Degree Final Project

Design of a modular BMS for EV focused on MotoStudent

Diseño de un BMS modular para EV enfocado a MotoStudent

Author

Xavier Anzuela Recasens

Supervisor

David Asiain Ansorena

Escuela Universitaria Politécnica La Almunia
2016

ESCUELA UNIVERSITARIA POLITÉCNICA
DE LA ALMUNIA DE DOÑA GODINA (ZARAGOZA)

Degree Final Project

Design of a modular BMS for EV focused on MotoStudent

Diseño de un BMS modular para EV enfocado a MotoStudent

424.16.96

Author: Xavier Anzuela Recasens

Supervisor: David Asiain Ansorena

Date: September 2016

CONTENTS

LIST OF FIGURES

LIST OF TABLES

INDICE DE ILUSTRACIONES

1. ABSTRACT

The project is based on the design of a Modular Battery Management System for a battery pack that will be in the Eupla Racing Team prototype racing in Alcañiz during the MotoStudent IV event.

The present memory explains the development of such design. It starts presenting a state of the art and a theoretical framework to give the reader some background and understanding before starting with the technical work. Knowledge of the existing technologies is a must to be able to design a reliable BMS.

The next section explains the development of the prototype. It is divided into several sections. MotoStudent regulations and the battery pack for which we are designing the BMS will be introduced to be able to start with the design.

We will first carry out a market study of the available monitoring chips and search for one with the optimal features for our application. Once the chip is selected the electronic design will start taking into account the suggestions given by the manufacturer in the datasheet. Also the extra features needed for our application will be added. For instance, the conditioning circuit for the chosen temperature sensors; in our case, NTCs.

For the hardware development we will have to choose an appropriate balancing current for a competition application and make the corresponding calculations such as trace width to withstand the current applied. A meticulous component selection will be elaborated to ensure we have a reliable prototype.

Finally, software will be divided into two stages: software development and the interface. The former will be based on the chip libraries offered by the manufacturer and adapted to our application, the latter will be developed with LabVIEW. The LabVIEW interface will be a key feature to ensure a testing procedure and a quick way to check the battery pack state during the competition.

Keywords:

Battery Management System, LiPo Technology, Cell Monitoring, Cell Balancing, Battery Security.

2. RESUMEN

Un sistema de administración de baterías (BMS sus siglas en inglés) es mandatario al emplear baterías basadas en Litio, este se encarga de la seguridad de las baterías y puede extender su vida útil más allá de un 30 % en los mejores casos.

Dada la inestabilidad de los sistemas de baterías basadas en litio la necesidad de emplear este sistema se ha incrementado y las celdas de Litio-Ion o Litio-Polímero, con sus altas tasas de carga y descarga, son óptimas en entornos de alta exigencia, por ejemplo, los vehículos, pero las ventajas que este sistema aporta poseen un precio, y es el de las altas temperaturas generadas en los ciclos de carga que aumenta la volatilidad de las celdas con el consiguiente riesgo de explosión.

El BMS monitoriza el estado de las celdas tanto en voltaje como temperatura y dependiendo del modelo también lo pueden hacer con la corriente. Dicha información pasa a un módulo inteligente (Maestro/Microcontrolador) que emplea los datos para balancear las celdas desbalanceadas. Con esto se consigue extender la vida de las baterías o bien desconectarlas para evitar situaciones de riesgo para los usuarios. [3]

Existen 3 tipologías: Distribuida, modular o centralizada, y 2 tipos de balanceo: Activo y Pasivo.

Para comenzar se debe evaluar la cantidad de celdas a monitorizar y la topología del paquete de baterías. Tal como se menciona, el paquete está compuesto por 130 celdas, 26 por substack y 5 substacks en total así que analizaremos las diferentes configuraciones topológicas y elegiremos la óptima.

Para comparar claramente como influencia cada topología en nuestra batería (Tabla 1) se compararán las características más relevantes: Cantidad de placas necesarias, cantidad de cables por placa, adaptabilidad del BMS si el paquete de baterías debe ser reducido o aumentado una vez que el prototipo está construido y una sencilla estimación de costes. A las tres primeras características se les asignará el mismo valor y la estimación del coste, mientras construimos un BMS para un prototipo de competición, puede ser una desventaja pues es más importante el desempeño del BMS en caso estudiado.

Se emplearán 3 colores: Verde, Amarillo y Rojo con 2,1 y -1 puntos respectivamente y la combinación con más puntos será la topología elegida.

Los términos de designación serán los siguientes:

- A mayor cantidad de circuitos, la complejidad del sistema aumenta, así como el espacio requerido.
- A mayor cantidad de cables en un circuito se incrementa la complejidad de obtener una configuración de cables sin perturbaciones.
- El que un BMS sea adaptable o no será definido por su flexibilidad para monitorear mayor o menos cantidad de celdas o substacks en caso de futuros cambios.

Tabla 1. Comparativa chips

Topología	nº Placas	Cables/Placa	Adaptabilidad	Estimación de coste	Puntuación
Distirbuido	130+Master	2	Sí	Alto	2
Modular	5+Master	27+18(Temp)	Sí	Medio	**4**
Centralizado	1	131+78(Temp)	No	Medio	0

Como se puede observar las topología y distribución de batería hacen que las topologías distribuida y centralizada sean inapropiadas mientras que la modular parece la indicada para nuestro propósito. No solamente obtiene las puntuaciones más altas sino también carece de características negativas. Se procede a decidir el tipo de balanceo.

Lo último que se necesita para proceder con el desarrollo es tipo de balanceo. Teniendo en cuenta lo mencionado en el marco teórico para cada tipo de balance encontramos que el más realista es el balanceo pasivo. Como es más sencillo y lo necesitamos para un prototipo de competición, y no para entornos urbanos, necesitaremos 5 giros al circuito que demandarán carga rápida y no requerirá de balanceo de la batería durante la descarga porque esto implicará una complejidad innecesaria en la competición. También, habiendo realizado una investigación previa en competiciones más maduras como lo es Formula Student Eléctrica encontramos que la mayoría de los equipos emplean balanceo pasivo por encima del activo.

Por consiguiente:

Topología -> Modular

Tipo de balanceo -> Pasivo

Una claro que tipo de BMS se quiere diseñar y desarrollar. Se procederá a buscar un chip de monitoreo de batería que pueda controlar un grupo de celdas empleando el balanceado pasivo. Como agregado se tendrá en cuenta a la regla 3.5.5 que determina que el 30% de la temperatura de las celdas debe ser monitorizado. En consecuencia, se requerirá un chip con sensor de temperatura incluido en sus circuitos o bien circuitos que admitan la conexión de sensores externos de temperatura con acondicionamiento también externo. En la investigación de mercado se encuentra el siguiente documento, [27]: Una tabla comparativa de chips de diferentes compañías que puede verse en el Anexo de tablas.

De la mencionada tabla se observan chips con aplicaciones comerciales de BMS, estos pertenecen a Texas Instruments y Linear Technology, los primeros han sido empleados por algunos profesionales para aplicaciones de administración de baterías [28] y los segundos con su chip LTC6803-1 han sido empleados comercialmente en el Freemens BMS y proyectos universitarios [29] [30].

En la Tabla 2 se ha realizado una selección de los chips de las dos compañías y resaltado sus características principales.

Tabla 2. Comparativa chips.

COMPANY	CHIP	NUMBER OF CELLS	BALANCING	EXTRAS
LINEAR	LTC6801	2-12	NO	NO BALANCING
	LTC6802-1	2-12	PASSIVE	2TEMP/2GPIO
	LTC6802-2	2-12	PASSIVE	2TEMP/2GPIO
	LTC6803-1	2-12	PASSIVE	2TEMP/2GPIO
	LTC6803-2	2-12	PASSIVE	2TEMP/2GPIO
	LTC6803-3(UPGRADE 03-1)	2-12	PASSIVE	2TEMP/2GPIO
	LTC6803-4(UPGRADE 03-2)	2-12	PASSIVE	2TEMP/2GPIO
	LTC6804-1	2-12	PASSIVE	BALANCING/5GPIO
	LTC6804-2	2-12	PASSIVE	BALANCING/5GPIO
TEXAS INSTRUMENTS	BQ76940	9-15	PASSIVE	BALANCING
	BQ76930	6-10	PASSIVE	BALANCING
	BQ76920	3-5	PASSIVE	BALANCING
	BQ76PL536A	3-6	PASSIVE	BALANCING/3GPIO
	BQ76PL56A-Q1	3-6	PASSIVE	BALANCING/3GPIO

La tabla anterior posee la información requerida para elegir uno de los chips que cubran los aspectos principales de topología y tipo de balanceo elegidos previamente. Todos poseen balanceo pasivo y para más de 3 celdas y poseen conectividad para medir temperatura. La característica necesaria para construir nuestro BMS modular siguiendo la reglamentación de MotoStudent. Los chips más avanzados de ambas empresas (LTC6804 y BQ76PL54A-Q1 [31] (una versión mejorada del BQ76PL54A)) son las opciones a considerar al cumplir con los requerimientos necesarios. Entonces se analizarán todos los detalles para tomar la decisión correcta considerando el tiempo necesario para el desarrollo del BMS y nuestros recursos. Una explicación más detallada de la tabla anterior se encuentra en el anexo de tablas.

La diferencia principal entre ambos chips de LTC son la comunicación con la placa base y los de TI no poseen comunicación SPI aislada, aunque se puede agregar. Tampoco incluye librerías y ejemplos de software que dificultan el desarrollo, por lo que nos inclinamos con el chip de LTC. El LTC-6804-1 de Linear Technology.

El BMS se compone de una unidad maestra y un número de esclavos determinado por la cantidad de substacks que se consideran módulos. La unidad maestra recibe la información y la distribuye en una configuración "Daisy-chain" y también se comunica con el bus CAN. Otras dos características del maestro son la lectura de la salida del paquete de baterías y el control de los contactores, por otro lado, los esclavos controlan el balance de tensión y temperatura de los substacks.

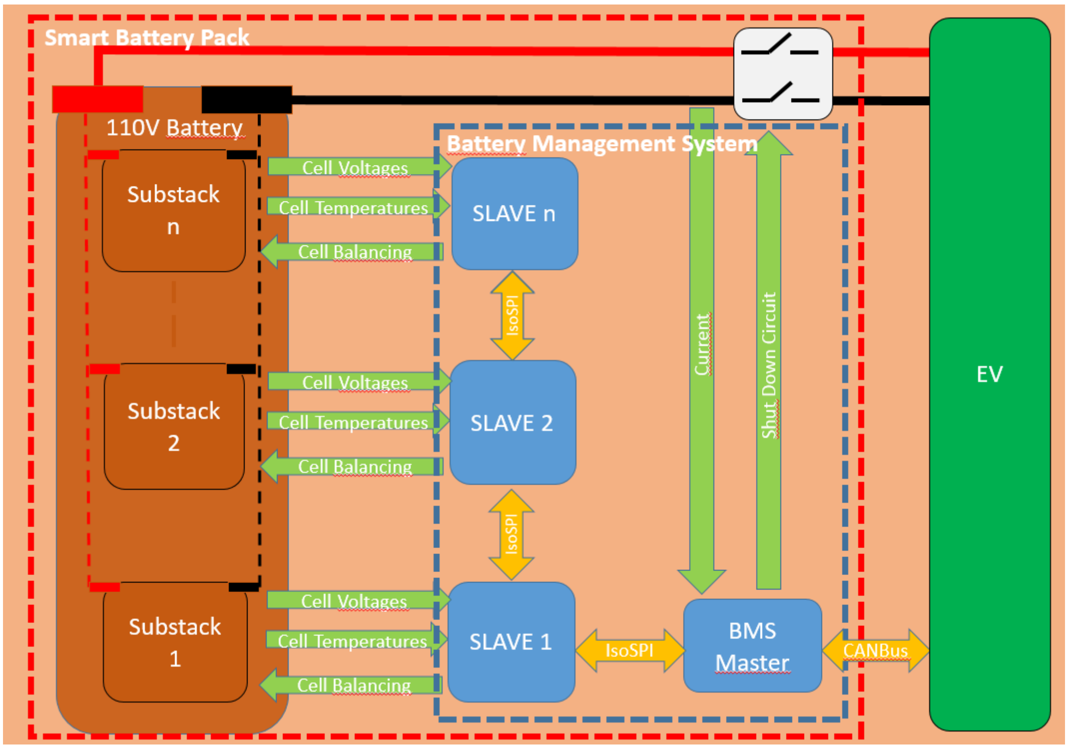


Ilustración 1. Diagrama General BMS

En resumen, nuestro BMS y Paquete de Baterías Inteligente unido se protege de altas y bajas tensiones y altas temperaturas mientras envía datos al Bus y entrega electricidad al vehículo.

El paquete de baterías consiste en 26 celdas en serie y cada chip posee 12 canales haciendo necesarios 3 chips para nuestro sistema. Para tener una potencia similar en cada chip ya que al menos se requieren 4 celdas conectadas para su funcionamiento, lo que resulta en 2 chips conectados a 8 celdas y uno a las 10 restantes.

La configuración por defecto del circuito cuando todos los canales son leídos se muestran en la (Ilustración 2) y se ve al C0 conectado a V- y V+ a la celda más alta con una resistencia de protección. También se observa las salidas "S" conectadas a los mosfets que controlan a las celdas para que entreguen el voltaje requerido por el chip, por último, observamos los filtros RC para prevenir lecturas erróneas.

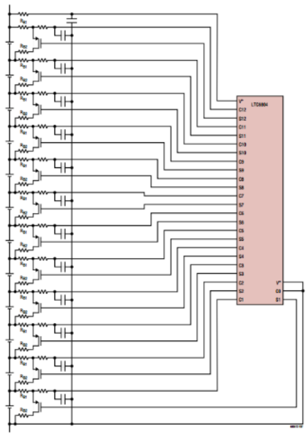


Ilustración 2.Configuración del chip

Ahora se debe atender al circuito recomendado por el fabricante y que sucede cuando menos de 12 celdas están conectadas. Para conectar 8 celdas se eliminan los canales C12, C11, C6 y C5 para 8 celdas y para 10 se eliminan sólo C12 y C6 y para remover una celda se pone en cortocircuito el canal con el anterior y el canal de balanceo no se conecta.

La configuración va a ser de 8-10-8 canales para monitorear y balancear 26 celdas y la (Ilustración 3) se muestra la configuración adaptada a las necesidades del equipo.

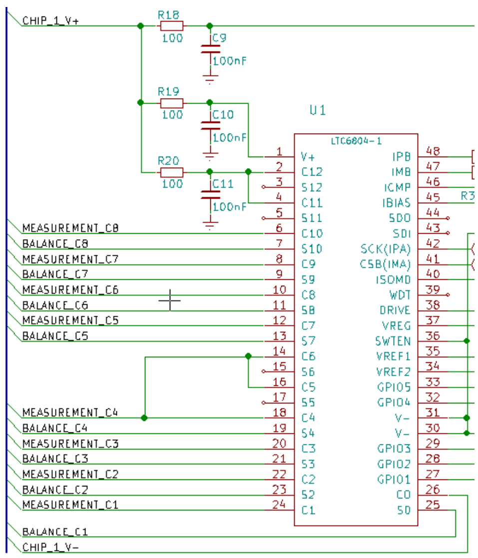


Ilustración 3. Configuración para medición de 8 canales

 Autor: **Xavier Anzuela Recasens**

424.16.96

Vamos a añadir un led en paralelo con la resistencia de balanceo para tener un indicador de balanceo. Esto nos facilitará el testeo y verificación del prototipo.

Para alimentar el chip utilizaremos un regulador lineal simple hecho a partir de un NPN en combinación con el DRIVE pin del chip como nos indica el fabricante en la (Ilustración 4).

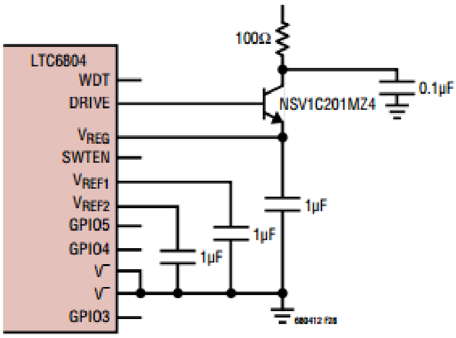


Ilustración 4. Regulador lineal simple para el LTC6804-1

Un aumento de la tensión ideal rompería el chip y un descenso causaría un mal funcionamiento. Una configuración DC-DC sería la recomendada, pero aumenta el espacio físico requerido y la solución para nuestro caso es el mostrado en la (Ilustración 4). Para construir el regulador la potencia que llega al colector del NPN se protege con un filtro RC y se realiza un bypass en el emisor del NPN con un capacitor de 1 uf. Por último, para prevenir el sobrecalentamiento del NPN al usarlo como regulador se remplaza por uno con mayor tolerancia térmica.

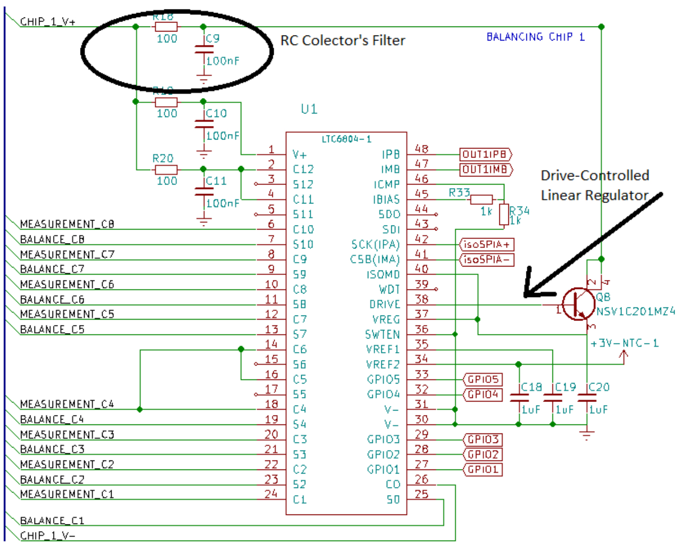


Ilustración 5.Configuración Regulador lineal.

Para la medición de la temperatura utilizaremos sensores NTC. El chip tiene 5 canales GPIO en nuestro caso utilizaremos 4 canales por chip para asegurar llegar al

30% del sensado requerido por MotoStudent. Utilizar todos los canales con menos celdas que 12 podría resultar en problemas de alimentación también utilizaremos NTC de 100k para reducir el consumo. La (Ilustración 6) muestra nuestra configuración para un chip. $Vref_2$ ofrece una salida constante de 3V regulados.

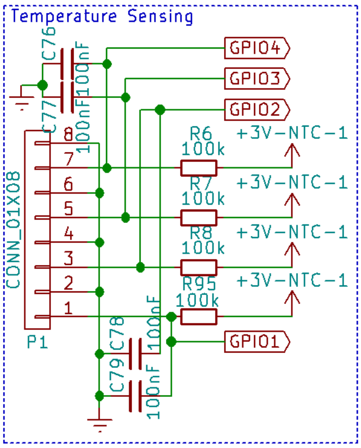


Ilustración 6. Módulo de sensado de temperatura

Este circuito nos dará una señal de voltaje en función de la temperatura para obtener la temperatura calcularemos la función de transferencia en base a las indicaciones del datasheet del componente. En este caso la NTC de AVX NJ28RA0104H. La grafica que veremos a continuación (Ilustración 7) muestra el resultado.

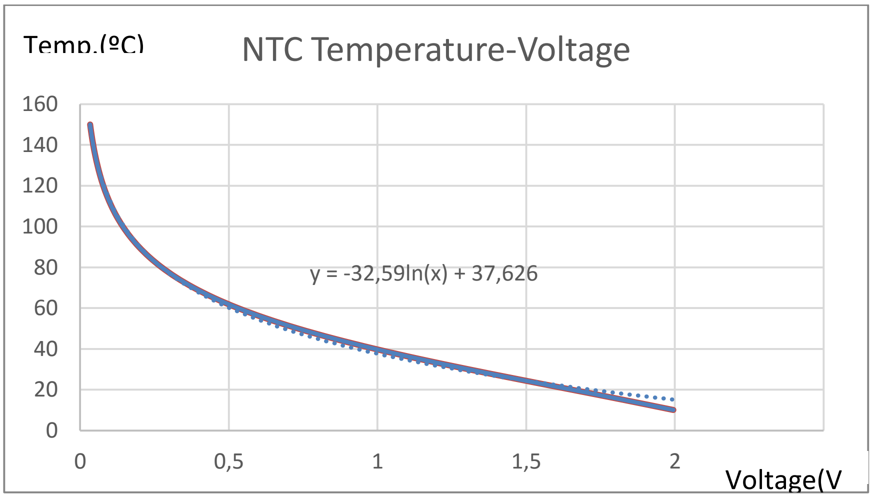


Ilustración 7. Función de transferencia de la NTC

Para las comunicaciones entre los chips y el master utilizaremos el medio isoSPI desarrollado por linear se basa en un canal físico que convierte las 4 señales SPI en una señal diferencial aislada permite la comunicación sin ruido a una distancia de hasta 100m. (Ilustración 8)

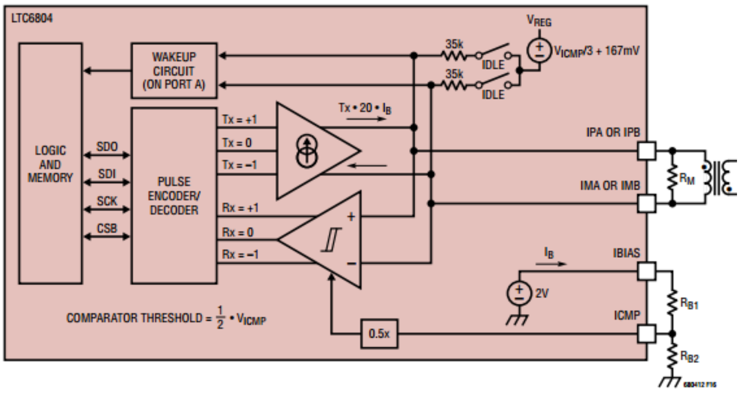


Ilustración 8. IsoSPI

La (Ilustración 9) muestra cómo se realizará la conexión entre los chips. Se requerirá un transformador para asegurar el aislamiento en el caso de comunicación entre chips se utilizará el PE-68386NL del cual se ha comprobado su eficacia en la placa de evaluación. Para la salida de comunicación entre placas un transformador con rechazo de modo común para filtrar posibles EMIs, en este caso el TG110-AEX50N5LF es un transformador con las características óptimas para la aplicación.

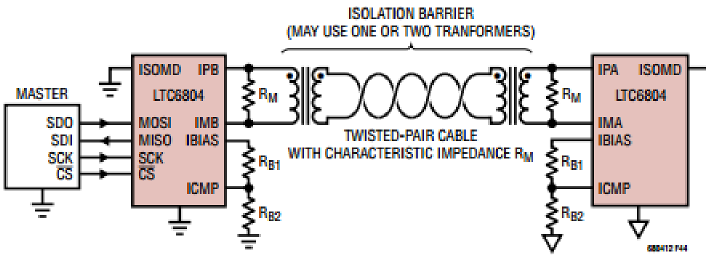


Ilustración 9. Conexión en cascada de dos chips.

La calidad de la comunicación dependerá de la intensidad de BIAS y el ancho del pulso determinará a que distancia nos podemos comunicar. En nuestro caso como sabemos lo delicada que es la aplicación decidimos tener la mejor señal posible, aunque aumente el consumo en este caso fijamos la corriente de BIAS en 1mA obteniendo un ancho de pulso de ±500mV.

Ahora se decidirá la corriente de balanceo. La mayoría de BMS comerciales tiene una corriente de balanceo de 100mA pero para una carga rápida es necesaria una corriente mucho mayor; la recomendada por los expertos del sector es de 1A [33]. Esta corriente debería ser eficaz para cargas de 2-3h. Para obtener esa corriente de balanceo es necesaria una resistencia de 4.2Ω. La serie SMW5 que tiene una resistencia capaz de disipar 5W de 4.3Ω. Teniendo esa corriente será necesario un Mosfet capaz de aguantar más de 1A sin romperse Fairchild tiene modelos capaces de aguantar 8A sin calentarse y capaces de disipar hasta 2.5W.

Para cumplir normativa se posicionarán las resistencias juntas y un disipador común para todas ellas donde se colocará una NTC para medir la temperatura del circuito de balanceo. De no hacer esto se tendrían que tener 130 sensores uno por resistencia de balanceo en vez de 5 uno por substack. El disipador FISCHER ELEKTRONIK ICK 40B será el disipador escogido ya que cumple tanto en dimensiones como en características.

Por último, se debe atender a la norma D.5.4.3 donde nos indica que el cableado del BMS debe ser de grado UL-94V0, FAR25 o equivalente. Se selecciona AWG22 de AlphaWire ya que cumple ese standard y aguantan hasta 8A. Ahora se definirán los conectores en ese caso los serie XAP de JST cumplen las necesidades expuestas. La (Ilustración 10) muestra el tipo de conexionado.

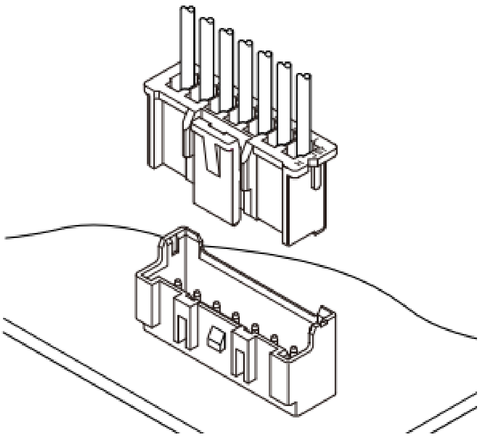


Ilustración 10. JST XA Series

El master se basará en un Arduino Mega, la primera opción era utilizar un arduino uno por su coste inferior pero debido a su inferior capacidad no podía procesar la información de los 15 chips. Se obtendrá un shield con las siguientes características para convertirlo en nuestra Master Board.

- Alimentación de 24V con DC-DC a 5V para el arduino.
- Módulo de comunicación CAN Bus.
- ADC de 18bit para el sensor de corriente HAIS 200-P
- Relé de desconexión de carga de 25A.
- Relé de desconexión de descarga de 10A.

En cuanto al software el siguiente diagrama de flujo (Ilustración 11) indica cual será el método que utilizado para monitorizar, balancear y asegurar la seguridad del sistema.

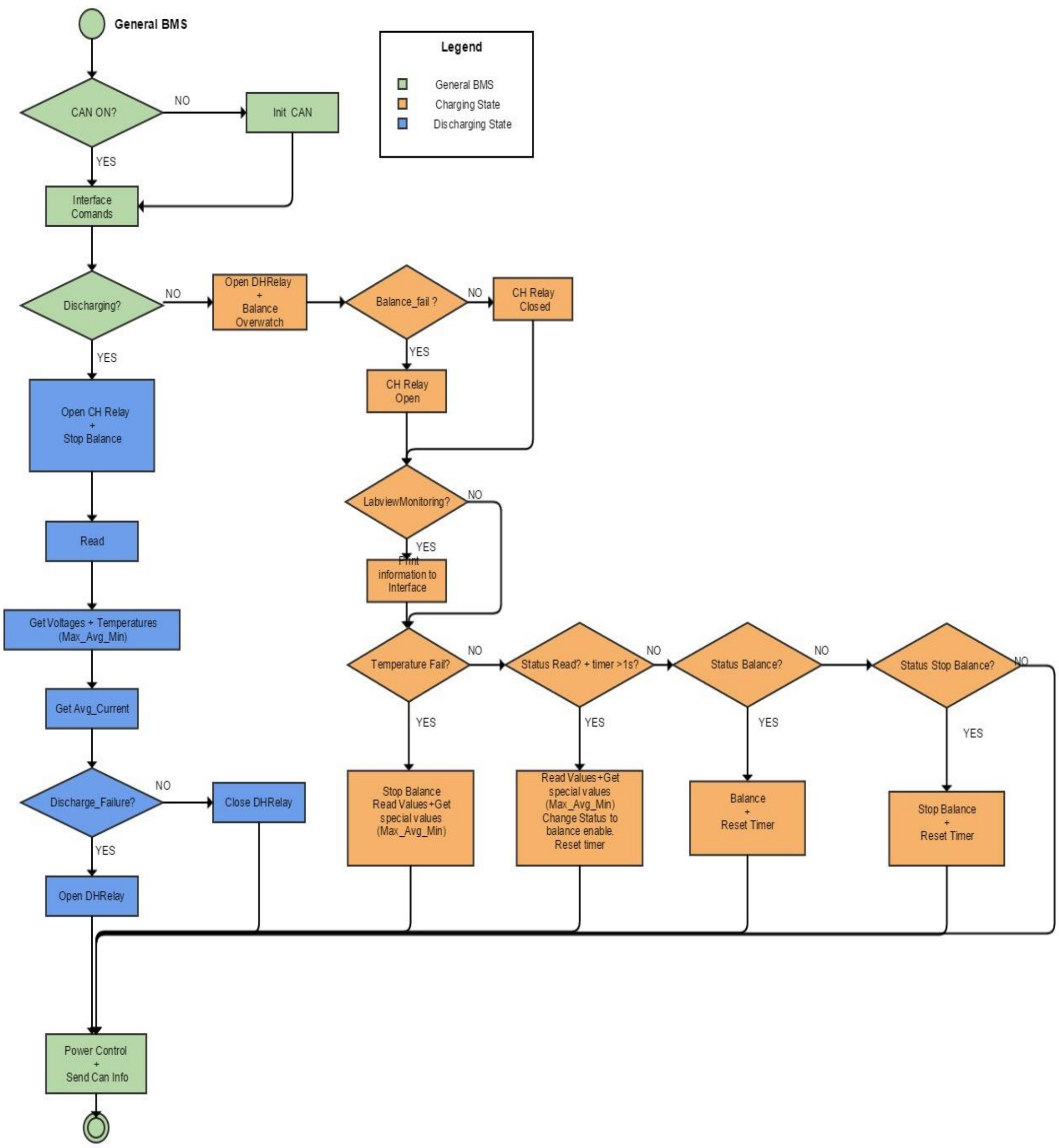


Ilustración 11. UML software BMS

Para terminar, se desarrollará una interfaz intuitiva para cualquier usuario utilizando LabVIEW. Esta interfaz mostrará el estado de todas las celas de cada substack y la temperatura leída por los sensores. También contendrá una pestaña de estado general donde chequear el estado general del sistema. Estado de los relés, el resto de maniobra de seguridad del vehículo, las gráficas indicando corriente y tensión de la batería y un emulador de estado de batería que indicara el estado de carga del sistema. Para esto utilizaremos el módulo VISA que ofrece LabVIEW y permite interpretar las señales que recibimos por el puerto serie. En la (Ilustración 12) se puede ver cómo sería la interfaz.

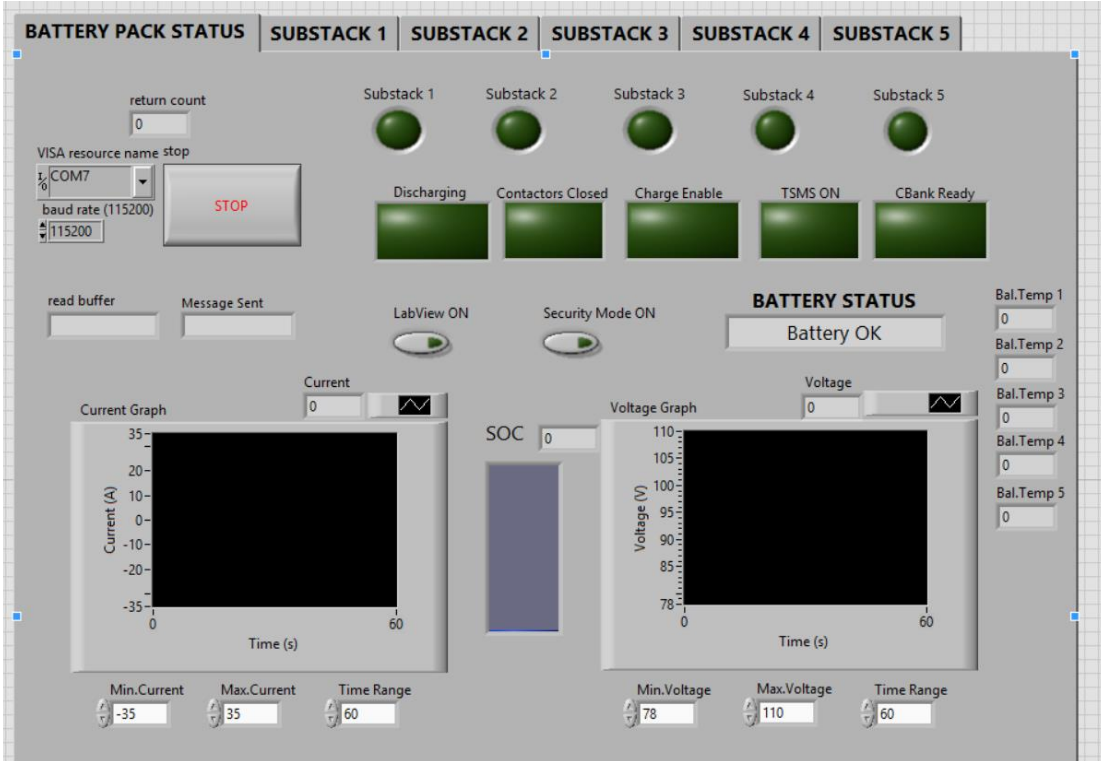


Ilustración 12. Interfaz LabVIEW:Estado General de la batería.

Palabras Clave:

Battery Management System, LiPo Technology, Monitorización de Celdas, Balanceo de Celdas, Seguridad de la Batería.

3. INTRODUCTION

In the second half of the 20th Century our society started to experience some environmental problems, such as high air pollution caused by the increased number of factories burning fossil fuels for their industrial processes and the established concept in society of "one vehicle per person". Not only do we have air pollution as an environmental issue but we also encounter global warming.

People started to become more aware of these issues and many wanted to contribute to a reduction of global emissions, many organizations joined to find a solution and treaties such as the "Kyoto Protocol" in 1992 were signed. But also a new concept was born: "Sustainable mobility".

One of the key points in sustainable mobility is the ability to create vehicles which have a reduced impact in the carbon print. Here is where concepts such as: Carbon neutral fuel, Electrical vehicle (EV), Hybrid Vehicle (HEV) or Green vehicle, are born.

With the need of alternative transportation systems, automobile brands started to research in how they could make a more environmentally friendly vehicle. [2]Toyota is known for having the most advanced technology in hybrid vehicles as it has been producing and selling them since the late 1990s. Also, we know Tesla was born in 2003 and it has had a great impact in the production of EVs as it created the first electric sports car called the Roaster with an autonomy of 320km per charge, but at a high cost.[3]

The birth of lithium-ion and lithium-polymer batteries have made this type of vehicles a reality, as these batteries are smaller, lighter and have a much higher energy density and life cycles than their antecessors, the lead-acid type. But these characteristics come with a drawback, these cells are much more volatile and unstable and need to be balanced and monitored in order to prevent risky situations and extend their life. [4]

The most common problem with this type of cells is the energy storage control as they have a very high capacity and a high discharge rate which implies easy heating which may result in dangerous situations. Also, large battery packs have HIGH VOLTAGE which always carries a risk of electric shock.

As vehicles with an electric power source are beginning to have a place in the automotive market, Battery Management Systems are in constant development. The complexity of the battery pack enhances the risks and so the BMS needs to be more reliable. Nowadays, only cars, buses and motorbikes are being designed but the next evolution surely will go to aviation and the naval industries and both of them require a much higher level of reliability because of the higher potential risks this entails.

The statements mentioned above inspired me to study this product and try to develop a suitable BMS for the competition MotoStudent Electric as this international competition demands a high specifications battery for a competition prototype which results in the inclusion of a reliable BMS to ensure the maximum security of the battery pack.

The regulations of the competition are very strict in battery security matters and so demand some specifications and safety means which can be difficult to find in commercial Battery Management Systems, opening an opportunity for a custom BMS development.

4. State of the Art

In order to be able to make an accurate design with the correct specifications demanded in the EV sector I will carry out a study of the current Battery Management Systems and what has been done in the past. I will divide this research into two sections: Background and Theoretical framework.

4.1. Background

Elithion's Lithiumate pro

Elithion is a BMS developer which has their own products such as Lithiumate pro but also offers customs BMS if you tell them the needed specifications (Figure 1) shows the Master board and control system for the slaves. Lithiumate pro is the most advanced BMS they have, it is a distributed BMS which has a common master for all its product variants and a series of different types of slaves for each battery chemistry.[5] In (Figure 2) we see how the slaves are connected to cells and we can see that we need 1 slave per cell requiring a great amount of cells for large battery packs.

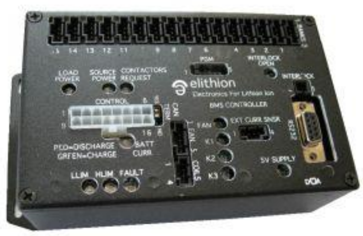


Figure 1. Lithiumate Pro. Taken from Elithium

The features of this BMS are as follows:[6]

- Overcharge/discharge and overcurrent protection.
- Distributed.
- State of charge monitoring.
- Passive Balancing: up to 3A.
- Temperature monitoring.
- Voltage range 1.25V-6.0V
- CAN bus communication.
- Current monitoring.
- Sampling rate approx. 600ms.

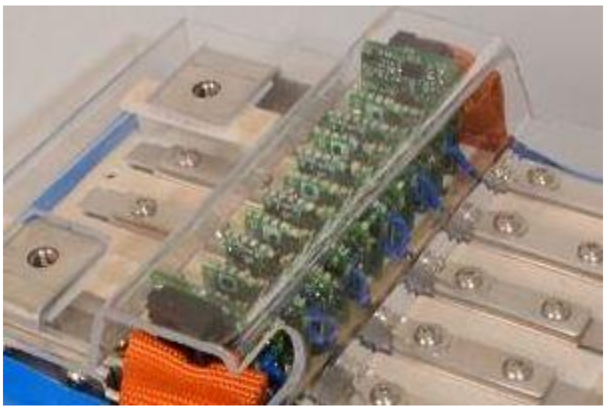


Figure 2. Slave Boards. Taken from Elithium

FlexBMS

Fraunhofer is a German BMS developer which offers a range of products connected with Battery monitoring and management. FlexBMS (see Figure 3) is an active balancing distributed BMS, its name comes from the flexibility that this type of BMS offers. In the central part of the image we can observe the intelligent chip used for battery management. This BMS connects each board in a daisy chain with an isolated serial 2 wire connection.[7]

Figure 3. Cell Management (Slave) & Battery Management Controller (Master). Taken from Fraunhofer

The features of this BMS are as follows:

- Overcharge/discharge and overcurrent protection.
- Distributed.
- Active balancing around 5A with a 78-91% efficiency.
- Cell voltage resolution 1mV.
- Temperature Sensing.
- Up to 800V system.
- Current Sensing.
- CAN bus interface.

This BMS is very similar to the one mentioned before but it relies on active balancing instead of passive balancing. Also, it is not fully distributed as all the Cell monitors go to a Module Management Controller and then the MMC sends the preprocessed information of a maximum of 16 cells to the BMC (Master).

Orion BMS

Ewert Energy Systems is an American company that develops solutions for plug-in hybrid and electric vehicles, energy sources and energy storage applications.[8]

One of their energy storage applications is the Orion BMS as seen in (Figure 4). It is a Passive Balancing Centralized Battery Management System, which requires a great amount of wiring. Orion offers a pre-wired Hardness to make the connection with the battery pack easier.[9] Its box is covered with a thermal heat sink to enhance passive balancing and avoid temperature problems.

Figure 4. Orion BMS Standard. Taken from Ewert Energy Systems

The characteristics of this BMS are as follows: [10] [11]

- Overcharge/discharge and overcurrent protection.
- Centralized.
- Passive Balancing around 0.1A.
- Maximum 108s Cells.
- 4 Thermistors.
- Voltage range 0.5V-5.0V
- Sampling Rate 30ms.
- Current Sensing.
- Dual CAN bus interface.

This model has a limitation of 108 due to the great amount of cables needed.

Because of its topology and distribution, these models are popular for small battery packs as it reduces the amount of electronics inside the actual pack.

MiniBMS V3

CleanPowerAuto LLC has created a simple BMS for $LiFePo_4$ cells. These cells are the most "stable" cells of the lithium family as they have lower discharge rates and longer life-cycles. These cells are very popular for small EV such as delivery motorcycles.

It is a distributed BMS called MiniBMS as seen in (Figure 5) which interconnects its boards with what it calls "OptoLoop" and opto-isolated interface they have created to simplify installation. It uses a passive balancing method.[12] [13]

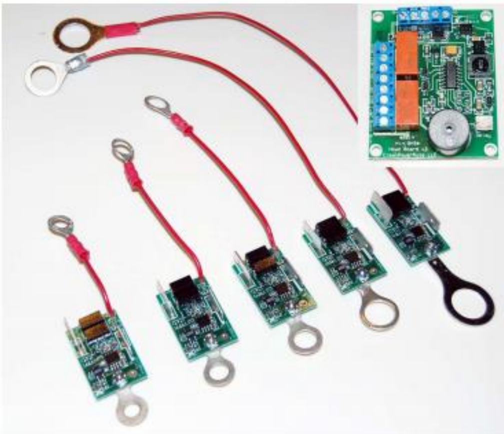


Figure 5. MiniBMS V3. Taken from CleanPowerAuto LLC

The features of this BMS are as follows:

- Support for $LiFePo_4$ cells, various cable lengths available to support a wide range of applications.
- Distributed.
- Passive balancing 0.75A.
- Resettable fuse protecting to protect from short circuit conditions, it shuts when current reached 1A.
- Over-Voltage and Cutoff-Voltage protection.
- No current sensing.
- Unlimited number of slave boards.

FreeSafe BMS

Freemens is a French startup born in 2014 in the University of Grenoble where they developed their first BMS used to control and monitor a 48V battery pack later on they started to grow and develop other boards for further voltage applications.

FreeSafe Extended is their new board for 2016 and can to manage from 6 to 24 cells. The board is designed for $LiFePO_4$ cells but can be customized for any type of cell up to 5V charge.[14]

In (Figure 6) we can see a FreeSafe XT slave board on the right-hand side of the board we can observe the 2 Battery Management chips made by Linear Technology. In this case LTC6803-1 as named in the datasheet. In the central part, we can observe the power resistors with which cells are balanced. Also we see the CAN communication connectors both Private Freemens CAN and a normal isolated CAN.

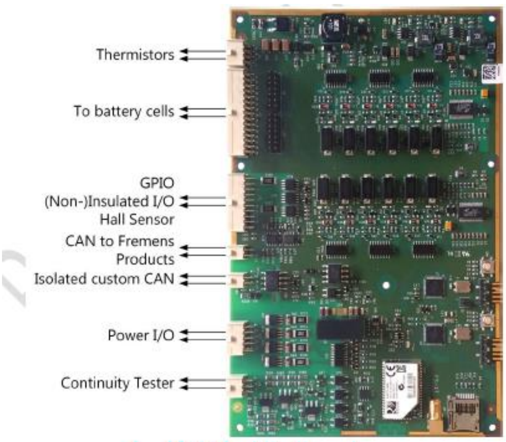


Figure 6. FreeSafe Slave Board. Taken from Freemens FreeSafe XT

The features of this BMS are as follows:

- Support for $LiFePo_4$ cells and adaptable to any type of cell up to 5V.
- Modular.
- Passive balancing 0.5-0.75A.
- Current Sensing up to 150A
- 2 thermistor prepared connectors with a 10K resistor conditioning circuit.
- OV and UV protection.
- CAN communication with Master.

Autor: **Xavier Anzuela Recasens**
424.16.96

4.2. THEORETICAL FRAMEWORK

4.2.1. *Battery.*

A battery pack is made of groups of serial or parallel cells in order to obtain the sufficient energy for their sourcing purpose. There are many parameters to take into account when selecting the type of battery for each application, all of them are equally important as they define the final characteristics of the pack. Each type of battery chemistry defines the characteristics.

4.2.1.1. Battery Types.

Over time many different types of battery technology have been developed and in general all technologies are valid but each one is used for a certain objective. However, the lithium technology is becoming the most popular as it is used in almost all electronic devices such as cell phones, laptops and it is the most popular in EV and HEVs. Despite the lead-Acid dominance in combustion vehicle batteries.

- Lead-Acid Technology: Low cost and high power but they have low energy density, they are very heavy and work in a very short temperature range. This type is very popular for internal combustion vehicles.

- Nickel Technology: Higher cost than Lead-Acid but their weight is lower, they also have higher energy density Ni-Cd is a known battery chemistry for this technology but they are forbidden for domestic purposes as Cd is a high pollutant. Their alternative is Ni-MH which is popular in Hybrid vehicle technology.[15]

- Lithium Technology: Lithium is a very light metal and has amazing electrical properties. Currently, it is the material which allows higher voltage cells up to 4.2 compared to the 3V which Nickel gives and 2.4V Lead-Acid technology offers. This technology is the chosen one by pioneer electrical vehicle companies such as Tesla.[3] And it has been used for years in the portable electronics sector. As they give high energy density, are light and their price is lowering as they start being mass produced.

- Graphene Technology: This technology is still in a research state even though a Spanish company is known to be developing a prototype graphene or lithium-graphene cell which may be an inflexion point for battery packs if they succeed. Graphene is known for having exceptional thermal and electrical properties and for its flexibility giving birth to all sorts of flexible electronics or flexible cells. Flexible and high energy density cells would be a revolution for the automotive electrical sector and all sector which need small and powerful power sources.[16] [17]

4.2.1.2. Nominal Voltage.

The nominal voltage of a cell is the reference value of voltage in which the cell will work most of its life cycle, in the case of lithium-polymer cells their nominal voltage is 3.7V.

4.2.1.3. Charge Condition.

Batteries have different parameters depending on whether they are in charging state or in discharge. Usually the charging condition has more restrictive parameters as the cells are not actually being forced beyond their usual working condition so maximum temperature of cells and the amount of current are usually lower than in discharge mode.

4.2.1.3.1. Max. Continuous Charge Current.

The maximum continuous charge current is a parameter specified by the manufacturer of the cells which sets the charging current. Usually battery packs are charged at this current rate as it provides the fastest charging rate. Most batteries cannot be charged the full time with the same current, many manufacturers suggest a charging protocol such as starting at a lower rate charge and then, continue with a faster charge.

4.2.1.3.2. Peak Charge Current.

The peak charge current is the maximum current the manufacturer suggest to apply for a short period of time to enhance the charging rate. Most cells can only stand this peak for singular periods of 1-3 seconds, it appears as a charging short boost. Manufacturers do not recommend charging beyond the continuous charge current but in some cases it may be needed when time is an issue.

4.2.1.3.3. Charge Voltage.

Battery packs are charged using their chemistry battery charging curve. Usually two types of curves exist a simple charging curve for normal current charging and a high efficiency curve and more complex to parametrize which adapts to each cell manufacturer, this last curve is used for quick charging

4.2.1.4. Discharge Condition.

In discharge the considerations to be taken into account change as the temperature levels that the cells can reach usually are less restrictive than during charge state. During this condition we have to take into account a balance current delivery from all the parallels of the battery pack, as a non-parallelized delivery may cause a quicker discharge of some parts of the battery pack which then will be assumed by the lower rate discharging cells making the pack unstable and reducing its efficient life.

4.2.1.4.1. Continuous Discharge Rate.

The continuous discharge rate is the nominal current that the battery can offer without suffering temperature issues and without reducing its life cycles. For instance discharging further current from the pack continuously may consume various life cycles in one single discharge.

4.2.1.4.2. Peak Discharge Rate.

It is the maximum discharge rate admitted by the cells without risking their integrity. Usually the manufacturer specifies which the peak current is and how much time you can discharge at that rate. Discharging at peak current implies temperature rising within the cells of the pack and further discharging from the time listed by the manufacture may damage the cells or even cause an accident. For this reason current measuring with the BMS is a must as the BMS can cut-off the connection between the battery and the power consumption source.

4.2.1.4.3. Cut-off Voltage.

The cut-off voltage is defined as the under-limit voltage from which the cells may lose capacity if they go below that point. In lithium-polymer cells the cut-off voltage is usually 3V.

4.2.1.4.4. Overcurrent.

Overcurrent is known as the event in which the cells are discharging at a higher rate that the one they are able to. This causes a higher degradation of the cells and is usually associated with cell temperature rising and high voltage depression. The usual protection for overcurrent is a fuse but the BMS may open contactors if overcurrent is detected to protect the cells.[18]

4.2.1.5. SOH

The State of Health (SOH) of a battery is an abstract concept which tries to simplify in one statement a conjunction of phenomena that produce the battery's degradation. This resumes in which point of its life cycle a pack is. The end of its life cycle varies depending on the application so as a general statement we consider a battery as "dead" once it can no longer provide the minimum power energy to feed the application it was powering.

Three symptoms of cell degradation are usually used to estimate SOH as they are easily detected.

- Capacity fade: We describe capacity fade as, "The reduction in the available energy and charge capacity of the battery over time."[3] This phenomena is a side effect of chemical degradation. Either the electrons or the lithium polymer are not able to reach the active material. This is usually caused by a damaged or deteriorated electrode.
- Impedance growth: Impedance growth is directly related to cell capability reduction. This is usually caused by the loss of active material in the surface area of reaction. Not only does this phenomenon affect cell capability but a single cell impedance growth may destabilize the full battery pack creating a butterfly effect where all cells have an accelerated degradation compensating the lower work rate damaged cells are experiencing.
- Increase self-discharge: A common feature of all battery chemistries is that the older they are, the quicker they discharge. A good BMS can compensate this problem by keeping cells balanced avoiding high current transfer between cells when they auto balance between them. An active balancing BMS can almost fully compensate this phenomenon and really extend battery life.

4.2.1.6. SOC.

The state of charge is an abstract concept which tries to resemble the amount of "fuel" that a battery pack has before it is completely discharged. It is a way in which we try to express how much more battery we do have. Nevertheless, lithium ion batteries are one of the most difficult technologies to determine the SOC in an accurate way as the high discharge rates they have do not help producing a liner curve. As shown in (Figure 7) their discharge curves are far from linear and have a middle state where it is very difficult to define if the battery is at 40% or 80% charge as we can see in the image below.[4]

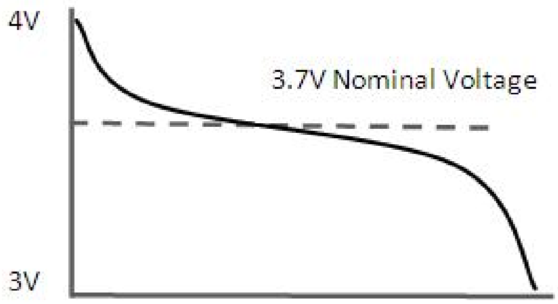


Figure 7. Lithium-Ion Discharge Curve. Taken from NI

Also we have to notice that we are speaking about SOC regarding a fully balance battery but when looking at an unbalanced pack, the cell with the minimum voltage usually states the battery voltage as if all the cells are equal they will discharge at the same rate so that lower voltage cell will delimitate the battery by regarding the battery as discharge once this cell hits the cut-off voltage. This phenomenon defines one of the most important reasons for having an efficient balancing system.

There are many ways of calculating the SOC. The easiest way is to make a linear relationship between total voltage and voltage left but as the curve is not linear it is very inaccurate. Another method is by characterizing the battery pack and extracting a transfer function with voltage as the variable; being this one of the most common methods to calculate SOC. Although there is a third method which is used to calculate SOC, it really calculates what is known as SOE, State of Energy which determines how many Watts-hour we have left from the fully charged state. This last method is the most accurate but also requires a high accuracy in both current and voltage measuring to achieve a highly reliable power consumption curve.

4.2.2. Battery Management Systems

A Battery Management System is a need when using a Lithium based battery pack, this system is in charge of the security of the battery pack and it can extend its life up to a 30% in the best cases. With the appearance of the lithium based battery cells and their instability the need for these systems increased. Lithium-ion cells or lithium-polymer cells have great charge and discharge rates which make them optimal for high-sustain systems like vehicles but these advantages come with a price. These cells easily gain heat in both charge and discharge situations and they are highly volatile when they heat up, the risk of explosion exists. The BMS monitors the state of the cells both in voltage and temperature (the specific characteristics depend on each model), and also may monitor current, once monitored an intelligent module (Master/Microcontroller) uses the obtained data to balance unbalanced cells to extend battery life or disconnects the power to avoid any risky situations in which users may be harmed.[4] (Figure 9) shows a simple block diagram of what a Battery Management System is meant to do.

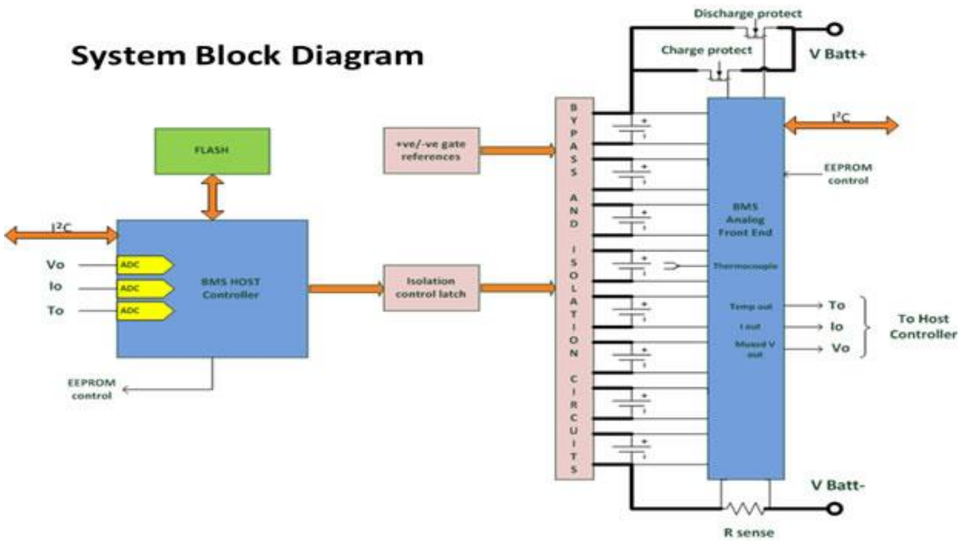


Figure 8. System Block Diagram for a typical BMS

4.2.2.1. BMS Topologies.

Battery Management Systems have 2 main topologies; the most typical topology is the Master-Slave where there is an intelligent board which controls other boards known as Slaves. These slaves do not have a microcontroller which analyses the data obtained by them, instead the "Master Board" receives all the data from the "Slaves" and orders the slave boards what to do with them. The other topology is based on a fully intelligent board which both gathers and analyses the data obtained from the battery.[19]

4.2.2.1.1. Distributed.

Master/Slave where each cell has a board monitoring it and an intelligent board gets the information of all cell boards (Slaves) and makes the decisions. Usually distributed BMS Slaves monitor one cell per slave but monitoring 3-4 cells still defines the distributed BMS as distributed. This is the most expensive architecture as you almost need one separated board per cell or small group of cells and usually takes more space in the battery pack than a Modular as you need a certain surface to ensure isolation.[20]

In (Figure 9) we can see an example of a distributed topology, how boards are set and how many cells each board takes cares of. As seen in (Figure 2) distributed BMS is good for small battery pack. As a board is required for each cell in a pack with a high amount of cells this BMS may be a problem in terms of space.

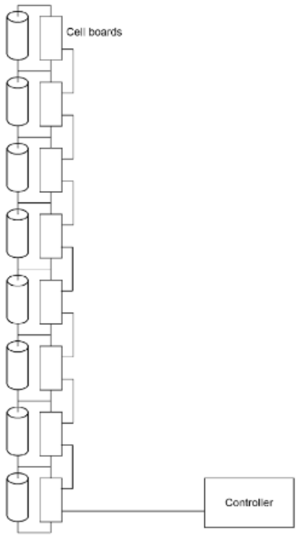


Figure 9. Distributed BMS Topology

4.2.2.1.2. Modular.

It is a hybrid between the mentioned above. It uses "Slave" boards which monitor more than one cell and all the information is sent to the "Master" so that it makes the decisions and tells the Slaves how to act and when to balance. As a hybrid topology there are many ways of implementing it either having a series of boards interconnected in a daisy chain and then report to the master (Figure 10) or as seen in (Figure 11) independent intelligent boards with a microcontroller which send independently information to the master which controls the security system but each individual micro controls monitoring and balancing of the cells. The former is more similar to distributed BMS and the latter more similar to Centralized topology.[4]

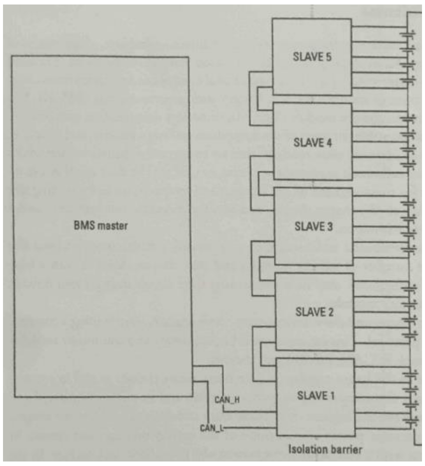


Figure 10. Modular BMS Topology 1

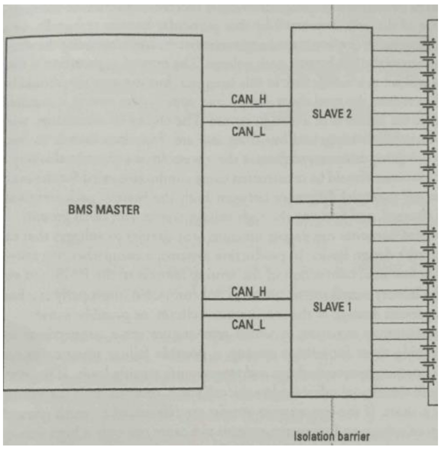


Figure 11. Modular BMS Topology 2

4.2.2.1.3. Centralized.

Also called monolithic systems, they are a type of architecture where a single board has the full circuit installed. This system is usually the cheapest as the full design is thought for a certain battery but the problem is the high density of cables when there is a high density of cells and the high power that the board is connected to in case of short circuit condition. Also many times this type of BMS is only usable for the application is designed for and remodeling the battery is not possible as the BMS may stop working.[20] In (Figure 12) we can observe a centralized BMS with passive balancing as we can see the balancing resistors at the bottom. As it has been mentioned before this type of BMS is good for small packs but we already see the high amount of cables needed for a 16-cell battery, so imagine using a centralized BMS for a 100 cell battery.

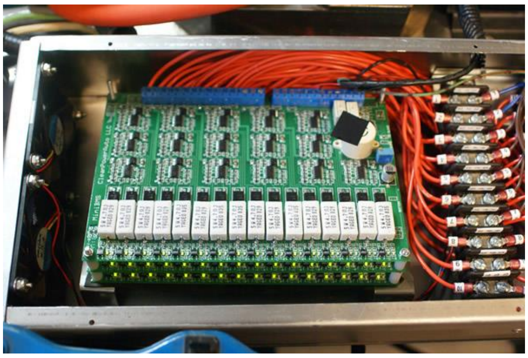


Figure 12. Centralized BMS. from Elcon

Table 3. BMS Topology synthesis

BMS Type	Cost	Software	Capacity	Adaptability
Distributed	Highest	Intermediate (depending on amount of slaves)	Flexible to all battery packs.	highest
Modular	Intermediate	Intermediate (depends on complexity of slaves)	Flexible to all battery packs.	high
Centralized	Lowest	Simple	Difficult connection in high density packs.	Specific design for each battery

4.2.2.2. BMS types.

Not only is the topology an important point to take into account when choosing or developing a BMS but also the purpose. If the application does not require energy regeneration passive balancing serves the purpose but when we enter more complex applications where efficiency is a must we are talking about active balancing.

4.2.2.2.1. Passive Balancing.

Passive balancing is the most common balancing method used in not high efficiency applications when quick charging and low energy consumption are a must. But it is not the most efficient method as we consume extra energy instead of redistributing it to lower charged cells. The benefits of this balancing technique are the easy setup of the system and a lower risk of cells being constantly charged and discharged because of a malfunction of a higher complexity software and hardware as the active balancing setup requires.[4] In (Figure 13) we can observe the simplicity of a passive balancing circuit, a Mosfet which, when activated, sets a resistor parallel to the cell and discharges it until the Mosfet finishes commutation. This is a typical passive balancing circuit where an internal Mosfet of the chip commutes when balancing is needed setting a small I_{bias} current that with R_2 sets a gate-to-source voltage which stimulates the external Mosfet gate commuting it and setting R_{Bal} in parallel which the cell which starts discharging the cell. V_{cell}/R_{Bal} sets the balancing current. When we have adjacent cells (Figure 14) this circuit may be a problem if two consecutive cells need balancing as VGS does not exist because if both internal Mosfet are set at the same time no current flow exists as adjacent I_{bias} current compensate each other. This phenomenon as “adjacent cell-balancing issue” [21]

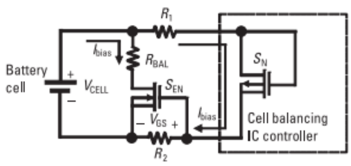


Figure 13. Passive Balancing Circuit for one cell.

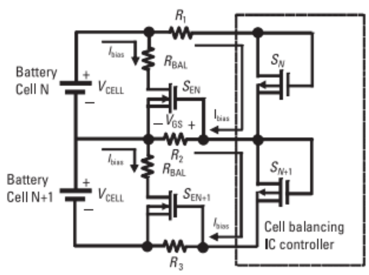


Figure 14. Adjacent cell-balancing

Some chips allow a different passive balancing circuit which is based on working directly with the Mosfet’s Gate with a digital signal. For instance we have the typical application of a Linear Technology Cell Balancing chip (Figure 15).

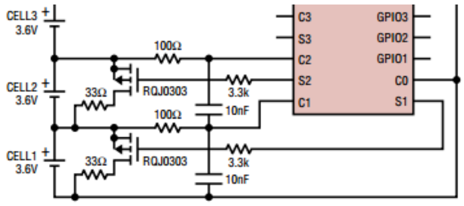


Figure 15. LTC6804-1 Passive balancing

In this case the internal Mosfet of the chip commutes and sends a 0 signal to the Channel-P Mosfet commuting it and setting the resistor in parallel as it works as a digital signal. No I_{bias} current interacts between cells avoiding the adjacent cell phenomena. The 3k3 resistor is used as a protection for the Mosfet as the normal state of the gate is a 5V output with which the resistor sets a very low current reception for the Mosfet.

4.2.2.2.2. Active Balancing.

Active balancing is a more complex way of cell control as it requires intelligent energy redistribution to avoid losing energy, active balancing is very popular in applications where every milliwatt-hour is precious.[21] Active balancing circuits use capacitive or inductive components to store extra charge and redirected it to the lower charged cells. These circuits have a high complexity and have various variants more or less complex but also more or less efficient.

(Figure 16) Shows a bidirectional buck-boost converter based active balancing circuit which can transfer energy from either direction, the problem with this circuit is that if a middle cell is extra charged and the lowest charge cell is the first or last cell the energy has to go through all cells until it reaches its destination cell. When the Sn signal is on, current starts flowing and charging the inductor, the diodes are used to avoid energy return when Sn signal reverts.[21] On the other hand we have the flyback converter. (Figure 17) shows a flyback converter where the energy of the bottom cell is transferred to the top cell when S1 is ON by feeding the primary of the transformer and storing the energy in the magnetic field once the S1 is turned OFF the energy stored in the magnetic field flows though the secondary up to the top cell the diode D is used to ensure the energy flows in one direction.[4]

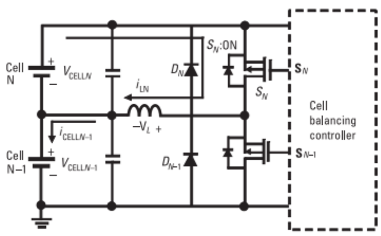


Figure 16. Buck-Boost converter. Active Balancing

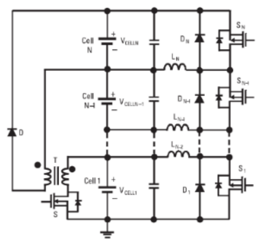


Figure 17. Flyback. Active Balancing

4.2.3. Temperature Sensing.

We have two types of temperature sensors: Analog sensors and digital sensors, digital sensors are more accurate but also more expensive than analog sensors so depending on the precision of the application an analog sensor will make the job or a digital sensor may be applied.

4.2.3.1. Thermistor

A thermistor is a variable resistance resistor whose resistance varies with temperature. There are two types of thermistors PTC and NTC:

- PTC: Positive Temperature Coefficient, the resistance of the resistor increases when temperature rises.
- NTC: Negative Temperature Coefficient, the resistance of the resistor decreases when temperature rises.

Both are analog temperature sensors and usually have a precision of 1ºC except for precision thermistor which have a precision of 0.05ºC but any error calculating their transfer function may affect that precision.

- Disc thermistor: Usually used for temperature measurement when not a high precision but a quick measurement is required. Easy connection and insertion to a circuit with their metallic legs, easily through-hole component or cables may be soldered to them to have a wired sensor. (Figure 18) Is an image of what Leadless Disc thermistors look like.

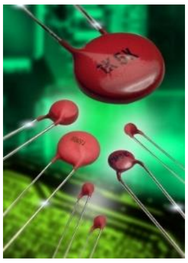


Figure 18. Disc thermistor. Obtained from AVX.

- Leadless Disc thermistor: Offer thermal and electrical stability and are usually used for some automotive applications, these are offered in many sizes and temperature ranges to adapt to each application easily. (Figure 19) Is an image of what Leadless Disc thermistors look like.

Figure 19. Leadless Disc Thermistor. Obtained from AVX.

- Accurate Leaded & Leadless thermistor: They are a more accurate version of the disc thermistor but also much smaller and designed for accurate and quick response measuring. They are widely used in industrial and automotive applications. (Figure 20) Is an image of what an accurate thermistor is and how small they look compared to the ones shown before.

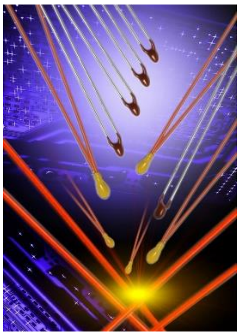


Figure 20 Accurate Thermistor. Obtained from AVX

4.2.4. Current Sensing.

A current sensor is a device that is used to detect electric current. It can either detect AC or DC in the wire that goes through it. Depending on the type of sensor it returns a proportional signal to the read current. This signal can be current or voltage.

Current sensors can be either open or closed-loop both measure AC and DC currents but they have their peculiarities.

Open loop sensors: The main problem with this type of sensors is they have a poor bandwidth and response time compared to their counterparts. On the other hand, their power consumption does not vary when more current is being read so they are great for measuring currents over 100A at an economical price although they are very bad sensors when there is a high temperature gradient occurring on the application.

In (figure 21) we see the working principle of these sensors. The conductor though which current is passing is surrounded by a magnetic core with a small air gap where a hall element is placed to create the hall effect and sense the magnetic flux. With the help of an amplifier we are able to read the small response the hall effect offers and therefore with a transfer function know the amount of current going through the conductor. The problem is that any temperature drift may disturb the hall effect and make an inaccurate reading.

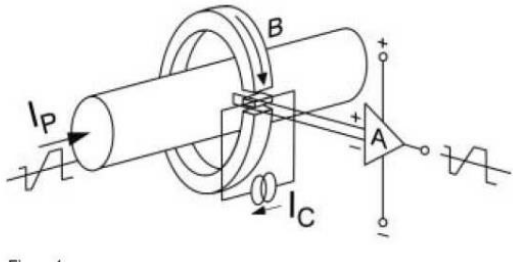


Figure 21.Open loop Current Sensor. Operation principle. Obtained from LEM

Closed-Loop sensors: They provide great electrical isolation. They offer a very fast response, high linearity and low temperature drift and the output is almost immune to electrical noise making this the perfect choice when needing high accuracy and fast readings. Their power consumption varies as current measured gets higher.[22] In (Figure 22) we can see how these sensors have a booster circuit which creates a compensating current in the secondary to create a magnetic flux equal in amplitude to the primary current magnetic flux but in opposite direction. This is used to eliminate the drift of gain with respect to temperature by creating a zero flux condition.[23]

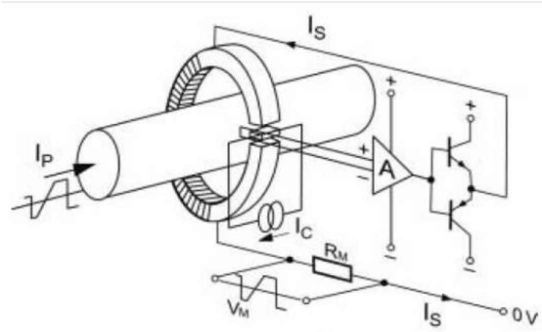


Figure 22. Closed loop Current Sensor. Operation principle. Obtained from LEM

4.2.5. SPI Communication.

Serial to Peripheral Interface is a communications protocol developed by Motorola between the eighties and the nineties to create a standard protocol of communication in embedded systems. However, this protocol is only for this purpose as it can only be used for short distance communication. It is a bidirectional synchronous serial communication interface where a master sends a signal to a slave and the slave returns a response to the master.

As shown in (figure 23) it is a 4-wire serial bus where we can distinguish two pairs of communication and control wires. SCLK(Clock) & SS (Slave Select) are the control pair. MOSI (Master-Output Slave-Input) & MISO (Master-Input Slave-Output) are the communication pair.

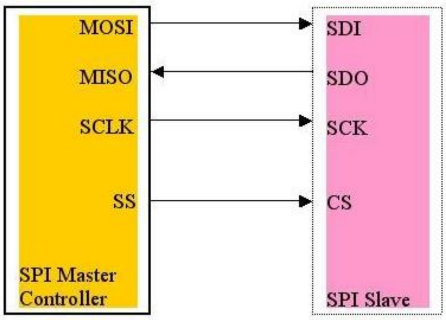


Figure 23. Simple SPI representation. Obtained from Embedded Systems Course of EE Herald.

The Clock sends a synchronous signal with synchronizes all slaves with the master and when the master wants to send or receive information from any of the slaves it selects with the corresponding SS the slave and starts the bidirectional communication with it. (Figure 24) Shows an example of communication, the master sends information through the MOSI and receives information from the slave through the MISO and the slave receives through the MOSI from the master and sends the information through MISO.

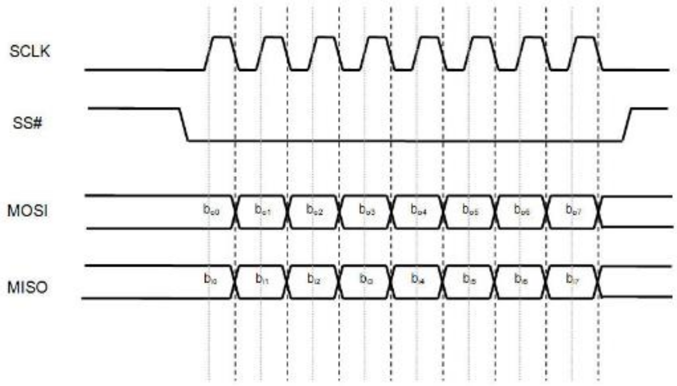


Figure 24. SPI communication pulses. Obtained from byte paradigm.

The main problem this protocol has is that it is limited in distance because of its tendency to capture electrical noise.

Linear Technology, in the search to solve this problem, created the isoSPI protocol. IsoSPI is an isolated communications protocol where the physical interface is single twisted-pair connection. Where we have IM(Isolated Interface Minus input/output) & IP(Isolated Interface plus input/output). To be able to use IsoSPI linear has created LTC6820 a chip that converts SPI to IsoSPI and vice-versa. (Figure 25) shows how we interconnect two devices using these isolated communication protocols. A very important issue with this communication channel is that we have to be sure that both cables have exactly the same impedance or the communication may fail or experience errors.[24]

Features:

-1Mbps isolated data communications.

-Simple Galvanic Isolation using standard transformers.

-Low EMI and noise susceptibility.

-Extend normal SPI distance up to 100m and ensures 10m 1Mbps communication speeds. As seen in (Figure 26) extracted from the LTC6820 datasheet.

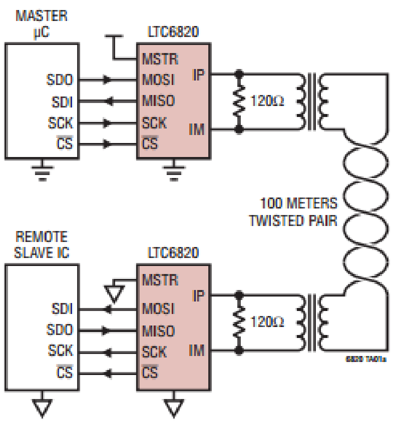


Figure 255. Two Devices Connected via IsoSPI.

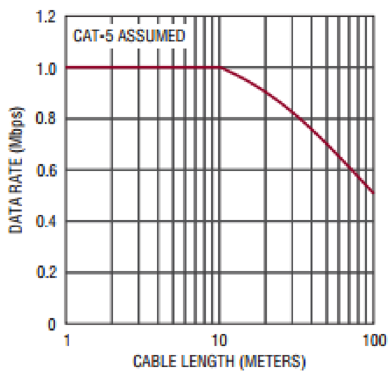


Figure 266. Speed vs Cable length

4.2.6. Can Bus Communication.

CAN Bus (Controller Area Network Bus) was developed in the mid 1980 by Bosch GmbH to provide a low cost communication for automotive applications. Nowadays it is also used in industrial factories, robotics, medical devices and aeronautic industry.

This bus can operate at speeds between 20kb/s to 1Mb/s the standardized speed by ISO/DIS 11898 for high speed application is 500kbit/s and 125Kbit/s for slow applications. Either way the speed chosen also depends on the bus length and the transceiver speed and sometimes it is not possible to adapt to the standards. [25] (Figure 27) shows a relationship between bus length and maximum speed limits.

Bit rate	bit time	bus length
1 Mb/s	1 μs	25m
800 kb/s	1.25 μs	50m
500 kb/s	2 μs	100m
250 kb/s	4 μs	250m
125 kb/s	8 μs	500m
62.5 kb/s	16 μs	1000m
20 kb/s	50 μs	2500m
10 kb/s	100 μs	5000m

Figure 27. Speed/Length relationship.[25]

There are two CAN protocols the Standard CAN and the Extended CAN. The extended CAN uses a 29 bit identifier instead of the normal 11-bit identifier and reaches up to 537 million identifiers besides the only 2048 different message identifiers.[26]

The Standard CAN structure is shown in (Figure 26):

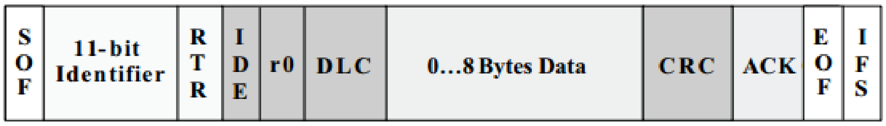


Figure 28. CAN 11-bit identifier

The Extended CAN structure is shown in (Figure 27):

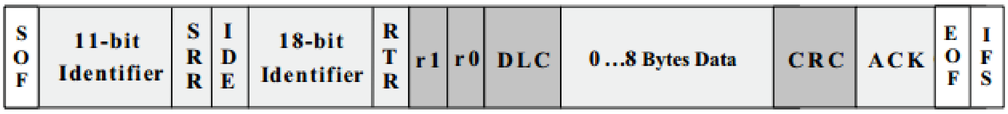


Figure 29. CAN 29-Bit Identifier

- SOF: This bit marks the start of a message a synchronizes the nodes on the bus after being idle.[26]
- 11-bit Identifier: States the priority of the message it contains, the lower the bit value the higher the priority it has in the bus.
- RTR- This bit is dominant when information is required from another node.
- IDE Standard- Indicates that a message with no identifier is sent.
- IDE Extended- Indicates more bits follow in this case the 18-bit extra identifier.
- R0- Has no use at the moment but it is reserved for future needs.
- DLC- Contains the number of bytes being transmitted.
- Date- 8 bytes of data may be transmitted per message.
- CRC- Checks the sum of bytes transmitted as an error control.
- ACK- When data arrives to a node that node checks this bit as saying no errors received. It acts as a checkpoint in each node.

- EOF- It marks the end of message.
- IFS- Contains the time required for a controller to correctly decipher the message is being sent.
- SRR- Replaces RTR from Standard CAN.
- R1- It is another reserve bit as r0.

The CAN physical layer is based in a pair of twisted cables CAN-H (CAN High) & CAN-L (Can Low). Usually the hose wiring for CAN has 4 cables the communication cables and the standard communication cables of 24V and GND.[25]

To avoid electrical signal reflection at the end of the node when a message is being sent, a resistor is placed between CAN-H and CAN-L at both terminations or in other words at each connection to a device. The standard is to use a 120Ω resistor as most applications do not have a connection of more than 40m but in theory that resistor must be calculated depending on the distance the physical layer reaches and the impedance of the cable used to transmit. (Figure 30) gives an example of which terminator resistor and cable type for each case.

Bus speed	Cable type	Cable resistance/m	Terminator	Bus Length
50 kb/s at 1000 m	0.75 ...0.8 mm^2 (AWG18)	70 mΩ	150 ... 300 Ω	600 .. 1000 m
100 kb/s at 500 m	0.5 ... 0.6 mm^2 (AWG20)	$<$ 60 mΩ	150 ... 300 Ω	300 ... 600 m
500 kb/s at 100 m	0.34 ...0.6 mm^2 (AWG22, AWG20)	$<$ 40 mΩ	127 Ω	40 ... 300 m
1000 kb/s at 40 m	0.25 ...0.34 mm^2 (AWG23, AWG22)	$<$ 26 mΩ	124 Ω	0 ... 40 m

Figure 30. Bus Cable and Termination Resistor Characteristics.[25]

5. DEVELOPMENT

The development of this Custom Battery Management System will be defined by two main things. The battery pack being developed for and the regulations of MotoStudent which as mentioned before, are very restrictive as they are designed to ensure full security during the competition.

To start with, we will take an inner look into the competition regulations followed by the definition of the battery pack. After the limitations of the system have been defined we will proceed to the electronic and PCB design and development, the chosen software to do so is KiCAD and Open Source software for PCB development similar to Altium.

Following the design comes PCB assembly and software development to start testing the reliability of the system. Also an interface will be needed to ensure an easier testing and also a way to check the battery status during tests and competition. LabVIEW will be the chosen software to develop this interface.

5.1. MOTOSTUDENT REGULATIONS.

D.3.5 Battery Management System – BMS

D.3.5.1 The installation of a battery management system is compulsory.

D.3.5.2 The BMS must read the voltage of each cell in order to keep the cells within the voltage limits established by the manufacturer.

D.3.5.3 In the case of a centralized BMS, each communication conductor between battery terminals and the printed circuit or the BMS must include a fuse to protect he conductor against possible shortcuts.

D.3.5.4 For distributed BMS the protection fuse may form part of the printed circuit or, otherwise, be fitted to the conductor.

D.3.5.5 The BMS must at least read the temperature of the cells in their hottest point by means of a compatible temperature sensor. It is compulsory to read the temperature of at least 30% of the cells. It is recommended that the system can indicate at least the temperature of the hottest cell near the maximum programmed temperature, before the heat shield disarms the propulsion system of the bike.

Autor: **Xavier Anzuela Recasens**
424.16.96

D.3.5.6 For distributed BMS, if the board has a temperature sensor it may be considered as the cell temperature if it is located reasonably near the distance from the hottest point of the cell.

D.3.5.7 Should a cell balancing system be used, there must also be a temperature sensor next to each dissipation resistance, to make sure that during the balancing period, the temperature indicated by the manufacturer of the resistance or the BMS is never exceeded. If all the resistances are unified by a common thermal dissipater it may be considered as one whole dissipation system and only one common sensor will be needed.

D.3.5.8 To improve the balancing speed, it is allowed to activate the artificial cooling of the battery container during the balancing process.

D.3.5.9 The use of the same temperature sensor for the balancing control function and the cell temperature control will only be valid if the conditions as regards the nearness to the hottest point of the batteries and the nearness to the dissipation resistance are complied with.

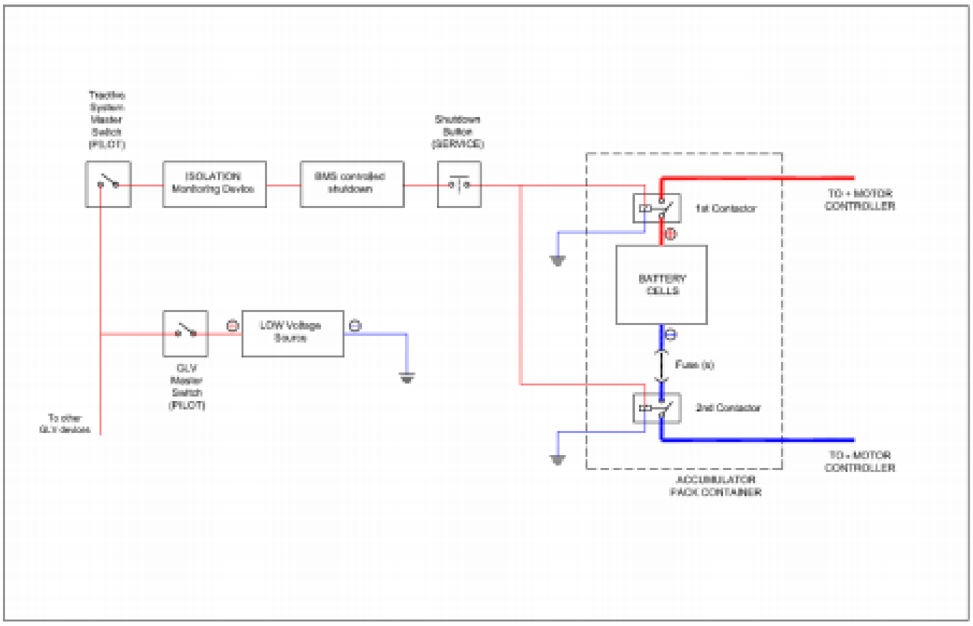
D.3.5.10 The BMS system must deactivate the vehicle traction if the voltage of one of the cells is discharged to the critical minimum voltage or if the critical maximum temperature of the cell is exceeded, according to the values indicated by the manufacturer. This deactivation is compulsory and must happen at the same as the contactors of the battery accumulator open. (Figure 31) defines the two disconnection circuits defined by MotoStudent to ensure that all teams at least have that minimum security in their prototypes.

D.3.5.11 In addition to the conditions set-out in Art. D.3.5.10, it is allowed to progressively limit the electric power delivered to the motor until being equal to zero in the critical voltage point of the cell or the maximum temperature of the cell.

D.3.5.12 The BMS must also deactivate the recharge system when the maximum voltage or temperature levels of the cell are reached. This deactivation may be progressive or prompt.

D.6.1.2 The disconnection system must follow one of the systems described below:

- Disconnection system with contactors that are directly controlled by the disconnection circuit:



- Disconnection system with contactors that are directly controlled by the controller:

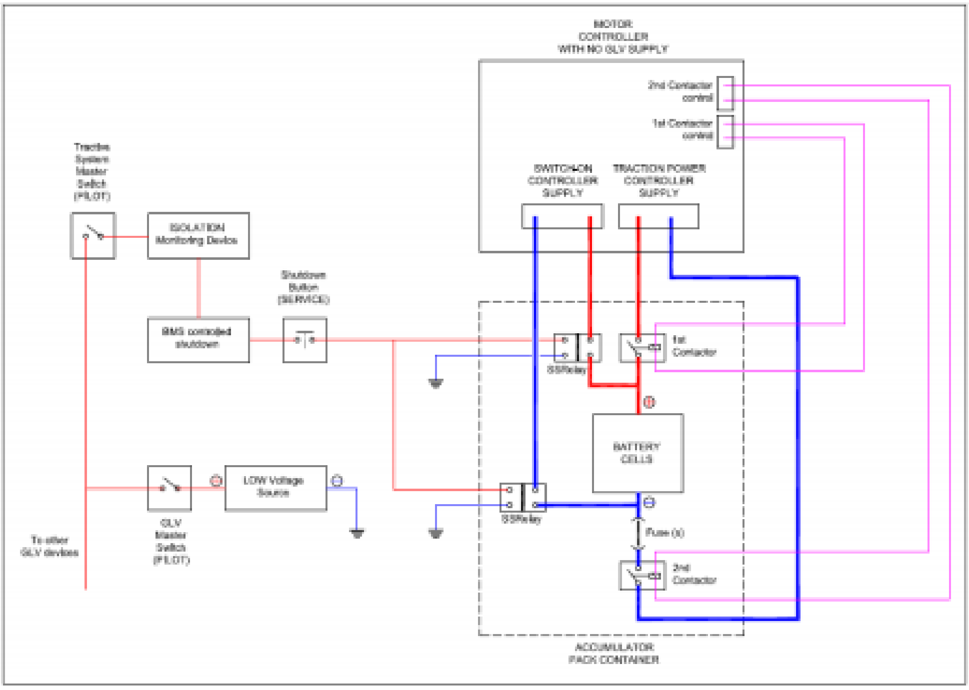


Figure 31. MotoStudent Electric 2015-16 Regulations disconnection system examples.

5.2. BATTERY DEFINITION.

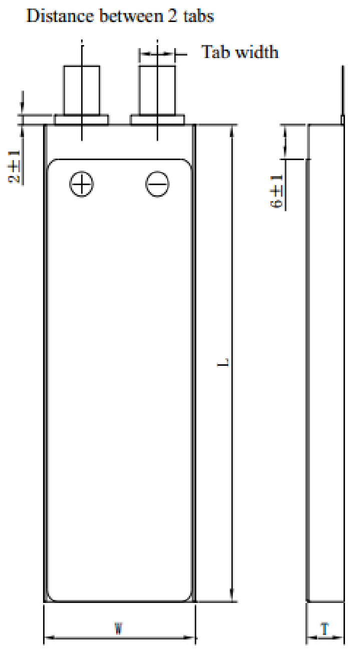
The main characteristics of the electric motor that will be provided are:

MOTOR PERM PMS 150 - RLS	
Type	AFPM Motor
Rated Power	13 kW
Cooling	Aire
Max speed	6.000 rpm (without *field weakening*)
Rated voltage	96 VDC
Rated Current	153 A
Peak Stall Current	550 A
Rated Torque	20,7 N.m
Peak Stall Torque	71 N.m
Motor Constant, Ke	0,0087 V/rpm
Motor Weight	22,3 kg
Encoder	RLS – RMB29AC01SS1, analogic sin-cosin, power +5V (independient GND)
Temperature sensor	KTY-84

Figure 32. Heinzmann PMS-150. Characteristics

The electric Motor has a peak rated power of 30Kw with air forced cooling. As we need maximum power we will build a battery pack that can offer approximately peak power of the motor as a nominal discharge rate so that with an effective cooling system we may use the engine with its maximum power.

We found a very interesting LiPo cell of 10.000mAh 5C1 and peak discharge current of 7C limited to 2 seconds.[27] With this cell we will define a 5Kw pack made by 5 individual 1Kw substacks parallelized to create the 5Kw battery. Each substack will be composed of 26 cells in series having a 110V-10Ah substack at full charge so a 110V-50Ah battery pack with a nominal discharge of 5C (250A) and a peak discharge of (350A). This will offer a nominal 27.500Kw discharge and a peak discharge of 38.500Kw.



◆标称容量 Typical Capacity①		10.0Ah
◆标称电压 Nominal Voltage		3.7V
◆充电条件 Charge Condition	最大持续充电电流 Max. Continuous charge Current	20.0A
	峰值充电电流 Peak charge current	40A（≤1 sec）
	电压 Charge Voltage	4.2V±0.03V
◆ 放电条件 Discharge Condition	最大持续放电电流 Continuous discharge Current	50.0A
	峰值放电电流 Peak discharge Current	80.0A(≤3 sec)
	截止电压 Cut-off Voltage	3.0V
◆交流内阻 AC Impedance(mOHM)		<2.5
◆循环寿命【充电:1.0C,放电:5C】 Cycle Life【CHA:1.0C,DCH:5C】		>100cycles
◆使用温度 Operating Temp.	充电 Charge	0℃~45℃
	放电 Discharge	-20℃~60℃
◆ 尺寸 Cell Dimension	厚度 Thickness(T)	8.3±0.2mm
	宽度 Width(W)	64±0.5mm
	长度 Length(L)	158.5±0.5mm
	极耳间距 Distance between 2 tabs	29±1.0mm
极耳尺寸 Tabs size and materials	Dimensions of tabs	0.2x15x10mm
	Material of tabs	Nickel-plate copper
◆重量 Weight(g)		188±5
①标称容量：0.5CmA,4.2V~3.0V@23℃±2℃ Typical Capacity:0.5CmA,4.2V~3.0V@23℃±2℃		

Figure 33. Melasta SLPB8564159-10.000mAh 3.7V 5C1 LiPo Cell

5.3. BATTERY MANAGEMENT NEEDS.

Once the battery has been defined and the regulations have been stated we proceed to enumerate the needs or characteristics our system requires to be able to make an intense market study to define how our system will be designed and developed.

Battery:

- Li-Po Cells 3V-4.25V 10.000mAh.
- 26 Cells in series per substack 5 substacks in parallel.
- 130 Cells in total.

Battery Management System:

- Overvoltage protection.
- Undercharge protection.

- Cell temperature protection at least 30% →39 Cells.
- Balancing Resistors Temperature sensing →130 resistors.
- Current sensing to avoid overcurrent.
- Total Voltage control & SOC.
- Battery State control (CH or DCH).
- Can Bus Communication with ECU.

Now that all the needs have been stated, we will proceed to select the ideal topology for our battery pack and which type of balance we will use. We first look at the market to see if any commercial BMS that has all of these features. The systems mentioned in the state of the art are the most popular battery managers and none of them has the right characteristics for our pack. The closest would be "Freemens FreeSafe" which monitors 24 cells has possibility for current sensing and Can Bus communications module. Lithiumate would be a good solution but as we have a 130 cell stack the amount of cables needed, the amount of boards would be too many, and we have more space limitations than an automobile so we do not consider it viable, the same problem occurs with Flex BMS. We do not have to forget we are building a Motorcycle.

5.4. TOPOLOGY AND BALANCING TYPE SELECTION.

As it has been stated in the theoretical framework, we have three possible topologies distributed, modular and centralized, and two types of balancing, passive and active.

To start with, we have to evaluate the amount of cells to be monitored and the topology of the battery pack. As we have mentioned before our pack will be made of 130 cells, 26 per substack being 5 substacks in total, let us see what would be each BMS topology configuration and choose the optimal.

To compare in a clear way what each topology would imply in our battery pack we design (Table 2) where we compare the most relevant features: Number of boards needed, amount of cables per board, adaptability of the BMS if battery pack needs to be reduced or enlarged once the prototype is built and a simple cost estimation. The first 3 features will be given the same value each and cost estimation as we are building a BMS for a competition prototype liability is more important than costs will be considered in case of drawn. We will use 3 colors: Green, Yellow and Red giving 2, 1, -1 points respectively, the combination with more points will be the chosen topology.

The terms of designation will be as follows:

- The more boards the complexion of the system may be higher and a greater space is needed.
- The greater amount of cables to a single board the more complex it is to make a non-disturbing cable setting.
- Whether a BMS is adaptable or not is defined by its flexibility to be able to monitor more/less cells or substacks if any future changes to the system occur.

Table 4. BMS options comparison.

Topology	Board nº	Cables/board	Adaptability	Cost Estimation	Score
Distributed	130+Master	2	Yes	High	2
Modular	5+Master	27+18(Temp)	Yes	Medium	**4**
Centralized	1	131+78(Temp)	No	Medium	0

As we can see the topology and distribution of out battery pack, makes both distributed and centralized topologies inappropriate and modular topology seems to suit the purpose. Not only does it have the higher score but it also has no negative features. In conclusion, we have our topology, and now we need to decide what type of balancing we will choose.

The last thing we need to define before we can proceed with our project is the balancing type. Taking into account what we have mentioned in the theoretical framework of each type of balancing. We realize the most realistic balancing type would be passive balancing. As it is simpler and we are not designing a BMS for a city EV but for a competition vehicle, it will have to do 5 laps so we need quick charging and we will not be balancing the battery during discharge as that would imply a higher complexity which will not be required for the purpose of the competition. Also having done a previous research on a much more mature similar competition for instance, Formula Student Electric all of the teams choose passive balancing over active balancing.

Therefore, we have the following:

Topology → **Modular.**

Balancing Type → **Passive.**

5.5. Chip Selection.

Now that we have clarified what type of BMS we want to design and develop, we will start a Battery Monitoring Chip research in which the chip monitors a group of cells and which has passive balancing in-built circuit. In addition, we have to remember regulation 3.5.5 which states that 30% of the cell temperatures have to be monitored. Therefore, we need a chip which also has in built temperature sensing circuits or general-purpose inputs from which to read temperature with and an external conditioning circuit. In our research for the existing chips in the market we found the following document [28], a comparative table of the chips available from different companies the table may be checked in annex 1.

From the table we selected the two companies from which we have found information about their chips in applications of commercial BMS. These are Texas Instruments and Linear Technology. Texas Instruments chips have been used by some professionals for developing battery management applications[29]. Linear Technology chip LTC6803-1 has both been used commercially in the case of Freemens BMS and in university projects[30] [31].

In table 3 we have done a selection of the chips this two companies have and outlined their main characteristics:

Table 5. LT and TI chip comparative

COMPANY	CHIP	NUMBER OF CELLS	BALANCING	EXTRAS
LINEAR	LTC6801	2-12	NO	NO BALANCING
	LTC6802-1	2-12	PASSIVE	2TEMP/2GPIO
	LTC6802-2	2-12	PASSIVE	2TEMP/2GPIO
	LTC6803-1	2-12	PASSIVE	2TEMP/2GPIO
	LTC6803-2	2-12	PASSIVE	2TEMP/2GPIO
	LTC6803-3(UPGRADE 03-1)	2-12	PASSIVE	2TEMP/2GPIO
	LTC6803-4(UPGRADE 03-2)	2-12	PASSIVE	2TEMP/2GPIO
	LTC6804-1	2-12	PASSIVE	BALANCING/5GPIO
	LTC6804-2	2-12	PASSIVE	BALANCING/5GPIO
TEXAS INSTRUMENTS	BQ76940	9-15	PASSIVE	BALANCING
	BQ76930	6-10	PASSIVE	BALANCING
	BQ76920	3-5	PASSIVE	BALANCING
	BQ76PL536A	3-6	PASSIVE	BALANCING/3GPIO
	BQ76PL56A-Q1	3-6	PASSIVE	BALANCING/3GPIO

The previous table has the information needed to choose one of the chips as it covers the main aspects of the topology and balancing type chosen previously. All of them have passive balancing measure more than 3 cells and have extra inputs for temperature measurement. The characteristics needed to construct our modular BMS and follow the regulations that MotoStudent requires. The most advanced chips from both companies are LTC6804 and BQ76PL54A-Q1[32] (enhanced version of BQ76PL54A) should be the options to choose as both have all the requirements we need. So we will need to look further into them to make the right decision taking into account the time we have to develop the BMS and our resources. A further expansion of the table above is found in the excels Appendix.

Table 6. LTC6804 & BQ76PL536 further comparison

Chip	Communication	Ev. Boards	Protection	Precision	Software
LTC6804-1	IsoSPI	DC1894B	OV/OC/UV	0,25%	Libraries Included
LTC6804-2	IsoSPI	DC1942C	OV/OC/UV	0,25%	Libraries Included
BQ76PL56A-Q1	SPI	BQ76PL56A-EVM	OV/OC/UV	0,25%	No libraries

The difference between LTC6804-1 and LTC6804-2[33] is based on the type of communication between the chips and the master board. LTC6804-1 uses Daisy-Chained communication a typical use of SPI and LTC6804-2 uses addressable communication protocols which are a bit more complex to program than daisy chainable.

The TI chip does not have an isolated SPI communication but isolation may be built externally, it also does not include libraries or example software.

As both chips work at registers level, a very low level in programming having no software example or libraries from where to work would be a handicap added to the already existing one: "creating a custom BMS". So that is a clear reason to use linear's chip instead of Texas.

The amount of time we have to develop this project sets us in the path of looking for the easiest path to construct our BMS in order to have a greater time to find and correct any possible errors. Having a greater time to debug both the code and the hardware will grant us with a more reliable final design and product.

So the selected chip will be **Linear Technology LTC-6804-1**[33].

5.6. ELECTRONIC DESIGN.

In (Figure 34) we have a high level diagram describing our energy storage system. In it we can see the relationship between the BMS and the battery pack and the security system, which results in a Smart Battery Pack. We also can observe the relationship that this smart battery pack has with the rest of the EV.

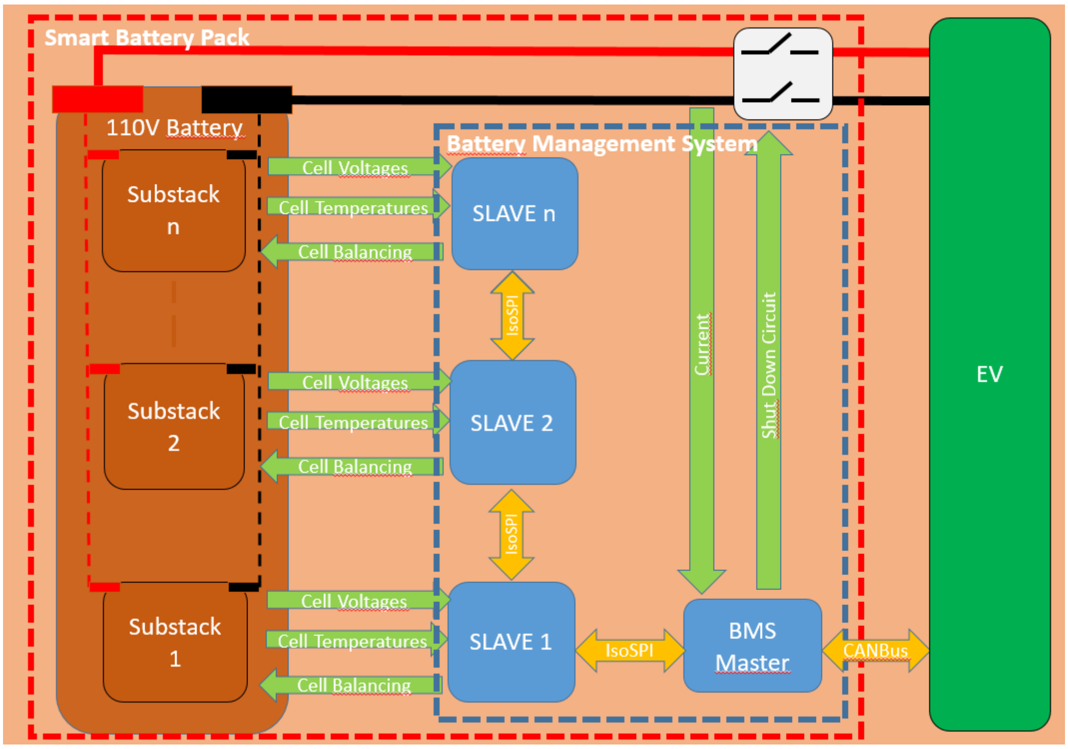


Figure 34. High Level diagram of the System

The Battery Management Systems is composed by a Master and a number of slaves depending on how many substacks the battery pack has. As we know we are designing a modular BMS so each slave is considered a module. The Master receives information via and Isolated SPI from all the slaves which are connected in a daisy-chain topology. It also communicates to the CAN Bus network of the system. The other two features of the master are current reading from the battery pack output and control of over the contactors of the battery pack to avoid both OV and UV. On the other hand, we have the slaves which receive both cell voltages and temperature measurements from the substacks and enable balancing when needed.

When we unify both BMS and Battery pack we have a smart battery pack which protects itself from Over-Voltage, Under-Voltage and Over-Temperature. It also sends the battery data to the CAN Bus network and powers the EV.

5.6.1. Balancing & Monitoring Circuit Module.

To start with, we have to take into account that our battery pack is made of 26 cells in series. As each LTC6804-1 has 12 channels we need at least 3 chips to suit our purpose. In order to have a similar power in each chip and knowing that each chip needs at least 3-4 cells connected to power up, we will try to make an equal distribution of cells for each chip. As 26 is not divisible by 3 we will have 2 chips connected to 8 cells each and one connected to the 10 cells left. As the 12 channels of LTC6804-1 are made of 2 multiplexers of 6 channels each (as shown in figure 35). We will try to have a symmetrical power input for both of them, that is why we do this even channel distribution.

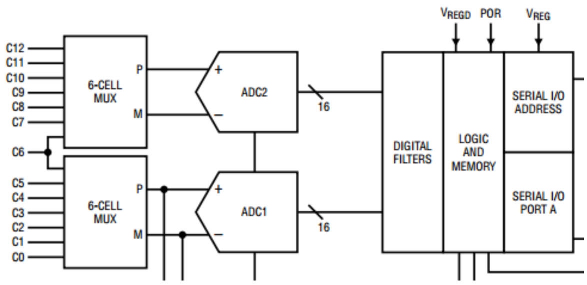


Figure 35. LTC6804-1 Reading Technique.

The default setup for the monitoring and balancing circuit when all 12 channels are read is the one shown in (Figure 36) where we can see that C0 is connected to V- and V+ is connected to the highest cell with a protection resistor. Also we see the "S" outputs, these are connected to the internal mosfets that trigger balancing on the connected cells when is required by the chip. We can observe RC filters to avoid any erroneous reading.

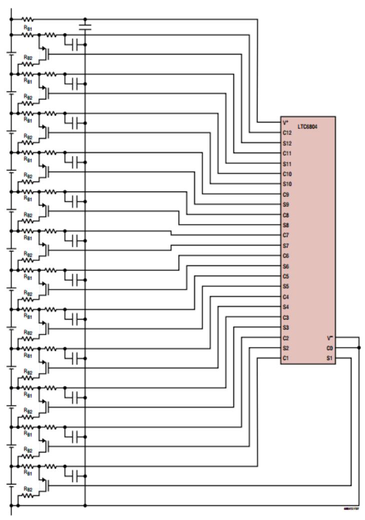


Figure 36. LTC6804-1 12Ch Monitoring & Balancing setup

Now we have to take a look to the recommended circuit stated in the datasheet and what happens when less than 12 cells are connected. So, from (Figure 37) we figure out actually how does the chip sense cells. Each "C" is a point where voltage is measured in comparison with the previous "C". Given that the chip has 2 internal multiplexers when having a less than 12 cell per chip configuration we remove higher cells for each multiplex. So as for an 8 configuration we remove C12, C11, C6 and C5. For a 10 configuration, we remove C12 and C6. To remove a cell, we simply short-circuit that channel with the previous channel and the balancing channel is not connected.

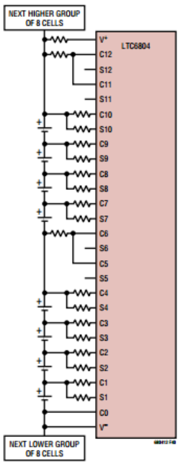


Figure 37. 8 Channel Ltc6804-1 Configuration

Our setup, as said before, is 8Ch-10Ch-8Ch to make a total of 26 Cells to be monitored and balanced. The following two figures 38 & 39 show the configurations adapted to our needs.

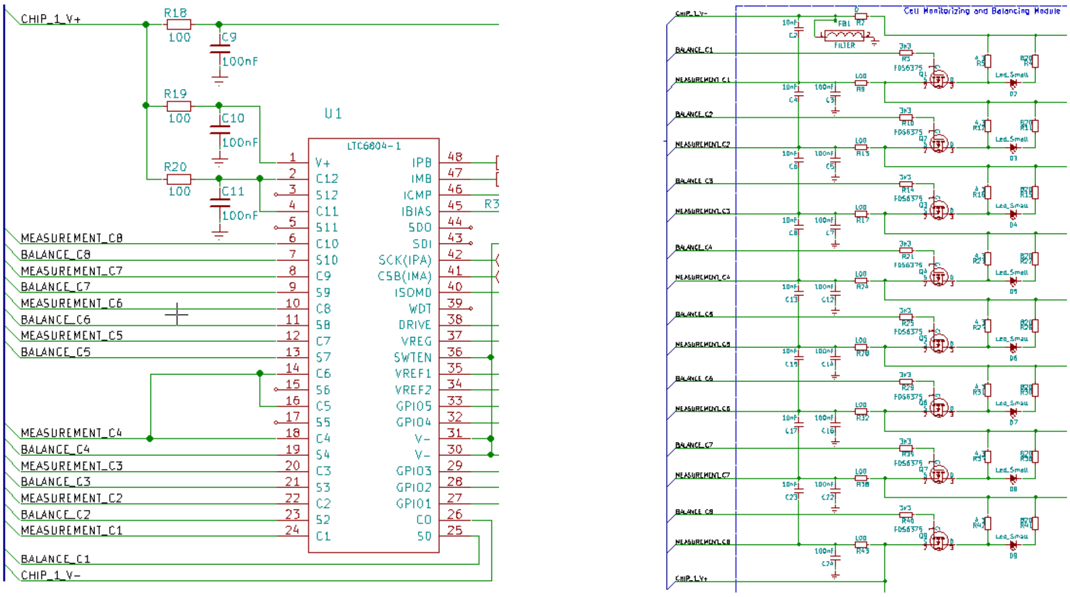


Figure 38. Chip & Monitoring circuit Setup for 8 Cells.

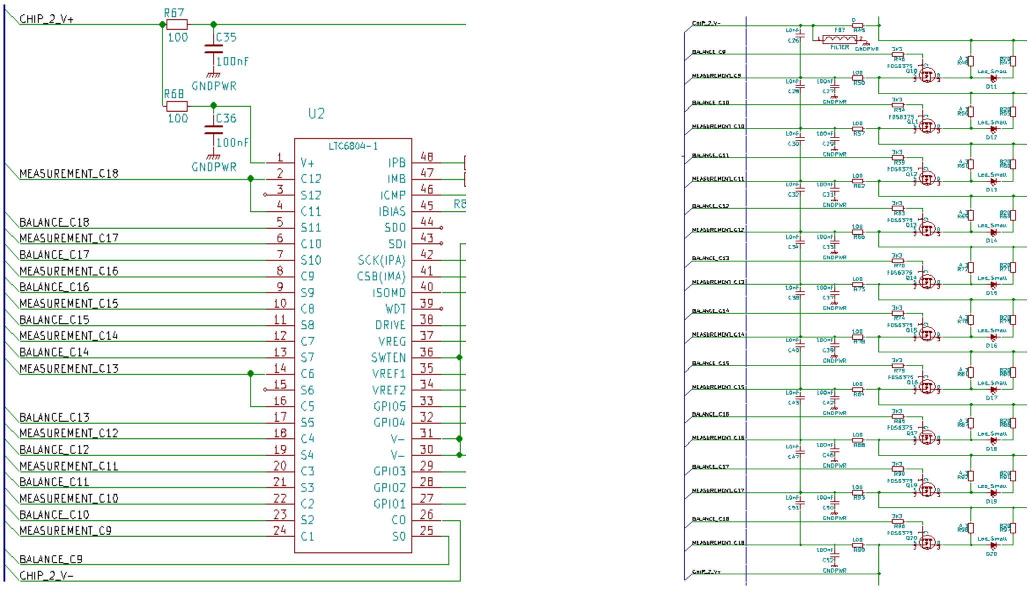


Figure 39. Chip & Monitoring circuit Setup for 10 Cells.

A clear difference from (Figure 36) circuit is that we actually have 2 filtering circuits, an RC circuit and a differential filter with a capacitor between channels. This does not mean we will actually use both filters but it gives us the option to choose between them or use both depending on the filtering needs we actually have once we try the system. Both configurations are mentioned in the datasheet as seen in (Figure 39) and both are implemented in the DemoBoard DB1894B Schematic which you can check in the components and DemoBoards appendix.

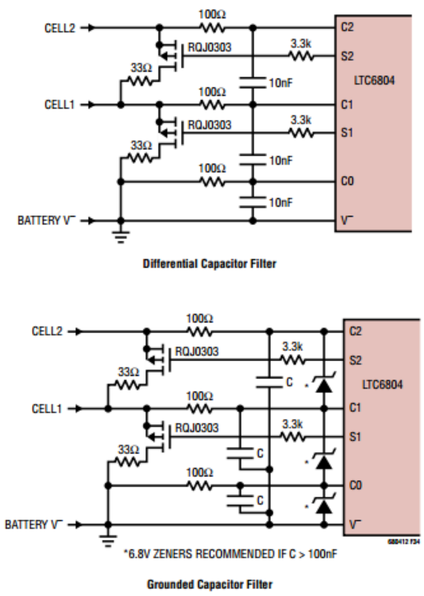


Figure 40. Filtering options for Ltc6804-1.

Last but not least, in both (Figures 38 & 39), we can see a modification to the recommended circuit has been added. A LED and a resistor are placed in parallel with the power resistor. The LED is there as a verification circuit to test the balancing circuit works correctly.

5.6.2. Chip Powering Set-up.

Now, for our chip to work, we have to define the setup. We already have a way to power our chip as the monitoring circuit has been designed, now we need to be able to send that power to the chip in a stable way. To do this we will build a simple linear regulator as it is specified in the datasheet, (Figure 41) shows the configuration.

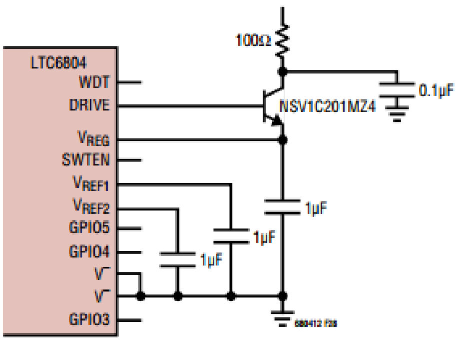


Figure 41. Simple Linear Regulator for Ltc6804-1

Ltc6804-1 draws its power from its VReg pin. VReg needs and input of 4.5-5.5V to get the chip to work. A higher voltage would result in the death of the chip and a lower voltage results in a malfunction of it, depending on how many functions are tried to be used at once. There are two ways of powering the chip through VReg, a DC-DC to 5V may be used powering it from the most significant cell and GND or DRIVE pin may be used to form a regulator combined with an NPN transistor, the only requisite for the NPN is to be able to have a collector power of more than 6V.

The DC-DC may be the more reliable option but it also may occupy a higher space, as the DemoBoard uses a linear regulator to power the chip and we know for sure it works correctly we decide to use this method as seen in (Figure 42). When building the regulator, we need to take into account 3 main things. The collector power connection of the NPN should be protected by a RC filter to avoid any transistors. On the other hand, the NPN emitter has to be bypassed with a 1uF capacitor, note that its value is recommended by linear and using higher capacitors may affect chips Wake-Up time. Lastly, if when tested the NPN heats up when being used as regulator a high thermal characteristic NPN should be used.

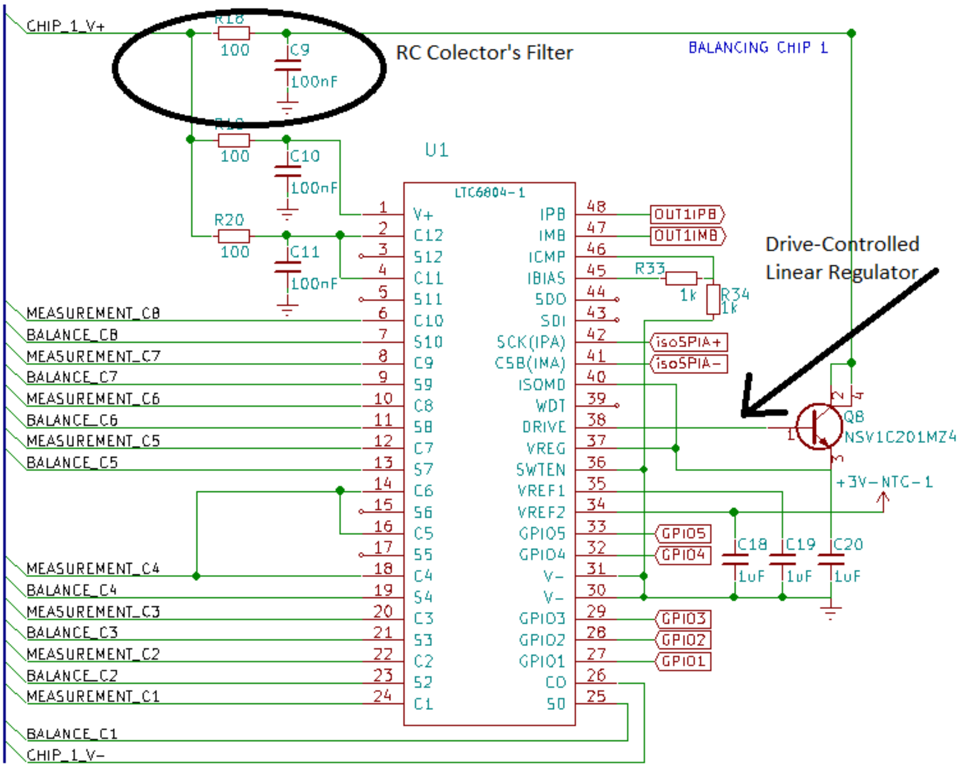


Figure 42. Chip Powering Setup

5.6.3. Temperature Sensing Module.

As the chip does not have inbuilt temperature sensing conditioning circuits but it has 5 General Purpose Outputs we will have to design conditioning circuits for temperature measurement. The chosen temperature sensors are NTC. Accurate leadless thermistors completely suit the purpose in our application. As they have an easy conditioning circuit and also they are the example used in the chips datasheet for battery management system applications.[33]

To start with, we will design the circuit for temperature measurement with NTC (Figure 43) shows the typical circuit:

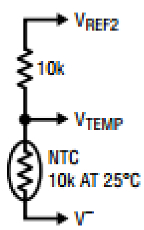


Figure 43. Typical temperature probe circuit.

We need to sense 30% of the cells this means we need at least 8 reading per slave and we have 15 channels available 5 per chip. Using 3 channels per chip would be enough but if a channel breaks we still sense at least 30% we will use 4 channels per chip and a total of 12 sensors per board. Having 46% of the cells monitored gives a higher specification to our BMS. We won't use all 5 GPIO, as tests with the evaluation board show that using all of them without all the channels connected results in power problems. In (Figure 44) we can see how we have configured the temperature sensing, the capacitors in parallel with the NTC are there to filter any possible electrical noise that may be sensed during readings.

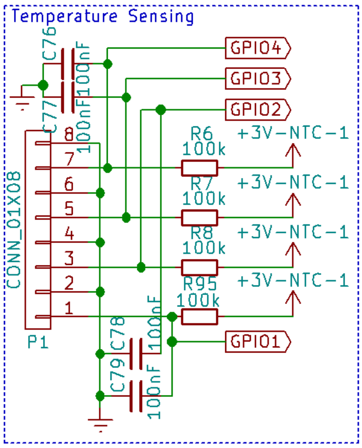


Figure 44. Temperature Sensing Module

$VRef_2$ is used as a power reference for the NTCs as it is recommended in the datasheet. VReg has 5V regulated power but its where the chip powers up so adding extra current draining systems there may cause a malfunction of the chip as too much current is tried to be drained from that point. $VRef_2$ is designed specially to be able to bias several NTCs connected to the GPIO pins, that is why we use it as source voltage. The biasing resistor of the NTC is chosen so that when being at 25º (NTC at its nominal resistance value) the voltage is 1.5V half of the source voltage 3V ($VRef_2$ output voltage).

5.6.4. Communications Module.

One of the main reasons for choosing this chip was its isolated serial communications capabilities. As shown in (Figure 45), an internal encoder/decoder turns the IsoSPI into standard SPI so that the chip is able to read them. To be able to decode the signals received from the chip by a microcontroller LTC6820 is needed to be used as a coder and decoder so that microcontroller and chip can communicate. Isolation is provided by the external transformer and the strength of the pulse of the transmission is set but Rb_1 and Rb_2. To ensure a reliable communication, all chips and devices need to have the same pulse strength, as different pulse strengths may cause interferences between devices as they are set in a daisy-chain topology.

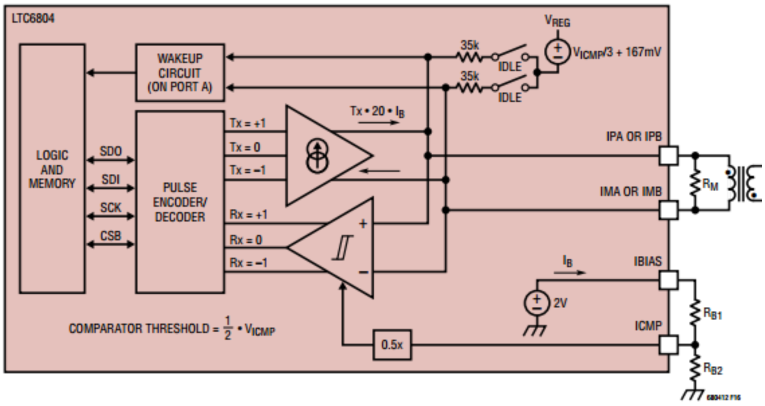


Figure 45. IsoSPI Interface[33]

We need to connect 3 chips in the same PCB and then we need to have an entrance communication module and an output module to connect with the next or previous chip. Note that the first slave will have to connect with the master and the master will require its own way to decode the information coming from the chips, this is where LTC6820 comes into scene. The datasheet states how the connection must be between chips in the same PCB and from one PCB to another. (Figures 46 & 47) are taken from the datasheet to enlighten us with the right circuit to be used.

Basically we see in (Figure 46) that two transformers with their Rm to ensure there is no signal reflection. The need of two transformers comes from the twisted-pair cable as the distance it covers from one PCB to another may catch some EMI. To avoid this EMIs a transformer with a common mode noise choke may be used instead of a standard 1:1 transformer.

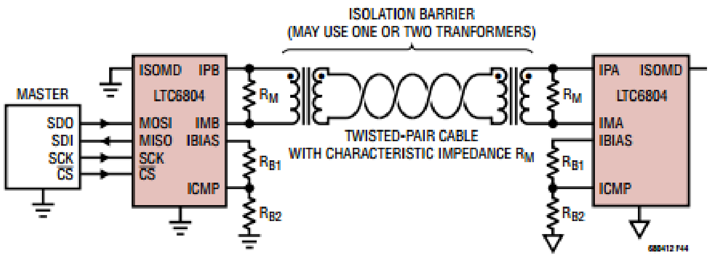


Figure 46. LTC6804-1 daisy-chain IsoSPI between 2 PCB.

In (Figure 47) we can see than only one transformer is needed when the chips are connected in the same PCB. But a Rm is needed in both sides of the transformer, as the communication is bidirectional. In this case standard transformer may be used as EMIs are very low in the same PCB and a standard transformer should be enough to filter them.

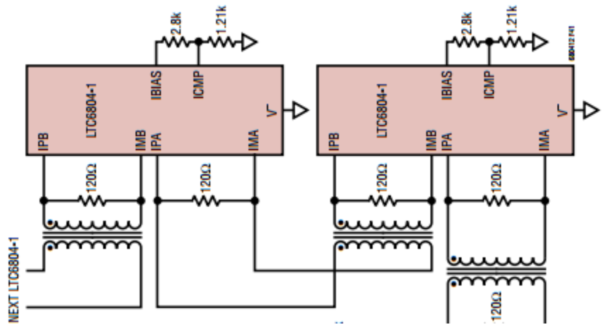
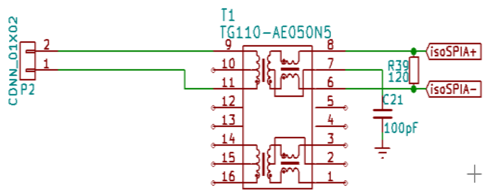


Figure 47. LTC6804-1 daisy-chain IsoSPI in same PCB

Taking into account all what we have read in the datasheet we have created the following circuits shown in (Figures 48 & 49).



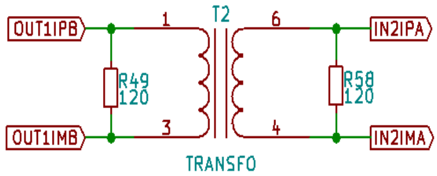


Figure 48. PCB IN/Out communications module *Figure 49. Communication between chips module*

5.7. Slaves PCB design & Components selection.

All the components selected can be checked in the components and DemoBoards appendix which includes the datasheets of all of them.

5.7.1. Balancing Circuit Module.

We have 10A cells so we have to decide which a correct balancing current would be. To do this, first of all, we will go back to what we have learned in the state of the art. We see that none of the commercial BMS go further than 1A-2A balancing currents. This is due to the temperature problems that going further than that may cause. At 4.2V balancing 2A means 8.4W of heat sink in each resistor.[4, p. 195]

Another important point is how quickly we want to be able to balance our battery pack. Many commercial BMS choose to charge the cells and once they are charged, balance the pack to make a final charge once balanced. This is a good option, but it can either be slow or very fast depending on the quality of the cells and the impedance difference between them. The closer to an ideal battery pack we are the better this method is.

To ensure the quickest charging possible the balancing current must be equal to the charging current. Our cells can only be charged at 0.5C which means 5Ah so to have an ideal BMS our balancing current would have to be 5A. Taking into account 5A means 21W heat sink in each resistor, but most of the time the heat sink would be at 3.6-3.9V so 18-19.5W. That amount of heat can damage the electronics and the battery if there is not a highly efficient refrigerating system. Knowing that the battery pack will be refrigerated by forced air that amount of heat would be too much.

A good balancing current would be 1A as it is 20% of the charging current and 4.2W seems an acceptable heat sink. Also Elithion one of the most prestigious BMS developers states that 1A[1] is an efficient balancing current for daily charged battery packs. In our case, the battery will need to be fully charged and balanced for each test of the competition. That is why, we need a quick balancing algorithm, the usual passive balancing in automotive industry is 100mA but for batteries charged and balanced overnight. Meaning more than 8h of charging instead of quick charges of 2-3h.

Although the final balancing current will not be chosen until we check if potency power resistors with those features exist and check both their dimensions and price. We will also need a mosfet which is compatible with those working conditions. We have to take into account that our substack has the following dimensions:

16mm to battery pack Wall,230mm substack length the height is 170mm. These are the maximum dimensions our slave can have.

Finally, we will have to take into account that we will need tracks with enough section to be able to carry 1A current to the balancing resistors, smaller tracks will result in circuit failure and the burning of these tracks.

5.7.1.1. Component Selection.

Power Resistor: As we have mentioned before we set our Balancing Current to 1A because the usual critical balancing takes places at the 4V-4.2V range we will take 4.2 as reference to calculate the power resistor. Once the resistor has been calculated we will need to know the amount of power that it will sink to be able to search the market for an adequate resistor.

$$V = R \cdot I \rightarrow 4.2V = R \cdot 1A$$
$$R = 4.2\Omega$$

$$P = V \cdot I = R \cdot I^2$$
$$P = 4.2W$$

This means that we need a resistor of approximately 4.2Ω and that is able to sink at least 4.2W. We find SMW5 resistors which are capable of power dissipation of 5W and 220ºC. They have resistors of 4.3Ω and their dimensions line up with the requirements of our PCB dimensions. Given that we have a maximum substack length of 230mm and 26 of these resistors together mean a space of 195mm. They also have a width of 6.8mm and we have a margin of 16mm from the cells to the battery wall. They seem to be ideal for our application, as they do not have the exact calculated resistance we should recalculate what the balancing current would be if these were the choice.

$$V = R \cdot I \rightarrow \; 4.2V = 4.3\Omega \cdot I$$

$$I = 0.98A$$

0.98A is acceptable as it is only a 2% less than the desired balancing current. This resistor is used in a similar BMS based on the previous version of our chip and with a balancing current of 1A-2A[30]. Also, this resistor has a version with a heat sink armor so we know that it is compatible with heat sink. We will use its plane surface to attach it with a thermal adhesive to the heat sink. To do this in an efficient way we will need to place all the resistors aligned so all their upper parts create a common plane in which to attach the heat sink.

Mosfet: Knowing that there will be a heat sink of 4.2W in the resistor and a current flow of almost 1A we should look for a mosfet that is able to stand a current flow of more than 1A and has thermal characteristics suited for an automotive application. Our chip uses negative logic to activate balancing so we will need to look for a P-Channel Mosfet.

We find a Fairchild Mosfet that is used for power management and battery protection with the following characteristics:

-Capability for driving continuously 8A and pulse currents of 50A.

-Power dissipation up to 2.5W.

-Operating temperature range -55º-175ºC.

-Supports up to 20V source voltage.

-S0-8 encapsulated which is easy to hand solder.

These features seem to fit for our application we will have to test the balancing circuit to see the reliability of the chosen components once all have been chosen.

LED: We look for a bright light emitting diode which will indicate if the balancing circuit is working. This is not a key feature for the system to work but it is ideal for testing the boards. We find a SMD green super bright led 0805 with a wide viewing angle and operating current of 2mA-25mA. But taking into account the temperature curve related to forward current seen in (Figure 50) we decide to set a current between 2mA and 5mA.

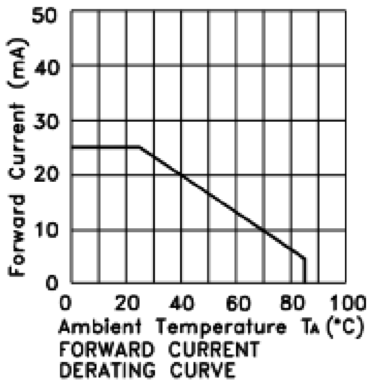


Figure 50. KP-2012LSGC. Forward Current-Temperature Curve.

Now we should calculate the LED's resistor we will calculate the resistor for 2mA and for 5mA to know what is the range of resistor values which can be used.

$$R_{LED} = \frac{(V_{IN} - V_{LED})}{I_{LED}} = \frac{(4.2V - 1.9V)}{2mA} = 1150\Omega$$

$$R_{LED} = \frac{(V_{IN} - V_{LED})}{I_{LED}} = \frac{(4.2V - 1.95V)}{5mA} = 450\Omega$$

Now that we have the values in which the led will operate to a temperature up to 85ºC we have to decide which resistor to choose. Between these values we have the following commercial resistors: 470Ω,680Ω, 820Ω & 1KΩ. We have some 820Ω in stock we decided to use this resistor value. In this case the led will consume 2.8mA dissipating 5mW in the LED and 6mW in the resistor.

ADC external Filter: LTC6804-1 has a built in delta-sigma ADC which has a SINC3 finite pulse response digital filter. This is a very good lowpass filter which can filter almost any higher frequency but the manufacturer has experienced that some fast transient noise still affects the measurement and to avoid this it recommends the use of RC filters. Bearing in mind that more than 100Ω resistance introduces an error to the measurement.[33, p. 58] To set the filter we will look into the ADC accuracy performance curves shown in the datasheet (Figure 50) for the filter selection based on tests carried out by the manufacturer.

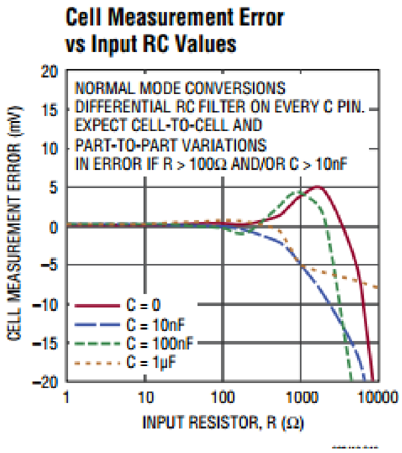


Figure 51. RC Filter Values for ADC Readings

As the chart notes that a capacitor of more than 10nF may result in erroneous readings we decide to use a filter of 100Ω and 10nF, the same one that is being used in their DemoBoard. Also, we decide to open the possibility of an RC grounded filter if a high amount of noise is present in the application.[33, p. 58] That filter will be of 100nF as used in the DemoBoard.[34]

For reference ground channel and first reference of the chip to cell measurement C0 we will not use this filter as the recommended circuit short-circuits C0 and V- Pin in the chip. As it is the first cell measurement and to protect the first channel we will use a 0Ω resistor to be used as a fuse if any short-circuit occurs during the BMS connection to the cells. To ensure there is no noise we will use a ferrite as a filter or a very low impedance coil for high frequency filtering.

Heat Sink: To avoid measuring the temperature of each balancing resistor we decide to put a common Heat Sink to all of them in order to measure the absolute heat of all of them. Also the Heat Sink adds dissipation properties to the circuit resulting in a lower temperature when balancing. Heat sink calculation is explained in (Figure 52). We will consider and ambient temperature of 25ºC and a temperature of 120ºC as the maximum temperature for the union to reach. As the idea is to put all the resistors together and at the same level, we will use a thermal tape usually used as an alternative when the heat sink do not have a way to be attached to the component. This thermal tape has a Thermal impedance of 1.8ºC/W it may be checked in the components annex.

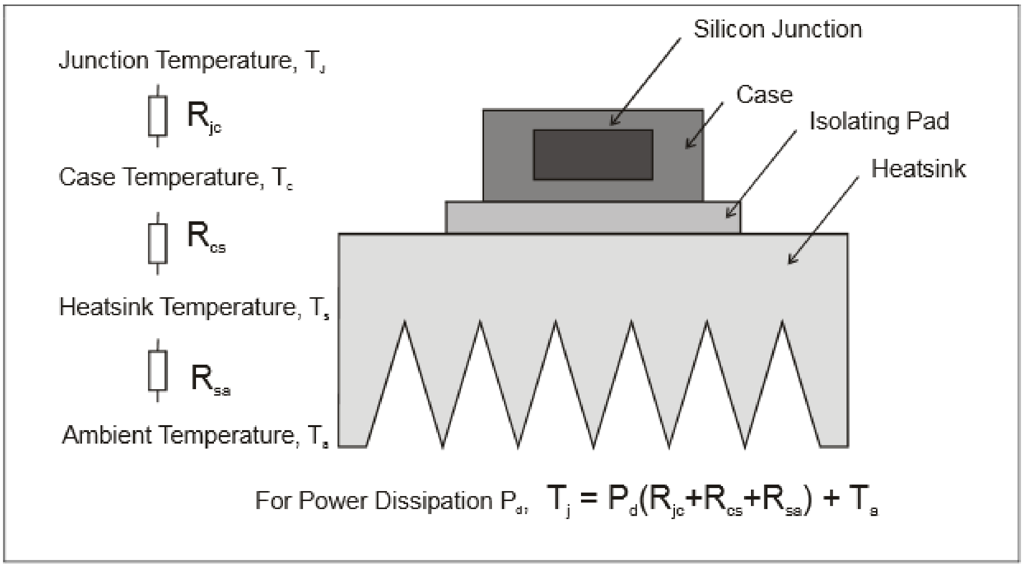


Figure 52 Heat-Sink calculation

T_j=120ºC

R_{jc}=1.8ºC/w

R_{cs}=5ºC/w

T_a=25ºC

P_d=5W

$$Rsa = \left[\frac{k \cdot Tj - Ta}{Pd}\right] - Rj = \left[\frac{0.6 \cdot 120 - 25}{5}\right] - 6.8 = 0.2ºC$$

That is the resistance of the heatsink needed for each resistor so when choosing an appropriate heatsink we need to take into account how many resistors it will cover.

We choose FISCHER ELEKTRONIK ICK 40B as our heat-sink because it has almost the same width of resistors length and has a length of 6,5 resistors so with 4 of these per slave we have enough to cover the whole lot of resistors. It has a thermal impedance of 8.5ºC/W. Taking into account that we needed at least 0.2ºC/W per resistor, totaling 6.5 Resistors per heat-sink we require at least 1.3ºC/W. So this heat-sink has the appropriate qualities for the purpose. Anyway there is a phenomenon which may occur. As all the resistors are placed together the amount of power concentrated may be higher than the expected and so air cooling may be used as an insurance.

5.7.2. Chip Set-up.

The location of the chips in the PCB will depend on 2 main factors: easier routing and communication channels.

For the use of isoSPI we have to take into account that each chip must be at a least a 2cm distance of the isolation transformer to avoid any magnetic field coupling. Moreover these transformers have to be closer than 2cm to the connector so the distance to each chip will be defined by this criterion and trying to cover all the space possible with the rest of the components.

5.7.2.1. Component Selection.

The selected NPN for the discrete 5V regulator will be NSV1C201MZ4 as it is used in the DemoBoard and it is typically used for DC-DC converters in power management. Its characteristics are 100V,2A and Low V_{CE} saturation. Also is qualified for automotive applications and has an automotive temperature working range. Collector-Emitter voltage will be less than 42V in the 10 cell chip and 33.6V in the 8 cell chips and this NPN is capable of admitting 100V.

5.7.3. Temperature Sensing Module.

The typical temperature sensor used in cell temperature sensing is a 10K NTC we decide to use 100K NTCs to reduce power consumption by 10. The usual conditioning circuits for 10K NTC are basically a 10K Resistor and a grounded capacitor in parallel with the NTC to filter any possible electrical noise. This capacitor is usually of 100nF for 10K NTC. As we will use a 100K NTC, the conditioning circuit will need to have a 100K resistor in order to read a voltage half of the reference voltage at ambient temperature (25ºC) as seen in (Figure 53)

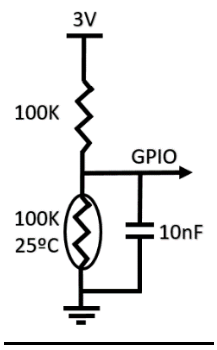


Figure 53.Temperature Sensing Circuit

Note that ifs the resistor is 10 times the usual one, the current will be 10 times smaller so power consumption will be reduced by 10. As the chip only powers the temperature circuit when it reads it to reduce power usage, the 100nF capacitor may be too big to fill with the current reduction caused by the 100K resistor so putting a 10nF capacitor may be a good way to ensure not having erroneous readings. Note that the capacitor is in parallel with the NTC so if the capacitor does not reach the measurement voltage the reading will be false.

Once the NTC is selected we will proceed to calculate the transfer function that defines it.

5.7.3.1. Component Selection.

Since we have decided to use 100K NTCs, now we have to choose an adequate one for our purposes. This means we need to look for a precision NTC with not very big dimensions as we do not have much space. While we were looking for information on temperature sensors we found NTCs used for automotive applications. In this case, AVX accurate leaded and leadless thermistors so we will proceed and check their catalog to see if there is a model that suits our needs.

NJ28RA0104H is a 100K thermistor that works in a range of -55º to 150ºC and has a response time of under 2 seconds.[35]

Once selected the sensor, we proceed to calculate the transfer function to be able to read temperature from the voltage input we receive via de GPIO as seen in (Figure 53). To do so we will have to know how does temperature affect the thermistor. (Tables 5 and 6) have been extracted from the datasheet of the NTC and from the general AVX thermistor characteristics datasheet.

Table 7. Accurate AVX Thermistor Types

Types	Rn at 25°C (Ω)	Material Code	B (K)	α at 25°C (%/°C)
N_ _ _ KA 0202	2,000	KA	3625 ± 1%	– 4.1
N_ _ _ MA 0302	3,000	MA	3960 ± 0.5%	– 4.5
N_ _ _ MA 0502	5,000	MA	3960 ± 0.5%	– 4.5
N_ _ _ MA 0103	10,000	MA	3960 ± 0.5%	– 4.5
N_ _ _ NA 0103	10,000	NA	4100 ± 1%	– 4.6
N_ _ _ PA 0203	20,000	PA	4235 ± 1%	– 4.8
N_ _ _ QA 0503	50,000	QA	4250 ± 1%	– 4.8
N_ _ _ RA 0104	100,000	RA	4380 ± 1%	– 4.9

* _ _ = Add type as outlined above (Example NJ 2 8)

The resistance variation due to temperature, depends on the material the NTC is made of. In our case, we have an "RA0104" thermistor which has a material code: 4380. Knowing the material code, we will proceed to go to (Table 6) and see the relationship between temperature and resistance variation.

Table 8.Thermic Factor for Material code RA4380

T	Material code B (K)									T
	QA 4250			R 4400			RA 4380			
(°C)	R (T) / R25	TF (%)	α (%/°C)	R (T) / R25	TF (%)	α (%/°C)	R (T) / R25	TF (%)	α (%/°C)	(°C)
- 55	1101.8	8.3	- 7.36	113.9	25.9	- 7.42	110.7	8.6	- 7.53	- 55
- 50	71.33	7.2	- 7.13	79.69	22.4	- 7.22	77.22	7.4	- 7.29	- 50
- 45	50.51	6.2	- 6.91	56.29	19.2	- 7.03	54.43	6.4	- 7.07	- 45
- 40	36.14	5.3	- 6.70	40.12	16.4	- 6.84	38.76	5.5	- 6.85	- 40
- 35	26.11	4.5	- 6.50	28.85	14.0	- 6.66	27.86	4.6	- 6.65	- 35
- 30	19.05	3.8	- 6.31	20.92	11.8	- 6.48	20.22	3.9	- 6.46	- 30
- 25	14.02	3.2	- 6.12	15.29	9.8	- 6.31	14.81	3.3	- 6.27	- 25
- 20	10.41	2.6	- 5.85	11.27	8.1	- 6.14	10.94	2.7	- 6.09	- 20
- 15	7.791	2.1	- 5.78	8.367	6.6	- 5.98	8.143	2.2	- 5.92	- 15
- 10	5.879	1.7	- 5.62	6.260	5.3	- 5.83	6.112	1.8	- 5.76	- 10
- 5	4.470	1.3	- 5.46	4.719	4.1	- 5.67	4.622	1.4	- 5.60	- 5
0	3.424	1.0	- 5.31	3.583	3.1	- 5.53	3.522	1.0	- 5.45	0
5	2.642	.7	- 5.17	2.739	2.3	- 5.38	2.702	.8	- 5.31	5
10	2.052	.5	- 5.03	2.108	1.5	- 5.24	2.087	.5	- 5.17	10
15	1.605	.3	- 4.90	1.634	.9	- 5.11	1.623	.3	- 5.03	15
20	1.263	.1	- 4.77	1.274	.4	- 4.97	1.270	.1	- 4.91	20
25	1.0000	0.0	- 4.65	1.0000	0.0	- 4.84	1.0000	0.0	- 4.78	25
30	.7965	.1	- 4.53	.7897	.4	- 4.72	.7920	.1	- 4.66	30
35	.6380	.3	- 4.42	.6273	.9	- 4.60	.6308	.3	- 4.55	35
40	.5139	.4	- 4.31	.5012	1.4	- 4.48	.5052	.5	- 4.43	40
45	.4162	.6	- 4.20	.4026	1.9	- 4.36	.4068	.6	- 4.33	45
50	.3388	.8	- 4.10	.3255	2.5	- 4.25	.3292	.8	- 4.22	50
55	.2771	1.0	- 4.00	.2644	3.1	- 4.14	.2678	1.0	- 4.12	55
60	.2278	1.2	- 3.90	.2159	3.8	- 4.04	.2189	1.3	- 4.02	60
65	.1881	1.4	- 3.81	.1772	4.5	- 3.03	.1797	1.5	- 3.93	65
70	.1560	1.7	- 3.72	.1462	5.2	- 3.83	.1483	1.7	- 3.84	70
75	.1300	1.9	- 3.63	.1212	5.9	- 3.74	.1228	2.0	- 3.75	75
80	.1088	2.1	- 3.55	.1009	6.7	- 3.64	.1022	2.2	- 3.67	80
85	.0914	2.4	- 3.47	.08441	7.4	- 3.55	.08537	2.5	- 3.58	85
90	.07708	2.6	- 3.39	.07093	8.2	- 3.46	.07160	2.7	- 3.50	90
95	.06527	2.9	- 3.31	.05985	9.0	- 3.38	.06029	3.0	- 3.42	95
100	.05547	3.2	- 3.24	.05072	9.8	- 3.29	.05095	3.2	- 3.35	100
105	.04731	3.4	- 3.17	.04315	10.6	- 3.21	.04322	3.5	- 3.28	105
110	.04049	3.7	- 3.10	.03686	11.4	- 3.13	.03679	3.8	- 3.21	110
115	.03160	12.2	- 3.06	.03478	3.9	- 3.03	.03143	4.1	- 3.14	115
120	.02996	4.2	- 2.96	.02720	13.1	- 2.98	.02693	4.3	- 3.07	120
125	.02590	4.5	- 2.90	.02349	13.9	- 2.91	.02316	4.6	- 3.01	125
130	.02246	4.7	- 2.84	.02036	14.7	- 2.84	.01997	4.9	- 2.94	130
135	.01953	5.0	- 2.78	.01771	15.6	- 2.77	.01728	5.2	- 2.88	135
140	.01704	5.3	- 2.72	.01545	16.4	- 2.71	.01499	5.4	- 2.82	140
145	.01490	5.5	- 2.67	.01353	17.2	- 2.64	.01305	5.7	- 2.77	145
150	.01307	5.8	- 2.61	.01188	18.1	- 2.58	.01138	6.0	- 2.71	150

We proceed to make and excel document (Tables Appendix) to make the appropriate calculations and extract a transfer function from the data obtained. To do so, we use temperature and resistance % variation columns from (Table 6) and calculate the resistance value for each temperature by 5ºC each value from -55 to 160ºC.

Our circuit is made up of 2 resistors in series with a reference voltage so to calculate the current that goes through the system we calculate as follows:

$$I = \frac{Vref}{R_1 + R_{NTC}}$$

And the voltage fall in the NTC resistor will be:

$$V_{NTC} = I \cdot R_{NTC}$$

From this last equation we will find the relationship between the read voltage from the GPIO and temperature. We obtain the following transfer function as seen in (Figure 54). To know the exact transfer function, we obtain a tendency line and we realize that a logarithmic tendency results in almost a perfect match obtaining:

$$NTC_{TEMP}(V) = -32.59 \cdot \ln(V) + 37.626$$

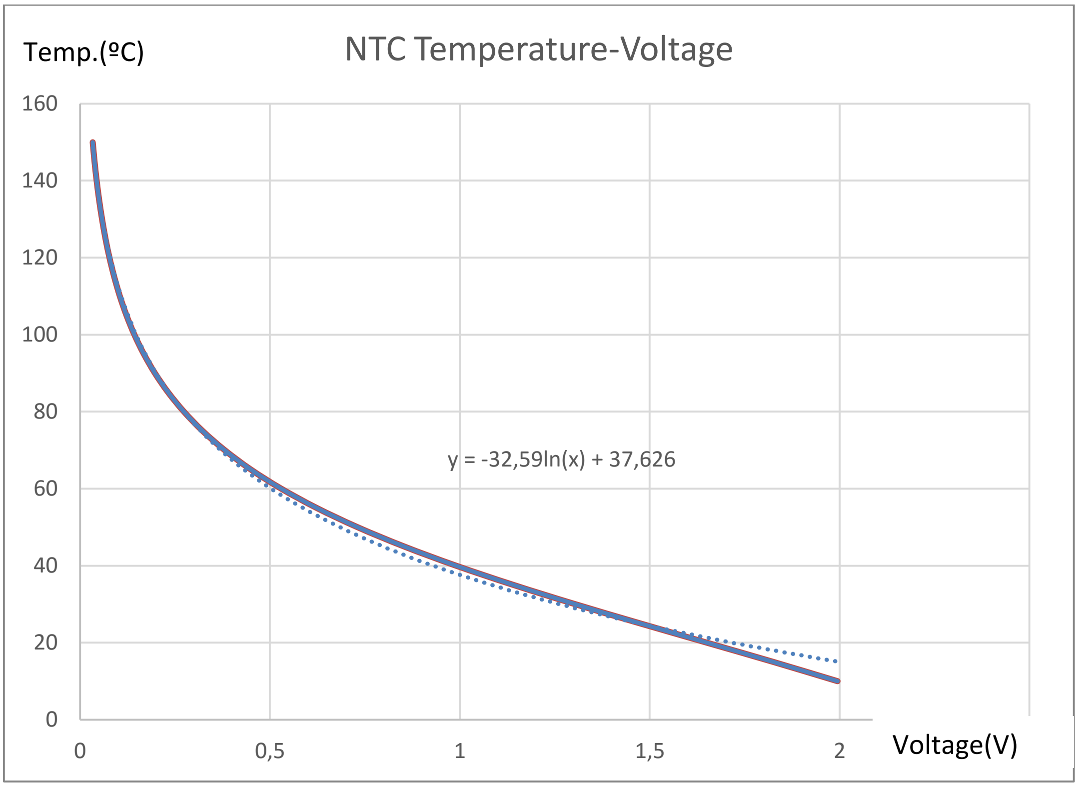


Figure 54. NTC Transfer Function

5.7.4. Communications Module.

The IBIAS is held at 2V when the chip is in awake state, the current driven by this pin sets the intensity of the signal that is transmitted. This current has to be in between 100uA and 1mA. The lower this current is, the smaller the amplitude of the differential signal is. Usually for cable lengths of less than 50m a current of 0.5mA is used and for higher lengths 1mA is used. This current is set by a resistor divider which we define as $R_{BIAS} = R_{B1} + R_{B2}$. The IP and IM pin currents are $20 \cdot I_B$.

On the other hand, we need two types of transformers as it has been commented above in the electronic design. One with a common mode choke, for the external communication, and the other a simple 1:1 transformer to create the isolation between chips.

When we proceed to route this section of the PCB we will have to be very careful that the impedance of the tracks IP and IM is the same as they carry a differential signal and so the higher the difference between both impedances, the higher the possibility of losing the signal or having erroneous readings.

5.7.4.1. Component Selection

IBIAS Resistors: As we have reduced power consumption by using 100K NTC we can ensure a good communication by setting the IB current to 1mA which sets the communication signal to the highest amplitude that the chip admits. We do this because communication is the most critical issue as the BMS are stuck to the battery pack meaning there is a high amount of EMIs which may try to interfere communication.

$$Ib = \frac{2V}{Rbias} = 1mA \rightarrow Rbias = Rb1 + Rb2 = 2K\Omega$$

$$Rb1 = 1K\Omega, \qquad Rb2 = 1K\Omega$$

$$Idrv = Ip = Im = 20 \cdot Ib = 20mA$$

$$Vicmp = 2V \cdot \frac{Rb2}{Rb1 + Rb2} = Ib \cdot Rb2 = 1mA * 1K\Omega = 1V$$

$$Vtcmp = 0.5 \cdot Vicmp = 500mV$$

Our drive current will be 20mA and the receiver comparators will detect pulses of ±500Mv. Knowing that our isolation barrier uses a 1:1 transformer with 120Ω resistors in each termination, the transmitted signal will have the following amplitude.

$$Va = Idrv \cdot \frac{120\Omega}{2} = 1.2V$$

Transformers: The datasheet has a list of transformers (Table 7) which have been tested and had positive results achieving communication from this list we will choose which transformers suit our needs.

Table 9.Recommended transformer table extracted from LTC6804 datasheet

MANUFACTURER	PART NUMBER	ISOLATION VOLTAGE	TURNS RATIO	TEMPERATURE RANGE	CM CHOKE	CENTER TAP
Halo	TG110-AEX50N5LF (Dual)	$1500V_{RMS}$	1:1	–45°C to 125°C	Yes	Yes
Halo	TG110-AE050N5LF (Dual)	$1500V_{RMS}$	1:1	–45°C to 85°C	Yes	Yes
Halo	TGR01-6506V6NL	$3000V_{RMS}$	1:1	–40°C to 105°C	No	No
Pulse	PE-68386NL	1500VDC	1:1	–40°C to 130°C	No	No
Pulse	HX1188NL (Dual)	$1500V_{RMS}$	1:1	–40°C to 85°C	Yes	Yes
Würth	7490100111 (Dual)	$1500V_{RMS}$	1:1	–40°C to 105°C	Yes	Yes
Würth	750340848	$3750V_{RMS}$	1:1	–40°C to 105°C	No	No
Sumida	CEP99	3500VDC	1:1	–40°C to 125°C	No	Yes

The chip can only stack up to a voltage of 420V so a 1500V isolation will be more than enough. As we are working in an automotive sector prototype we need to use components which are able to stand temperatures up to 125ºC. This leaves us only with 3 options: TG110-AEX50N5LF, PE-68386NL & CEP99. Considering that TG110 is the only transformer of the selected ones which has a common mode choke this will be the transformer used for external communication. CEP99 has a higher isolation than PE-68386NL but we know that PE-68386NL is used in the LTC6820 evaluation board that we use to communication the master with the slaves so we decide to use PE-68386NL because we know it works perfectly and also it is cheaper than CEP99. Also there is no need of a higher isolation if the external transformer has a lower isolation. Our isolation will be limited by the external isolation.

5.7.5. Connectors and Cable Selection.

For this selection we will have to bear in mind the working conditions that they will have to endure. Current, temperature small space… Also the connector must be easily connected and disconnected to ensure an easy maintenance of the system.

To sum up, there is one limitation coming from the regulations:

D.5.4.3 The cables or conductors pertaining to the HVS must be non-flammable, grade UL-94V0, FAR25 or equivalent.

Cable: As we do not have much space we need a thin cable which does not become a problem when crimping it for being too small and that can stand more than 1A. AlphaWire AWG22 has the perfect features, it can withstand up to 7A and is the smallest cable we can find which crimps easily. It is a Teflon cable which completely meets the regulations stated by MotoStudent being able to work in temperatures further 100ºC. We select white color since it not used in any other subsystem of the vehicle thus ensuring that ensure if we see a broken white cable it is from our system. To distinguish between each connector we will use a Letter Code for both temperature and voltage sensing. The code will be as follows:

X_C_YY → Being XX the substack number and YY the cell number for voltage sensing. For instance, cell 3 of substack 1 will be defined as follows. *1C03*

X_S_YY → Being XX the substack number and YY the sensor number for temp. sensing. For instance, sensor 12 of substack 1 will be defined as follows. *1S12*

Connectors:

The system will have a total of 8 connectors per slave. These 8 connectors will be divided in 3 types 2 positions connectors for communication, 8 positions for temperature sensing and 9 positions for voltage sensing and balancing. The idea is to have all the connectors of the same type to standardize. We are looking for connectors compatible with AWG22 and that have male and female counterparts. Another important thing to decide is if we want 90º connectors or straight connectors. In our case, as we have a small amount of space to work will we assume that straight connectors will help to easy connect and disconnect when maintenance is needed.

We find the perfect match in the XA series of Japan Solderless glass-filled PA66 nylon. These terminals which are designed for a low insertion force and have a guided insertion mechanism as shown in (Figure 55).

These will be the connectors we will use:

Male:

- JST XAP-09V-1
- JST XAP-08V-1
- JST XAP-02V-1

Female:

- JST B09B-XASK-1
- JST B08B-XASK-1
- JST B02B-XASK-1

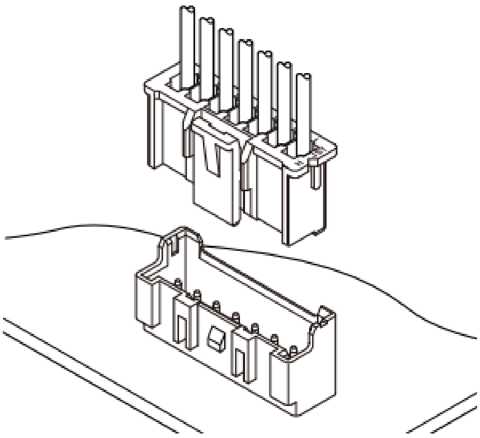


Figure 55. JST XA Series

5.7.6. PCB Design Rules & 3D representation.

To proceed with our design, we will need to set the design rules of the PCB. We will have two types of rules. Once for balancing system and the other for the chip interconnection with each module. We need to design rules because we have two limitations. One is the chip's small patches and the other is that we will have a current of 1A when balancing, this means we need tracks which can withstand that amount of current without breaking.

For the chip circuit all the connections to components which to directly to the chip we will use the following design rules:

- Security Margin of 0.2mm.
- Trace width of 0.25mm. (Limited by the chip patch sizes.)
- Hole trace diameter of 0.5mm.
- Hole diameter of 0.25mm.

For the balancing circuit we will first need to calculate the trace minimum width for 1A currents. To do so we will use the following formulas provided by Advanced Circuits a known USA PCB manufacturer.[36]

The amount of copper in a trace is measured in OZ so all the calculations will be done in OZ instead than in mm as it is the common unit for PCB development. Once we have the result we will convert them into millimeters.

$$Area(mils^2) = \frac{Current\ (A)}{(k \cdot \Delta{}^{\circ}C^{b})^{\frac{1}{c}}}$$

$$Width(mils) = \frac{Area(mils^2)}{Thickness(oz) \cdot 1.378 \left(\frac{mils}{oz}\right)}$$

$$1mil = \frac{1}{1000} inches = 0.0254mm$$

Constants k, b, c are extracted from the Generic Standard on Printed Board Design IPC-2221 a document which is used as a standard for PCB development and manufacture.[37]

These factors depend on the traces, they are different for internal traces, traces which are between PCB layers, and external traces, traces which are in contact with air.

Internal layers:

k=0.024 b=0.44 c=0.725

External layers:

k=0.048 b=0.44 c=0.725

We are developing a 2-layer PCB so our constants will be the external layer constants. We consider a current of 1A and a temperature rise of 15º making a working temperature of 40º a usual working temperature in automotive applications. The thickness of the copper will be 1oz.

$$Width(mm) = 0.611mm$$

We know that with a width of 0.611mm we will have a temperature gradient of 15ºC as we have enough space to do wider traces we will do 1mm traces that is the maximum that we can achieve with the connects we have selected to have a lower gradient and avoid PCB heating at maximum. With this width the gradient will be of about 6ºC.

Our design rules for the balancing circuit will be as follows:

- Security Margin of 0.2mm.
- Trace width of 1mm. (Limited by the chip patch sizes.)
- Hole trace diameter of 2mm.
- Hole diameter of 1mm.

Now that we have the design rules we proceed to place the components and trying to divide them by modules and occupy the minimum possible space. The chosen software for the PCB development is KiCAD as it is an Open Source PCB designer that is well known and used by the engineering student community.[38] Here we have the rooted PCB in (Figure 56).

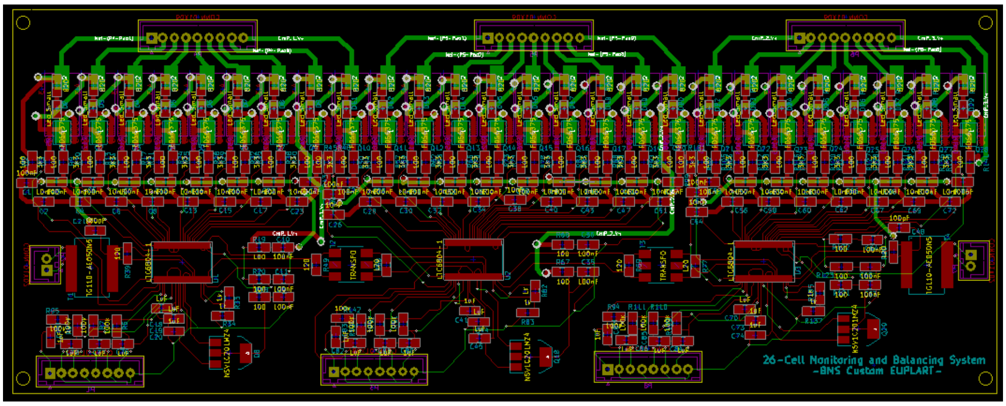


Figure 56. Rooted Slave

A good feature of KiCAD, despite being an Open Source software, is that it has a 3D module included which lets the user see a preliminary 3D view of the PCB he develops. Most professional PCB design software include this feature but they have and immense library of components. KiCAD has a vast library but many are missing and have to be manually included. In (Figures 57 and 58) we can see what the slave PCB will look like.

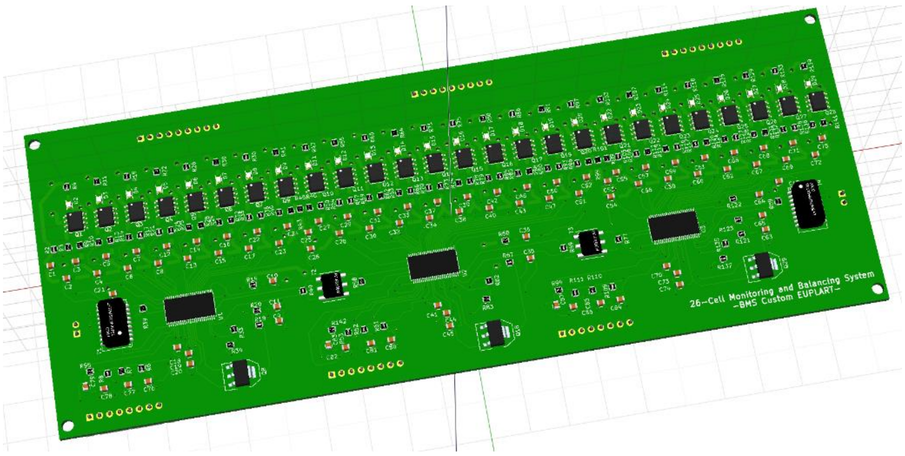


Figure 57. Slave 3D top view

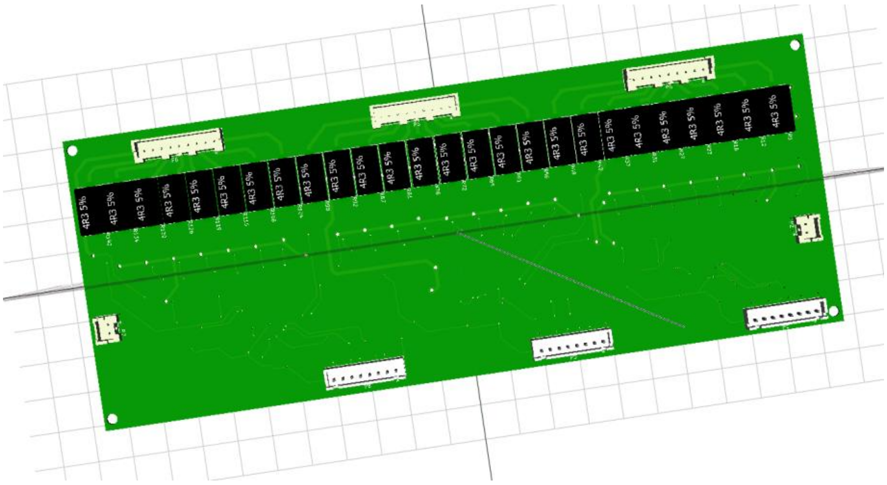


Figure 58. Slave 3D bottom view

5.8. MASTER CHARACTERISTICS.

Provided that we are developing a modular BMS, we need some sort of intelligent element which takes control of the situation and makes the decisions with the data obtained by the slaves. In this case, our master will be made of an Arduino Mega with a shield that another team member will design.

The evaluation kit provided by Linear for LTC6804-1 is made of a modified arduino Uno (Linduino), the IsoBoard and an evaluation board with 12 channels and the chip. The reason for using an arduino Mega instead of using Linduino is because arduino Uno does not have enough capacity to withstand the information of 15 chips as we tried by modifying the number of chips parameter in the LTC6804-1 library.

Using an arduino Mega will allow us to have 4 times more memory and we will be able to withstand 15 chips information and develop a strong and robust software which will prevent any possibility of EMIs by implementing different software filters to avoid them.

The shield that we will use will be the one seen in (Figure 58) and will have the following features:

- 24V power.
- DC-DC 24V-5V to power the arduino.
- CAN Bus communication module.
- 18-Bit Resolution ADC that we will use for current sensing.
- 25A Relay for CH disconnection circuit.

➢ 10A Relay for DH disconnection circuit.

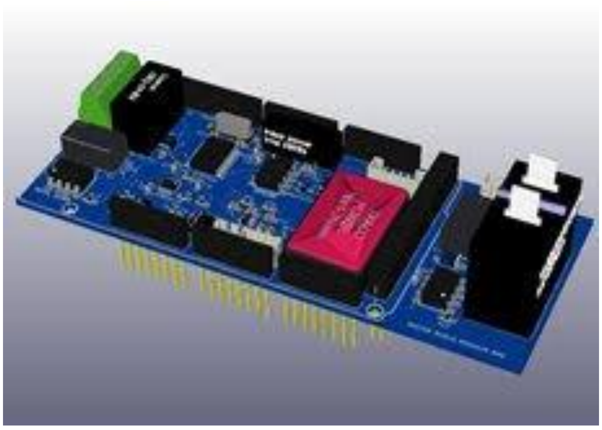


Figure 59. Master BMS Shield

5.9. SOFTWARE DEVELOPMENT.

To be able to start with the software development we acquire a DemoBoard of the chip and we try to learn how the chip works. Linear has code libraries compatible with arduino for all their products and offers an interface for some of them. In this case there is an interface for our DemoBoard. (Figure 60) shows the interface where we can select the number of demo boards used, set parameters such as OV or UV, read both cells and GPIO and we also see what registers we are writing to when setting parameters giving us a clue of a way to interact with the chips.

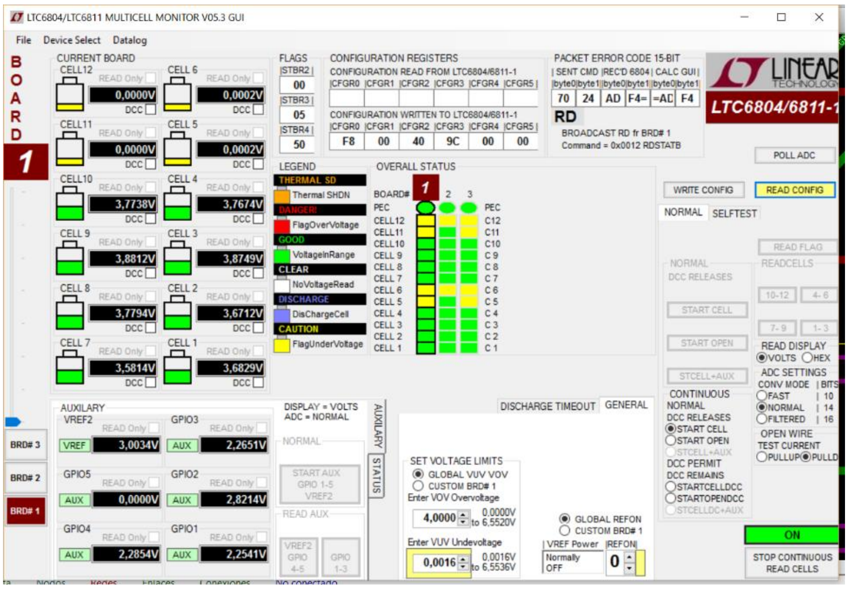


Figure 60.Linear Interface using 3 boards

Once we know how to program the chip and we are able to read cells without the use of the interface we will proceed to develop the code for our BMS.

5.9.1. LTC6804-1 Evaluation Board and Library.

(Figure 61) shows a simulation of what will be our application. In here we see 2 LTC Evaluation Boards with one chip each, the isolation board which decodes de IsoSPI into SPI sending it to the Linduino board which emulates our master. This setup will be used to learn how to use the chip libraries and develop a code for our application. To start with we will use the interface as shown in the previous image to have a high level view of how the chip works.

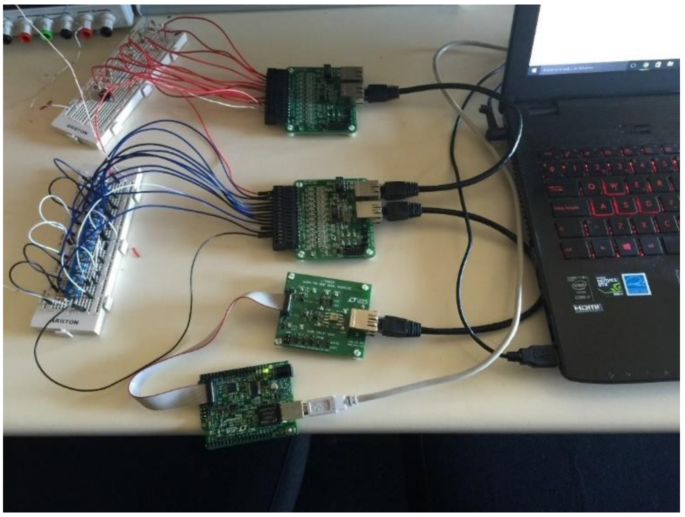


Figure 61.LTC6804-1 Testing Daisy-Chain connection and Cell Readings

The chip library has the following functions, each of them will be explained and related with what they actually do with the chip.

void LTC6804_initialize();

This function sets the SPI clock to 1MHz and calls the set_ADC() function of LTC6804-1 to set the parameters to be read.

void set_adc(uint8_t MD, uint8_t DCP, uint8_t CH, uint8_t CHG);

This function from the library is used to set both ADC reading periods and the amount of channels of both cells and GPIO to be converted. Also sets balancing channels in enable or disable. The configuration is decided as we see in (Figure 62).

Start GPIOs ADC Conversion and Poll Status	ADAX	1	0	MD[1]	MD[0]	1	1	0	0	CHG [2]	CHG [1]	CHG [0]
Start Cell Voltage ADC Conversion and Poll Status	ADCV	0	1	MD[1]	MD[0]	1	1	DCP	0	CH[2]	CH[1]	CH[0]

NAME	DESCRIPTION	VALUES							
MD[1:0]	ADC Mode	MD	ADCOPT(CFGR0[0]) = 0			ADCOPT (CFGR0[0]) = 1			
		01	27kHz Mode (Fast)			14kHz Mode			
		10	7kHz Mode (Normal)			3kHz Mode			
		11	26Hz Mode (Filtered)			2kHz Mode			
DCP	Discharge Permitted	DCP							
		0	Discharge Not Permitted						
		1	Discharge Permitted						
CH[2:0]	Cell Selection for ADC Conversion			Total Conversion Time in the 6 ADC Modes					
		CH		27kHz	14kHz	7kHz	3kHz	2kHz	26Hz
		000	All Cells	1.1ms	1.3ms	2.3ms	3.0ms	4.4ms	201ms
		001	Cell 1 and Cell 7	201µs	230µs	405µs	501µs	754µs	34ms
		010	Cell 2 and Cell 8						
		011	Cell 3 and Cell 9						
		100	Cell 4 and Cell 10						
		101	Cell 5 and Cell 11						
		110	Cell 6 and Cell 12						

Figure 62. Set up Registers LTC6804-1

void LTC6804_adcv(); & void LTC6804_adax();

If we want to set the ADC frequency individually or change it in any moment, we can do it with these functions which modify the MD Registers shown in the table above. In our application the default setup will be used.

uint8_t LTC6804_rdcv(uint8_t reg, uint8_t total_ic, uint16_t cell_codes[][12]);

This is used to read back the cell voltages. In the function first we have to set how many cells we want to read. We can either read all the cells of each IC or a group of cells. "cell_codes[IC][Cell] is where all the values will be stored. The following text extracted from the library (Figure 63) shows the exact configuration.

```
@param[in] uint8_t reg; This controls which cell voltage register is read back.

        0: Read back all Cell registers

        1: Read back cell group A

        2: Read back cell group B

        3: Read back cell group C

        4: Read back cell group D

@param[in] uint8_t total_ic; This is the number of ICs in the daisy chain(-1 only)

@param[out] uint16_t cell_codes[]; An array of the parsed cell codes from lowest to highest. The cell codes will
 be stored in the cell_codes[] array in the following format:
 |  cell_codes[0][0]| cell_codes[0][1] |   [0][2]|    .....     |  cell_codes[0][11]|  cell_codes[1][0] | cell_codes[1][1]|  .....   |
 |------------------|------------------|-------------------|---------------|--------------------|--------------------|------------------|----------|
 |IC1 Cell 1        |IC1 Cell 2        |IC1 Cell 3         |    .....      |  IC1 Cell 12       |IC2 Cell 1          |IC2 Cell 2        | .....    |
```

Figure 63. LTC6804-1 Library Cell reading explanation

To know which these register groups are, we will check the datasheet. (Figure 64) shows register A, the other register groups are made also of 3 cells each. From them we can understand why we receive a 16bit value as the read voltage by the ADC is stored in 16bit registers per cell.

Table 37. Cell Voltage Register Group A

REGISTER	RD/WR	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CVAR0	RD	C1V[7]	C1V[6]	C1V[5]	C1V[4]	C1V[3]	C1V[2]	C1V[1]	C1V[0]
CVAR1	RD	C1V[15]	C1V[14]	C1V[13]	C1V[12]	C1V[11]	C1V[10]	C1V[9]	C1V[8]
CVAR2	RD	C2V[7]	C2V[6]	C2V[5]	C2V[4]	C2V[3]	C2V[2]	C2V[1]	C2V[0]
CVAR3	RD	C2V[15]	C2V[14]	C2V[13]	C2V[12]	C2V[11]	C2V[10]	C2V[9]	C2V[8]
CVAR4	RD	C3V[7]	C3V[6]	C3V[5]	C3V[4]	C3V[3]	C3V[2]	C3V[1]	C3V[0]
CVAR5	RD	C3V[15]	C3V[14]	C3V[13]	C3V[12]	C3V[11]	C3V[10]	C3V[9]	C3V[8]

Figure 64. Cell Voltage Register Group A

int8_t LTC6804_rdaux(uint8_t reg, uint8_t nIC, uint16_t aux_codes[][6]);

This works in the same way as the one above but it returns the values stored in the auxiliary ADC. In this case the read information is stored in aux_codes[IC][GPIO or VREF] (Figure 65) shows how the setup must be done to receive all the data or part of the data depending on the applications. In our case we need to read back all the registers because we use 4 GPIO.

```
@param[in] uint8_t reg; This controls which GPIO voltage register is read back.

        0: Read back all auxiliary registers

        1: Read back auxiliary group A

        2: Read back auxiliary group B

@param[in] uint8_t total_ic; This is the number of ICs in the daisy chain(-1 only)

 @param[out] uint16_t aux_codes[][6]; A two dimensional array of the gpio voltage codes. The GPIO codes will
 be stored in the aux_codes[][6] array in the following format:
 |  aux_codes[0][0]| aux_codes[0][1] |  aux_codes[0][2]|  aux_codes[0][3]|  aux_codes[0][4]|  aux_codes[0][5]| aux_codes[1][0] |aux_codes[1][1]|  .....    |
 |-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------------|-----------|
 |IC1 GPIO1        |IC1 GPIO2        |IC1 GPIO3        |IC1 GPIO4        |IC1 GPIO5        |IC1 Vref2        |IC2 GPIO1        |IC2 GPIO2      |  .....    |
```

Figure 65. Library Auxiliary reading explanation

(Figure 66) shows what is stored in each register and we see that group B stores Vref which is the voltage used to power up the NTC we have in our board. This can help us adjust an accurate parametrization of them as we will now the exact voltage supply.

Table 41. Auxiliary Register Group A

REGISTER	RD/WR	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
AVAR0	RD	G1V[7]	G1V[6]	G1V[5]	G1V[4]	G1V[3]	G1V[2]	G1V[1]	G1V[0]
AVAR1	RD	G1V[15]	G1V[14]	G1V[13]	G1V[12]	G1V[11]	G1V[10]	G1V[9]	G1V[8]
AVAR2	RD	G2V[7]	G2V[6]	G2V[5]	G2V[4]	G2V[3]	G2V[2]	G2V[1]	G2V[0]
AVAR3	RD	G2V[15]	G2V[14]	G2V[13]	G2V[12]	G2V[11]	G2V[10]	G2V[9]	G2V[8]
AVAR4	RD	G3V[7]	G3V[6]	G3V[5]	G3V[4]	G3V[3]	G3V[2]	G3V[1]	G3V[0]
AVAR5	RD	G3V[15]	G3V[14]	G3V[13]	G3V[12]	G3V[11]	G3V[10]	G3V[9]	G3V[8]

Table 42. Auxiliary Register Group B

REGISTER	RD/WR	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
AVBR0	RD	G4V[7]	G4V[6]	G4V[5]	G4V[4]	G4V[3]	G4V[2]	G4V[1]	G4V[0]
AVBR1	RD	G4V[15]	G4V[14]	G4V[13]	G4V[12]	G4V[11]	G4V[10]	G4V[9]	G4V[8]
AVBR2	RD	G5V[7]	G5V[6]	G5V[5]	G5V[4]	G5V[3]	G5V[2]	G5V[1]	G5V[0]
AVBR3	RD	G5V[15]	G5V[14]	G5V[13]	G5V[12]	G5V[11]	G5V[10]	G5V[9]	G5V[8]
AVBR4	RD	REF[7]	REF[6]	REF[5]	REF[4]	REF[3]	REF[2]	REF[1]	REF[0]
AVBR5	RD	REF[15]	REF[14]	REF[13]	REF[12]	REF[11]	REF[10]	REF[9]	REF[8]

Figure 66. Auxiliary register groups

void LTC6804_clrcell(); & void LTC6804_clraux();

These functions are used to clear both Cell and Auxiliary registers and set them to 0xFF. Our application does not need them.

void LTC6804_wrcfg(uint8_t nIC,uint8_t config[][6]); & int8_t LTC6804_rdcfg(uint8_t nIC, uint8_t r_config[][8]);

These functions are very important because they are used to set the parameters of the chip and also are used to enable balance when we want to balance a cell. To do so we will use CFGR4 & CFGR5 as the DCC bits when they are set to 1 balance is enabled for that cell. In our application OV and UV will be defined in the master board and not in the chips as we are giving all the intelligence to the master.(Figure 67) shows what can be set in each configuration register and which bit corresponds to each variable.

Table 36. Configuration Register Group

REGISTER	RD/WR	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CFGR0	RD/WR	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	REFON	SWTRD	ADCOPT
CFGR1	RD/WR	VUV[7]	VUV[6]	VUV[5]	VUV[4]	VUV[3]	VUV[2]	VUV[1]	VUV[0]
CFGR2	RD/WR	VOV[3]	VOV[2]	VOV[1]	VOV[0]	VUV[11]	VUV[10]	VUV[9]	VUV[8]
CFGR3	RD/WR	VOV[11]	VOV[10]	VOV[9]	VOV[8]	VOV[7]	VOV[6]	VOV[5]	VOV[4]
CFGR4	RD/WR	DCC8	DCC7	DCC6	DCC5	DCC4	DCC3	DCC2	DCC1
CFGR5	RD/WR	DCTO[3]	DCTO[2]	DCTO[1]	DCTO[0]	DCC12	DCC11	DCC10	DCC9

Figure 67.LTC6804-1 Configuration registers

5.9.2. BMS Software.

5.9.2.1. General UML

This flow diagram shows the general working method of our battery management system. The software ensures CANBus communication is enabled and checks if any command has been sent from the serial interface. After and external signal defines if the battery is being charged or is in discharge state. Depending on which state the security procedure is different. For security while charging the discharging contactor are open to avoid any possible current flow to the rest of the EV.

The first thing that will be checked will be the charging failure values to stop charging if needed. Once it has been done if the interface is enable the battery pack information will be sent to the interface. Now the battery balancing procedure starts we enter in a case selection where we check if a high temperature flag has stopped balancing but reading is still available until normal values are restored and balancing is available again. The balancing procedure will be the following. Read→Balance for 1 second→No balancing for 1 second→Read again. Balancing has to be stopped to be able to read correct values as reading while balancing will result in reading a 4.3Ω in parallel with the cell.

Discharging state reads the values then calculates maximum, average and minimum voltage + temperatures as important data to be stored. Then average current is calculated and with all the data we enter an over watch which checks the values and compares them to the maximum values admitted if any values are overpassed contactors are opened and the system disconnected from the battery pack.

The progress finished calculating the consumed power to know the SOC and all data is sent to the CAN bus.

SOC calculations are done with a simple algorithm:

Consumed Power in one second is = Total voltage · Average Current in on second

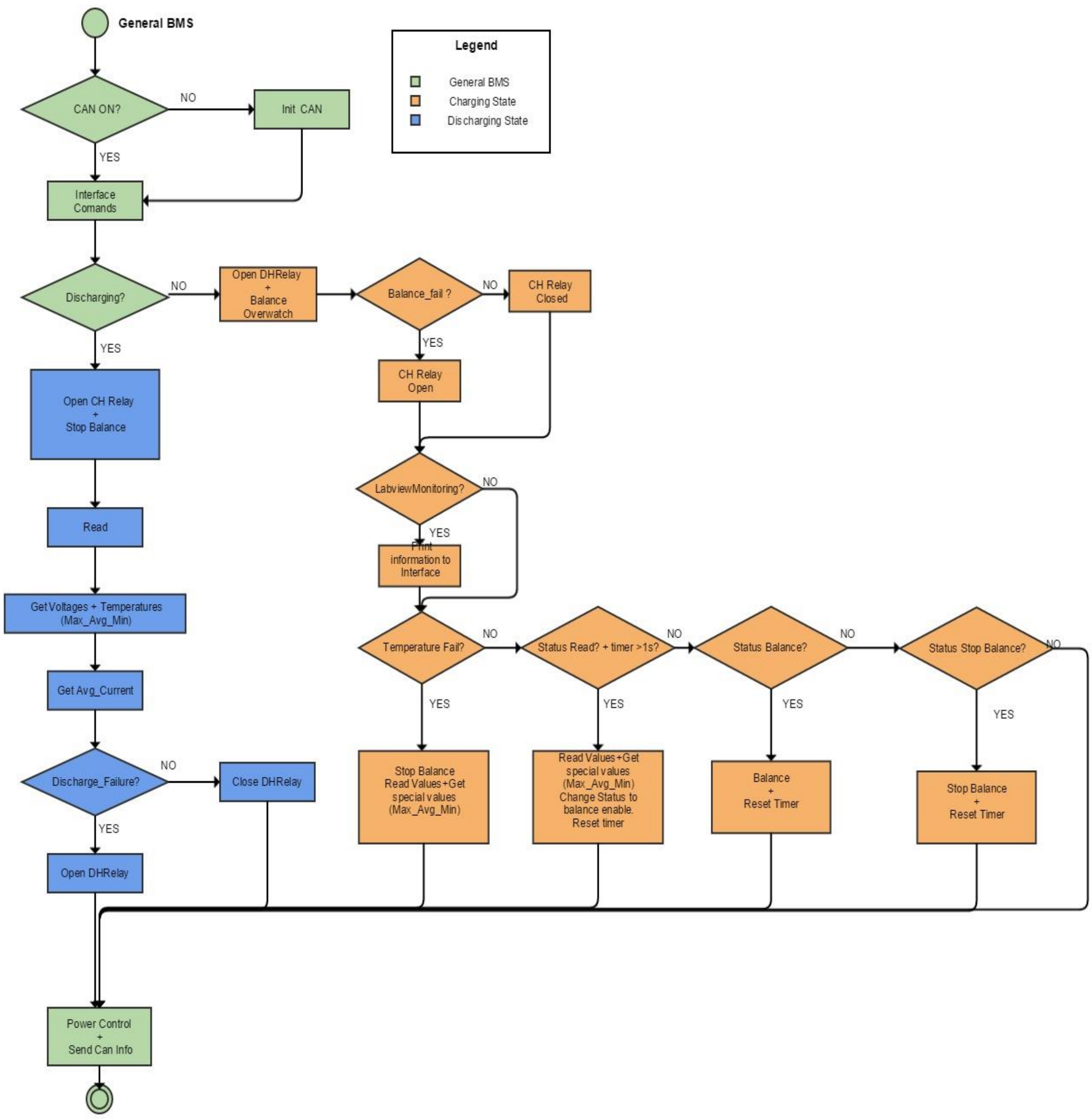
From the above relationship we obtain Jules by multiplying by 0.000277778 we obtain W·h as we know the total W·h our battery pack offers at full charge.

Power Left=Power Available - Consumed power.

Knowing that the total power of our battery pack fully charged is 109.2V*10Ah so 5460Wh we calculate SOC like this:

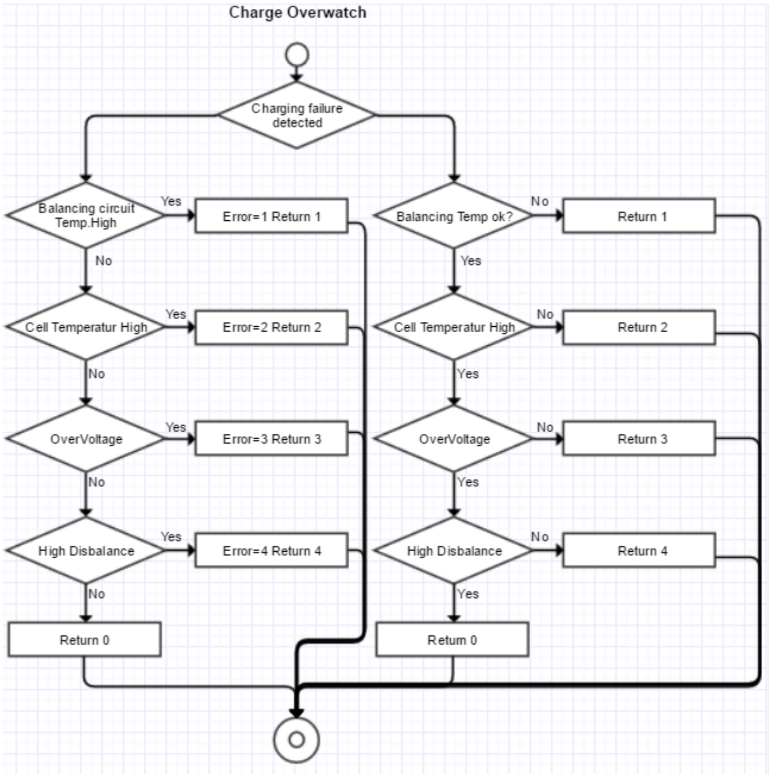
$$SOC\ (\%) = \frac{Power\ Left\ (Wh)}{5460\ (Wh)} \cdot 100$$

Autor: **Xavier Anzuela Recasens**
424.16.96

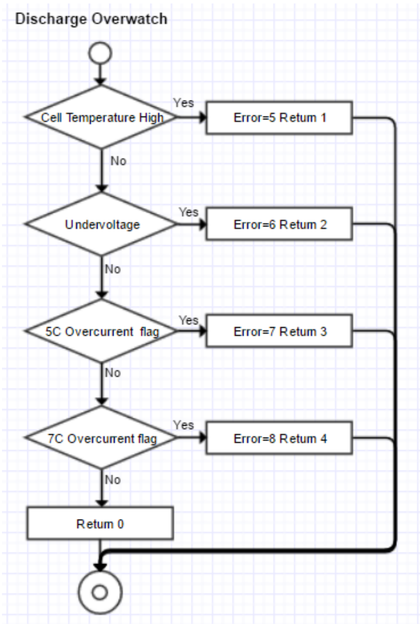


5.9.2.2. OverWatch Functions

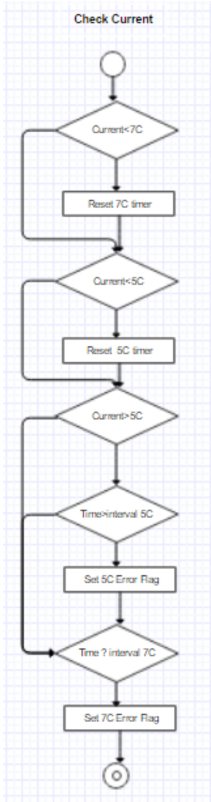
Charging: We over watch Balancing temperature, cells temperature and overvoltage also if any of this errors occur we have a set of reset values to enable charge again these values are defined as secure values for continuing with the charging operation. To avoid any EMI intervention, we set a filter if max voltage is more than 4.4V we consider that noise has affected the reading and no security is applied. We use that filter knowing that when a cell reaches 4.25V, the system will stop charging and we will never reach 4.4V.



Discharging: This function checks that the cell temperature does not reach the critical value, Undervoltage protection and checks if any current protection flag has been activated. In this case, no reset values are set as the EV has to be checked for security. Only a manual reset can restart the bike is any of these errors occur.

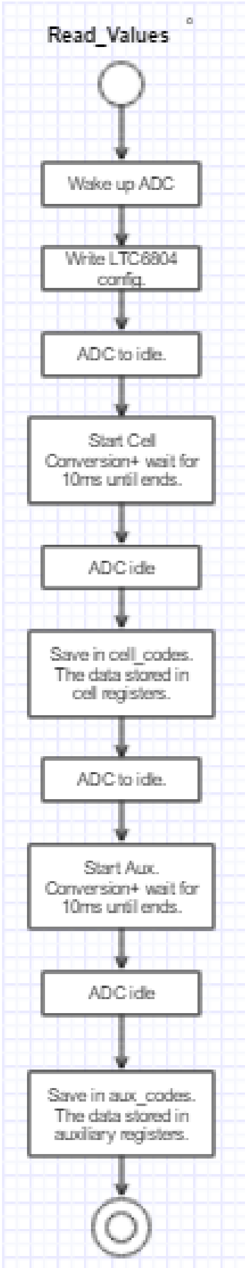


Check Current: We have two current timers one for 5C and the other for 7C. We know that our cells can only be discharged for 2s at 7C and that discharging at a higher density of 5C is not good for cells we set a maximum time of 5s discharging at 5C to protect the cells from heating and a quicker degradation. While current is under these values both timers are continuously reset to 0. If any of them are not reset before the interval an error flag is sent and contactors are opened by the discharge overwatch function.

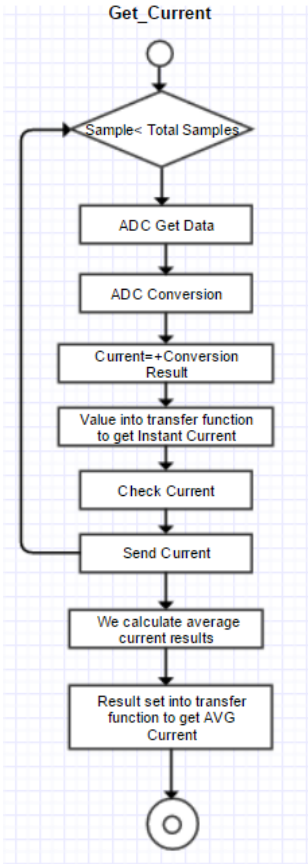


5.9.2.3. Reading Functions

Get Voltage and Auxiliary Readings: This function continuously broadcast to all chips in the daisy chain to obtain the readings from both ADC. It results in the filling of two matrix variables which store the data of each chip. The resultant voltages obtained from the ADC will have the following scale: 1V=10.000units.

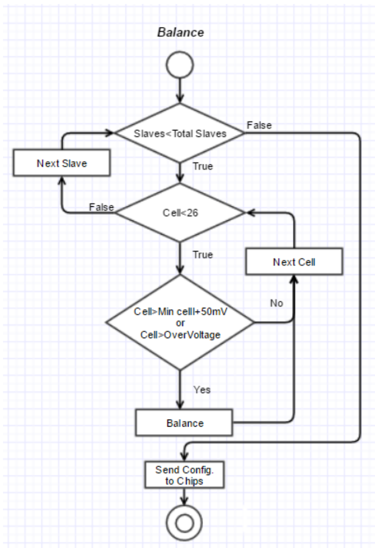


Get Current: This function is used to get both Average Current and Instantaneous current. We adjust the function loop to take 1s so that we obtain an average current for 1s and have an easy way to calculate power consumption.

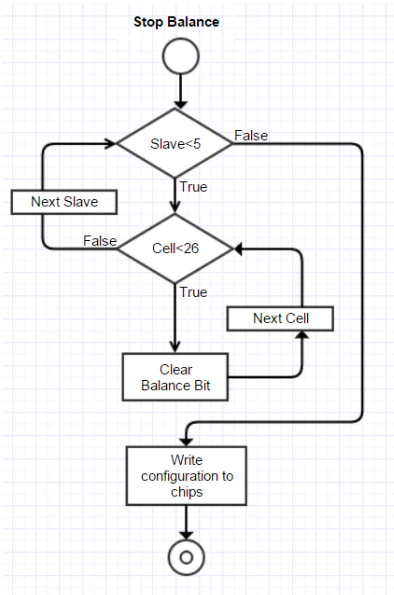


5.9.2.4. *Balancing Functions*

Balance: This function checks all the cells slave by slave looking for cells, which have a higher voltage than the minimum cell + an offset of 50mV, which after many test was selected as the optimum balancing, offset. Also checks if any cell is over a certain voltage we define as over voltage balancing which is 4.1V so for the last 100mV, which are in the quick charging, part of the LiPo curve all cells are balanced until they reach 4.20V.



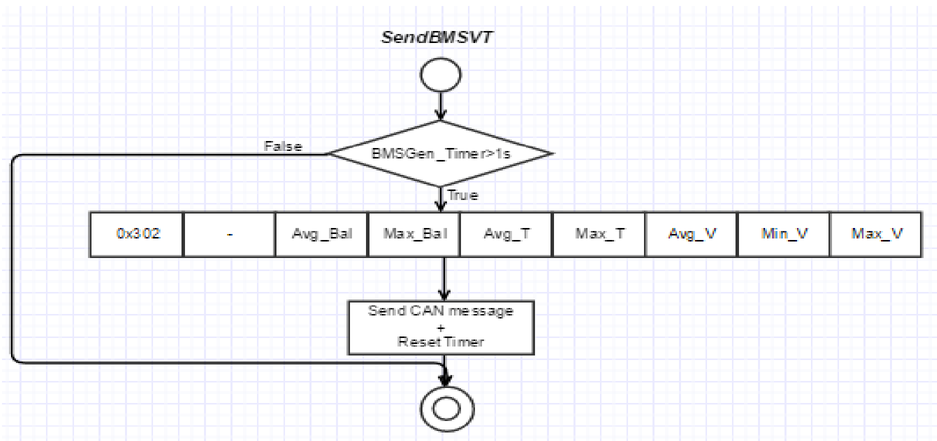
Stop Balance: This function checks each cell in each slave and sets the cell balancing register to 0 which disables balance. Once all the registers have been set to 0 the information is sent to the daisy-chain to be written into all the chips.



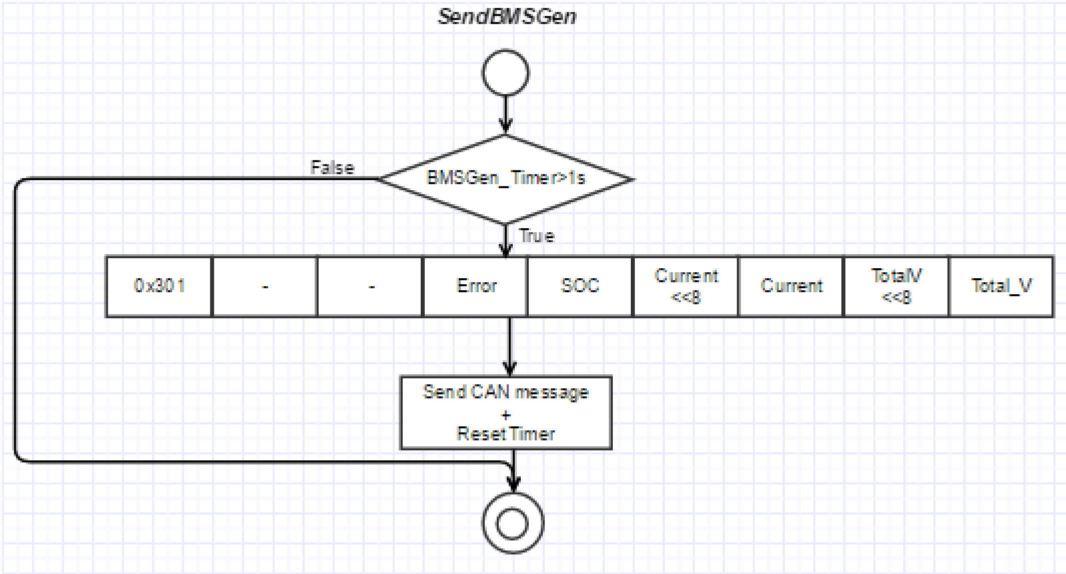
5.9.2.5. *CAN Bus Functions*

The functions below have the same working procedure, a set of values are inserted into an 8 bit array and the information is sent by CAN to the ECU. The data to be sent is defined by the team needs. The time when each message is sent is adapted to the capacities of the ECU. For instance, the full battery pack information was disabled as the ECUs processor had too much information coming from all the subsystems and smaller messages where sent with critical values as maximum voltage or Minimum voltage.

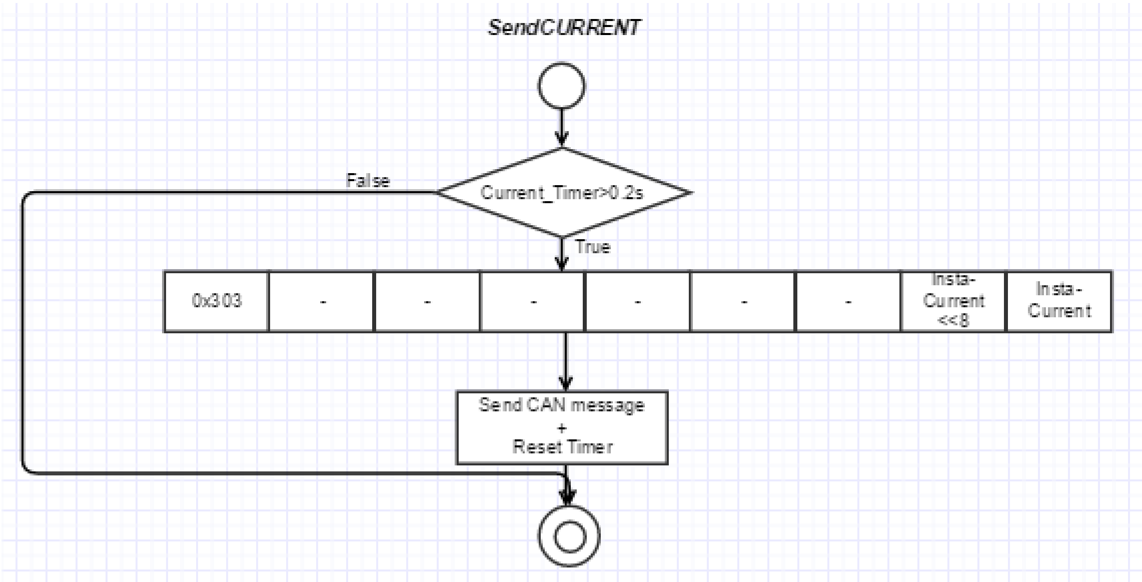
BMS Voltage-Temperature:



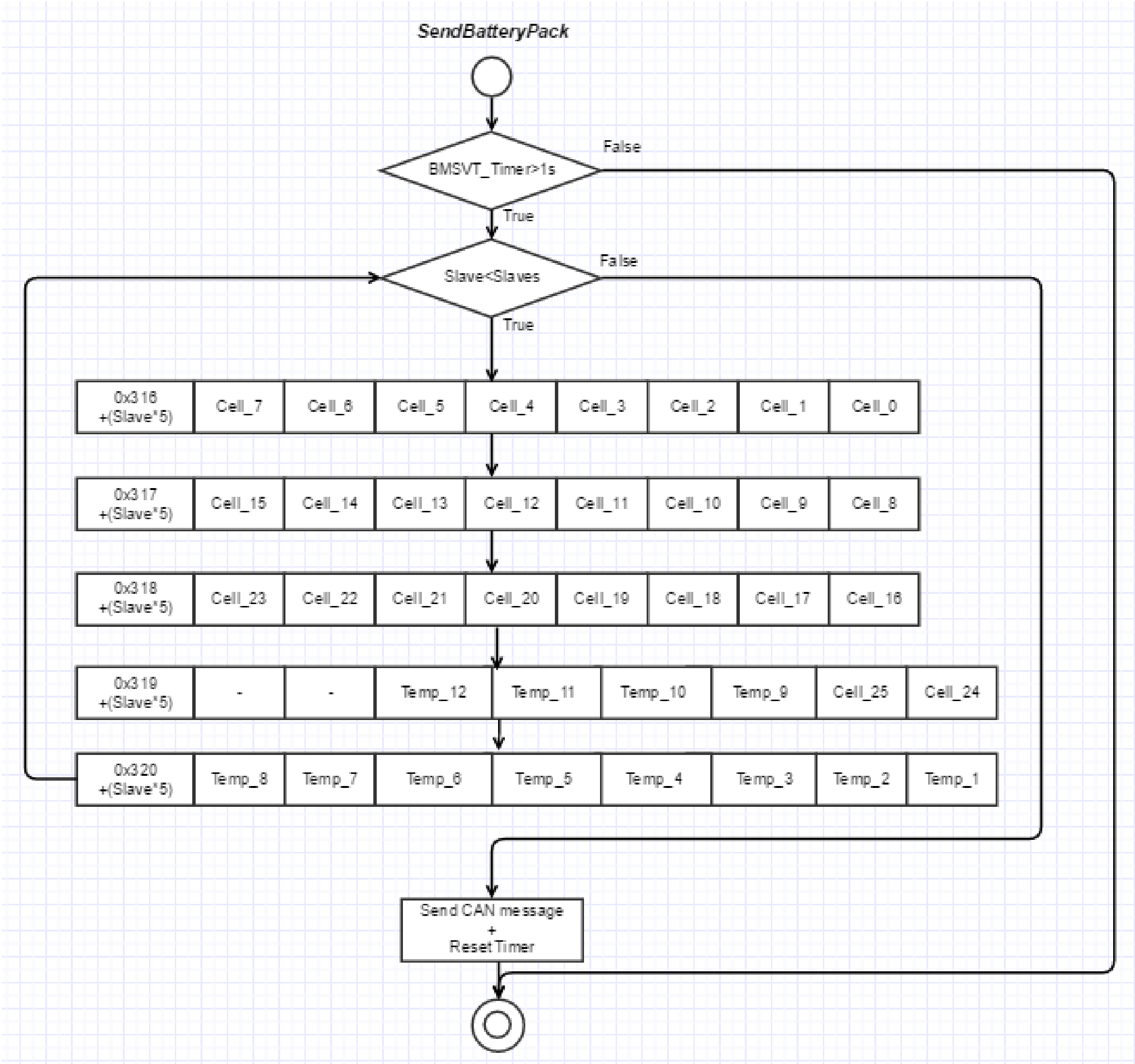
BMS General Values:



BMS Instant Current:



BMS Total Data:



Autor: **Xavier Anzuela Recasens**
424.16.96

5.10. LabVIEW Interface.

The goal of this software development is to obtain a stable and user-friendly interface to be able to test the Custom BMS designed boards in order to be able to avoid using the LINEAR software designed to test their demo boards. While the Custom BMS is composed by 3 chips and we have a total of 5 slaves, and the demo software only admits 10 chips at time and gives information about all the 12 monitoring channels of the chip. The idea is to create an interface which resembles our battery pack, giving a general status of the security maneuver and the status of each of the different stacks.

The software will be very useful to test the efficiency of the boards and also to check their reliability. LabVIEW is a very useful visual programmer which enables to make user-friendly interfaces.

As all the boards are connected via IsoSPI with the master the connection to LabVIEW will be accomplished by this method.

The interface will have the following features:

-Cells state for each Substack, both visual state (color tank) and Voltage Value.

-Temperature reading.

-Balancing enabling/disabling method for each cell.

-Total Voltage for each Substack.

-Average Voltage for each Substack.

-General Battery Pack Status, Substack Read, Contactors State, Voltage, Security Mode, State(CH/DCH), LabVIEW enable/disable.

5.10.1. Development.

5.10.1.1. Interface requirements:

Since we are working in a verification interface, this will need to be able to check that every single module of the BMS is well functioning. To do this firstly the software has to be developed, a software which sends to the serial port the following information: Cell Voltage, GPIO Voltage, State of all the relays, Current, Total Voltage per Substack.

Once this software is done, we decide how to process the information in LabVIEW:

We will define cell states with a color code: Overvoltage (red), Undervoltage (yellow), Balancing (Blue), Good condition (green). Cells will be represented as tanks which fill as voltage grows; Cell voltage will be shown in a numerical indicator to show exact voltage. The LTC6804-1 library returns voltage in the following way 40.000=4V so we will need to make the conversion in LabVIEW. Our system is made up of 26 cells per substack so we will have to take that into account when making the interface. Balancing interrupters or buttons must be added to be able to check each balancing channel is working. A total voltage indicator will be placed in each substack.

Our BMS uses GPIO pins to sense temperature with NTCs so once we receive the voltage we will need to use the NTC transfer function to convert it to ºC. A good representation for temperature can be a thermometer. Our system is composed by 12 sensors where sensor 8 is the balancing circuit temperature. The measuring ratio must be adjusted as 11 sensors will only reach a maximum of 65ºC (Max. Temp. admitted by cells) and the balancing circuit will go to 180ºC where unsoldering risk appears.

The general state must show the state of each one of the relays and it can also show which substack is read each time to ensure that the software keeps reading all the boards information and does not stay in one or avoids any. It may also show a graph of the total voltage evolution which may be used to characterize the battery and another graph may show the current consumption to check that the security maneuver opens contactors when overcurrent occurs.

5.10.1.2. Communication setup.

The communication method chosen is SPI so we will need to find the VISA module in LabVIEW which offers the possibility to read serial data from an external device.

The VISA module is very simple it can be used both to either READ from the Serial port or send information to the Serial port.(Figure 68)

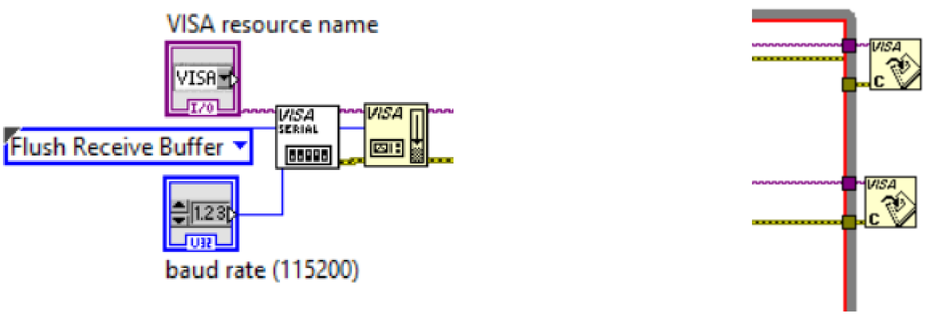


Figure 68. LabVIEW VISA resource

VISA module needs a start connection and an end connection. These connections are usually outside the main loop given that they are just the Serial Setup, the left image shows the modules where you select the port and the baud rate. Also a flush has been added to clear the SPI each time the program starts to avoid residual information that may destabilize the application. The end is just the output if needed.

To read or write the serial information we need the following created SubVI part of the VISA module which can be seen in (Figure 69).

Figure 69. Visa subVI

5.10.1.2.1. Read Setup.

The reading setup needs a "scan from string" function which feeds on the information received by the VISA Read block, the VISA Read will read as many bytes it receives at a time as long as they are equal to or lower than the number of bytes you set it to read. (Figure 70) shows our setup to read the serial port.

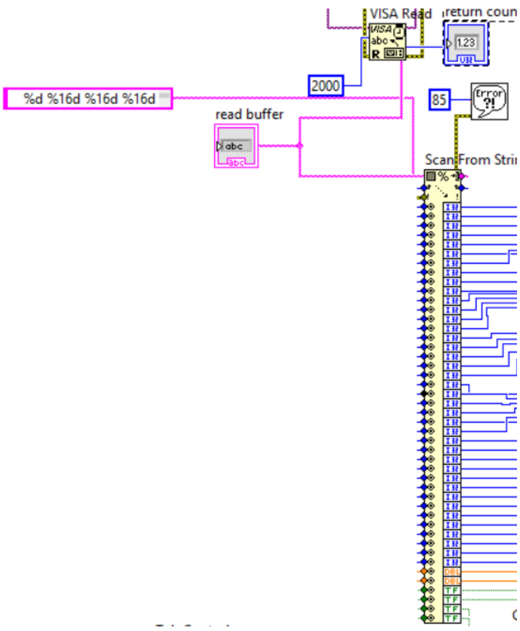


Figure 70.Interface Serial port reading setup

The scan from string will divide the strings sent into different variables, to set how the information is read you need to tell the scan what type of variables to look for and how they are sent for instance our software sends each parameter spaced so that with the read buffer function we can verify that the information received is correct.

To do this we double click the block and this window appears:

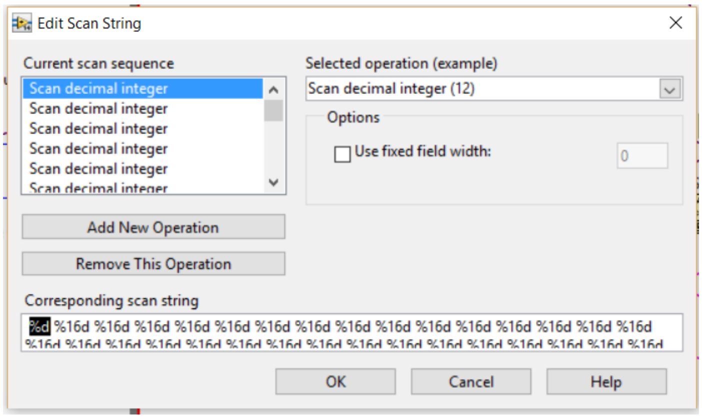


Figure 71.Scan string window

Here we state which type of variable we read: float, integer, double integer, Boolean.

It is very important to choose the correct variable as further interactions with it will be considering its type, for instance, if you set a Boolean as an integer when you connect a LED indicator to it, it will not work as it will read 1 or 0 as numbers but instead as TRUE/FALSE statements.

5.10.1.2.2. Write Setup.

The write setup is very easy. We just need to connect a string to the VISA Write, as only one string variable can be connected if different information or messages have to be sent. The solution is to create a global variable (Figure 72) where we can save the information to be sent.

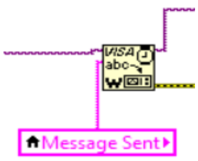


Figure 72.Write to serial port SubVI with global variable

5.10.1.3. Distribution.

As we are creating an interface we need to optimize the space we have and try to fit it all into a single computer screen because not all users will be able to have multiple screens and it can be tedious to keep moving up and down the screen in a testing and verifying app.

5.10.1.3.1. Tabs.

The needs for the interface are as follow, General State of the Battery plus each substack cells and temperature stare. A good answer to how to show up all this information would be Tabs, creating a Tab based interface will make a smoother layout and an easy way to switch between information modules. To do this LabVIEW has a block called tabs which allows to create and infinite number of tabs(Figure 73), name them and by just dragging a group of things into the Tab they become part of it and can also be easily removed or moved into another tab just as easily as they were inserted.

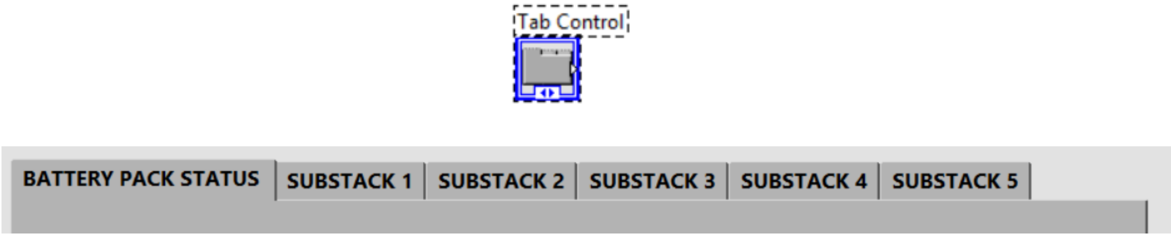


Figure 73. LabVIEW Tab option

5.10.1.3.2. Case Structure.

As we have 130 cells and 60 temperature sensors we have many options to get this information. One will be having all the information sent at a time and read a massive buffer, but as we have 5 substacks we can take advantage of the event case structure, which can be seen in (Figure 74). This allows us to divide information received into different instances depending on the control variable. In our case the control variable will be "Substack" and we will send each time, which substack we are reading and save the information in the correct instance.

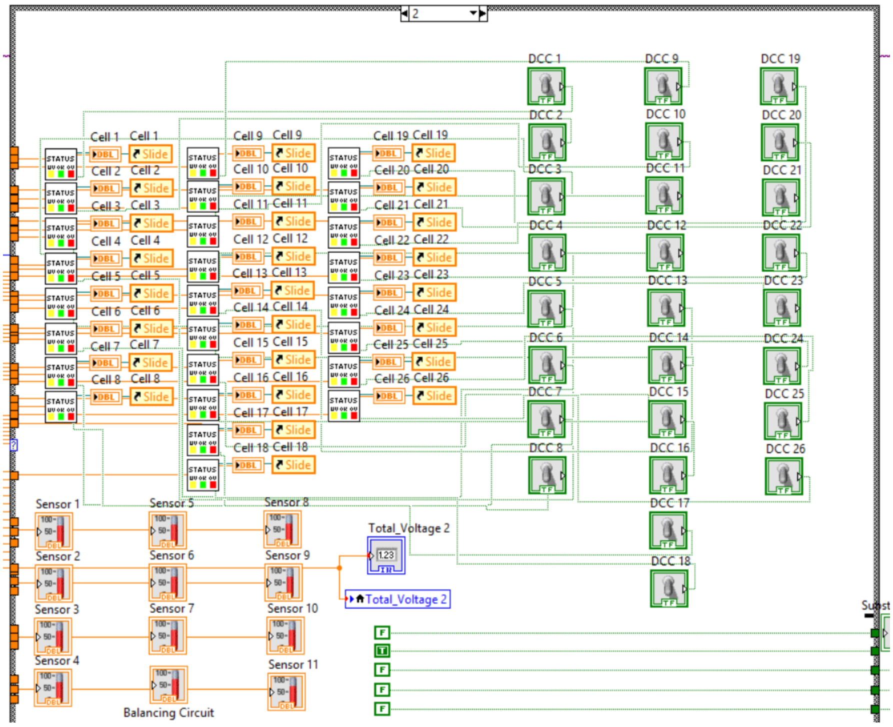


Figure 74. Interface "Substack" case structure resource.

To be able to track which substack we are reading, we have inserted 5 states, one per substack, which are connected to an external LED. The problem with the case structure is that if some information goes out of the case in each case there must be information sent through that output, that is why we create the 5 states and in each instance a different one is true and all are sent out to different LEDS.

As we see above, each case includes a 26 cell reading, balancing activators for each cell, temperature reading, total voltage indicator and the substack read indicator. General pack information is not part of any case as it must be constantly updated so it is part of the main loop.

5.10.1.3.3. Event Structure.

Only one string can be sent with the VISA write so we need a way to update the value of the string for it we can use the event structure (see Figure 75) in which you select an event; for instance, the use of a switch and you create a small function to be done each time that happens. We will use this to activate/deactivate balance for each cell and to change between modes. The verification mode is one of the modes of the main code, to avoid changing codes to verify the boards we create inside the main code a LabVIEW mode and a security mode which is used to stop balance at all times and avoid the boards C-MOSS to become unstable at some point. The security mode appears when we realize the MOSS used is so sensible that it spontaneously activates, so to verify the board and avoid instability we create this mode. In the final code this mode is removed as boards are stable.

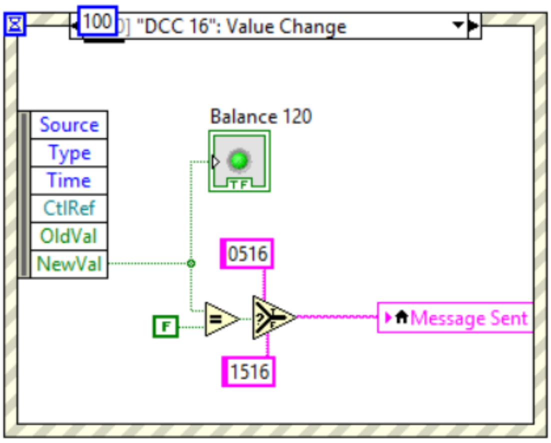


Figure 75. Event Case for balancing enabling

We use a condition block so that we can differentiate when we enable or disable balance and in each case a 4-digit string is sent. The first Number indicates Mode 0 for stop balance, 1 for balance, 2 for security and 3 for LabVIEW.

<u>0 and 1: </u>the next 3 digits define substack and cell being (MODE, SUBSTACK, and CELL) in a two-digit format. We also added a LED to show balancing to imitate the board.

Example 1: 0114→Stop Balance Substack 1 Cell 14.

Example 2: 1403->Balance Substack 4 Cell 3.

<u>2 and 3:</u> the forth digit indicates on or off being 1 on and 0 off. We use a switch for these two cases.

Example 1: 3001→Start LabVIEW mode.

Example 2: 3000→Stop LabVIEW mode.

Serial printing consumes microprocessor time and is not needed for the actual application. That is why we create the LabVIEW mode to only print information while we are in testing mode.

5.10.1.4. Creating SubVI.

LabVIEW permits to unify a block of code and create a SubVI which can be used multiple times. It would be the same as a function in written code languages. To do so you have to select the block which you want to make into a VI then click "EDIT" tab on the top left corner and select "create a new SubVI". To insert created SubVI into the code, click on tools tab and then select "Find created Vi".

For this project we have created 3 SubVI:

-Color Changing Tanks.

-Voltage Conversion.

-Temperature Conversion.

The VISA SETUP may have been mashed into a SubVI but it resulted into some problems so it was decided to undo it.

SubVI are blocks with inputs and outputs. These are limited by the patterns given by LabVIEW so you cannot create a 1000 input VI, the maximum is set on 20 inputs or outputs pattern. SubVI may be made of other SubVI to have more complex blocks.

5.10.1.4.1. Color Changing Tanks.

We stated the need of a color changing tank (Figure 76) to show the state of each cell, 3 colors indicate the voltage condition of the cell: Correct, Over, Under. Four colors indicate balancing mode. We used a logical TRUE/FALSE statement to give priority to each color first we created the LOWER/UPPER limits and then we set the Balance condition coming from a switch. Voltage and the switch are the inputs in this case and the output is the "Cell" which is the tank and the slide with parameter modifier that changes the color of the slide.

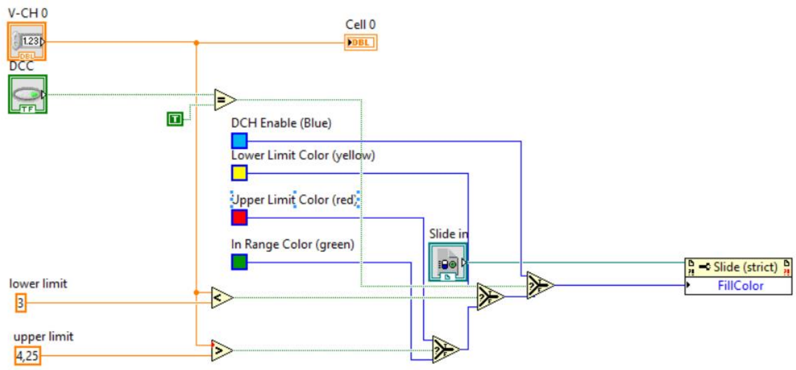


Figure 76. Change tank color SubVI

5.10.1.4.2. Voltage Conversion.

The software in the board returns the voltage in as follows 40.000→4V so we need to make an operation where we divide the value by 10^4 to get Volts. The SubVI shown in (Figure 77) will convert the integer value from the input into a float as an output. The VI has 8 Channels so that we can convert 8 values at a time. Not all the channels need to be used each time but it gives higher flexibility for future applications.

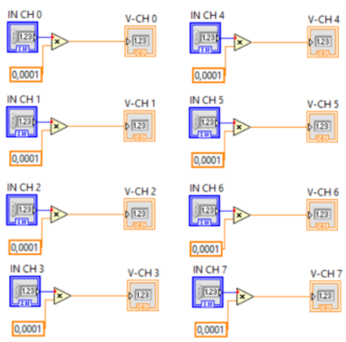


Figure 77. Voltage conversion SubVI

5.10.1.4.3. Temperature Conversion.

The GPIO pins are read in volts as they are general purpose inputs/outputs in our case we use them as inputs for NTC sensors to measure temperature in selected cells. We previously calculated the transfer function for the sensors and we proceed to make an intelligent block where voltage is the input and we get temperature in ºC as an output.

The transfer function is as follows:

$$f(v)=-32{,}59\ln(v)+37{,}404$$

As we are using 4 GPIO per chip we create a 4 input/output temperature converter SubVI.

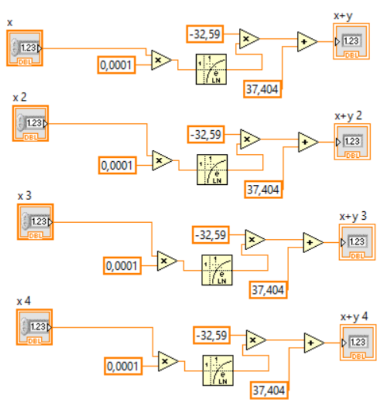


Figure 78. Temperature conversion SubVI

5.10.1.5. General Status.

The general state, which we can see in (Figure 79) simply shows the state of the Boolean variables of the software, such as the read substack, battery status (CH/DCH), status of the contactors, charging relay status, TSMS (ON/OFF) and if the Charge Bomb is ready to rearm the contactors. Also the VISA configuration is done from this tab and the STOP button to stop the loop is here.

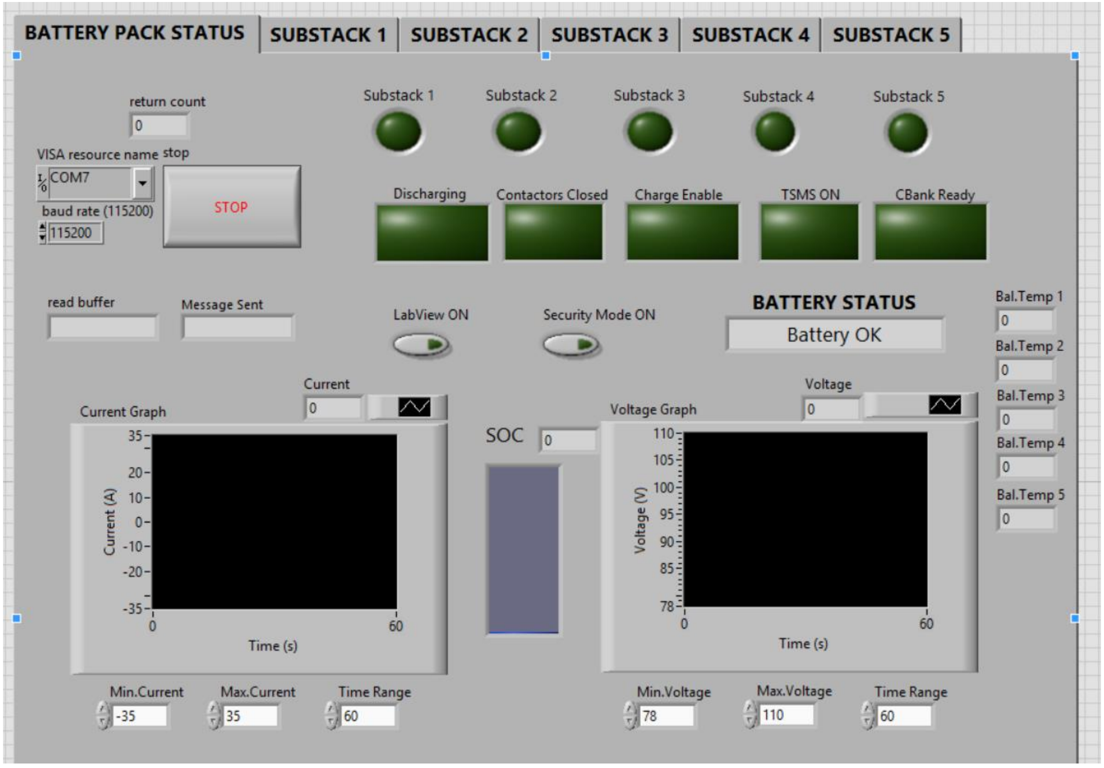


Figure 79. General Status of the battery pack view

The STOP button is a must when we work in a loop as it is the only way to softly stop the program the other way is interrupting it which may lead to errors in the LabVIEW app when trying to restart it.

Two graphs are added to show current rate and total Voltage though time allowing the possibility to analyze a substack or the full battery pack, the axis of this graphs are variable so that can be used for different tests and can easily adapt to the needed reading precision. The State of Charge will also be shown; a tank has been chosen to represent it as it resembles to the typical cell illustration in electronic devices.

A battery status box is added to show what causes any battery failure, the default value of the box reads "Battery OK". This box is used mainly for testing purposes of the security maneuver to ensure everything works by using more restrictive values and forcing a security shutdown. This feature is basically a case structure, where the input is the error variable send by the master when any failure occurs, and a string showing which failure happened for each case.(Figure 80)

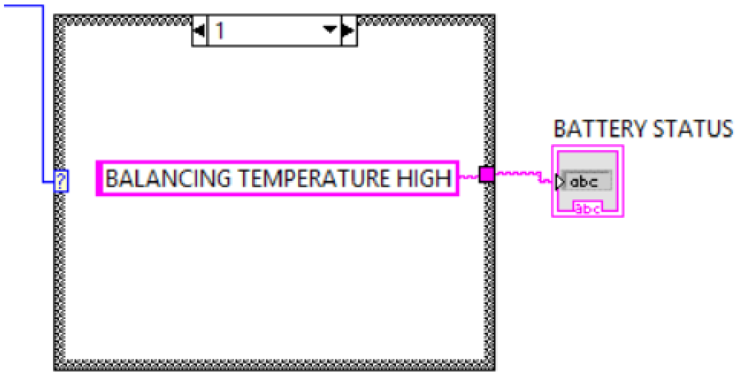


Figure 80.Error Flag advise

After many tests we realize that the cells do not heat with the current fed by our charger we decide to add a state variable of the balancing circuit temperature of each substack (Figure 81). These are added, avoid keep changing tabs while charging and keep control of which circuit needs refrigeration. To do so, we create a global variable of each balancing circuit and we print each of them in the main tab. We have to create a variable because we want to keep the information of the balancing circuit in each substack tab. As the input from the master does not give temperature and we converted in a SubVI to avoid using that SubVI again we create the variable from its output.

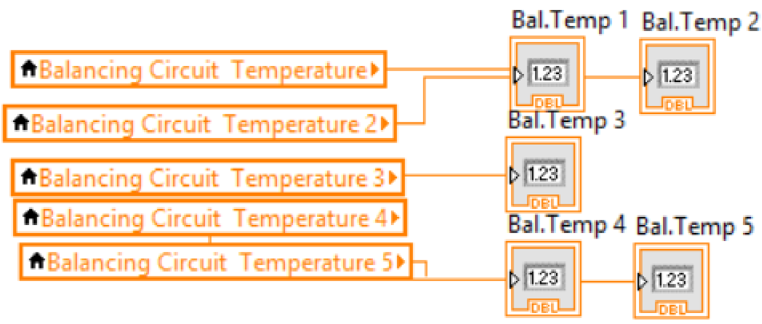


Figure 81.Balancing temperature display.

5.10.1.5.1. Graph axis modifier

Depending on the test which wants to be done, we may want to modify the time displayed, the voltage or the current limits. To do this we need to right click the object we want to access its attributes and create a property node. The process will be shown in (Figure 82).

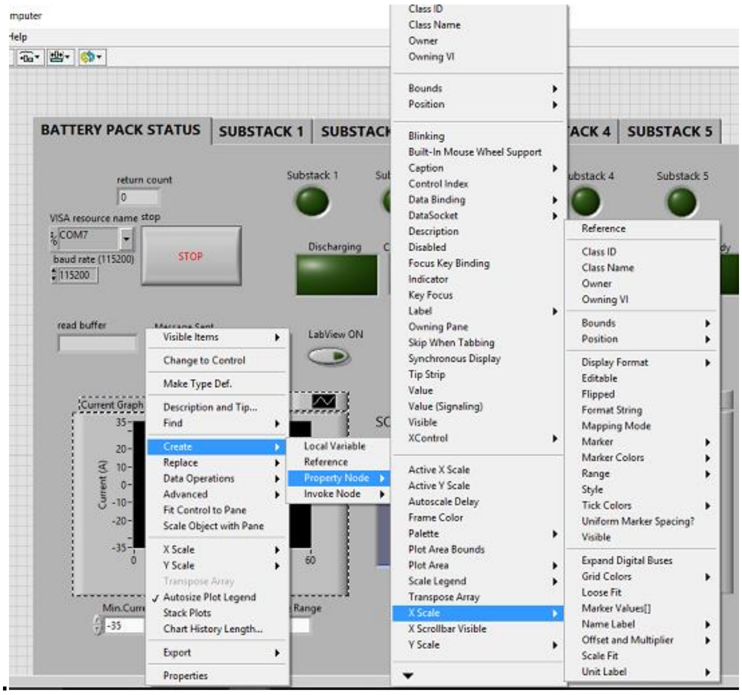


Figure 82.Property node creation

We create the following property nodes(Figure 83):

- Maximum Voltage
- Minimum Voltage
- Maximum Current
- Minimum Current
- Time Range for both current and voltage.

*Note that when the property node is created is in read state.

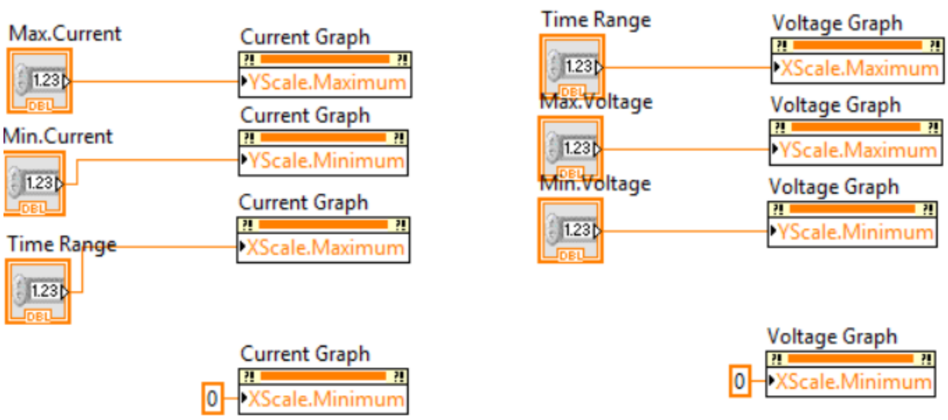


Figure 83.Graph node settings.

5.10.1.6. *Application Builder.*

LabVIEW has the option to convert your VI into an actual application. To do so; the user goes to tools and selects "build application EXE" as shown in (Figure 84).

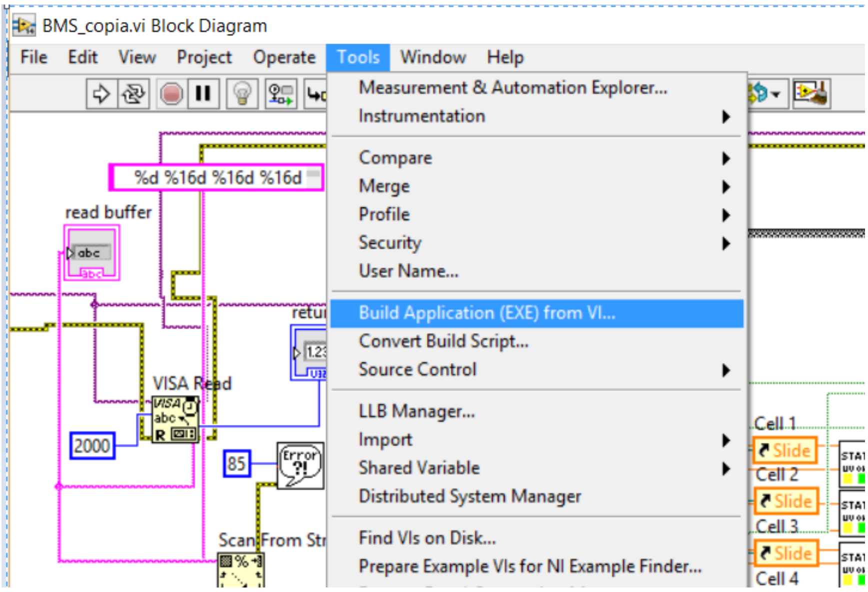


Figure 84. Application builder option.

Once clicked a new window appears in which you can give name to the application and enter the data describing it or even create an icon for it. This gives a professional output to a working project made in LabVIEW and it can be used without the need of LabVIEW installed. (Figure 85)

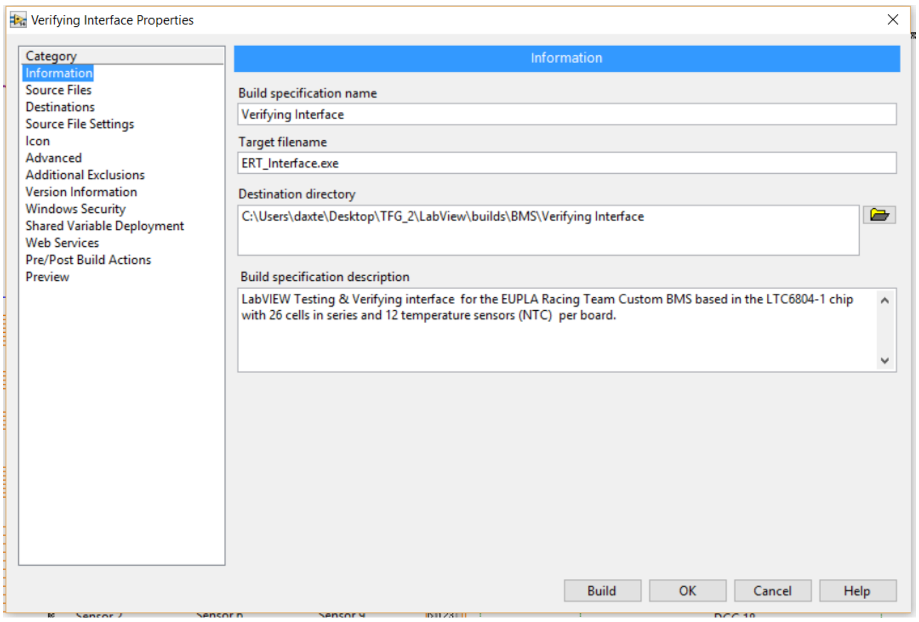


Figure 85. Application name and description window.

5.11. ATTACHMENT TO BATTERY PACK.

5.11.1. Connection details.

Now we will proceed to detail how the slave should be connected by describing where each pin of each connector in the PCB should be connected to.

Communication connectors:

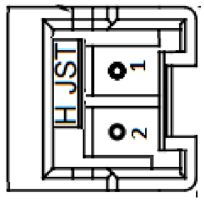


Table 10. 2 Pin JST Connectors layout

PIN\CONNECTOR	P2	P7
1	IM-IN	IP-OUT
2	IP-IN	IM-OUT

Temperature sensor connectors:

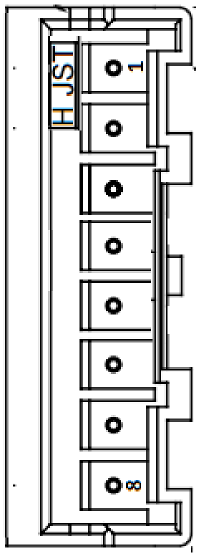


Table 11. 8 Pin JST Connectors layout

PIN\CONNECTOR	P1	P3	P8
1	NTC 1	NTC 5	NTC 9
2	NTC 1	NTC 5	NTC 9
3	NTC 2	NTC 6	NTC 10
4	NTC 2	NTC 6	NTC 10
5	NTC 3	NTC 7	NTC 11
6	NTC 3	NTC 7	NTC 11
7	NTC 4	NTC Bal.	NTC 12
8	NTC 4	NTC Bal.	NTC 12

Autor: **Xavier Anzuela Recasens**
424.16.96

Cell connectors:

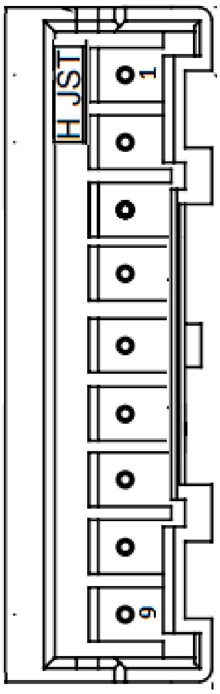


Table 12. 9 Pin JST Connectors layout

PIN\CONNECTOR	P4	P5	P6
1	GND	Cell 9	Cell 18
2	Cell 1	Cell 10	Cell 19
3	Cell 2	Cell 11	Cell 20
4	Cell 3	Cell 12	Cell 21
5	Cell 4	Cell 13	Cell 22
6	Cell 5	Cell 14	Cell 23
7	Cell 6	Cell 15	Cell 24
8	Cell 7	Cell 16	Cell 25
9	Cell 8	Cell 17	Cell 26

Also we have to bear in mind that the NTC will be placed on the bottom of the cells and in between two cells to have a higher sensing of cell temperature. In (Figure 86) we can see how the NTC are place in each substack and are holder by Kapton Tape, a special thermic resistant tape.

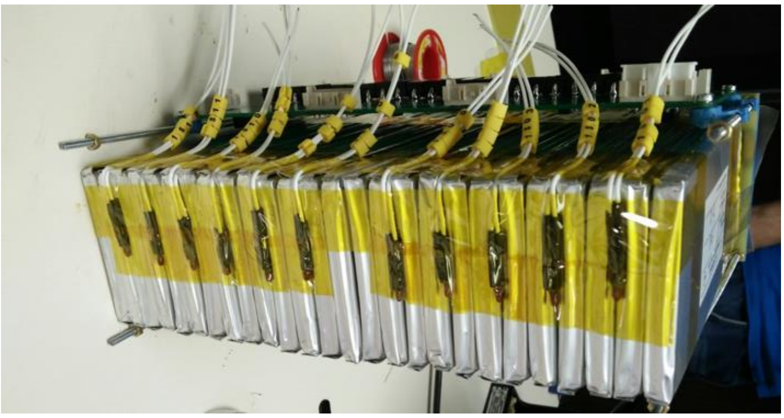


Figure 86. NTC placement to substack

5.11.2. Slave attachment to Substack.

Each slave has 4 M3 holes. These have been marked in (Figure 87)

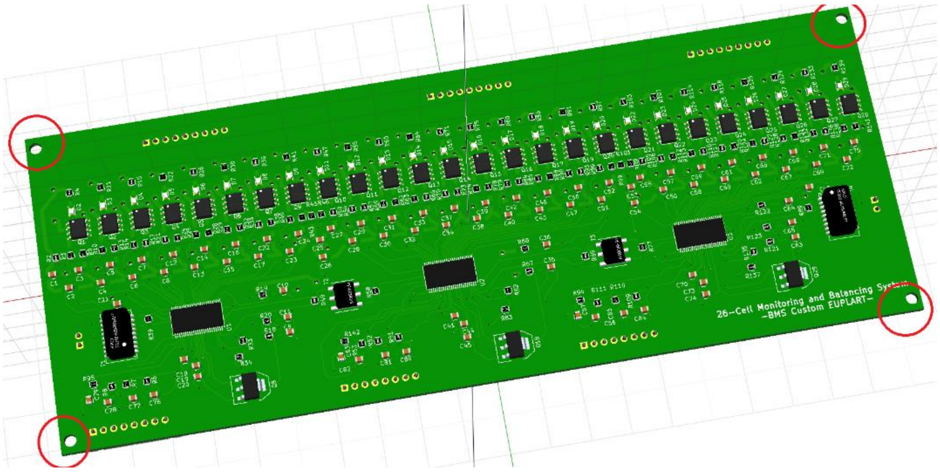


Figure 87. Slave attachment M3 holes.

In the case of our battery pack we will use the rods that make the structure of the substack to attach a PVC 3D printed piece designed by the battery pack manager. For other applications the customer will have the option to use this holes for the attachment or it may decide another attachment method.

6. RESULTS

6.1. HARDWARE

In (Figure 88) we have the final result of our design a completely functional BMS Slave, we can see both the version with the heat-sink and without it. We can observe that the design is very symmetric and has used the same rooting strategy for all 3 chips using all the available space.

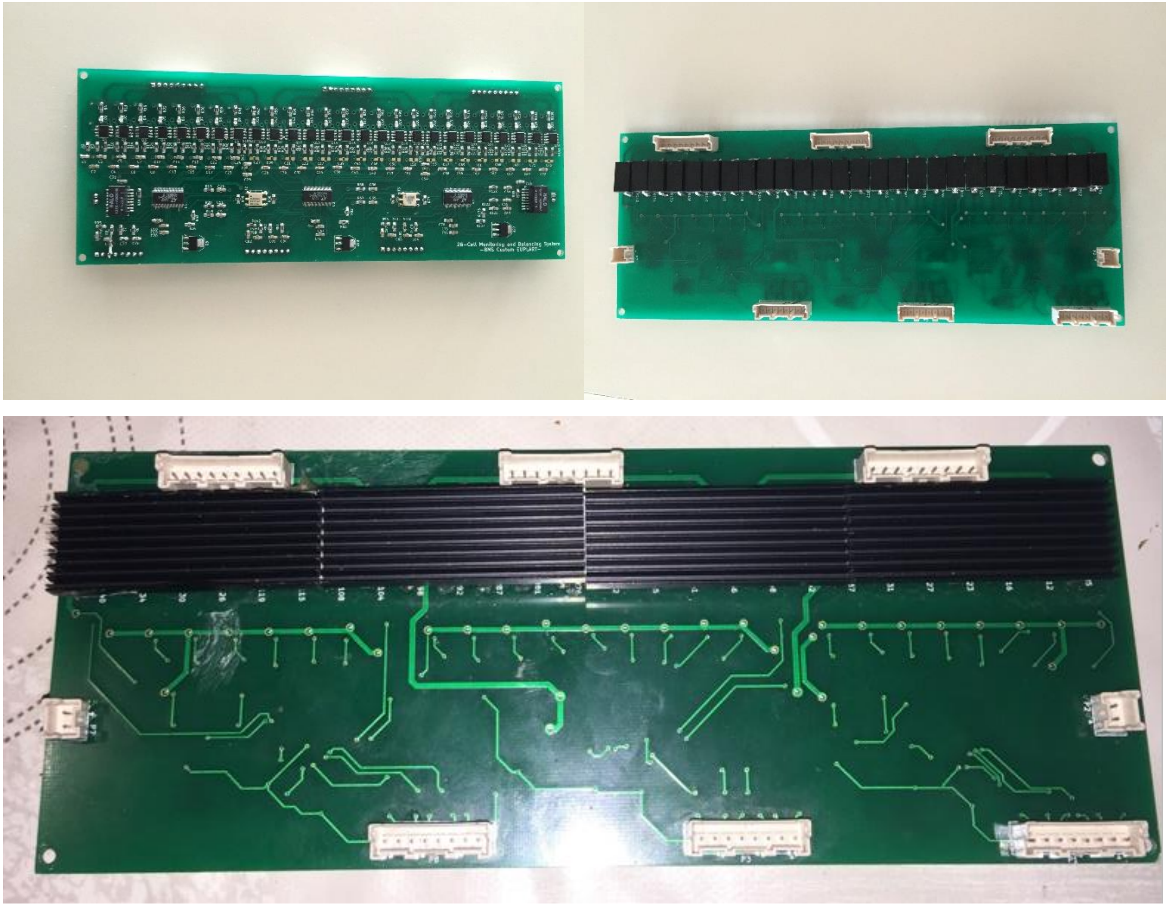


Figure 88. BMS Board views. Top-Bottom without heat sink-Bottom with heat sink.

In (Figure X), we see that the position of the connectors helps the manipulation of the slave while attached to the substack. The perpendicular position of the connectors saves space and gives accessibility to the board in case any repair is needed.

6.2. SOFTWARE & INTERFACE

(Figure 89) is an image of the created Interface; we see the General Battery Pack Status:

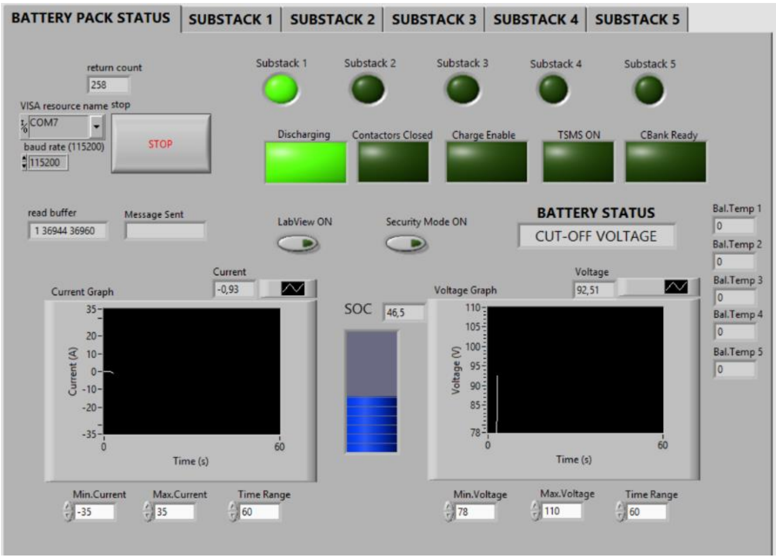


Figure 89. BMS General Battery pack status final version

(Figure 90) shows the representation of a substack, all the substacks have the same appearance.

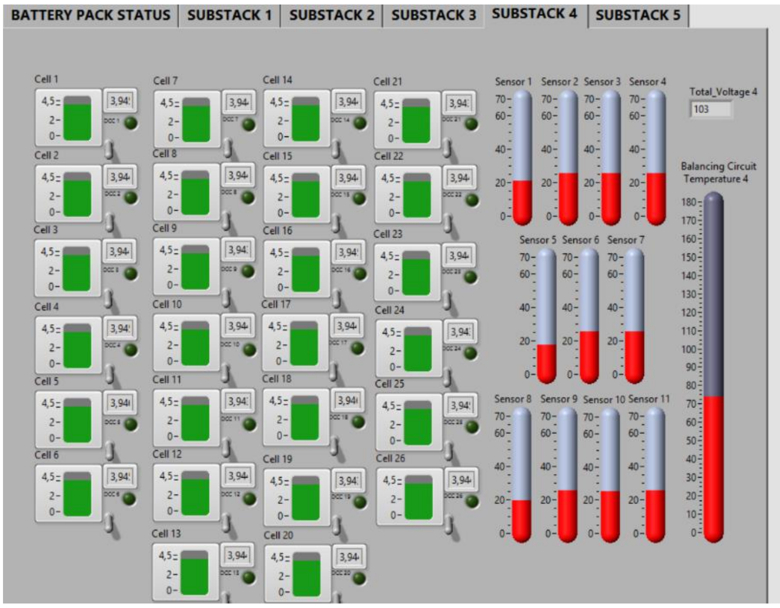


Figure 90.General Substack State while balancing

The thermometers rise as temperature is sensed in the NTCs and the cell tanks fill up. The levers are used to enable Balancing and when they are ON the LED lights up as the tank becomes blue if there is overvoltage in a cell or Undervoltage the tank turns red for OV and yellow for UV. This can be appreciated in (Figure 91).

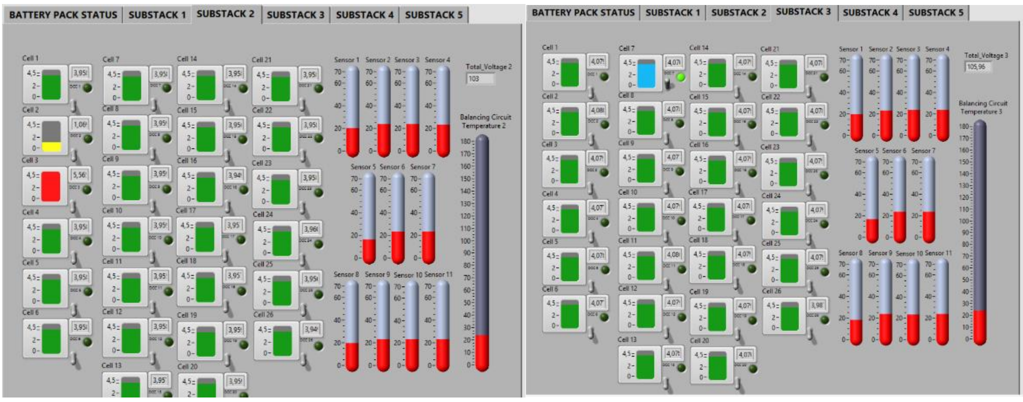


Figure 91. UnderVoltage/OverVoltage/Balancing cell examples

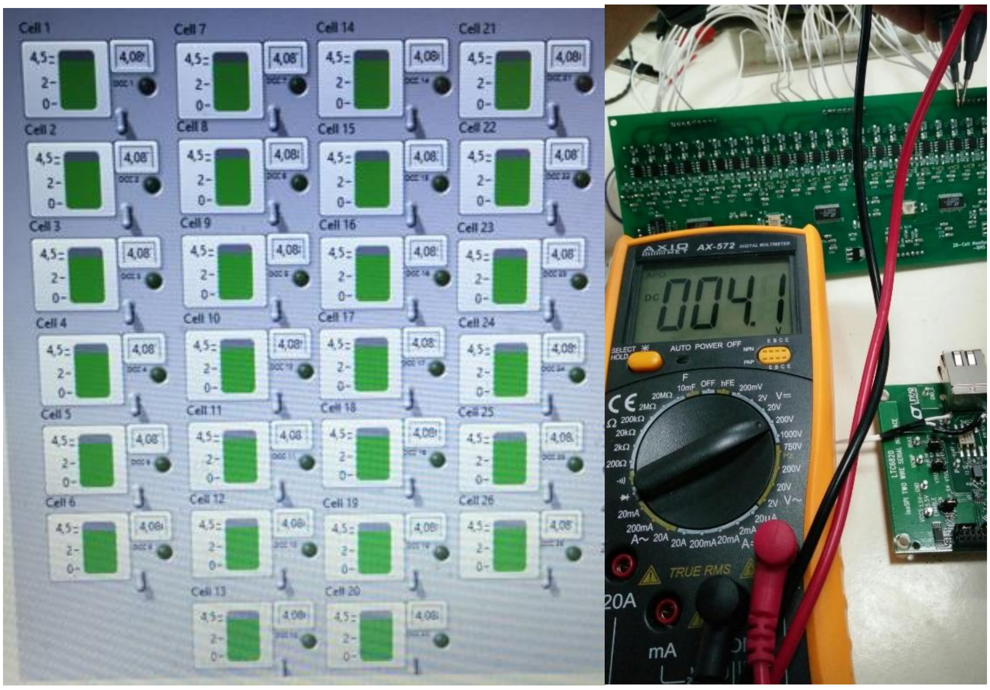


Figure 92. Readings verification

In (Figure 92) we can see that the readings are correct as the multimeter shows 4.1V and the readings in the interface shows 4.08V. Meaning that the reading is very accurate. The LTC6804-1 has a precision of ±0.1mV as we only show 2 decimals in the interface the precision of the reading we are viewing is of ±10mV but the software has a higher precision.

6.3. Cell Balancing

In the image below we can see how we were able to fully charge the battery pack to almost 100% with all the cells balanced with a 10mV maximum difference between them. We also can see the maximum state of charge we were able to reach in the other image. (Figure 93)

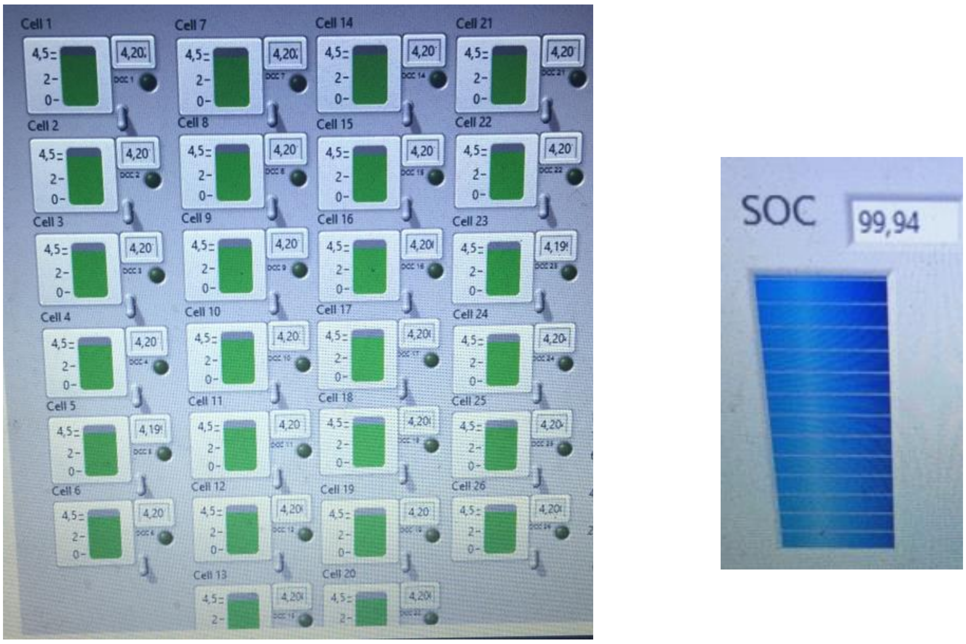


Figure 93. Perfectly Balanced cells and Maximum SOC.

Being able to reach almost 100% SOC is very important as is the amount of energy available before the shutdown circuit opens contactors to protect the battery pack. The more balanced are the cells the higher charge we can have without as the limit is set by the cell which reaches OV status. Although the energy available is set by the lowest level cell as it will be the first to reach UV status and the cut-off voltage protection will prevent the EV from using more energy.

During the event, after the second round of classification laps we had to charge the battery pack to have the maximum battery possible for the race. We had the chance to charge it for 1 minute before we thought we had to do a better lap in the classification to be able to be in the race. In (Figure 94) we see how during that period we reduced the max disbalance between cells by 60mV meaning that the balancing algorithm was doing its job.

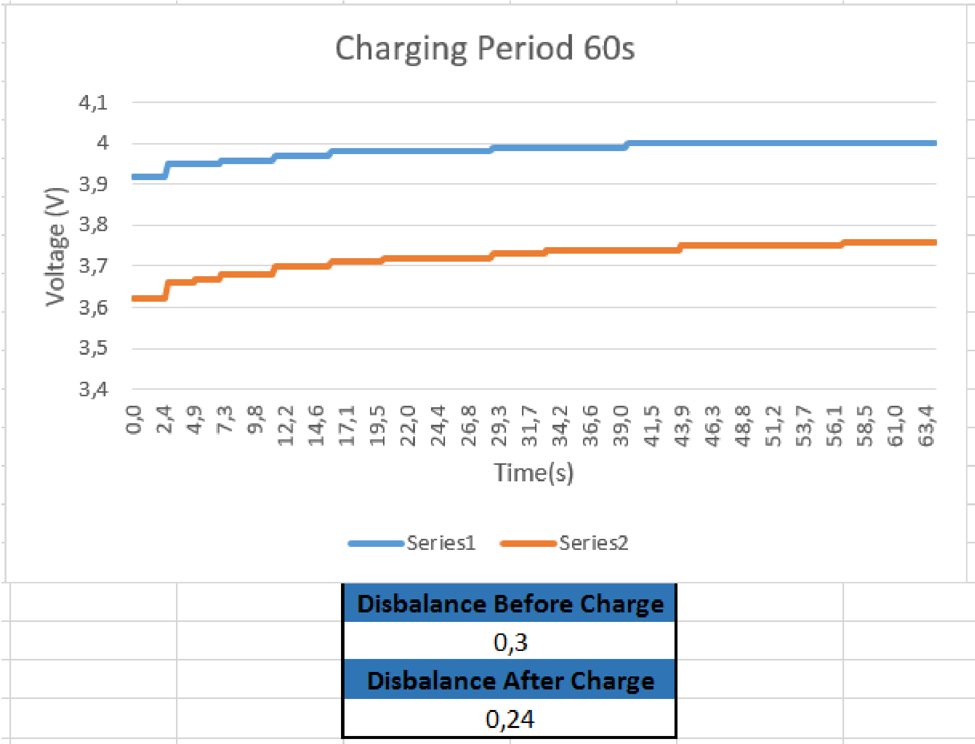


Figure 94. Difference between max. and min. cell during small charging period

The following plot extracted from an excel table with the data of all the substacks in a fully charged battery pack we can see that the difference between the highest cell and the lowest cell of all the substacks has a disbalance of 10mV. (Figure 95)

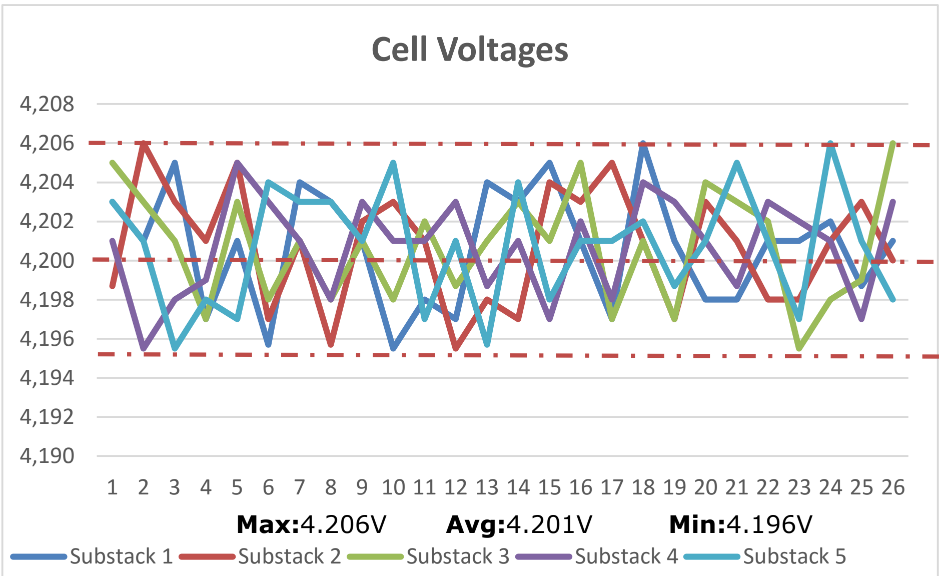


Figure 95. Balanced Cells of all 5 substacks 10mV disbalance

The excel can be checked in the spreadsheet annex. From the data of the cells we obtain the following values:

Max. Voltage	4,196V
Min. Voltage	4,206V
Median	4,201V
Avg. Voltage	4,201V
Typical Deviation	0,0010
Standard Deviation	0,0028

The smaller the deviation values are the better our system is working as it means the differences between the values of each cell are very similar. Meaning, the cells are correctly balanced.

During one test to check is the UV protection worked we disconnected the last cell of a substack to check if the cut-off voltage flag arrised. The result may be seen in (Figures 96 & 97).

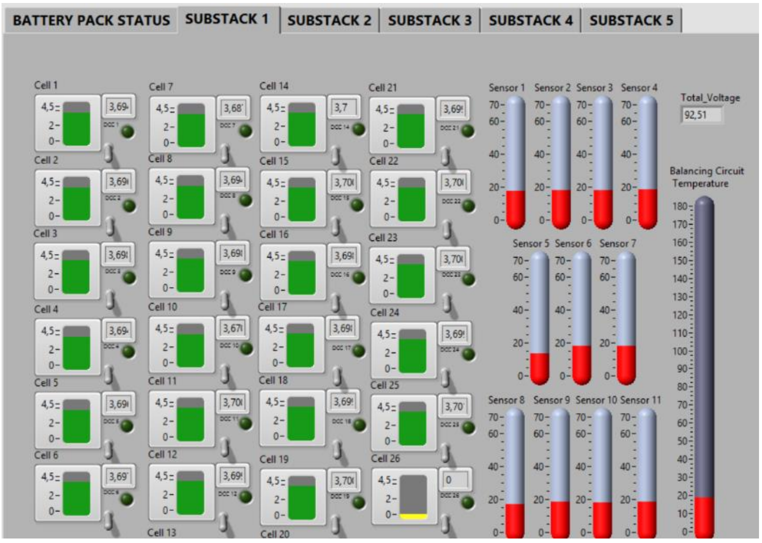


Figure 96. Substack with C26 disconnected.

Autor: **Xavier Anzuela Recasens**
424.16.96

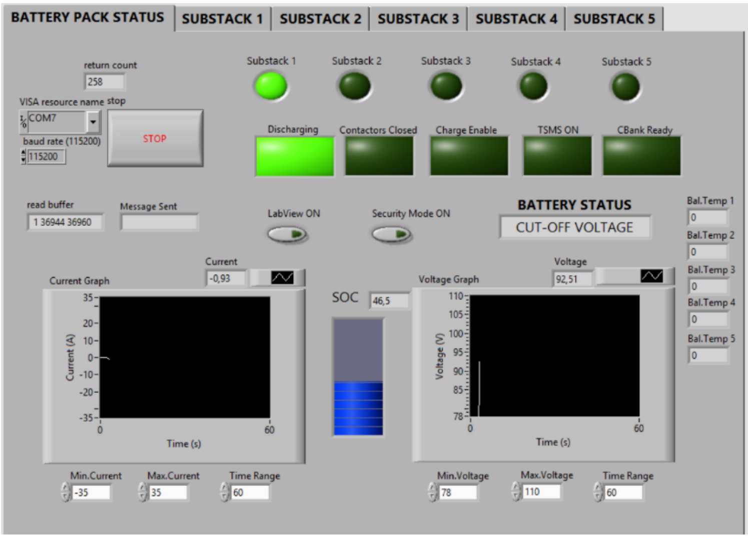


Figure 97. UV protection and Open contactors

6.4. Tests Rounds Data.

We already have seen communications between slaves and the interface work but also communication with the CAN Bus was an objective. These graphs show the readings we received from a 5-minute test we did in a karting circuit next to the Motorland Circuit in Alcañiz. (Figure 98)

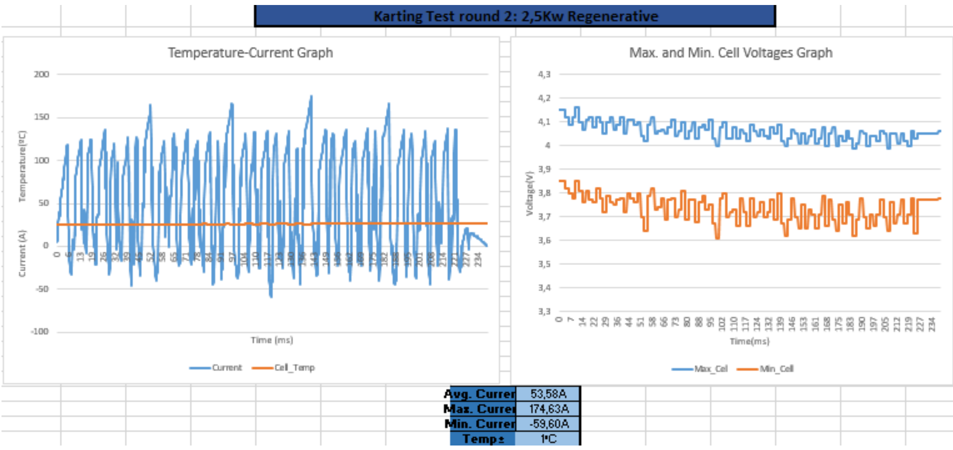


Figure 98. Karting Alcañiz 5 min. test 1

We can see that the evolution of the maximum cell and the minimum cell share the same pattern. Although, the minimum cell always has a greater voltage decrease, this may mean some cells have a higher discharge rate. The test had an average current of 53.6A and peak currents of -59.6A and 174.63A. The maximum current the battery can discharge is of 250A from there on the battery is degraded. The maximum regenerative current is 1C so -50A in this test we are already beyond that point but as long as not cell temperature is increased we should not worry.

The next test in the same karting circuit, is done with a higher regenerative current and higher power. The test length will also be of approximately 5 minutes. (Figure 99)

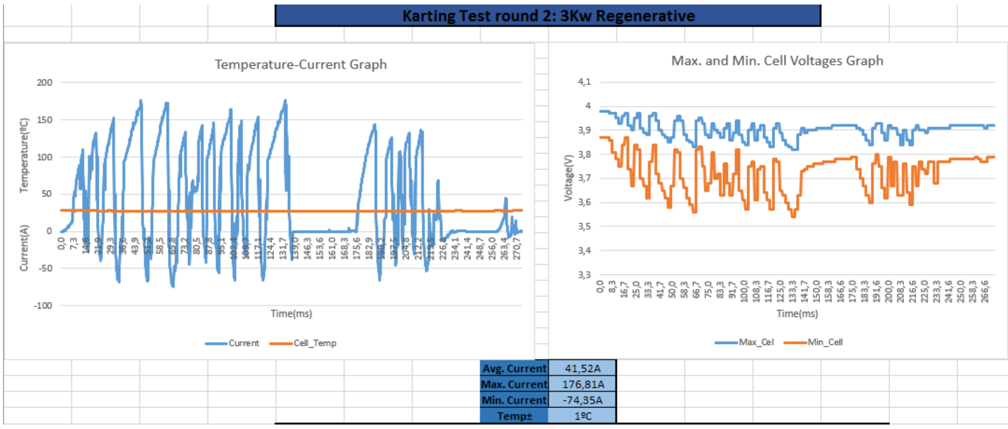


Figure 99.Karting Alcañiz 5,5 min. test 2

As the regenerative energy has been augmented there is a higher returning current -74.34A although, it has only been a peak. We have to bear in mind that this peaks may be very harmful at a long term. As the temperature gradient is still stable we consider that the battery is working correctly. No values are exceeded from the maximum number so no security issues have occurred.

The next test we see was made during a qualification round we only did 2 laps to be able to punctuate and save battery. This test was done before the other two tests but as we have two circuits the karting circuit and the Motorland circuit we decided to divide test by the location.(Figure 100)

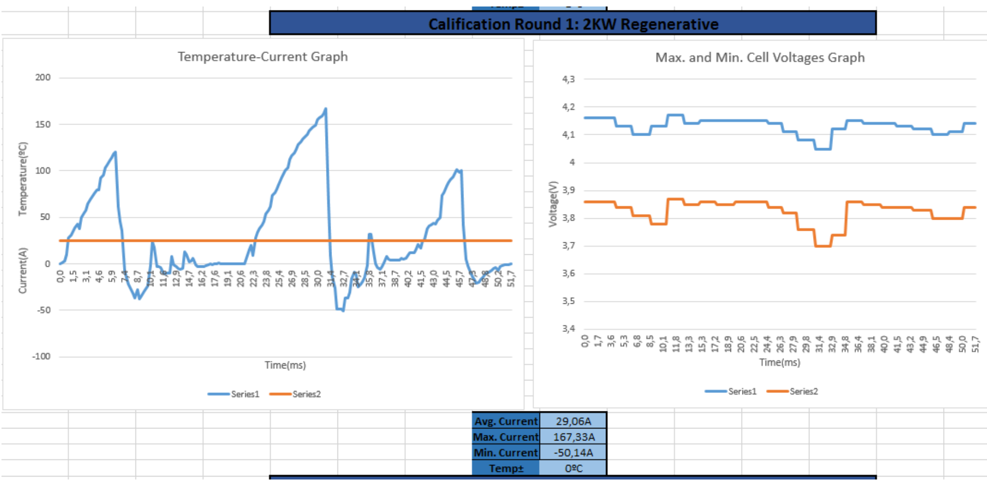


Figure 100. Qualification round 1

In this case, no temperature increase has occurred and the regenerative current is exactly at 1C so no further regenerative should be applied for battery security. The amount of regenerative current is also determined by other factor which have to be valued by the rest of the team. The tendency of the maximum and minimum cell voltages is the same in all cases there is always a slightly higher discharge rate in some cells.

The final test, was made during the last event day just before the race. In this test all the power of the engine was release by the regulator and so we should see the highest current and see how much current the engine actually asked for. This is an 8-minute test.(Figure 101)

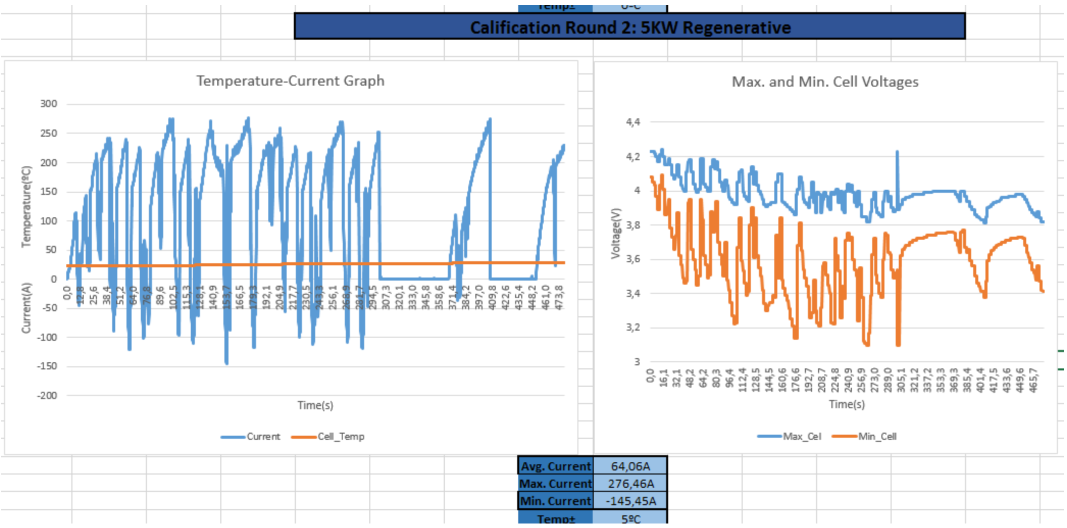


Figure 101. Qualification round 2

We got to a peak current of 276.46A this was higher that 5C the BMS would have stopped the EV for security if a that current was released for a period of 10 seconds. No advises are made by current further from 5C except for the advice that I can only stand 3 seconds at 7C. We set a 10 seconds period for current over 5C for security and in order to preserve cells.

We can see that the EV was stopped by the BMS as we see a peak near to 3V the system was set at 3.1V cut-off voltage because we had observed that the range between 3.3-3V was a very short time to avoid cells discharging further from 3V we decided to cut before that.

*Note that there is a very high returning current or -145.45A. This current is clearly damaging cells.

7. CONCLUSIONS

We have achieved a functional Battery Management System which successfully balances the battery packed it has been designed for. The system has been tested during the full MotoStudent IV competition and whenever a security issue has appeared the system has deactivated the EV. A software should be updated to have an Undervoltage timer this will prevent the system from stopping if a cell has a greater discharge rate or is damaged. During the event some cells had where damaged because as we were in a competition we used the cells in a range further from their nominal working conditions. The damaged cells discharged quicker when the pack was discharging and reached cut-off voltage (3V) for some milliseconds and then when back to their actual voltage. This occurred during current peaks and the BMS stop the EV for security as it was meant to.

The interface is fully functional and accessible with a serial cable to USB. An application has been created so it can be used in any computer. Any further modifications of the interface will depend on the final customer and their actual needs. The actual state of the interface is perfectly usable for testing issues and also for applications. During the competition the interface was very useful as it gave the team a full image of the battery pack state each time the EV came back to the box; it gave a quick response to the questions the team could have about the battery pack and easy to understand under the pressure of a competition. The interface was very helpful because it was clear, if any cells where damaged where shown in red and cells near to cut-off voltage where shown in yellow as an alert, if everything was "ok" the cells where shown in green. If the EV did not start because the BMS was stopping it from starting the Interface clearly showed a message indication why the security system was acting. Also it was really helpful during charging as it showed which balancing circuit had to be refrigerated. For commercial use and not for a competition usage a smaller balance current would be recommended which will make disappear temperature issues during balancing but also will make charging much slower. The typical balancing current for daily life applications is 100mV that would be achieved with a 42Ω resistor. The interface can be upgraded by showing which cells are balanced during charging, this was not added to our interface because it was first meant to be only for testing and all the information be read by the ECU but the ECU could not handle all that amount of information so we added a serial connector to the battery pack to be able to extract the information.

The high amount of data being read by serial made it impossible to read the CANBus from the master. Although it was not needed for the application as we had the source code and we were able to modify it, it would be a good idea to correct that issue for commercial use and be able to modify parameters such as OV and UV by CANBus reading. Our master had an 8-bit microprocessor maybe it was too limited for the application, upgrading to a 32-bit microprocessor with CANBus included would be a good idea. ARM Cortex M4 may suit our needs.[39]

As it was a competition application and the actual tests were short-timed a quick balance was needed and no active balancing was needed as there was no need for energy regeneration. If the application was headed to a commercial EV or an endurance competition vehicle where energy regeneration is a must a modification should be applied to the system. By using LTC3300-1 in combination with LTC6804-1 and active balancing Battery Management System can be achieved, linear offers a typical application schematic from which it can be developed.[40]

While we were choosing which chip was the best for our application we chose LTC6804-1 basically because we had a clear example and software libraries from which we could work. A new challenge would be to develop a BMS with the TI chip bq76pl536 and compare the results with the Linear Technology BMS.

8. BIBLIOGRAFY

[1] "Li-Ion BMS - White Paper - How much balancing current do you need?" [Online]. Available: http://liionbms.com/php/wp_balance_current.php. [Accessed: 01-Nov-2016].

[2] "Toyota Global Site | The Hybrid that Started it All." [Online]. Available: http://www.toyota-global.com/company/toyota_traditions/innovation/nov2008_feb2009_1.html. [Accessed: 04-Jul-2016].

[3] "About Tesla | Tesla Motors." [Online]. Available: https://www.teslamotors.com/about. [Accessed: 04-Jul-2016].

[4] P. Weicker, *A SYSTEMS APPROACH TO Lithium-Ion Battery Management*, 2014 ARTECH., vol. 1, 1 vols. ARTECH HOUSE.

[5] "Elithion - Lithiumate pro." [Online]. Available: http://elithion.com/lithiumate_pro.php. [Accessed: 05-Jun-2016].

[6] "Elithion - Orion BMS Comparison." [Online]. Available: http://elithion.com/orion_bms_comparison.php. [Accessed: 05-Jun-2016].

[7] Fraunhofer IIS, "2015_01_15_Datasheet_FlexBMS1.pdf," *Fraunhofer.de*, 06-May-2016. [Online]. Available: http://www.iis.fraunhofer.de/content/dam/iis/de/doc/lv/los/energie/2015_01_15_Datasheet_FlexBMS1.pdf. [Accessed: 05-Jun-2016].

[8] "Ewert Energy Systems |." [Online]. Available: http://www.ewertenergy.com/. [Accessed: 06-Jun-2016].

[9] "Pre-wired Harnesses | Orion Li-Ion Battery Management System." [Online]. Available: http://www.orionbms.com/products/pre-wired-harnesses/. [Accessed: 06-Jun-2016].

[10] "Comparison | Orion Li-Ion Battery Management System." [Online]. Available: http://www.orionbms.com/comparison/. [Accessed: 06-Jun-2016].

[11] "Orion BMS – Standard Size | Orion Li-Ion Battery Management System." [Online]. Available: http://www.orionbms.com/products/orion-bms-standard/. [Accessed: 06-Jun-2016].

[12] Orion, "MiniBMS." CleanPowerAuto, 2014.

[13] "CleanPowerAuto LLC – Lithium Battery Management Systems." [Online]. Available: http://cleanpowerauto.com/. [Accessed: 30-Sep-2016].

[14] A. Collet, "FreeSafe Extended Freemens BMS." Freemens, 2016.

[15] M. Gaig Font, "Evolució del sistema d'emmagatzematge d'energia d'un vehicle de la Formula Student," 2015.

[16] "Graphenano productos con grafeno - Baterías, Pinturas y Hormigón." [Online]. Available: http://www.graphenano.com/es/. [Accessed: 22-Oct-2016].

[17] "La empresa española Graphenano fabricará baterías de grafeno que extienden la autonomía de los coches eléctricos." [Online]. Available: http://www.lavanguardia.com/motor/tendencias/20160210/302049349400/empresa-espanola-graphenano-baterias-grafeno-extension-autonomia-coches-electricos.html. [Accessed: 22-Oct-2016].

[18] W. A. Hermann and S. I. Kohn, "Response to over-current in a battery pack," US9257729 B2, 09-Feb-2016.

[19] T. H. Phung, A. Collet, and J.-C. Crebier, "An Optimized Topology for Next-to-Next Balancing of Series-Connected Lithium-ion Cells," *IEEE Trans. Power Electron.*, vol. 29, no. 9, pp. 4603–4613, Sep. 2014.

[20] D. Andrea, *Battery Management Systems for Large Lithium-ion Battery Packs*, Artech House., vol. 1, 1 vols. ARTECH HOUSE.

[21] Artech, *Battery Power Management for Portable Devices*, vol. 1, 1 vols. UK: ARTECH HOUSE.

[22] "The Basics of Current Sensors | DigiKey." [Online]. Available: http://www.digikey.com/en/articles/techzone/2012/sep/the-basics-of-current-sensors. [Accessed: 22-Oct-2016].

[23] "Taking Transducer Technology to the Limits of Current Measurement Accuracy | PowerGuru - Power Electronics Information Portal." [Online]. Available: http://www.powerguru.org/taking-transducer-technology-to-the-limits-of-current-measurement-accuracy/. [Accessed: 22-Oct-2016].

[24] LinearTechnology, "LinearTechnology LTC6820." .

[25] M. Di Natale, "Understanding and using the Controller Area network," *Inst Eecs Berkeley Edu~ Ee249fa08Lectureshandoutcanbus2 Pdf*, 2008.

[26] W. TKIET and D. P. India 2nd Kunal, "International Journal of Informative & Futuristic Research ISSN: 2347-1697."

[27] Melasta, "DataSheet Melasta SLPB8564159." 2016-2015.

[28] "Liion Chip Comparison." [Online]. Available: http://liionbms.com/html/BMS_IC_table.html. [Accessed: 29-Oct-2016].

[29] W. Tiashu, K. Taeyoung, and P. Sangyoung, "Battery management & application for energy efficient buildings," Design Automation Conference 2014.

[30] A. Collet, "Freemens SAFE TX." .

[31] "SparkFun Electronics • View topic - LTC 6803-1 BMS." [Online]. Available: https://forum.sparkfun.com/viewtopic.php?f=14&t=36965. [Accessed: 29-Oct-2016].

[32] TEXAS INSTRUMENTS, "bq76pl536a." 2014.

[33] LinearTechnology, "LTC6804-/LCT6804-2 Multicell Battery Monitors." 2015.

[34] LinearTechnology, "DC1894B - LTC6804-1 Evaluation Board." 2014.

[35] AVX, "Accurate Thermistor-NTC28RA0104H." .

[36] "PCB Printed Circuit Board File Creation Calculator | Advanced Circuits." [Online]. Available: http://www.4pcb.com/trace-width-calculator.html. [Accessed: 14-Nov-2016].

[37] SISKO COLORADO, "IPC_2221." Colorado University, May-2003.

[38] "KiCad EDA." [Online]. Available: http://kicad-pcb.org/. [Accessed: 14-Nov-2016].

[39] "Cortex-M4 Processor - ARM." [Online]. Available: https://www.arm.com/products/processors/cortex-m/cortex-m4-processor.php. [Accessed: 24-Nov-2016].

[40] "LTC3300-1 - High Efficiency Bidirectional Multicell Battery Balancer - Linear Technology." [Online]. Available: http://www.linear.com/product/LTC3300-1. [Accessed: 21-Nov-2016].

Relación de documentos

(X) Memoria ....126 páginas

(_) Planos .... 22 páginas

(_) Anexo Componentes y Demoboards......225 páginas

(_) Anexo Tablas .... 4 páginas

La Almunia, a 29 de noviembre de 2016

Firmado: Xavier Anzuela Recasens

Etiquetas para CD/DVD

eupla

Escuela Universitaria Politécnica
La Almunia de Doña Godina
Zaragoza

Nº TFG:
424.16.96

Director:

Fdo:
David Asiain Ansorena

Título TFG:

Design of a modular BMS for EV focused on MotoStudent

Autor:

Xavier Anzuela Recasens

September 2016

eupla

Escuela Universitaria Politécnica
La Almunia de Doña Godina
Zaragoza

Nº TFG:
424.16.96

Director:

Fdo:
David Asiain Ansorena

Título TFG:

Design of a modular BMS for EV focused on MotoStudent

Autor:

Xavier Anzuela Recasens

September 2016

ESCUELA UNIVERSITARIA POLITÉCNICA

DE LA ALMUNIA DE DOÑA GODINA (ZARAGOZA)

Design of a modular BMS for EV focused on MotoStudent

Diseño de un BMS modular para EV enfocado a MotoStudent

424.16.96

Autor: Xavier Anzuela Recasens

Director: David Asiain Ansorena

Fecha: September 2016