Feasibility study of Li-Fe battery packs for a 12v car system

MASTER-THESIS

MA 709

written by

Gernot Hehn

Institut für Elektronik
der Technischen Universität Graz
Leiter: Univ.-Prof. Dipl.-Ing. Dr.techn. Pribyl Wolfgang

in cooperation with

AMS Ag

Evaluator: Ass.Prof. Dipl.-Ing. Dr.techn Winkler Gunter
Supervisor at AMS: Dipl.-Ing. Manfreg Brandl

Graz, September 2012

Contents

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 September 20, 2012 .....

(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.

September 20, 2012 .....

(signature)

Abstract

Vehicle electronics have been hugely influenced by the enormous advantages that have been made in the fields of electronic and microelectronics. New appliances like climate control vehicle safety systems (ABS/ESP/...), modern route and navigation systems, on board entertainment and passenger comfort applications have increased the demand for power and energy in cars dramatically. Adding to these new environmentally driven trends like start/stop, regenerative breaking, electrical power steering and mild hybrid solutions are increasing the demand for energy even more. The load on a modern mid to high end limousine power system has reached levels that will wear out the battery in an unsatifactorily short time. Targeting these problems two solutions are presently discussed by the big German automotive companies. Either a higher voltage (preferrably 48V) board net is implemented, partially substituting the classic 12V net, or the battery chemistry is changed towards a more robust and cycle resistant LiFePO4 system.

This thesis will evaluate the viability of the second solution. The goal is to enable understanding of the differences in battery chemistry, why they help target today's problems, and to explain necessary changes in the cars system to make the transistion. It will also propose a battery management system that performs all the necessary operations to keep the Lithium battery within its operating conditions. The last chapter will outline the practical evaluation of this system in a real world application and show how well the new system performs compared to a classic lead-acid battery system.

Zusammenfassung

Die Fahrzeugelektrik wurde enorm beeinflusst durch die Fortschritte in den Feldern Elektronik und Mikroelektronik. Neue Zusatzausstattungen wie Klimaanlage, Sicherheitssysteme (ABS/ESP/...), moderne Navigationsgeräte, integrierte Entertainment Systeme und Applikationen für den Fahrerkomfort haben die Energie und Leistungsanforderungen in Autos drastisch erhöht. Zusätzlich zu diesen Systemen haben auch durch Umweltbewusstsein getriebene Trends wie start/stop Automatik, regeneratives Bremsen, elektrische Servolenkung und mild Hybrid Antriebe den Bedarf an elektrischer Leistung im Auto weiter in die Höhe geschraubt. Die elektrische Last in einem modernen Mittel- und Oberklassewagen hat einen Bereich erreicht in dem eine klassische Batterie in unzulänglicher Zeit ihre Funktion aufgibt. Um diese Probleme in den Griff zu bekommen werden derzeit zwei Lösungen von den deutschen Automobilkonzernen diskutiert. Entweder ein Bord Netz mit höherer Spannung (vorzugsweise 48V) wird implementiert das die Aufgaben des 12V Netzes zum Teil übernimmt, oder die Batteriechemie wird geändert, zu einer LiFePO4 Chemie mit wesentlich besserer Zyklenfestigkeit und Hochstromeigenschaften. Diese Arbeit wird die Machbarkeit der zweiten Lösung beleuchten. Ziel ist es das Verständnis für verschiedene Batteriechemien aufzubauen und zu zeigen warum mit diesen die heutigen Probleme gelöst werden können. Auch die nötigen Änderungen, die für diese Umstellung am Auto selbst vorgenommen werden müssten, werden besprochen. Diese Lösung wird anhand eines Batterie Management Systems, welches alle notwendigen Operationen beherrscht um eine Lithium Batterie innerhalb ihrer Operationsbedingungen zu betreiben, beleuchtet. Das letzte Kapitel wird sich mit der Evaluierung diese Systems in einem handelsüblichen Fahrzeug auseinandersetzen und dabei einen Vergleich zu herkömmlichen Lösungen bieten.

1 Car Batteries

1.1 Typical Car 12V Electric System

The typical electric system in today's car would look somewhat like Figure 1.1 The alternator, which is directly coupled to the engine block, will supply a uniform 12V net. The battery will be charged until it reaches a defined maximum voltage. A control unit within the alternator will regulate its exciting current so that the output voltage won't surpass a level of $13.8 \approx 14.4V$. With the application of a Battery management system and the knowledge about the State of Charge (SOC) of the battery it has become possible to only charge the battery when necessary or in desired conditions (regenerative breaking for example). This also allows to stop charging the battery when the motor power is needed elsewhere (for example in acceleration phases).

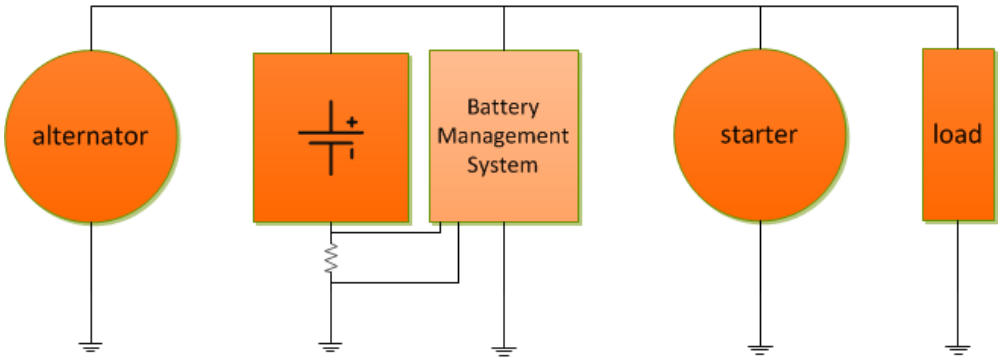

Figure 1.1: Typical car battery system

1.2 Standard Car Batteries (wet cell lead-acid)

The car battery most commonly used today is a "flooded cell" wet type lead-acid battery. The battery consists of 6 serially connected primary lead acid-cells with a typical voltage of 2V each. A single cell typically consists of a negative electrode made of spongy or porous lead, the positive electrode made of lead oxide and an electrolyte made out of 35% Sulfuric Acid and 65% water. Additionally there is a permeable separator separating the electrodes to stop them from accidentially connecting. As the name of the battery suggests both electrodes are submerged in the electrolyte. The chemical reaction that allows these batteries to work is as follows:

Overall reaction:

$$\mathrm{PbO_2 + Pb + 2\,H_2SO_4} \underset{charge}{\overset{discharge}{\rightleftharpoons}} \mathrm{2\,PbSO_4 + 2\,H_2O}$$

At the negative terminal:

$$\mathrm{Pb + SO_4^{2-}} \underset{charge}{\overset{discharge}{\rightleftharpoons}} \mathrm{PbSO_4 + 2\,e^-}$$

At the positive terminal:

$$PbO_2 + SO_4^{2-} + 4\,H^+ + 2\,e^- \underset{charge}{\overset{discharge}{\rightleftharpoons}} PbSO_4 + 2\,H_2O$$

1.3 AGM Batteries

AGM (Absorbed Glass Mat) batteries belong to the group of VRLA (Valve Regulated Lead Acid), "sealed" or "maintainance free" batteries. AGM batteries use a very fine fiber Boron-Silicate glass mat seperator to immobilize the sulfuric acid in the battery. This type of battery was invented for military applications in the 80's. The main advantages compared to a classic car battery are

- **Lower self discharge rate**
- **Higher number of charge cycles**
- **Less capacity degradation**
- Safer than wet batteries
- Virtually maintainance free
- Can withstand hazardous environments better

Their main disadvantage is that they are more expansive ($\approx 20-30\%$) compared to standard batteries.

However car manufacturers tend to use these kind of batteries in modern cars featuring Start/Stop to get acceptable lifetimes.

1.4 Handling

Like all batteries lead-acid batteries have special requirements concerning their operation. These can be split into charging and discharging conditions.

1.4.1 Charging

As Figure 1.2 shows there are 4 distinct phases in the charging process

1. Bulk Charge or CC (Constant Current) mode:

   In this mode the battery is somewhere between 30% and 80% SOC (discharge below 30% SOC will damage the battery permanently) The charger applies a constant current somewhere between $\frac{1}{10}$ and $\frac{1}{3}$ of the battery's capacity. The voltage climbs to a level called the "quick charging voltage". This phase is where most charge is absorbed by the battery in the shortest amount of time.

2. Absorbtion Charge or CV (Constant Voltage) mode:

   In this mode the battery receives another 15% of charge. This phase is qualified by the constant voltage applied to the battery. The current drawn by the battery slowly decreases due to the chemical process in the battery coming close to equilibrium.

3. Equalization Charge Mode:

   In this mode a modest constant current in the range of 5% to 10% is applied to push the battery to the gasing phase on purpose. This mode helps to deliver the remaining 5% of capacity into the battery and to equalize the charge between the 6 cells in a typical system. This phase is typically not used in automotive application as gasing is

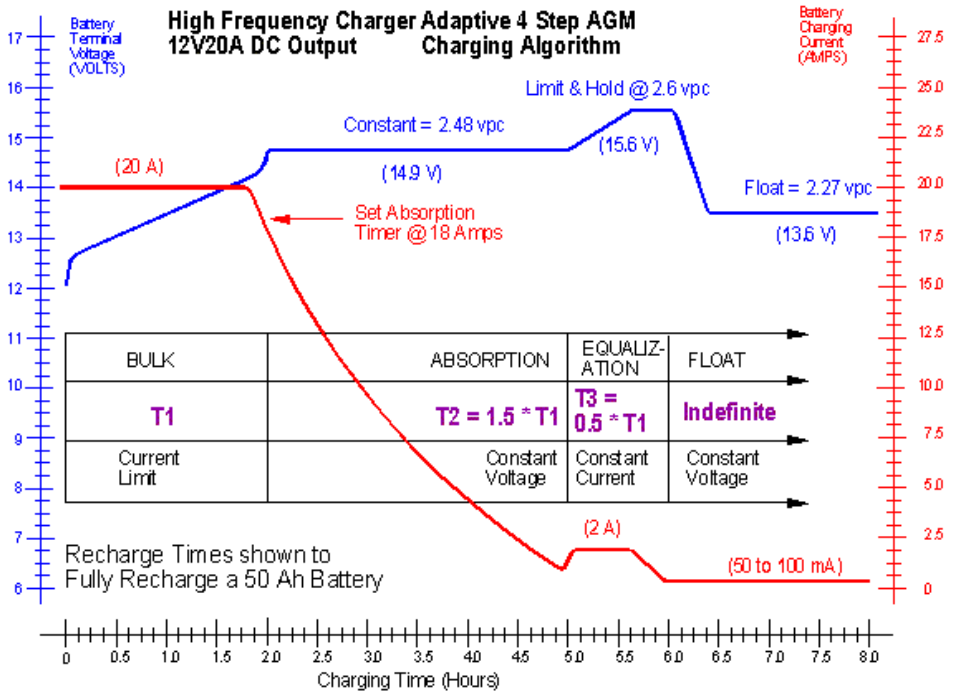

Figure 1.2: Diagram of the 4 lead-acid battery charging phases [24]

highly undesired in a car (the gas consists of elementar Hydrogen and Oxygen and is highly explosive).

4. Float Maintenance Charge Mode:

   This mode is only used in charging systems for battery storage. Typically a very small current in the 50 to 100mA range is applied to compensate the battery's self discharge process. This mode is not implemented in a car system as there is no power source other than the battery to charge from during the time the car is parked.

1.4.2 Discharging

Like every chemical process the discharging process of a lead acid battery is dependent on the temperature at which the reaction takes place, the concentration of the active partners in the electrolyte and the rate of Ion transfer (ie. the amount of current drawn from the battery)

This leads to the discharge performance of a Lead-acid battery being significantly dependent on

- Discharge current relative to the total capacity of the Battery (C-rating)
- The percentage of capacity used during a cycle (aka. cycle depth)
- The temperature of the battery during discharge

This means that the amount of charge the battery is able to deliver as well as the number of cycles the battery is able to endure is dependent on those parameters.

A dependence which is illustrated in Figure 1.3 through 1.5

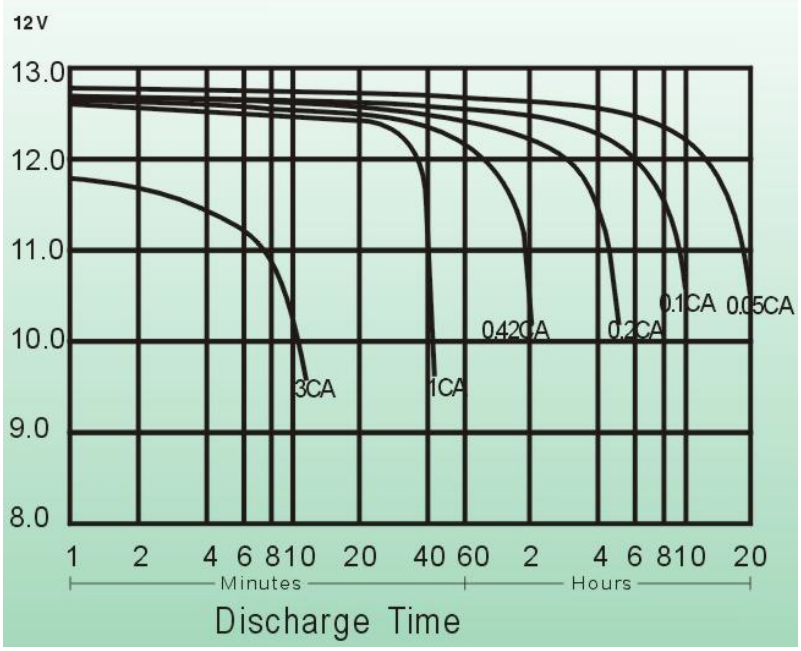

Figure 1.3: Influence of the discharge current on the usable capacity [5]

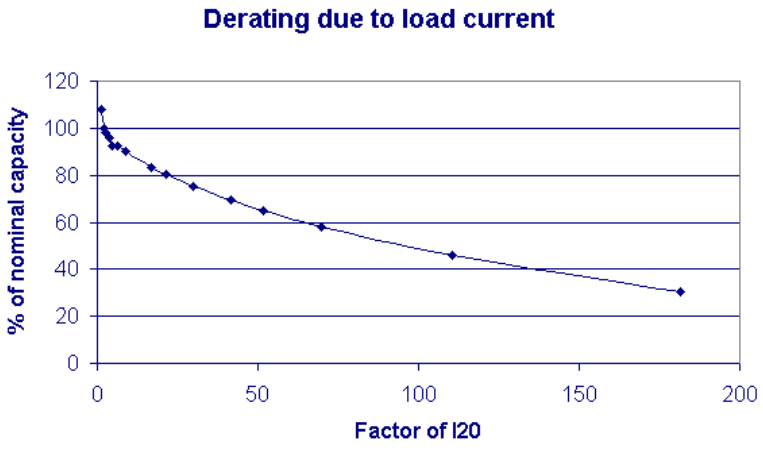

Figure 1.4: Influence of the discharge current on the usable capacity [4]

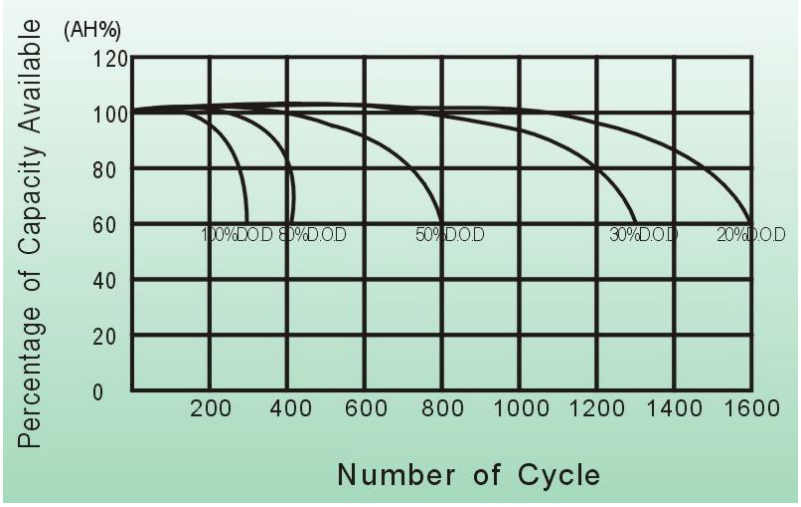

Figure 1.5: Influence of the discharge current on the cycle life [5]

1.5 Problems with Today's System

A classic car system typically used its battery only once per drive to start the motor and supply power to lights etc. when the car was parked. In a modern car system the load on the battery is mounting drastically due to the increased number of electrical appliances. In addition to that a start stop automatic makes it necessary to cycle the battery more than once during a drive cycle and use a higher percentage of its capacity. This leads to the battery beeing constantly in a charge state somewhere around 50% SOC rather than being fully charged most of the time as this was the classic case.

This leads to early battery failures in the field and the necessity to change batteries more routinely than it was the case before.

2 Li-Batteries

"One of the important things to understand about lithium batteries is that the term 'Lithium Battery' summarises a whole lot of different battery chemistries. In fact the technology in this field is still changing, so I can't even give a full overview of the existing technologies." [23]

In this text I will focus on the LiFePo4 type of batteries and the differences in handling compared to lead acid batteries to point out the problems and advantages arising from this change in battery chemistry.

2.1 LiFePo4 Batteries

To start off let us have a look at a LiFePo4 Cell and what it is made of.

This type of cell consists of a cathode made out of LiFePo4 material, an anode of graphite or hard carbon and a lithium ion conducting electrolyte, usually an organic solvent mixed with a lihium salt. The chemical reaction taking place in such a battery is as follows:

$$\mathrm{LiC_{6(s)}} + \mathrm{Fe^{3+}PO_4} \underset{charge}{\overset{discharge}{\rightleftharpoons}} 6\,\mathrm{C_s} + \mathrm{Li^{+}Fe^{2+}PO_{4(s)}}$$

2.2 Comparison with Lead Acid Batteries

The table 2.1 gives a good overview of the difference in properties between lead acid batteries and LiFePo4 batteries. I have added the $LiCoO^2$ type of lithum batteries for comparison. This is the type used in most modern computers and cell phones. Although these cells have an even higher energy density, they are not considered for automotive application since they are not intrinsically safe and suffer from a process called thermal runaway. This process can lead to an explosive decomposition of the battery when overcharged, which has led to a lot of recalls of batteries where the protection mechanism wasn't adequate [19] & [18]. Needless to say that such an instable nature disqualifies these batteries for usage in automotive applications.

Summarized the main advantages of LiFe batteries are the following:

- $\approx$ 4 times higher cycle life
- $\approx$ 3 times the energy density which is shown in Figure 2.1
- Very flat voltage curve 2.2
- High charge rates
- Low self discharge
- Unlike lead acid batteries, can be left in a partially discharged state for extended periods without causing permanent damage
- Maintenance free for the life of the battery
- Can be operated in any orientation
- No heavy metals used

In contrast these new batteries also have the following disadvantages:

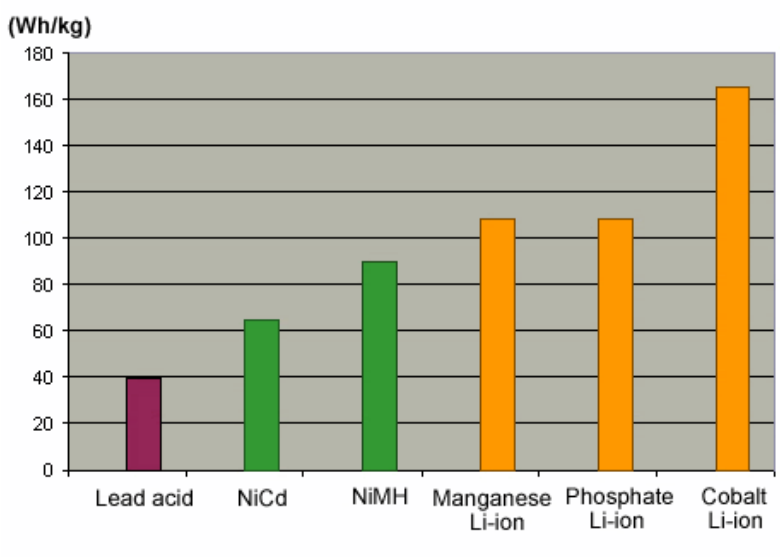

Figure 2.1: Comparison of energy density of various battery types [20]

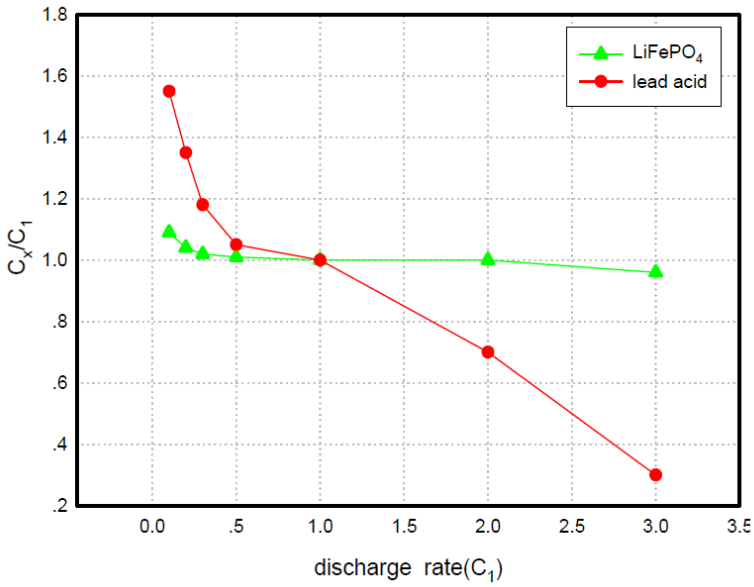

Figure 2.2: Comparison of discharge curves of various battery types [7]

- $\approx$ 2 - 3 times the costs
- Problems with low temperatures
- More critical in over and undercharge conditions

Battery Type	*Lead Acid*	*LiFe*	$LiCoO^2$
Commercialisation	1881	2002	1990
Cell Voltage [V]	2	3,2	3,7
Energy by Weight [Wh/Kg]:	30-40	90-110	130-200
Specific Power [W/Kg]	180	300	250 - 340
Energy by Volume [Wh/L]	60-75	220	340-400
Max charge rate	1C	3C	5C
Continous discharge rate	3C	3C-10C	10C-30C
Peak discharge rate	10C	10C-30C	30C-70C
Recharge Time	$\geq$ 10 Hours	1 Hour	15min - 1hour
Cycle Life [Cycles]	500 - 800	2000 - 3000	500-1000
Self Discharge per month	5%-25%	$\approx$ 3%	$\approx$ 5
Temp Range [°C]	-15 to 40	0 to 45	0 to 45
Preferred Charge Method	CC/CV	CC/CV	CC/CV
Average Energy Cost [$/kWh]	$100	$300	$200
Size [%]	100	75	30
Weight [%]	100	35	25

Table 2.1: Comparison of Lead Acid and Lithium cells

2.3 Handling

As discussed in the disadvantage section above, Lithium based batteries are very susceptible to over- and underload conditions. This requires the usage of specialised electronics which keeps the batteries within their specified parameters under all conditions. We can summarize the tasks such an electronic has to fulfill as:

- Measure the pack voltage
- Measure the individual cell voltages
- Do a plausability check of pack voltage versus sum of cell voltages
- Measure the current being charged into or discharged from the battery
- Measure the temperature of the battery stack
- Try to equalize the charge between the different cells of a stack
- Compare the measured values against the maximum ratings described by the battery manufacturer
- If any of the values is out of spec, disconnect the battery immediately
- Try to estimate the amount of charge still available in the battery
- Report the battery status and the system status to the ECU (typically via a LIN interface)

2.3.1 Charging

The charging system used for Li-type batteries is the CC (Constant Current)/CV (Constant Voltage) system. This is the same as the first two phases in a lead acid charging process.

The third phase may never be used due to the reaction of lithium batteries to overcharge conditions and the fourth phase is mostly unnecessary due to the low self discharge rate that all lithium batteries have in common. A typical charging curve (taken form the datasheet [6]) is shown in Figure 2.3

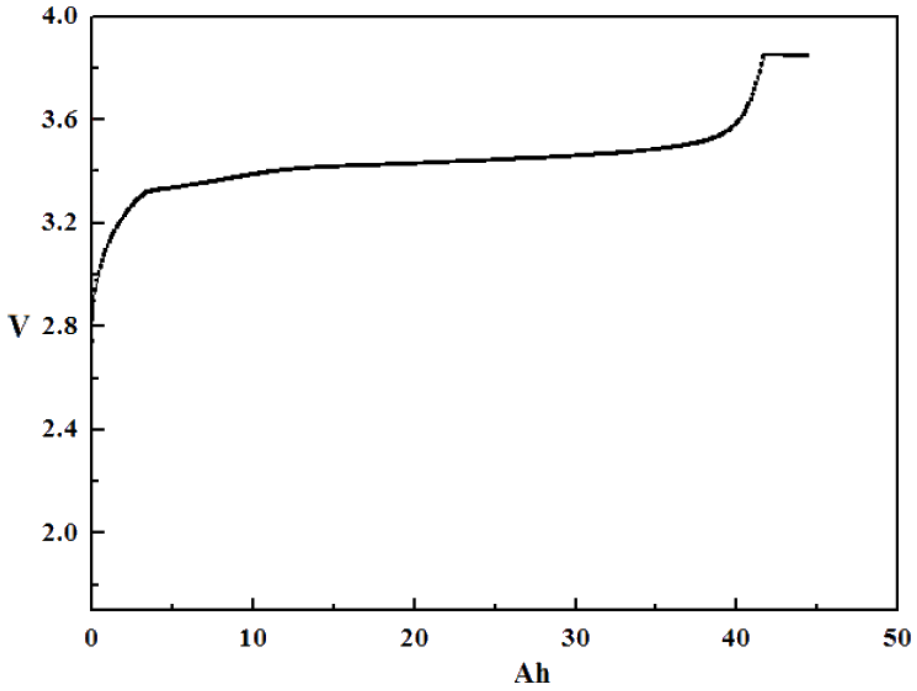

Figure 2.3: Plot of charging operation

2.3.2 Discharging

As with lead acid batteries the discharging characteristics hugely depend on current consumption (Figure 2.4) and temperature (Figure 2.5) as well as aging effects and the SOC.

These effects have to be addressed by the battery management in order to allow safe operation of the battery in its final application.

2.3.3 Balancing

Balancing is the process we use to equalize the charge of individual cells within a battery pack. There are a couple of different topologies in use to achieve these goals. [23] gives a good overview of the various possibilities. In this work I will however concentrate on active balancing using an inductive coupler to transport charge between various cells. As discussed in section 5.2 we are using an AS8505 chipset to enable this functionality. A block diagram of this chipset can be seen in Figure 2.6.

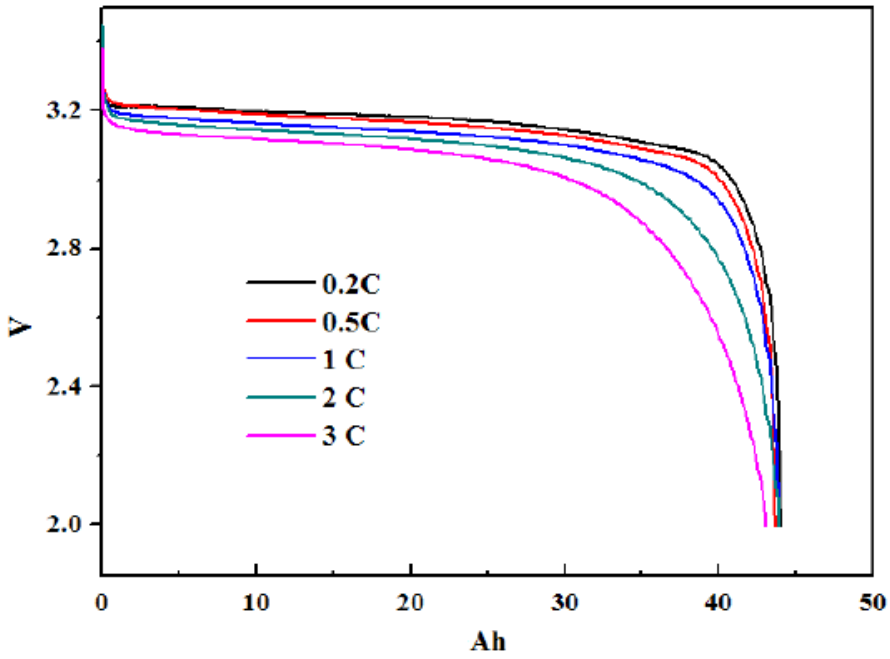

Figure 2.4: Plot of discharging operation at various discharge rates

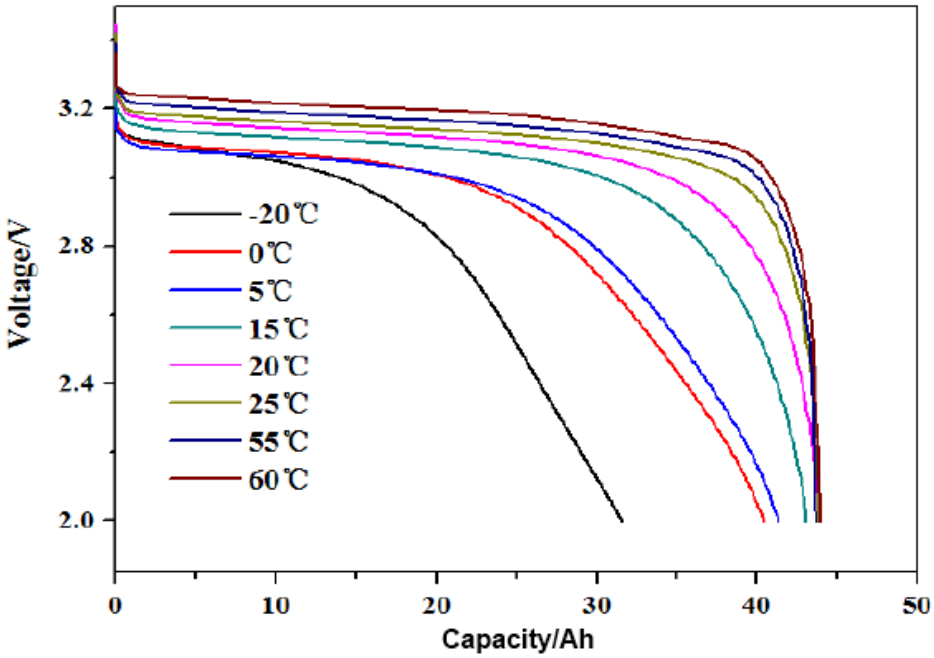

Figure 2.5: Plot of discharging operation at various temperatures

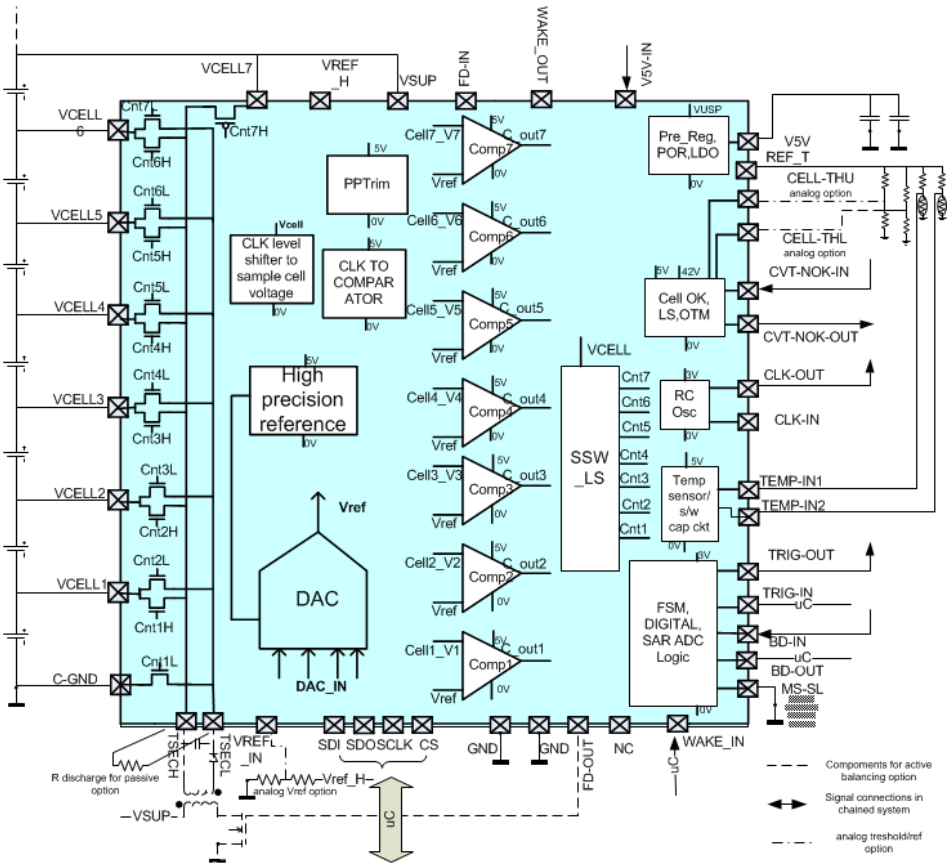

Figure 2.6: Block diagram of the AS8505/06 chip including peripheral components

3 LIN

3.1 Basics

'LIN (Local Interconnect Network) is a concept for low cost automotive networks, which complements the existing portfolio of automotive multiplex networks.'[21]

LIN is a single wire serial communication bus that was mainly developed for automotive applications which don't require the complexity and speed of CAN and want to profit from a cost effective simple solution. The main benefits of the LIN architecture are:

- one master multiple slave network
- low cost implementation using standard uart (Universal Asynchronous Receiver Transmitter) hardware
- possibility to synchronize slaves that don't have a stable clock source
- low cost single-wire implementation
- speed up to 20 kbit/s.

In the LIN Interface the master has control over all bus functions. It initializes transfers to and from the slaves and manages the bus activity schedule. The slave waits for its PID (Protected IDentifier) and responds with a pre defined answer.

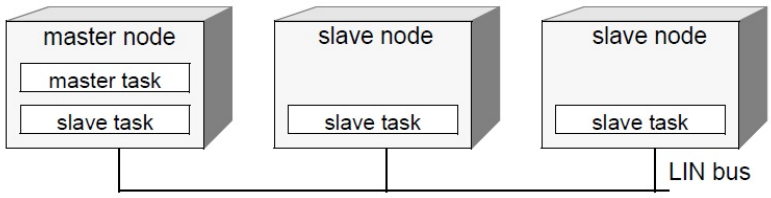

Figure 3.1: Typical LIN network

3.2 Protocol

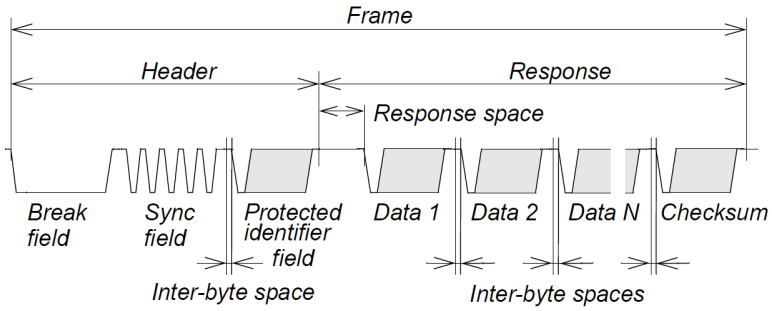

Figure 3.2: LIN Frame

The LIN Protocol describes a LIN frame consisting of

- Break field
- Sync field
- PID
- 1-8 bytes of data
- Checksum

The first three items are considered to be the header and are always sent by the master. The data bytes and checksum are the slaves response to the master request. LIN protocol defines an LSB first bit order. We will now take a look at the function of each item

3.2.1 Break Field

The break field stands at the beginning of each LIN communication. It is essential to signal the beginning of a new transmission and to wake up all devices connected to a LIN Bus (if they are in sleep mode). The break field shall be at least 13 nominal bit times of dominant value (low voltage) followed by a break delimiter as shown in Figure 3.3 The slave node shall use a threshold of 11 slave bit times to detect the break field in order to allow for clock variations.

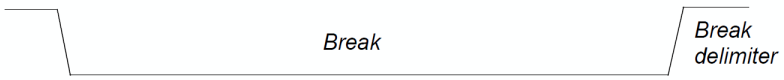

Figure 3.3: LIN Break field

3.2.2 Sync Field

The Sync field always transmits the data value 0x55 and is used to calibrate the bit clock of the slave with the master. This is essential if the slave uses a cheap low accuracy rc oscillator. Modern microcontrollers have integrated hardware that auto detects the sync field and makes the baud (data transfer rate) calibration easier.

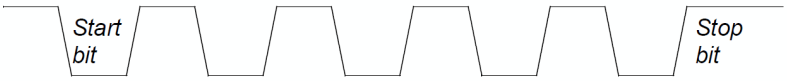

Figure 3.4: LIN Sync field

3.2.3 PID Protected IDentifier Field

The protected identifier consists of the frame identifier (Bits 0 to 5) and two parity bits (6 and 7). ID Values 0 to 59 are used for signal carrying frames, 60 and 61 for diagnostic frames and configuration data 62 and 63 are reserved.

The Parity calculation works as follows:

$$P0 = ID0 \oplus ID1 \oplus ID2 \oplus ID4$$
$$P1 = \neg(ID1 \oplus ID3 \oplus ID4 \oplus ID5)$$

The mapping of the PID frame is shown in Figure 3.5

Figure 3.5: LIN PID field

3.2.4 Data

The Data frame carries between 1 and 8 bytes of data. Data is always transported LSB first. This also applies to data fields that span over multiple bytes. The LSB is always transmitted in the byte that is sent first.

3.2.5 Checksum

The last Field of a LIN transmission is always a Checksum. According to the new LIN 2.2 specification, the Checksum is calculated over the data bytes as well as the PID. The checksum is an inverted eight bit sum with carry. For example if we transmit the following information: PID=20 Data1 = 80 Data2 = 200 Data3 = 14 we would get

$$20 + 80 + 200 + 14 = 314 - 255 = 59$$

$$\neg 59 = 196$$

So the checksum would be 196. The receiver checks this by adding the PID and Data fields and the Checksum. The result should always be 0xFF.

3.2.6 Frame Types

The LIN protocol distinguishes between three types of frames

- Unconditional Frame
- Event triggered Frame
- Sporadic Frame

Unconditional Frames are in the PID range between 0 and 59. The header is transmitted by the master in the allocated time slot. The response of the slave shall be immediate. All subscribers to this frame will receive it and make it available to their application.

Event triggered Frames are used to request data from slaves only in case the data has changed. Subscribers to this frame only respond if the requested value has changed since the last request.

In addition to these frames, Diagnostic Frames (with PID 60 and 61) carry a special transport layer and are used for in-system diagnosis and calibration.

3.2.7 LIN Schedule

The master has knowledge about all PIDs used in this particular network. It uses a predefined schedule table to request data from the slaves and makes these data available to the application. The default flowchart for such a schedule looks like this:

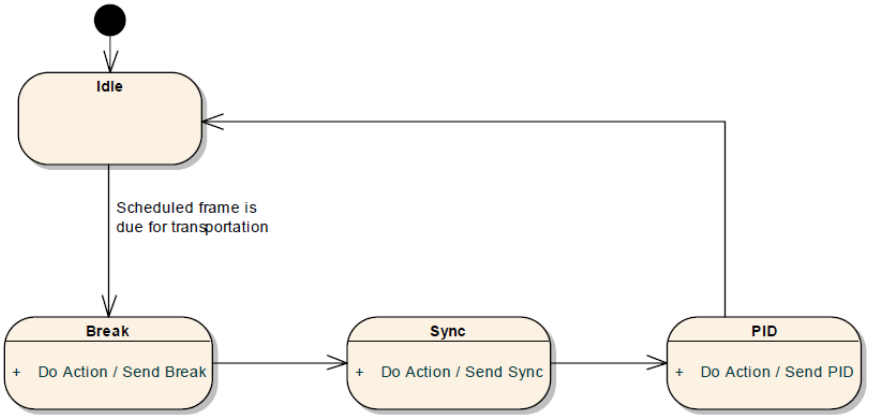

Figure 3.6: LIN Master Task [21]p.41

For our purpose we did not implement a schedule table into our LIN master as can be seen in the software flowchart 4.1 and rather let the application running on the PC decide when and which PID is going to be sent. This makes the overall software design easier and allows for more flexibility during testing, furthermore we do not need scheduling since there is only one LIN slave in this particular network.

The LIN slave on the other hand has to follow a procedure shown in Figure 3.7

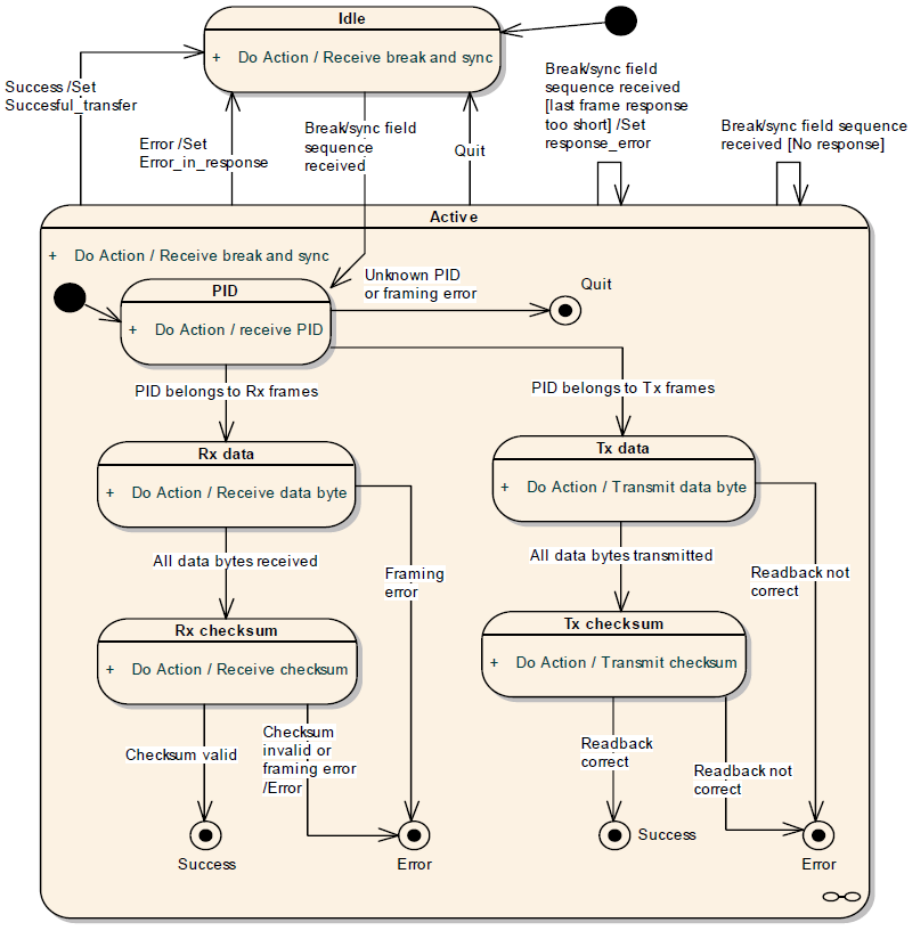

Figure 3.7: LIN Slave Task [21]p.43

The implementation of this behavior is discussed in detail in the Firmware section 5.4.

4 USB2LIN

In order to test the functionality of the whole platform in general and the LIN system in detail and to create a user friendly way to access the module data on a computer a small USB to LIN interface was developed.

4.1 Operation

This system uses the power of a modern PIC 18f14K50 microcontroller to implement a fully USB compatible PC Interface that is represented as a virtual COM PORT on PC side and the AS8530 LIN SBC to provide a standard compatible LIN interface.

In Figure 9.1 you can see that this chip is powered by an AS1340 DC/DC converter which provides the 12V supply necessary for correct LIN operation. The switch S2 allows for system reset as well as for the possibility to update the firmware via an integrates USB HID bootloader. For system diagnosis two LEDs are placed which display the status of the board (connected/driver installed/operational).

4.2 Choice of Components

Microcontroller: The PIC 18F14K50 was chosen for its integrated hardware USB slave functionality and LIN Interface. Good application support for both these Interfaces is also provided by Microchip.

Lin System Basis Chip: The AS8530 is a house internal LIN SBC. It provides the necessary level translations for the 12V LIN interface as well as load dump and ESD protection. It also allows for remote wakeup via LIN.

Dc/Dc: The AS1340 high efficiency DC/DC converter uses an internal 1.4A MOSFET to reduce the external part count and makes the small board layout possible.

4.3 Schematic & Layout

Schematic 9.1 and Layout9.2 can be studied in the Appendix section of this document. P1 is a connector to the TTL compatible UART/LIN Interface of the microcontroller, which allows easy observation of the LIN Bus through a logic analyzer. P2 is the connector to the LIN Interface. It also outputs the 12V power rail which can supply up to 200mA of power to a slave application. Remember however that there is no short circuit protection, so the USB port of your computer might get damaged in an overload condition. J3 is a PicKit compatible header to allow direct programming and debugging of the PIC controller. If the Pic programmer is used to upload firmware directly, you will however lose the Bootloader functionality. It is worth noting that the LIN interface is ESD protected by the internal ESD protection of the AS8530 chip, however the USB interface does not sport any ESD protection other than that of the microcontroller itself.

Additionally I want to give a short connector pin listing starting from left to right with the USB connector facing to the left side and the populated side of the board facing upwards.

PGC|PGD|GND|VCC|MCLR RX|GND|TX GND|LIN|12V

4.4 Firmware

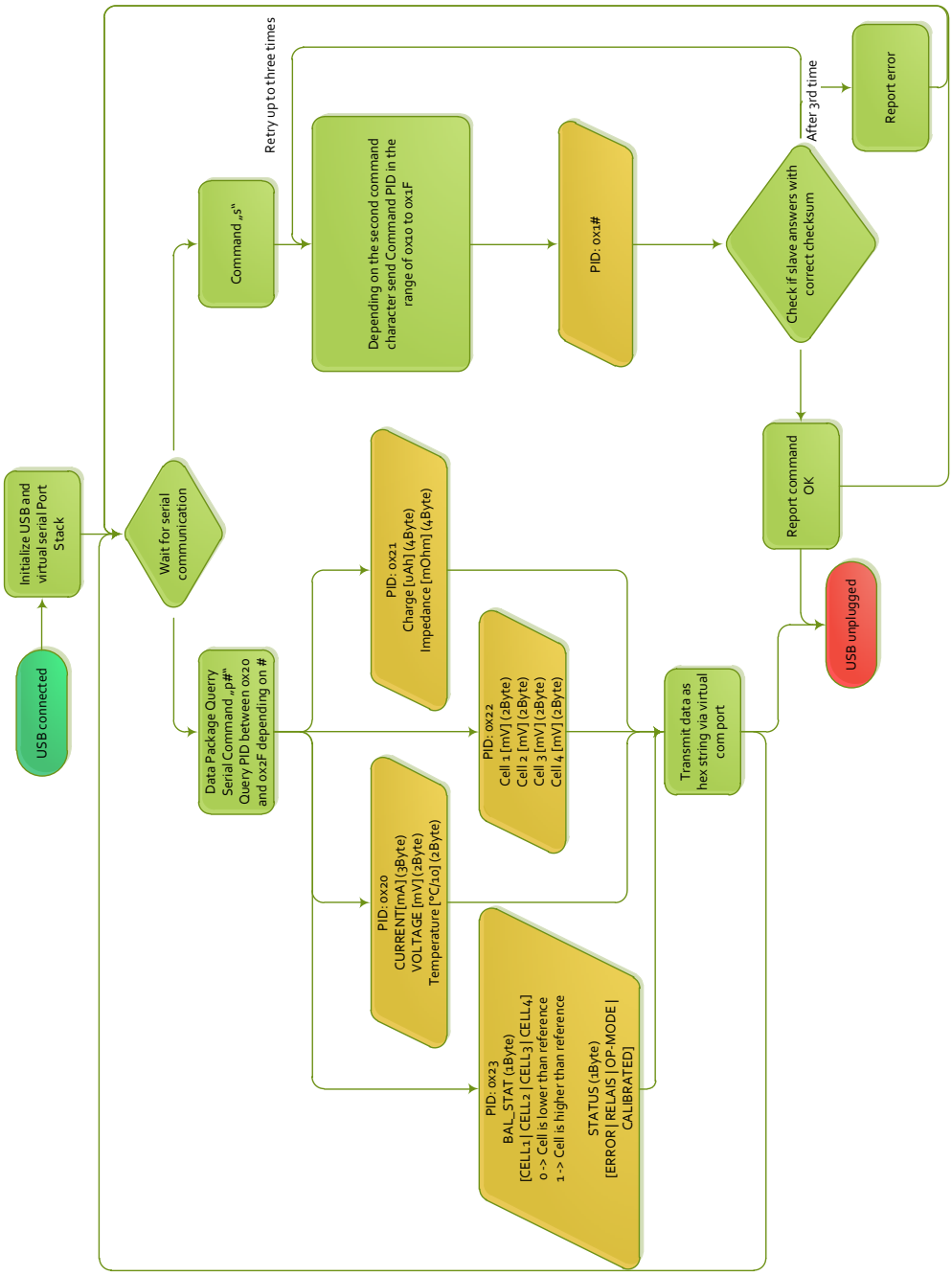

Figure 4.1: USB2LIN Software Flowchart

As one can see in Figure 4.1 the software flow of the USB2LIN Interface is based around a virtual serial port presented to the PC side. This port allows connection with any software capable of serial communication with communication settings *115200kBaud 8N1*. After connection to a USB port the USB2LIN Firmware awaits the entry of a human readable command. The Firmware will accept either a "p" [ASCII: 0x70] data package query followed by a hex number between 0 and F and upon reception it will start to request the information associated with this PID or and "s" [ASCII: 0x73] followed by a hex number between 0 and F which will prompt it to send the according Command PID in the range of 0x10 to 0x1F. Upon reception of the slave answer it will check the data for integrity with the help of the LIN checksum and report it back to the PC via the virtual serial interface as human readable ASCII data. In case of an error the Interface will try to resend the command up to three times. If this doesn't work, it will report an error to the host with the string "er".

4.5 Bootloader

The PIC18 Microcontroller is programmed with the microchip HID-Bootloader available as part of the Microchip Application Libraries[22] After installation of this package the necessary software can be found under *"Microchip Solutions v2012-12-05\USB\Device - Bootloaders\HID \HIDBootloader (Windows).exe"* This program will allow you to upload new firmware to the microcontroller in use via the USB connection and an easy to use graphical interface.

To upload new firmware perform the following steps:

1. Start the HIDBootloader according to your operating system
2. Press the button on the USB2LIN board
3. While pressing the button connect the board to the USB interface of the Computer
4. The Bootloader software will recognize the attached board and display "Device attached"
5. Click the "Open Hex File" button and select the new firmware file
6. Click the "Program/Verify" button. This will program the new firmware
7. After successful flash operation (as in Figure 4.2) disconnect the board from USB and then reconnect it
8. The new Firmware should now be in operation

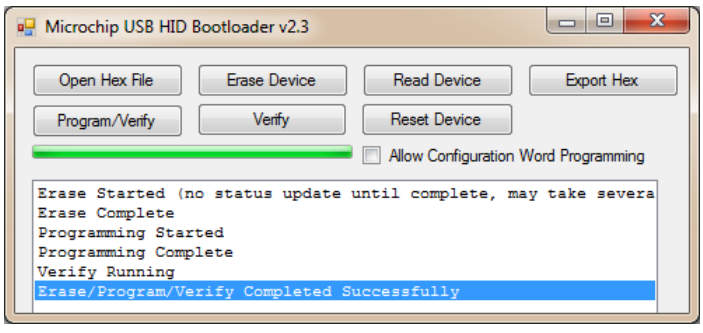

Figure 4.2: HID Bootlader after successful flashing operation

5 LiFe Battery Management

As was discussed in the section about Battery technologies the usage of lithium based batteries necessitates the installation of a battery management system. In the following chapter I will introduce a concept for a system that provides all the features required to safely operate a LiFePo4 Battery in a car environment.

The main design concerns for such a system were:

- *Accuracy*

  The system has to achieve an accuracy of <10mV and <100mA over the whole automotive temperature (-40 - +125°C) range and the complete voltage and current range of the batteries

- *System Safety*

  The system has to be able to detect malfunctions and react to them accordingly.

- *Processing Power*

  For the implementation of a complex battery model, which is essential for good SOC/SOH estimation, a fast processor is required to compute the mathematical model within an acceptable time frame

- *Power consumption*

  The targeted power consumption in standby mode was $100\mu A$ for the whole system.

5.1 Block Diagram

The Block diagram (Figure 5.1) gives a good overview of the interaction of the different functional blocks in use. The batteries are connected to the AS8515 ADC chip as well as the AS8506 balancing chip. These two ICs supply most of the functionality required in this system. A battery disconnect switch and a 100 $\mu\Omega$ Shunt are connected in series with the power path of the battery, which is available at two pole plugs on the top of the case like in a classic lead acid battery. USB and JTAG connectivity have been implemented to make the debugging process easier. However the production version would obviously only have the LIN connection to communicate with the ECU.

5.2 Choice of Components

Microcontroller:
Since the trend in Battery Management Solutions points towards ever more complex mathematical models for the battery chemistry and behavior, the major decision factor for the microcontroller used was to supply enough power for such a model in a small affordable package. The ARM Cortex M Series of microcontrollers is gaining widespread acceptance in the developer community due to the possibility of choosing from a number of different development and debugging tools and the possibility to change chip manufacturers whilst staying in the same architecture. The microcontroller of our choice is the *Fujitsu MB9AF312K[9]*. It belongs to the FM3 Series of Fujitsu microcontrollers which are based on the ARM Cortex-M3 architecture and sport a wide variety of peripheral units. With 128Kbyte of Flash storage for program data and 32Kbyte of Work Flash, a maximum operating Frequency of 40MHz, a number of peripheral functions including LIN,USB and multiple SPI interfaces in a small 48pin LQFP package, it is well equipped to serve as the processing unit in this device.

Data acquisition + System Basis Chip:
To measure total current, four cell voltages, total pack voltage and temperature the system is using the *AS8515 IC[3]*. This chip provides two independent ADC channels with programmable gain amplifiers to support a wide measurement range. Measurement of battery voltage is supported through resistive attenuator with disable function for power saving in standby. After analog to digital conversion and digital filtering, the resulting digital values are accessible through 4-wire serial interface. The device is powered directly from the battery through an integrated linear regulator and provides a 3.3V supply for the microcontroller. Additionally the IC integrates a LIN 2.1 transceiver which is used as the main communication interface in this application.

Active Balancer:
The balance portion of the circuit is handled by the *AS8506[2]* active balancing IC. This chips supports balancing up to 7 cells via integrated transmission gates and an external flyback transformer which is commutated via the chip. The balancing decision is made by simultaneous comparison of all cell voltages against an internal or external reference voltage. Cells which are below reference/average will receive charge packages from the external transformer. Communication with the microcontroller is handled via a 4 wire serial interface as well as the Balance done, Cell Voltage NOK, Wake and Trigger signals.

Voltage Level Translator
To translate the logic signals between the 5V domain used by the active balancing circuit and the 3.3V domain used by the rest of the circuit a "TI TXB0108 8-BIT BIDIRECTIONAL VOLTAGE-LEVEL TRANSLATOR WITH AUTO-DIRECTION SENSING AND $\pm 15 - kV$ ESD PROTECTION"[13] chip was used. The auto direction sensing feature comes in handy since we need to translate signals in both directions and didn't want to use two chips. Additionally the chip features a very low power consumption of just $4\mu A$

Flyback Transformer
In order to operate the active balancing we need a flyback transformer that translates the

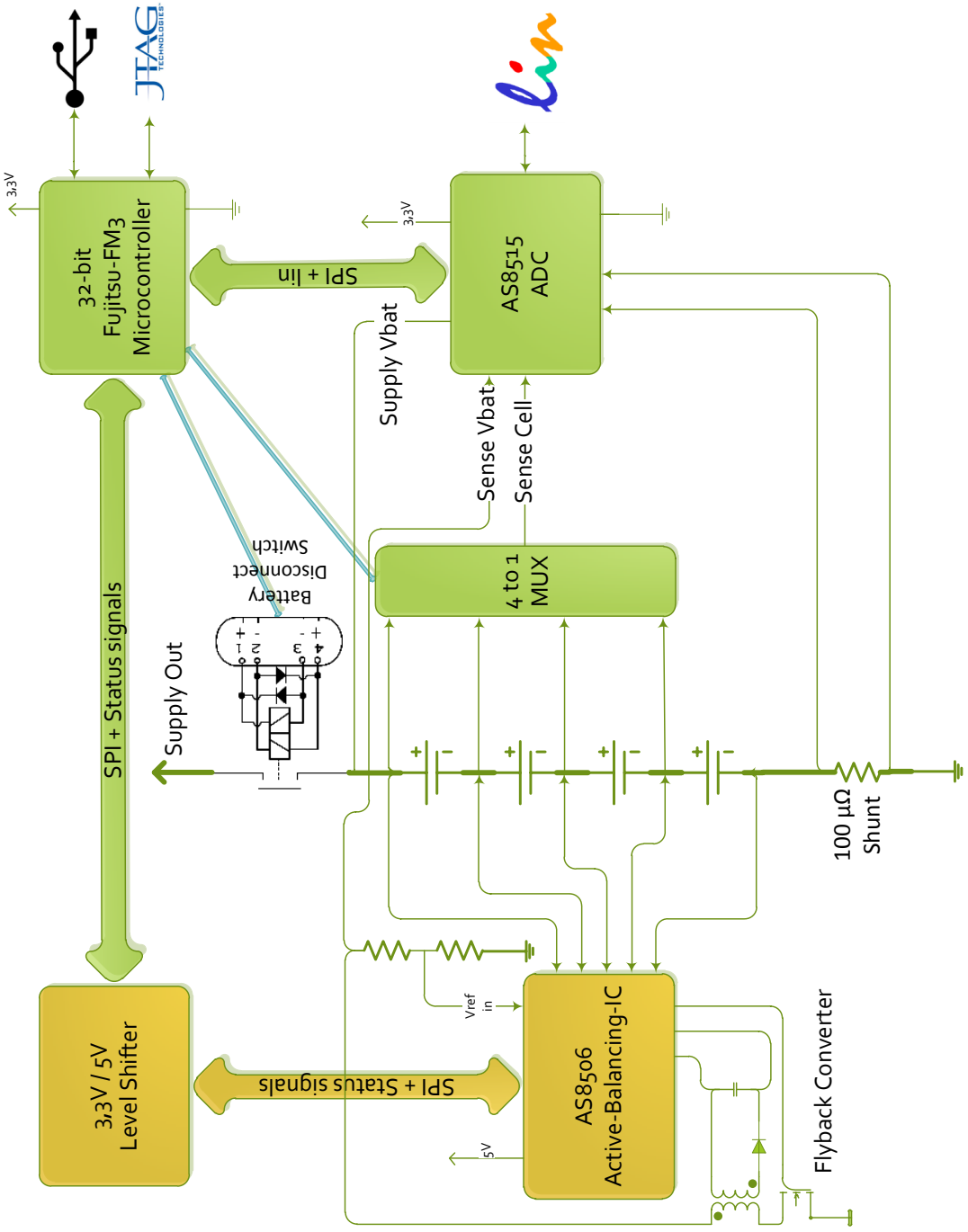

Figure 5.1: Battery Management Hardware Block Diagram

$\approx 12.8V$ pack voltage to a single cell voltage between $2.0V - 3.8V$. The typical output current should be in the range of 100mA. So we can start with the following design considerations:

$U_{in/nom} = 12.8V \;\; U_{out} = 3.8V \;\; I_{out} = 0.1A \;\; f = 100kHz \; wr(windingratio) = 1 \; duty = 25\%$

The minimum required inductance required for continuous mode operation is given by the formula:

$$L = \frac{T * wr^2 * U_{in}^2 * U_{out}}{2 * I_{ag} * (U_{out} + wr * U_{in})^2}$$

where I_{ag} is given by

$$I_{ag} = \frac{\Delta I}{2} * \frac{t_{off}}{T} \;\; = frac0.12 * 0.75 = 0.0375A$$

With this information we can calculate the required inductance to be at least

$$L = \frac{1E-5 * 1 * 12.8^2 * 3.8}{2 * 0.0375 * (3.8 + 1 * 12.8)^2} = 83.42\mu H$$

For our tests we used a Wuerth WE-FLEX[16] 749196111 type transformer. This transformer has 6 individual windings that can be connected in parallel or series. Our configuration uses 3 windings in series for both primary and secondary side, giving as a 1:1 winding ratio and an inductance of $246\mu H$. Since this is about triple the inductance necessary for continuous operation we decrease the current ripple to

$$I_{ag} = \frac{T * wr^2 * U_{in}^2 * U_{out}}{2 * L * (U_{out} + wr * U_{in})^2} = 0.0136A$$

and a typical current ripple of

$$\Delta I = \frac{2 * I_{ag} * t}{t_{off}} = 0.036A$$

In the real application however we had to increase the duty cycle to 35 to 40% to achieve the 100mA continous charge output into the battery, as can be seen in Figure 5.2

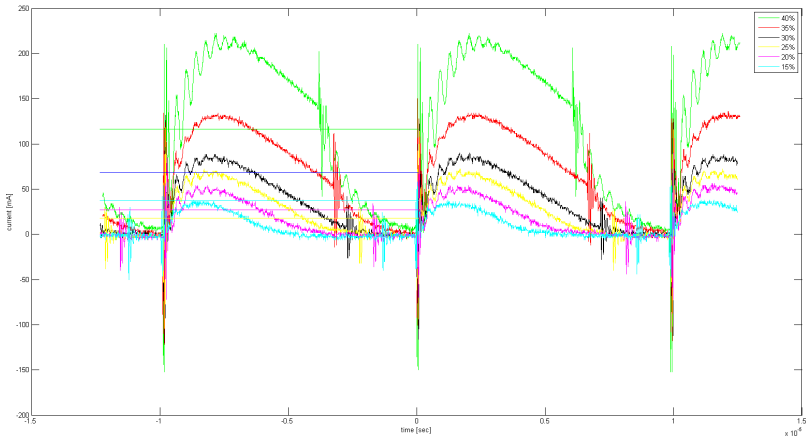

Figure 5.2: Secondary current of the flyback transformer

ESD Protection
ESD protection was added to the USB port to protect from problematic discharges in the environment of a car whilst driving. The protection device of choice was a ST USBLC6-2 [15]. It offers a two data line protection with very low capacitance and is available in a tiny SOT23-6L package.

5.3 Schematic & Layout

Schematic 9.1 and Layout 9.2 can be studied in the Appendix section of this document. At this point I want to emphasize the crucial ideas that influenced this design and what impact they have on system performance.
The board is split into two sections. On the right hand side you can see the balancing subsection comprised of the AS8506, the flyback transformer and a few necessary passive components. On the left hand side we have the AS8515 placed directly overhead the current measurement, alongside the Fujitsu microcontroller and close to the 5pin battery connection, the multiplexer. The reason for this split was to make these parts easily discernible as the board will be used for demonstration purposes at ams. For low offset measurements this design has to observe the following rules for the AS8515:

- The placement of the AS8515 should be as close as possible to the shunt (for a good thermal coupling and to keep the traces as short as possible)
- The traces leading to the shunt may not be coupled to the ground plane, instead they should be connected to the center point of the shunt solder connection only
- The voltage sense pin and the supply pin shall be routed separately to the point where power is supplied to the board, so as not to introduce voltage drop on the sense line due to the current being consumed
- The current shunt has to be soldered directly to the board using 2 3x12mm solder pads

For the 4 channel mux we used a common centroid layout and high precision resistors to build a divider that barely is affected by temperature deviation.

Furthermore the board uses a 4 layer PCB with one Ground and one Power Plane inside so as to reduce EMI.

The top portion of the board features a lot of communication ports. From left to right and top-down we have connections for LIN (P1 LIN,GND), USB, UART(J11 3.3V-Level GND, RX, TX), serial-programming (J13 MD0, VCC), the connection for the battery disconnect switch (J15 SW1, GND, SW2), the single supply (J2 VSUP, GND), the battery connector (J1, BAT4, BAT3, BAT3, BAT1, GND) and the dcdc connector (J5 DCDC, VSUP). The bottom portion features a standard JTAG connector using the pinout described in [1]. The connection for the shunt is battery at the top and load at the bottom.

At last we included an LED (D10) to indicate the operation state of the board.

5.4 LiFe Battery Management Firmware

The Software was written in KEIL μVision4 [12] in ANSI-C and debugged using a SEGGER j-link [8] JTAG adapter. The block diagrams (Figure 5.3 & 5.4) give an overview of the software operation.

The flow starts when the battery is attached to the board. The board initializes the two measurement systems and tries to load the calibration data from the work flash area of the chip. If that fails, default calibration values are used. After calibration an initial measurement is performed to check if the battery is operating within its SOA (safe operating area). In case of a success the battery is connected to the load by activating the battery disconnect switch and the board goes on doing continuous measurements. At the end of each measurement it will detect if any of the individual cells has drifted away from the average cell voltage and in

that case activates the balancing feature. The board will keep on measuring for 5 seconds and enter the standby mode if no communication is received. If however the board receives a LIN request, it will handle this request and reset the timeout timer. So in order to keep the system in measurement mode regular LIN requests have to be broadcast to the board (which accords to the LIN Standard). If at any point a measurement is taken which doesn't comply with the SOA of the battery, the battery is immediately disconnected using the battery disconnect switch, and the board enters a reset state in which it will respond to status requests but won't allow any other activity. Only a full reset, which can be done via a LIN command or by disconnecting the battery, will bring the system back to normal operation.

The available LIN requests can be seen in the block diagram (Figure 5.4). Lin communication is handled by an interrupt service routine which evaluates the checksum of the broadcast LIN message and makes the transported data available to the main program.

Communication to the AS8515 is handled by an interrupt service routine described in Figure 5.5 triggered by the new measurement available interrupt of this chip. This ISR will activate the switches in the mux to measure the individual cell voltages, besides pack voltage and temperature, and discard each first measurement after the mux is changed in order to isolate the measurement results from parasitic switching effects.

After being triggered the AS8505/06 chip autonomously handles the whole balancing process. This process is done in three phases:

1. Wait for trigger: In this phase the Chip waits for a trigger signal to begin the balancing process:
2. Compare phase: In the compare phase AS8505 detects number of cells connected to the device. The connected cell voltages are then compared with upper & lower thresholds and cell target voltage of all cells connected.
3. Balance phase: The balance phase is basically charging cycle in case of active balancing and discharging cycle in case of passive balancing. The balance phase is divided into seven time slots. The device will move through all seven time slots irrespective of number of cells connected to the device. One time slot is assigned to each cell (sequential order) for charging or discharging. In each timeslot the the CVT_NOK flag and the result of the compare phase is checked for the according cell. If the cell is marked for charging, the corresponding transmission gates are switched on and a PWM signal is sent to the flyback converter. At the end of the current time slot the PWM is turned off the switches are disconnected and the device moves to the next timeslot.

This is also explained in Figure 5.6

Functional Safety is provided by:

- Comparison of the individual cell voltage measurements against the pack voltage measurement to detect problems with the mux
- Shunt connection breaks are detected by applying a small current from the internal current source
- Measurement of the Interrupt Timing of the AS8515 to determine correct operation
- A hardware watchdog is implemented in the Fujitsu controller to detect software hangups

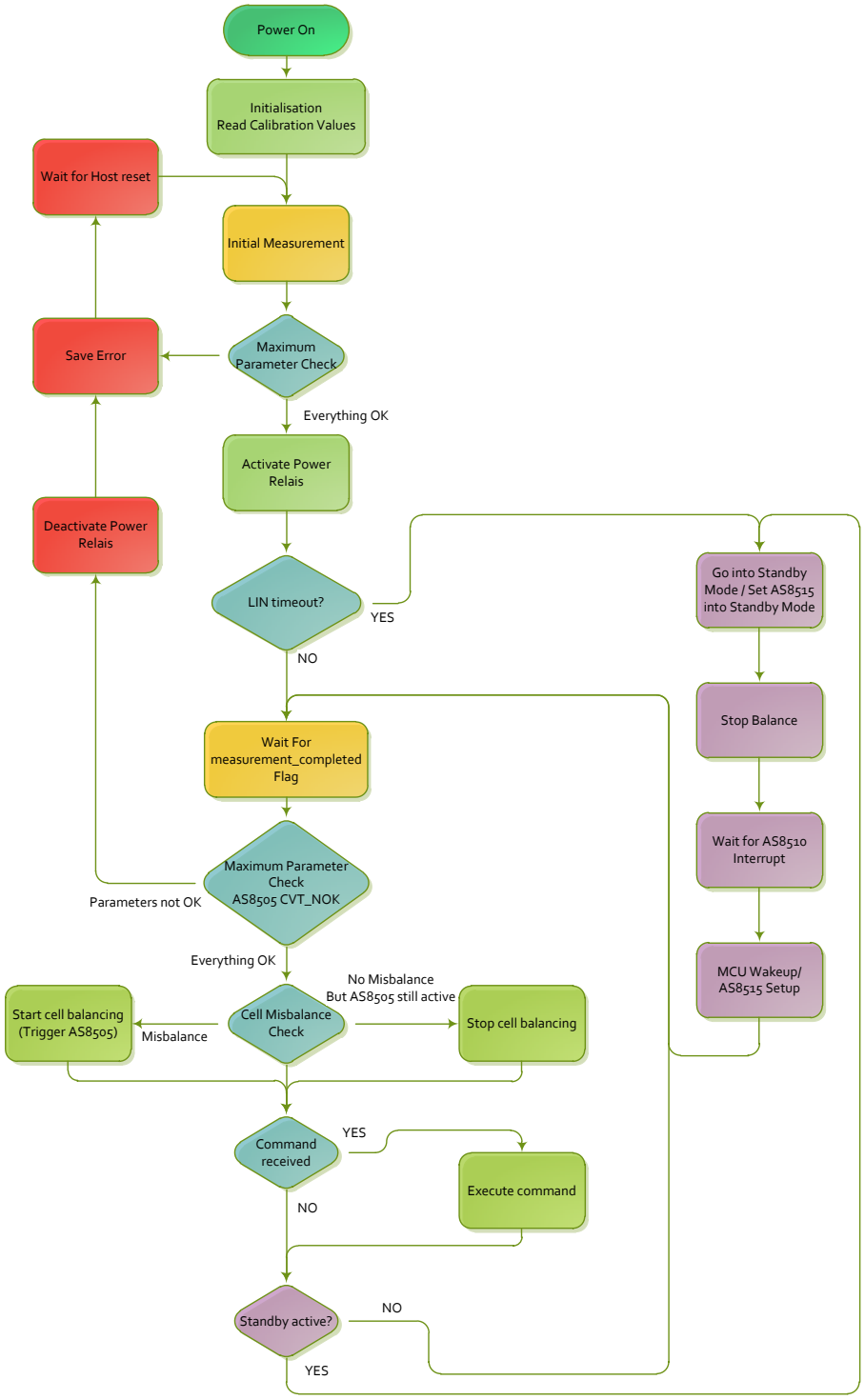

Figure 5.3: LiFe Management Main Block Diagram

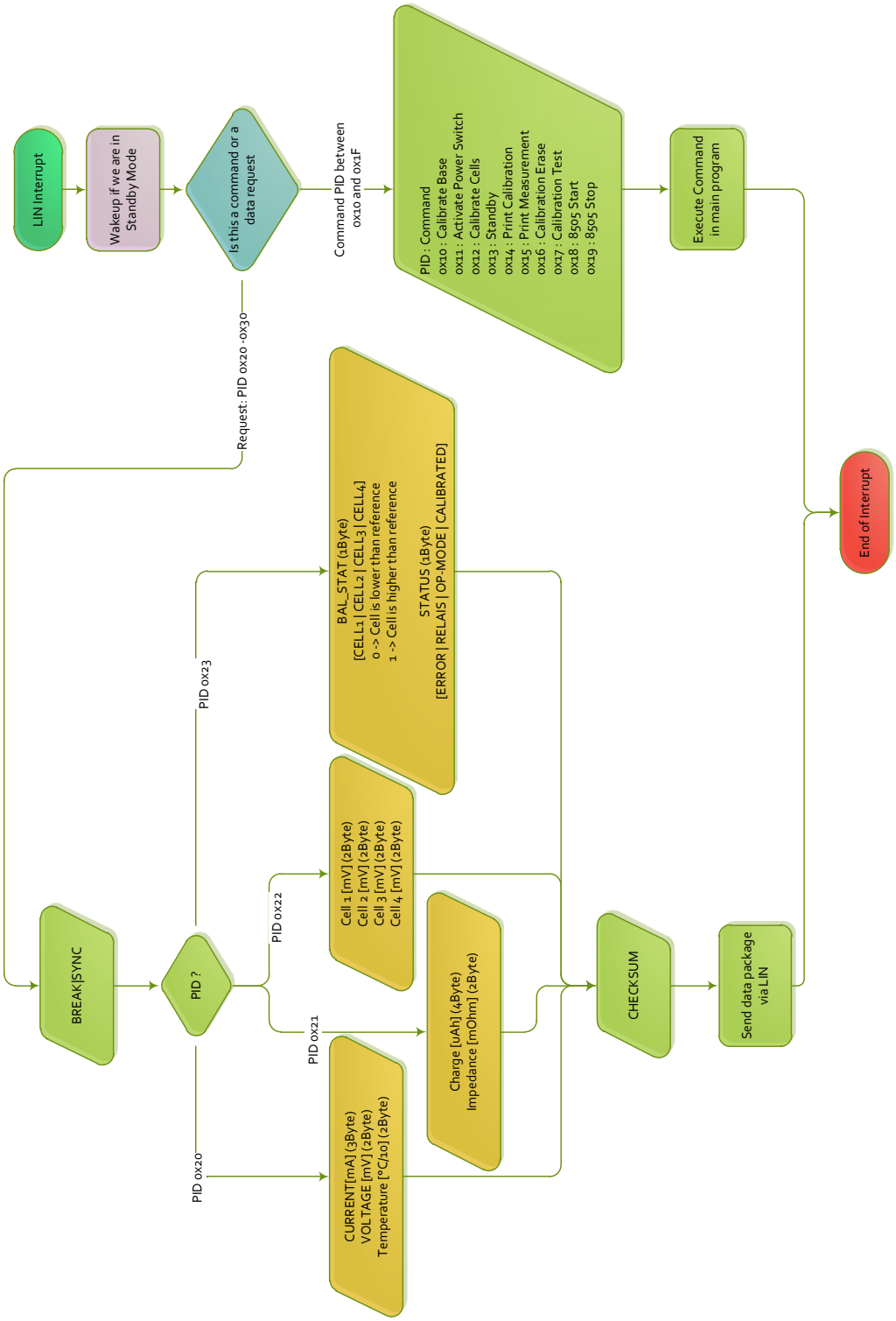

Figure 5.4: LiFe Management LIN Block Diagram

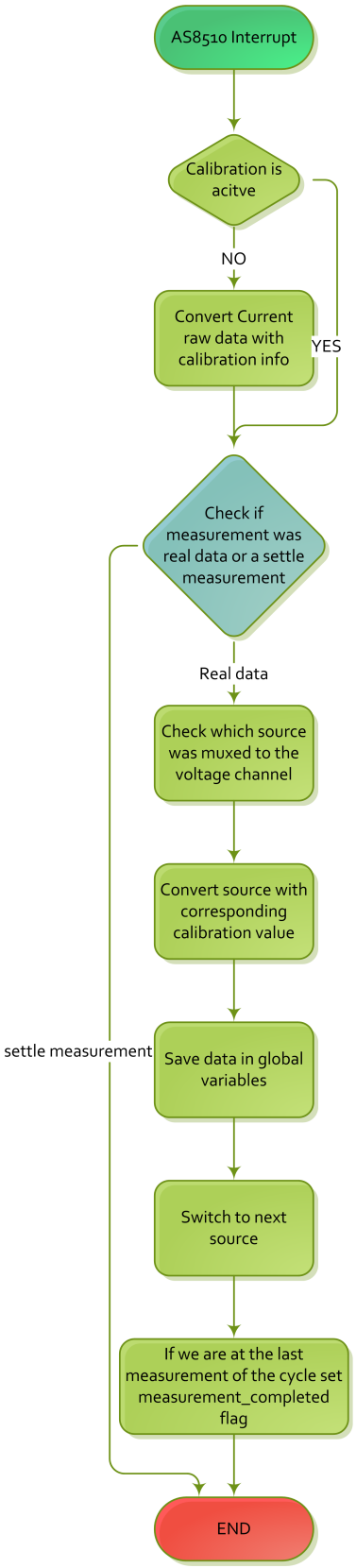

Figure 5.5: LiFe Management AS8515 Interrupt Block Diagram

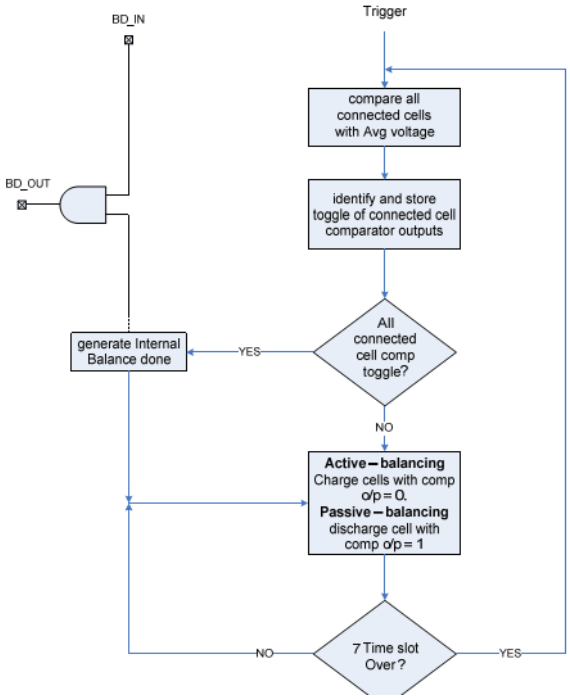

Figure 5.6: AS8505 state diagram

6 PC-Software

The PC Software was developed in C# using Microsoft Visual Studio Express [11] and runs on Microsoft Windows XP and upwards. It requires the .NET Framework 4, which can be downloaded from [10] and the Microchip CDC driver (which is part of the Microchip Application Library [22] Subfolder /USB/Device - CDC - Basic Demo/inf) for the USB2LIN Board. The software utilizes the excellent ZedGraph library [17] for all plotting functions. It was developed to display data from the battery management board which it requests via LIN interface using the USB2LIN converter board. A Screenshot of the Software can be seen in Figure 6.1

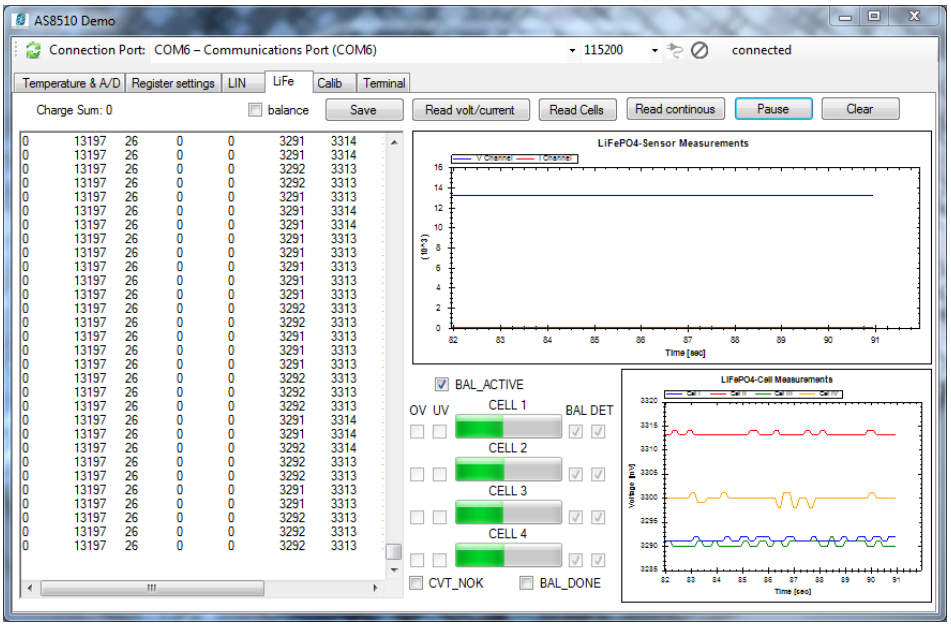

Figure 6.1: PC-Software screenshot

The Software Screen is partitioned in 3 parts.
The top part is used for connecting to the USB2LIN interface.
The Software will automatically enumerate all USB serial devices and display the associated COM port number as well as the Hardware Identifier in the drop down menu at the top of the software, which makes it easy for the user to recognize the correct device without looking it up in the Windows Hardware Manager. The button to the left is to restart the enumeration process if new devices have been plugged in since the program start. The buttons to the right allow connecting and disconnecting the device. A status field will show if the connection succeeded.
A tab handler provides access to different portions of this software.

As the software was developed for a few different demonstration boards, we will only use the Life and the calibration tab.
The Life tab consists of:

- The button portion

  Save: To save all raw data into a .csv file

Read volt \ current: to make a single read of pack voltage and current

Read cells: to make a single cell voltage reading

Read continuous: to start continuous reading and plotting of all values (update rate 10Hz)

Pause: to pause the continuous reading

Clear: to clear all display fields

- A Label showing the integrated charge sum in [mAh]
- The left hand side which will display the raw data read back from the Life-Battery Management Board.

  Values are from left to right

  Current[mA]

  Voltage[mV]

  Charge Integration Sum[μAh]

  Impedance [$m\Omega$]

  Cell Voltages 1 to 4 [mV]

  AS8505 Balance Register

  System status byte

    Bit 1 Balance is done

    Bit 2 Cell voltages are not ok

    Bit 3 Balancing active

- The upper graphics will display instantaneous current [mA] and pack voltage [mV]
- The lower graphics will display instantaneous cell voltages [mV] of each individual cell.
- The middle part visualizes the data from the balancing IC. It shows which cells are selected for balancing, and which, if any, are over or under their specified voltage range.

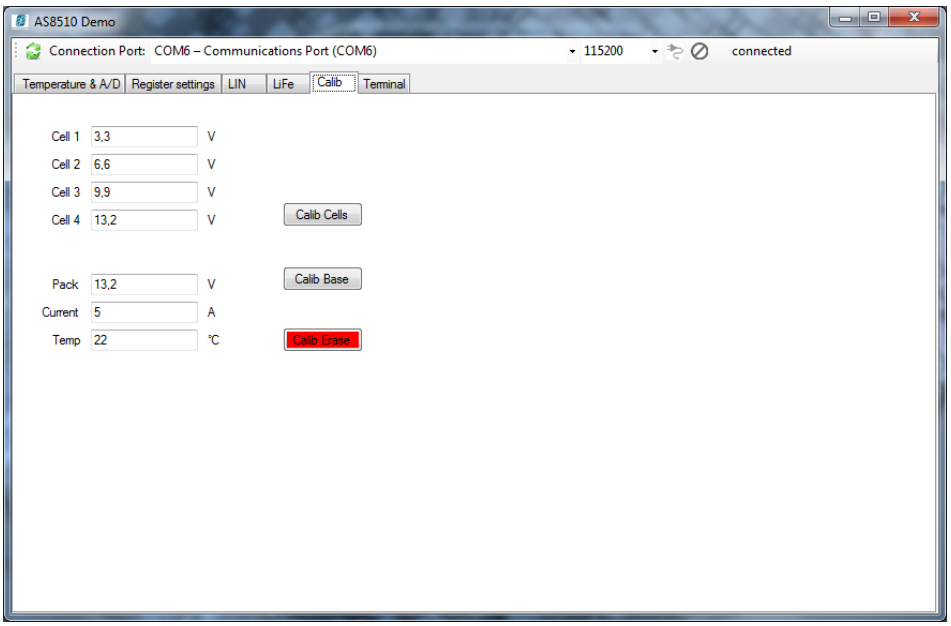

Figure 6.2: PC-Software calibration tab

The Calib tab shown in Figure 6.2 consists of:

- Input fields for the Cell voltages [V]
- Button Calib Cells: to activate the calibration. Keep in mind that a measurement will be taken and the raw data compared to the values you input in the fields, and then a calibration set is stored on the microcontroller
- Input fields for the pack voltage [V] Current [A] and Temperature[C]
- Button Calib Base: same as Calib cells but here the calibration for the whole pack and the temperature is performed
- Button Calib Erase: All calibration data will be erased from the microcontroller

The whole software is available as an installer package, which will not only install the software but also the driver into a specified directory. When Windows asks for a driver for the new device, just point it to the install directory driver sub folder.

7 Evaluation

7.1 Setup

To evaluate our Battery Management System we created two evaluation platforms:

7.1.1 Car Battery Setup

As a real world application we constructed a 12V car battery replacement consisting of 4 40Ah LiFePo4 Prismatic cells made by HiPower [6], out battery management electronics and a battery disconnect switch from Tyco electronics [14] encased in a construction of PMMA (acrylic glass). The rendering in Figure 7.1 shows the layout of the 4 individual cells. A plastic intermediate floor seperates the batteries from the electronics. The top has two terminal posts, similar to common lead acid batteries, to make replacement as easy as possible. A LIN connector is accessible from the top via an RJ12 connector.

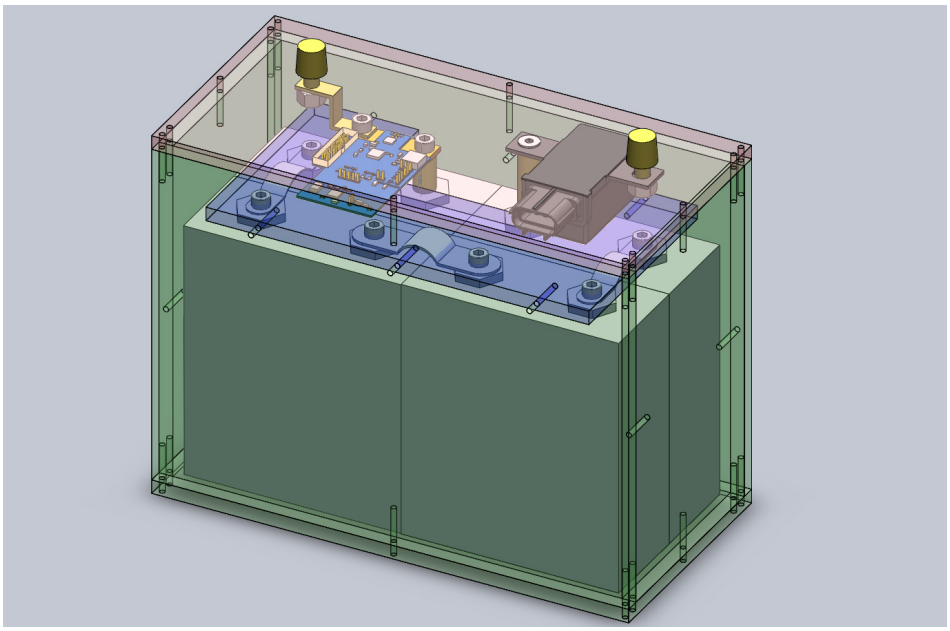

Figure 7.1: Rendering of the Battery Assembly

The batteries performance data taken from the datasheet are listed in Table 7.1

This battery was integrated in a Citroen C5 shown in Figure 7.2 & 7.3

7.1.2 Lab Setup

Furthermore we used a small 4cell LiFePO4 Battery Pack (Figure7.4) with 4Ah capacity for lab evaluation and to test the functionality of the battery management as well as the balancing.

Nominal Voltage	3.2V
Maximum continuous discharge current	120A
Maximum peak discharge current	400A
Maximum charge current	120A
Volumetric energy density	142.7Wh/l
Gravimetric energy density	95.5Wh/kg

Table 7.1: LiFe Battery Performance Data

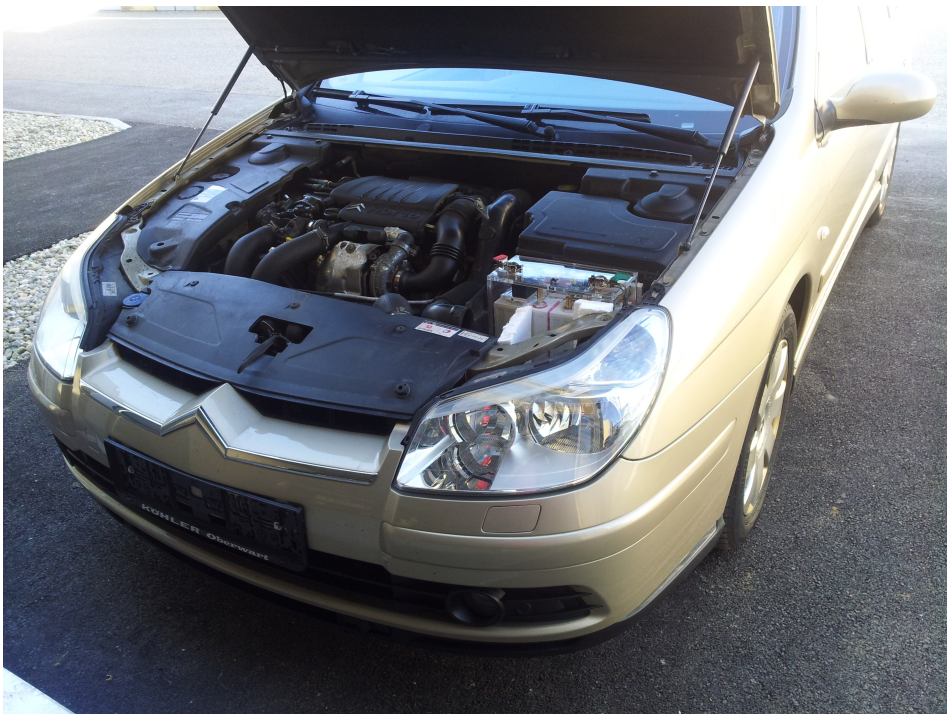

Figure 7.2: Citroen C5 with replacement battery installed

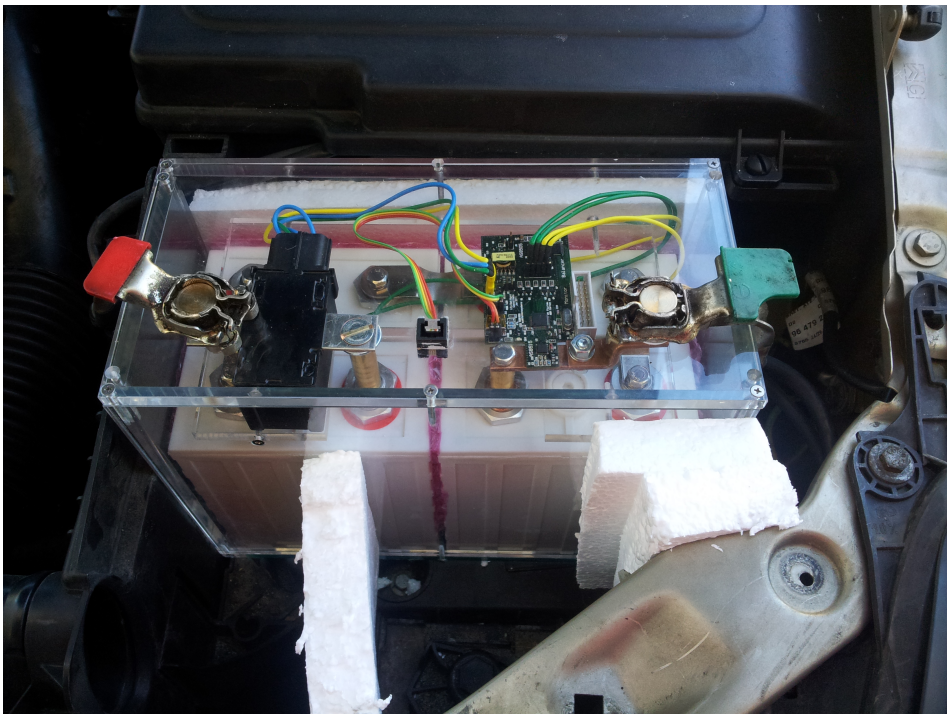

Figure 7.3: Citroen C5 replacement battery closeup

Figure 7.4: 4Cell LiFePo4 Pack from TopFuel

7.2 Balancing

To evaluate the balancing function, determine its functionality and to estimate what duty cycle would bring the best balance between time efficiency and self heating of the chip we did a couple of lab measurements with the small 4cell LiFePO4 battery.

First we compared internal and external supply for the balancing process. The difference can be seen in Figures 7.5 & 7.6

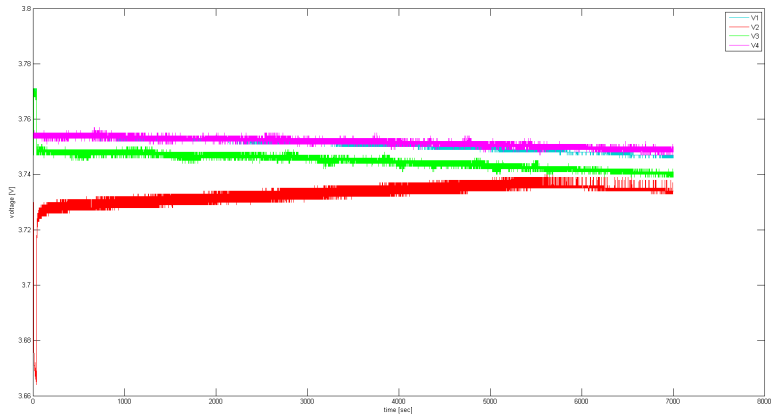

Figure 7.5: Balancing with external supply

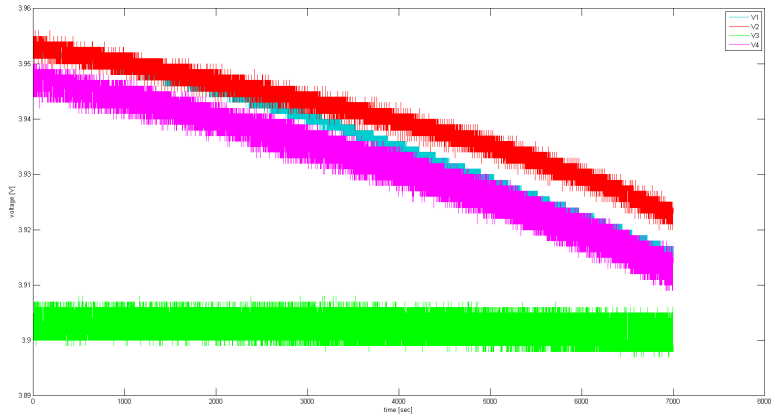

Figure 7.6: Balancing with internal supply

The reason why the balancing with external source isn't charging the battery as much as would be expected lies in the energy consumption of the board. Let us look at an example. We assume that only one of the cells is below the balancing threshold and therefore will get charged by the balancer IC. From that we can calculate that the average current being charged into the battery is:

$$\frac{100mA}{7cycles} = 14,2mA \; average \; charging \; current$$

The power consumption of the whole board in active state is roughly 12mA leaving only $\approx 2mA$ as active charging current.

During testing of the balancing features we also stumbled upon a chip internal latch up behavior which can be observed in Figure 7.7.

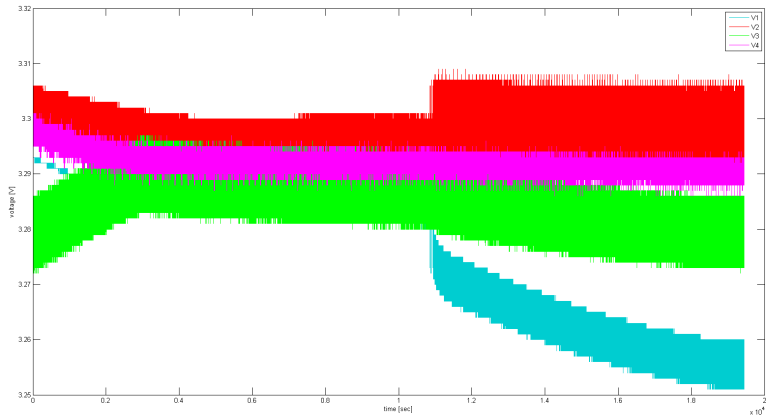

Figure 7.7: Balancing failure

The condition that triggers this behavior is when the current flowing into the lowest most cell exceeds a value of $\approx 100mA$.

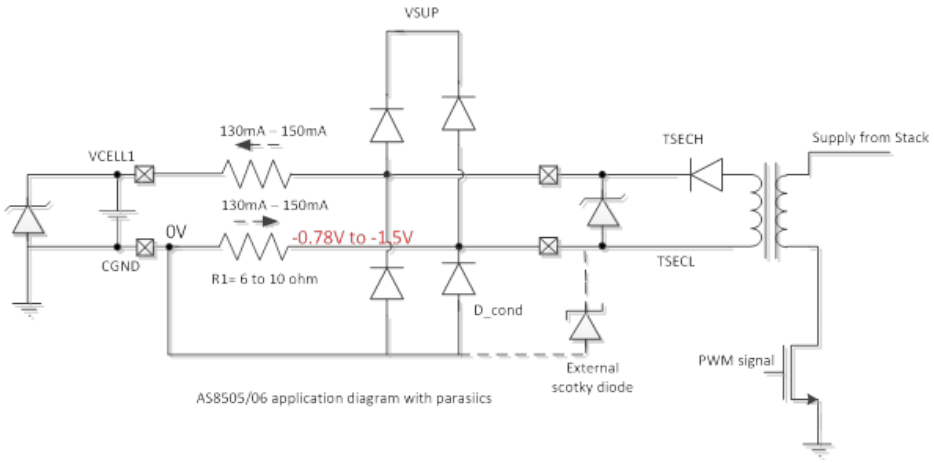

Figure 7.8: Schematic of the portion of the AS8505/06 responsible for the problem

With high charging currents from the secondary voltage drop across R1 can be about 0.78V to 1.5V. As the CGND is connected to ground and TSECL is floating, TSECL can go to -0,78V to -1.5V. This will trigger the ESD protection diode on the TSECL. Current will pass through the D_cond and can trigger the latch-up as the current is in the order of 130mA and a peak current of 300mA.

Two solutions for this problem seem viable:

- Put a 4.3V zener diode between TSECL and TSECH, which limits the over voltage but is wasteful. If the cells are fully discharged, then this may not work
- Decrease the PWM duty cycle to a value where this effect is not triggered.

A comparison of both variants can be seen in Figures 7.9 & 7.10

Judging from these graphs it seems that the added diode consumes a lot of the balancing power. Therefore it makes more sense to use lower balancing currents than to introduce this

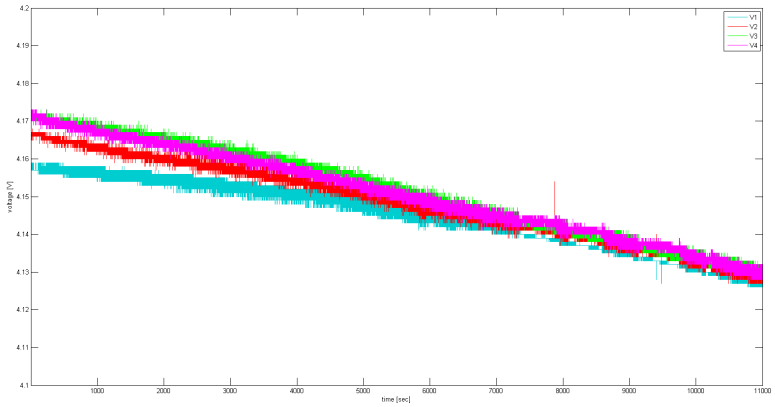

Figure 7.9: Balancing with Zener Diode

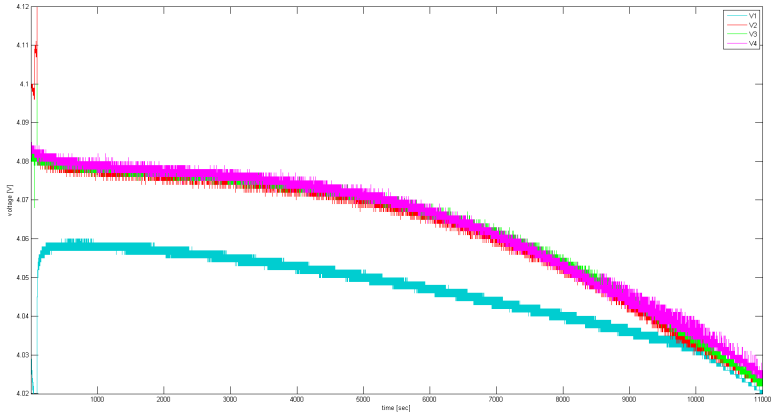

Figure 7.10: Balancing with lower current

wasteful diode.

7.3 Power Consumption

Power consumption was evaluated for the four operation states the board can enter. All measurements were done using the Agilent 34401A Digital Multimeter.

Measurement + Balancer active: In this state all electrical systems are active. The typical power consumption here is $\approx 18mA$

Measurement active: In this mode the balancer is in sleep mode whereas the microcontroller and the AS8515 remain active. Power consumption in this mode is $\approx 10mA$

Measurement standby + Balancer woken We run into this mode when the system is in sleep mode but wakes up to perform a balancer check. For this operation the AS8505/06 is woken via the WAKE pin and remains active for 1.2sec Power consumption in this mode is $\approx 9mA$

Full standby: In this mode the microcontroller enters STOP Mode, the AS8515 is in Standby Mode 1 and the AS8505/06 is also in standby. Current consumption in this mode is $\approx 1mA$, which is quite a lot for full standby. The reason for this lies in a few shortcomings of this hardware version as follow:

1. The resistive voltage divider for the reference of AS8505/06 is always connected drawing about $300\mu A$ of current. This will be solved by the next chip sample version which has an internal disconnect transistor to power down the voltage divider.
2. The AS8515 Standby Mode is not working correctly due to a bug in the recognition of the EN pin signal leading to another $300\mu A$ of current consumption. This bug is also known and will be corrected in the final chip version.

The total power consumption can be split in the following sections:

Table 7.2: LiFe Management Power Consumption

Device	Estimated Power Consumption	Measured Power consumption
AS8515	5	5
MB9AF312K	3,9	4
AS8505/06	10	9

As can be seen from Table 7.2 the estimated power consumption matches the realized power consumption pretty well. A total power consumption of below 20mA is very much acceptable for a system such as this. The standby power consumption however should be reduced down to $\approx 100\mu A$. An issue that will be resolved by the final version of the respective chip sets.

7.4 Accuracy

Accuracy measurements were done using an Agilent E3631A 80W Triple Output Power Supply, an Agilent 34401A Digital Multimeter and a TTi CPX400A PSU. We tested linearity of all four individual cell voltage measurements as well as common mode dependency of the individual voltage measurements. In addition we also tested the full voltage and current channel for accuracy.

In the first test (shown in Figure 7.11) we sweeped each of the individual cell voltages in 0.1V increments and measured all cell voltages. As can be seen the voltage sweep doesn't influence any of the other cell voltage readings. Also the measured voltage remains accurate down to 0.5V (observable in Figure 7.12) at which point we start making a measurement error, which

is irrelevant, because the cell voltages won't ever reach such a level unless they have long passed their Safe Operating Area.

The second test was to show how well the system performs if the voltages are varying quickly. For this test we ramped the single voltages in 1V increments and observed all the measured cell voltages. We can observe that the voltage jumps in one cell can lead to a wrong reading in another cell. The reason for this is the way the individual cell voltages are obtained. We are using a multiplexer to measure the individual tabs of the cell stack. The cell voltages are obtained by subtracting these values from each other. Because these values can not be obtained simultaneously (because we are only using one ADC channel) it can happen that one of the cell values is already updated to the new stack voltage while the other is not, resulting in a faulty calculation of cell voltage. This can be seen in Figure 7.13. To circumvent this one could use multiple synchronous ADCs or a sample and hold circuit for each voltage. Both solutions are costly, require more board space and are not required in the practical application since the single cell voltages can't be properly read during operation of the alternator anyway. It is however important that the total pack voltage measurement and the current measurement are synchronized, so that we may calculate internal resistance of the battery pack or use a model to correctly estimate the battery's SOC.

The last test was to show total pack measurement accuracy for current and voltage. This test was performed by stepping the voltage from 5V to 18V in 0.1V increments and the current from 1A to 40A. Measurement accuracy is limited by the accuracy of the supplies in use, which is 0.05% + 20 mV for the Voltage supply and $0.3\% \pm 2digit(20mA)$ for the current source and the remaining output ripple, which is $\leq 350Vrms/2mVpp$ for the voltage supply. The voltage channel was however corrected by the usage of the Agilent Multimeter with an accuracy of 0,0020% of the measurement value + 0,0006% of the range value giving an uncertainty of

$$0.002 * 0.01 * 12 + 0.0006 * 0.01 * 100 = 0.84mV$$

The conclusion we can draw from these measurements is that the accuracy of the measurement system is within the uncertainty of the supply. Overall accuracy can be established to be better than 2mV of 0.04% for the full voltage measurement and better than 80mA or 0.5% for the current measurement.

7.5 Safety Shutdown

The autonomous shutdown in case the battery SOA was breached turned out to be too unforgiving to allow normal operation in a car (as can be seen in Figure 7.16). The load during ignition triggered the safety shutdown more often than not.In order to allow normal operation in a car while still being able to safely disconnect the battery in case of an error a shutdown counter was integrated in the battery management. With each measurement cycle if one of the SOA criteria is violated this count is incremented, each cycle if it is not the case it will be decremented (down to 0). If the counter reaches a critical level, a shutdown is initiated. The value of the counter was chosen so that the extreme conditions during ignition would be tolerated for $\approx 4seconds$.

7.6 Car Application

As a final test we put the battery management to use in a real world car application. As explained above a Citroen C5 was retrofitted with a LiFePO4 battery pack. This car was then started and driven around for a number of times. Figure 7.17 shows one of the test drive cycles we recorded with the software.

We can distinguish a number of load cases in this cycle:

- Standby from 0 to 20 seconds
- Wakeup from 20 to 60 seconds

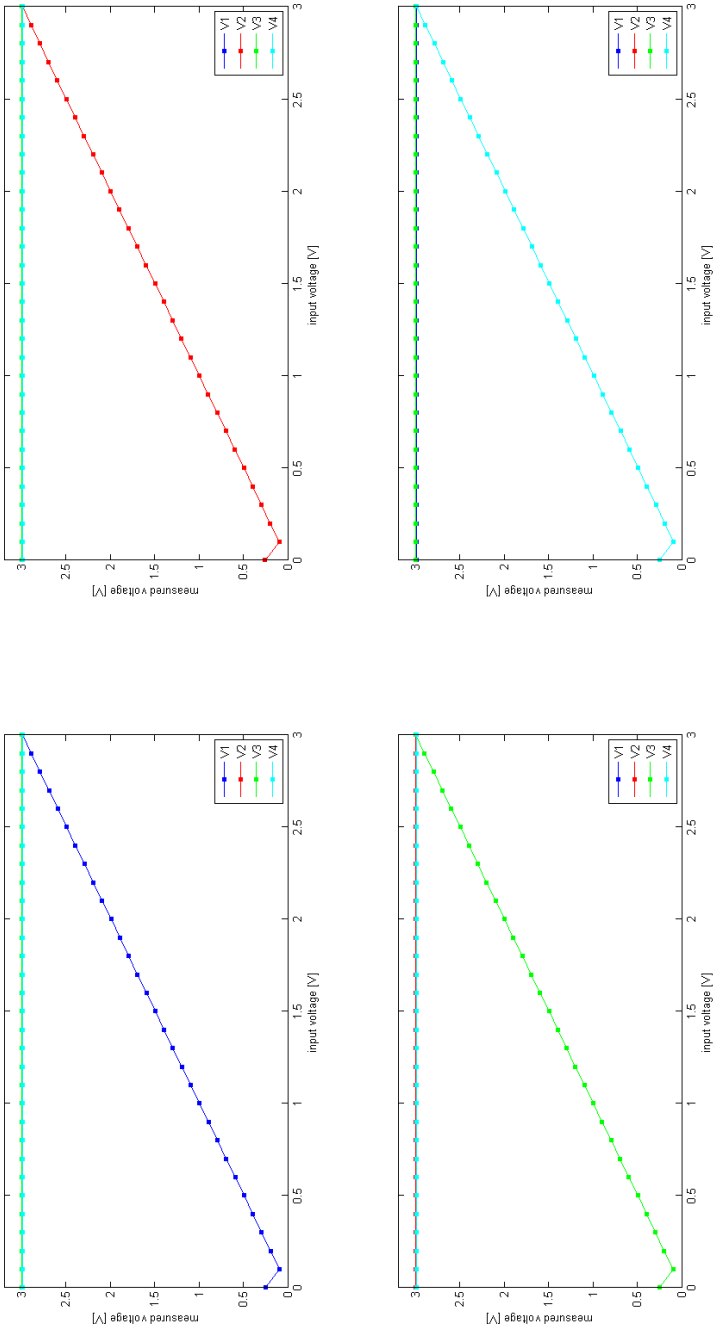

Figure 7.11: Common Mode rejection and linearity for the four individual channels

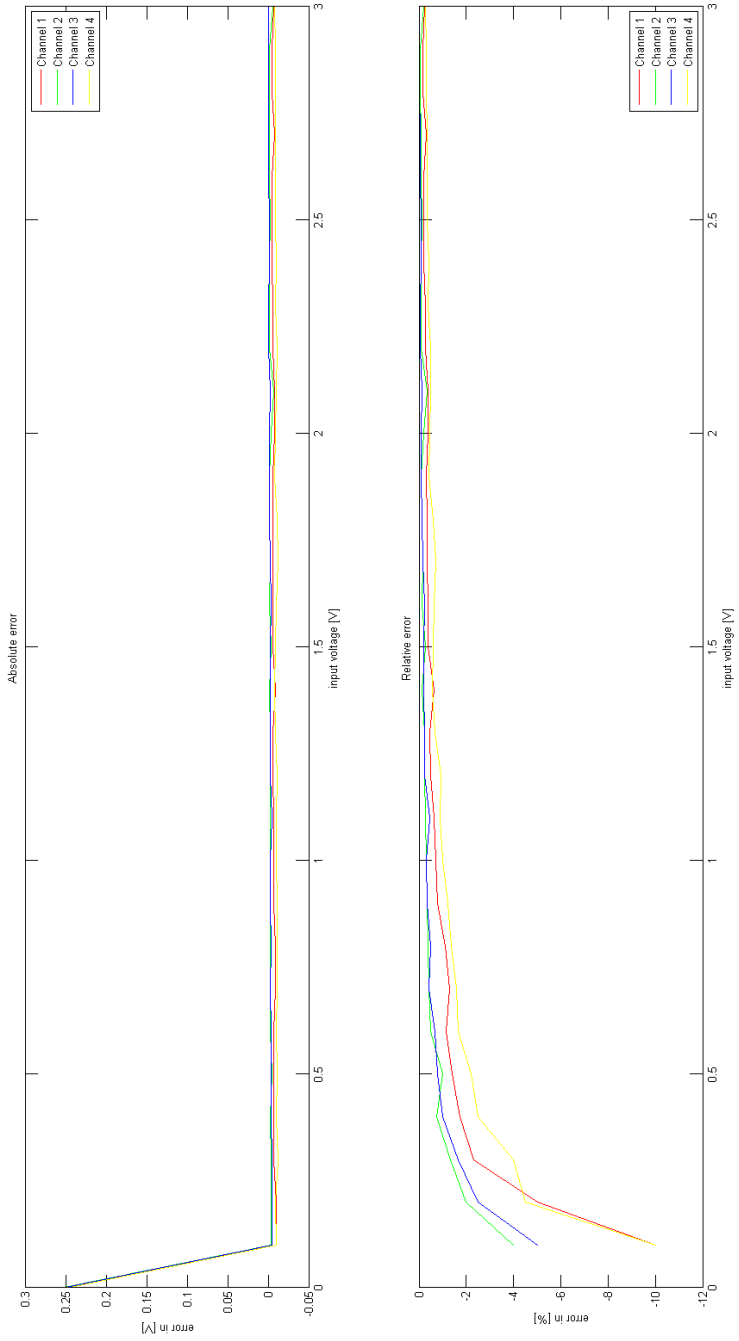

Figure 7.12: Absolute and relative accuracy for the four individual channels

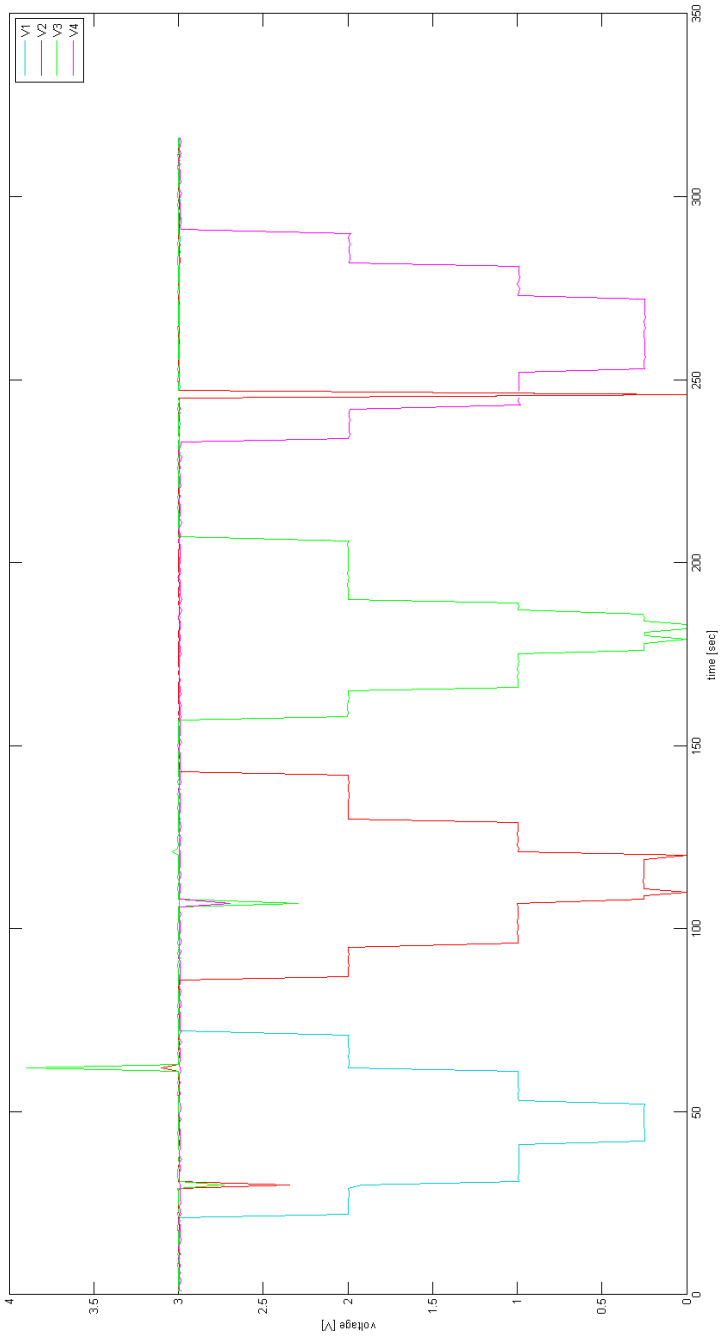

Figure 7.13: Common Mode rejection for fast changing voltages

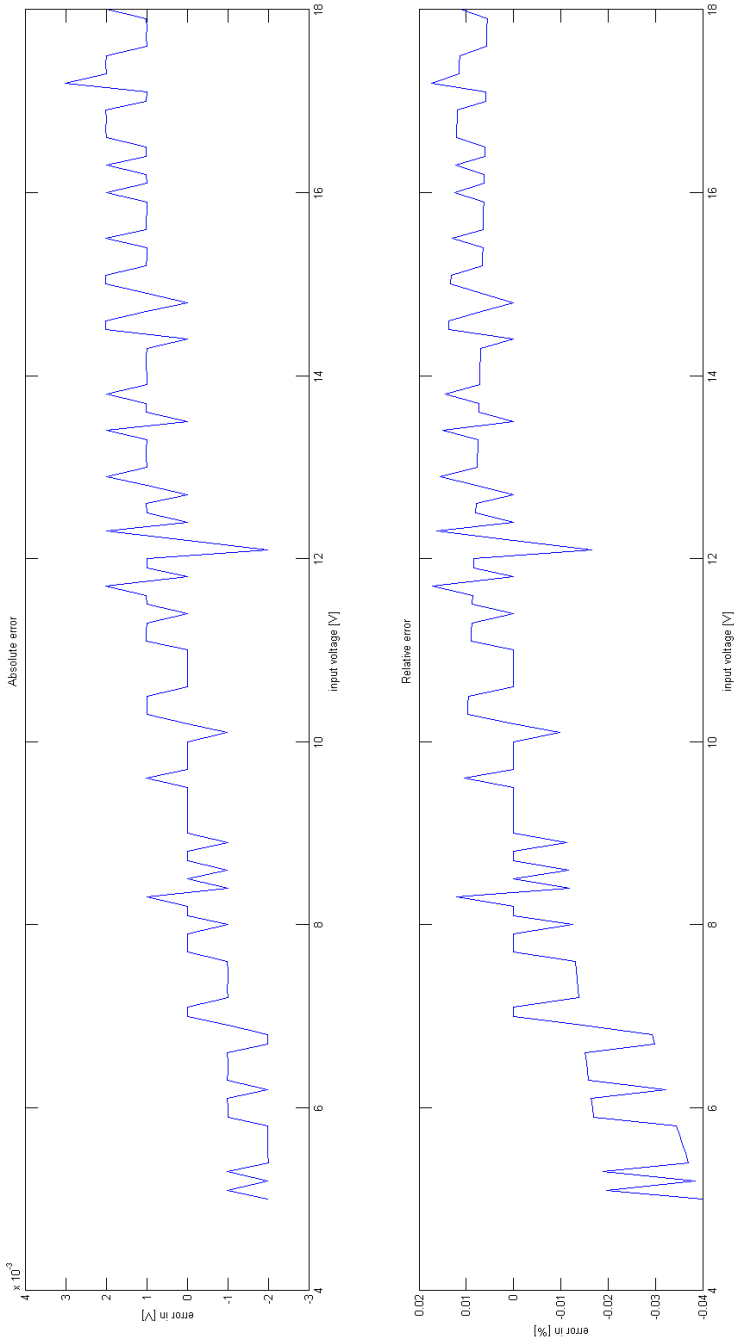

Figure 7.14: Total Voltage measurement error

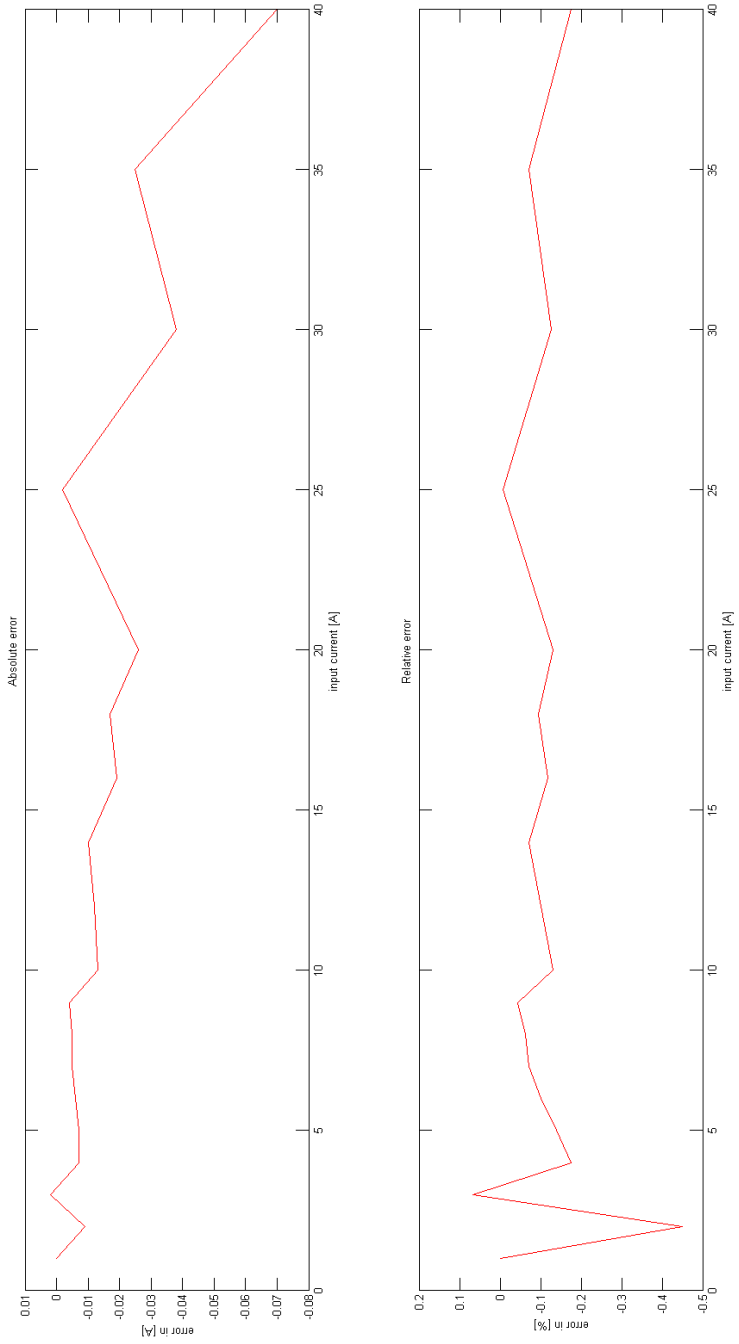

Figure 7.15: Total Current measurement error

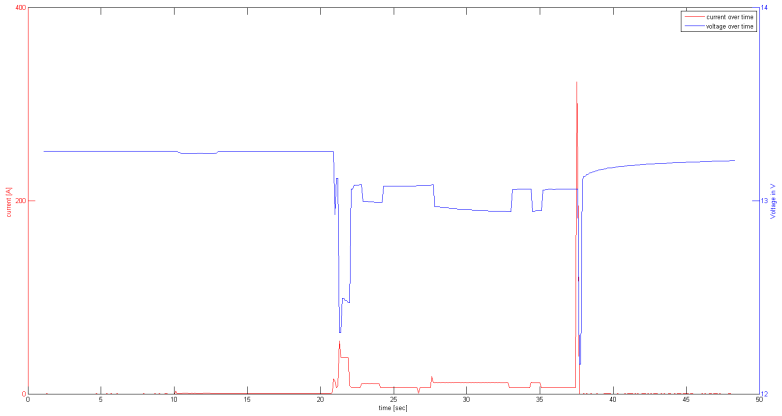

Figure 7.16: Shutdown during ignition of the car

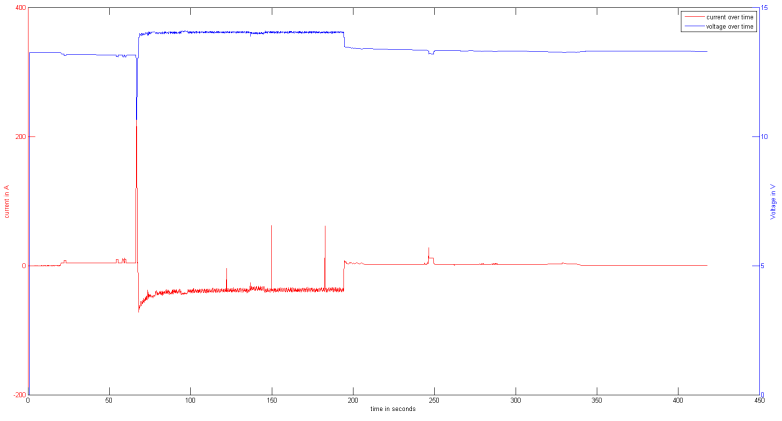

Figure 7.17: Test Car driving cycle

- Ignition at 60 seconds with current draw of $\approx 350A$
- Charging phase where the alternator charges the battery while the diesel engine is running. Charge current $\approx 50A$ till 200 seconds
- Shutdown where current is again drawn from the battery up to 350 seconds
- Subsequent Power down of the various systems

Looking closer at the current and voltage curves we can see that the current has a huge ripple of $\approx 10A$, which is also represented in the single cell voltage measurements seen in Figure 7.18

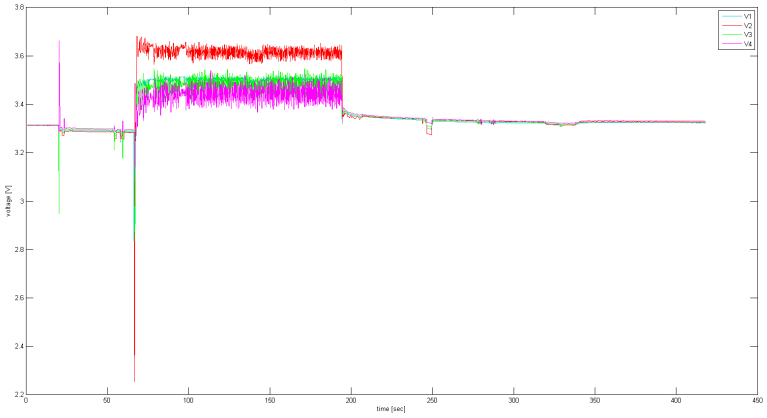

Figure 7.18: Test Car driving cycle single cell plot

The reason for this lies in the way a car alternator operates. It is basically a dc excited synchronous generator with a regulator that changes the excitation current so that the output voltage of the alternator stays at $\approx 14.2V$ and the output current stays below the maximum rated current for the alternator. The output of the alternator therefore is a 3-phase AC voltage which is then rectified by a 6 diode network. The battery is then used as a buffer to stabilize the board voltage, leading to the ripple in current flowing to the battery.

8 Conclusion

From the measurements performed and discussed in the evaluation section 7 we can conclude that the replacement of existing lead acid batteries with modern LiFePo4 based battery packs is a viable and promising solution for the problems car manufacturers are facing with todays implementation. The 4 series connected LiFe cells match very well with the voltage domain of a classic 12V battery and therefore do not require any changes in the alternator or the board power system. The life management and balancing board discussed in this thesis performs all the necessary functions to safely operate such a new cell stack in any classic car application nicely. Therefore it provides a ready-made swap in solution for existing cars as well as a good testing platform. With the addition of a full featured battery state of charge and state of health diagnostic algorithm it will also support advanced features like active breaking and mild hybrid support.

9 APPENDIX

9.1 USB2LIN Schematic and Layout

9.2 LiFe Battery Management Schematic and Layout

9.3 3D Drawing of the Retrofit Battery

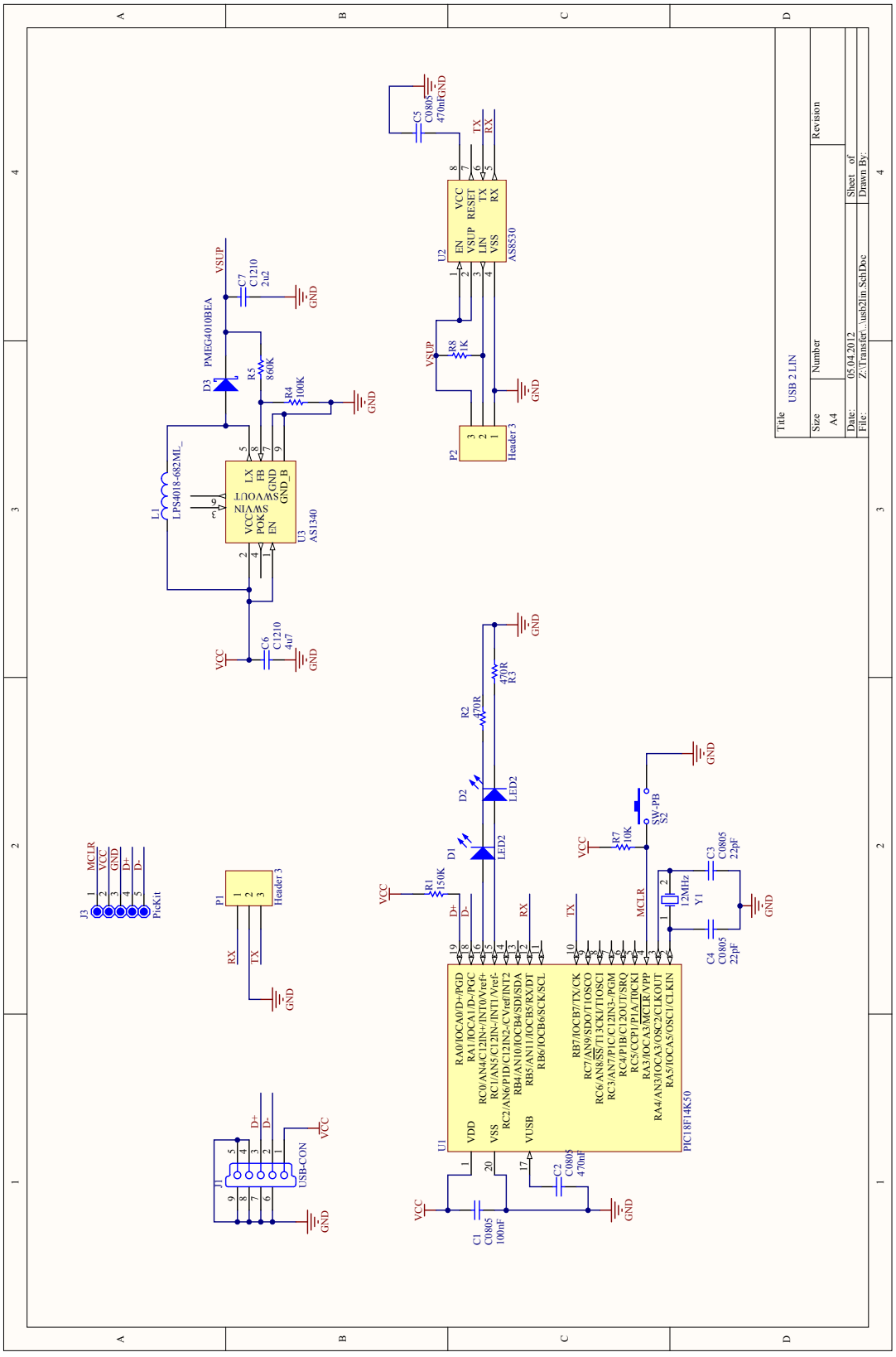

Figure 9.1: USB2LIN Schematic

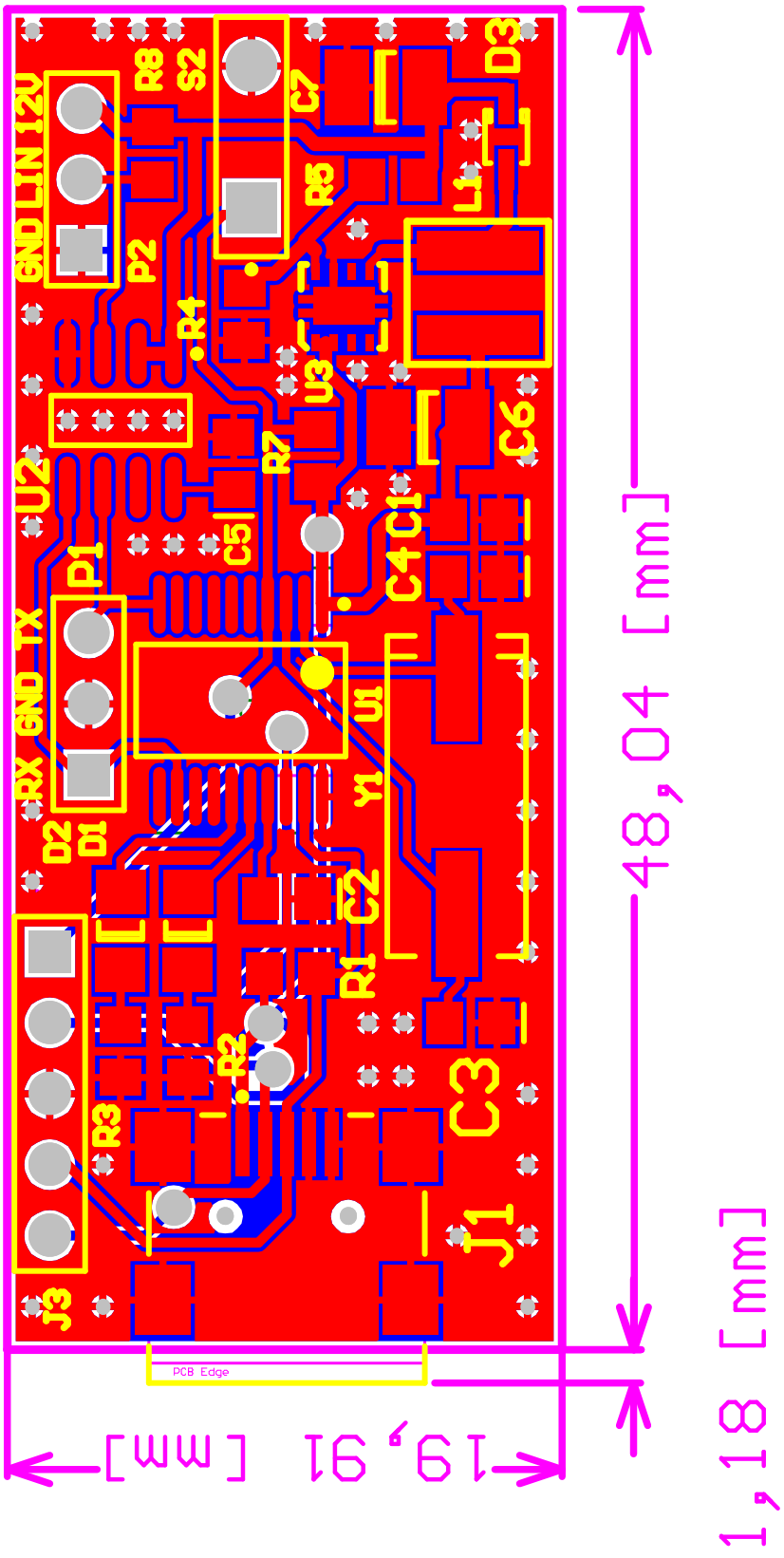

Figure 9.2: USB2LIN Layout

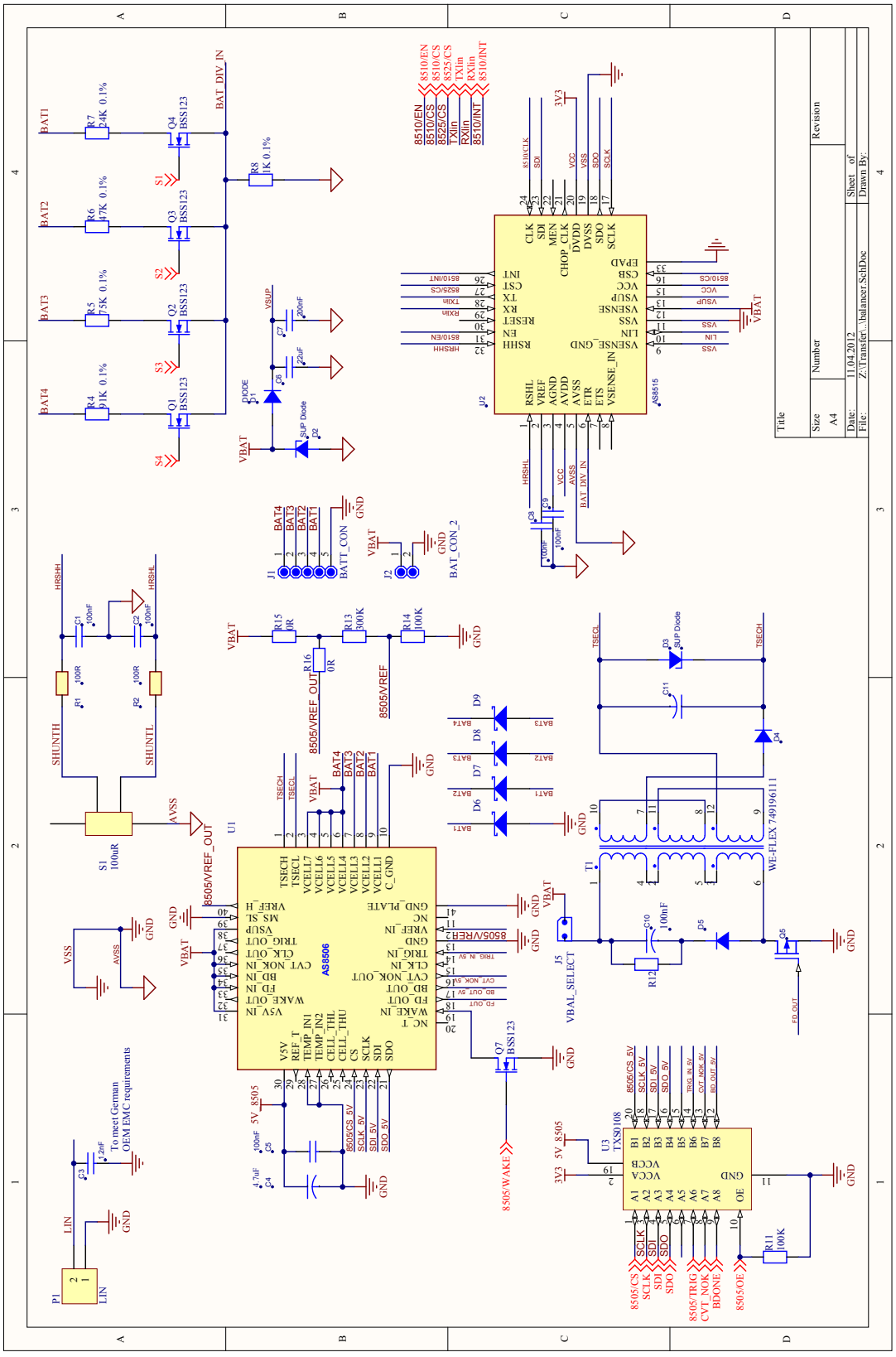

Figure 9.3: LiFe Management Schematic Page 1

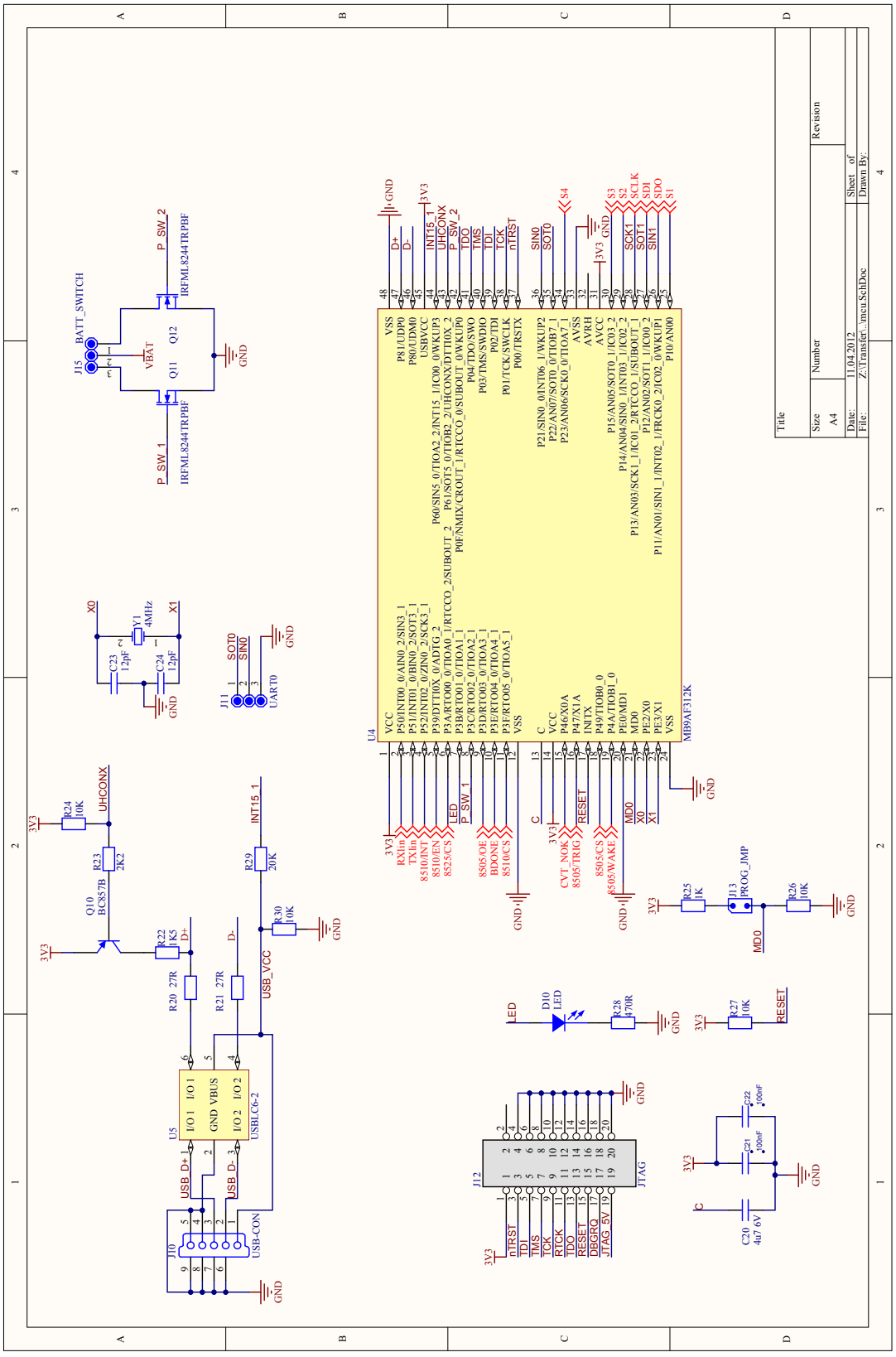

Figure 9.4: LiFe Management Schematic Page 2

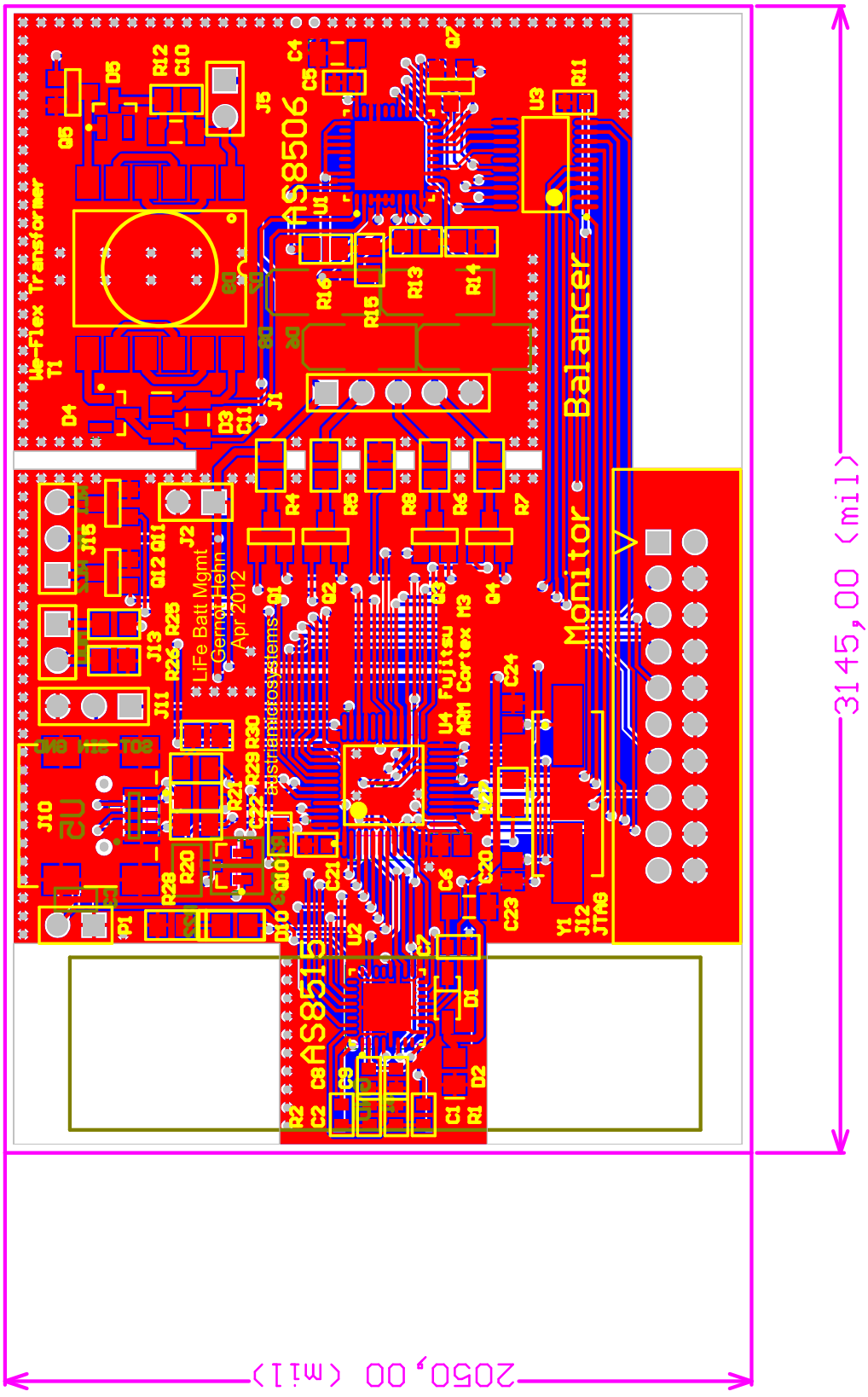

Figure 9.5: LiFe Management Layout

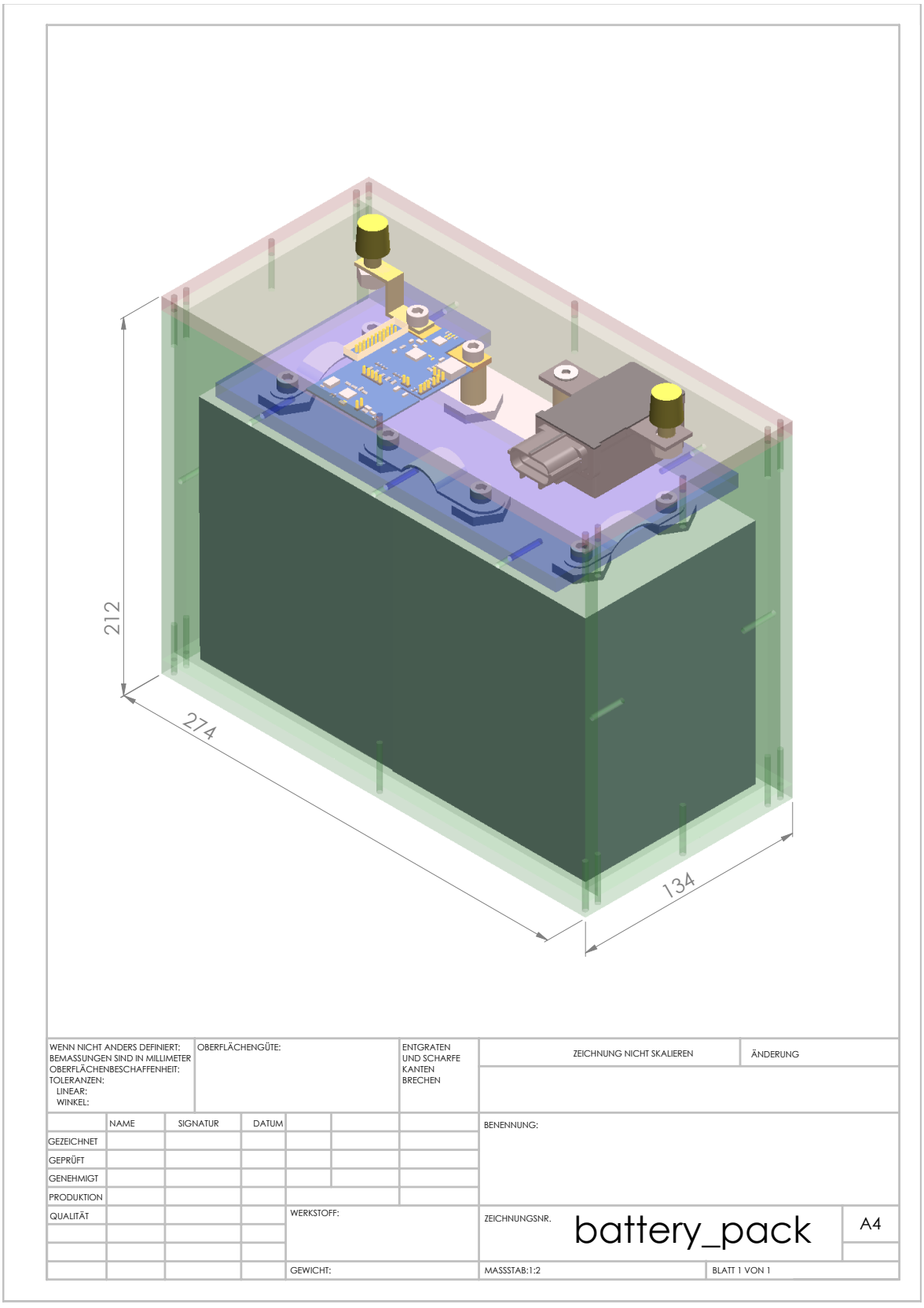

Figure 9.6: Long side board

List of Figures

Bibliography

[1] Arm standard jtag. http://www.keil.com/support/man/docs/ulinkpro/ulinkpro_hw_if_jtag20.htm.

[2] As8506 datasheet.

[3] As8515 datasheet.

[4] Batteries. http://homepages.which.net/~paul.hills/Batteries/BatteriesBody.html.

[5] Datasheet sp36-12. http://www.greensaver.cn/en/Product/manage/upload/picupload/sp36-12.pdf.

[6] Hipower lifepo4 battery list. http://www.hipowergroup.com/downdb.php?file=uploadfile/2011/0115/20110115123901167.pdf.

[7] Introduction lifepo4. http://www.swisstmeeting.ch/tl_files/images/Energy_Power/Technologie%20LiFePO4%20Akkus%20PSE.pdf.

[8] J-link. http://www.segger.com/jlink.html.

[9] Mb9a310k series datasheet. http://edevice.fujitsu.com/fj/DATASHEET/e-ds/MB9AF311K-DS706-00029-0v01-E.pdf.

[10] Microsoft .net framework installer. http://www.microsoft.com/downloads/de-de/details.aspx?familyid=0a391abd-25c1-4fc0-919f-b21f31ab88b7.

[11] Microsoft visual studio. http://www.microsoft.com/germany/express/.

[12] μvision ide overview. http://www.keil.com/uvision/.

[13] Txb0108 product page. http://www.ti.com/product/txb0108.

[14] Tyco battery disconnect switch. http://www.te.com/catalog/pn/en/1-1414939-4?RQPN=1-1414939-4.

[15] Usblc6-2 product page. http://www.st.com/internet/analog/product/93353.jsp.

[16] We-flex flexible transformer for dc/dc converter. http://katalog.we-online.de/kataloge/eisos/index.php?language=en&pf=WE-FLEX.

[17] Zedgraph. http://zedgraph.sourceforge.net/index.html.

[18] Sony recalls notebook computer batteries due to previous fires. http://www.cpsc.gov/cpscpub/prerel/prhtml07/07011.html, 2006.

[19] Laptop fires prompt sony battery recall again. http://www.wired.com/gadgetlab/2008/10/laptop-fires-pr/, 2008.

[20] Lifepo4 batteries: A breakthrough for electric vehicles. http://www.metaefficient.com/rechargeable-batteries/innovative-lifepo4-batteries-electric-vehicles.html, 2008.

[21] Lin specification package revision 2.2a. *LIN Consortium*, 2010.

[22] Microchip application libraries. http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2680&dDocName=en547784, 2012.

[23] Gernot Hehn. Diploma thesis: Design of a battery balancing prototype. June 2010.

[24] Battery Tender. Battery charging basics. http://batterytender.com/resources/battery-basics.htm, February 2012.