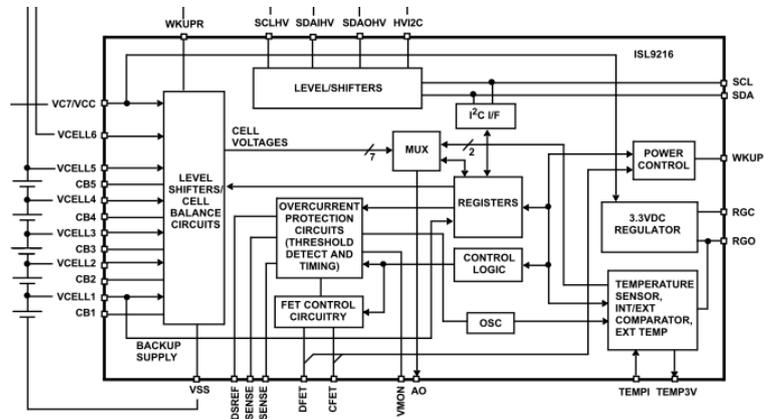


Abbildung 51: Blockschaltbild des ISL9216 Bausteines

### 7.2.1.4 ATA6870

Der ATA6870 ist ein mess und monitoring Baustein für Lithium und NiMH Zellen. Im Gegensatz zu anderen Bausteinen besitzt er für jede einzelne Zelle einen 12-bit ADC. Ein interner 3,3V Spannungsregler ermöglicht die Versorgung z.B. eines Mikrocontrollers. Ein negativer Aspekt des Bausteines ist das er nur 6 Zellen pro Baustein messen kann. Eine Kaskadierung der Bausteine und Kommunikation mit einem Mikrocontroller erfolgt über eine SPI-Schnittstelle.

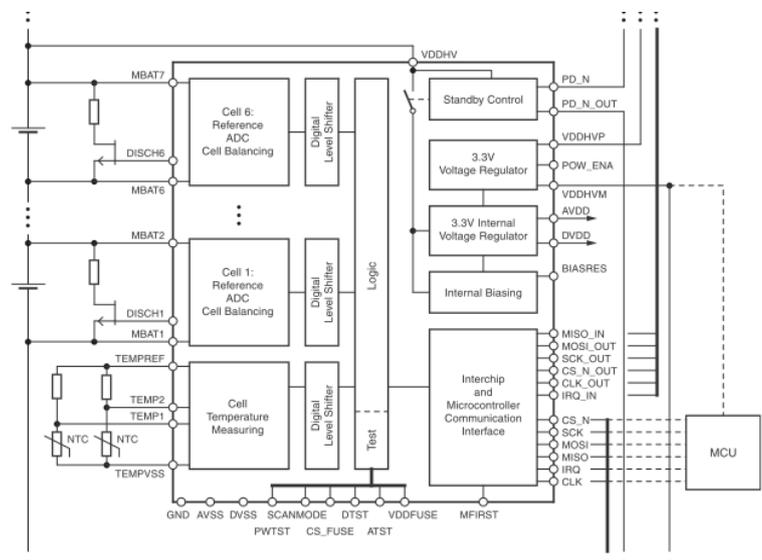


Abbildung 52: Blockschaltbild des ATA6870 Bausteines

### 7.2.1.5 MAX11068

Ist ein programmierbarer mess und monitoring Baustein für Lithium, NiMH und SuperCaps Zellen. Er besitzt einen internen 12-bit AD-Wandler, interne Schalter, eine 2.5V Referenzspannung, 2 Messkanäle für die Temperaturmessung und eine I<sup>2</sup>C Bus Schnittstelle für die Kommunikation mit einem Mikrocontroller. Dieser Baustein zeichnet sich dadurch aus dass er einerseits 12 Lithium Zellen auf einmal messen kann und dass er mit einer externen Spannung versorgt werde kann was die Messung nur einer NiMH Zelle möglich macht.

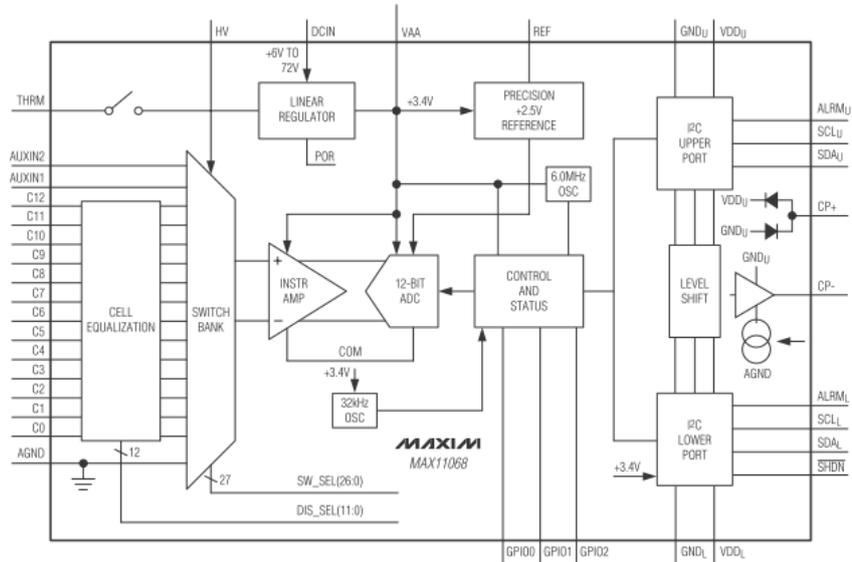


Abbildung 53: Blockschaltbild des MAX11068 Bausteine

Tabelle 3: Gegenüberstellung der verschiedenen Balancierungs-Bausteine

	Mindestzellenanzahl	Genauigkeit [mV]	Schutzfunktion	Kaskadierbar	Zellen/IC	Anzahl ICs max	Anzahl Zellen max	I[μA] Sleep
LTC6802	4	1,5	nein	ja	12	7	84	85
LTC6803	1	1,5	nein	ja	12	7	84	85
ISL9216	3	-42 bis +58	Ja	ja	7	2	14	10
ATA6870	3	-10 bis +10	nein	ja	6	16	96	10
MAX11068	1	-10 bis +10	nein	ja	12	31	372	75

### 7.3 Die Auswahl des Balancierungsbausteines

Bei unserer Auswahl des Balancierungsbausteines spielen die Gründe wie: Versorgungsspannung, minimale Zellenanzahl, maximale Zellenanzahl und Akku-Type eine Rolle. Für uns wichtig ist das der Baustein über eine externe Versorgungsspannung versorgt werden kann, so dass das Gerät auch bei nicht anschließen eines Akku-Packs betriebsbereit bleibt. Denn ob ein Baustein auch extern versorgt werden kann, hängt vom inneren Aufbau des Bausteines ab, nämlich wie das Substrat aufgebaut ist. Dieser muss dafür ausgelegt sein, was aus dem Dioden-Diagramm ersichtlich ist. Man muss verschiedene Fälle unterscheiden. **Erster Fall:** es liegt keine Akku-Packs Spannung vor, aber die Versorgungsspannung ist angeschlossen. **Zweiter Fall:** es ist ein Akku-Pack angeschlossen aber dieser ist defekt. Ausschlaggebend ist dabei ob eine der Dioden aus dem Dioden-Diagramm des Bausteines leitend werden kann oder nicht, da dies zu einem Defekt des Bausteines führen würde.

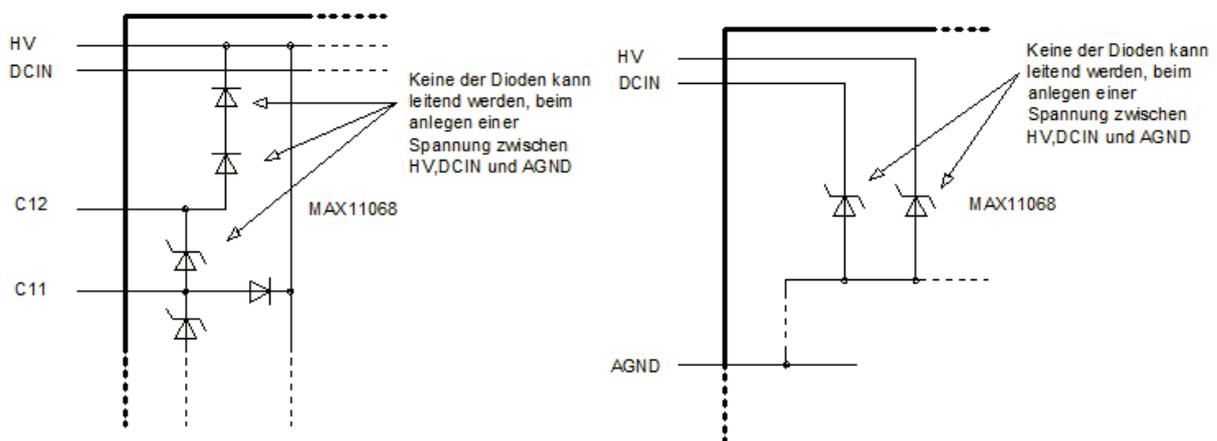


Abbildung 54: Die Abbildung des relevanten Bereiches der internen Diodenstruktur des MA11068

Kann der Baustein extern versorgt werden, dann ist die Bausteinversorgung unabhängig von der minimalen Anzahl der Akku-Zellen. Benötigen andere Bausteine eine mindest Zellenanzahl von z.B. 3 bis 6 Zellen, je nach Akku-Chemie (NiMH hat 1,2V pro Zelle und Li-Ion hat 3,7V pro Zelle), so kann an extern versorgten Bausteinen auch eine einzige NiMH-Zelle angeschlossen werden. Ebenfalls ein wichtiger Parameter ist die maximal mögliche Anzahl von anschließbaren Akku-Zellen, diese sollte mindestens 10-Zellen vom Type Li-Ion betragen. Würde der Baustein weniger als 10-Zellen schaffen, dann

müssten zwei Bausteine verwendet werden, die dann zusammenschaltet werden. Dies führt zu zusätzlichen Problemen, da der obere Baustein mit seinem „Ground“-Potential nicht auf „0V“ liegt, sondern auf dem Potential der obersten Zelle des unteren Bausteines. Wenn nun ein  $\mu\text{C}$  mit dem Baustein kommunizieren möchte, entsteht nun auf der „Ground“ (Bezugsgröße) -Leitung der Signalleitung eine Spannung weit höher als die maximal zulässige. Um dieses Problem zu lösen, muss eine Potentialtrennung zwischen den Bausteinen und dem  $\mu\text{C}$  durchgeführt werden. Dazu wird ein OPTO-Koppler in die Signalleitungen des  $\mu\text{C}$  geschaltet, dies führt zu einem zusätzlichen Schaltungsaufwand. Außerdem können bei der Verwendung von OPTO-Kopplern Störungen in der Übertragung entstehen. Das wird am nachfolgendem Beispiel des Atmega ATA 6870 ersichtlich:

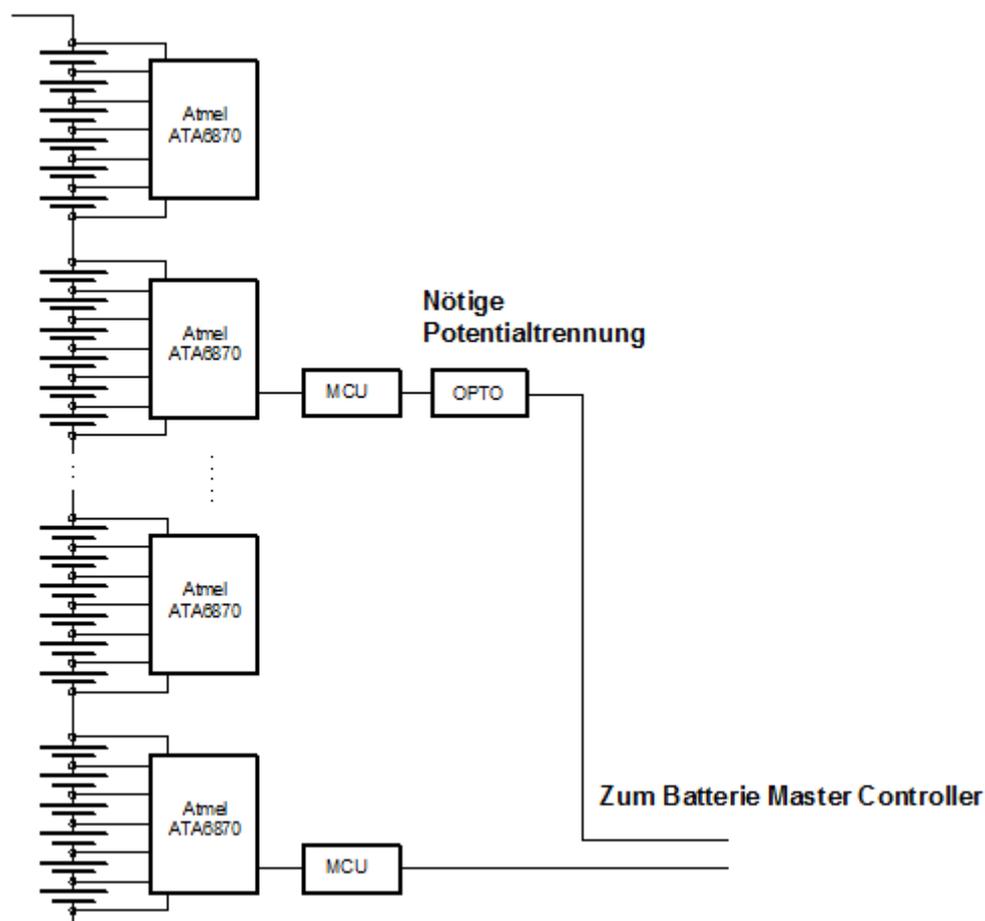


Abbildung 55: Das Blockschaftbild der Zusammenschaltung von mehreren Atmega 6870 Bausteinen

Ausgehend von den Anforderungen und der Verfügbarkeit kommt für unsere Anwendung nur der Baustein von MAXIM der „MAX11068“ in Frage. Auf der nachfolgenden Seite ist das Dioden-Ersatzschaltbild des Bausteines dargestellt, dabei sind die relevanten Pin-Anschlüsse mit schwarzen Rechtecken markiert.

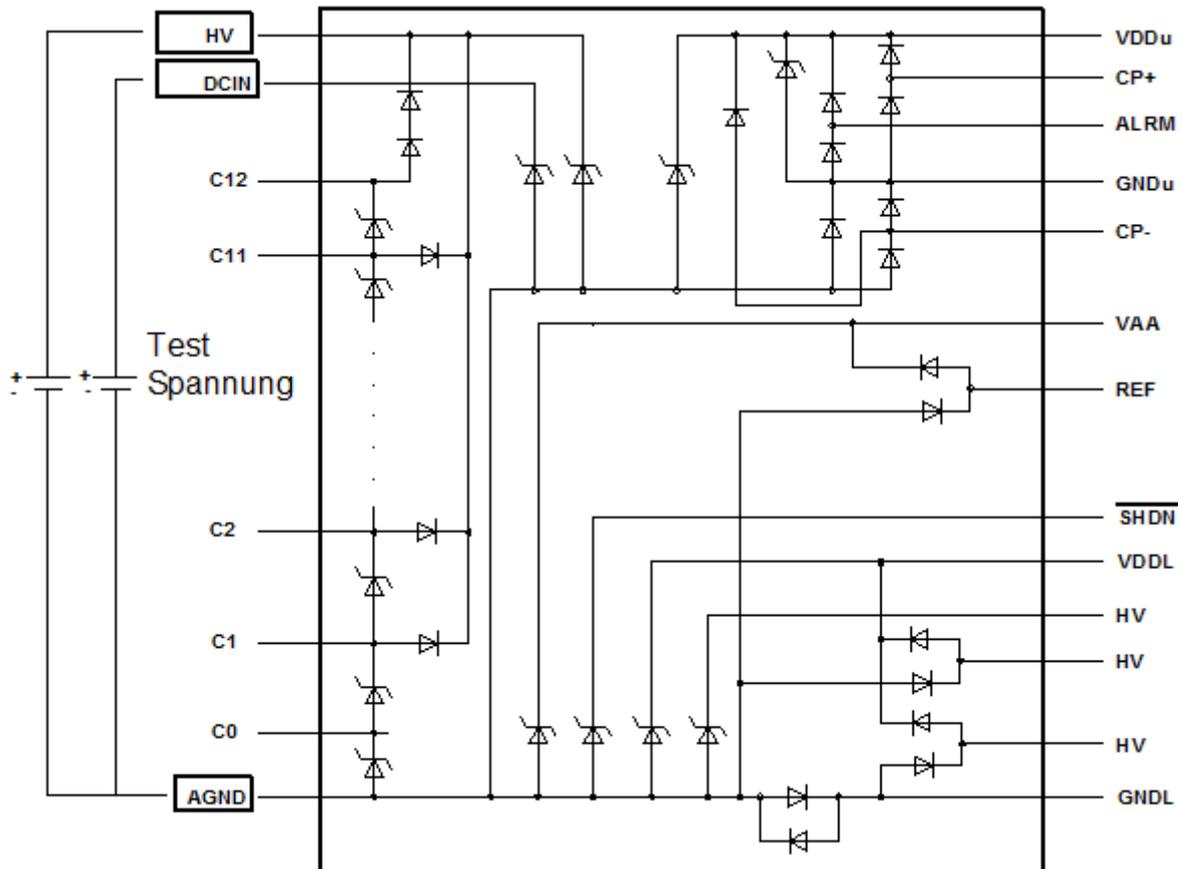


Abbildung 56: Das Dioden-Ersatzschaltbild des Bausteines des MAX11068

Die Dioden sind teilweise ein Ersatzschaltbild für die internen CMOS-Strukturen und teils handelt es sich um Suppressordioden. Beim Baustein MAX11068 ist ausschlaggebend ob die Pin Anschlüsse „DCIN“ und „HV“ bei einer gewissen Pegelkombination gegen Pin „AGND“ eine leitende Diode bewirkt werden kann. Aus dem Ersatzschaltbild der internen Diodenstrukturen ist ersichtlich dass nicht der Falls sein kann.

## 7.3.1 Der Baustein MAX 11068 im Detail

### 7.3.1.1 Allgemein

Beim Max 11068 handelt es sich um einen hochintegrierten, programmierbaren, Hochspannungs „Batterie-Monitoring“ und einer „Smart Data“ –Erfassung Schnittstelle, mit 12 Kanälen. Dieser Baustein ist gedacht für den Einsatz in Batterien im Bereich der Elektromobilität, wie zum Beispiel bei Hybrid Auto Akku-Packs, Elektroauto Akku-Packs und für jegliches System das eine lange Reihen von sekundären Batterie-Zellen (NiMH, Li-Ion) zusammengeschaltet hat. Der Baustein enthält eine einfache „State Machine“ und einen hochgeschwindigkeits  $I^2C$ - Bus für die serielle Kommunikation. Am Eingang des Bausteines befindet sich ein 12 Kanaliger Hochspannungs-Multiplexer, jede Zelle wird dabei differentiell gemessen. Der „Full-Scale“ Messbereich reicht von 0V bis 5V, wobei eine volle Genauigkeit von 0,5V bis 4,7V garantiert wird. Die Eingangs Multiplexer/„Switch Bank“ ermöglicht die Messung jeder Zelle einzeln in einer seriell zusammengeschalteten Reihe von Batterie-Zellen. Ein 12-Bit sukzessiver Approximations Analog-Digital Wandler dient zur Digitalisierung der Zellenspannungen, dabei können alle 12 Zellen innerhalb von  $107\mu\text{s}$  gemessen werden. Der MAX11068 verwendet einen „Zwei-Scan-Ansatz“ für die Messung der Zellenspannungen und Korrektur der Messfehler. Die erste Phase des „Scans“ ist die Erfassung der Spannung aller 12 Zellen. Die zweite Phase ist die Fehler-Löschung-Phase, dabei wird der Analog-Digital Eingang zerhackt um die Fehler zu entfernen. Dieser Zwei-Phasen-Ansatz liefert hervorragende Genauigkeit über einen großen Temperaturbereich und in Systemen mit großem Rauschsignal. Der MAX11068 beinhaltet einen internen Oszillator der eine Systemtaktrate von 6 MHz erzeugt bei einer Genauigkeit von +/- 3%. Der Stromverbrauch des Bausteines beträgt im Datenerfassungsmodus weniger als 2,0mA. Der Stromverbrauch reduziert sich im „Standby-Modus“ auf  $75\mu\text{A}$  und im „Shutdown-Modus“ auf weniger als  $1\mu\text{A}$ . Der Baustein ist in einem 39-Poligem Gehäuse verpackt im Format TSSOP



- **CHARGE PUMP**

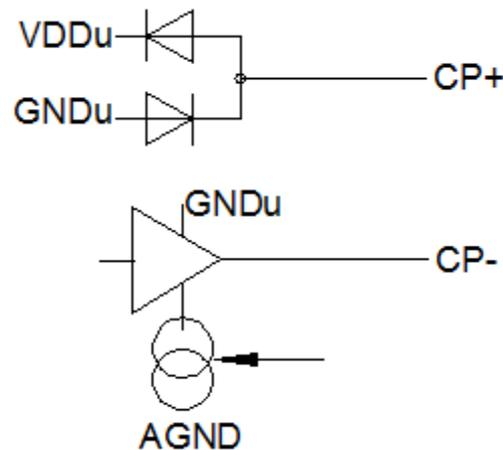


Abbildung 59: Die "Charge Pump" des MAX11068-Bausteines

### 7.3.1.3.1 Funktionsbeschreibung der „Charge Pump“

Eine „Charge Pump“ ist einer Art von DC-DC-Wandler das Kondensatoren als Energiespeicher-Elemente verwendet, um entweder eine höhere oder niedrigere Spannung zu schaffen. Die „Charge Pump“ gewährleiste dass das obere I<sup>2</sup>C-Port-Modul immer mit einer Spannung von 3,4V versorgt wird. Sprich je nach Spannungs-Niveau des GND<sub>U</sub>-Anschlusses(dieser ist verbunden mit der obersten Akku-Pack Zelle) generiert die „Charge Pump“ die nötige Spannung am V<sub>DDU</sub> -Anschluss damit sich eine Spannungsdifferenz von 3,4V zwischen den Anschlüssen GND<sub>U</sub> und V<sub>DDU</sub> ergibt. Dadurch ist es möglich mehrere Baustein zu kaskadieren. Damit entfällt für das kaskadieren von mehreren Bausteinen die Notwendig von Optokopplern, die zu Störungen im Kommunikation s-Weg führe können.

- **CELL EQUALIZATION**

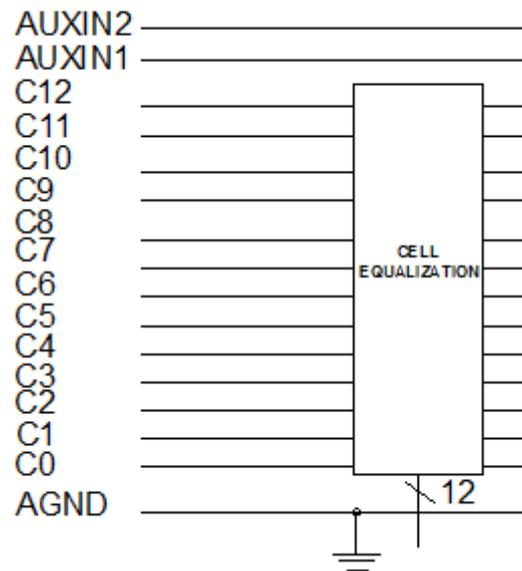


Abbildung 60: Das "Cell Equalization"-Modul des MAX11068-Bausteines

### 7.3.1.3.2 Funktionsbeschreibung der „Cell Equalization“

In der „Cell Equalization“ Einheit sind die 12 Balancierungs-Schalter untergebracht, die zum balancieren der Batterie-Zellen dienen. Diese internen Schalter können Ströme bis zu 200mA leiten. Der MAX11068 Baustein kann eine Verlustleistung von 1.2W absorbieren was die Anzahl der eingeschaltet balancierungs Schalter begrenzt wenn ein Strom von 200mA fließen soll. Mit einem Balancierungsstrom von 110mA pro Zelle, können alle 12 Schalter zur gleichen Zeit eingeschaltet werden.

- **SWITCH BANK**

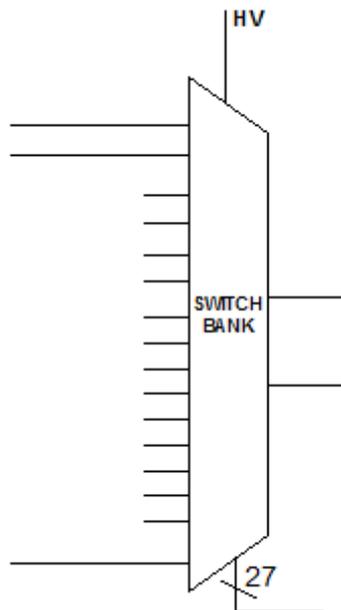


Abbildung 61: Das "Switch Bank"-Modul des MAX11068-Bausteines

### 7.3.1.3.3 Funktionsbeschreibung der „Switch Bank“

Die Zellen-Eingänge werden differenziell gemessen. Die „Switch Bank“ –Komponente beinhaltet deswegen Schalter die jede Zelle einzeln zu einem Instrumentierungsverstärker zu schalten, bevor die Zelle zu einem Analog –Digital Umsetzer geschaltet wird. Diese Auswahl der zu messenden Zelle erfolgt über einen hochspannungs Multiplexer (Pin „HV“)

- **I<sup>2</sup>C UPPER PORT**

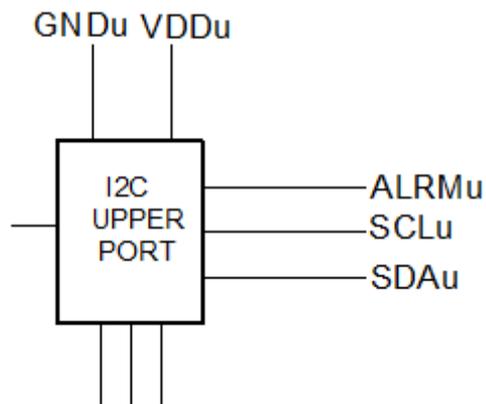


Abbildung 62: Das "Upper Port"-Modul des MAX11068-Bausteines

#### 7.3.1.3.4 Funktionsbeschreibung des „I<sup>2</sup>C Upper Port“ Modules

Dieses Modul dient zur Kommunikation über einen I<sup>2</sup>C-Bus mit einem darüber liegenden Baustein. Dieses I<sup>2</sup>C-Modul liegt mit seinem „Ground“-Potential auf der obersten Batterie-Zelle. Dabei sind GND<sub>U</sub> und VDD<sub>U</sub> die Versorgungsanschlüsse des Modules, wobei GND<sub>U</sub> gleichzeitig der Versorgungseingang der „Charge-Pump“ ist und VDD<sub>U</sub> über den Ausgang der „Charge-Pump“ versorgt wird. SCL<sub>U</sub> ist die Taktleitung des I<sup>2</sup>C-Buses pegelverschoben um den Betrag der höchsten Zellenspannung, und SDA<sub>U</sub> ist die Datenleitung des I<sup>2</sup>C-Buses ebenfalls pegelverschoben um den Betrag der höchsten Zellen. Durch diesen Aufbau der Buskommunikation mit benachbarten Bausteinen ist keine Potentialtrennung des Kommunikationsweges mit Optokopplern zwischen den Bausteinen nötig. Die ALRM<sub>U</sub>-Leitung gibt bei Unterspannung/Überspannung, Übertemperatur/Untertemperatur ein Fehlersignal aus.

- **I<sup>2</sup>C LOWER PORT**

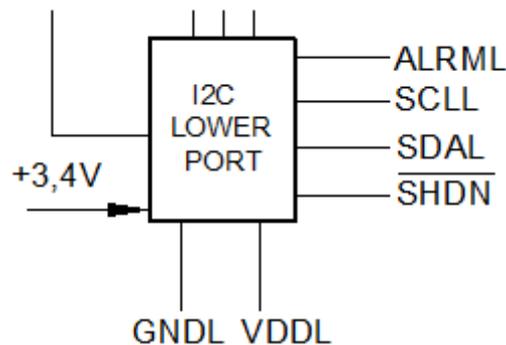


Abbildung 63: Das "Lower Port"-Modul des MAX11068-Bausteines

### 7.3.1.3.5 Funktionsbeschreibung : des „I<sup>2</sup>C Lower Port“ Modules

Dieses Modul dient zur Kommunikation über einen I<sup>2</sup>C-Bus mit einem darunterliegenden Baustein. Das Modul wird dabei mit einer internen Gleichspannung von 3,4V versorgt. Die SCL<sub>L</sub>-Leitung ist die Taktleitung und die SDA<sub>L</sub>-Leitung die Datenleitung des I<sup>2</sup>C-Buses. GND<sub>L</sub> und VDD<sub>L</sub> sind die Versorgungsanschlüsse des Modules. Im Falle des untersten Bausteines in einer Baustein-Kette wird GND<sub>L</sub> mit „Ground“-Potential verbunden und VDD<sub>L</sub> mit der Baustein-Betriebsspannung.

### 7.3.1.4 Zellen Balancierung

Der Zellen-balancierungs Schaltkreis des MAX11068 besteht aus internen Schaltern und externen Widerständen um den nötigen Entladestrom, der von der Zellenspannung abhängig ist, zum ausbalancieren der Zellen fließen zu lassen.

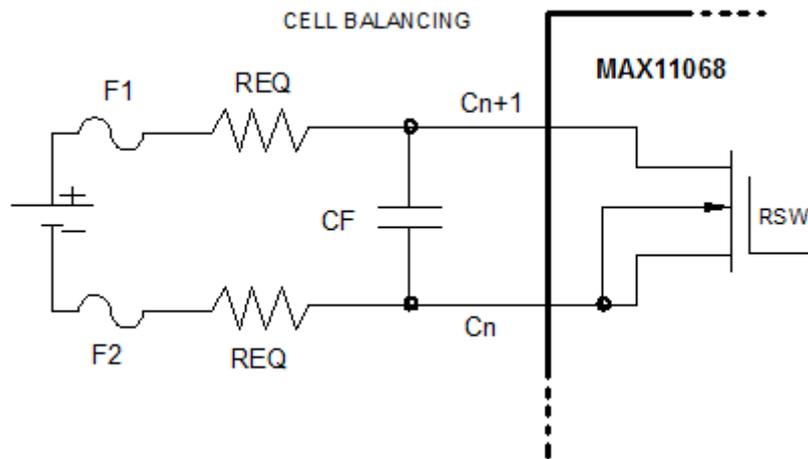


Abbildung 64: Die internen Balancierungs-Schalter des MAX11068-Bausteines

Folgende Einschränkungen müssen bei der Verwendung der Grundsaltung berücksichtigt werden.

- Maximale Verlustleistung im Package
- Messung während einer Zellen-Balancierung
- Schutz vor „open-circuits“-Fehlern am Batterie-Stack

Der Max11068 enthält 12 unabhängig voneinander gesteuerte Schalter, die einen typischen On-Widerstand von  $6\Omega$  haben mit einer  $\pm 50\%$  Abweichung abhängig von der Temperatur. Der Max11068 besteht aus einem 38-pin TSSOP Gehäuse mit einer maximalen begrenzten Leistung von  $1.2698W$  und einem junction-to-ambient thermischen Widerstand von  $63\text{ NC/W}$  für eine Multilayer-Platine. Diese Parameter sind fundamentale Grenzen für die Verlustleistung des Bausteines und unterliegen einer sorgfältigen Überlegung bei der Verwendung der internen Zellen-Balancierungs-Schalter, da die Schalter den dominierenden Verbraucher im Gerät darstellen. Die maximale Anzahl der Zellen-Balancierungs Schalter die zu einem beliebigen Zeitpunkt aktiviert werden können, berechnet sich wie folgt:

$$\text{Maximale Anzahl von aktivierten Schaltern} = \frac{0,7 * P_{MAX}}{I_{BALANCE} * 2 * R_{SW}}$$

wobei :

$$I_{BALANCE} = \frac{V_{CELL}}{2 * R_{EQ} + R_{SW}}$$

$$P_{MAX} = 1,2698W$$

$$R_{SW} = 6\Omega, \text{ typisch}$$

Basierend auf den Berechnungen, unterstützen zwei benachbarte interne Zellen-Balancierungs Schalter einen Entladestrom von 250mA pro Zelle. Dabei ist ein  $R_{EQ}$  von 0,5W erforderlich.

#### 7.3.1.4.1 Messung während der Zellenbalancierung

Bei Verwendung der internen Zellen-Balancing Schalter wird die gemessene Spannung an den Eingängen  $C_N$  bis  $C_{N+1}$  durch den externen Widerstand  $R_{EQ}$  reduziert. Für eine genaue Zellen-Spannungs Messung müssen die internen Balancierungs-Schalter deaktiviert werden, da diese nicht automatisch während einer Konvertierung deaktiviert werden. Nach dem diese Schalter deaktiviert worden sind, muss der Eingangsspannung eine gewisse Ruhepause ( $t_{SETTLE}$ ) gewährleistet werden, diese ist abhängig von den externen Komponenten  $C_F$  und  $R_{EQ}$ , bevor eine Zellen-Messung vorgenommen wird.

$$t_{SETTLE} = 10 * R_{EQ} * C_F$$

#### 7.3.1.4.2 Stromvariationen während der Aktivierung von benachbarten Zellen- Schaltern

Wenn benachbarte Zellen-Balancing Schalter aktiviert werden, fließt ein viel höherer Entladestrom als gewünscht. Die Abbildung zeigt den Fall wenn zwei benachbarte Schalter aktiviert werden. Der resultierende Entladestrom ( $I_{DIS}$ ) berechnet sich dabei wie folgt:  $I_{DIS} = \frac{V_{CELL5} + V_{CELL4}}{2 * R_{EQ} + 2 * R_{SW}}$ , aus der Gleichung ist ersichtlich das der Entladestrom mit der Anzahl der benachbarten aktiven internen Zellen-Balancierungs-Schaltern wächst.

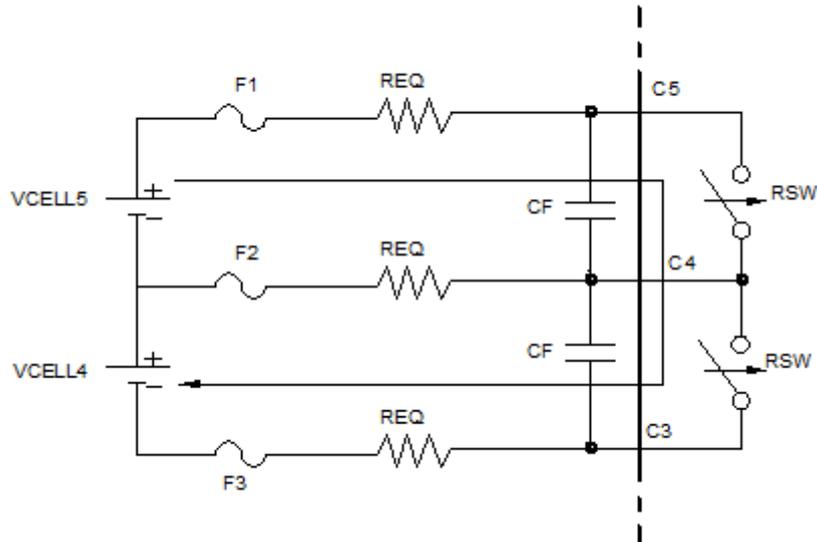


Abbildung 65: Die internen Balancierungs-Schalter des MAX11068-Bausteines

Das liegt daran dass die Zellen-Spannung über den aktiven Schaltern und dem  $R_{sw}$  proportional wächst, während der  $R_{EQ}$  gleich bleibt, egal wie viele benachbarte Schalter aktiv sind. Folglich wächst der Zähler der Entladestrom-Gleichung schneller als der Nenner. Auf diesen Effekt muss Rücksicht genommen werden, um eventuelle Schäden des Bausteines zu vermeiden.

### 7.3.1.4.3 Externe Zellenbalancierungs Schaltung

Der MAX11068 ermöglicht eine externe Zellen-balancierung in dem der Transistor des externen Stromkreislaufes durch den internen Schalter gesteuert wird. Wenn der interne Schalter geschlossen wird, dann wird über die Widerstände  $R_{BIAS1}$  und  $R_{BIAS2}$  externe Bipolar oder MOSFET Transistor eingeschaltet.

#### 7.3.1.4.4 Externe Zellenbalancierung mit einem Bipolar Transistor

Bei der Verwendung eines Bipolaren-Transistors ist es empfehlenswert einen mit einer Stromverstärkung ( $h_{FE}$ ) von 100 zu wählen und mit einer  $V_{CE}$  Spannung die mindestens der Batterie-Pack Spannung entspricht um Schäden zu vermeiden. Typische Komponenten Werte für einen 500mA Zellen-Entladestrom sind  $R_{BIAS} = 80\Omega$ ,  $R_{EQ} = 8\Omega$ . Geeignete Bipolar-Transistoren wären zum Beispiel der MJD50 (für hohe Spannungen) oder MMBTA05 (für niedrige Kosten und geringe Spannungen).

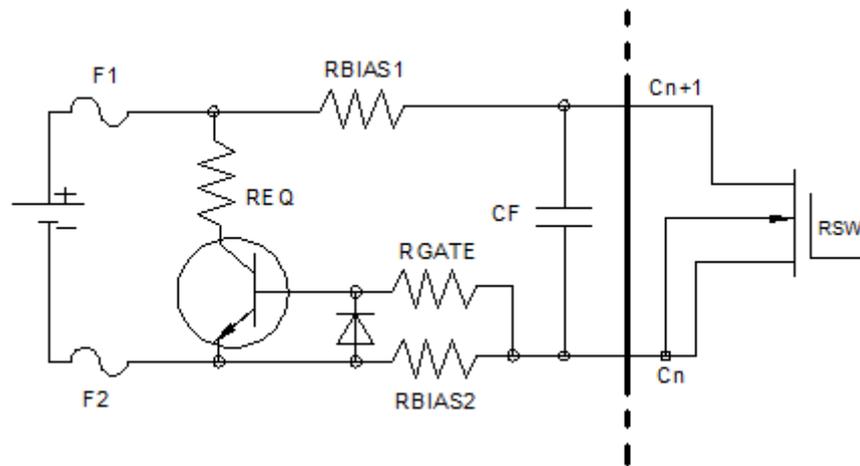


Abbildung 66: Die internen Balancierungs-Schalter mit externen Bipolaren-Transistoren

#### 7.3.1.4.5 Externe Zellenbalancierung mit einem MOSFET

Bei der Verwendung eines externen MOSFET ist es empfehlenswert einen mit einer niedrigen  $V_{GS}$  Spannung zu wählen (typischerweise 1,2V) und mit einer  $V_{DS}$  Spannung die mindestens für die gesamte Akku-Pack Spannung ausgelegt ist, sollte eine offene Schaltung im Zellen-Stapel auftreten. Wenn ein MOSFET mit niedrigerer Nennspannung gewählt wird, sollte Sicherungen in Serie verwendet werden um die Schaltung zu schützen. Typische Komponenten Werte für einen Zellen-Entladestrom von 500mA sind:

$$R_{BIAS1} = 10k\Omega$$

$$R_{GATE} = 47\Omega$$

$$R_{EQ} = 8\Omega$$

MOSFET vom Typ NTK3134N (geringe Koste, geringe Spannung)

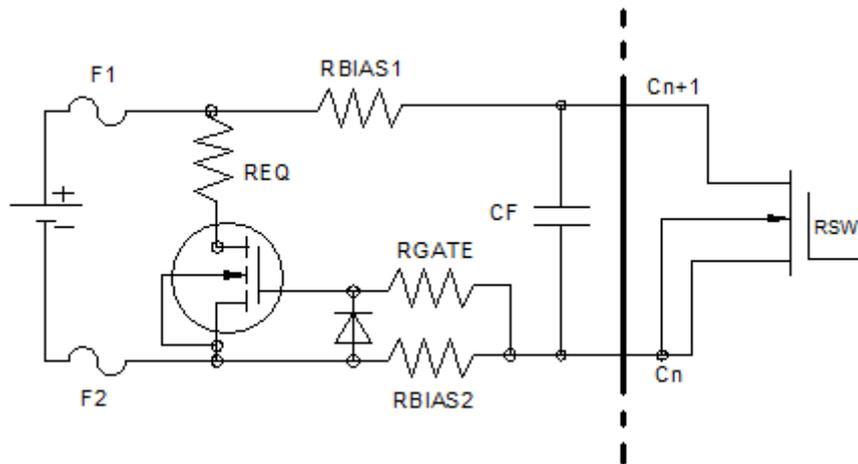


Abbildung 67: Die internen Balancierungs-Schalter mit externen MOSFET-Transistoren

Der  $R_{BIAS}$  Widerstandswert in Verbindung mit dem Eingangs-Bias-Strom fügt einen kleinen Messfehler von weniger als 1mV ein, für den ungünstigsten Fall bei einem  $R_{BIAS}$  von 10k $\Omega$ . Ein Vorwiderstand von nicht weniger als 470 $\Omega$  sollte in Reihe mit dem Transistor-Gate geschaltet werden, um die Schaltung robuster zu machen.

### 7.3.1.5 I<sup>2</sup>C Interface

Der MAX11068 benutzt eine SMBus-Leiter physische I<sup>2</sup>C Schnittstelle mit maßgeschneiderten I<sup>2</sup>C –Befehls Protokoll für die Kommunikation mit dem Host-System und von Modul zu Modul. Jeder Baustein enthält zwei I<sup>2</sup>C-Ports, einen Master und einen Slave. Der Slave-Port ist der untere Anschluss und ist bezogen auf den Baustein-Ground und kommuniziert mit dem Host-Master oder mit dem Master eines Bausteins auf dem unteren SMBus-Leiter. Der obere Anschluss ist ein Master-Port, dieser ist „level-shifted“ und bezogen auf GND<sub>U</sub>. Er dient zur Kommunikation mit Bausteinen die höher auf der SMBus-Leitung liegen und sammelt Daten die dann zurück zum Host weitergeleitet werden. Für die erfolgreiche Implementierung eines Kommunikationsbuses ist ein standard I<sup>2</sup>C hardware Master wie er in vielen Mikrocontrollern gefunden wird oder ein Master implementiert durch Firmware und „general-purpose“ I/O-Pins, alles was

erforderlich ist um erfolgreich einen Kommunikationsbus zu implementieren. Dieser „level-shifted“ duale Portaufbau erlaubt es mehrere Bausteine problemlos zusammenzuschalten ohne die Notwendigkeit einer komplexen und kostspieligen galvanischen Trennung der Kommunikationswege. Das ist besonders vorteilhaft weil es bei der Verwendung von Optokopplern in Kommunikationswegen zu Störungen in der Übertragung kommen kann.

### 7.3.1.5.1 I<sup>2</sup>C-Schnittstellenbetrieb

Die physikalische I<sup>2</sup>C-Schnittstelle für jeden Baustein MAX11068 besteht aus einem Master-Block und einem Slave-Block. Der Master-Block ist pegelverschoben und wird bezogen auf die  $GND_U$ -Versorgungsspannung. Ein digitaler Regler steuert jeden Block und koordiniert die Weitergabe von Befehlen und Daten zwischen den beiden nach Bedarf. Die beiden Standard I<sup>2</sup>C-Schnittstellen Pins für alle Ports sind SCL für den seriellen Daten-Takt und SDA für die serielle Datenleitung. Zusätzlich gibt es noch zwei Status Pins, das sind der massebezogene  $ALRM_L$  Ausgang und der pegelverschobene  $ALRM_U$  Eingang. Diese Pins dienen als Interrupt-Signale die der Host verwenden kann die Gesundheit des Busses zu bestimmen. Um die pegelverschobenen I/O-Pins zu versorgen wird eine pegelverschobene  $VDD_U$ -Versorgungsspannung durch eine interne Ladungspumpe bezogen auf  $GND_U$  generiert. Diese Versorgungsspannung bietet eine Pullup-Spannung auf das Niveau der pegelverschobenen Bus-Kommunikationssignale. Die Abbildung 75 zeigt die vereinfachte Darstellung der I<sup>2</sup>C-Schnittstelle.

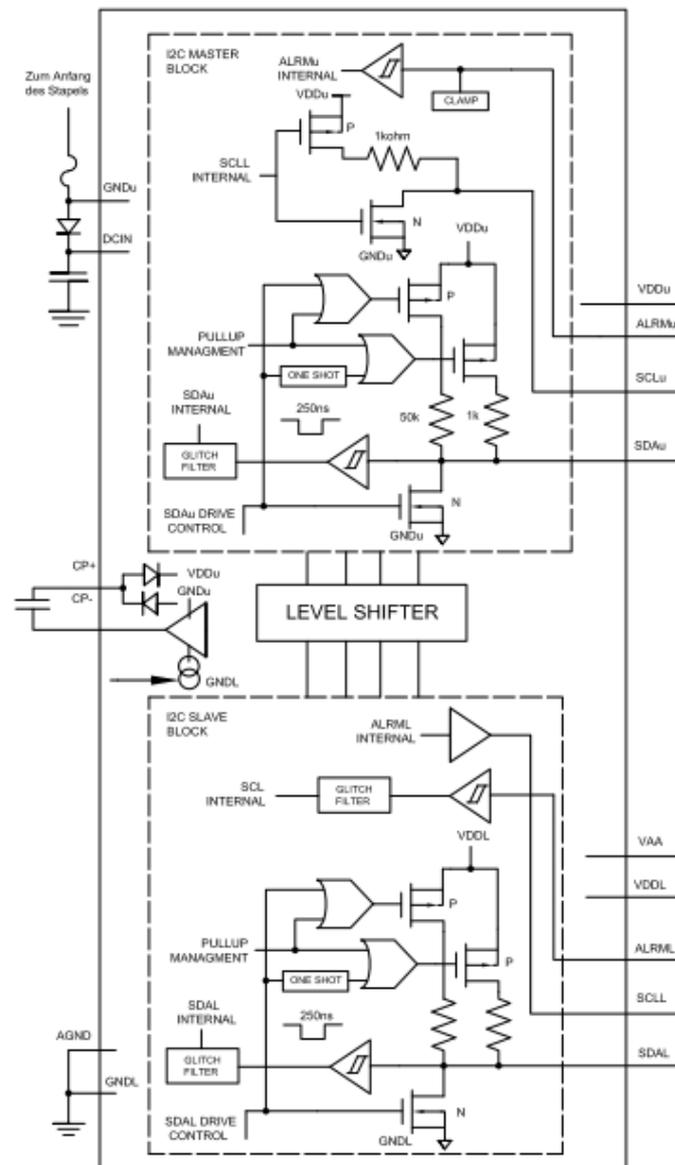


Abbildung 68: Blockschaltbild des gesamten I<sup>2</sup>C-Portes des MAX11068

Jeder Port enthält einen bidirektionalen SDA-Pin mit einem internen Pullup-Treiber. Der SCL-Pin für den unteren Slave-Port dient nur als Eingang, während der obere Master-Port SCL-Pin einen 1kΩ Pullup-Treiber. Glitch-Filter und Schmitt-Trigger Puffer an den Eingängen minimieren die Kommunikationsfehler der Eingangssignale. Jeder Port ist so entworfen dass er sowohl mit einer AC und einer DC-Kopplung betrieben werden kann. Alle Signal-Pins haben einen schwachen 150kΩ Pullup zu der V<sub>DD</sub>-Versorgung um den übliche Ruhezustand des I<sup>2</sup>C-Buses festzulegen.

### 7.3.1.5.2 I<sup>2</sup>C Befehlszusammenfassung

Der MAX11068 unterstützt sieben verschiedene Befehle. Es gibt zwei wesentliche Formate, ein für „ReadAll“ und das andere für den Rest der Kommandos. Mehrere Befehle verlangen vom Host das Senden eines PEC-Bytes oder von einer Befehls-Kette das senden eines PEC-Bytes an den Host. Das ist die Umsetzung eines SMBus PEC-Algorithmus der ein CRC-8 Prozess ist.

### 7.3.1.5.3 I<sup>2</sup>C Befehlsliste des MAX11068

Tabelle 4: Kurzbeschreibung der Befehle des MAX11068

Funktion	Beschreibung	PEC Byte
<b>HELLOALL</b>	Dieser Befehl setzt die Geräte-Adresse des ersten Teils in der Kette. Allen anderen Teilen wird dann eine automatisch inkrementierte Adresse zugeordnet. Der HelloAll-Befehl sollte nach jedem Aus- und Wiedereinschalten ausgeführt werden.	Kein
<b>ROLLCALL</b>	Liest 2 Bytes von jedem Gerät in der Kette, zu dem auch das Adressbyte gehört. Wenn 0xFF zurückgegeben wird, hat der Host die Adressen aller Geräte.	Kein
<b>SETLASTADDRESS</b>	Der Host informiert alle Geräte die beim ROLLCALL-Befehl bestimmt wurden. So weiß jedes Gerät wann ein PEC-Byte zu erwarten und zu erzeugen ist.	Vom Host benötigt
<b>WRITEALL</b>	Sendet einen gemeinsamen Befehl an alle aktiven Geräte an der Kette.	Vom Host benötigt
<b>READALL</b>	Liest die verfügbaren Daten aus dem Geräte-Register angegeben durch das Befehls-Code Byte für jedes Gerät in der Kette.	Gesendet zum Host
<b>WRITEDEVICE</b>	Schreibt Daten nur auf ein bestimmtes Zielgerät.	Benötigt vom Host

### 7.3.1.5.4 I<sup>2</sup>C Befehlsformate

#### Write Word Format

START	DEVICE OR GLOBAL ADDRESS	WR	ACK	COMMAND CODE/ REGISTER ADDRESS	ACK	DATA LOW [7:0]	ACK	DATA HIGH [15:8]	ACK	STOP
	7 bits			8 bits		8 bits		8 bits		

Abbildung 69: Bit-Format des "Write Word"-Befehls des MAX11068

#### ReadAll Format, Single Device

START	DEVICE OR GLOBAL ADDRESS	WR	ACK	COMMAND CODE/ REGISTER ADDRESS	ACK	SR	ADDRESS	RD	ACK	DATA LOW [7:0]	ACK	DATA HIGH [15:8]	NACK	STOP
	7 bits			8 bits			7 bits			8 bits		8 bits		

Abbildung 70: Bit-Format des "Read All"-Befehls des MAX11068

#### I<sup>2</sup>C Address Byte Encoding

I2C ADDRESS BIT	7	6	5	4	3	2	1	R/Wb
Broadcast Address (default value)	B7	B6	B5	B4	B3	B2	B1	1/0
	0	1	0	0	0	0	0	1/0
HELLOALL	1	1	A0	A1	A2	A3	A4	0
WRITEDEVICE	1	0	A0	A1	A2	A3	A4	0

Abbildung 71: Bit-Format des "I2C Address Byte"-Befehls des MAX11068

### 7.3.1.5.4.1 „HELLOALL“ Befehl

Der „HELLOALL“-Befehl dient zum eigentlichen initialisieren des Bausteines bzw. der Baustein-Kette und teilt jedem Baustein eine eindeutige Geräte-Adresse zu. Er sollte ausgeführt werden nach jedem aus und einschalten des Bausteines zur Neukonfiguration aller Adressen. Der „HELLOALL“-Befehl ist ein Standard I<sup>2</sup>C-Adress Byte, wo die ersten zwei Bits „11“ sein müssen, die nächsten 5 Bits die gewünschte Adresse des ersten MAX11068 Bausteines in der SMBus-Kette darstellen und das letzte Bit das Standard I<sup>2</sup>C-R/Wb-Bit ist. Diese Bit sollte für diesen Befehl immer 0 sein.

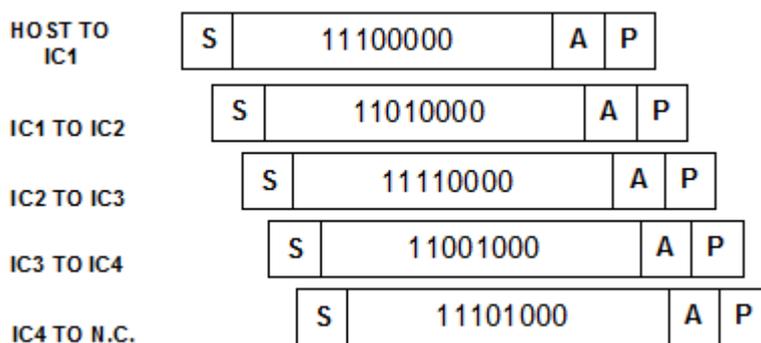


Abbildung 72: Bit-Format des "HALLOALL"-Befehls des MAX11068

### 7.3.1.5.4.2 „ROLLCALL“ Befehl

Der „ROLLCALL“ Befehl wird verwendet um die Anzahl der Bausteine in der Kette zu ermitteln. Er sollte ausgeführt werden nach dem „HELLOALL“ Befehl und nach jedem aus und wieder Einschalten der Versorgungsspannung. Das Format für diesen Befehl ist ähnlich dem „READALL“-Befehl mit dem Unterschied das 0xFF zurückgesendet wird anstatt des PEC und „Data Check“ Bytes. Der „ROLLCALL“-Befehl liest immer das Adressregister aus, dieses kann nur mit diesem Befehl ausgelesen werden.

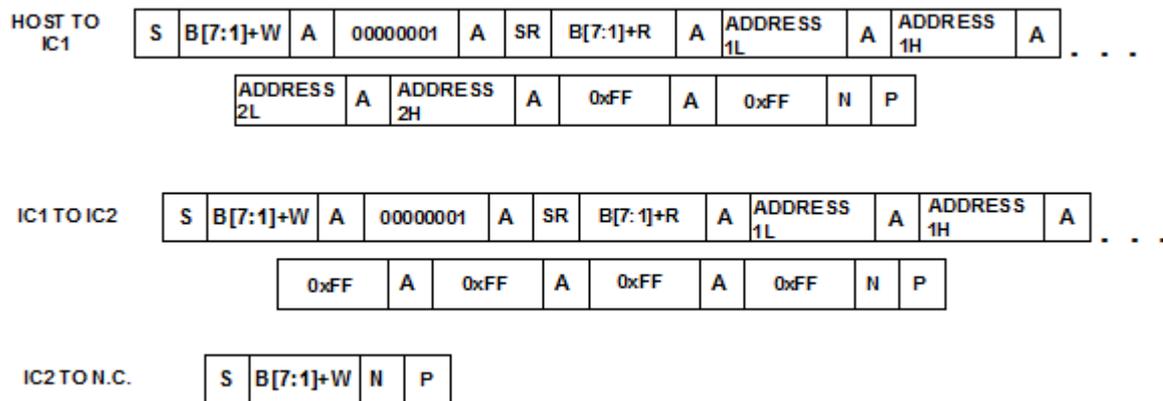


Abbildung 73: Bit-Format des "ROLLCALL"-Befehls des MAX11068

### 7.3.1.5.4.3 „SETLASTADDRESS“ Befehl

Dieser Befehl wird verwendet um jeden MAX11068 Baustein einer SMBus-Kette zu sagen welcher Baustein-Adresse die letzte ist. Jeder Baustein muss diese Information wissen um das PEC-Byte richtig zu platzieren im Datenstrom, während relevanter Kommunikationsoperationen. Der I<sup>2</sup>C-Master bestimmt die Identität des letzten Bausteines durch den „ROLLCALL“-Befehl, welcher immer dem „SETLASTADDRESS“-Befehl vorausgehen sollte. Ist die letzte Baustein-Adresse einmal bekannt initiiert der Host den „SETLASTADDRESS“ –Befehl dieser schreibt die Information in das Adress-Registers. Bei nur einem Baustein erfolgt die eigentliche Initialisierung nur durch diesen Befehl.

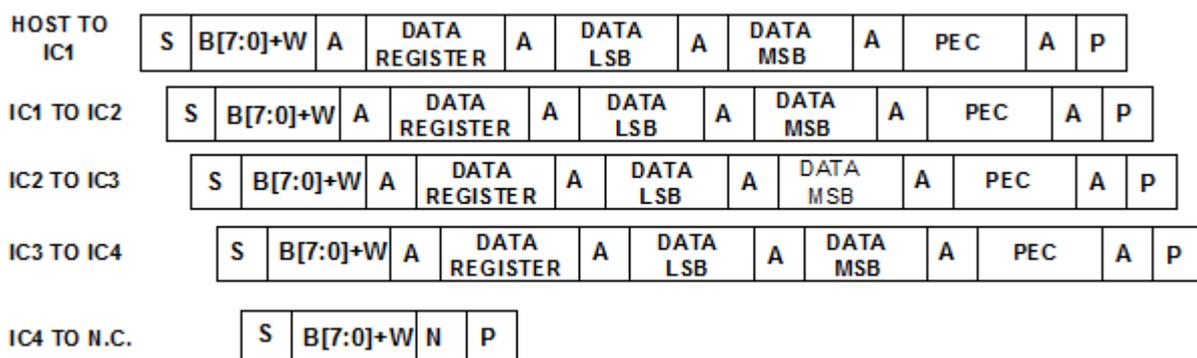


Abbildung 74: Bit-Format des "SETLASTADDRESS"-Befehls des MAX11068

### 7.3.1.5.4.4 „WRITEALL“ Befehl

Der „WRITEALL“-Befehl ermöglicht es einen vorgegebenen Wert in ein bestimmtes Register zu schreiben in alle aktiven Max11068 -Bausteinen zur gleichen Zeit. Da die meisten konfigurations Inforamtionen für alle Bausteine gleich sind, ermöglicht der „WRITEALL“-Befehl einen schnelleren Zugriff auf die Register der Bausteine, als jeden Baustein einzeln anzusprechen.

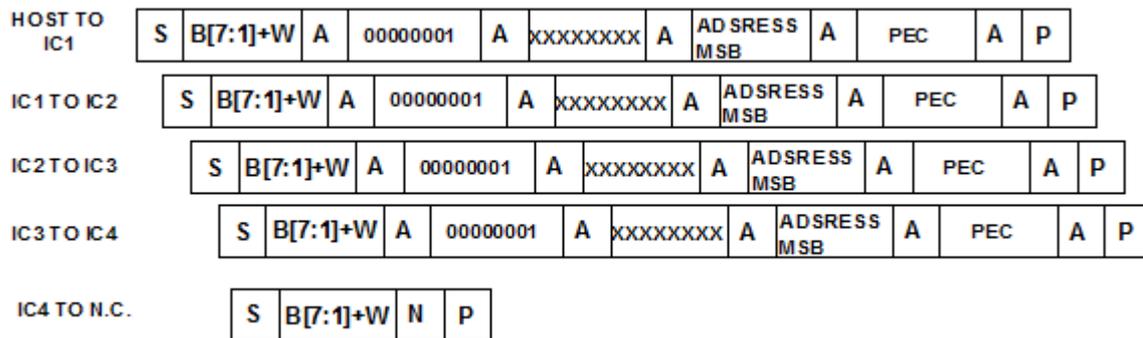


Abbildung 75: Bit-Format des "WRITEALL"-Befehls des MAX11068

### 7.3.1.5.4.5 „READALL“ Befehl

Dieser Befehl wird verwendet um Register-Informationen aus einem Stapel von Bausteinen abzurufen, das ist der einzige Weg um Register-Werte auszulesen (mit Ausnahme des Adressregisters welches mit dem „ROLLCALL“-Befehl ausgelesen wird).

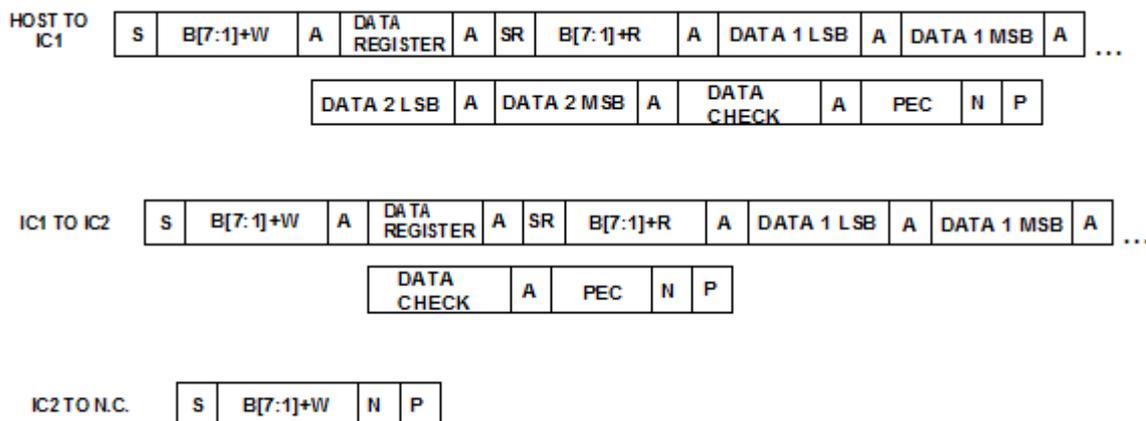


Abbildung 76: Bit-Format des "READALL"-Befehls des MAX11068

### 7.3.1.5.4.6 „WRITEDEVICE“ Befehl

Dieser Befehl ermöglicht es in ein Register eines Bausteines, innerhalb der SMBus-Kette, zu schreiben. Der Befehl ist dem „WRITEALL“-Befehl sehr ähnlich außer dass das I2C-Adress-Byte das feste MSB 0b10 enthält, gefolgt von der Baustein-Adresse A[4:0].

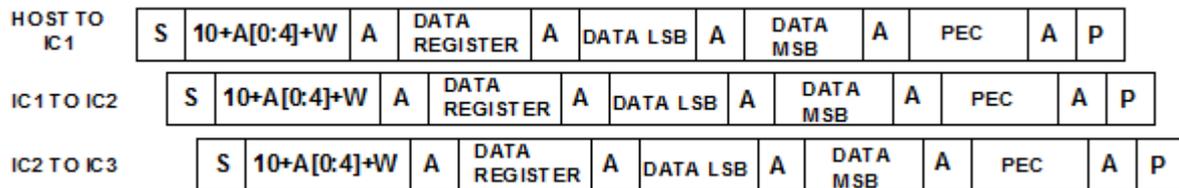


Abbildung 77: Bit-Format des "WRITEDEVICE"-Befehls des MAX11068

### 7.3.1.6 Beschreibung der Register des MAX11068

Tabelle 5: Die Register des MAX11068

Register Adresse	Register Name	R/W	Beschreibung
0x00	VERSION	R	Beinhaltet kodierte Informationen über das Bausteinmodell, Nummer und Version
0x01	ADRESSE	R/W	Das Auslesen dieses Registers liefert die I2C-Adresse des Bausteines und die Baustein-Adresse des letzten Bausteines in der SMBus-Kette.
0x02	STATUS	R/W	Zum Lesen der „status flags“, um „status flags“ zu löschen 0s schreiben.
0x03	ALRTCELL	R/W	Zum Lesen der Zellen-Alarm „status flags“
0x04	ALRTOVCELL	R/W	Zum Lesen der Überspannung Zellen-Alarm „status flags“
0x05	ALRTUVCELL	R/W	Zum Lesen der Unterspannung Zellen-Alarm „status flags“
0x06	ALRTOVEN	R/W	Überspannung Zellen-Alarm ermöglicht für Zellen 1-12
0x07	ALRTUVEN	R/W	Unterspannung Zellen-Alarm ermöglicht für Zellen 1-12
0x08	ADCCFG	R/W	AUX Kanal ermöglichen, Alarm ermöglichen und „scan“ Kontrolle
0x09	CELLEN	R/W	Zellen Messung ermöglichen
0x0A	GPIO	R/W	GPIO2 zu GPIO0 Konfiguration
0x0B	BALCFG	R/W	Zellen-Balancierungs Schalter Steuerung
0x0C	ACQCFG	R/W	Übernahme Zeit kontrol Register für die „auxiliary“ Analog-Eingänge
0x0D	SCANCTRL	R/W	Messung-Scan Steuerung
0x0E	FMEA	R/W	„Failure-mode effects“ Analyse, Status und Kontrolle
0x0F	BROADCAST ADRESSE	R/W	Broadcast Adresse

<b>0x10</b>	TOTAL	R	Ergebnis für die Gesamtsumme aller CELLE <sub>N</sub> Messungen
<b>0x11</b>	MAXCELL	R	Ergebnisregister für die höchste/maximale gemessenen Zellenspannung während eines Scans
<b>0x12</b>	MINCELL	R	Ergebnisregister für die kleinste/minimale gemessene Zellenspannung während eines Scans
<b>0x18</b>	OVTHRCLR	R/W	Überspannungs-Schwelle löschen
<b>0x19</b>	OVTHRSET	R/W	Überspannungs-Schwelle setzen
<b>0x1A</b>	UVTHRSET	R/W	Unterspannungs-Schwelle setzen
<b>0x1B</b>	UVTHRCLR	R/W	Unterspannungs-Schwelle löschen
<b>0x1C</b>	MSMTCH	R/W	Zellen-„mismatch“-Schwelle
<b>0x1E</b>	AINOT	R/W	AUX-Eingang Übertemperatur-Schwelle
<b>0x1F</b>	AINUT	R/W	AUX-Eingangs Untertemperatur-Schwelle
<b>Messung</b>			
<b>0x20</b>	CELL1	R/W	Ergebnisregister für die ADC-Wandlung von C1
<b>0x21</b>	CELL2	R/W	Ergebnisregister für die ADC-Wandlung von C2
<b>0x22</b>	CELL3	R/W	Ergebnisregister für die ADC-Wandlung von C3
<b>0x23</b>	CELL4	R/W	Ergebnisregister für die ADC-Wandlung von C4
<b>0x24</b>	CELL5	R/W	Ergebnisregister für die ADC-Wandlung von C5
<b>0x25</b>	CELL6	R/W	Ergebnisregister für die ADC-Wandlung von C6
<b>0x26</b>	CELL7	R/W	Ergebnisregister für die ADC-Wandlung von C7
<b>0x27</b>	CELL8	R/W	Ergebnisregister für die ADC-Wandlung von C8
<b>0x28</b>	CELL9	R/W	Ergebnisregister für die ADC-Wandlung von C9
<b>0x29</b>	CELL10	R/W	Ergebnisregister für die ADC-Wandlung von C10
<b>0x2A</b>	CELL11	R/W	Ergebnisregister für die ADC-Wandlung von C11
<b>0x2B</b>	CELL12	R/W	Ergebnisregister für die ADC-Wandlung von C12
<b>0x40</b>	AIN1	R/W	Ergebnisregister für die ADC-Wandlung von AUXIN1
<b>0x41</b>	AIN2	R/W	Ergebnisregister für die ADC-Wandlung von AUXIN2
<b>0x44</b>	DIAG	R	Ergebnisregister für die ADC-Wandlung für den Diagnose-Front-End-Test

### 7.3.1.6.1 Als Beispiel das MAXCELL-Registers des Bausteines MAX11068

Tabelle 6: Das "MAXCELL"-Register des MAX11068-Bausteines

BIT	NAME	FUNKTION
D15	D11	12-Bit ADC Konvertierungs-Ergebnis der höchsten Zellen-Spannung
D14	D10	
D13	D9	
D12	D8	
D11	D7	
D10	D6	
D9	D5	
D8	D4	
D7	D3	
D6	D2	
D5	D1	
D4	D0	
D3	CH3	Die Zellen-Nummer der Zelle mit der maximalen Zellen-Spannung. Haben mehrere Zellen das gleiche Maximum haben, beinhaltet dieses Feld die Zelle mit der höchsten Zellen-Nummer
D2	CH2	
D1	CH1	
D0	CH0	

## 7.4 Schaltungstechnische Veränderungen am Evaluation Board des MAX11068

Um den Balancierungs-Baustein „MAX11068“ mit einer externen Spannung versorgen zu können, mussten einige schaltungstechnische Veränderungen am „Evaluation Kit“ vorgenommen werden. Denn nur so ist es möglich dass der Baustein auch mit nur einer NiMH-Zelle betrieben werden kann. In der folgenden Skizze ist der Baustein im beschalteten Zustand dargestellt mit den vorgenommenen Veränderungen an der Schaltung:

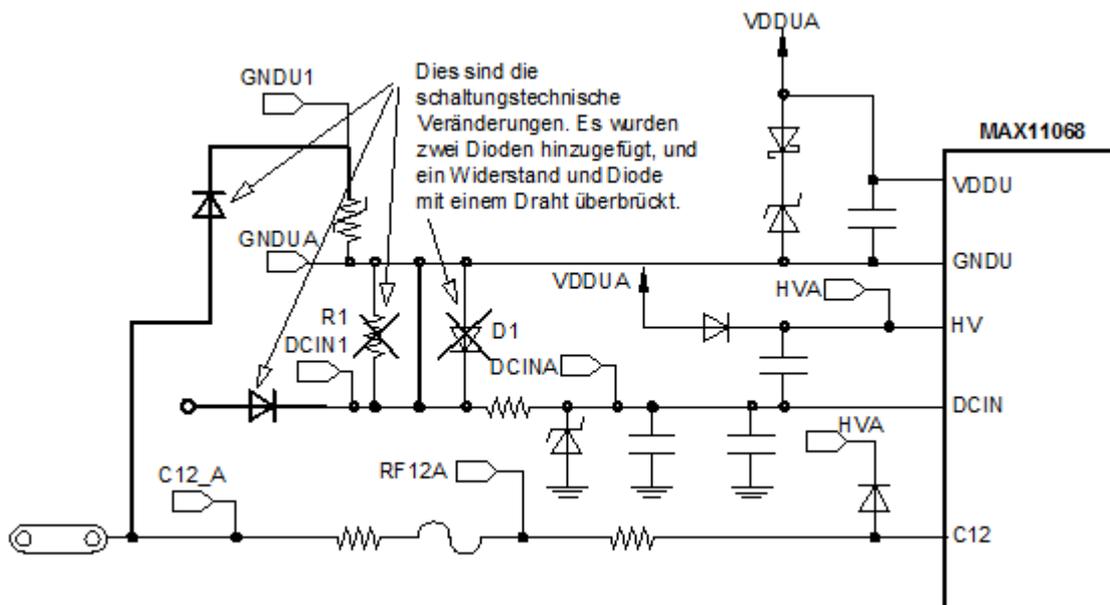


Abbildung 78: Die Schaltungstechnischen Veränderungen vorgenommen am Evaluation-Board des MAX11068

Die Diode  $D_{99}$  ist nötig um zu verhindern dass die Spannung am Anschluss  $C_{12}$ , die Akku-Pack Spannung, durch eine steigende Spannung am Eingang „DCIN“ nach oben gezogen wird (ansteigt). Die Diode  $D_1$  und der Widerstand  $R_1$  sind nicht mehr nötig, weil der Baustein über „DCIN“ nicht mehr über die Akku-Pack-Spannung versorgt wird, sondern direkt mit einer externen Versorgungsspannung. Diese Veränderungen der Schaltung haben es uns ermöglicht den Baustein unabhängig vom Zellen-Typ und Anzahl der Zellen zu machen. Damit wurde erreicht das der Baustein auch beim anschließen nur einer NiMH-Zelle betreiben werden kann.

**Wichtiger Hinweis!**

Der Grund warum die Spannungsversorgung nicht über den „HV“- Eingang erfolgt ist, weil dadurch der Betrieb der „Charge Pump“ beeinträchtigt wird, das bedeutet die „Charge Pump“ braucht für die Funktionsfähigkeit eine mindest Spannung, wenn aber nun die externe Versorgungs-Spannung zu klein sein sollte, ist die „Charge Pumpe“ nicht funktionsfähig und damit könnte das eventuell zu nicht gewünschten Nebeneffekt führen, die vielleicht den Schaltungsbetrieb beeinflussen. Deshalb haben wir uns entschieden die externe Versorgung über den „DCIN“ Eingang zu führen, einfach der um sicher zu sein das der normale Betrieb des Bausteine so wenig wie möglich verändert wird. Somit ist der Betrieb aller Komponenten des Bausteines gewährleistet.

## 8 Der Strommessbaustein DS2764

### 8.1 Allgemeines

<sup>14</sup>Der DS2764 ist ein hoch präzisions Baustein für Lithium-Ionen Batterien zum Batterie-Monitoring und Datenerfassung mit einer Sicherheitsschutzvorrichtung für kostengünstige Batterie-Pack Anwendungen. Dieser Baustein mit niedrigem Energieverbrauch hat integriert eine genaue Temperaturmessung, Spannungs- und Strommessung und zusätzlich einen nichtflüchtigen Datenspeicher zum Abspeichern von Parametern zu denen natürlich auch die akkumulierte Strommenge sprich die Strommenge die in den Akku-Pack geflossen ist zählt bzw. kann der Baustein die restliche Kapazität eines Akku-Packs abschätzen. Durch eine I<sup>2</sup>C-Schnittstelle ermöglicht der DS2764 dem Host-System einen lese und schreib-Zugriff auf die Status und Kontroll-Register, Instrumentierungs-Register und dem Allzweck-Datenspeicher des Bausteines. Die 7-Bit Slave Adresse ist „field programmable“ und ermöglicht dem Host einen Zugriff auf theoretisch 128 Bausteine. Die Auflösung der Temperatur, Spannungs- und Strommessung ist ausreichend für Anwendungen wie Batteriestands-Anzeige, Restkapazität Einschätzung und Sicherheit für Akku-Packs. Die Temperatur wird dabei von einem „On-Chip“-Sensor gemessen wodurch ein separater Thermistor nicht notwendig ist. Die Bidirektionale Strommessung und Akkumulation wird entweder mit Hilfe eines internen 25mΩ Widerstandes oder eines externen Schuntwiderstandes gemessen. Zur Speicherung von wichtigen Batterie-Daten in einem nicht flüchtigen Speicher der auch bei einer Batterieerschöpfung, Kurzschlüssen oder ESD-Ereignissen unbeeinflusst bleibt steht ein EEPROM-Speicher zur Verfügung. Der Speicher mit seinen 40 Byte ist aufgeteilt in zwei 16 Byte Blöcke und einen 8 Byte Block. Jeder Block kann individuell gesperrt oder geschützt werden, dies bietet zusätzliche Sicherheit. Anwendung findet der Baustein in Geräten wie PDA's, Mobil Telefonen, Digitalen Kameras usw.

---

<sup>14</sup> MAXIM DS2764 Datenblatt

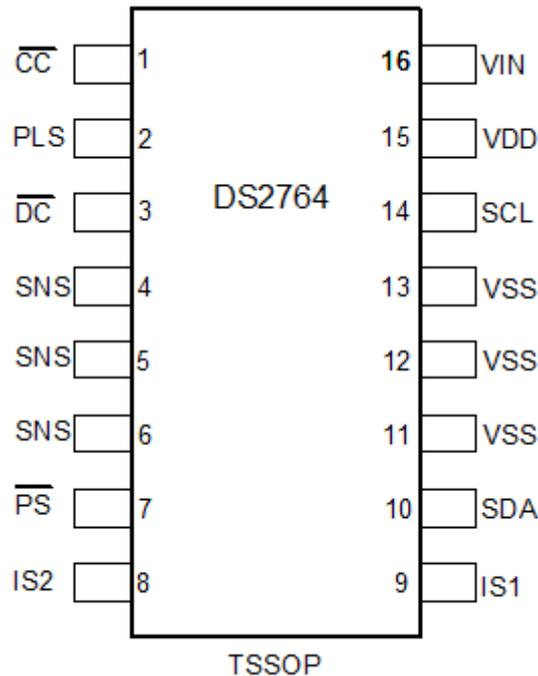


Abbildung 79: Die Gehäuse und Pin-Abbildung des Bausteines DS2764

## 8.2 Die Strommessung

Der DS2764-Baustein misst kontinuierlich den Stromfluss in die und aus der Batterie bzw. den Akku-Pack durch die Messung des Spannungsabfalles über einen Shunt-Widerstand. Dabei gibt es zwei Möglichkeiten

- 1) Die Verwendung des internen 25mΩ Widerstandes.
- 2) Die Verwendung eines externen Benutzerdefinierten Shun-Widerstandes.

In beiden Fällen wird die gefilterte Spannung-Differenz über den Pins  $IS_1$  und  $IS_2$   $V_{SI} = (V_{S1} - V_{S2})$  gemessen. Ein positiver  $V_{SI}$ -Wert ergibt einen Strom der in die Batterie fließt, während ein negativer  $V_{SI}$ -Wert einen Strom ergibt der aus der Batterie fließt. Die Spannung  $V_{SI}$  wird mit einer hohen Auflösung von 12-Bit gemessen. Abbildung 87 zeigt das Datenformat des Stromregisters.

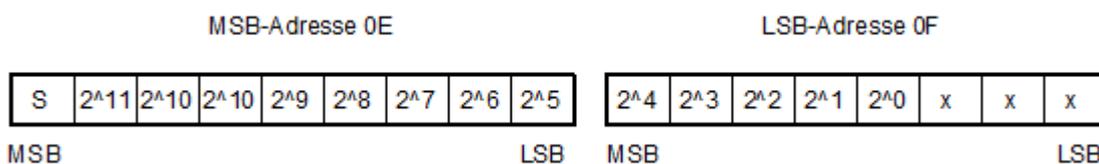


Abbildung 80: Das Register für das Resultat der Strommessung des Bausteines DS2764

## 8.3 Die Strommengenmessung

Der Strom-Akkumulator ermöglicht die Abschätzung der Restkapazität des Akku-Packs durch den Stromflusse in und aus der dem Akku-Pack. Ein Stromfluss in den Akku-Pack erhöht den Wert im Strom-Akku-Register, während ein Stromfluss aus dem Akku-Pack den Wert im Strom-Akku-Register dekrementiert. Der Wert des Stromes bzw. die Daten werden im Zweierkomplementformat gespeichert. Wird der interne Messwiderstand verwendet speichert der Baustein den Messwert in Ampere-Stunden ab, mit einer Auflösung von 0,25 mAh bei einem Messbereich von bis zu 8,2 Ah. Bei Verwendung eines externen Shunt-Widerstandes arbeitet der Baustein mit einer Auflösung von 205 mAh bei einem Messbereich von bis zu 6,25 Ah.

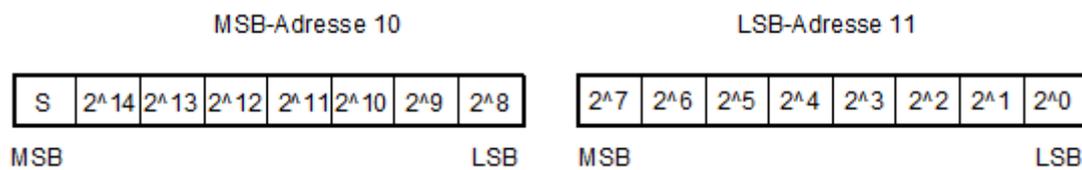


Abbildung 81: Das Register für das Resultat der Strommengen-Messung des Bausteines DS2764

### 8.4 Blockschaltbilde des DS2764 Bausteines

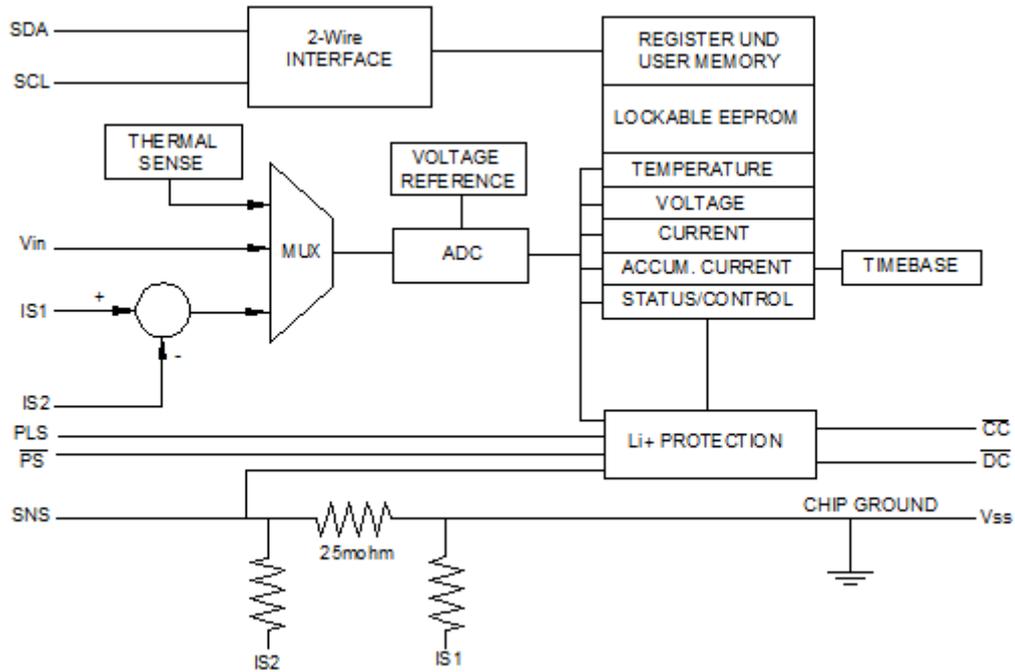


Abbildung 82: Das interne Blockschaltbild des Bausteines DS2764

### 8.5 Anwendungsbeispiel des DS2764

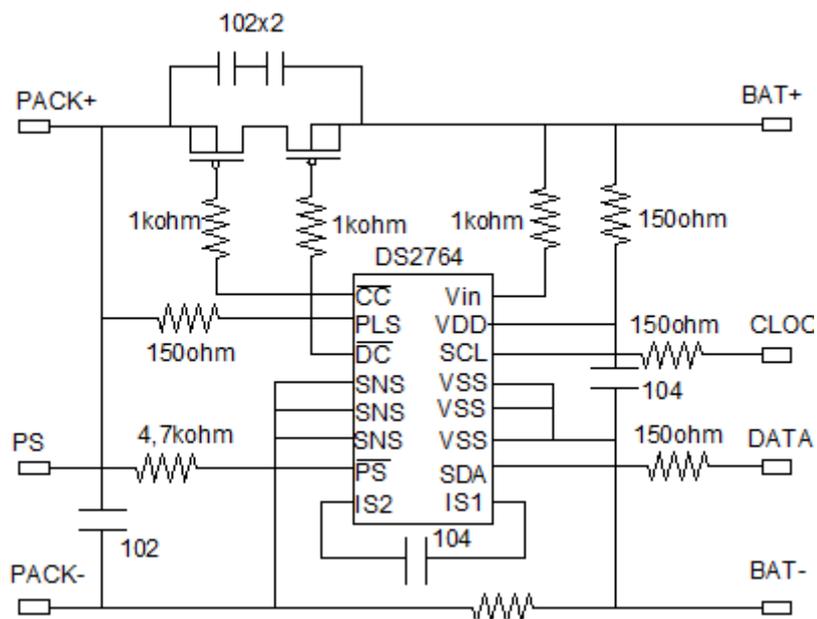


Abbildung 83: Die beispielhafte Beschaltung des Bausteines DS2764



## 8.7 Abbildung des Blockschaltbildes der Schaltung mit den System-Massen

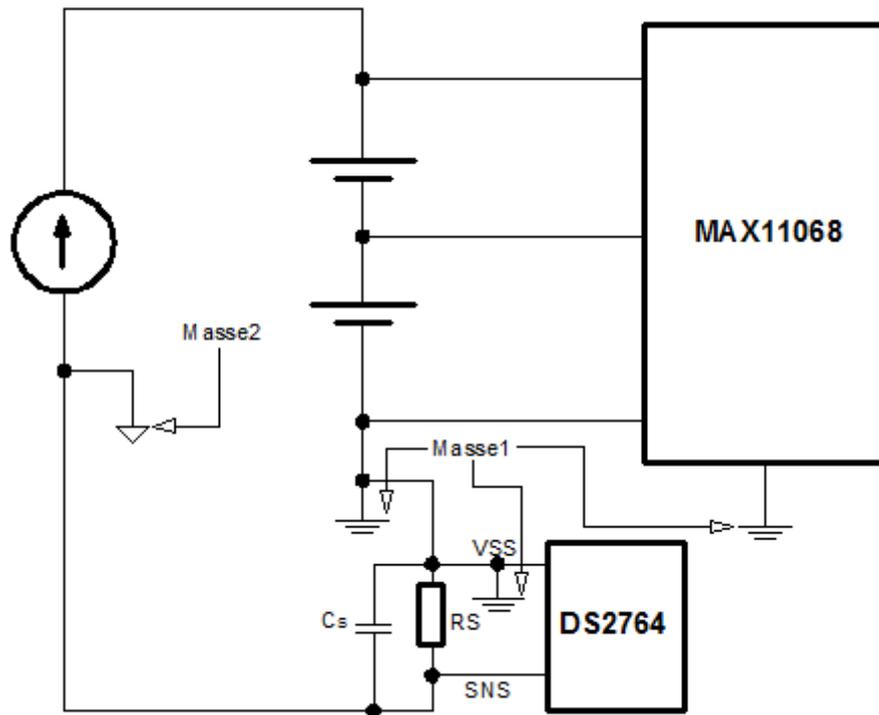


Abbildung 85: Darstellung der beiden Massen der Schaltung

## 9 Der „Charge Pump“- Baustein ICL7660

### 9.1 Allgemeines

<sup>15</sup>Beim Baustein Intersil ICL7660 handelt es sich um eine CMOS Stromversorgungs-Baustein, welcher in der Lage ist eine Versorgungsspannungskonvertierung von einer positiven auf eine negative Spannung durchzuführen. Aus Spannungen von +1,5V bis +10V macht er eine komplementäre Ausgangsspannungen von -1,5V bis -10V. Der ICL7660 Baustein kann auch so beschaltet werden dass er als ein Spannungsverdoppler fungiert der Spannung von +18.7V liefern kann bei +10V Eingangsspannung. Ebenso ist es mit der entsprechenden Beschaltung möglich eine Eingangsspannung zu dividieren oder multiplizieren.

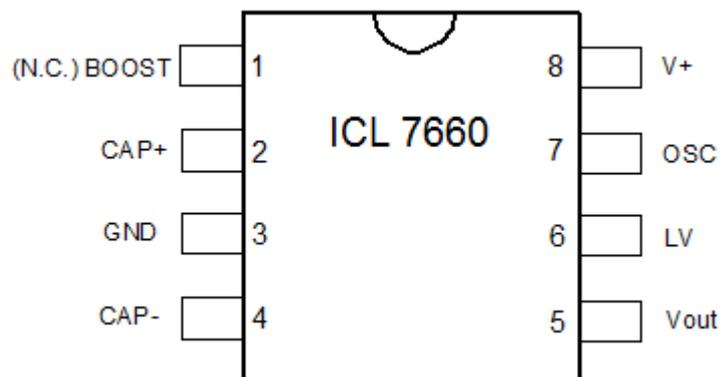


Abbildung 86: Die Gehäuse und Pin-Abbildung des Bausteines ICL7660

### 9.2 Die Funktionsbeschreibung

Das Funktionsprinzip der „Charge Pump“ Spannungskonverter beruht darauf das zuerst eine Ladung in einen Speicherkondensator fließt, von diesem Speicherkondensator wird die Ladung dann in einen Reservoir-Kondensator übertragen. Die Abbildung 95 veranschaulicht diesen Vorgang. In der ersten Hälfte jedes Zykluses sind die Schalter  $S_1$  und  $S_3$  geschlossen und die Schalter  $S_2$  und  $S_4$  geöffnet. Damit ist der Speicherkondensator zunächst auf mit  $V_+$  verbunden und lädt sich auf. In der zweiten Hälfte jedes Zykluses sind die Schalter  $S_2$  und  $S_4$  geschlossen und die Schalter  $S_1$  und  $S_3$  geöffnet. Nun ist der Pluspol des Speicherkondensators mit der Masse verbunden, dadurch kommt es zu einer Umpolung des Kondensators und eine negative Ladung wird zum

<sup>15</sup> MAXIM ICL7660 Datenblatt

$V_{out}$  verschoben. Der Speicherkondensator  $C_1$  liegt jetzt parallel zum Reservoir-Kondensator  $C_2$ . Wenn die Spannung am Kondensator  $C_2$  kleiner ist als die Spannung am Kondensator  $C_1$  fließt eine Ladung von  $C_1$  zum  $C_2$  solange bis sich die Spannungen über den beiden Kondensatoren gleich ist. Wenn der Schalter auf der linken Seite liegt wird der Kondensator  $C_1$  auf  $V_+$  aufgeladen. Wenn der Schalter auf die rechte Seite übergeht wird der Schalter über  $V_{out}$  entladen. Die pro Zyklus übertragene Ladung beträgt  $\Delta Q = C_1 * (V_+ - V_{OUT})$ . Wenn der Schalter mit einer Frequenz  $f$  getaktet wird, beträgt der resultierende Strom  $I = f * \Delta Q = f * C_1 * (V_+ - V_{OUT})$ .

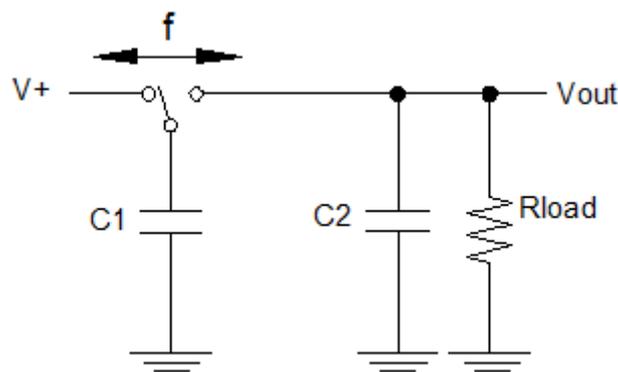


Abbildung 87: Ersatzschaltbild des schaltendem Kondensators des Bausteines ICL7660

Der ersatz Widerstand berechnet sich zu  $Requiv = \frac{1}{f * C_1}$ , wobei  $f$  die Hälfte der Oszillatorfrequenz ist. Dieser Widerstand  $Requiv$  ist ein wesentlicher Bestandteil des Ausgangswiderstandes von „Switch-Capacitor“-Schaltungen wie der des ICL7660.

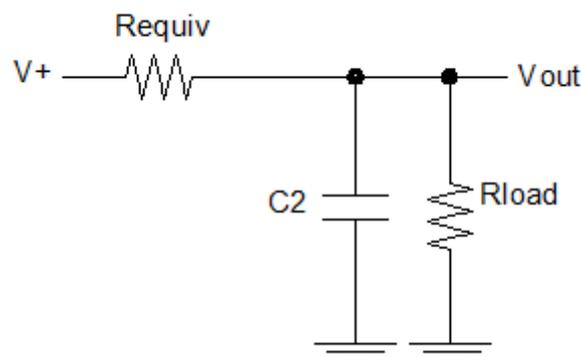


Abbildung 88: Ersatzschaltbild mit Requiv-Widerstand des Bausteines ICL7660

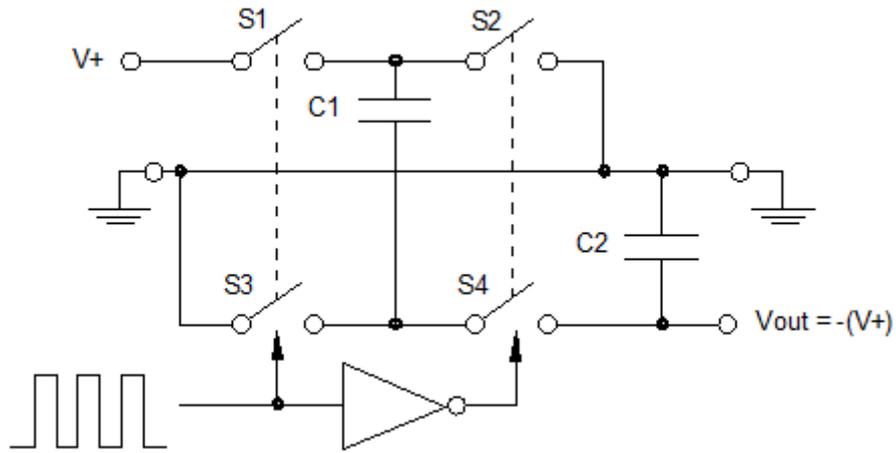


Abbildung 89: Ersatzschaltbild des idealen Spannungsinverters des Bausteines ICL7660

### 9.3 Anwendungsbeispiel des Bausteines ICL7660

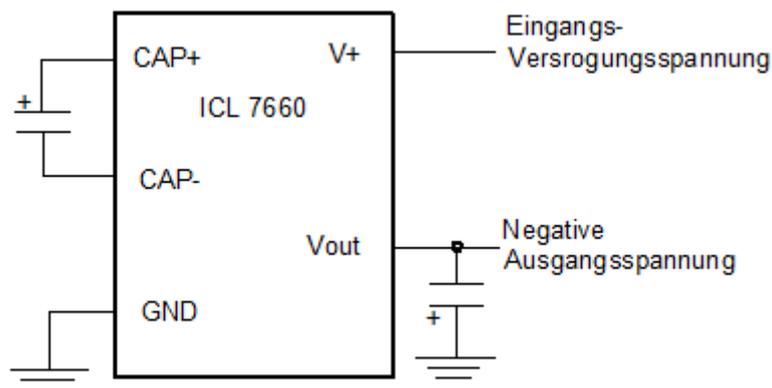


Abbildung 90: Anwendungsbeispiel des Bausteines ICL7660

## 10 Das FTDI-Modul (UM232R mit FT232R-Chip)

### 10.1 Allgemeines

<sup>16</sup>Mit dem FTDI-Modul ist es möglich das zum Beispiel Mikroprozessoren, MCUs und FPGAs mit einem USB-Host (zum Beispiel PC) Daten austauschen können. Dabei übernimmt das FTDI-Modul das komplette USB-Handling (sowohl Protokoll als auch elektrisches Interface). Das Interface zur Mikrocontrollerseite ist seriell (RxD,TxD mit TTL-Pegel) inklusive der kompletten Handshakesignale zur Steuerung des seriellen Datentransfers. Damit wird die Kommunikation zwischen einem Mikroprozessor zu einem USB-Host wesentlich vereinfacht, weil dadurch die Notwendigkeit entfällt ein kompliziertes und aufwendiges USB-Protokoll auf Mikroprozessorseite zu erstellen, da ja auf der Mikroprozessorseite eine relative einfache USART-Schnittstelle verwendet wird.

### 10.2 Schaltung

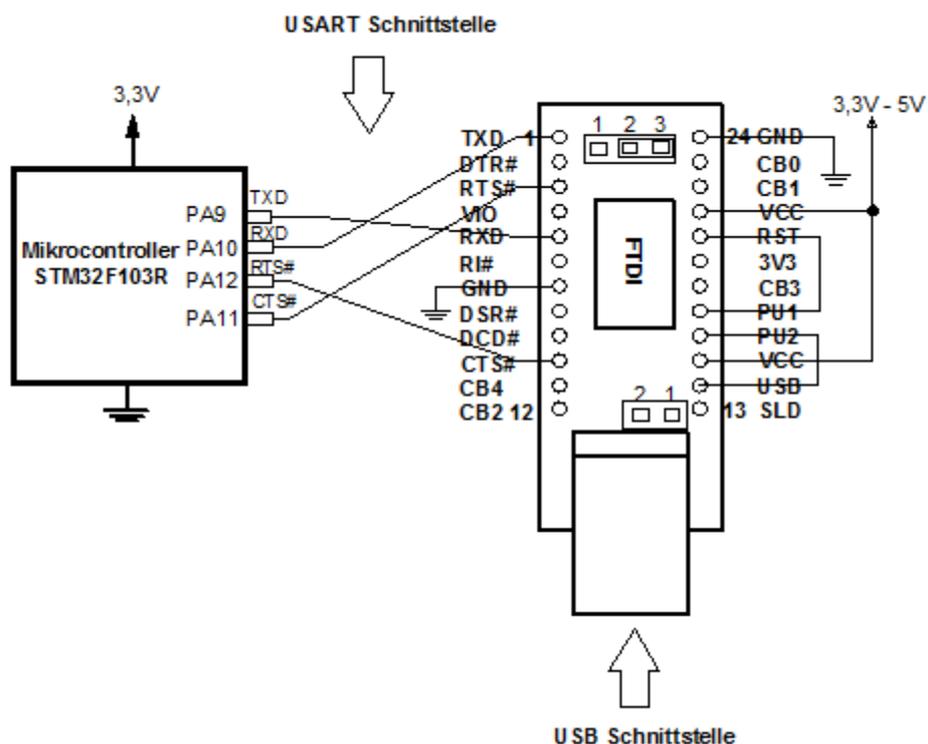


Abbildung 91: Beschaltung und Anwendung des USART-Modules FT232R

<sup>16</sup> FTDI Chip FT232R Datenblatt

## 10.3 Die USART-Schnittstelle

### 10.3.1 Allgemeines

„Universal Synchronous/Asynchronous Receiver Transmitter“ , kurz USART ist ein elektronisches Bauelement welches es ermöglicht eine digitale serielle Schnittstelle zu realisieren. Die USART-Schnittstelle dient zum Senden und Empfangen von Daten über eine Datenleitung, diese bildet den Standard der seriellen Schnittstellen an PCs und Mikrocontrollern. Die Schnittstelle ist mit verschiedenen Interfaces auch im industriellen Bereich sehr verbreitet (zum Beispiel EIA-232 oder EIA-485).

### 10.3.2 Beschreibung der Hardware

Bei der asynchronen Übertragung wird kein eigenes Taktsignal auf einer Übertragungsleitung verwendet. Stattdessen synchronisiert sich der Empfänger über den Übertragungsrahmen, durch die Schaltflanken des Start- und Stopp-Bits, sowie eine eingestellte Bitrate bzw. auch Baudrate genannt. Da der Beginn einer Übertragung mit einem Start-Bit zu einem beliebigen Zeitpunkt erfolgen kann, leitet sich dadurch der Name asynchrone Schnittstelle ab. Damit die Synchronisation bei einem längerem Datenstrom nicht verloren geht, ist die Anzahl der übertragbaren Bits eingeschränkt somit wird sichergestellt dass es zu keiner Fehlerinterpretation des Datenstromes und somit zu einer fehlerhaften Übertragung kommen kann. Die serielle Schnittstelle im synchronen Betrieb arbeitet mit dem Bit-Takt aus der Übertragungsleitung, somit benötigt sie keinen Rahmen aus Start/Stop-Bits.

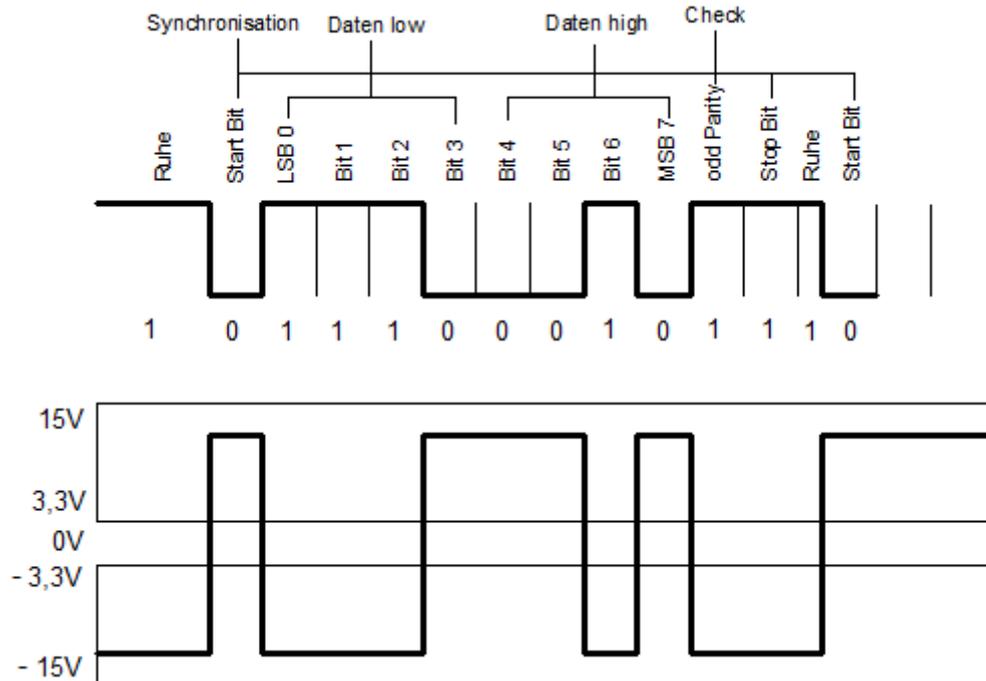


Abbildung 92: Übertragungsprotokoll der USART-Schnittstelle



## 11.1 Warum JTAG

Als der Integrationsgrad von Integrierten Schaltungen immer größer wurde, war es irgend wann unmöglich die tausenden internen Flip Flops und Register auf ihre Funktionsfähigkeit zu testen. Zum Testen der Struktur eines ICs mussten aber die internen Flip Flops und Register steuerbar und beobachtbar werden. Deshalb erfand Eichelberger den als „Scan-Path“ bezeichneten Lösungsansatz. Dabei wird vor jedem Flip Flop-Eingang ein Multiplexer geschaltet, dadurch können alle Flip Flops des ICs zu beliebig zu einem langen Schieberegister zusammenschaltet werden. Somit wird jedes Flip Flop und Register steuerbar und beobachtbar.

## 11.2 Funktionsweise

Das JTAG-System besteht aus folgenden Komponenten:

- Dem „Test Access Port“ (TAP) mit den entsprechenden Steuerleitungen.
- Dem „TAP-Controller“ eine „State Machine“ die die Testlogik steuert.
- Aus zwei Schieberegistern, dem „Instruction Register (IR)“ und dem „Data Register DR“.

## 11.2.1 Das „Test Access Port“

Der „Test Access Port“ besteht aus fünf Steuerleitungen:

1. Dem „Test Data Input“ (TDI). Ist der Serielle Eingang der Schieberegister.
2. Dem „Test Data Output“ (TDO). Ist der Serielle Ausgangs der Schieberegister.
3. Dem „Test Clock“ (TCK). Ist das Taktsignal für die gesamte Testlogik.
4. Dem „Test Select Input“ (TMS). Mit dieser Leitung wird bestimmt in welchen folgenden „State“ die „State Machine“ des „Test Access Port“ bei der nächsten positiven Signalflanke am TCK-Eingang übergeht.
5. Dem „Test Reset“ (TRST). Resetet die Testlogik.

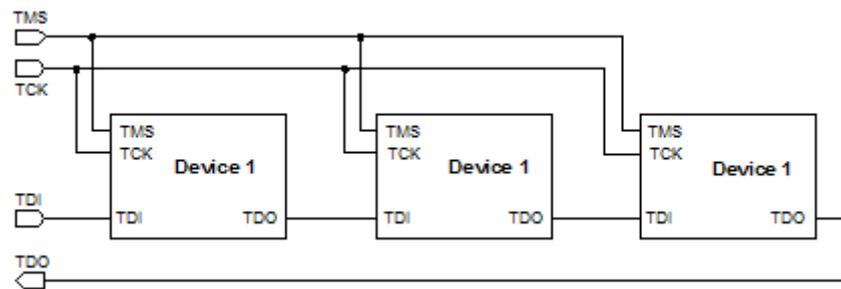


Abbildung 94: Darstellung der JTAG-Chain

## 11.2.2 Der TAP-Controller

Beim TAP-Controller handelt es sich um einen Zustandsautomaten der von der TCK-Leitung getaktet wird. Die TMS-Leitung bestimmt dabei in welchen Folgezustand der Zustandsautomat bei dem nächsten Takt an der TCK-Leitung springen soll. Der TAP-Controller-Zustandsautomat hat sechs stabile Zustände (Zustände in denen der Zustandsautomat verbleiben kann), diese sind: „Test Logic Reset“, „Run-Test/Idle“, „Shift-DR“, „Exit1-DR“, „Shift IR“, „Pause-DR“ und „Pause-IR“. Der Zustand „Test Logic Reset“ setzt die Testlogik zurück, der Zustand „Run Test /Idle“ dient für Wartezeiten. Die Zustände „Shift-IR“ und „Shift-DR“ dienen dazu die beiden Register zu schieben. Die beiden Pause-Zustände diene dazu die Schiebeoperationen zu unterbrechen.

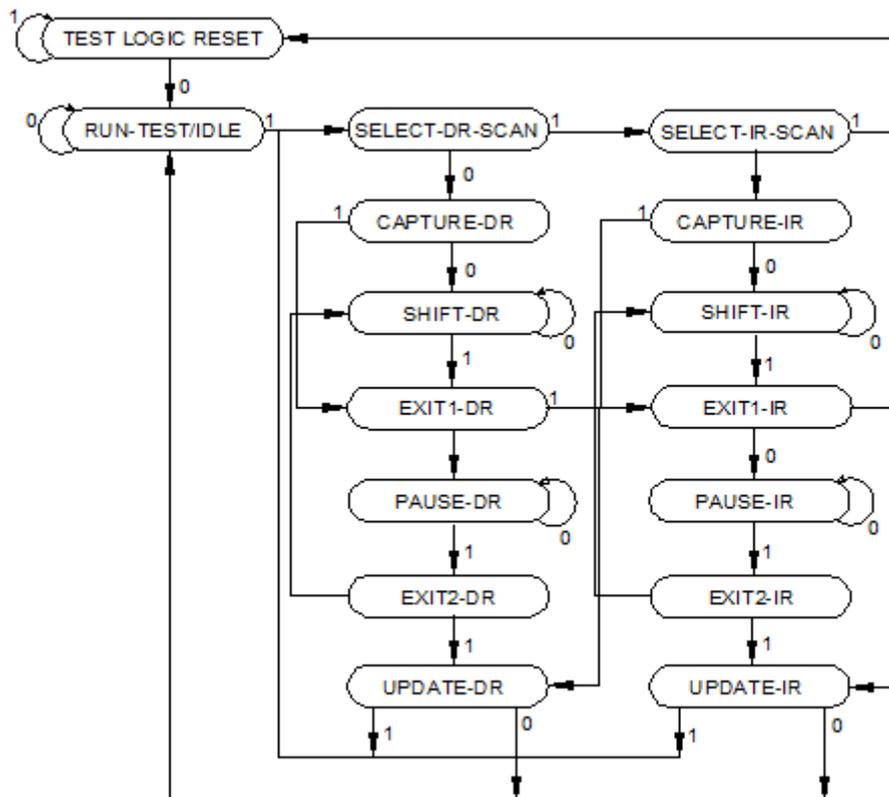


Abbildung 95: Das "State Diagram" des TAP-Controllers

### 11.2.3 Die Register eines JTAG-fähigen Systems

Ein JTAG fähiger Baustein hat unter anderem zwei Register das „Instruction Register“ und das „Data Register“. Das „Data Register“ teilt sich dabei in vier Register auf. Die Bits werden vom TDI-Eingang zum TDO-Ausgang durchgeschoben. Wenn mehrere ICs in einer „JTAG-Chain“ hintereinander geschaltet sind, müssen die Bits die für einen bestimmten IC vorgeschrieben sind erst durch alle anderen IC durchgeschoben werden. Über das „Instruction Register“ wird festgelegt durch welches der vier Datenregister die Bits durchgeschoben werden.

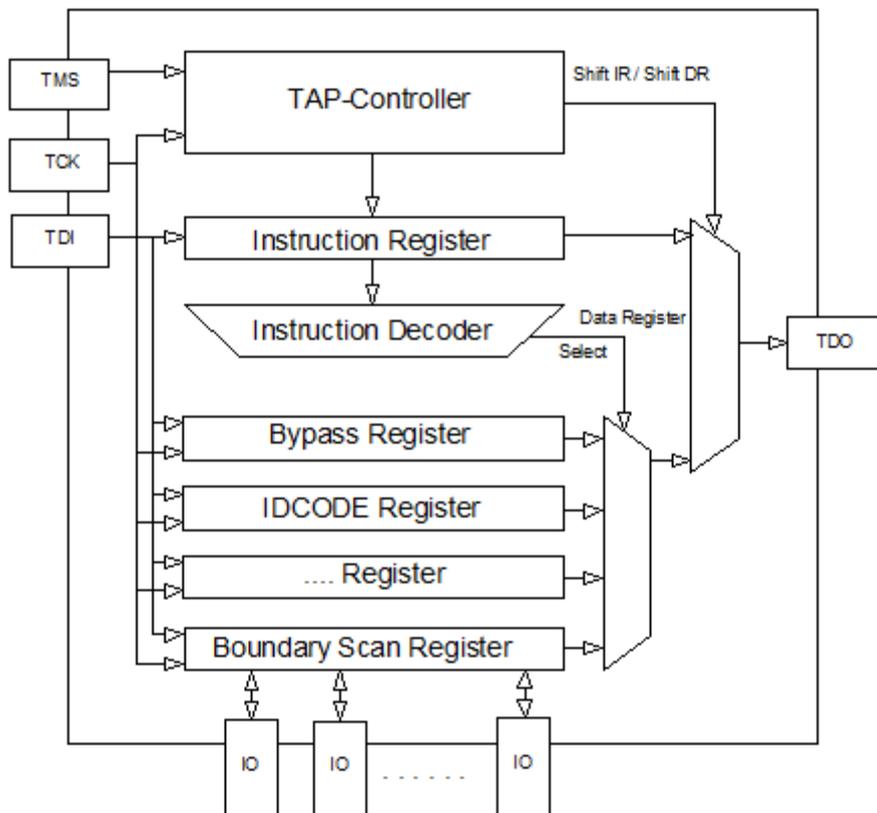


Abbildung 96: Die Darstellung des Register des JTAG Access Ports

## 12 Die Steuerung des Ladegerätes

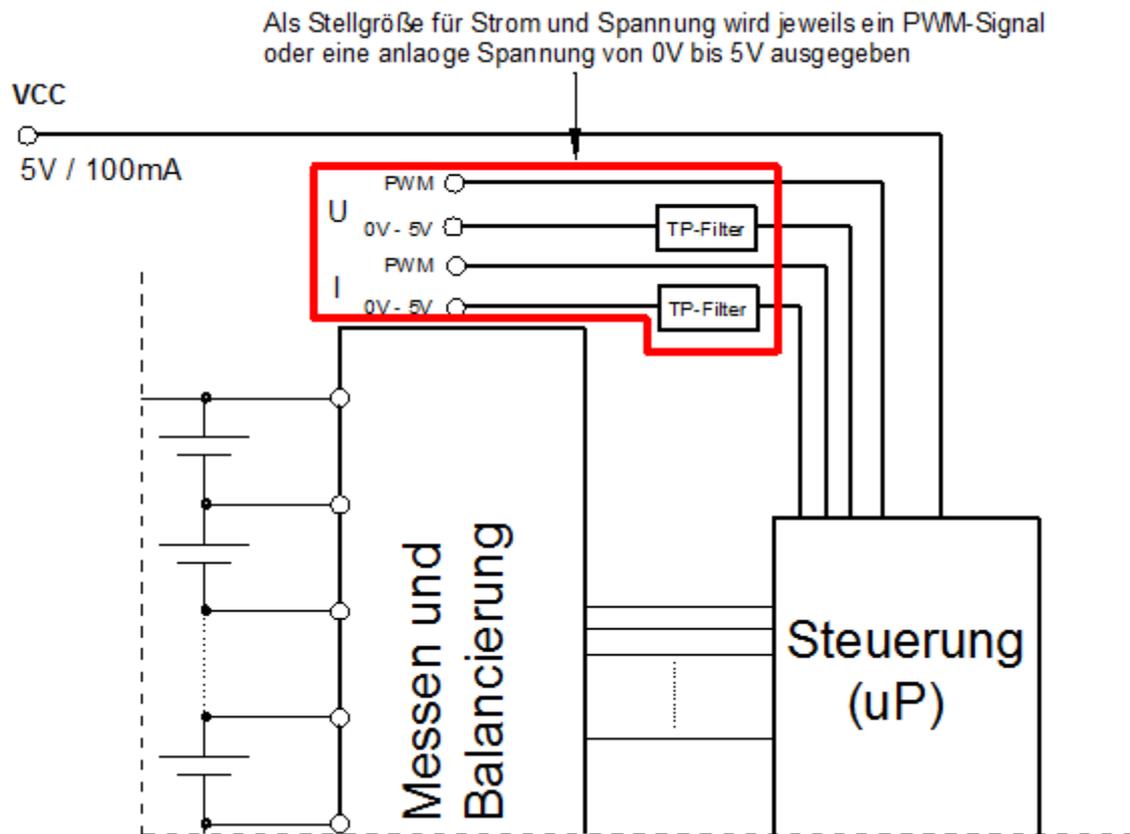


Abbildung 97: Die Gesamtschaltung mit den Stellgrößen für das steuerbare Netzgerät

## 12.1 Prinzipschaltbild der Steuerausgänge

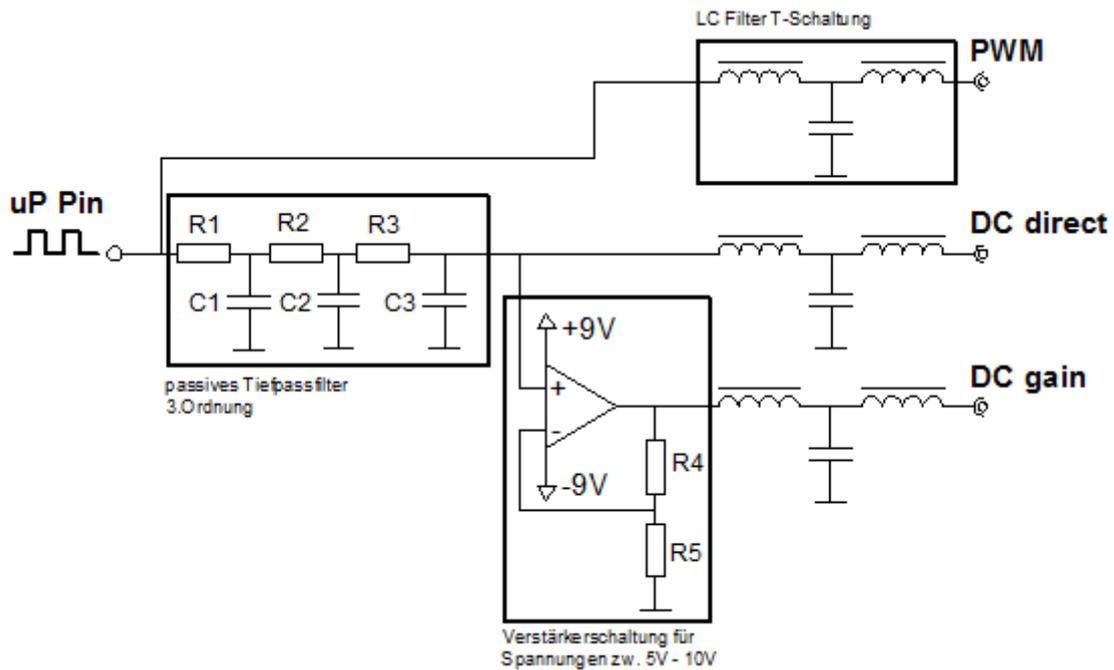


Abbildung 98: Das Blockschaltbild der Steuerausgänge für das steuerbare Netzgerät

## 12.2 Das passive Tiefpassfilter 3.Ordnung

Für die Tiefpassfilterung des PWM Signales wird ein passiver Tiefpassfilter der 3.Ordnung verwendet. Die Widerstände  $R_1$ ,  $R_2$ ,  $R_3$  und die Kapazitäten  $C_1$ ,  $C_2$ ,  $C_3$  wurden nach simplen Verfahren berechnet, bei dem es nicht notwendig ist eine aufwendige Gleichungen aufzustellen und diese zu lösen. Die Werte der Widerstände für den passiven Tiefpassfilter betragen  $R_1 = 1k\Omega$ ,  $R_2 = 10k\Omega$  und  $R_3 = 100k\Omega$ . Der Nachteil dieser Art der Filter ist die gegenseitige Beeinflussung der Einzelpässe und die Ein und Ausgangsbelastung wirken sich negative auf den Maximalpegel aus.

## 12.3 LC Filter, T-Schaltung

Bei der erstmaligen Ansteuerung des steuerbaren Netzgerätes, mit einer Gleichspannung, ist sofort aufgefallen dass das Netzgerät über seine Steuereingänge eine Störgröße aussendet, welche bei den Steuerspannungen zu einem Überschwingeng führt. Um diese Störgröße zu eliminieren wurde ein LC Tiefpassfilter in T-Schaltung an den  $\mu P$ -Steuerausgängen eingefügt. Durch das Fehlen größerer Verlustwiderstände ist die Grunddämpfung im Durchlassbereich viel geringer. Neben diesen Vorteilen ist das

Resonanzverhalten der einfachen LC-Glieder ein Problem. Die Filter müssten auf den angeschlossenen Lastwiderstand angepasst werden.

## 12.4 Verstärkerschaltung

Die nicht invertierende Verstärkerschaltung dient lediglich zum Verstärken der Steuergleichspannungen auf Werte größer 3,3 V.

### 13 Blockschaltbild des Gesamtsystems

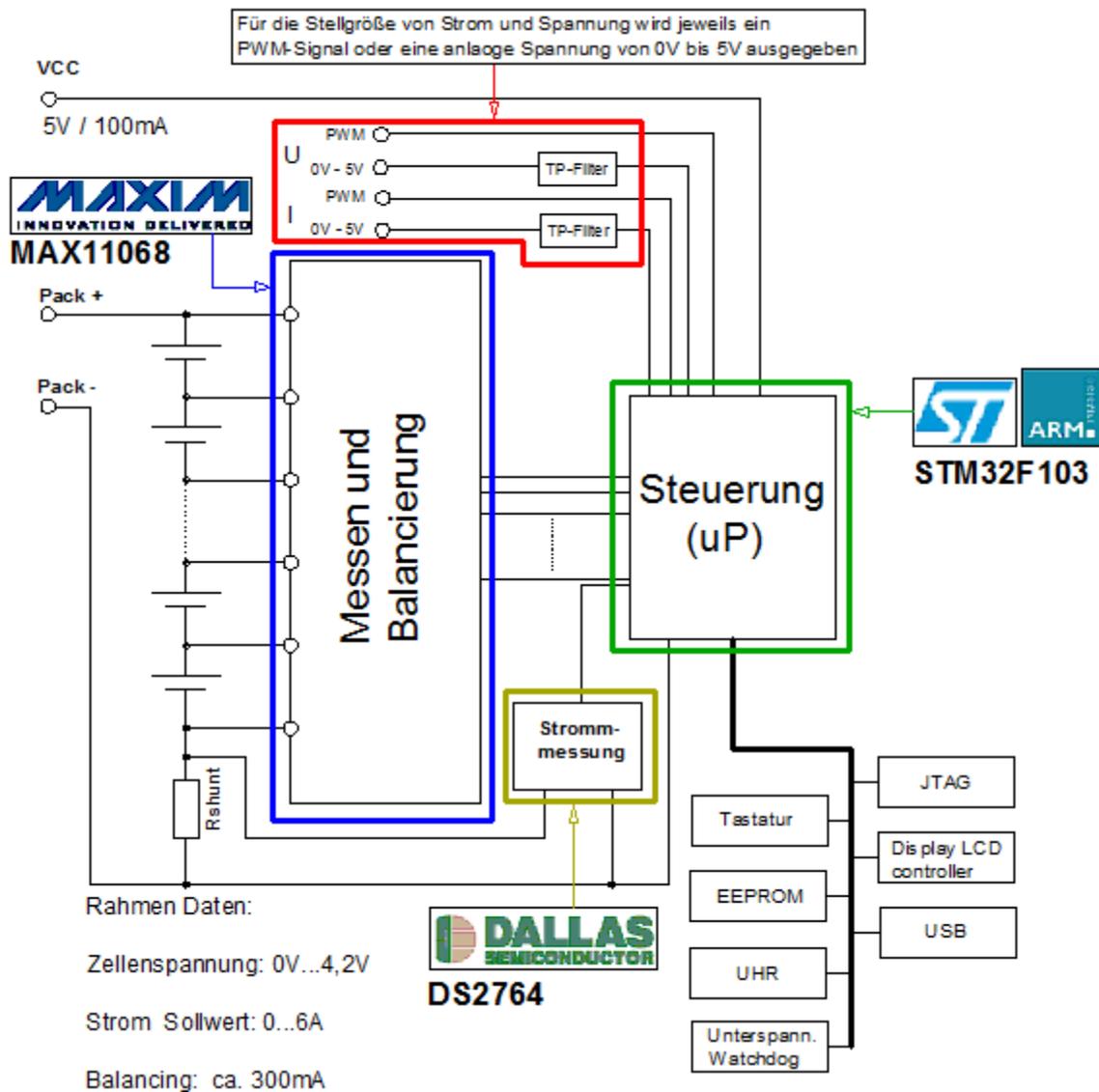


Abbildung 99: Blockschaltbild des Gesamtsystems

### 13.1 Abbildung des Schaltplanes der Messplatine

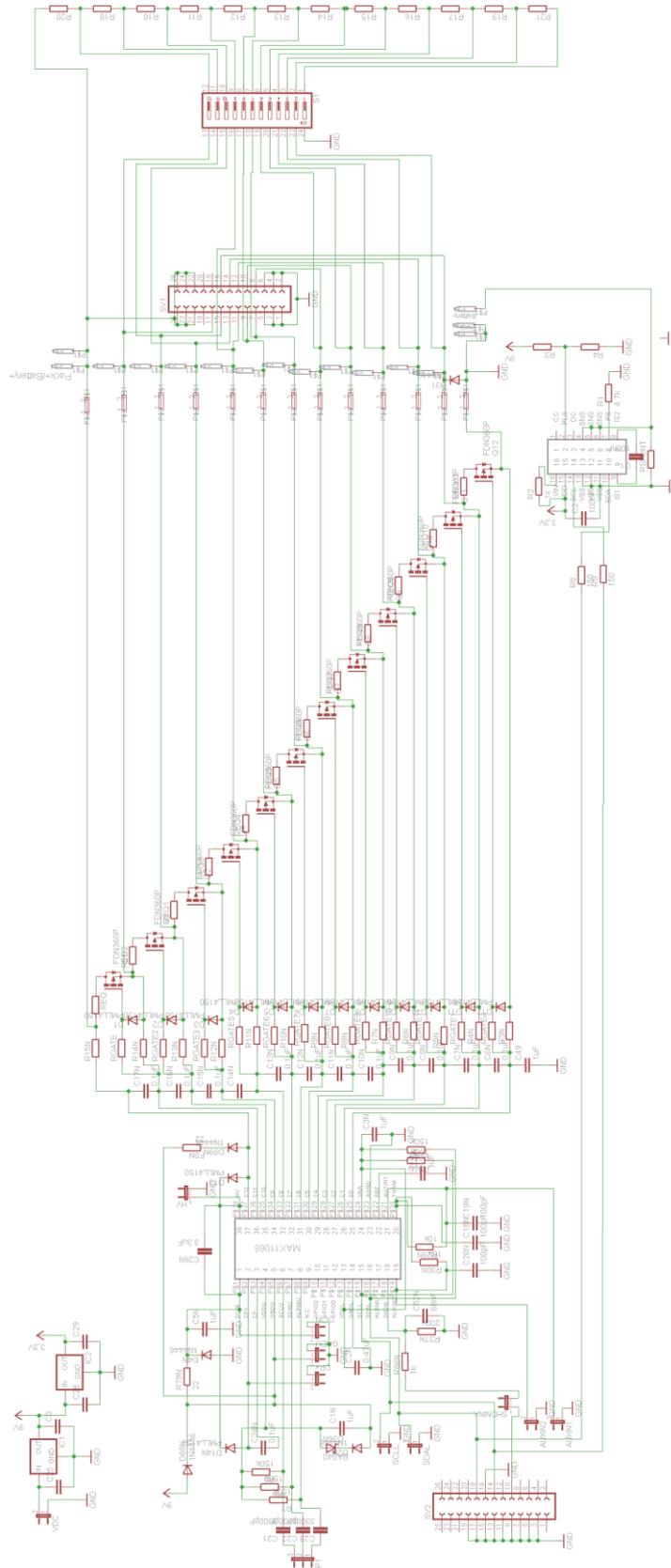


Abbildung 100: Beschaltung des Bausteine MAX11068 und des DS2764

### 13.2 Abbildung des Layouts der Messplatine

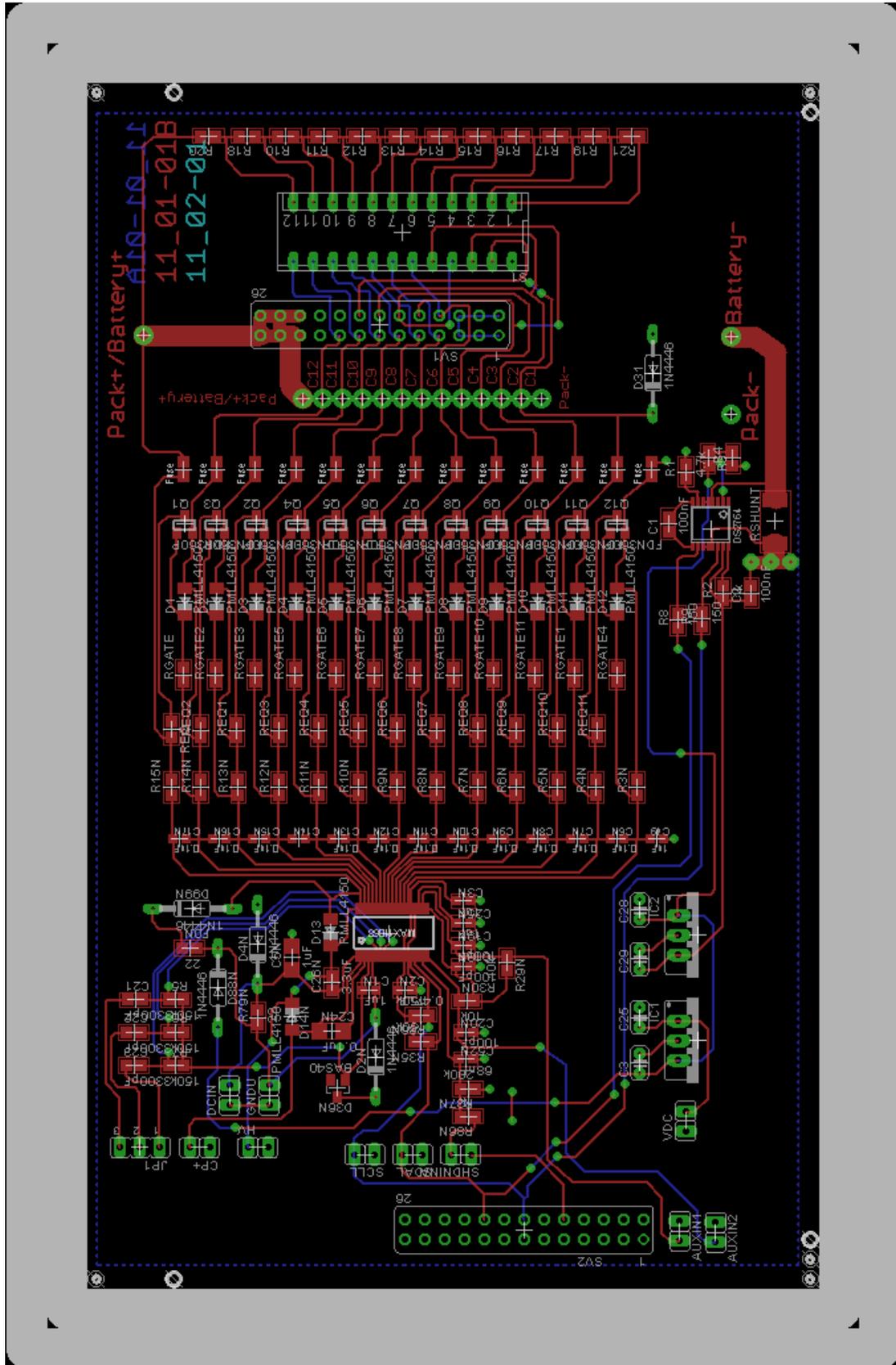


Abbildung 101: Layout der Messplatine mit den Bausteine MAX11068 und DS2764

### 13.3 Abbildung des Schaltplanes der Steuerungsplatine

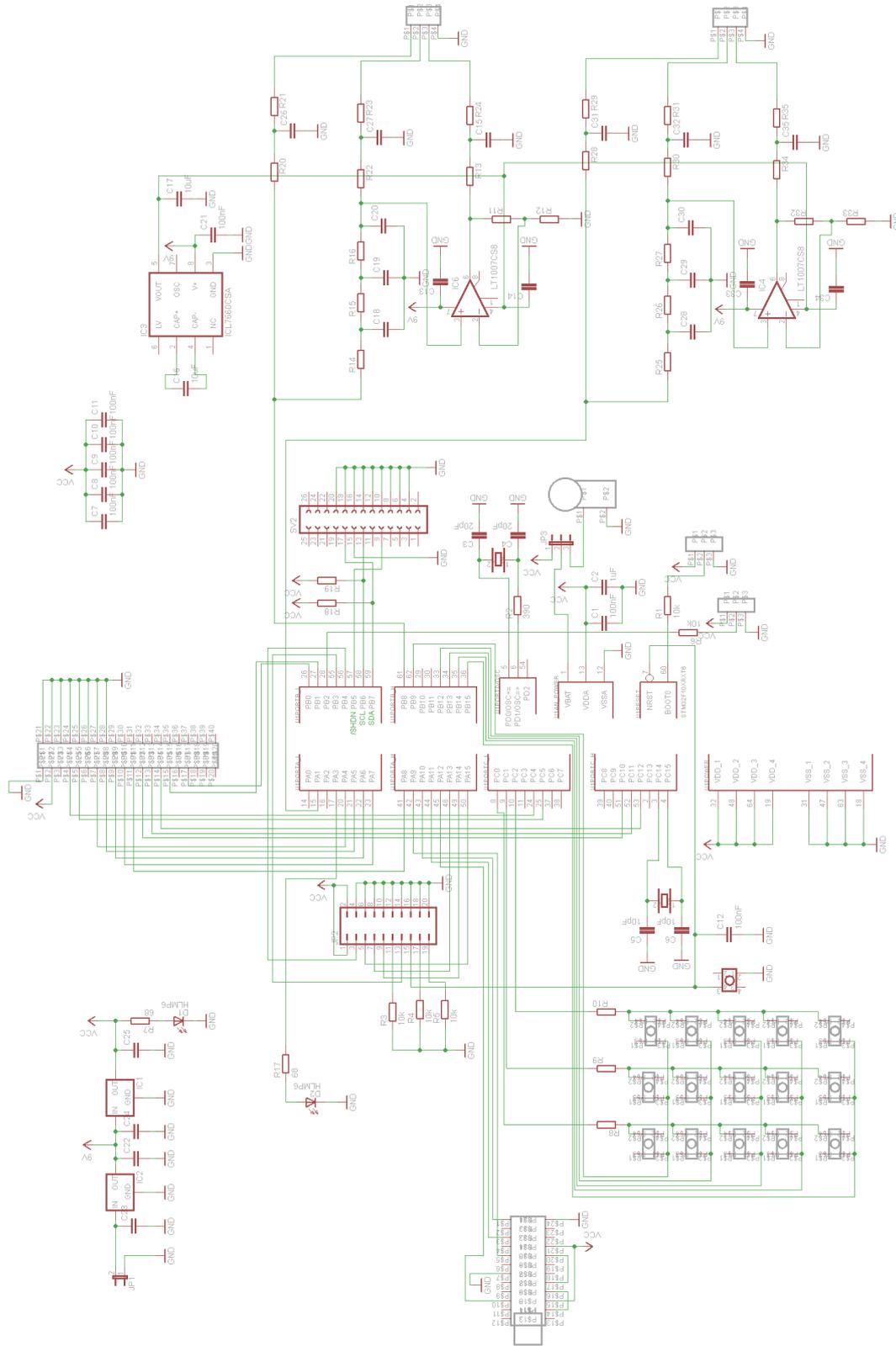


Abbildung 102: Schaltplan der Steuerungsplatine

### 13.4 Abbildung des Layouts der Steuerungsplatine

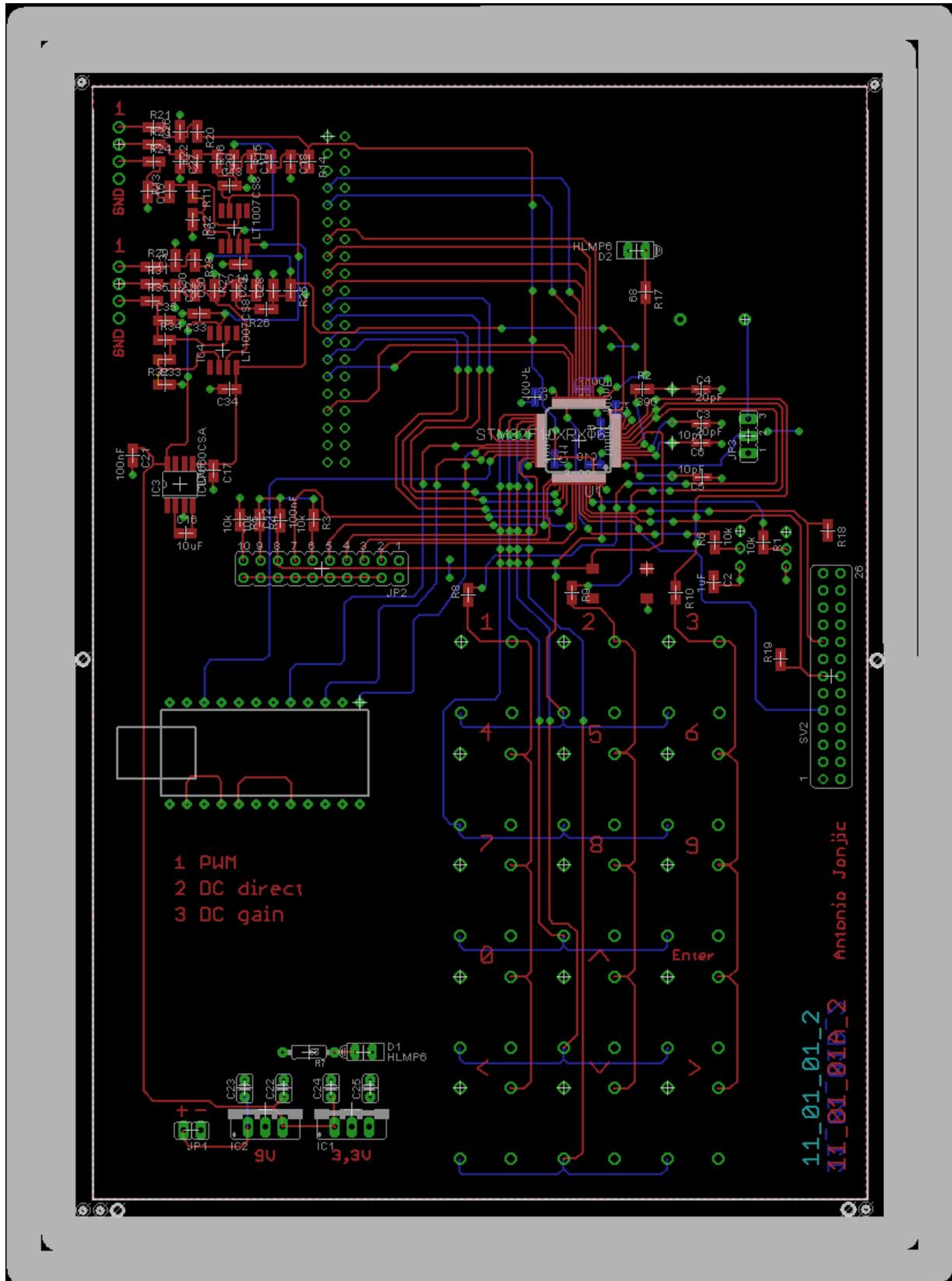


Abbildung 103: Layout der Steuerungsplatine

## 14 Der ARM-Prozessor

### 14.1 Einleitung

Die ARM Prozessor Architektur wurde ursprünglich 1983 von der englischen Firm Acom für eine Familie von 32-Bit-Mikroprozessoren entwickelt, nach dem RISC-Konzept. ARM steht für „*Advanced RISC Machine*“. Es war als Entwicklungsprojekt gestartet worden, von einem Team unter der Leitung von Sophie Wilson und Steve Furber als Nachfolger für die bis dahin auf dem 6502 Prozessor basierenden Computer. Hauptsächliche Unterschiede zu x86-Prozessoren sind das sie kleiner gebaut werden, Energiesparsamer sind und geringere Leistung haben. Bei x86-Prozessoren hat ausschließlich Intel das Patent, während Armprozessoren von einer Vielzahl von verschiedenen Herstellern entwickelt und produziert werden. Durch ihren geringen Energiebedarf haben sich die ARM-Chips vor allem im „embedded computing“ Bereich zur meistgenutzte Architektur entwickelt. Fas alle derzeitigen „Smartphones“ beinhalten einen oder mehrere lizenzierte ARM-Prozessoren.

### 14.2 Eigenschaften

Die ARM-Architektur zeichnet sich vor allem durch einen effizienten Befehlssatz aus und einer guten Optimierung der Ausführungsgeschwindigkeit in Abhängigkeit der Stromaufnahme. Außerdem erlaubt sie eine kompakte Umsetzung in einem ASIC-Design. Da es sich beim der ARM-CPU um eine RISC-Architektur handelt kennt diese nur drei Kategorien von Befehlen:

- Befehle zum Zugriff auf den Speicher (Speichern, Laden)
- Arithmetisch und logische Befehle auf Werte in Registern
- Befehle zur Änderung des Programmflusses (Sprünge)

## 14.3 Besonderheiten des Befehlssatzes

Der ARM Prozessor-Architektur verfügt über einige Besonderheiten welche die Effizienz der Architektur steigern.

- Alle Befehle können bedingt ausgeführt werden:
- Damit könne Programmsprünge vermieden werden, wie z.B. if-else-Abfragen. Programmsprünge leeren die Pipeline des Prozessors und es entstehen Wartezyklen, deswegen vermeidet man diese.
- Die ARM Prozessor-Architektur verfügt über einen Barrel-Shifter im B-Pfad der ALU. Damit ist möglich bei allen Befehlen, die mit zwei Operanden arbeiten, auch die Angabe eines 4-Bit-weiten Shift- oder Roll-Faktors.



### 14.4 Blockschaftbild des STM32F103

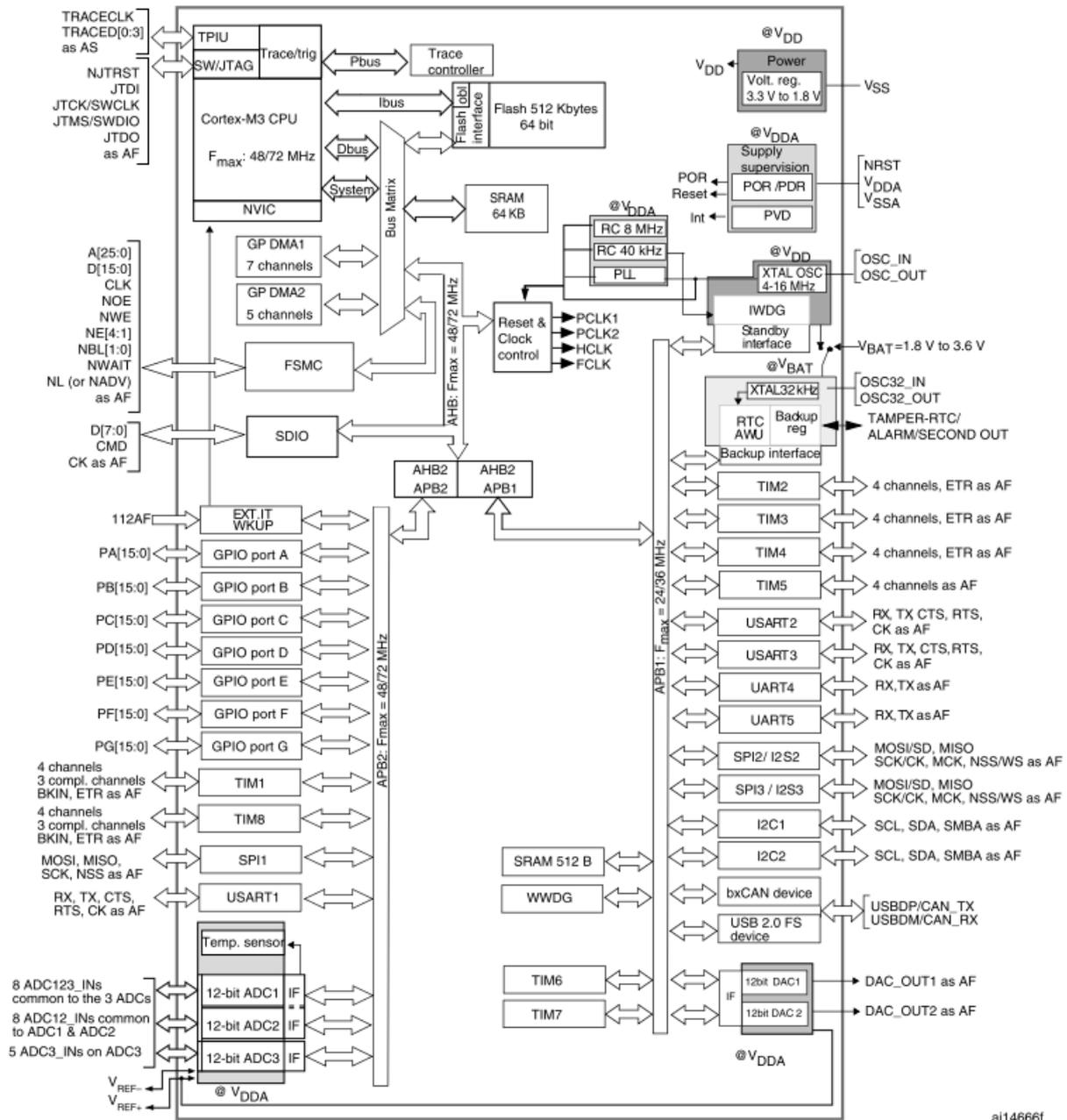


Abbildung 104: Blockschaftbild des STM32F103 ARM-Cortex M3 Mikrocontroller

## 15 Softwaredokumentation

### 15.1 Einbinden der STM32F10x Standard Peripheral Library

1) Downloaden der „STM32F10x\_StdPeriph\_Lib\_V3.5.0“ Datei

2) Öffnen der „ Project.uvproj “ Datei aus dem Ordner

...\stm32f10x\_stdperiph\_lib\STM32F10x\_StdPeriph\_Lib\_V3.5.0\Project\STM32F10x\_StdPeriph\_Template\MDK-ARM

3) In der Menüleiste den Menüpunkt „Project“ öffnen

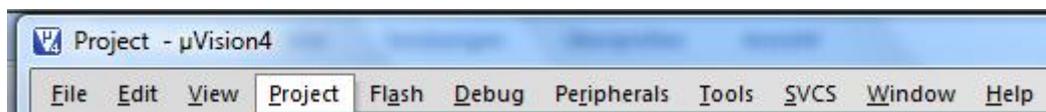


Abbildung 105: Programm Menü der Keil-Entwicklungsumgebung

4) Den Menü-Unterpunkt „Options for Target `STM32100E-EVAL`“ auswählen

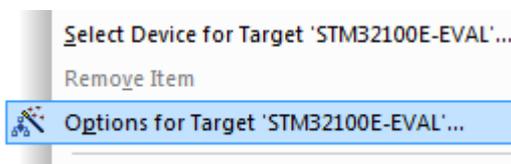


Abbildung 106: Projektuntermenü der Keil-Entwicklungsumgebung

5) Im Konfigurationsfenster „Options for Target `STM32100E-EVAL`“ den Menüpunkt „Output“ auswählen und dort den Optionspunkt „Create Library“ anwählen

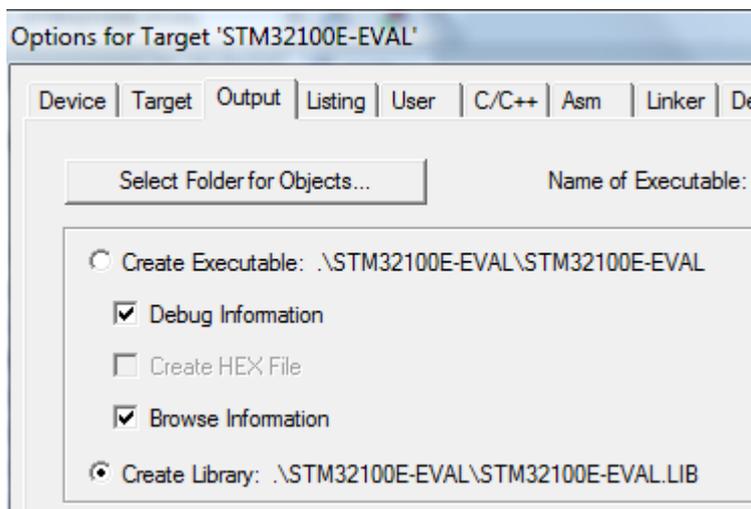


Abbildung 107: Projektoptionsmenü der Keil-Entwicklungsumgebung

- 6) Das Projekt wird nun kompiliert und es wird eine Library-Datei, in dem Ordner „ STM32100E-EVAL“, die „ STM32100E-EVAL.lib“ erzeugt.
- 7) Diese Library-Datei („ STM32100E-EVAL.lib“) wird nun in das eigene Projekt eingefügt

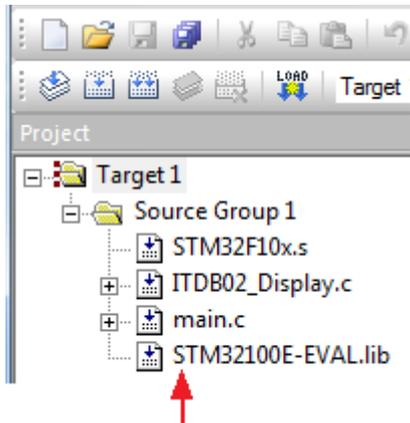


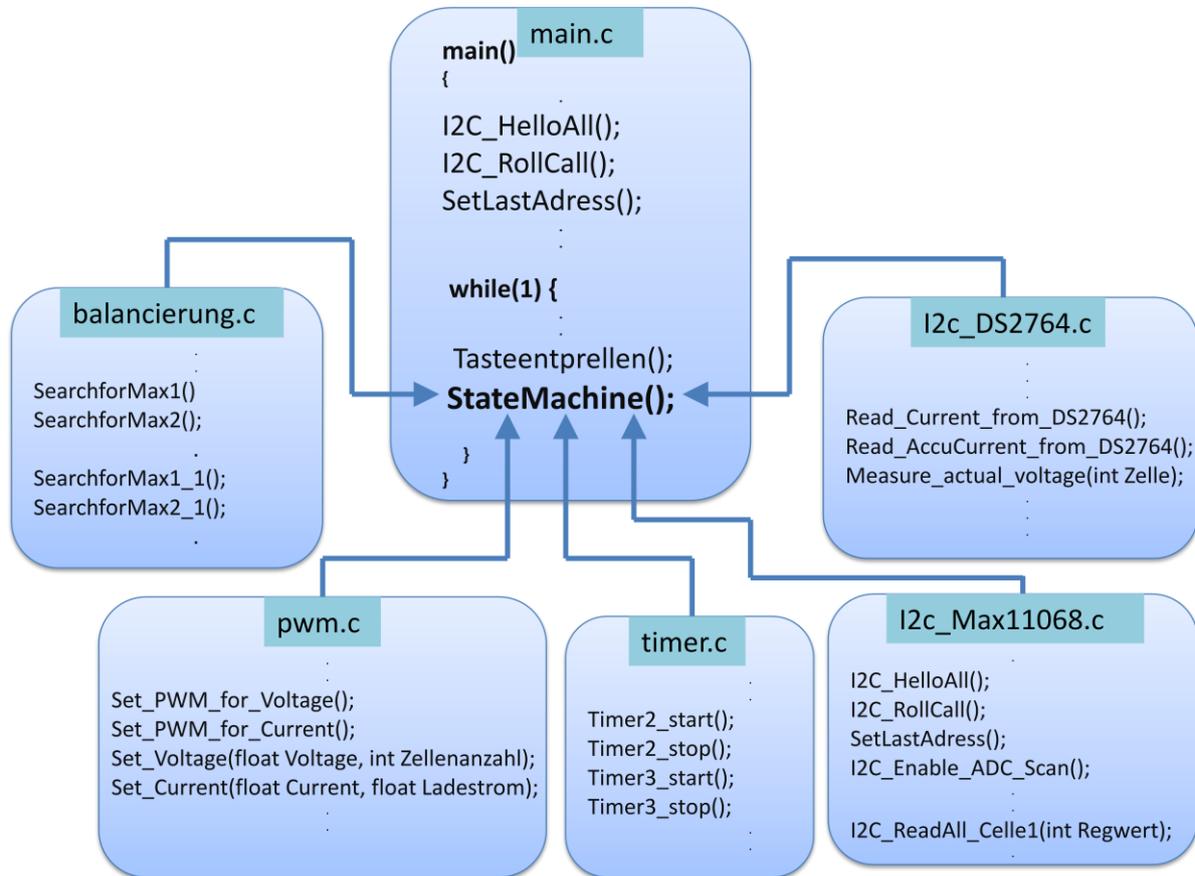
Abbildung 108: Projektbrowser der Keil-Entwicklungsumgebung

- 8) Nun können die Funktion der „Standard Peripheral Library“ benutzt werden, wie zum Beispiel:

- `GPIO_SetBits(GPIOB,GPIO_Pin_1);`
- `GPIO_ResetBits(GPIOB,GPIO_Pin_1);`

Dazu müssen die einzelnen Header-Dateien in der entsprechenden Datei eingebunden werden, wie zum Beispiel `#include <stm32f10x_gpio.h>`

## 15.2 Die Programmstruktur der Firmware



## 15.3 Auszug aus der „main.c“ Datei

```

/*****Einbinden der Header-Dateien*****/
#include <stm32f10x_rcc.h>
#include <stm32f10x_gpio.h>
#include <stm32f10x_i2c.h>
#include <stm32f10x_tim.h>
#include <stm32f10x_nvic.h>

#include <math.h>
#include <stdbool.h>
#include <stdint.h>
#include "main.h"
#include "ITDB02_Display.h"
#include "i2c.h"
#include "pwm.h"
#include "balancierung.h"
#include "timer.h"
#include "i2c_DS2764.h"
/*****globale Variablen deklaration/definitionen und Funktionsdeklarationen*****/

/*****Zeichenbuffer für die sprintf-Funktionen*****/
char buffer1[20];
char buffer2[20];
char buffer3[20];
:

/*****globale Variablen für Ladespannung und Ladestrom*****/
float Ladespannung_Li;
float Ladestrom_Li;
int Zellenanzahl_Li;
float Voltage;
float Current;
float AccuCurrent;
/*****Variablen für die Tastaturfunktion*****/
int PTaste1=0;
int PTaste2=0;
int PTaste3=0;
:
int ccr=0;
int i;
int b;
int d=100;
/*****Variablen für die Zustände der Statemachine*****/
int state=1;
const int Hauptmenue=1;
const int Menue_1=2;
const int Menue_1_1=21;
const int Menue_1_2=22;
const int Menue_2=3;
:

/*****Variablen für die Timerfunktionen*****/
unsigned int counter=6;
/*****Funktionsdeklarationen*****/
void Tasteentprellen(void);
void StateMachine(void);
void SearchforMaxresultat(void);
/*****ENDE*****/

//Beginn der Programmfunktion

int main()
{
InitLCD(); //Initialisieren des TFT-Displays
I2C_InitConf(); //Initialisieren der I2C1 Schnittstelle
I2C_InitConf_DS2764(); //Initialisieren der I2C2 Schnittstelle muss sein da sonst der I2C2-Port den I2C1-Port auf
"LOW" zieht
LCD_Clear(BLUE); //Lösche Bildschirm
LCD_SetCursor(50, 50); //Setze Bildschirmcursor auf position
GPIO_SetBits(GPIOB,GPIO_Pin_5); //SHDN des Bausteines ausschalten, Baustein einschalten

```

```

//delay(1000);                //Eine Sekunde dem Baustein zeit geben

g_draw_string(10,220,"Programmeneue auswaehlen:");
g_draw_string(10,190,"1 Zellen messen");
g_draw_string(10,170,"2 Zellen balancieren");
g_draw_string(10,150,"3 Zellen laden");
g_draw_string(10,130,"4 Versionsregister auslesen");

I2C_HelloAll();                //Überflüssig für die initialisierung da Adresse bereits eingeschrieben ist
I2C_RollCall();                //Überflüssig für die initialisierung da nicht in Register geschrieben wird
SetLastAdress();              //Ist die eigentliche initialisierung des MAX11068, schreibt die LAST Adress ins Adressregister

while(1) {

Tasteentprellen();            //Funktion zum entprellen der Tastatur und einlesen der Tastenzustände
StateMachine();                //Die StateMachine steuert die verschiedenen Programmzustände

}
}

void StateMachine(void)
{
                                //Variablen für die Funktionen der StateMachine als static deklariert
static int i=0;
static int Zellenkapazitaetvariable_Li=0;
static int Zellenkapazitaet_Li=0;
                                :

switch(state){

    case Hauptmenue:
    if(i){
g_draw_string(10,220,"Programmeneue auswaehlen:");
g_draw_string(10,190,"1 Zellen messen");
g_draw_string(10,170,"2 Zellen balancieren");
g_draw_string(10,150,"3 Zellen laden");
g_draw_string(10,130,"4 Versionsregister auslesen");
i=0;
}

    if(PTaste1){
i = 1;
LCD_Clear(BLUE);
state=Menue_1;
}
                                :
    if(PTaste14){
LCD_Clear(BLUE);
i=1;
}
break;

    case Menue_1_1:
    if(i){
I2C_Select_Cells();
I2C_Enable_ADC_Scan();
g_draw_string(10,220,"Messergebnis:");
g_draw_string(10,200,"Zelle1:");
I2C_ReadAll_Celle1(0x20);
g_draw_string(10,185,"Zelle2:");
                                :
break;
                                :
    case Menue_3_1:
    if(i){
g_draw_string(10,220,"Li-Ion Batterie-Lademenu:");
g_draw_string(10,190," Anzahl der Zellen:");
g_draw_string(10,170," Ladestrom:");

```

```

g_draw_string(10,150," Zellenkapazitaet:");
g_draw_string(10,130," Uebernehmen");
g_draw_string(10,80,"Weiter mit Enter");
i=0;
}
if(PTaste14){
g_draw_string(10,80," ");
state=Menue_3_1_1;
}
break;
:
case Menue_3_1_6:
if(i){
g_draw_string(9,220,"Li-Ion Batterie-Ladevorgang:");
g_draw_string(10,190,"Ladeschlussspannung:");
Ladespannung_Li=Zellenanzahl_Li*4.2; //Initialisierung der Ladespannung
sprintf(buffer7,"%0.1fV",Ladespannung_Li);
g_draw_string(190,190,buffer7);
g_draw_string(10,170,"MAX.Ladestrom:");
Ladestrom_Li=Ladestrom1*Zellenkapazitaet_Li; //Initialisierung des Ladestromes
sprintf(buffer8,"%0.0fmA",Ladestrom_Li);
g_draw_string(190,170,buffer8);
i=0;
}
i=1;
state=Menue_3_1_7;
break;

case Menue_3_1_7: //hier beginnt der Ladevorgang
if(i){
g_draw_string(2,120,"Ladevorgang laeuft! ");
g_draw_string(2,90,"Ladespannung:");
Measure_actual_voltage(Zellenanzahl_Li);
g_draw_string(2,70,"Ladestrom:");
Read_Current_from_DS2764();
g_draw_string(2,50,"Gelad.Strommenge:");
Read_AccuCurrent_from_DS2764();
g_draw_string(2,15,"Abbruch mit 0");
i=0;
Timer3_start(); //Ladevorgang beginnt
}
I2C_InitConf(); //Reinitialisierung des I2C Portes um das Datenregister zu löschen, sonst entstehen Probleme
Voltage = Measure_actual_voltage(Zellenanzahl_Li);
Current = Read_Current_from_DS2764();
AccuCurrent = Read_AccuCurrent_from_DS2764();

if(PTaste4){ //Ladevorgang abbrechen
i=1;
I2C_InitConf();
TIM3->CR1 = 0x00;
LCD_Clear(BLUE);
state=Menue_3;
}
break;
:
} //StateMachine Ende
}

```

```

void Tasteentprellen(void)
{
  unsigned char Pin_Wert1=0;
  unsigned char Pin_Wert2=0;
  unsigned char Pin_Wert3=0;
  :
  static int Taste_Vorher1=0;
  static int Taste_Vorher2=0;
  static int Taste_Vorher3=0;
  :
  /*******PortB konfigurierung*****/
  GPIO_InitTypeDef GPIOB_Init;
  GPIOB_Init.GPIO_Pin = GPIO_Pin_9 | GPIO_Pin_12 | GPIO_Pin_13 | GPIO_Pin_14 | GPIO_Pin_15;
  GPIOB_Init.GPIO_Speed = GPIO_Speed_50MHz;
  GPIOB_Init.GPIO_Mode = GPIO_Mode_IPD;

  GPIOB_Init.GPIO_Pin = GPIO_Pin_0 | GPIO_Pin_1 | GPIO_Pin_2;
  GPIOB_Init.GPIO_Speed = GPIO_Speed_50MHz;
  GPIOB_Init.GPIO_Mode = GPIO_Mode_Out_PP;

  /*******Clock Configuration*****/
  RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB, ENABLE); //Port B Clock enable
  GPIO_Init(GPIOB,&GPIOB_Init);

  /*******Einlesen der Port-Pin Eingänge*****/
  GPIO_SetBits(GPIOC,GPIO_Pin_0);
  Pin_Wert1 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_9);
  Pin_Wert2 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_12);
  Pin_Wert3 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_13);
  Pin_Wert4 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_14);
  Pin_Wert5 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_15);
  GPIO_ResetBits(GPIOC,GPIO_Pin_0);
  GPIO_SetBits(GPIOC,GPIO_Pin_1);
  Pin_Wert6 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_9);
  Pin_Wert7 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_12);
  Pin_Wert8 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_13);
  Pin_Wert9 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_14);
  Pin_Wert10 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_15);
  GPIO_ResetBits(GPIOC,GPIO_Pin_1);
  GPIO_SetBits(GPIOC,GPIO_Pin_2);
  Pin_Wert11 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_9);
  Pin_Wert12 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_12);
  Pin_Wert13 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_13);
  Pin_Wert14 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_14);
  Pin_Wert15 = GPIO_ReadInputDataBit(GPIOB, GPIO_Pin_15);
  GPIO_ResetBits(GPIOC,GPIO_Pin_2);
}

  /*******Entprellung*****/
  if(Pin_Wert1==1 && Taste_Vorher1==0){
    PTaste1=1;
    Taste_Vorher1=7;
  }else{
    PTaste1=0;
    if (Taste_Vorher1>1) Taste_Vorher1--;
    if (Taste_Vorher1==1 && (Pin_Wert1==0)) Taste_Vorher1=0;
  }
  if(Pin_Wert2==1 && Taste_Vorher2==0){
    PTaste2=1;
    Taste_Vorher2=7;
  }else{
    PTaste2=0;
    if (Taste_Vorher2>1) Taste_Vorher2--;
    if (Taste_Vorher2==1 && (Pin_Wert2==0)) Taste_Vorher2=0;
  }
  if(Pin_Wert3==1 && Taste_Vorher3==0){
    PTaste3=1;
    Taste_Vorher3=7;
  }else{
    PTaste3=0;
    if (Taste_Vorher3>1) Taste_Vorher3--;
  }
}

```

```
if (Taste_Vorher3==1 && (Pin_Wert3==0)) Taste_Vorher3=0;  
:  
}  
} //Taste entprellen Ende
```

## 15.4 Auszug aus der „i2c\_MAX11068.c“ Datei

```

/*****Einbinden der Header-Dateien*****/
#include <stm32f10x_rcc.h>
#include <stm32f10x_gpio.h>
#include <stm32f10x_i2c.h>
#include "balancierung.h"

/* ***** Initialisierung/Konfiguration des I2C Buses***** */
void I2C_InitConf(void)
{
    RCC->APB1ENR |= 0x00200000; //I2C1 Peripherie-Block einschalten
    RCC->APB2ENR |= 0x9; //Alternative Port-Funktion einschalten und IO PortB einschalten
    GPIOB->CRL |= 0xee000000; //Vieleicht auch Fehlerquelle, set PB6 and PB7 as alternative output with open drain
    I2C1->CR2 = 0x2; //Setze "Control register 2"
    I2C1->CCR = 0x50; //Setze "Clock control register", Clock-Frequenz 50kHz
    I2C1->TRISE = 0x2; //Setze "TRISE register"
    I2C1->CR1 = 0x411;
}

void I2C_HelloAll(void)
{
    I2C1->CR1 |= 0x100;
    while(!(I2C1->SR1 & 0x0001)); //Warte bis im Status Register das "Start-Bit" SB=1 gesetzt wurde
    I2C1->DR = 0xE0;
    while(!(I2C1->SR1 & 0x0002)); //Warte bis im Status Register das "Adress sent-Bit" ADDR=1 gesetzt wurde
    while(!(I2C1->SR2 & 0x0001)); //Lese Status Register 2 weil uP darauf wartet
    I2C1->CR1 |= 0x0200; //Setze Stop Bit
}

void I2C_RollCall(void)
{
    unsigned int temp1;
    unsigned int temp2;
    unsigned int temp3;
    unsigned int temp4;
    unsigned int temp5;
    unsigned int temp6;
    unsigned int temp7;

    char buffer1[20];
    char buffer2[20];
    char buffer3[20];
    char buffer4[20];
    char buffer5[20];

    I2C1->CR1 |= 0x100; //setze Startbit
    while(!(I2C1->SR1 & 0x0001)); //Warte bis im Status Register das "Start-Bit" gesetzt wurde
    I2C1->DR = 0x40; //Sende Broadcastadress mit R/W als Write
    while(!(I2C1->SR1 & 0x0002)); //Warte bis im Status Register das "Adress sent-Bit" ADDR=1 gesetzt wurde (durch lesen wird
    es gelöscht)
    while(!(I2C1->SR2 & 0x0001)); //Lese Status Register 2 weil uP darauf wartet
    I2C1->DR = 0x01; //Zu lesendes Register
    while(!(I2C1->SR1 & 0x0004)); //Warte bis Byte gesendet wurde
    I2C1->CR1 |= 0x100; //repeated Startbedingung
    while(!(I2C1->SR1 & 0x0001)); //lösche Start-Bit durch lesen des SR1-Registers
    I2C1->DR = 0x41; //Broadcastadresse mit R/W als Read
    while(!(I2C1->SR1 & 0x0002));
    while(!(I2C1->SR2 & 0x0001)); //lösche ADDR-Bit durch lesen des SR1 und SR2 Registers

    while(!(I2C1->SR1 & 0x40)); //warte bis DR gelesen wurde (RxNE=1)
    temp1 = I2C1->DR;
    while(!(I2C1->SR1 & 0x40)); //warte bis DR gelesen wurde (RxNE=1)
    temp2 = I2C1->DR;
    while(!(I2C1->SR1 & 0x40)); //warte bis DR gelesen wurde (RxNE=1)
    temp3 = I2C1->DR;
    while(!(I2C1->SR1 & 0x40)); //warte bis DR gelesen wurde (RxNE=1)
    temp4 = I2C1->DR;
    while(!(I2C1->SR1 & 0x40));
    temp7 = I2C1->DR; //lese PEC Byte
    I2C1->CR1 |= 0x0200; //setze STOP-Bit
}

```

```

temp5 = (temp1>>1 & 0x3F);
}

void SetLastAdress(void)
{
I2C1->CR1 |= 0x100; //setze Startbit
while(!(I2C1->SR1 & 0x0001)); //Warte bist im Status Register das "Start-Bit" gesetzt wurde
I2C1->DR = 0x40; //Sende Broadcastadress mit R/W als Write
while(!(I2C1->SR1 & 0x0002)); //Warte bis im Status Register das "Adress sent-Bit" ADDR=1 gesetzt wurde (durch lesen wird
es gelöscht)
while(!(I2C1->SR2 & 0x0001)); //Lese Status Register 2 weil uP darauf wartet
I2C1->DR = 0x01;
while(!(I2C1->SR1 & 0x0004));
I2C1->DR = 0x00; //Sende erstes Daten-Byte
while(!(I2C1->SR1 & 0x0004)); //Warte bis erstes Daten-Byte gesendet wurde
I2C1->DR = 0x1F; //Sende zweites Daten-Byte
while(!(I2C1->SR1 & 0x0004)); //Warte bis zweites Daten-Byte gesendet wurde
I2C1->CR1 |= 0x1020; //enable PEC-Berechnung und sende PEC-Byte
I2C1->CR1 |= 0x0200; //setze Stopbit STOP=1
}

void I2C_Select_Cells(void)
{
I2C1->CR1 |= 0x100; //setze Startbit SB=1
while(!(I2C1->SR1 & 0x0001)); //Warte bis im Status Register das "Start-Bit" SB=1 gesetzt wurde
I2C1->DR = 0x40; //Schreibe 0x40 ins Datenregister und starte Übertragung
while(!(I2C1->SR1 & 0x0002)); //Warte bis im Status Register das "Adress sent-Bit" ADDR=1 gesetzt wurde (durch lesen wird
es gelöscht)
while(!(I2C1->SR2 & 0x0001)); //Lese Status Register 2 weil uP darauf wartet
I2C1->DR = 0x09; //Schreibe 0x09 ins Datenregister und starte Übertragung
while(!(I2C1->SR1 & 0x0004));
I2C1->DR = 0xFF; //Low data byte senden Celle1 enable
while(!(I2C1->SR1 & 0x0004)); //High data byte senden
I2C1->DR = 0xFF;
while(!(I2C1->SR1 & 0x0004));
I2C1->CR1 |= 0x1020; //enable PEC-Berechnung und sende PEC-Byte
I2C1->CR1 |= 0x0200; //setze Stopbit STOP=1
}

void I2C_Enable_ADC_Scan(void)
{
I2C1->CR1 |= 0x100;
while(!(I2C1->SR1 & 0x0001));
I2C1->DR = 0x40;
while(!(I2C1->SR1 & 0x0002));
while(!(I2C1->SR2 & 0x0001));
I2C1->DR = 0x0d; //zu schreibendes Register
while(!(I2C1->SR1 & 0x0004)); //Low data byte
I2C1->DR = 0x01; //High data byte
while(!(I2C1->SR1 & 0x0004));
I2C1->DR = 0x00;
while(!(I2C1->SR1 & 0x0004));
I2C1->CR1 |= 0x1020; //enable PEC und sende PEC Byte
I2C1->CR1 |= 0x0200; //setze Stopbit STOP=1
}

void I2C_StatusRegister(void)
{
I2C1->CR1 |= 0x100;
while(!(I2C1->SR1 & 0x0001));
I2C1->DR = 0x40;
while(!(I2C1->SR1 & 0x0002));
while(!(I2C1->SR2 & 0x0001));
I2C1->DR = 0x02; //zu schreibendes Register
while(!(I2C1->SR1 & 0x0004)); //Low data byte
I2C1->DR = 0x00; //High data byte
while(!(I2C1->SR1 & 0x0004));
I2C1->DR = 0x00;
while(!(I2C1->SR1 & 0x0004));
I2C1->CR1 |= 0x1020; //sende PEC Byte
I2C1->CR1 |= 0x0200; //setze Stopbit STOP=1
}

```

```

void I2C_ReadAll_Celle1(int Regwert)
{
  unsigned int temp1;
  unsigned int temp2;
  unsigned int temp3;
  unsigned int temp4;
  unsigned int temp5;
  float temp6;
  char buffer1[20];

  I2C1->CR1 |= 0x100;           //setze Start-Bit
  while(!(I2C1->SR1 & 0x0001)); //lösche Start-Bit durch lesen des SR1-Registers
  I2C1->DR = 0x40;             //Broadcast-Adresse mit R/W als Write
  while(!(I2C1->SR1 & 0x0002)); //Warte bis im SR1-Register das "Adress sent-Bit" ADDR=1 gesetzt wurde (durch lesen
  //wird es gelöscht)
  while(!(I2C1->SR2 & 0x0001)); //Lese das SR2-Register weil uP darauf wartet
  I2C1->DR = Regwert;          //Zu lesende Register Celle1;
  I2C1->CR1 |= 0x100;          //repeated Start Bedingung
  while(!(I2C1->SR1 & 0x0001)); //lösche SB-Bit durch lesen des SR1-Registers
  I2C1->DR = 0x41;             //Broadcast-Adresse mit R/W als Read
  while(!(I2C1->SR1 & 0x0002)); //lösche ADDR-Bit durch lesen des SR1 und SR2 Registers
  while(!(I2C1->SR2 & 0x0001)); //warte bis DR gelesen wurde (RxNE=1)
  while(!(I2C1->SR1 & 0x40));   //DR lesen, kein NACK-Bit notwendig!
  temp1 = I2C1->DR;            //warte bis DR gelesen wurde (RxNE=1)
  while(!(I2C1->SR1 & 0x40));   //DR lesen
  temp2 = I2C1->DR;            //setze Stopbit STOP=1
  I2C1->CR1 |= 0x0200;

  temp3 = temp1>>4;
  temp4 = temp2<<4;
  temp5 = temp4 | temp3;
  temp6=(float)temp5*0.00124;
  Zellenspg1=temp6;
  sprintf(buffer1,"%0.4fV",temp6);
  g_draw_string(90,200,buffer1);
}

```

## 15.5 Auszug aus der „i2c\_DS2764.c“ Datei

```

/*****Einbinden der Header-Dateien*****/
#include <stm32f10x_rcc.h>
#include <stm32f10x_gpio.h>
#include <stm32f10x_i2c.h>
#include "balancierung.h"

/***** Initialisierung/Konfiguration des I2C Buses*****/
void I2C_InitConf(void)
{
    RCC->APB1ENR |= 0x00200000; //I2C1 Peripherie-Block einschalten
    RCC->APB2ENR |= 0x09; //Alternative Port-Funktion einschalten und IO PortB einschalten
    GPIOB->CRL |= 0xee000000; //Vielleicht auch Fehlerquelle, set PB6 and PB7 as alternative output with open drain
    I2C1->CR2 = 0x2; //Setze "Control register 2"
    I2C1->CCR = 0x50; //Setze "Clock control register", Clock-Frequenz 50kHz
    I2C1->TRISE = 0x2; //Setze "TRISE register"
    I2C1->CR1 = 0x411;
}

float Read_Current_from_DS2764(void)
{
    unsigned int temp1;
    unsigned int temp2;
    unsigned int temp3;
    unsigned int temp4;
    unsigned int temp5;

    float temp6;
    float temp7;
    char buffer1[20];

    TIM3->DIER = 0x00; //Der Timer 3 muss vor dem Kommunikationsbeginn abgeschaltet werden da sich sonst das
    //Programm aufhängt
    I2C1->CR1 |= 0x100; //setze Startbit SB=1
    while(!(I2C1->SR1 & 0x0001)); //Warte bis im Status Register das "Start-Bit" SB=1 gesetzt wurde
    I2C1->DR = 0x68; //Schreibe 0x68 ins Datenregister und starte Übertragung
    while(!(I2C1->SR1 & 0x0002)); //Warte bis im Status Register das "Adress sent-Bit" ADDR=1 gesetzt wurde (durch lesen wird
    //es gelöscht)
    while(!(I2C1->SR2 & 0x0001)); //Lese Status Register 2 weil uP darauf wartet
    I2C1->DR = 0x0E; //Lese das AccuCurrent MSB-Register das LSB-Register wird dann automatisch adressiert
    while(!(I2C1->SR1 & 0x0004));
    I2C1->CR1 |= 0x100; //repeated Start Bedingung
    while(!(I2C1->SR1 & 0x0001)); //lösche SB-Bit durch lesen des SR1-Registers
    I2C1->DR = 0x69; //Slave Adresse mit R/W als Read (1)
    while(!(I2C1->SR1 & 0x0002));
    while(!(I2C1->SR2 & 0x0001)); //lösche ADDR-Bit durch lesen des SR1 und SR2 Registers

    I2C1->CR1 |= 0x400; //Acknowledge enable, wichtig!

    while(!(I2C1->SR1 & 0x40)); //seltsam! warte bis DR gelesen wurde (RxNE=1)
    temp1 = I2C1->DR; //DR lesen (MSB-Current)

    //I2C1->CR1 |= 0x800; //NACK position festlegen, NACK nach dem nächstem gelesenen Byte, funktioniert nicht!
    I2C1->CR1 &= 0xFBFF; //ACK löschen erzeugt ein NACK, wichtig! da sonst der Baustein im Endzustand festsetzt

    while(!(I2C1->SR1 & 0x40)); //seltsam! warte bis DR gelesen wurde (RxNE=1)
    temp2 = I2C1->DR; //DR lesen (LSB-Current)

    I2C1->CR1 |= 0x0200; //Stop Bit setzen
    temp4 = temp2>>3;
    temp3 = temp1<<5;
    temp5 = temp4+temp3;
    temp6=(float)temp5*0.000015625;
    temp7=temp6*(1/0.025);
    sprintf(buffer1,"%0.6fA",temp7);
    g_draw_string(160,70,buffer1);
    TIM3->DIER = 0x01; //Der Timer 3 muss hier wieder eingeschaltet werden
    return temp7;
}

```

```

float Read_AccuCurrent_from_DS2764(void)
{
    unsigned int temp1;
    unsigned int temp2;
    unsigned int temp3;
    unsigned int temp4;

    float temp5;
    float temp6;
    char buffer1[20];

    TIM3->DIER = 0x00;           //Der Timer 3 muss vor dem Kommunikationsbeginn abgeschaltet werden da sich sonst das
    //Programm aufhängt
    I2C1->CR1 |= 0x100;         //setze Startbit SB=1
    while(!(I2C1->SR1 & 0x0001)); //Warte bis im Status Register das "Start-Bit" SB=1 gesetzt wurde
    I2C1->DR = 0x68;           //Schreibe 0x68 ins Datenregister und starte Übertragung
    while(!(I2C1->SR1 & 0x0002)); //Warte bis im Status Register das "Adress sent-Bit" ADDR=1 gesetzt wurde (durch lesen wird
    //es gelöscht)
    while(!(I2C1->SR2 & 0x0001)); //Lese Status Register 2 weil uP darauf wartet
    I2C1->DR = 0x10;           //Lese das AccuCurrent MSB-Register das LSB-Register wird dann automatisch adressiert
    while(!(I2C1->SR1 & 0x0004)); //repeated Start Bedingung
    I2C1->CR1 |= 0x100;         //lösche SB-Bit durch lesen des SR1-Registers
    while(!(I2C1->SR1 & 0x0001)); //Slave Adresse mit R/W als Read (1)
    I2C1->DR = 0x69;           //lösche ADDR-Bit durch lesen des SR1 und SR2 Registers
    while(!(I2C1->SR1 & 0x0002));
    while(!(I2C1->SR2 & 0x0001));

    I2C1->CR1 |= 0x400;         //Acknowledge enable, wichtig!

    while(!(I2C1->SR1 & 0x40)); //seltsam! warte bis DR gelesen wurde (RxNE=1)
    temp1 = I2C1->DR;           //DR lesen (MSB-AccuCurrent)

    I2C1->CR1 &= 0xFBFF;        //ACK löschen erzeugt ein NACK

    while(!(I2C1->SR1 & 0x40)); //seltsam! warte bis DR gelesen wurde (RxNE=1)
    temp2 = I2C1->DR;           //DR lesen (LSB-AccuCurrent)

    I2C1->CR1 |= 0x0200;        //Stop Bit setzen

    temp3 = temp1<<8;
    temp4 = temp3+temp2;
    temp5 = (float)temp4*0.00000625;
    temp6 = temp5*(1/0.025);
    sprintf(buffer1,"%0.6fAh",temp6);
    g_draw_string(160,50,buffer1);
    TIM3->DIER = 0x01;
    return temp6;
}

float Measure_actual_voltage(int Zellenanzahl)
{
    unsigned int Regwert;
    float Spannung;
    char buffer1[20];

    switch(Zellenanzahl){

        case 1:
            TIM3->DIER = 0x00;
            I2C_Select_Cells();
            I2C_Enable_ADC_Scan();
            I2C_ReadAll_Celle1B(0x20);
            TIM3->DIER = 0x01;
            Spannung = Zellenspg1;
            break;

        case 2:
            TIM3->DIER = 0x00;
            I2C_Select_Cells();
            I2C_Enable_ADC_Scan();
            I2C_ReadAll_Celle1B(0x20);
            I2C_ReadAll_Celle2B(0x21);
    }
}

```

```
TIM3->DIER = 0x01;
Spannung = Zellenpg1 + Zellenpg2;
break;

case 3:
I2C_Select_Cells();
I2C_Enable_ADC_Scan();
I2C_ReadAll_Celle1B(0x20);
I2C_ReadAll_Celle2B(0x21);
I2C_ReadAll_Celle3B(0x22);
Spannung = Zellenpg1 + Zellenpg2 + Zellenpg3;
break;
:
case 4:
:
break;
:
}
sprintf(buffer1,"%0.4fV",Spannung);
g_draw_string(160,90,buffer1);
return Spannung;
}
```

## 15.6 Auszug aus der „pwm.c“ Datei

```

/*****Einbinden der Header-Dateien*****/
#include <stm32f10x_rcc.h>
#include <stm32f10x_tim.h>
#include <stm32f10x_gpio.h>
#include <stm32f10x_nvic.h>
#include "main.h"
/***** definition der Funktionen für die Stellgrößen*****/

void Set_PWM_for_Voltage(int ccr)
{
    RCC->APB1ENR |= 0x4;           //Timer 4 enable
    RCC->APB2ENR |= 0x8;           //Port B clock enable und (AF-Block enable) wurde hier nicht eingeschaltet?
    GPIOB->CRH |= 0xb;            //Alternate Output Push-Pull

    TIM4->PSC = 0x1;               //Frequenzverteiler -> 40kHz
    TIM4->ARR = 0x64;              //legt die Frequenz des PWM-Signales fest = 100;
    TIM4->CCR3 = ccr;              //legt das Tastverhältniss des PWM-Signales fest
    TIM4->CCMR1 = 0x00;
    TIM4->CCMR2 = 0x68;           //Kanal 3, preload enable und PWM-Mode1
    TIM4->CR1 |= 0x80;            //autoreload
    TIM4->CCER = 0x100;          //Kanal 3 output enable
    TIM4->EGR = 0x1;             //initialisiert alle Register
    TIM4->CR1 |= 0x1;            //starte Timer
}

void Set_PWM_for_Current(int ccr)
{
    RCC->APB1ENR |= 0x1;           //Timer 2 enable
    RCC->APB2ENR |= 0x4;           //Port A clock enable und (AF-Block enable) wurde hier nicht eingeschaltet?
    GPIOA->CRL |= 0xb0;           //Alternate Output Push-Pull

    TIM2->PSC = 0x1;               //Frequenzverteiler -> 40kHz
    TIM2->ARR = 0x64;              //legt die Frequenz des PWM-Signales fest
    TIM2->CCR2 = ccr;              //legt das Tastverhältniss des PWM-Signales fest
    TIM2->CCMR2 = 0x00;
    TIM2->CCMR1 = 0x6800;         //Kanal 2, preload enable und PWM-Mode1
    TIM2->CR1 |= 0x80;            //autoreload
    TIM2->CCER = 0x10;           //Kanal 2 output enable
    TIM2->EGR = 0x1;             //initialisiert alle Register
    TIM2->CR1 |= 0x1;            //starte Timer
}

void Set_Voltage(float Voltage, int Zellenanzahl)
{
    static int ccr=0;

    switch(Zellenanzahl){

        case 1:
            ccr=10;
            break;

        case 2:
            ccr=20;
            break;

        case 3:
            ccr=30;
            break;

        case 4:
            ccr=40;
            break;

        case 5:
            ccr=50;
            break;

        case 6:
    
```

```
    ccr=60;
    break;

    case 7:
    ccr=70;
    break;

    case 8:
    ccr=75;
    break;

    case 9:
    ccr=80;
    break;

    case 10:
    ccr=85;
    break;

    case 11:
    ccr=90;
    break;

    case 12:
    ccr=100;
    break;
}

Set_PWM_for_Voltage(ccr);
}

void Set_Current(float Current, float Ladestrom_Li)
{
    static int ccr=0;

    if (Current < Ladestrom_Li && Current < 4){
        if (ccr<100)ccr++;
        else ccr=0;

        Set_PWM_for_Current(ccr);
    }
}
```

## 15.7 Auszug aus der „timer.c“ Datei

```

/*****Einbinden der Header-Dateien*****/
#include <stm32f10x_rcc.h>
#include <stm32f10x_tim.h>
#include <stm32f10x_nvic.h>
/***** definition der Funktionen für die Stellgrößen*****/

void Timer2_start(void)
{
    /* Initialisieren der Timer-Struktur */
    TIM_TimeBaseInitTypeDef TIM_TimeBaseStructure;

    /* Initialisieren der Interrupt-Struktur */
    NVIC_InitTypeDef NVIC_InitStructure;

    /* Timer 2 Clock enable */
    RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM2, ENABLE);

    /* Timer 2 global Interrupt enable*/
    NVIC_InitStructure.NVIC_IRQChannel = TIM2_IRQChannel;
    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0;
    NVIC_InitStructure.NVIC_IRQChannelSubPriority = 1;
    NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;
    NVIC_Init(&NVIC_InitStructure);

    /* Timer 2 konfiguration */
    TIM_TimeBaseStructure.TIM_Period = 350;
    TIM_TimeBaseStructure.TIM_Prescaler = 32000;
    TIM_TimeBaseStructure.TIM_ClockDivision = 0;
    TIM_TimeBaseStructure.TIM_CounterMode = TIM_CounterMode_Up;

    TIM_TimeBaseInit(TIM2, &TIM_TimeBaseStructure);

    /* Timer 2 interrupt enable */
    TIM_ITConfig(TIM2, TIM_IT_Update, ENABLE);

    /* Timer 2 enable */
    TIM_Cmd(TIM2, ENABLE);
}

void Timer2_stop(void)
{
    TIM2->CR1 = 0x00;           //Timer 2 stoppen
}

void Timer3_start(void)
{
    /* Initialisieren der Timer-Struktur */
    TIM_TimeBaseInitTypeDef TIM_TimeBaseStructure;

    /* Initialisieren der Interrupt-Struktur */
    NVIC_InitTypeDef NVIC_InitStructure;

    /* Timer 3 Clock enable */
    RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM3, ENABLE);

    /* Timer 3 global Interrupt enable*/
    NVIC_InitStructure.NVIC_IRQChannel = TIM3_IRQChannel;
    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0;
    NVIC_InitStructure.NVIC_IRQChannelSubPriority = 1;
    NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;
    NVIC_Init(&NVIC_InitStructure);

    /* Timer 3 konfiguration */
    TIM_TimeBaseStructure.TIM_Period = 350;
    TIM_TimeBaseStructure.TIM_Prescaler = 7500;
    TIM_TimeBaseStructure.TIM_ClockDivision = 0;
    TIM_TimeBaseStructure.TIM_CounterMode = TIM_CounterMode_Up;

    TIM_TimeBaseInit(TIM3, &TIM_TimeBaseStructure);
}

```

```
/* Timer 3 interrupt enable */
TIM_ITConfig(TIM3, TIM_IT_Update, ENABLE);

/* Timer 3 enable */
TIM_Cmd(TIM3, ENABLE);
}

void Timer3_stop(void)
{
TIM3->CR1 = 0x00;           //Timer 3 stoppen
}
```

## 15.8 Auszug aus der „balancierung.c“ Datei

```

/*****Einbinden der Header-Dateien*****/
#include <stm32f10x_rcc.h>
#include <stm32f10x_gpio.h>
#include <stm32f10x_i2c.h>
#include "balancierung.h"
/***** definition der Funktionen für die Balancierung*****/

volatile float Zellenspg1;
volatile float Zellenspg2;
volatile float Zellenspg3;
volatile float Zellenspg4;

:

volatile int Zelle12=0;
volatile int Zelle34=0;
volatile int Zelle56=0;
volatile int Zelle78=0;
volatile int Zelle910=0;
volatile int Zelle1112=0;
volatile int Zelle1234=0;
volatile int Zelle5678=0;
volatile int Zelle9101112=0;
volatile int Zelle12345678=0;
volatile int Zelleresultat=0;

volatile float Wert12=0;
volatile float Wert34=0;
volatile float Wert56=0;
volatile float Wert78=0;
volatile float Wert910=0;
volatile float Wert1112=0;
volatile float Wert1234=0;
volatile float Wert5678=0;
volatile float Wert9101112=0;
volatile float Wert12345678=0;
volatile float Wertresultat=0;

void balancieren(int Registerwert)
{
I2C1->CR1 |= 0x100; //setze Startbit SB=1
while(!(I2C1->SR1 & 0x0001)); //Warte bis im Status Register das "Start-Bit" SB=1 gesetzt wurde
I2C1->DR = 0x40; //Schreibe 0x40 ins Datenregister und starte Übertragung
while(!(I2C1->SR1 & 0x0002)); //Warte bis im Status Register das "Adress sent-Bit" ADDR=1 gesetzt wurde (durch lesen wird
es gelöscht)
while(!(I2C1->SR2 & 0x0001)); //Lese Status Register 2 weil uP darauf wartet
I2C1->DR = 0x0B; //Schreibe 0x0B ins Datenregister und starte Übertragung
while(!(I2C1->SR1 & 0x0004));
I2C1->DR = Registerwert; //Low data byte senden
while(!(I2C1->SR1 & 0x0004));
I2C1->DR = 0x00; //High data byte senden
while(!(I2C1->SR1 & 0x0004));
I2C1->CR1 |= 0x1020; //enable PEC-Berechnung und sende PEC-Byte
I2C1->CR1 |= 0x0200; //setze Stopbit STOP=1
}

void SearchforMax1(void)
{
if (Zellenspg1>Zellenspg2)
{
Zelle12=1;
Wert12=Zellenspg1;
}
else if(Zellenspg1<Zellenspg2)
{
Zelle12=2;
Wert12=Zellenspg2;
}
}

void SearchforMax2(void)

```

```
{
if (Zellenspg3>Zellenspg4)
{
    Zelle34=3;
    Wert34=Zellenspg3;
}
else if (Zellenspg3<Zellenspg4)
{
    Zelle34=4;
    Wert34=Zellenspg4;
}
}

:
void SearchforMax1_1(void)
{
if (Wert12>Wert34)
{
    Zelle1234=Zelle12;
    Wert1234=Wert12;
}
else if (Wert12<Wert34)
{
    Zelle1234=Zelle34;
    Wert1234=Wert34;
}
}

void SearchforMax2_1(void)
{
if (Wert56>Wert78)
{
    Zelle5678=Zelle56;
    Wert5678=Wert56;
}
else if (Wert56<Wert78)
{
    Zelle5678=Zelle78;
    Wert5678=Wert78;
}
}

void SearchforMax3_1(void)
{
if (Wert910>Wert1112)
{
    Zelle9101112=Zelle910;
    Wert9101112=Wert910;
}
else if (Wert910<Wert1112)
{
    Zelle9101112=Zelle1112;
    Wert9101112=Wert1112;
}
}

void SearchforMax1_2(void)
{
if (Wert1234>Wert5678)
{
    Zelle12345678=Zelle1234;
    Wert12345678=Wert1234;
}
else if (Wert1234<Wert5678)
{
    Zelle12345678=Zelle5678;
    Wert12345678=Wert5678;
}
}

void SearchforMax2_2(void)
{
```

```
if (Wert12345678>Wert9101112)
{
    Zelleresultat=Zelle12345678;
    Wertresultat=Wert12345678;
}
else if (Wert12345678<Wert9101112)
{
    Zelleresultat=Zelle9101112;
    Wertresultat=Wert9101112;
}

void SearchforMaxresultat(void)
{
    SearchforMax1();
    SearchforMax2();
    SearchforMax3();
    SearchforMax4();
    SearchforMax5();
    SearchforMax6();
    SearchforMax1_1();
    SearchforMax2_1();
    SearchforMax3_1();
    SearchforMax1_2();
    SearchforMax2_2();
}
```

## 16 Zusammenfassung und Ergebnisse

Alle Schaltungen des Gerätes wurden erfolgreich entwickelt und parktisch umgesetzt. Die Schaltpläne und Layouts der Schaltungen wurden in EAGLE® entworfen und es konnten dadurch Verbesserungen und Korrekturen an den Schaltplänen und Layouts schnell und praktisch vorgenommen werden. Die prototype Konstruktion des Gerätes ist Abbildung xy zu entnehmen. Das Gerät besteht aus zwei separaten Platinen, die über einen 26 pin Stecker physikalisch verbunden sind. Die zwei Platinen werden als Steuerungsplatine und Messplatine bezeichnet. Auf der Steuerungsplatine befindet sich der Mikrocontroller STM32F103 mit TFT-Display und Tastatur, sowie einem Tiefpassfilter der die Gleichspannungs-Stellgröße für Strom und Spannung erzeugt. Auf der Messplatine befindet sich der Mess- und Balancierungsbaustein MAX11068, und der Strommessbaustein DS2764 mit den entsprechenden äußeren Beschaltungen durch passive elektronische Bauelemente. Die Tastatur und das TFT-Display bieten die Möglichkeit die Ladeparameter komfortabel einzustellen und zu beobachten sowie Messdaten zu verarbeiten.

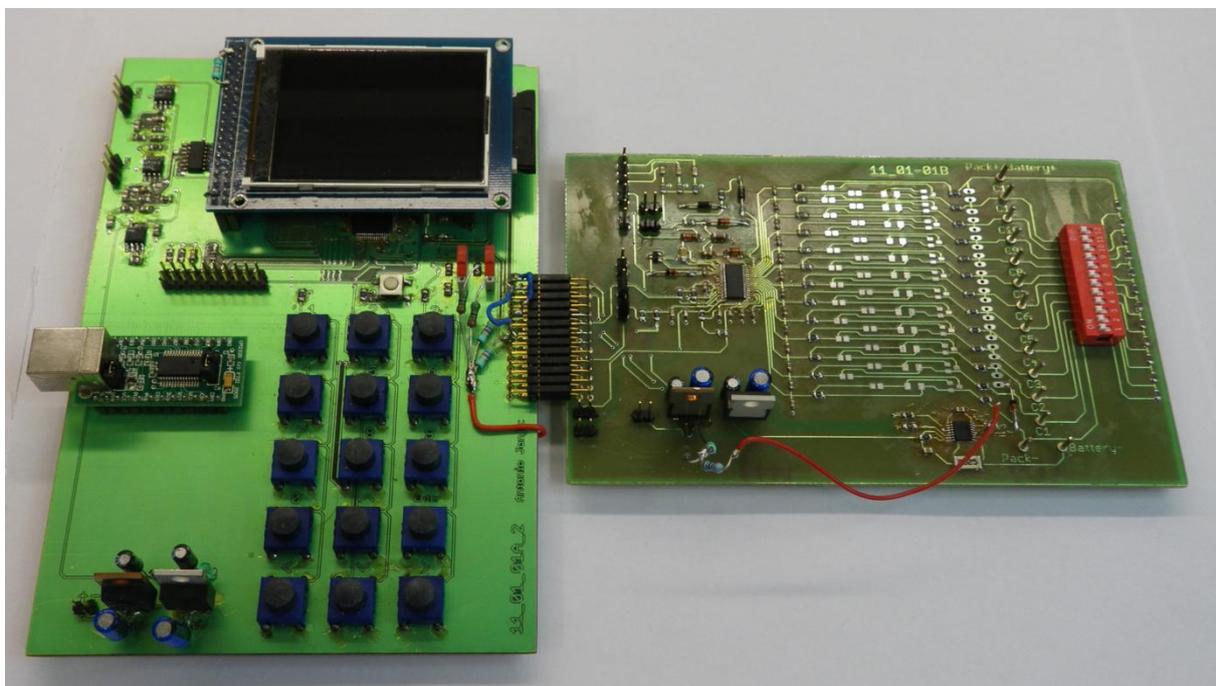


Abbildung 109: Konstruktiver Aufbau des Mikrocontroller gesteuerten Lihtium-Ionen Batterieladegerätes

Als steuerbares Netzgerät wird das Gerät PSI800T der Firma Elektro-Automatik GmbH verwendet. Dieses steuerbare Netzgerät bietet die Möglichkeit die Ausgangsspannung und den Ausgangsstrom über eine Gleichspannung, an den beiden Steuereingängen,

einzustellen. Bei beiden Steuergleichspannungen kann zwischen zwei Spannungsintervallen umgeschaltet werden, zwischen 0V - 3,3V und 0V - 5V.



Abbildung 110: Das steuerbare Netzgerät

Das Softwaremenü ist intuitiv und übersichtlich aufgebaut. Im Startmenü kann zwischen vier Programmpunkte ausgewählt werden.

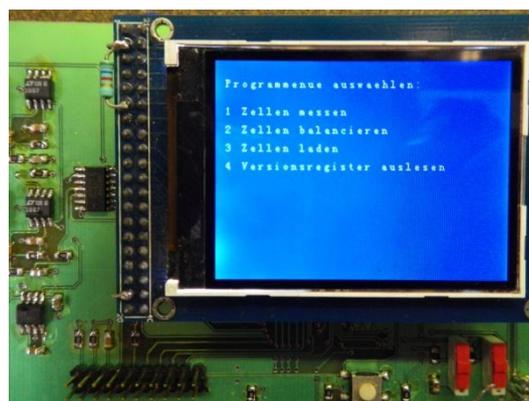


Abbildung 111: Hauptmenü des Programmes

Diese sind, „Zellen messen:“ hier werden alle Zellenspannung des angeschlossenen Akkupacks gemessen und dargestellt,

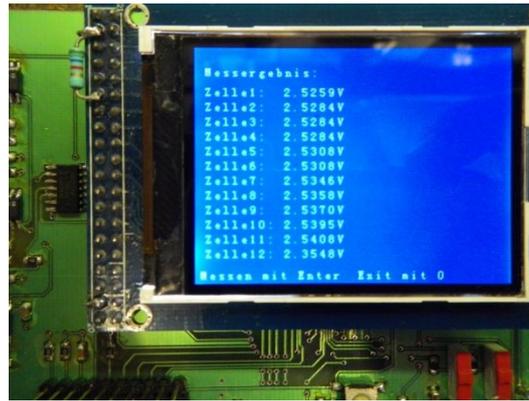


Abbildung 112: Messergebnisse Menü des Programmes

„Zellen balancieren“: hier wird die Zelle mit der höchsten Zellenspannung bestimmt und diese dann per Tastendruck entladen, „Zellen laden:“ hier gelangt der Benutzer in ein Untermenüpunkt. In diesem Untermenüpunkt kann er zwischen „Li-Ion“ und „NimH“-Batterien wählen. Anschließend gelangt er in den Ladeuntermenüpunkt, wo der Benutzer die Anzahl der zu ladenden Zellen, den Ladestrom und die Zellenkapazität



Abbildung 113: Ladeparameter-Menü des Programmes

auswählen kann. Beim anschließendem Bestätigen der Eingabe mit „ja“ erscheint schließlich das Lademenü.



Abbildung 114: Ladevorgang-Menü

Im Menüpunkt „Versionsregister auslesen“ wird der Inhalt des Versionsregisters des Bausteines MAX11068 ausgelesen und angezeigt. Während der Entwicklung der Firmware waren vor allem die Ansteuerung des graphischen Displays und die I<sup>2</sup>C-Bus Kommunikation herausfordernd. Bei der I<sup>2</sup>C-Bus Kommunikation zwischen den Bauteilen und dem Mikrocontroller müssen die Übertragungsvorschriften für die Eröffnung und den Abschluss einer I<sup>2</sup>C-Bus Kommunikation streng eingehalten werden, wie zum Beispiel das Übertragen des PEC-Bytes und des „NACK“-Bits. Während einer I<sup>2</sup>C-Bus-Kommunikation dürfen keine Timer-Interrupt-Serviceroutinen bedient werden, da sich sonst die Software aufhängt. Um dies zu verhindern habe ich vor jedem I<sup>2</sup>C-Bus Kommunikations-Funktion den Timer ausgeschaltet und am Ende der Funktion wieder eingeschaltet. Durch die Beachtung dieser Problematik und einen relativ sauberen Programmierstil entstand eine stabile Software.

## 17 Abbildungsverzeichnis

Abbildung 1: Die Volta Säule .....	16
Abbildung 2: Funktionsweise eines chemischen Energiespeichers .....	17
Abbildung 3: Darstellung eines galvanischen Zelle mit Zinn und Kupfer als Reaktionspartner .....	18
Abbildung 4: Darstellung einer galvanischen Zelle ausgeführt als Blei-Säure Zelle .....	20
Abbildung 5: Darstellung einer elektrolytischen Zelle ausgeführt als Blei-Säure Zelle .....	21
Abbildung 6: Querschnitt einer Zink-Kohle Batterie .....	22
Abbildung 7: Querschnitt einer Zink-Kohle Batterie .....	22
Abbildung 8: Querschnitt einer Alkali-Mangan Batterie .....	23
Abbildung 9: Querschnitt einer Kopfzelle .....	23
Abbildung 10: Querschnitt einer Blei-Säure Batterie .....	24
Abbildung 11: Querschnitt einer Nickel-Cadmium Batterie .....	26
Abbildung 12: Querschnitt einer Nickel-Metall-Hydrid Batterie .....	27
Abbildung 13: Querschnitt einer Lithium-Ionen Batterie .....	29
Abbildung 14: Darstellung von Lithium-Ionen Zellen im gängigem Gehäuseformat 18650 .....	30
Abbildung 15: Die Gegenüberstellung verschiedener chemischer Speichersysteme im Ragone Diagram .....	31
Abbildung 16: Die Entlade-Charakteristik für zwei unterschiedliche Stromstärken einer Blei-Zelle .....	33
Abbildung 17: Die Thermische Abhängigkeit der Kapazität einer Blei und NiCd-Zelle .....	34
Abbildung 18: Die Entlade-Charakteristik eines Blei-Akkus bei unterschiedlichen Belastungen .....	35
Abbildung 19: Die Abhängigkeit der Kapazität der Zelle von der Temperatur .....	36
Abbildung 20: Die Entlade-Charakteristik unter verschiedenen Belastungen .....	37
Abbildung 21: Die Temperaturabhängigkeit der Kapazität einer NiCd-Zelle .....	37
Abbildung 22: Die Entlade-Charakteristik unter verschiedenen Belastungen .....	39
Abbildung 23: Die Abhängigkeit der Kapazität von der Temperatur und Belastung .....	40
Abbildung 24: Die Selbstentladung der NiMH-Zelle .....	41
Abbildung 25: Die Entlade-Charakteristik einer Petrolkoks/LiNiO <sub>2</sub> -Zelle .....	42
Abbildung 26: Die Selbstentlade-Charakteristik einer C/LiNiO <sub>2</sub> -Zelle .....	42
Abbildung 27: Die Entlade-Charakteristik der LiMn <sub>2</sub> O <sub>4</sub> -Zelle .....	43
Abbildung 28: Die Entlade-Charakteristik der verschiedenen Speichersysteme .....	44
Abbildung 29: Die Gegenüberstellung der verschiedenen Speichersysteme .....	44
Abbildung 30: Charakteristiken der Konstanzspannungs-Methode .....	48
Abbildung 31: Charakteristiken der Konstanzspannungs-Methode .....	49
Abbildung 32: Charakteristiken der Konstanzspannungs-Methode .....	49
Abbildung 33: Charakteristiken der Konstanzspannungs-Methode mit Strombegrenzung .....	50
Abbildung 34: Charakteristiken der Konstanzspannungs-Methode .....	51
Abbildung 35: Charakteristiken der Konstanzspannungs-Methode mit Strombegrenzung .....	51
Abbildung 36: Impulsladung mit erhöhter Spannung .....	52
Abbildung 37: Ladekurve eines NiCd-Akkus bei einem Strom von 0,1C .....	53
Abbildung 38: Ladecharakteristik bei verschiedenen Ladeströmen .....	54
Abbildung 39: Vergleich der Ladekurven der NiCd und NiMH Batterien .....	55
Abbildung 40: Vergleich der internen Temperatur während des Ladens einer NiCd und NiMH-Zelle .....	56
Abbildung 41: Ladekurve von Strom und Spannung einer Lithium-Zelle .....	59
Abbildung 42: Verteiltes Kommunikationsmodell einer Battery-Pack Architektur .....	62
Abbildung 43: SMBus Kommunikationsmodell einer Battery-Pack Architektur .....	63
Abbildung 44: Balancierung mit ohmschen Widerständen .....	64
Abbildung 45: Balancierung mit Schalter .....	65
Abbildung 46: Balancierung mit geschalteter Kapazität .....	66
Abbildung 47: "State of charge"-Kennlinie bei Balancierung mit geschalteter Kapazität .....	67
Abbildung 48: Balancierung mit geschalteter Induktivität .....	67
Abbildung 49: Blockschaltbild des LTC6802 Bausteines .....	69
Abbildung 50: Blockschaltbild des LTC6803 Bausteines .....	69
Abbildung 51: Blockschaltbild des ISL9216 Bausteines .....	70

Abbildung 52: Blockschaltbild des ATA6870 Bausteines .....	70
Abbildung 53: Blockschaltbild des MAX11068 Bausteines .....	71
Abbildung 54: Die Abbildung des relevanten Bereiches der internen Diodenstruktur des MA11068 .....	72
Abbildung 55: Das Blockschaltbild der Zusammenschaltung von mehreren Atmega 6870 Bausteinen .....	73
Abbildung 56: Das Dioden-Ersatzschaltbild des Bausteines des MAX11068 .....	74
Abbildung 57: Die Gehäuse und Pin-Abbildung des Bausteines MAX11068 .....	76
Abbildung 58: Das Blockschaltbild des MAX11068-Bausteines .....	76
Abbildung 59: Die "Charge Pump" des MAX11068-Bausteines .....	77
Abbildung 60: Das "Cell Equalization"-Modul des MAX11068-Bausteines .....	78
Abbildung 61: Das "Switch Bank"-Modul des MAX11068-Bausteines .....	79
Abbildung 62: Das "Upper Port"-Modul das MAX11068-Bausteines .....	80
Abbildung 63: Das "Lower Port"-Modul das MAX11068-Bausteines .....	81
Abbildung 64: Die internen Balancierungs-Schalter des MAX11068-Bausteines .....	82
Abbildung 65: Die internen Balancierungs-Schalter das MAX11068-Bausteines .....	84
Abbildung 66: Die internen Balancierungs-Schalter mit externen Bipolaren-Transistoren .....	85
Abbildung 67: Die internen Balancierungs-Schalter mit externen MOSFET-Transistoren .....	86
Abbildung 68: Blockschaltbild des gesamten I <sup>2</sup> C-Portes des MAX11068 .....	88
Abbildung 69: Bit-Format des "Write Word"-Befehls des MAX11068 .....	90
Abbildung 70: Bit-Format des "Read All"-Befehls des MAX11068 .....	90
Abbildung 71: Bit-Format des "I2C Address Byte"-Befehls des MAX11068 .....	90
Abbildung 72: Bit-Format des "HALLOALL"-Befehls des MAX11068 .....	91
Abbildung 73: Bit-Format des "ROLLCALL"-Befehls des MAX11068 .....	92
Abbildung 74: Bit-Format des "SETLASTADDRESS"-Befehls des MAX11068 .....	92
Abbildung 75: Bit-Format des "WRITEALL"-Befehls des MAX11068 .....	93
Abbildung 76: Bit-Format des "READALL"-Befehls des MAX11068 .....	93
Abbildung 77: Bit-Format des "WRITEDEVICE"-Befehls des MAX11068 .....	94
Abbildung 78: Die Schaltungstechnischen Veränderungen vorgenommen am Evaluation-Board des MAX11068 .....	97
Abbildung 79: Die Gehäuse und Pin-Abbildung des Bausteines DS2764 .....	100
Abbildung 80: Das Register für das Resultat der Strommessung des Bausteines DS2764 .....	100
Abbildung 81: Das Register für das Resultat der Strommengen-Messung des Bausteines DS2764 .....	101
Abbildung 82: Das interne Blockschaltbild des Bausteines DS2764 .....	102
Abbildung 83: Die beispielhafte Beschaltung des Bausteines DS2764 .....	102
Abbildung 84: Die Beschaltung des Bausteines DS2764 in unserer Anwendung .....	103
Abbildung 85: Darstellung der beiden Massen der Schaltung .....	104
Abbildung 86: Die Gehäuse und Pin-Abbildung des Bausteines ICL7660 .....	105
Abbildung 87: Ersatzschaltbild des schaltendem Kondensators des Bausteines ICL7660 .....	106
Abbildung 88: Ersatzschaltbild mit Requiv-Widerstand des Bausteines ICL7660 .....	106
Abbildung 89: Ersatzschaltbild des idealen Spannungsinverters des Bausteines ICL7660 .....	107
Abbildung 90: Anwendungsbeispiel des Bausteines ICL7660 .....	107
Abbildung 91: Beschaltung und Anwendung des USART-Modules FT232R .....	108
Abbildung 92: Übertragungsprotokoll der USART-Schnittstelle .....	110
Abbildung 93: Anschlussplan der JTAG-Schnittstelle mit dem STM32F103 Mikrocontroller .....	111
Abbildung 94: Darstellung der JTAG-Chain .....	113
Abbildung 95: Das "State Diagram" des TAP-Controllers .....	114
Abbildung 96: Die Darstellung des Register des JTAG Access Ports .....	115
Abbildung 97: Die Gesamtschaltung mit den Stellgrößen für das steuerbare Netzgerät .....	116
Abbildung 98: Das Blockschaltbild der Steuerausgänge für das steuerbare Netzgerät .....	117
Abbildung 99: Blockschaltbild des Gesamtsystems .....	119
Abbildung 100: Beschaltung des Bausteine MAX11068 und des DS2764 .....	120
Abbildung 101: Layout der Messplatine mit den Bausteine MAX11068 und DS2764 .....	121
Abbildung 102: Schaltplan der Steuerungsplatine .....	122
Abbildung 103: Layout der Steuerungsplatine .....	123

Abbildung 104: Blockschaltbild des STM32F103 ARM-Cortex M3 Mikrocontroller .....	126
Abbildung 105: Programm Menü der Keil-Entwicklungsumgebung .....	127
Abbildung 106: Projektuntermenü der Keil-Entwicklungsumgebung .....	127
Abbildung 107: Projektoptionsmenü der Keil-Entwicklungsumgebung.....	127
Abbildung 108: Projektbrowser der Keil-Entwicklungsumgebung .....	128
Abbildung 109: Konstruktiver Aufbau des Mikrocontroller gesteuerten Lithium-Ionen Batterieladegerätes ....	148
Abbildung 110: Das steuerbare Netzgerät.....	149
Abbildung 111: Hauptmenü des Programmes.....	149
Abbildung 112: Messergebnisse Menü des Programmes .....	150
Abbildung 113: Ladeparameter-Menü des Programmes.....	150
Abbildung 114: Ladevorgang-Menü .....	150

## 18 Tabellenverzeichnis

Tabelle 1: Gegenüberstellung der verschiedenen Speichersysteme .....	45
Tabelle 2: Gegenüberstellung der Lebensdauer bei verschiedener Entladeschlussspannung .....	58
Tabelle 3: Gegenüberstellung der verschiedenen Balancierungs-Bausteine .....	71
Tabelle 4: Kurzbeschreibung der Befehle des MAX11068.....	89
Tabelle 5: Die Register des MAX11068 .....	94
Tabelle 6: Das "MAXCELL"-Register des MAX11068-Bausteines .....	96

## 19 Literaturverzeichnis

1. GSG. (Jänner 2008). Abgerufen am November 2011 von <http://www.gsg-physik.net>
2. Günther Schmitt. (2008). *Mikrocomputertechnik mit Controllern der Atmel AVR-RISC-Familie; 4.Auflage*. Oldenbourg Verlag.
3. Halaczek, Thaddäus Leonhard, Radecke, Hans Dieter. (1998). *Batterie und Ladekonzepte*. Franzis Verlag GmbH.
4. Heinz-Albert Kiehne und 7 Mitautoren. (2001). *Gerätebatterien; 3.Auflage*. Expert-Verlag.
5. Isidor, Buchman. (2001). *Batteries in a portable world; 2nd edition*. Cadex Electronics Inc.
6. Maxim Integrated Products. (2010). 12-Channel, High-Voltage Sensor, Smart Data-Acquisition Interface. Sunnyvale, California, USA.
7. Maxim Integrated Products. (2006). High-Precision Li+ Battery Monitor with 2-Wire Interface. Sunnyvale, Californien, USA.
8. Okawa Denshi. (Jänner 2008). *OKAWA Electric Design*. Abgerufen am Dezember 2011 von <http://www.okawa-denshi.jp>
9. RCDC. (April 2009). Abgerufen am April 2011 von <http://www.rc-drift-szene.de>