USER MANUAL c-BMS

Advanced battery management system for automotive, industrial and storage battery pack application

2.5.11

Contents

1 Introduction

1.1 Document purpose and structure

The purpose of this manual is to provide the user of the Lithium Balance c-BMS systems with an overview of the BMS system and to allow the user to configure the system hardware and set-up basic parameters in the software.

1.1.1 Disclaimer

We have taken every precaution to ensure that all information provided in this manual is correct and up to date. However, Lithium Balance assumes no responsibility for damage to persons and property arising as a result of following recommendations and/or procedures described in this manual. Furthermore, Lithium Balance assumes no responsibility for any infringements of rights of third parties which may result from the use of this manual.

1.1.2 General safety warning

Please, carefully read all sections describing safety issues and precautions before setting up, operating, or performing service work on any BMS controlled system! Failure to do so could result in reduced system performance, damage to the system, personal injury or even casualties.

1.1.3 Copyright and patent information

Patents owned or applied for by Lithium Balance are listed below.

Area	Number	Status	Area
ZT/CN	200780048774	Granted	Master/Slave
US/US	8,350,529	Granted	Master/Slave
EPO/EU	07817888.6	Granted	Master/Slave
US/US	13/735,946	Application	Passive Balancing
PCT/WO	IB2015/055441	Application	EIS in BMS
PCT/WO	EP2016/066287	Application	EIS in BMS
PA/DK	2016 00228	Granted	Active Balancing

1.1.4 Abbreviations

c-BMS	Compact Battery Management System
BPU	Battery Protection Unit – a unit containing e.g. switches and fuses for battery protection
DoD	Depth of Discharge
BMS	Battery Management System
SoC	State-of-Charge
SoH	State-of-Health
VCU	Vehicle Control Unit
DLC	Data Length Code (of CAN Frame)
NTC	Negative Temperature Coefficient, thermistors
GPIO	General Purpose Input Output
PSU	Power Supply Unit

1.2 BMS Overview

The Lithium Balance Battery Management Systems are multi-functional and highly configurable systems which can manage all types of Li-ion cell chemistries and packages. The system addresses several aspects of battery management such as:

- **Safety**-the BMS monitors the essential parameters (cell voltage, current and temperature) and adjusts charge and discharge current levels to ensure safe operation. If for some reason the BMS cannot bring the system to operate within safe limits it will abort the operation by opening contactors/relays to the battery.
- **Performance**- The BMS monitors the battery performance, e.g. State of Charge and optimizes operation, e.g. by balancing the battery to ensure that all cells can be utilized to provide the maximum energy.
- **Communication and control**- the BMS communicates with users and connected equipment and **controls** safety related system elements like switches and chargers.

Figure 1.1 below shows a block diagram of a typical c-BMS system.

1. The c-BMS printed circuit board in connected to auxiliary equipment by dedicated cables.
2. The BMS communicates with other equipment over UDS CAN. To interface computers a (PEAK) CAN adaptor is required.
3. Configuration parameters to the BMS is configured via the Lithium Balance BMS Creator operating on a Windows PC. A correct configuration of interfaces to auxiliary equipment, safety critical parameter threshold and system response in case of errors is critical to the safe operation of the system.
4. Operating parameters (e.g. current and voltage) are read **from** the BMS via a standard CAN bus interface software. Here Lithium Balance recommends the use of the Busmaster program.

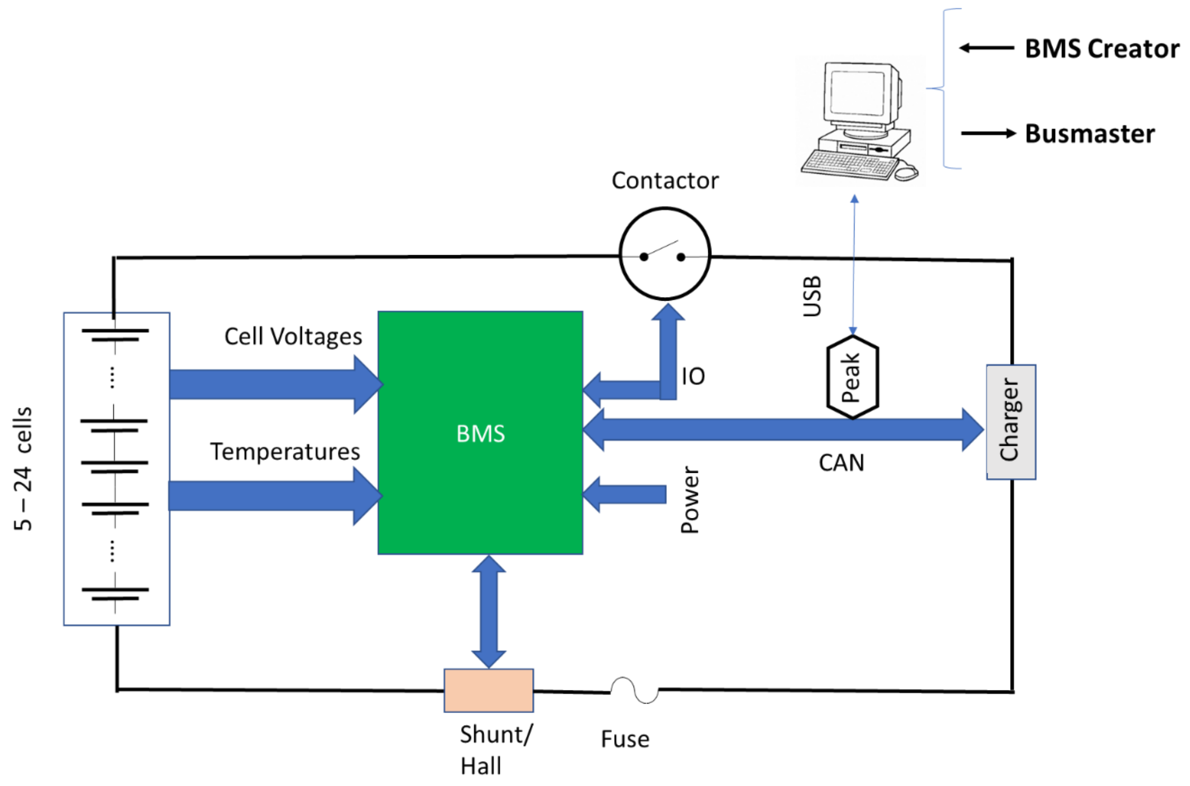


Figure 1.1: c-BMS system block diagram

1.2.1 Hardware

Physically, the c-BMS system is a single board containing:

- Central functions like current monitoring, data processing (e.g. SoC calculations) and external communication.
- Battery cell monitoring such as cell voltage and temperature monitoring.

The power to the central functions are provided from an external 12V power source (e.g a lead acid battery or via a DC/DC converter from the main battery), while the power to the cell monitoring functions are provided directly from the main battery.

1.2.2 Firmware

The advanced functions of the BMS systems are enabled by the firmware running in the advanced microprocessor. Firmware is already present on the BMS boards shipped from Lithium Balance. They are configured without a configuration, which means the boards will start up in bootload mode. Firmware versions and configurations can be updated using the Lithium Balance "BMS Creator" PC tool.

1.2.3 BMS Creator

A PC configuration and interface tool called 'BMS Creator' is provided to configure the BMS. These functions include for example:

- configuring the BMS including setting up the safety limits for battery operation and defining associated actions if these thresholds are exceeded.
- setting up CAN communication with the auxiliary equipment.
- Optimizing system operation, e.g. controlling charger regulations

1.3 Safety

1.3.1 Battery safety

To achieve safe operation of lithium-ion batteries, these must operate within the safe limits of:

- cell voltage
- cell/battery current
- cell/battery temperature

Exceeding these limits can trigger chemical processes that may lead to rapid degradation of battery performance. In more extreme cases, exceeding the limits can result in out-gassing from the batteries and in battery fire. In the worst cases, this may have fatal consequences.

Toxic Gas

Risk of fire

Risk of explosion

One of the main functions of any BMS is to ensure that the battery operates within the safe limits for cell voltage, temperature and current under all circumstances. To perform this safety critical function, the BMS must be correctly installed, configured and operated as described in this manual.

1.3.2 Electrical safety

Apart from the chemical related safety aspects of working with batteries and BMSs, batteries are powerful electrical devices with potential voltage levels exceeding 1000/100 V and maximum current levels typically exceeding several hundreds of Amperes.

All personnel working with BMS systems and batteries should therefore be properly trained in handling high voltage/current installations. The legally required training is regulated by national/regional standards such as EN 50110-1 (EU) and IS 5216 (India). As a general precaution when working with battery systems, Lithium Balance recommends that all personnel:

- wear electrically insulating gloves
- use electrically insulated tools
- disconnect the BMS from high voltage sources such as the charger and the main battery whenever possible

1.3.3 Safe design

The battery system must be adequately configured and tested in order for the BMS to ensure safe and proper operation in all operating conditions. The system configurations needed for a safe battery system may vary depending on the application. However, for all systems Lithium Balance strongly recommends that the three guidelines below are **always** followed:

- The safe operating cell limits (voltage, current and temperatures) of the battery cells must be configured correctly in the BMS before the first operation of the battery. The operating cell limits must be supplied directly by the cell manufacturer, e.g. taken from the **cell datasheet**.

- The system must be able to stop charging when the maximum cell voltage has been reached. This requires:
    a. robust and well-tested **communication between the BMS and the charger**
    b. a **switch** which can disconnect the charger from the battery in case the direct communication between the BMS and the charger fails.

       Such **redundant charger control** is required as communication failures could happen in case of e.g. inadequately mounted connectors, defective or broken communication cables or excessive electrical noise.
- It must be possible to block excessive current from the battery in order to ensure safe operation and handling of batteries. For this purpose, a fast switch or a quick blow fuse should be placed in the main current path. This will typically be next to the battery or inside the battery as a 'mid pack' switch or fuse. Even if a switch is used, fuses are in general recommended as they can prevent hazardous situations such as the following:
    a. The load draws an excessive current from the battery which could lead to rapid overheating of the battery cells.
    b. The battery is accidentally short-circuited.

1.3.4 Testing for safe operation

Before the BMS is put into operation, proper operation of the BMS with the battery system must be validated under supervision. Specifically, it is recommended to actively verify the correct function all safety features listed in the previous section 1.3.3 during a supervised full charging and balancing of the battery.

It is strongly recommended that the first charge cycle is always supervised to verify that charging stops when the maximum cell voltage level has been reached. The anticipated maximum duration of the first charge cycle can be calculated as the battery capacity (Ah) divided by the charger current.

1.4 Handling

This section covers handling, storage and shipping for the BMS boards only. Storage, handling and shipping of lithium batteries typically requires special precautions. Please refer to the documentation from the battery supplier for such precautions.

1.4.1 Electrostatic discharge

Electrostatic discharge (ESD) can damage or destroy sensitive electronic components (typically semiconductors) on the BMS boards. When handling the boards, it is therefore critical to comply with good ESD prevention workmanship standards like the following:

- For storage and shipping, please keep the BMS boards in the antistatic bags in which they are shipped from Lithium Balance.
- Always wear a grounded antistatic wrist-strap in contact with the skin when handling the boards outside the antistatic bags.
- Ensure that all personnel wear antistatic smocks (clothes).
- Ensure that all personnel receive education and training on ESD preventive measures in general.

A detailed guideline for safe handling of ESD sensitive devices can be provided by Lithium Balance.

1.4.2 Storage

The BMS boards must be stored at temperature and humidity levels that do not exceed specified operating conditions. See data sheet for further details.

If the BMS boards are to be stored for longer periods of time, it is recommended to store them at temperatures below 25 ˚C and humidity levels below 70% RH.

1.4.3 Shipping

When shipping the BMS boards, it is important that the boards are:

- kept in the antistatic bags in which they are supplied
- packaged individually to prevent boards from touching physically, as this could cause damage to the boards during transportation. If possible, please use cardboard boxes with individual slots for each board, similar to those used for shipping from Lithium Balance.

Figure 1.2: Example of BMS board packaging in individual slots

1.4.4 Disposal

At the end of their service life, BMS boards must be disposed of as waste electrical and electronic equipment (WEEE). Follow appropriate guidelines in force locally (e.g. Directive 2012/19 in the EU) regarding collection and disposal.

1.5 Document structure

The manual is structured to support easy installation of BMS's in battery systems:

- Chapter 2 describes the BMS hardware and how to physically connect the BMS to the battery pack and other auxiliary units
- Chapter 3 describes the BMS functions, and the key parameters used for configuring the BMS
- Chapter 4 describes the key aspects of CAN bus communication
- Chapter 5 describes the BMS Creator used for configuring the BMS
- Chapter 6 describes auxiliary components and how they interact with the BMS
- Chapter 7 summaries the main steps of the BMS installation in a Quick Installation Guide
- A number of appendices are found in Chapter 9

1.6 Key Specifications

1.6.1 Environmental conditions and test

Standard operating conditions which will not reduce system lifetime are Temperature: 15 – 35 ˚C, Humidity: 25 – 75% RH, air pressure 860 to 1060 mBar.

The BMS system has been environmentally tested as listed below:

EMC Emission	CISPR 12, 16, 22, 25: EMC susceptibility per UN Addendum 9: regulation No.10 rev. 5 ISO 11452: EMC susceptibility 20 MHz– 2 GHz per UN Addendum 9: regulation No.10 rev. 5
EMC, Reversed polarity protection on 12 / 24 V.	ISO 16750-2: Electrical Loads (Reversed, protection only).
EMC, Immunity	ISO 11452-4, Immunity Bulk current injection, 60 mA from 20 MHz to 400 MHz. per UN Addendum 9: regulation No.10 rev. 5 ISO 11452-2, Absorber chamber test, 30 V/m from 200 MHz to 2 GHz. per UN Addendum 9: regulation No.10 rev. 5 ISO 7637-2: Immunity to Electrical disturbances from conduction and coupling. Pulse: 1, 2a, 2b, 3a, 3b, 4. For 12 V & 24 V. per UN Addendum 9: regulation No.10 rev. 5 IEC 61000-4-4: Immunity of ESA to electrical fast transient/burst disturbances conducted along AC and DC power lines. ±2 kV.
Temperature/Humidity	ISO 16750-4: Climatic Loads (Code G: -40 to 85˚C). Tested up to 93% RH at 55˚C
Vibration	ISO 16750-3: Mechanical Loads (Sinusoidal and random vibration)

1.6.2 Balancing

Balancing scheme	Passive balancing (bleeding)
Balancing current	200 mA @ 4.2 VDC max

1.6.3 Voltage sensing

Cell voltage range	0-5 V. High accuracy (see next line) for 1 – 4.5V
Cell voltage accuracy	<±1.5 mV over the entire temp range and from 1 – 4.5V
Cell voltage sampling frequency	10 Hz typical

1.6.4 Current sensing

Measurement schemes	1-Shunt. 2-Single or dual channel hall sensor. 3-CAN based current sensor.
Current measurement by shunt[1]	±450 mV
Shunt measurement accuracy	±0.5 mV
Hall measurement range	0 – 2.5 V Current in (Charging) 2.5 – 5V Current Out (Discharging)
Hall measurement accuracy	±1.5 mV

1.7 c-BMS specification

1.7.1 Dimensions

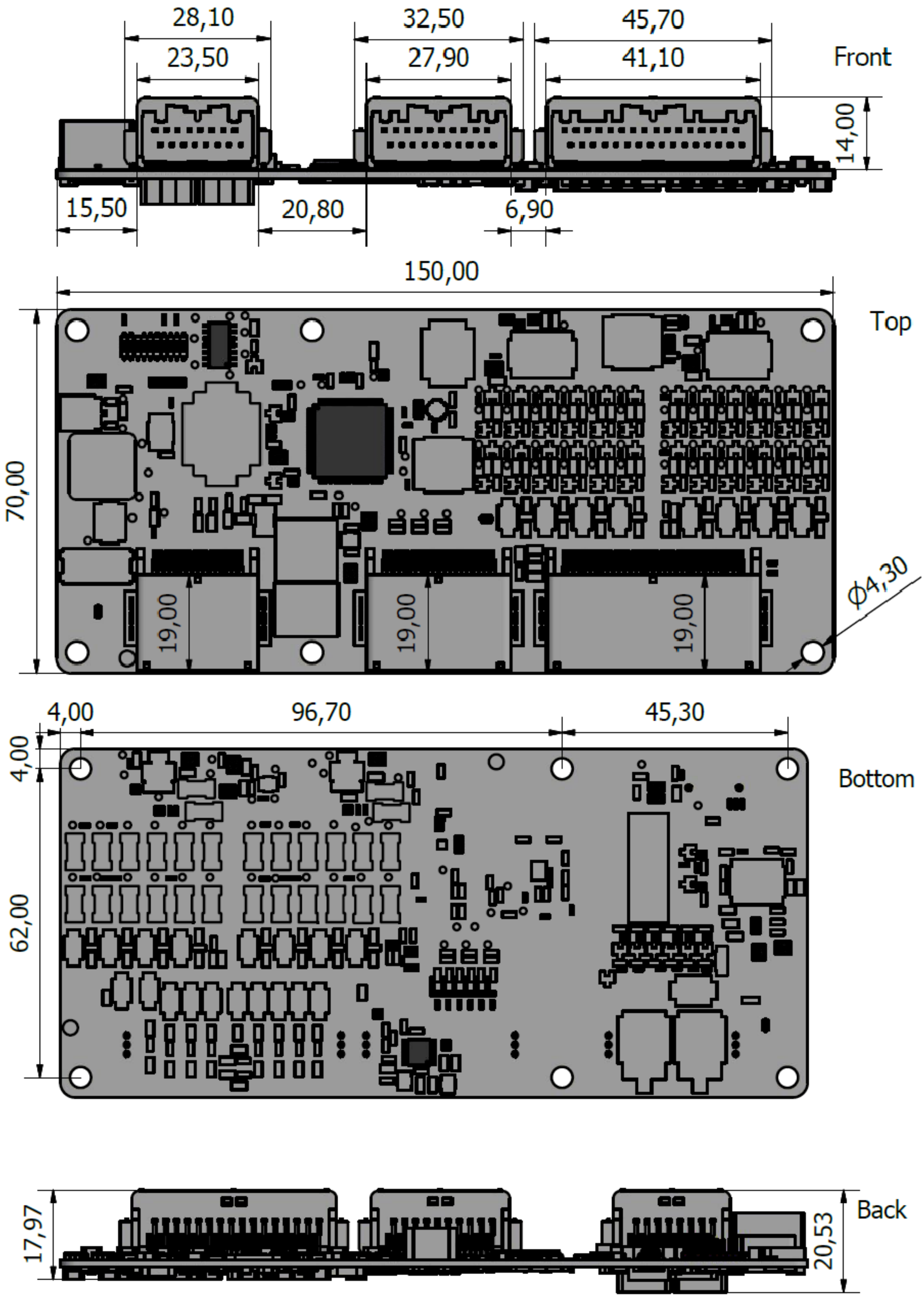
Dimensions	150.0 x 70.0 x 18.1 mm
Weight	< 93 g

3D model of the board is available upon request.

1.7.2 Battery limits

Parameter	**Interval**
Maximum number of Cells	24
Minimum number of cells	The CMU requires at least 11 V from the main battery to power the cell monitoring part. For most lithium battery types 4-5 cells are sufficient to supply minimum 11V.
Max battery capacity	2000 Ah[2]
Max battery voltage	120 V

[1] 100-1000 $\mu\Omega$ shunt
[2] Software setting. Can be changed if required


28,10
23,50
32,50
27,90
45,70
41,10
Front
14,00
15,50
20,80
6,90
150,00
Top
70,00
19,00
19,00
19,00
Ø4,30
4,00
96,70
45,30
4,00
62,00
Bottom
17,97
20,53
Back

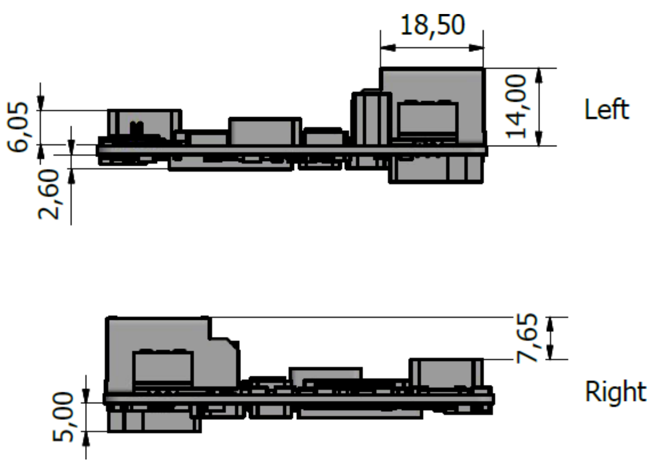


Figure 1.2: c-BMS Dimensions, Note: All dimensions are in mm.

1.7.3 Voltage isolation

High Voltage isolation	Up to 300 V according to ISO 6469-3-2011

1.7.4 Power supply and consumption

Supply voltage	12 VDC (6-30 VDC)
Power consumption from supply voltage (12V) – Active mode	<1.2 W @ 12V
Power consumption from Main battery – Active mode	<0.6 W @ 48V DC
Power consumption from supply voltage (12V) – Sleep mode	<0.6 mW
Power consumption from Main battery – Sleep mode	<2 mW @ 48V

1.7.5 Temperature sensing

No of temp sensors	4 on-board + 6 externals (typically used for cells)
Temperature sensor type	NTC, 10 kΩ @ 25 $^{\circ}$C, β = 3900 [3]
Temperature measurement tolerance	BMS circuit tolerance < $\pm 1.0^{\circ}$, for the entire temperature range. NTC temperature tolerance has to be added.

1.7.6 Communication & control

Charger + general purpose communication	CAN bus 2.0 A/B 11 bit and 29 bit ID, Isolated towards the battery pack, connected to BMS Gnd Protocol UDS on CAN 125, 250, 500, 1000 kBit/sec
For BMS Creator	UDS on CAN as described above
Analogue Charger control	Pulse Width Modulation (PWM)
Configurable input/output ports	4
Digital input ports	4

[3] Sensor characteristics are user-configurable for improved sensor support

2 c-BMS - Printed Circuit Board

This chapter explains the basic functions of the c-BMS printed circuit board and how to connect it to the rest of the battery system.

Please note that the BMS can operate at high voltages and care must be taken when installing or performing maintenance tasks on the BMS and the battery system.

2.1 Mechanical Overview

A mechanical overview of the c-BMS board is shown below indicating connectors. LEDs and labels. These are described in more details below.

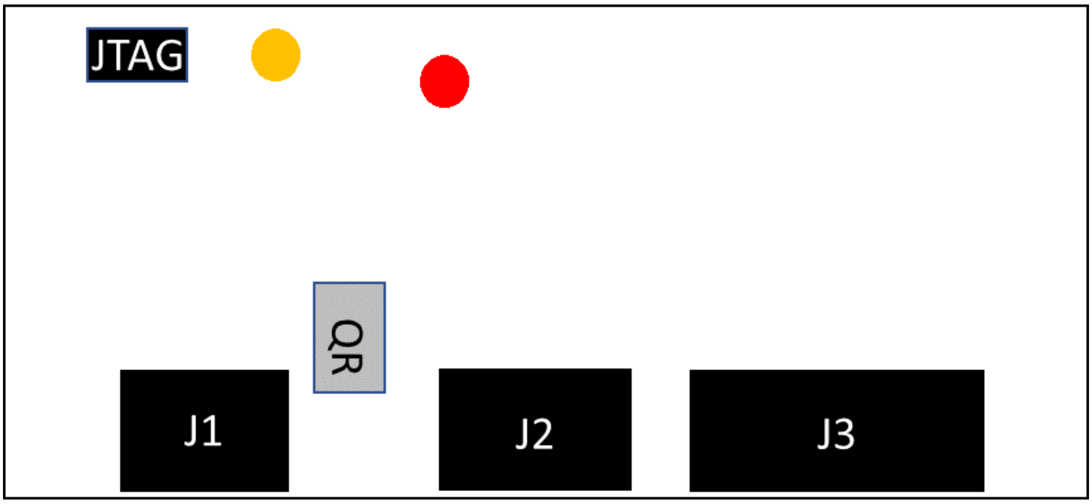


Figure 2.1: c-BMS board overview.

2.1.1 Connectors

The c-BMS board contains three connectors (J1, J2 and J3) for customer interfaces. Furthermore, a JTAG connector is available. This is used for flashing software during production and will in general not be used by the customer.

2.1.2 LEDs

The c-BMS board also contains 2 light emitting diodes (LED).

1. The red LED is indicating whether a hardware fault has occurred (always on) but will also give a very short blip at 'power on' to indicate that power has been connected.
2. The orange LED is indicating operating mode of the c-BMS,
   - A FLASHING ORANGE LIGHT means it is in configuration/boot mode and ready to or in the process of being programmed with either a new configuration or a new application SW (see section 5 for more information).
   - A SOLID ORANGE LIGHT indicates that the application could not be started and the BMS is not operating. This may be because of a defect in the BMS, or it may be because all the necessary components of the BMS are not powered (see section 2.3).

2.1.3 Labels

The QR label (marked QR on Figure 2.1) contains information about:

- Part number and revision version of the board
- Serial number including the production date

To read the QR-labels, Lithium Balance suggests to use dedicated QR readers. If smart phones are used the “QR Reader from TapMedia Ltd” have been tested successfully.

2.2 Connectors and cables

To simplify installation, Lithium Balance provides one cable per connector (J1, J2 and J3). These cables can be ordered together with the BMS boards.

2.2.1 J1 Connector - I/O, power and CAN interface - 16-pin The J1 connector provides interfaces for:

- Digital Input
- Configurable Input/Output
- Board Power and Ground
- CAN communication

The J1 connector can be connected with the 100930 and 100930.1 cables from Lithium Balance.

- On the 100930.1 cable, the CAN connection is provided with twisted pairs
- On the 100930 cable, a separate shielded cable is used for the CAN connection. This cable is is recommended for environments with a significant amount of electrical noise or for system with long cable lengths.

The J1 connector PIN numbers and associated 100930 cable colour coding are summarized in Table 2.1.

PIN No.	PIN Name	100930.x Colours	IN/OUT	Description
1	CAN LOW	Brown + Shielded*	In/Out	s-CAN data signal
2	No Connection			New Dig input for future use
3	GPIO6	Brown	In/Out	Configurable IO signal
4	GPIO5	Green	In/Out	Configurable IO signal
5	INPUT2	Orange	In	Configurable Input signal
6	INPUT1	Blue	In	Configurable Input signal
7	GND	Black	In	Gnd for board
8	IGNITION	Grey	In	Get system out of sleep mode
9	CAN HIGH	White + Shielded*	In/Out	s-CAN data signal
10	No Connection			New Dig input for future use
11	GPIO8	Orange	In/Out	Configurable IO signal
12	GPIO7	Violet	In/Out	Configurable IO signal
13	INPUT4	White	In	Configurable Input signal
14	INPUT3	Yellow	In	Configurable Input signal
15	GND	Black	In	Gnd for board
16	Power	Red	In	(12 V) Power Supply input

Table 2.1: Overview of J1 and 100930 connections. ALWAYS, connect both GND pins 7 and 15 to allow for high relay return current (>1A).

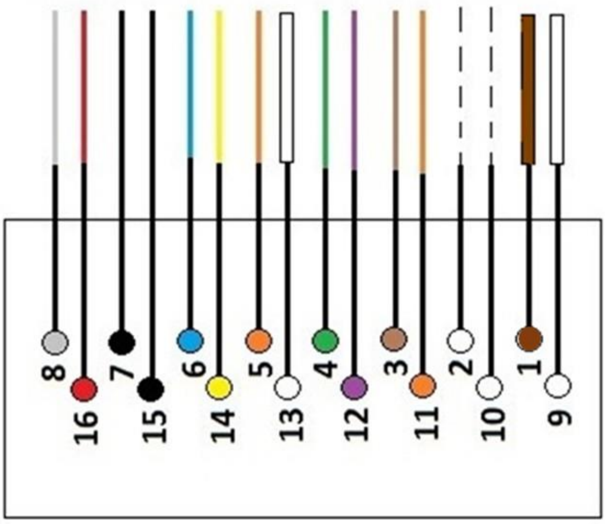


Figure 2.2: Colour coding overview of the cable 100930 used for IO, Power & CAN connections

2.2.2 J2 Connector - Temperature, shunt and hall effect sensor – 20 pin. The J2 connectorprovides interfaces for:

- Battery/Cell temperatures
- Shunt and Hall sensor interfaces

The J2 connector can be connected with thermistors and current sensors via the 100931 cable from Lithium Balance. J2 connector PIN numbers and associated 100931 cable colour coding are summarised in Table 2.2 below.

PIN No.	PIN Name	100931 Color coding	IN/OUT	Description
1	Shunt-	Orange	In	SH-
2	Hall Low	Yellow	In	HALL EFFECT SENSOR LOW
3	Hall High	Orange	In	HALL EFFECT SENSOR HIGH
4	NC			No Connection
5	T6 Gnd	Black	In	TEMP CHANNEL 6 GND
6	T5 Gnd	Black	In	TEMP CHANNEL 5 GND
7	T4 Gnd	Black	In	TEMP CHANNEL 4 GND
8	T3 Gnd	Black	In	TEMP CHANNEL 3 GND
9	T2 Gnd	Black	In	TEMP CHANNEL 2 GND
10	T1 Gnd	Black	In	TEMP CHANNEL 1 GND
11	Shunt+	Grey	In	SH+
12	Hall Gnd	White	Out	HALL EFFECT SENSOR GND
13	Hall 5V	Red	Out	HALL EFFECT SENSOR 5V
14	NC			No Connection
15	T6	Blue	In	TEMP CHANNEL 6
16	T5	Violet	In	TEMP CHANNEL 5
17	T4	White	In	TEMP CHANNEL 4
18	T3	Yellow	In	TEMP CHANNEL 3
19	T2	Green	In	TEMP CHANNEL 2
20	T1	Orange	In	TEMP CHANNEL 1

Table 2.2: Overview of J2 and 100931 connections.

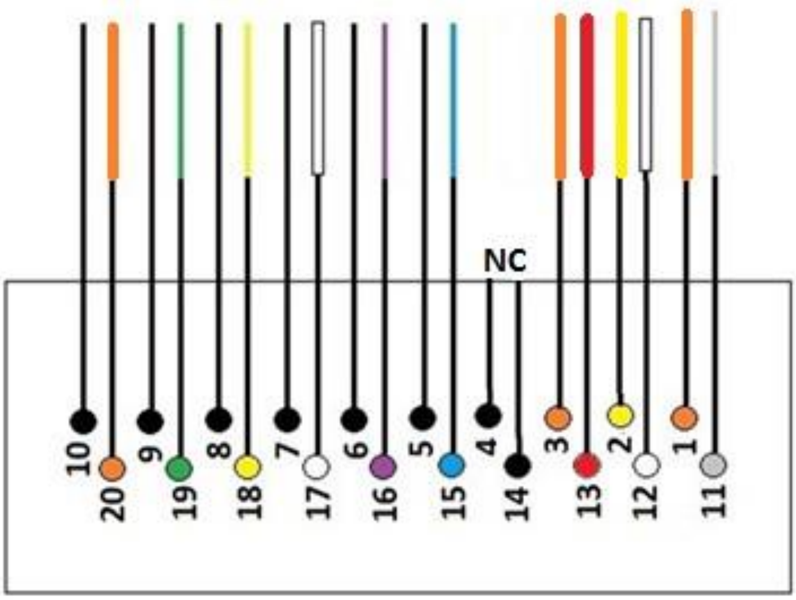


Figure 2.3: J2, Color coding overview of the cable100931 used for connecting Temperature and current measurement connections..

2.2.3 J3 - Cell and battery voltages– 32 pin

The J3 connector provides interfaces for up to 24 cell voltages The J3 connector can be connected to the cells with the 100932 cable from Lithium Balance. J3 connector PIN numbers and associated 100932 labels are summarised in Table 2.3 below.

PIN No.	PIN Name	100931 labels	IN/OUT	Description
1	BATLINK2	Red Wire		For applications supported by future SW releases
2	Pwr 2	Top 12	IN	POWER TO ASIC2
3	Cell 24+		IN	CELL24
4	Cell 22+	Top 10	IN	CELL22
5	Cell 20+	Top 8	IN	CELL20
6	Cell 18+	Top 6	IN	CELL18
7	Cell 16+	Top 4	IN	CELL16
8	Cell 14+	Top 2	IN	CELL14
9	Pwr 1	Low 12	IN	POWER TO ASIC1
10	Cell 12+		IN	CELL12 (ltc1)
11	Cell 10+	Low 10	IN	CELL10
12	Cell 8+	Low 8	IN	CELL8
13	Cell 6+	Low 6	IN	CELL6
14	Cell 4+	Low 4	IN	CELL4
15	Cell 2+	Low 2	IN	CELL2
16	Cell 1-	Low 0	IN	CELL1 low side
17	BAT LINK 1	Red wire	IN	For applications supported by future SW releases
18	Cell 23+	Top 11	IN	CELL23
19	Cell 21+	Top 9	IN	CELL21
20	Cell 19+	Top 7	IN	CELL19
21	Cell 17+	Top 5	IN	CELL17
22	Cell 15+	Top 3	IN	CELL15
23	Cell 13+	Top 1	IN	CELL13
24	Gnd2	Top 0	IN	GND TO ASIC2, see Figure 2 4
25	Cell 13-		IN	CELL13-
26	Cell 11+	Low 11	IN	CELL11
27	Cell 9+	Low 9	IN	CELL9
28	Cell 7+	Low 7	IN	CELL7
29	Cell 5+	Low 5	IN	CELL5
30	Cell 3+	Low 3	IN	CELL3
31	Cell 1+	Low 1	IN	CELL1
32	Gnd 1	(Low 0)	IN	GND TO ASIC1, see Figure 2 4

Table 2.3: Overview of J3 and 100932 connections.

The implications of the ‘Top’ and ‘Low’ labels are described further in section 2.3.8

Please note that

- there are no colouring codes on the J3 cable 100932. Here the wires are labelled with the ‘assignment’ text. To make easier distinction between the ‘Top’ and ‘Low’ wires, the ‘Top’ wires are marked with yellow labels (as indicated above) and the ‘Low’ wires with white labels as also indicated in Table 2.3.
- The length of the 100932 cable is 1 m. Longer lengths of voltage sense cable can be used as the maximum current drawn in the measurement wires is very low (<2μA). Consequently, the voltage drop across any reasonable dimensioned measurement length will be negligible. However, the system is EMC tested with voltage sense cable lengths up to 2 m, which is therefore recommended as the maximum voltage sense cable length.

2.3 c-BMS features and functions

The c-BMS consists of three main blocks, which are galvanically isolated from each other, please see Figure 2.4:

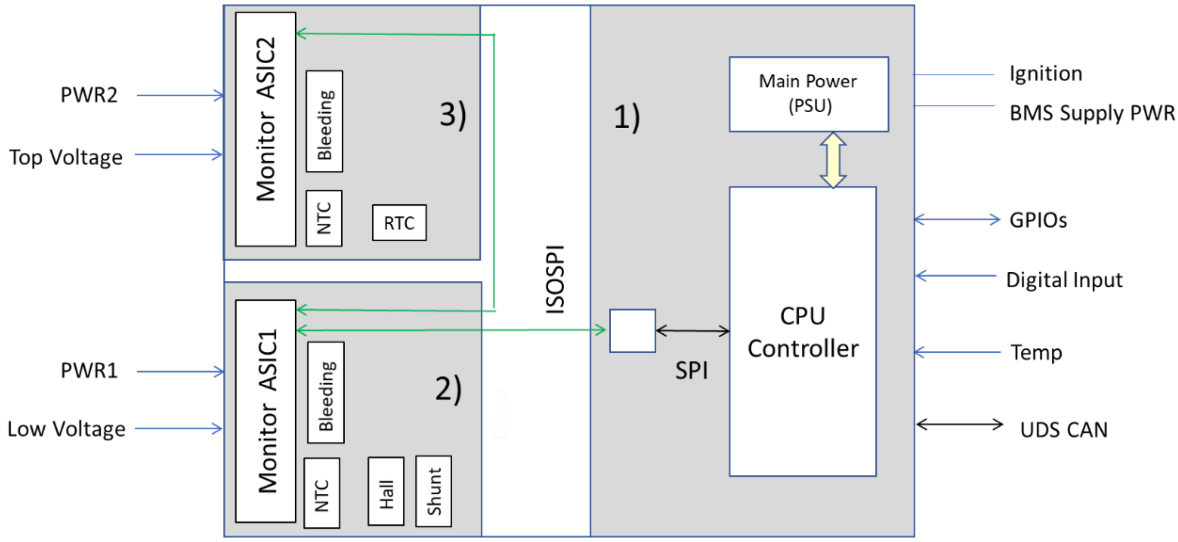


Figure 2.4: c-BMS block diagram. RTC indicates 'Real-Time Clock, please see section 9.13.3. NTC indicates internal (Negative Temperature Coefficient) temperature sensors.

1. The main part which includes the central microprocessor (CPU) and is powered by the main (12V) power
2. A cell measurement circuit (Monitor ASIC1) for the 'Low' voltage cell inputs. This circuit also controls the Shunt and Hall sensors.
3. A cell measurement circuit (Monitor ASIC2) for the 'Top' voltage inputs. This circuit also controls the Real time clock of the system.

Both Monitoring ASICs needs to be powered (PWR1 and PWR2 on Figure 2.4) with minimum 11V from the main battery via the cell monitoring wires[1]. Please see requirements for the cell monitoring connections in section 2.3.8.

2.3.1 Main Power supply (PSU)

The BMS Power (nominally 12 or 24V) and ground needs to be provided to the Main Power Supply Unit (PSU) which provides regulated power to section1 of the board, see Figure 2.4

In many applications, this power is provided from a battery – typically a lead acid battery already used in the system for other purposes. If independent power supplies are not available where the BMS is used, the main power can also be provided from the main battery via a DC/DC converter.

Please note that two ground inputs (Pin 7 and 15) are provided on the J1 connector. Please connect both to ground as they are both used to sink the relative high in-rush current flow into the BMS, which can be experienced when contactors are switched.

The PSU includes an ignition pin which can be used to bring the BMS out of sleep mode, when a voltage >3 V is applied.

[1]If they are not powered, the BMS will not start up and the orange LED will be on.

2.3.2 Powering from main battery

As described above the power supply to the monitoring section is provided by the main battery to the two monitoring ASICs from the 'Low' and 'Top' sections [2]of the main battery. As the two ASICs power other functions e.g. Real Time Clock and Hall sensor, the power usage of the 'low' and 'top' sections of the main battery may vary. An example of typical of power usage of the two sections for a 48 V battery pack is given in Table 2.4.

	Active mode - Shunt	Active mode - Hall	Sleep mode
Top – Cells	< 300 mW	< 300 mW	<1.7 mW
Low Cells	< 350 mW	< 1300 mW	<0.2 mW

Table 2.4: An example of typical of power usage of the 'top' and 'low' section for a 48 V battery pack. In sleep mode, the extra power usage of the top cells is caused by the powering of the Real Time Clock. When using a Hall sensor, the power consumption depends on the specific Hall sensor. The measurements here are for a DHAB s06.

2.3.3 GPIO (General Purpose Input/Output) connections

The BMS supports up to 4 digital GPIOs for example for control of relays.

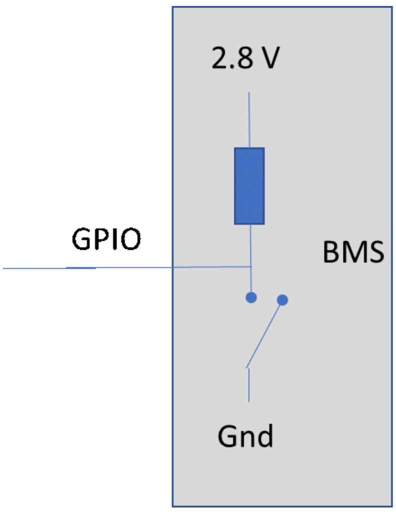


Figure 2.5: BMS IO operating as an actively low contactor driver. In the OFF state, the IO is connected to the V Supply, in the ON state it is connected to ground

These I/Os are low side switched (active low), meaning that they short to the BMS ground (0 V) when active and are pulled up to app 3V when not active as indicated on Figure 3.7.

- In output mode, the I/Os are either in a high impedance mode (off mode) or directly to ground (on mode).
- In input mode, the I/Os will detect a low (active) signal for voltages < 0,5V and a high (not active) signal for voltages between 1 and 30V.

In terms of maximum ratings, the I/Os support:

- Peak currents of up to 4,5A/1sec
- Continuing current up to 1A per I/O
- Operation specified up to 30V with load dump protection up to 52V
- Absolute minimum input level: -0,3V

2.3.4 Digital inputs

The c-BMS supports up to four digital inputs (only inputs), which can be used for example for setting the BMS in a specific mode (e.g charging), to initial balancing and to provide feedback signals from contactors.

Their input levels are characterized by:

- < 0,5V is low and active level
- 1-30V is not active level (high).
- Absolute minimum input level: -0,3V

[2]'Low' and 'Top' are described further in section 2.3.8.

2.3.5 CAN bus connection

A CAN bus connection (so-called s-CAN) to and from the BMS is provided with a 120 Ω termination resistance on the BMS board. The user should assure correct termination in the far end as indicated on Figure 2.6, below.

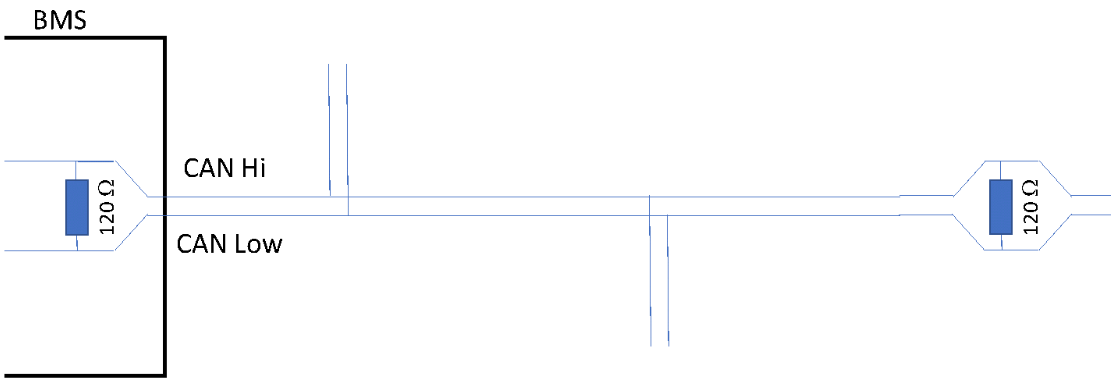


Figure 2.6: Sketch of CAN connection for more details please refer to ISO 11898

The s-CAN bus is galvanically isolated from the battery and is referenced to the BMS GND.

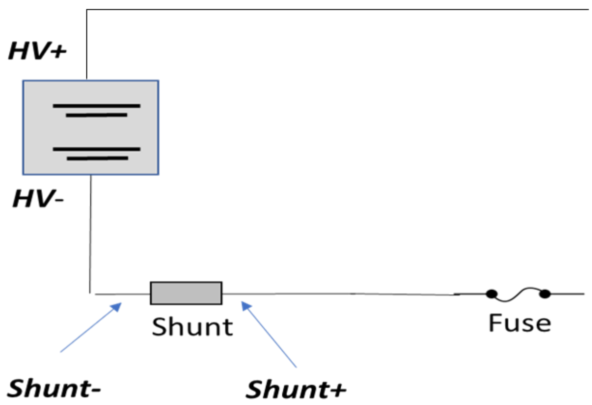
For communication between the BMS and the BMS Creator a CAN adapter is necessary.

Lithium Balance can only guarantee correct operation if the PCAN-USB adapter from Peak systems is used.

Figure 2.7: PCAN-USB adapter from Peak systems

2.3.6 Shunt

The battery current can be measured over a 'Shunt resistor', which can be monitored by the BMS by connecting the Shunt+ and Shunt- signal, as indicated on Figure 2.8, to the BMS (J2 pin 11 and 1). Please note that:



The shunt circuits are internally referenced to the low voltage side of the main battery, the HV- level in Figure 2.8. Consequently, the shunt MUST always be placed on the negative side of the battery pack as close as possible to battery.

The 'shunt-' signal is defined as the voltage connected to the 'most negative terminal' of the battery pack.

Figure 2.8: Shunt and High voltage signal definitions

2.3.7 Hall Sensor

The BMS system current can also be measured by a Hall effect sensor, where:

- Hall sensor voltages > 2.5V correspond to current In = charging
- Hall sensor voltages < 2.5V corresponds to current out = discharging
- Hall sensor voltage = 2.5V corresponds to no current. The 2.5V center voltage can be offset adjusted via configuration parameter.

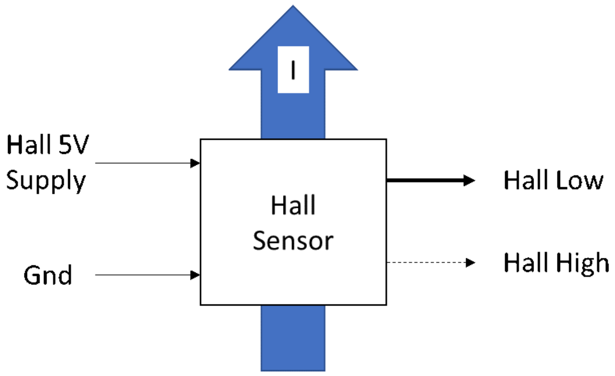


Figure 2.9: Hall sensor input and output signals

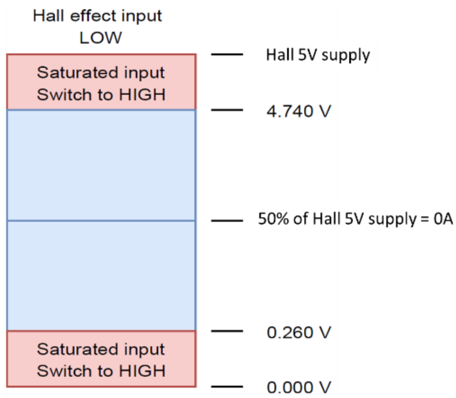


Figure 2.10: Hall sensor voltage levels

The BMS system provides a 5V supply- and Gnd signal to the sensor as shown on Figure 2.9. The BMS system can utilise both dual- and single channel hall sensors.

- If a dual-channel Hall sensor is used, the BMS will always use the ‘low’ current input until this is saturated (the limits are user-configurable). Once the ‘low’ current channel is saturated, the BMS will then switch to use the ‘high’ current input, please see Figure 2.10. As a dual channel Hall sensor, the system has been tested with the DHAB Hall effect current sensor family from LEM.
- If a single channel Hall sensor is used, the ‘Hall high’ input (J2 Pin 3) on the BMS must be connected to the HALL_5V pin (J2 Pin 13). This ensures that if the LOW channel is saturated in either positive or negative (please see Figure 2.10), the BMS will register a maximum input current with a positive sign. Once a Hall sensor has been installed and configured via the BMS creator (see section 5) it is strongly recommended:
- To verify that the Hall sensor has been turned the right way, i.e. that a positive current is measured when the system is charging.
- To set the Hall sensor offset.

2.3.8 Cell voltage connections < >24 cells

When connecting the c-BMS to a battery, it is very important to note that the cell connections are used to provide both monitoring voltages and power to the two monitor ASICs, please see Figure 2.4.

The 100932 cable has been designed to support these functions for systems with >12 cells by internally connecting the c-BMS ASIC power pins to the appropriate cell wires. An example of how to connect the 100932 cable to battery cells are given below for a case with 19 cells, please see Figure 2.11. This approach assures that:

- Both monitoring ASICs gets the necessary power
- Unused cell voltage inputs are terminated correctly

Connection examples for other numbers of cells are found in Appendix/Section .

For correct connection of > 12 cell please connect the lower half of cells to ‘low’ and the upper half to ‘top’. Start connecting ‘from the bottom’, i.e. from ‘Low 0’ and ‘Top 0’ and move ‘upwards’. If the number of cells is unequal, please connect one more cell to ‘low’

1. E.g. if 19 cells are used connect 10 cells to ‘Low’ (Low0 to Low10) and 9 cells to ‘Top’ (top0 to top9), see Figure 2.11

   Make sure to short unused wires on 100932 to the highest connected wire on the ‘Low’ and ‘Top’ side respectively. For 19 cells used:
2. 10 cells are connected to the Low side, so short Low 12 and 11 to Low 10
3. 9 cells are connected to the Top side so short Low 12, 11 and 10 to Top 9

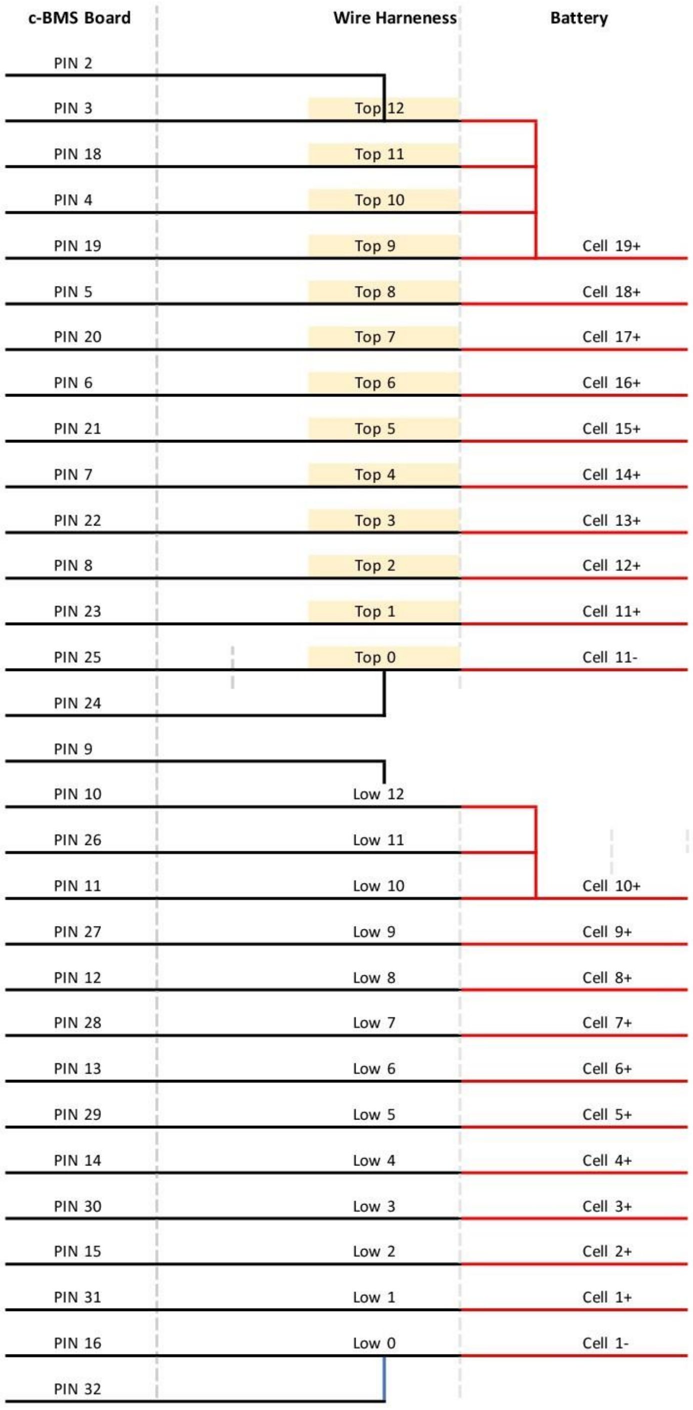


Figure 2.11: Example for 19 cell connection.

2.3.9 Cell voltage connections 12 or fewer cells

When connecting the c-BMS to a battery with 12 or fewer cells please NOTE THAT A MODIFIACTION TO THE 100932 CABLE IS REQUIRED. For 12 or fewer cell, please use the following steps:

- Connect all cells to the low side.
- Connect LOW 0 (GND for ASIC1) with Top 0 (GND for ASIC2)

At the TOP 12 label, **separate the PIN2 wire from the PIN3 wire and**

- Connect all Top sense (1 – 12) wires to Top 0
- Connect the PIN2 wire (ASIC2 PWR) to the low 12 (ASIC1 wire)

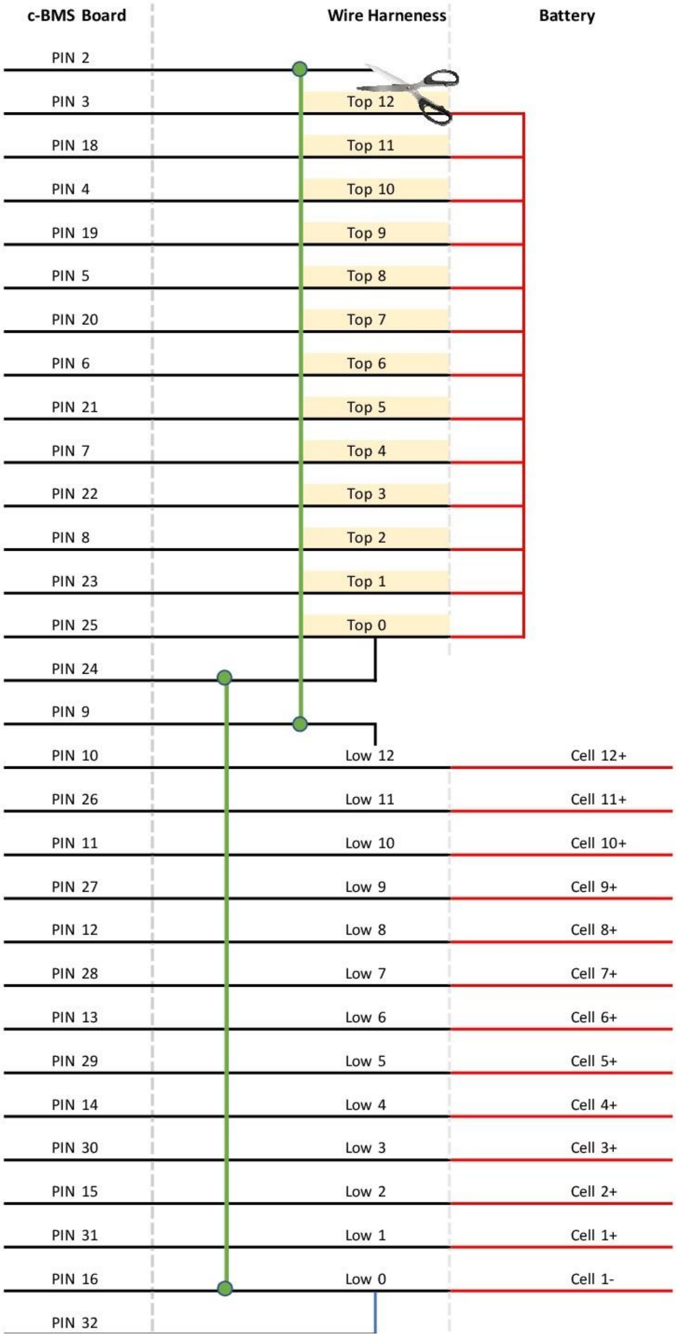


Figure 2.12: Example for 12 cell connection.

An example of a 10-cell configuration is shown in section .

2.3.10 Temperature sensors

The c-BMS can interface up to 6 thermistors (NTC temperature sensors) for monitoring the temperature of the cells and other relevant components (e.g. a shunt). The temperature sensor connections are available on the J2 connector. One end of each thermistor (e.g. #3) must be connected to a temperature input (e.g. T3) and the other end to the corresponding Gnd (in this case Gnd3). Please note, that it is not important which end of the thermistor that is connected to the Gnd pin.

The temperature sensor detection system is designed for 10 kΩ NTC sensors with a default β-value of 3900. However, for increased support, the resistance-to-temperature characteristics are user-configurable through a set of parameters. NTC configuration uses Steinhart-Hart coefficients for best resistance to temperature calculation. Separate calculation tool available for coefficients determination.

Furthermore, the c-BMS board has 4 internal thermistors for measuring the board temperature. These are used to control the PCB temperature, so the cell balancing (bleeding) can be turned off the bleeding, when the board temperature gets too high. The bleeding temperature limit is 90˚C and the bleeding will be turned on as soon as the board temperature again gets below 90˚C.

2.3.11 Bleeding circuit

Bleeding the current away from individual battery cells is a key BMS function which ensures that when a battery is charged, all cells will be charged to their maximum capacity (100% SoC). For the c-BMS, bleeding is done by switching in a bleeding resistor between the '+' and '-' side of the individual cell. This is shown schematically in Figure 2 13 where cell N is bleeding, while cell N+1 is not.

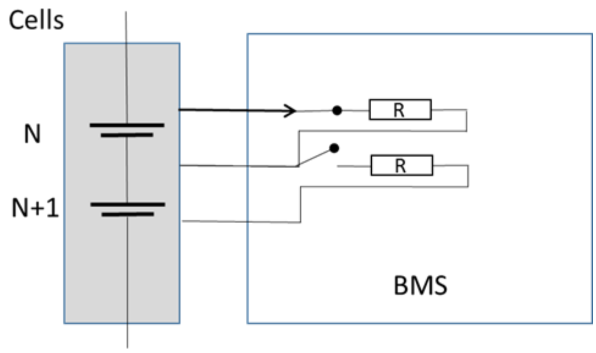


Figure 2.13: Bleeding principle

The maximum bleeding current for each cell is 200 mA and when many cells are bleeding at the same time, the BMS board may heat up. For optimal balancing performance, it is therefore important that adequate ventilation/cooling is provided to the 'back' side of the board, where the bleeding resistors and transistors are placed.

3 BMS Functionality

This section describes the main functionality of the BMS including the key parameters necessary for the configuration of the BMS.

3.1 Activating primary measurements

At the battery cells, the BMS measures cell voltages, external temperatures, and pack current.

3.1.1 Current

The pack current can be measured either by a current sensing SHUNT, or a HALL effect sensor, or CAN based current sensor. These sensors types are described further in section and section . Please note that by definition Current In = Charging gives positive current levels for a correctly configured system, whereas discharging gives negative current levels.

Both shunt and Hall sensors have to be configured in the BMS Creator (System Configuration pane) with respect to measured-Voltage-to-displayed-current conversion, please see 'Shunt resistance' and 'Hall sensor scaling' in Figure 3.1.

The BMS allows the use of hall sensors with a single or dual channel ('high' and 'low' current outputs, please see section 2.3.7). If a single output Hall sensor is used, the two Hall sensor scaling MUST BE configured with the same gain in BMS Creator.

Furthermore, the possible (0-input) offset of the shunt and Hall sensors can be corrected. This is often required for Hall sensors. In this case the offset can be adjusted after the scaling factors have been written from the BMS creator to the BMS board:

- Make sure there is no current in the system and then read the Hall sensor current (14 ID_PACK_HALL) as listed in Appendix: Data ID Map.
- Insert the offset value (in mA) via BMS creator to get the current level to 0. Note that offset values are 'added', meaning that If the BMS displays a negative current, write a corresponding positive offset and vice versa.
- Write the new configuration to the BMS and check that the current reading is now 0

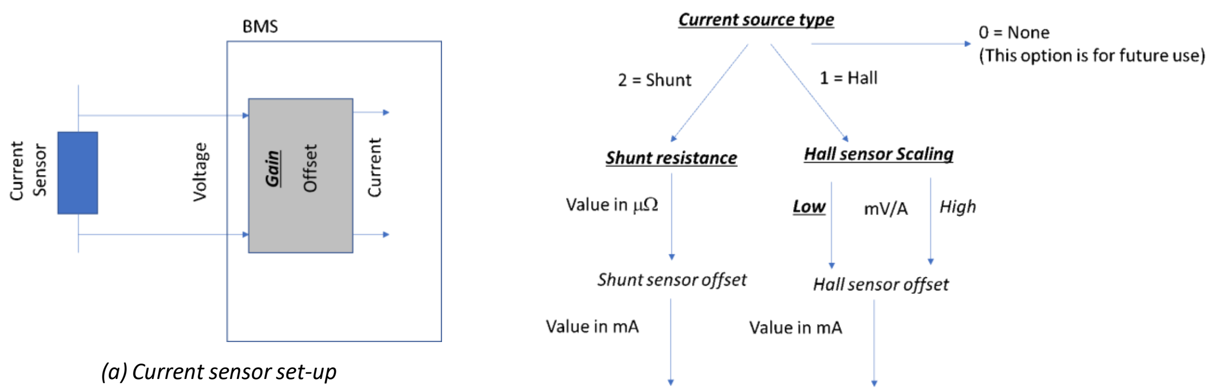


Figure 3.1

3.1.2 Temperature

The actual temperature inputs used by the BMS needs to be activated in the BMS Creator. This is done by assigning one bit per sensor and to set the bit to 1 if the input is used and to 0 if it is not used. For example, to activate temperature sensor 6, 3 and 2, the relevant binary number is 100110 which corresponds to the decimal number 38. This should then be inserted in the "Temp. sensors enabled" field of the "Operational limits" pane of the BMS Creator.

Temp sensor	6	5	4	3	2	1
Used	Yes			Yes	Yes	
Binary	1	0	0	1	1	0
Decimal	38					

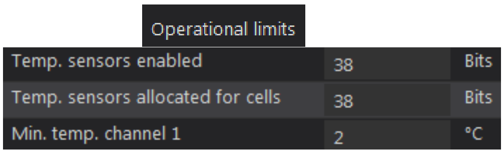


Figure 3.2: Configuration example for activating temperature sensors

The c-BMS is designed to measure up to 6 temperatures sensors.

The temperature sensor inputs can be configured to behave as cell temperature inputs or AUX temperature inputs. When the temperature inputs are configured (bit set to '1') as cell temperature inputs via the 'Temp. sensors allocated for cells' parameter, are configured min. and max cell temperatures used for when associated errors occur. As a summary, user can define how many out of 6 available temperature sensors are assigned for cell temperature, and how many other temperature sensors are not assigned for cell temperatures but assigned for other units, for example shunt or inverter's temperature. Ifa temperature channel is allocated for cells, BMS will take the cell temperature settings into account when evaluating the temperature for that particular channel. Else, BMS will take "Min/Max. temp. channel 1-6" limits set by user in BMS creator.

To enable cell temperature functionality the sensor channel must be enabled, and the cell temperature channel must correspond to the temperature channel.

Note: It is **necessary to have at least one cell temperature** configured to be able to use features depending on cell temperatures, such as dynamic current limits, state of power calculation and so on.

3.1.3 Cell Voltage

The BMS is designed to measure a configurable number of cell voltages. This is done by enabling or disabling corresponding cell number in BMS creator in CMU Config menu.

Figure 3.4: Configuration example for activating voltage sensors

The c-BMS is designed to measure up to 24 cell voltages. The BMS can also work with less than 24 cells connected and therefore needs to be configured via the BMS Creator to know the actual number of cell voltages to read.

This is done independently for the two voltage sensor ASICs (referred to for this purpose as CMU1 = Low and CMU2 = Top). CMU1 holds the settings of first 12 cells of C-BMS, while CMU2 holds the settings of cell number 13 to 24.

3.2 Safety and thresholds of cell parameters

If a lithium ion battery is operated outside the specified limits for current, voltage or temperature, the battery may be permanently damaged or in extreme cases catch fire. The Lithium Balance BMS therefore constantly monitors battery and cell current, voltage, and temperature to prevent the battery from operating outside its safe operating area. Please note that the safe operating area depends on the lithium cell chemistries and the application. Consequently, safety thresholds must be configured by the user.

If any primary parameter exceeds a safe operating limit, the BMS will do the following:

- Raise an error signal corresponding to this specific parameter. E.g. via CAN
- If this error signal persists, the BMS can open the contactor to block the battery current and get the system into a safe operating domain.

For the BMS, the voltage, current and temperature safety thresholds can be configured freely to allow the use of the BMS for all lithium cell chemistries and for all applications.

Please note that setting these thresholds correctly is essential to the safety, lifetime and energy storage capacity of the battery. Parameter settings must be based on the requirements of the specific application and the battery information provided by the battery manufacturer, e.g. in datasheets.

3.2.1 Operational Current limitation

The BMS measures the current running through the battery to ensure that the current is kept within user specified limits. This is required to protect both battery and auxiliary components like switches, fuses and cables.

If the BMS detects that the actual current is beyond the current limits it will immediately raise an error and, depending on configuration also open contactors.
The BMS operates with three current limits that are used for thresholds:

- Maximum current in to the battery.
- Maximum current out of the battery.
- Maximum i2t sum for peak currents.

The first two current limits are continuous current limits which means they are used in most situations as limits while BMS is operating.

The third current limit called “Max. i2t” is used to allow short periods of peak currents beyond the continuous current limits, this is explained in an individual subsection 3.2.1.2 Advanced current limitation, i2t.Advanced current limitation, i2t.

When performing checks for current limits against the thresholds the BMS will allow for small variations before triggering an error:

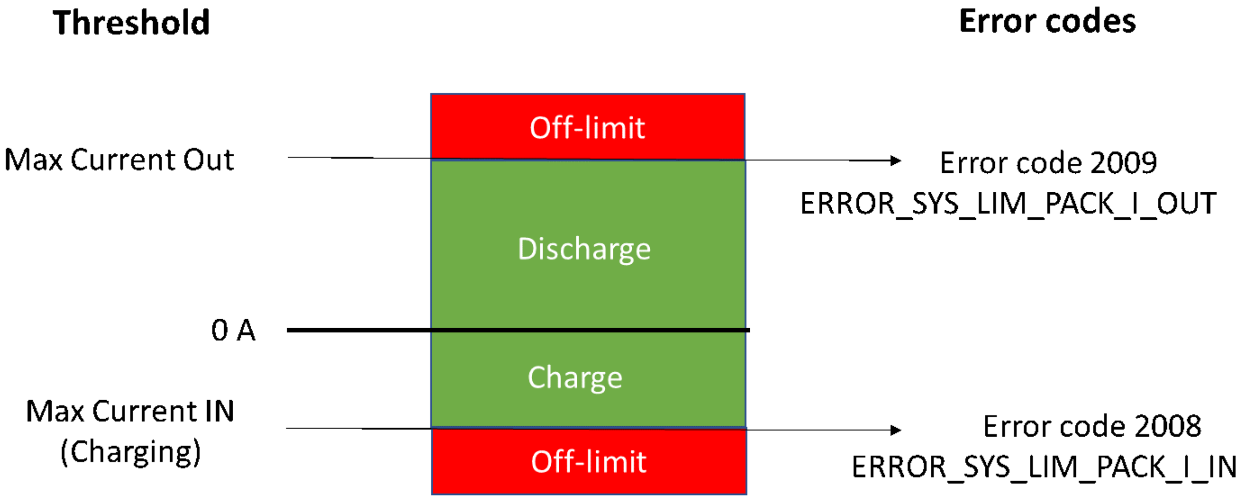
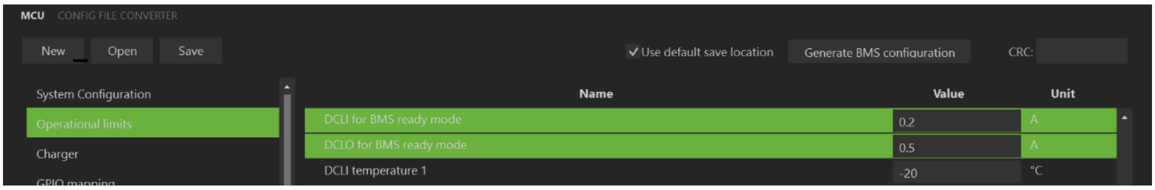


Figure 3.5: Overview of safety thresholds relating to the battery current

- If the BMS is in Ready mode (see section s3.4.3), the dead bands are configurable as follows:



- In active state the dead bands are:
  - Current in, continuous mode: <u>none</u>.
  - Current out: <u>none</u>.

3.2.1.1 Continuous current limits

The BMS defines two configurable data arrays for continuous current limits:

1. Dynamic current limit in (DCLI)
2. Dynamic current limit out (DCLO)

These current limits can be broadcasted on the CAN bus to make sure other units are performing in accordance with current limits from the battery manufacturer. The BMS will generate an error if the current limits are exceeded.

The BMS uses two data tables, one for DCLI and another for DCLO. The tables have two dimensions: one dimension being the SOC, and the other the cell temperatures. There are 7 custom temperatures to define, and 11 slots for the SOC. The SOC slots are not user-defined. There is a slot for each 10 % SOC, starting at 0 % SOC.
A graphical representation is shown for both DCLI and DCLO in Figure 3.6.

	0 %	10 %	20 %	30 %	40 %	50 %	60 %	70 %	80 %	90 %	100 %
-20 ºC	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A
-10 ºC	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A	0.0 A
-5 ºC	5.0 A	5.0 A	5.0 A	5.0 A	5.0 A	5.0 A	5.0 A	5.0 A	5.0 A	5.0 A	5.0 A
0 ºC	50.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	80.0 A	10.0 A
15 ºC	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	80.0 A	10.0 A
25 ºC	150.0 A	150.0 A	150.0 A	150.0 A	150.0 A	150.0 A	150.0 A	150.0 A	150.0 A	90.0 A	10.0 A
45 ºC	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	100.0 A	80.0 A	10.0 A

Table 3.1: Example of DCLI configuration table. The top row is fixed, while entries in other rows are user-defined.

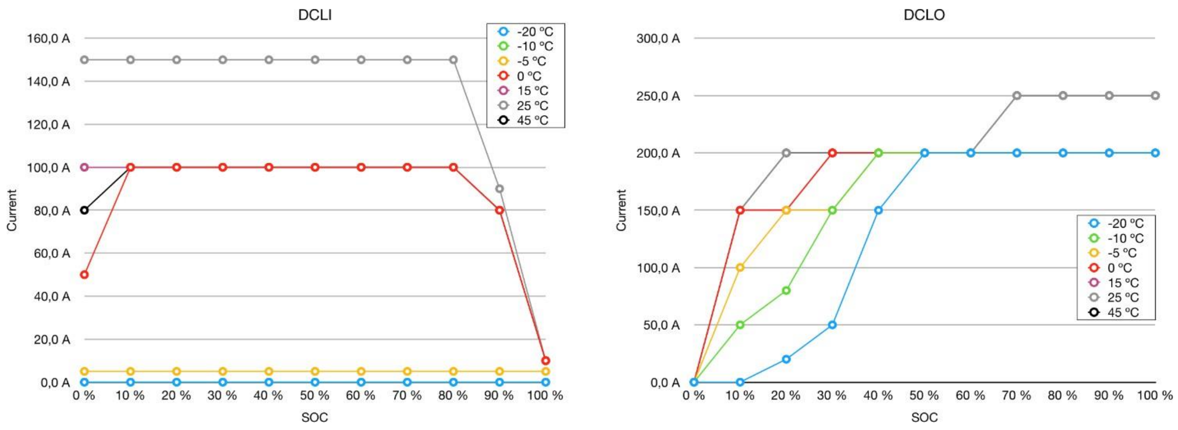


Figure 3.6: Example DCLI and DCLO curves. Values are examples only.

The BMS interpolates data between the datapoint in the tables, e.g.: When SOC is at **85 %** and measured temperature is at **12 ˚C**, the 4 datapoint to interpolate are as shown in Table (based on Table).

	80 %	90 %
15 ºC	100.0 A	80.0 A
5 ºC	150.0 A	90.0 A

Table 3.2: Cross-section of example DCLI table.

Interpolation then returns the result of DCLI = **99.0 A** as seen on the figure on the right. Since the temperature may differ across a battery pack the BMS checks whether the current is limited by the highest temperature or the lowest temperature.

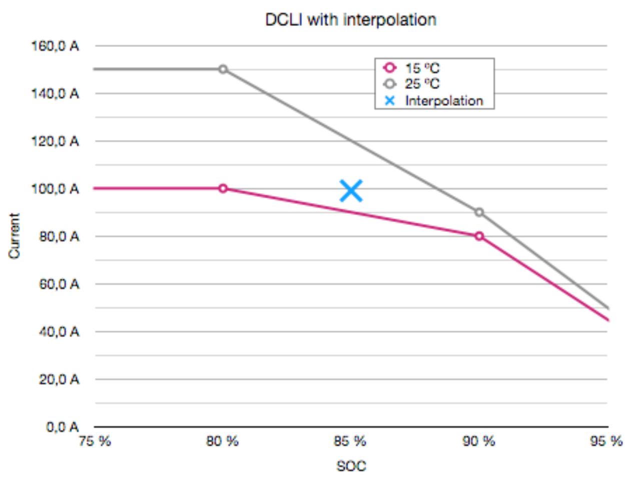


Figure 3.7: Interpolation example for DCLI

NOTE: If the DCLI table specifies 0.0 A in certain SOC/temperature scenarios, the BMS will not be allowed to request a charge current. This enforces the requirement that DCLI data must be greater than 0.0 A if the BMS should be allowed to charge.

NOTE: There are many parameters BMS takes into account while calculating charge current request. The charge current is considered as IN current. For that reason, BMS charge current request algorithm is aslo taking dynamic current limit IN (DCLI) data into account. The charge current request calculated by BMS follows the limits of DCLI, which is a fucntion of temperature and SOC. In case user defined DCLI as, for example, 0 Ampere for 95% SOC and 45 C-degree, BMS will request 0 A from the charger. It is because the charge current request algorithm cares what DCLI limits are. The charge current request cannot be higher than what DCLI is.

3.2.1.2 Advanced current limitation, i2t

The BMS calculates the "i2t-parameter', which can be considered as a digital implementation of a melting fuse. The intention is to assure that the BMS can break the battery current if it significantly exceeds the specified maximum current in or out levels for extended periods of time.

The I^2t parameter reflects the 'accumulated' excess power provided by the battery. By excess power is here understood that:

- The system is specified for a max. current out and a max. current in, any current exceeding these limits are considered 'excess' current.
- The electrical power is proportional to the square of the current. Hence, the BMS constantly calculates the accumulated excess power $(I-I_{max})^2$
  - If the actual current is **larger** than the current limit, the excess power is **added** to I^2t
  - If the actual current is **smaller** than the current limit, the excess power is **subtracted** from I^2t, until 0 is reached

The I^2t parameter is probably best understood through an example. Consider a forklift where the maximum currents are:

- Max. Current In (Charging) = 75A
- Max. Current Out (Discharging) = -100A

According to the system design an excess current of 200 A (i.e.,275 A during charging and 300A during discharging) is considered acceptable for 2.5 seconds. Consequently, the I2t threshold is set to $(200A)^2$ x 2.5s = 100.000 A^2s The forklift is now performing the following actions, please also refer to Figure 3.8.

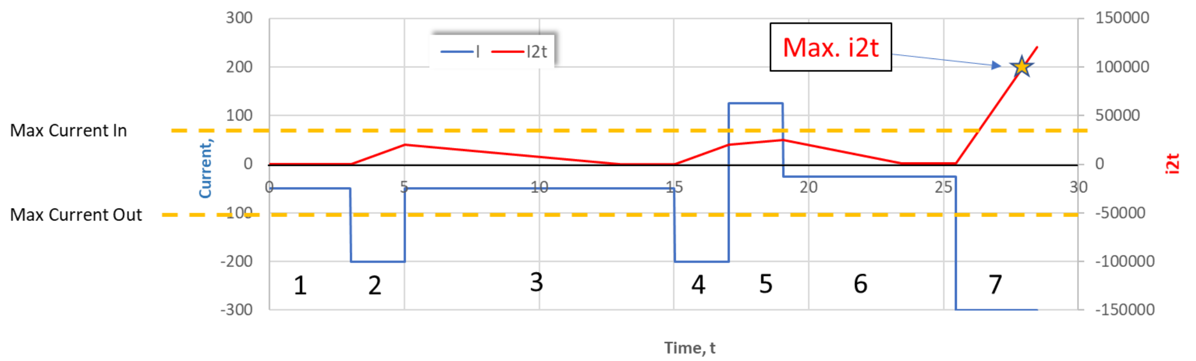


Figure 3.8: Forklift example for the i2t parameter

1. Initially the forklift drives around picking up a load. This requires only 50A which is smaller than the Max Current Out of 100A. As the I^2t parameter is already 0, it stays there.

2. A load is now lifted to a shelf which require 200A. This exceeds the Max Current Out and the I^2t parameter therefore increases.
3. The forklift picks up a new load. This requires only 50A which is smaller than the Max Current Out and the I^2t parameter therefore decreases to 0 and then stays there.
4. A load is now lifted to a shelf as under point 2.
5. The driver regrets and lowers the load again. In this process 125A of regenerative power I generated. This exceeds the Max Current In of 75A and the I2t parameter therefore continues to increase, though at a lower slope.
6. The forklift picks up a new load and the I^2t parameter decreases to 0.
7. A **heavy** load is now lifted to a **high** shelf which require 300A for several seconds. The I^2t parameter therefore increases significantly. In this case the I^2t threshold of $100.000A^2$s is passed and the "2010 ERROR_SYS_LIM_PACK_I2T" error is activated by the BMS.

3.2.2 Temperature

The external temperatures measured by the BMS can be configured for temperature maximum and minimum thresholds as shown in Figure 3.9. Here the max and min temperature of the individual temperature sensor can be configured independently in the "Operational Limits" pane of the BMS Creator.

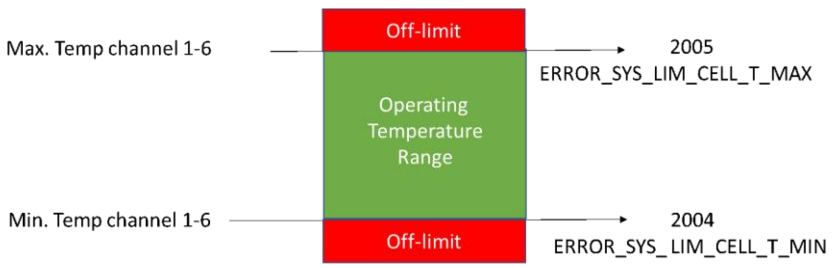


Figure 3.9: Overview of safety thresholds relating to the cell temperatures.

3.2.3 Cell voltage

To achieve reliable and robust operation of a battery system, it is essential to always keep the voltages of all lithium cells within well-defined limits.

- Too low cell voltages may lead to chemical changes in the battery electrolyte which will degrade the battery and potentially produce a combustible gas.
- Too high cell voltages may cause the positive electrode to decompose which can lead to degradation and cause excessive heating of the cell.

The maximum and minimum thresholds for safe operation can be configured in the "Operational Limits" pane of BMS Creator. Two other voltage levels to configure are the "Cell Voltage Target" and the "Charge Complete dead band". When charging, the system adjust charger and balancing current levels to assure that at the end of charging all cell voltage are at the "Cell Target Voltage" with an accuracy defined by the "Charge Complete dead band". These two parameters are configured in the "Charger" pane of the BMS Creator

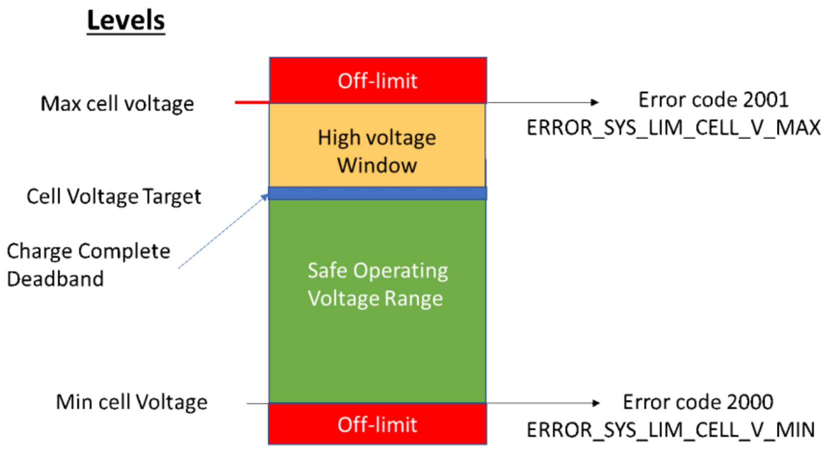


Figure 3.10: Overview of safety related cell voltage settings.

3.3 Derived metrics = Calculated parameters

The BMS uses the measured data to quantify different operational parameters of the battery.

3.3.1 Cell voltage statistics

Every time the cell voltages have been updated, update interval is configurable, the BMS will calculate statistics for every measured cell, this includes (please see Appendix: Data ID):

- The minimum cell voltage (ID_CELL_V_MIN_VAL)
- The maximum cell voltage (ID_CELL_V_MAX_VAL)
- The average cell voltage (ID_CELL_V_AVG)
- The voltage sum of measured cells (ID_PACK_V_SUM_OF_CELLS)
- The number of cells measured (ID_CELL_V_NUM_AVALIABLE)

3.3.2 Capacity and SoC

State of charge (SoC) is a key battery parameter describing the electrical charge (Q = current x time) left in the battery before charging is required. SoC is the battery equivalent of a fuel gauge in a traditional vehicle. The unit of SoC is percentage points (0% = empty; 100% = full).

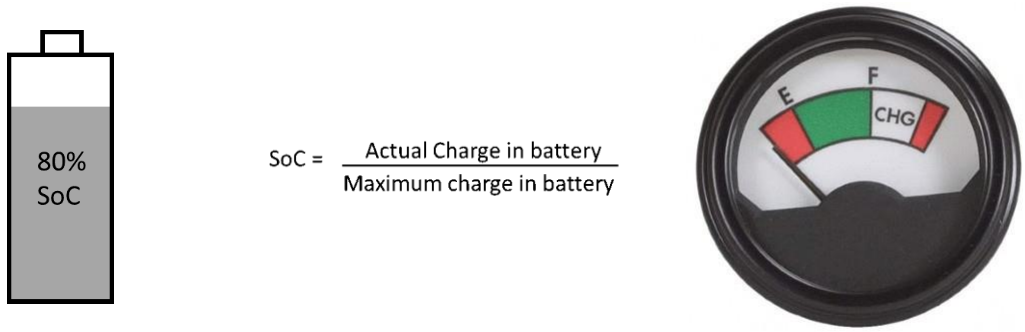


Figure 3.11: SoC describes the relative amount of energy left in the battery and can be seen as the equivalent of a fuel gauge in a car

The state of charge is measured by the BMS using a method known as Coulomb counting. Every time the pack current has been measured (either by SHUNT or HALL effect sensor or CAN current sensor) the number of ampere-seconds will be added or subtracted from a capacity counter called ”remaining capacity”.

When the BMS has detected a fully charged battery (see section 3.3.3) it will calibrate the ”remaining capacity” so it is equal to the expected full capacity (this is typically the same as the design capacity for new batteries). The ”remaining capacity” is always calculated with a high resolution of 1 ampere-second, and also available in a lower resolution of 0.1 ampere-hour.

The high resolution ”remaining capacity” is used to calculate the state of charge (SoC) of the battery, this is done by dividing with the expected full capacity. Thus the SoC is a measure of how close the ”remaining capacity” is to the expected full capacity.

The SoC is available in an internal version which ranges from -300 % to +300 %, but also available in a trimmed variant where the value can be a scaled version of the internal SoC. E.g. if the trimmed SoC is set up to be between 10 % (min) and 90 % (max) of the internal SoC it will display 0 % when the internal SoC is at or below 10% and 100 % when the internal SoC is at or above 90 %, see Figure 3.12 for a visual representation.

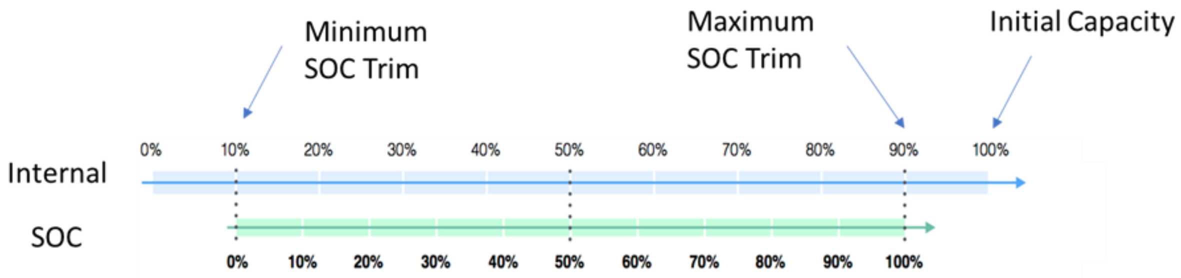


Figure 3.12: SoC mapping between internal and trimmed data.

The “Initial Capacity”, “Minimum SOC Trim” and “Maximum SOC Trim” are all configured in the “System Configuration” Pane of the BMS Creator.

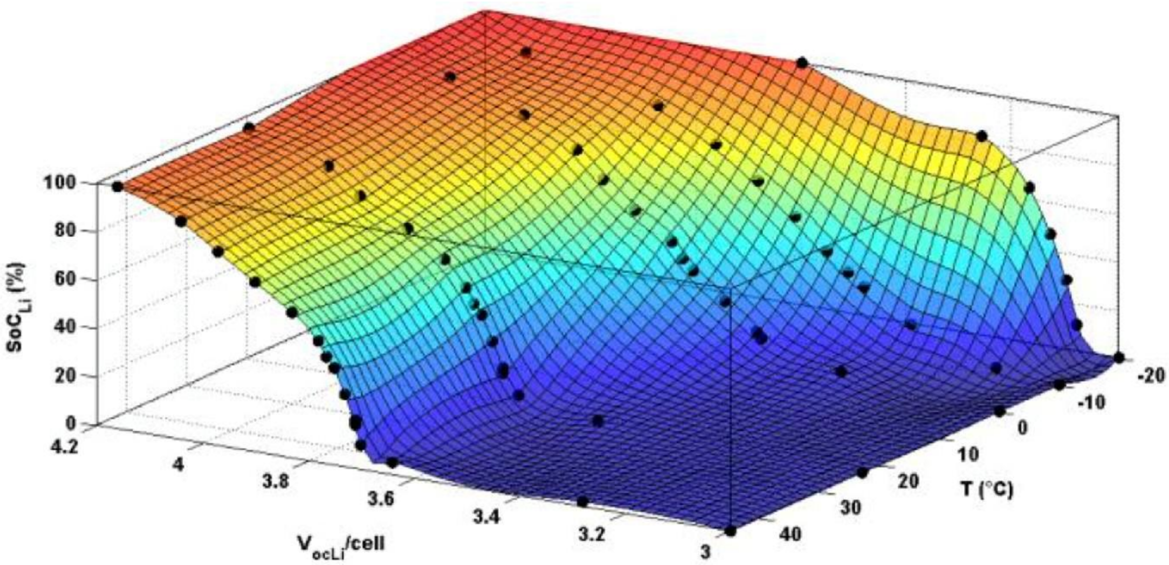
3.3.3 Open circuit voltage SoC calibration

In certain scenarios, the state of charge (SoC) calculation from coulomb counting can become inaccurate and may need to be calibrated. The BMS can then calibrate the SoC in two scenarios:

1. At end of charge when all cell voltages have been sufficiently balanced. For end-of-charge calibration the BMS will ensure that each cell voltage is within a certain threshold of the target cell voltage, while the charge current must be below another threshold.
2. After a period of inactivity with no power being drawn from, or delivered to, the Li-ion cells. This is achieved by using the relationship between the open circuit voltage (OCV) of the Li-ion cells and their SoC. Due to the electrochemical processes the OCV will be higher when the battery is charged (100 % SoC) than it is when discharged (0 % SoC). As the temperature can influence the OCV this parameter is considered when performing the calibration. This method will be used to calibrate the SoC, when the BMS is powered up (unless functionality is disabled by user).

The OCV can be considered equal to the terminal voltage of the cell when enough time has passed and the change in voltage (dV/dt) is lower than a certain threshold. The effect of time is used as a quality factor for the SoC calibration, where “time” means the duration between last power-off and the current power-on[1]. The quality factor is then defined as a value in the range of 0-100 % depending on the time needed for the voltage at the cell terminals to reach OCV state.

The data for the OCV vs SoC vs temperature is unique for different Li-ion chemistries, manufacturers and sizes of batteries and must be obtained from the manufacturer or from experiments. The BMS can work with one to five datasets at different temperatures. The figure below shows a graphical representation of five datasets (black dots) for OCV vs SoC at the temperatures: -20 °C, -10 °C, 0 °C, 23 °C, 45 °C.



The SoC selected for calibration ($SOC_{calibrated}$) is found by interpolation in the 3-dimensional data set (OCV, SoC, temperature), and the quality factor (k) determines the significance of the calibration SoC (SOC_{OCV}) with regards to the originally coulomb-counted SoC ($SOC_{original}$l):

$$SOC_{calibrated} = SOC_{original} + k \cdot (SOC_{OCV} - SOC_{original})$$

The BMS uses minimum OCV and minimum temperature when calculating the SOC. The BMS can be configured to a maximum quality factor (k) of less than 100 % in case the calibration should never rely fully on the OCV factor. See appendix 8.5 for an example of an experiment to determine the configuration values. Lowest cell voltage is used for OCV together with the lowest temperature in the calculation oof $SOC_{calibrated}$

[1]It is assumed that the Li-ion cells have been at rest while the BMS has been powered off.

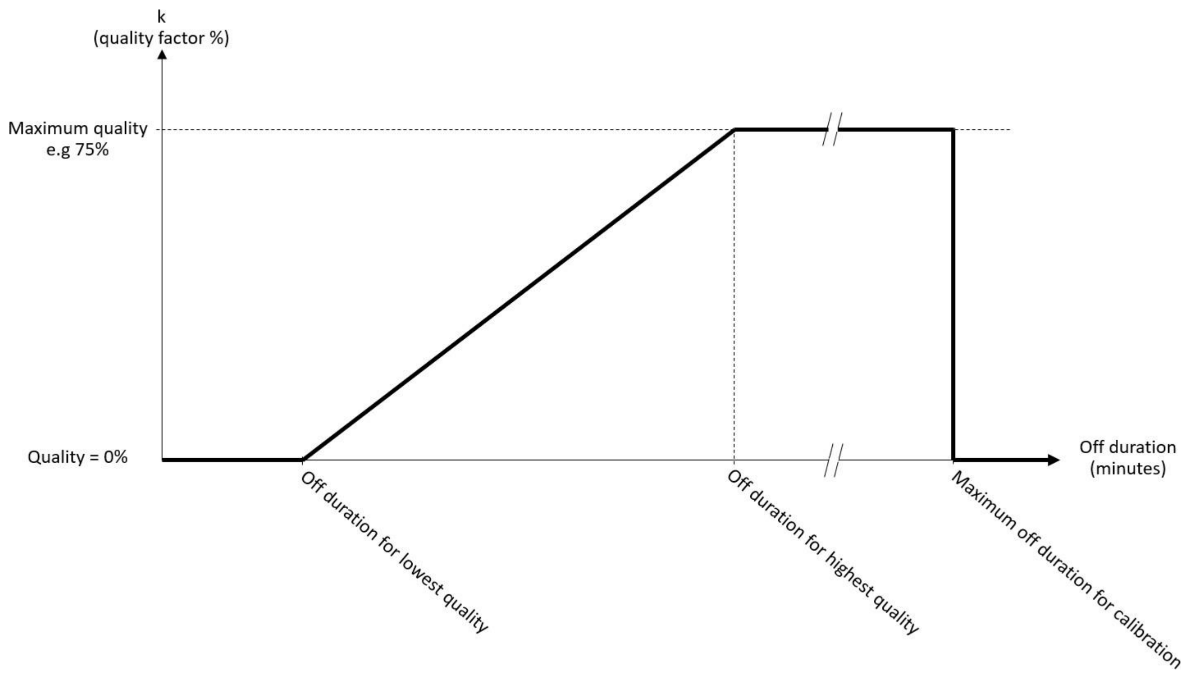


Figure 3.13: k-factor explanation.

3.3.4 State of health

State of health (SoH) is a percental measure of performance degradation of a battery pack. As Li-ion batteries age their performance will decrease. The performance degradation can primarily be divided into two categories: Internal resistance increase and capacity fade. SoH is thus a function of SOH_R, the part related to internal resistance, and SOH_Q, the part related to capacity fade, as shown in Equation 3.0. SOH_R is currently not considered for SoH.

$$SOH = f(SOH_R, SOH_Q)$$

Equation 3.0: SoH as a function of internal resistance and capacity fade.

3.3.4.1 Determining SOH_Q

The capacity fade phenomenon is caused by active material becoming electrochemically inactive and no longer contributing to the processes inside the battery. This will have the effect of decreasing the available capacity of the battery, Q_{full}. To quantify the capacity degradation, the State of Health for capacity (SOHQ) is defined as in Equation 3.0.

$$SOH_Q = \frac{Q_{full}}{Q_{nom}}$$

Equation 3.0: Capacity part of SoH.

Q_{nom} is the nominal capacity, typically as specified on the datasheet. Q_{full} has the relationship shown in Equation 3.0. Q_{SoCOCV} is the SoC as determined from the OCV-SoC calibration (see section 3.3.3).

$$Q_{full} = f(Q_{SoC_{OCV}}, Q_{cc})$$

Equation 3.1: Determining full capacity based on remaining current and SoC calibration

Q_{cc} is the coulomb count since last full charge. The conditions for performing an SoH calibration are as follows.

1. The BMS off duration is such that OCV-SoC calibration is performed with the maximum OCV-SoC calibration quality.
2. The actual SoC is within one of two user-configured SoC windows.

The OCV-SoC calibration, which forms the basis of the SoH estimation, is only performed when the BMS has been powered down for at least the user-configured relaxation time of the battery cells, but not long enough for the cells to have experienced significant self-discharge.

Because the SoH estimation depends heavily on the correctness of the OCV-SoC calibration, the confidence in the quality must be at a maximum to allow SoH calibration. This means that the maximum relaxation time of the cells (BMS off time) must be met. Also, because the OCV-SoC curve may contain sections where small errors in voltage causes large errors in SoC, two windows of the SoC(V) curve where Q_{full} calculation is allowed are specified as shown in Figure 3.14.

Figure 3.14: Example SoC(V) curve with windows where SoH calibration is allowed.

Initially, when there is no SoH estimation available in the BMS, Q_{full} is set equal to Q_{nom}, the nominal capacity of the battery pack. Then, when the conditions above are met, Q_{full} is corrected from Equation

k is a user-configured weight that determines how much trust to put in the most recent estimation

$$Q_{full} = (1 - k)Q_{full} + k{\cdot}f(Q_{SoC_OCV}, Q_{cc})$$

Equation 3.2: Qfull weighted equation

of Q_{full}. The "k" value for each window is same for both the windows and it can be configured with the help of "SOH SOC calibration window 1, weight" parameter. The recommended value for this parameter is 0.1. The new value of SOH_Q is then updated as in Equation 3.0. Additional information about configuring the SOC-SOH parameters are described in this document present in our knowledge base.

3.3.5 State of power

State of power (SoP) is a term denoting the constant power available for use at various intervals, e.g.

1. Continuously
2. The next 3 seconds
3. The next 10 seconds
4. The next 30 seconds

This has many applications, such as determining whether a quick overtaking in an electric vehicle is feasible (e.g. power available the next 10 seconds). SoP is dependent on factors such as cell temperatures and SoC. The BMS allows for fully configuring an SoP table for user-defined temperatures and SoC as shown in Table 3.3.

	SoC X	...	**SoC Z**
Temperature X	Interval 1	...	Interval 1
	Interval 2	...	Interval 2
	Interval 3	...	Interval 3
	Interval 4	...	Interval 4
...	...	...	...
Temperature Z	...	...	...

Table 3.3: Table structure for SoP.

For each intersection between a user-defined temperature and SoC, the corresponding SoP value for up to four intervals can be specified. The intervals corresponding to the current temperature and SoC can be output by the BMS.

Temperatures and SoC specified must be in increasing order, that is “Temperature X” must be of lower value than “Temperature Z” and likewise for SoC.

The BMS performs the operations for both minimum and maximum cell temperatures and calculates the most conservative (lowest value) outputs. If one or more cell temperature sensors are diagnosed as faulty, the SoP will be zero for all intervals.

If the actual cell temperatures and SoC do not match the configured values, the BMS will linearly interpolate between configured values.

The data for the SoP table is typically provided by the battery manufacturer and can be entered directly in the available units of intervals.

3.4 Operating modes and state machine

The BMS is designed as a state machine which can be in a number of different modes as shown in Figure 3.15. Before entering a new state, an error checklist is verified to allow access to the new state:

- If no errors on the relevant checklist are present , the BMS will enter the error mode (see section 3.3.3 and 3.4.3).
- If no errors are present, the BMS will move to the next level in the state machine

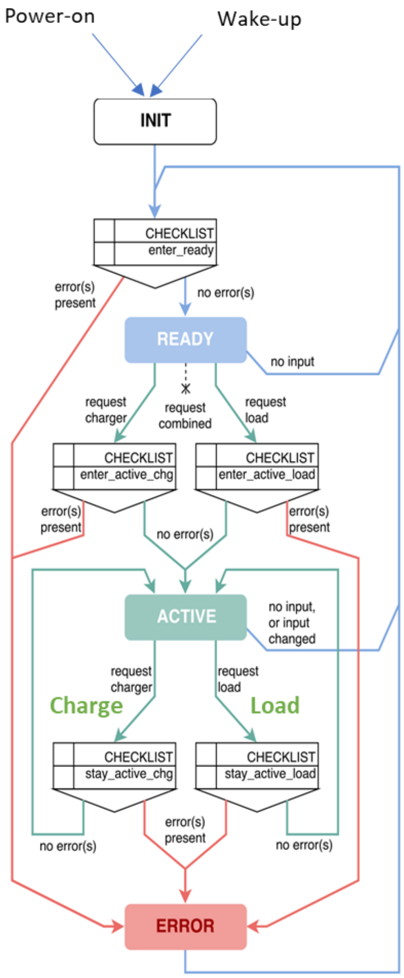


Figure 3.15: BMS state diagram

The BMS is started up by applying power to the BMS. This requires that:

- The main supply voltage (nominal 12V) is fed to the BMS
- The BMS is not in sleep mode

After starting-up, the BMS enters the basic initialisation (INIT) mode. If no critical errors are present, it will immediately move on to the READY mode

3.4.1 Boot/Configuration mode

The BMS boot-mode indicates a mode outside of the state diagram of Figure 3.15. In this mode the system is ready to receive new firmware or new configuration (*.XLM) files. Once these have been successfully installed the system will leave boot mode and go into the so-called application mode, which corresponds to entering the state diagram of Figure 3.15.

NOTE: The outputs to contactors etc. is incative in Boot/Configuration mode <u>opening all the contactors</u>!

3.4.2 Sleep Mode

The BMS supports sleep mode, which is entered by activating an I/O if this function is configured, see section 3.9 Waking up the BMS can be done in two ways:

- Via the ignition-pin please see section 2.3.1/**Error! Reference source not found.**.
- Via a CAN wake-up frame a described in section 4.5.1.

3.4.3 Ready mode

In READY mode, contactors stay open, but otherwise all system functions are active. When either the ‘Request Charger’ or the ‘Request Load’ IO signals are detected, the system will enter the ‘Active Charge’ or the ‘Active Load’ mode Please note that if BOTH the ‘Request Charger’ or the ‘Request Load’ signals are active, the system will enter the ‘Active Charge’ mode.

3.4.4 Active mode

In the ACTIVE mode, contactors will be closed as described in section and current can flow to and from the battery pack. The BMS will leave active mode again:

- If the system is turned off
- If both the Request Charger’ and the ‘Request Load’ signals are in-activated in which case the BMS will go back to Ready mode
- If a critical error appears and the system enters error mode

Please note that in the active mode, no limitations are set to the allowed current flow direction in ‘Active Charge’ nor in ‘Active Load’ mode. This implies that the BMS will accept charge current in ‘Active Load’ mode and the BMS will accept that current is drained from the battery in ‘Active Charge’ mode.

3.4.5 Error Mode

In Error mode, the BMS will set the battery in a passive state by assuring that no current is drawn from or sent to the battery:

- The contactors will open
- The current requested by the BMS will be set to 0

The system will leave the error mode when the critical error which triggered the error mode has disappeared. To avoid system instabilities the system waits a period known as the ‘De-bounce’ time’ before it can re-enter the active modes (close contactors) after a critical error has forced the contactors to open.

3.5 Contactor control

3.5.1 Contactor configuration

The BMS supports a full four contactor layout arrangement to connect and disconnect both the load and the charger galvanically, please see Figure 3.16:

- Load positive
- Load negative
- Pre-charge
- Charge negative

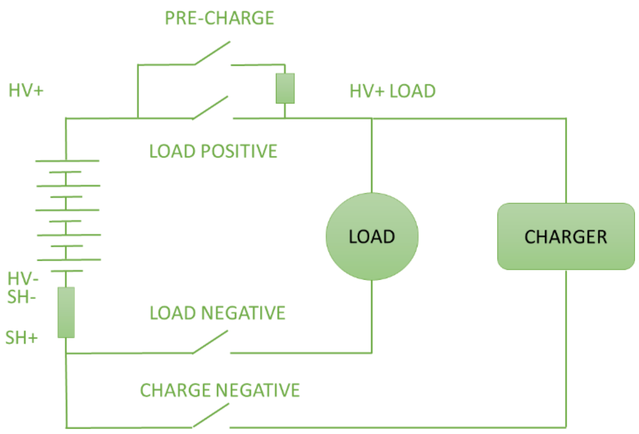


Figure 3.16: Contactor arrangement for the BMS

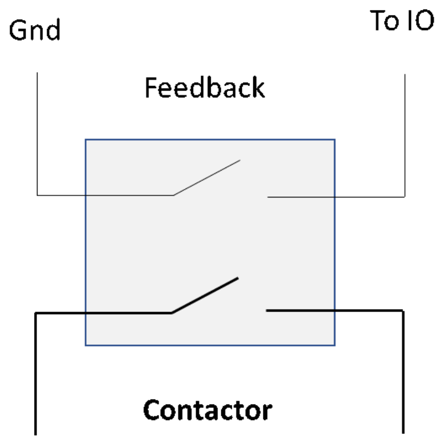


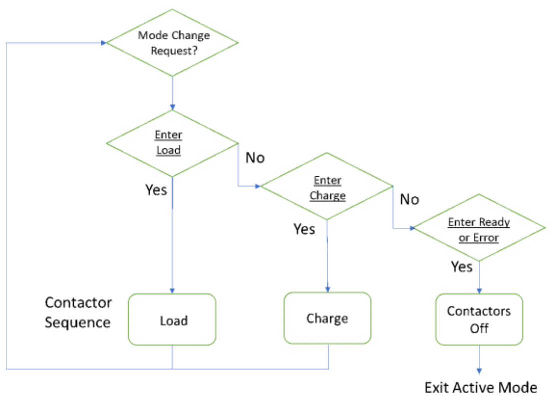
Figure 3.17: Contactor Feedback

Furthermore, each contactor may include a feedback switch/signal, which is a simple low current switch mirroring the position of the main contactor. By connecting one end of the feedback switch to Gnd and the other to a BMS IO, the actual position of the Contactor can be monitored by the BMS.

Any combination[2] of actually used contactors with or without feedback can be configured in the BMS Creator by the GPIO settings as described in section 3.9. If any of the four potential contactors is not used/configured, the standard actions (see Figure 3.19) associated with this unused contactor will simply be skipped.

3.5.2 Contactor activation

In all modes except, the ACTIVE mode, contactors are always off. In the active mode three different contactor control/enable sequences exists and are activated when the BMS in the active mode receives a mode change request, please see Figure 3.18.



1. Entering Active Charge mode. This is entered if the system is in not in ‘Charge mode’ and the “Request Charger” signal becomes/is activate.
2. Entering Active Load mode. This is entered if the system is in not in ‘Load mode’ and the “Request Load” signal becomes/is activated
3. Contactors off. This is entered when the system leaves active mode either to go to error mode or to go to ready mode.

Figure 3.18: Overall contactor control scheme in ACTIVE mode.

[2]Some combinations may be limited by the total number of available GPIOs, depending of how many GPIOs that are used for other purposes.

3.5.3 Activate Load sequence

When the BMS enters the "Activate Load" contactor sequence, the BMS will control the contactors as shown in Figure 3.19.

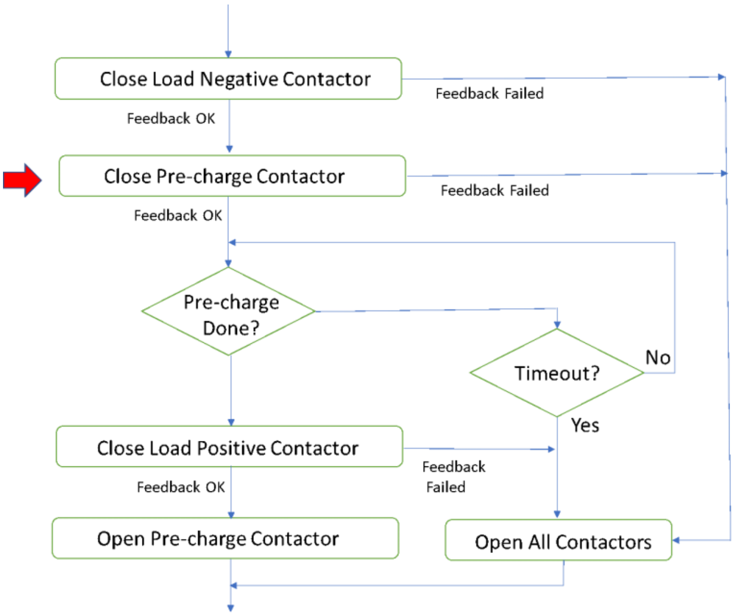


Figure 3.19: Contactor sequence for activating load. The red arrow on the 'Close Load Negative Contactor' indicates that this is the only action which is different between the 'activating load' and 'activating charge' contactor sequences

Please note that for all contactors where feedback is enabled, the BMS will after each change in contactors state:

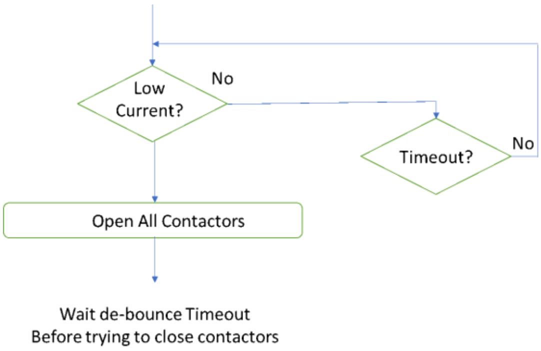
- Test for correct feedback, i.e. if the new contactor state should be OPEN, the BMS will check that the feedback signal matches the OPEN state.
- If the feedback failed the sequence will be aborted. If this 'Feedback' error is configured in the checklist "Errors – Stay load", this will eventually send the system into error mode.

Before entering the contactor activation sequence, the BMS will check if the 'de-bounce time' has expired and if not wait until this is the case. The de-bounce time is a system stabilising time, determining the time needed from opening all contactors, see section 3.5.5 , until the system is again allowed to close the contactors.

3.5.4 Activate Charger sequence

The active charger contactor sequence is identical to the active load sequence except for the first contactor closing where the 'Charge Negative Contactor' is closed rather than the 'Load Negative Contactor'

3.5.5 Contactors Off sequence



When requested to open contactors (Contactors Off), the BMS will, to protect the contactors, not open these until either,

- The current is below the ”Maxthe” Max contactor break current.”
- The time passed since ‘the Contactors Off’ request exceeds the “Contactors off timeout”.

A flow diagram is shown in Figure 3.20 and specific examples sketched in Figure .

Figure 3.20: Contactor Off sequence.

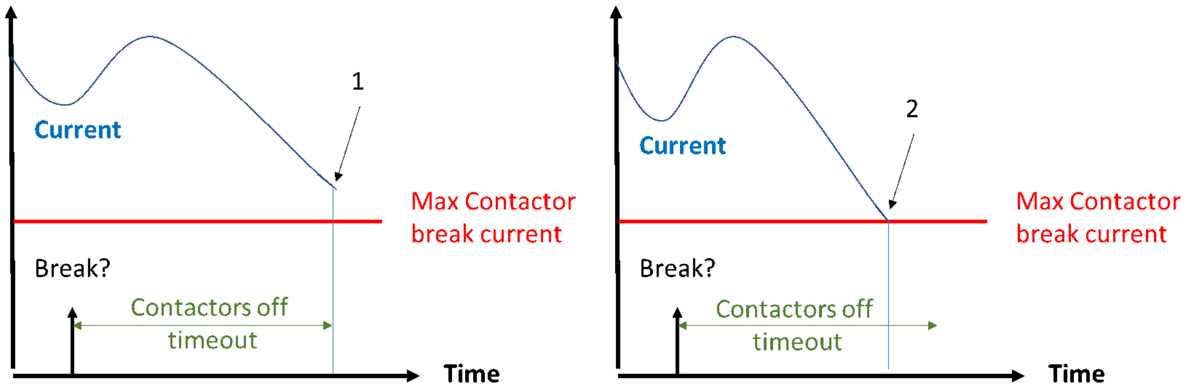


Figure 3.21: In the first example the current does not get below the desired level before the “Contactor off timeout” has expired. Consequently, the contactor opens at point ‘1’ corresponding to the “Contactor off timeout” time. In the second example the current reaches the “Max contactor break current” before the “Contactor off timeout” Consequently, the contactor opens at point ‘2’.

3.5.6 Contactor timing

Configuring the timing of the contactors is done in part2 of the “Operational limits” pane of the BMS creator.
Here, the two parameters highlighted by yellow in below need to be configured.
Please note that all contactors in the system will operate likewise for:

- Max contactor break current.
- Contactors off timeout

Max. contactor break curr.	500.00	mA
Max. precharge end curr.	200.00	mA
Max. contactor retries	2	
Contactors off timeout	2.0	sec

Figure 3.22: Contactor Off sequence.

3.6 Error handling

The BMS monitors internal and external functionality to ensure a safe operation of the battery application. Each internal and external function is able to trigger an error that is stored in the BMS . For each active error there is a timer error, an origin of the error and the error code . Please see section 8.2 for a list of error codes and origins.

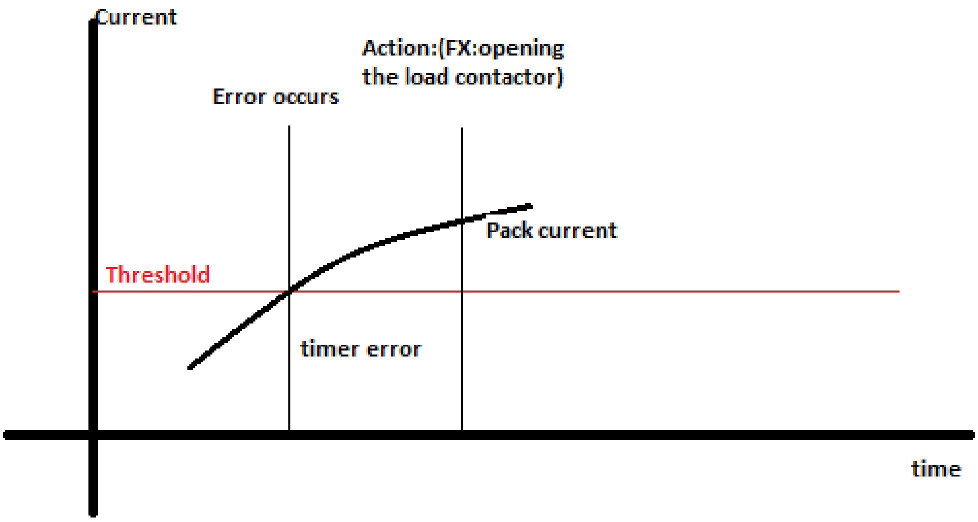


Figure 3.23: Example of error handling.

An example of error handling functionality is sketched in Figure 3.23.
When a threshold setting for a given parameter, X, is crossed, an error associated with this particular parameter is activated in the BMS (e.g. pack current exceeds the limit) with a timer error (e.g. 5 seconds).

3.6.1 Error checklists

An active error in the BMS does not by itself cause an action, such as opening contactors. Errors can be configured to control the BMS state machine (see section 3.3.3), which in turn controls the contactors. In the BMS Creator, this is accomplished by filling out the error check lists as shown in Figure 3.26.

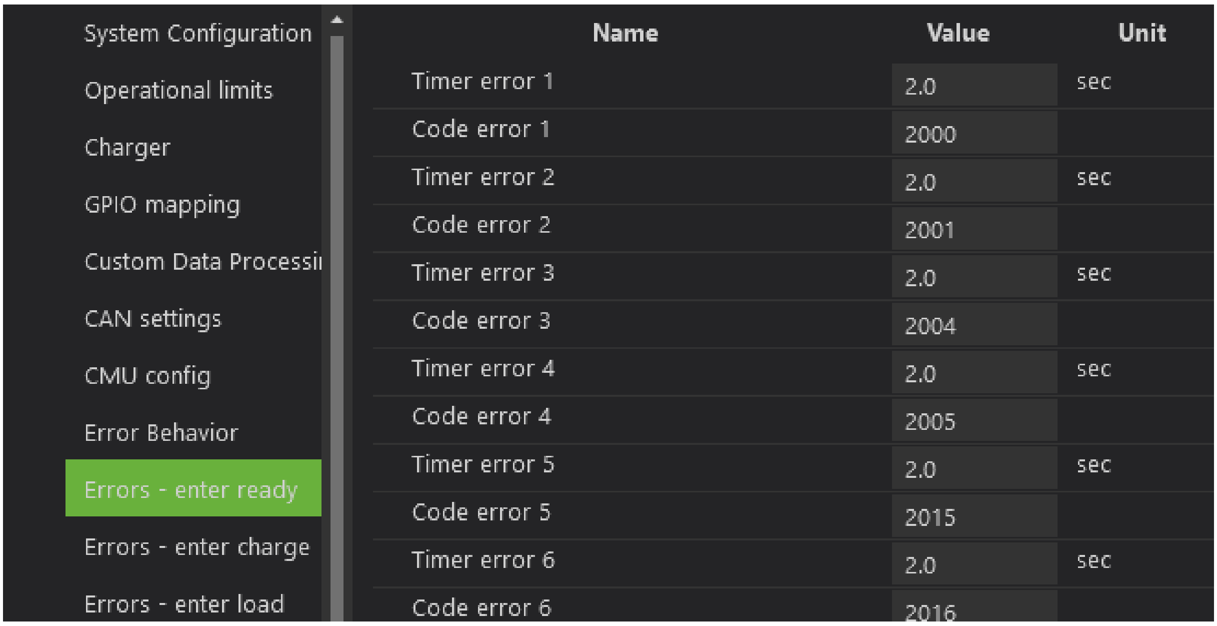


Figure 3.24: Part of the error checklist configuration for entering the Ready state.

These checklists control the BMS state machine state transitions and are thus one of the most important and necessary configuration steps to take, when setting up a system.

To ensure correct system operation it is important to configure the error checklists to match the application. Some settings might vary from application to application and opening of contactors might not be a desirable in all applications and situations.

First step is to consider what battery related errors are to be acted on in terms of relay protection, when the mode is active. In load mode, the ”enter” and ”stay” related checklists should be populated with errors and fault conditions, that if not handled by contactor protection, would damage the battery or cause a dangerous situation. For active load mode, this might be a current overload or a temperature overload. For active charge mode, it might be cell over voltage, temperature overload or sensor errors.

The differentiation between ”enter” and ”stay” is useful to define entering a mode with a perfect system without any problems, and then defining when not to stay in that mode by the use of the ”stay” checklist. An example would be to define that entering load mode is not possible with a bad temperature sensor, but while in the mode, one bad temperature sensor does not exit the mode.

The ”error codes” in the checklists are tightly connected to the ”Timer error” therefore it is important to evaluate at what delay time the error code in the checklist should take effect.

3.6.2 Assigning actions to errors

To prevent entering or staying in a specific state under some specific condition, the relevant error code and timer error should be inserted.

As an example, to prevent entering the Ready state when the “Max Cell Voltage” is exceeded, the error code 2001 has to be inserted in the “Code error” field of the error pane. This is shown for the Code error 2 field in Figure 3.26.

The delay at which the contactors are opened is configured in the associated “Timer error” field.

In Figure 3.26, 2s delay has been chosen for error 2, corresponding to contactor opening 2 seconds after error occurrence.

Please note that error actions are configured individually for the different system states. These states are described further in section 3.3.3. This implies for example, that the user can configure the system to react in one way to a given error when charging and in a different way when discharging.

3.6.3 Default Error Behavior

Default Error behavior is part of configuration in BMS creator to define the default clear time for all of the errors. Figure 3.26.
Default error type field must be 0(Other types are reserved for the future purpose)

Default clear time is the minimum time the error will be reported before it is cleared.

For instance if an over current occurs in a very short time the the error will be reported for the default clear time. In this case BMS will have time enough to log or send out the error by CAN.

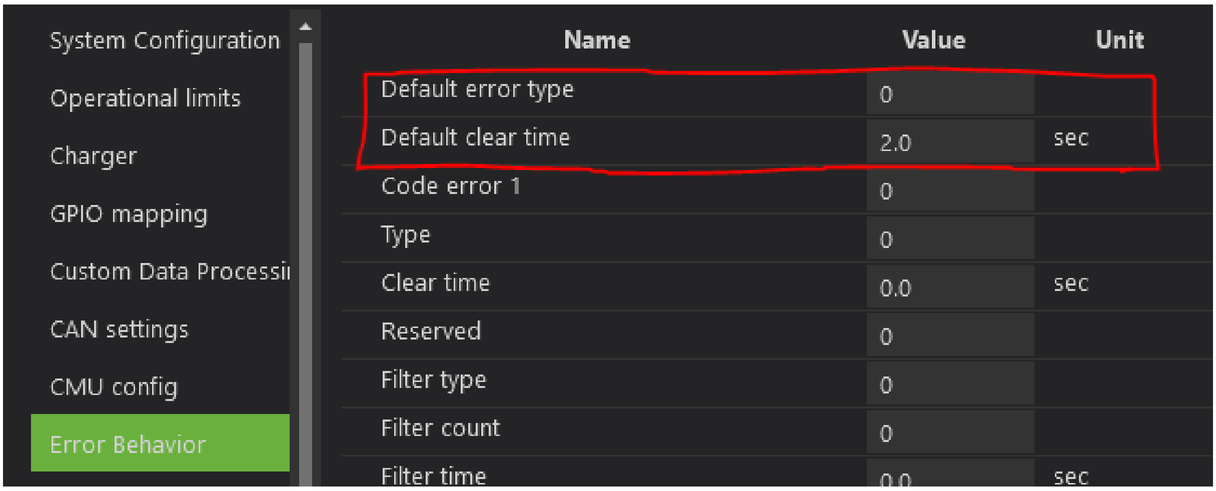


Figure 3.25: Default Error Behavior Configuration

3.6.4 Error Behavior Advanced

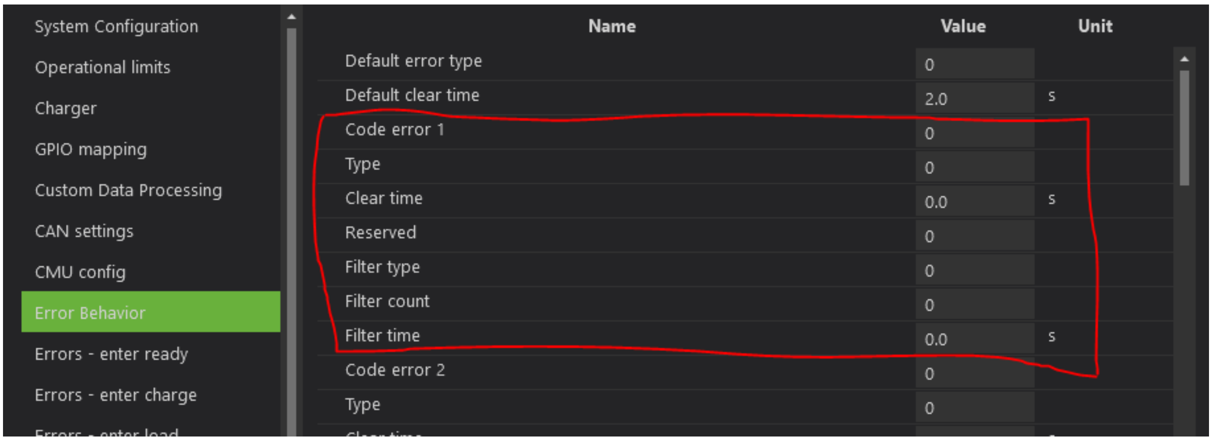


Figure 3.26: Advanced Error Behavior Configuration

- Code error 1

  Contains the error code for an error that required special handling.
- Type

  0 : Normal. The error will be reported for (at least) Clear time

  1: Auto Clear. The error will auto-clear after Clear time

  2: Reserved. Do not use

  3: Reserved. Do not use

  4: One-shot. The error will be reported as soon as it is detected and gets cleared upon resetting the 12V supply to the MCU

  5: One-shot auto clear. The error will be reported once between reboots and will auto clear after Clear time

- Clear Time

  Used according to the Type field.
- Reserved

  Reserved for additional features
- Filter type

  0: None. No filter is applied to the error

  1: Counter. The error is reported after Filter count occurrences. If the error is cleared, the counter is reset.

  2: Timer. The error is reported Filter time seconds after it is raised. If the error is cleared the timer is reset.
- Filter count

  Specifies a count to be used with the filter. See Filter type
- Filter time

  Specifies a timer to be used with a filter. See Filter type

3.7 Charging and Balancing

Charger control is essential to a well-functioning BMS. If charging cannot be stopped when the cells are fully charged, unsafe scenarios may occur. Consequently, the BMS is designed to support redundant ways of stopping charging:

1. The primary method for stopping or modifying charging is via a communication protocol from the BMS, as described in this section.
2. The back-up method for stopping charging is to break the current path between the charger and the battery by opening all contactors.

The BMS is designed to request charge current from an external charger when the user has activated charging using an I/O (see chapter 3.9), if no errors prevent the BMS from operating in that mode. The errors that prevent entering and staying in charge mode is set up by the user using the BMS Creator. When in charge mode, the BMS will request a current over either CAN or PWM, depending on configuration.

3.7.1 CAN Charging

For CAN Charging functionality to function properly it is necessary to configure the correct CAN frame format according to the charger equipment manufacturer specifications. The information is typically available in their data sheet or upon request. The BMS has a number of built-in CAN charger formats that may be used instead of a user-configured CAN frame.

3.7.2 PWM

When PWM functionality is enabled the BMS will output a PWM signal at 1 kHz with a variable duty cycle. The active period of the duty cycle is when it is pulled to ground. A configurable setting is available to invert the signal if needed. It is also possible to configure the minimum and maximum duty cycle for the PWM output. The BMS will then adjust the duty cycle until the measured current into the batteries matches the requested current for charging.

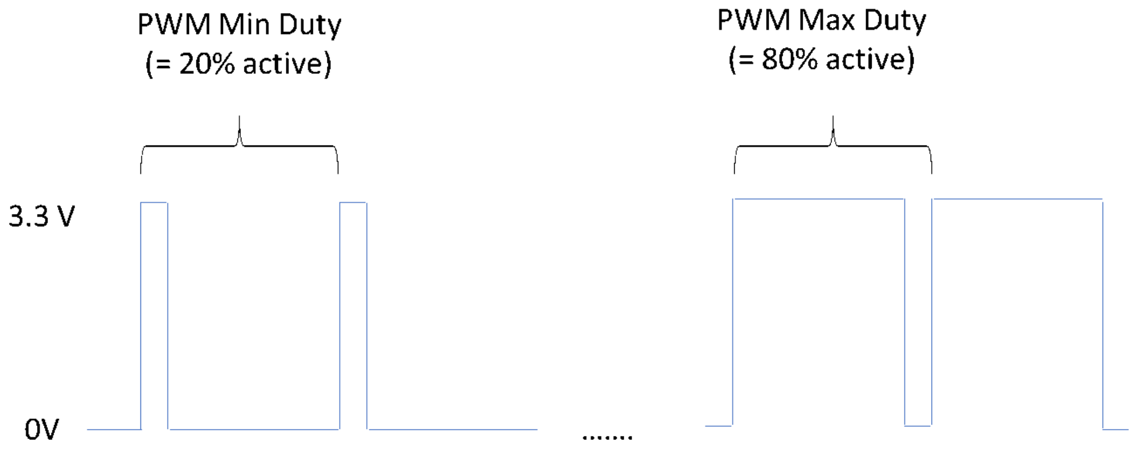


Figure 3.27: Examples of PWM signal with 20% and 80% Max Dutycycle. In this case the PWM operation is non-inverted (PWM Signal Inverted = 0 in the BMS Creator). If inverted operation had been selected, PWM signals would have been 0V when 'active' and 3.3 V when 'passive'.

If it is required to control a charger by an analog voltage it is possible to convert the PWM signal to an analogue control voltage by using for example a low-pass filter.

3.7.3 Requesting charge current and possible errors

The requested charge current is controlled by a software PID controller with configurable coefficients to allow the user to optimise the charger control for the specific battery system.

The PID controller determines the requested charge current (I_{Charge}) as a function of the difference between the desired target cell voltage (V_{target}) and the actual maximum cell voltage (V_{max}), ie.

$$I_{Charge} = PID(V_{target} - V_{Max})$$

The current requested by the PID will always adhere to the dynamic current limitations defined in section 3.2.1.1.
In charge mode, if the actual current does not match the requested current within a configurable deadband, the BMS will report errors. One exception to this is is if charging is complete (current request is thus 0), but the BMS is delivering a larger amount of current than the charging current deadband: in this case, no error is reported. This is to prevent incorrect charging errors if a DC/DC converter or similar load is connected to the BMS during charging.

Please note that special error settings are for the 'request current' function. This is found in the BMS Pane 'Errors – request current'. These errors are set in the same way as the other errors panes, but in case of a critical error here the actions are different from the other errors settings. In case of a 'Request current' error:

- The BMS does not go into error mode, but stays in the present mode and does not change contactors.

- The requested current is set to 0.

Note! The PID controller determines the charge current request while taking dynamic current limit IN (DCLI) levels into account. It means that BMS does not request current higher than what user specified as IN current limit (charging current limit) for various temperature and SOC% condition. User must set DCLI settings properly so BMS will not end up requesting 0 A unintentionally. For example, if user defined DCLI as 0 A for 5 °C and all SOC% values, BMS will not ask any current higher other than 0 A when the temperature input is 5°C.

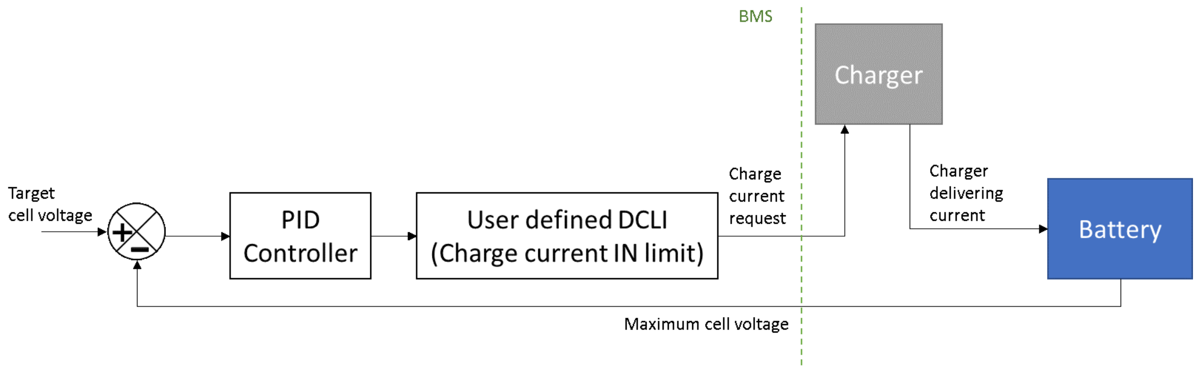


Figure 3.28: Block diagram showing the relation of charge current request with PID controller and dynamic current limit IN (DCLI).

3.7.4 Cell Balancing

Cell balancing is a key BMS function which ensures that when a battery is charged, all cells will be charged to their maximum capacity (100% SoC). This is essential for getting the full capacity out of a battery, as the maximum capacity of a battery is determined by the cell with the least capacity.
The BMS will only perform the balancing

- if an I/O has been configured to activate the balancing functionality (see section 3.9).
- This specific I/O is activated, i.e. connected to Gnd.

When the configured I/O has been activated, the BMS bleeds all cells which have a voltage higher than the balancing limit voltage. The balancing process will be allowed (but not necessarily active) as long as the configured I/O is active. Balancing will occur as long as all cell voltages are not approximately equal, possibly paused by high internal board temperatures, until such temperatures no longer exceed the temperature limit of the board. The balancing limit/target is controlled by the BMS, and the values are given by: **Lowest cell voltage + balancing deadband**, if the current is lower than the charge complete deadband for current OR if the highest cell voltage is close to the target cell voltage - charge complete deadband for cell voltages. Else: **average cell voltage**.
Balancing is initiated when the input pin configured for 'Activate Balancing' is pulled to ground. This can either be done actively by the user or automatically whenever charging starts by hardware connecting the 'Request Charge Active' input pin to the 'Activate Balancing'' pin. Charging and balancing is ended when all cell voltages are within the balancing deadband voltage around the target.
A feature exists that allows forced bleeding of specific cells specified by a bitmask received over the CAN interface. The CAN frame ID is configured in the CAN settings in BMS Creator. The BMS must continuously receive this frame to continue the cell bleeding.

3.7.4.1 PWM Balancing

During balancing, If the board temperature is increasing significantly and reaches a temperature closer to max internal board temperatures (900C), it is recommended to enable the Balancing PWM duty cycle. This PWM duty cycle can be configured in the Creator software to a fixed value and it can also be optionally changed via CAN. The PWM period is defined as 1000ms and if for example duty cycle is configured as 60%, then ON time is 600ms and OFF time 400ms.The Balancing PWM duty cycle directly depends on the highest (if the CMU uses more than 1 PCB temperature sensor) temperature of the CMU board. The BMS also derates the max balancing PWM duty cycle further when the internal board temperature of the CMU approaches the max allowed PCB temperature. The derating implementation is as follows,

- If the Highest_CMU_PCB_temp[C] < Max_allowed_PCB_temp[C] – 5°C; then the PWM = Max balancing PWM duty cycle
- If the Highest_CMU_PCB_temp[C] ≥ Max_allowed_PCB_temp[C] – 5°C; then the PWM = 0.8 * Max balancing PWM duty cycle
- If the Highest_CMU_PCB_temp[C] ≥ Max_allowed_PCB_temp[C] – 4°C; then the PWM = 0.6 * Max balancing PWM duty cycle
- If the Highest_CMU_PCB_temp[C] ≥ Max_allowed_PCB_temp[C] – 3°C; then the PWM = 0.4 * Max balancing PWM duty cycle
- If the Highest_CMU_PCB_temp[C] ≥ Max_allowed_PCB_temp[C] – 2°C; then the PWM = 0.2 * Max balancing PWM duty cycle
- If the Highest_CMU_PCB_temp[C] ≥ Max_allowed_PCB_temp[C] – 1°C; then the PWM = 0

The Max_allowed_PCB_temp = 90°C for the CMU and in general the same duty cycle is applicable to all CMU's. The only case when CMU duty cycles may vary is when the CMUs have different temperatures in the range of Max_allowed_PCB_temp[C] – 5[C] < T < max_allowed_PCB_temp[C] – 1[C].

The procedure for setting-up the PWM balancing is described in this document present in our knowledge base.

3.7.5 Charge complete

The charging process will end when the following three criteria are met at the same time:

- The min. and max. cell voltage must be within the "charge complete deadband for voltage" from the target cell voltage, e.g. +/- 10.0 mV from 3600.0 mV.
- If the "Charge complete deadband V" is greater than 20 mV, then the **Maximum cell voltage** shall be within the 20mV dead band from the "cell voltage target".

  Note:-This condition will be ignored if the "charge complete dead band V" is less than or equal to 20 mV.
- The measured current must be within the" charge complete deadband for current" from 0 A.

Name	Value	Unit
Charge complete deadband I	100	mA
Charge complete deadband V	10.0	mV

Figure 3.29: Charge complete criteria in BMS Creator in Charger Menu.

Once these criteria are met the BMS will stop the charge process and calibrate the remaining capacityto be equal to the expected full capacity. At the same time the flags "fully charged" and "fully charged latched" will be set active. When the two criteria are no longer met the "fully charged" flag will be deactivated, but the "fully charged latched" flag will be active until the BMS has registered at least 0.1Ah being discharged from the battery pack.

3.7.6 Selecting PID parameters

When optimizing the PID parameters for a given system it is important to notice, that setting any of the coefficients to zero will ensure that the term is not used, e.g. if the D coefficient is set to zero the controller will become a PI controller.

Furthermore, PID regulation can be characterized as ‘over damped’, ‘critically damped’ or ‘under damped’ as sketched in Figure 3.30.

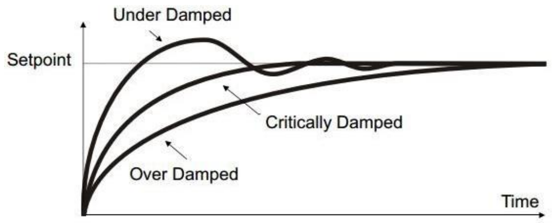


Figure 3.30: PID Damping

- By BMS Creator default settings, the PID controller is overdamped
- To ensure the fastest charge response it is advisable to tune the PID coefficients to achieve a critically damped response.
- Be aware that using underdamped responses will cause the cell voltages to increase beyond the target cell voltage, thus creating a potential unsafe scenario.

Optimising PID parameters is not a simple task. The basic approach recommended for tuning PID parameters, is to make sure that no oscillations occur. To do this a simple procedure can be applied:

- Set: [Kp to the value of ’Max. charge current’ / 100; Ki = 0.001; Kd = 0.0], this should make the PID controller over-damped and will typically result in a reduction in charge current (lower than maximum for most of the charging time, and not only a charge current reduction at higher cell voltages).
- If charge current is still oscillating, reduce Kp further in 0.01 decrements.
- If charge current is not oscillating, set Kp = 0.1. If no oscillations occur increase again by 0.05. It is up to the user to find the correct value, depending on timings in the specific charger setup. A too low Kp will result in the cells never charging completely (unless Ki is used to correct the charge current).
- Once Kp is at the value where oscillation occurs, reduce it slightly and set Ki = 0.02. Increasing Ki further will cause the PID controller to correct the charge current quicker.

Depending on the system setup a few iterations of experiment is required to achieve the optimal PID control parameters. Please note that:

- An error can occur if the oscillation of the current becomes so large that “the actual” current exceeds the requested current by more than 500 mA.
- Moreover, the risk of exceeding the 500 mA limit increases when the charger resolution approaches 500 mA. Consequently, it is desirable to have charger resolutions well below 500 mA, e.g. at 10 mA or less.

3.8 Pre-charge

When the main contactors of a battery pack close and the battery is connected to the load or charger, an in-rush current will flow until the voltage level of the load/charger is equal to the voltage level of the battery. As the resistance in the main battery path can be very low (< <1 Ω), very high in-rush currents may flow in the main battery path.

In most battery systems, the load and chargers have high capacitance at the high-current side facing the battery. These high capacitances (C1 and C2 on Figure 3.31) imply that the high in-rush currents may be present for quite a while until the voltage in C1 or C2 matches the battery pack voltage. High

current levels for extended periods of time can damage the system; for example, the connection pads of the main contactor may weld. Welded contactors cannot switch off, which is an unacceptable safety risk in a battery system.
Consequently, special measures are often needed to avoid damage to contactors when these are switched on. Such measures will typically be the use of:

- Dedicated pre-charge functions built into the charger and load
- A dedicated pre-charge circuit in the battery pack

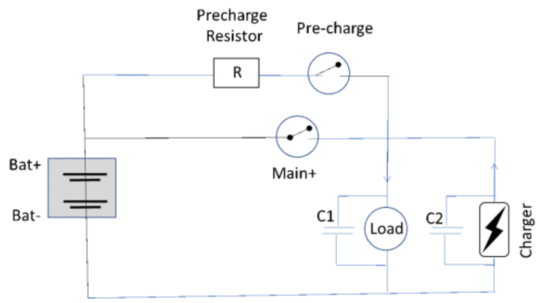


Figure 3.31: Battery system using a pre-charge circuit

A pre-charge circuit in the battery pack, typically consists of a pre-charge contactor and a pre-charge resistor as shown in Figure 3.31. When the system is to be started up, the following sequence is used:

1. The pre-charge contactor is switched on while the Main+ contactor is off.
2. The in-rush current flows through the pre-charge circuit, and the current is limited by the pre-charge resistor.
3. After a while, the voltages have equalised, and the main contactor can be switched on safely.
4. Once the main switch is on, the pre-charge switch can be switched off again.

If the load and charger of a given battery system do not provide special pre-charge measures, it is fair to assume that a pre-charge circuit is needed in the battery pack. Selecting the pre-charge resistor is not a trivial task and is beyond the scope of this manual. If you need assistance in designing a pre-charge circuit for a specific battery system, you are very welcome to contact Lithium Balance.

3.8.1 Pre-charge parameter configuration

Configuring the operating parameters of a pre-charge circuit is done in part2 of the “Operational limits” pane of the BMS creator. Here, the four parameters circled in need to be configured.

Name	Value	Unit
Max. precharge end curr.	200.00	mA
Max. contactor retries	2	
Contactors off timeout	2.0	sec
Precharge timeout	10.0	sec
Contactor retry timeout	2.0	sec

Figure 3.32: Pre-charge parameters to be configured

- *"Max pre-charge end curr."* must be reached before the main contactor engages.
  In addition to the current, a minimum time must expire. This is defined as half the time "Pre-charge timeout".
- *"Pre-charge timeout"* is the time allowed for the current to get below the "Max pre-charge end curr." level. If the current is not below the limit within the configured time is the pre-charge aborted and all contactors opened.
- If the pre-charge sequence has been aborted it will be 're-tried' the number of times specified by *"Max contactor retries"*
- The interval between each retry is specified as *"Contactor retry timeout"*

These parameters are exemplified in Figure 3.35 below

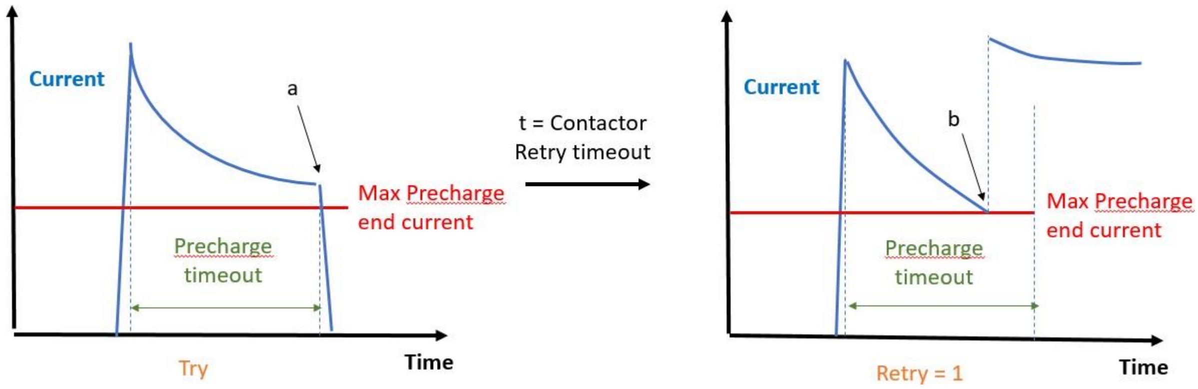


Figure 3.33: In the first pre-charge try, the actual current is above the max end current limit after the pre-charge timeout has expired. Consequently, the pre-charge contactor opens again at point 'a' and the pre-charge is aborted. After the timeout period and new pre-charge attempt (retry 1) starts. Here, the actual current gets below the max end current limit in due time and at point 'b' the Main+ contactor is opened.

3.9 I/O settings for state control and contactor configuration c-BMS

The BMS has a number of I/Os which are essential to the function and state control of the BMS.

- 4 signals (IO No 1-4) are input only.
- 4 signals (IO No 5-8) can be used for either input or output.
  - Please note that a special case exists for I/O 8, which is the only I/O that can support PWM output.

Configuration of the IOs is performed with the BMS Creator “GPIO Mapping” pane (see section 5.5), where the I/Os are assigned to different functionality. In this pane any functionality that has been configured with the I/O as 0 will not be enabled.

3.9.1 I/O control via CAN or CDP

All the 8 GPIOes in c-BMS can be controlled by either CAN or CDP. To control the I/O as output via CAN or CDP do the following changes in your configuration:

1. Select the desire GPIO output under the GPIO mapping in creator configuration
2. Fill the value of the selected GPIO output with a BMS data ID within the range of “CAN_RX_DATA” from 50001 to 50025 or alternatively one of the ID’s “CDP_OUTPUT” (ID’s range 1005 to 1029)
3. In case of using CAN to control the GPIOes , configure a CAN Rx frame to recive data for the above configured GPIO output. The CAN frame must store the CAN data in the same BMS data ID location as configured for the GPIO output

When the c-BMS receive a relevant CAN message, which is configured in GPIO mapping and the message is not zero then the relevant output will be turned on. Cycle time of receiving message must be less than the configured Rx frame Timeout interval. CDP output value can also turn the relevant output on if it is not zero.
Here is a configuration example to control the GPIO 8 as an output via CAN, where the CAN ID is 400 and the received CAN data is stored in the BMS data ID 50001 location.

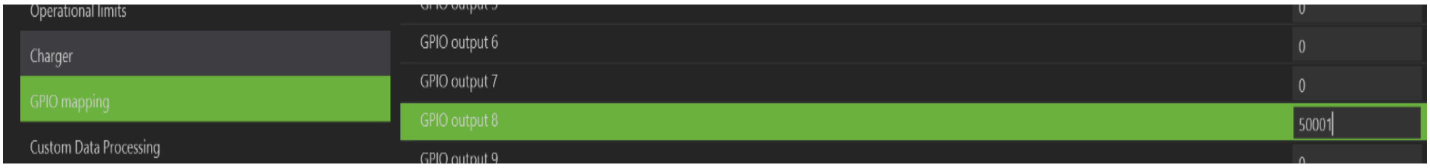


Figure 3.34: Output 8 configuration to be controled by CAN

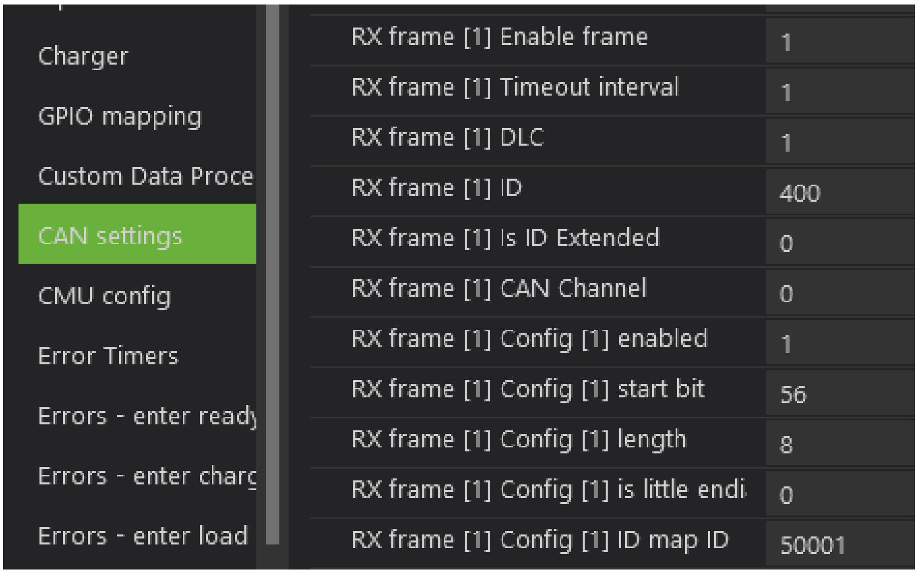


Figure 3.35: CAN configuration to control output 8

3.4 Auto off

The auto-off function is used for turning off BMS MCU power to avoid "idle" BMS power when the battery not is in use. The configurable MCU Power-hold GPIO output will deactivate and turn off power when the auto power off function takes action. External components should be used to support the auto off function. A principle circuit for the auto off and external required components is shown in Figure

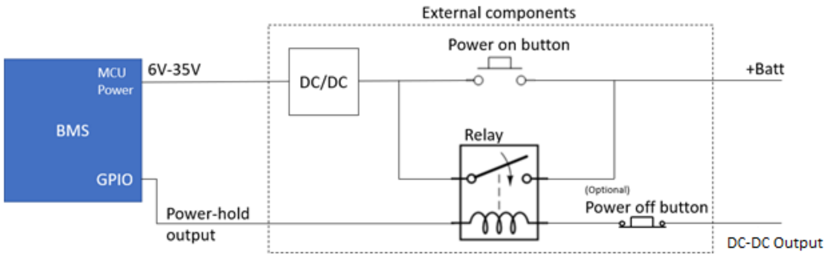


Figure 3.36: Auto off principle circuit

Schematic note: Please remember to add a protective fly-back diode across the relay coil.

The GPIO Power-hold must be connected to a normally open (NO) relay that when closed will supply the BMS MCU with power. If using a DC/DC converter for the MCU power then it is recommend to place the relay switch in front of the DC/DC input, because the DC/DC idle power then will be avoided in power off state. Across the relay switch is a non-latching power on push button used to power up the MCU. When the MCU powers on is the GPIO Power-hold output activated which makes the relay keeping MCU power on until the MCU auto off function decides to turn off and deactivate the relay.

Auto off situations where the Power-hold output is deactivated:

- BMS in READY or LOAD mode:
  - Drain current too low for a given time. Both the drain current level and the time is configurable. The drain current too low function is used for situations where the current drawn from the battery indicates idle. E.g. a vehicle is parked, and it has been forgotten to turn off power. With this auto off function is power automatically turned off avoiding unnecessary battery draining.
  - Cell voltage too low for a given time. Both the cell voltage level and the time is configurable. The minimum cell voltage of the battery pack is compared with the cell voltage level for this function.
    The cell voltage too low function is used to avoid undercharged battery cells when forgetting to turn off the battery. The function can be used to have a reserve level of battery energy for e.g. moving a vehicle to a charger location.

- BMS in CHARGE mode:
  - Charge current too low for a given time. Both the charge current level and the time is configurable.
    The charge current too low function is used to avoid draining of the battery when charge current does not occur, e.g. if the charger has a fault.
- Fully charged battery for a given time. The time is configurable.
  The fully charged function is used to turn of the battery and charging when the battery is fully charged. When not turning of the BMS and battery will the battery after reaching fully charged typically drain until the BMS decide to re-charge the battery. This draining and re-charging can end in an endless loop.

4 CAN communication and set-up

The BMS supports Controller Area Network (CAN) communication in accordance with ISO 11898-2:2003 and ISO 11898-5:2007. The implementation supports

- CAN 2.0 A (11-bit identifiers)
- CAN 2.0 B (29-bit identifiers).

The BMS is able to run at 4 different CAN speeds: 125 kbps, 250 kbps, 500 kbps, 1000 kbps. Detailed information of how to set up CAN frames (e.g. 11 or 29 bit identifiers) are found in section 4.2.

4.1 Configurable CAN data format

The CAN data to and from the BMS is structured in CAN frames each containing an identifier and up to 8 bytes. The bytes in a CAN frame are specified from left to right, meaning the left most byte is denoted byte 0, while the right most byte is denoted byte 7.

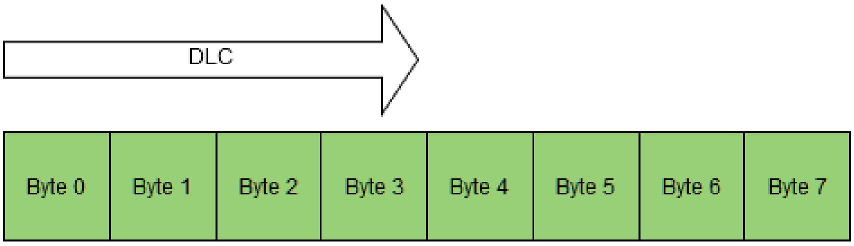


Figure 4.1: CAN frame byte ordering.

The Data Length Code (DLC), denotes the byte wise length of a given frame. The DLC counts from CAN frame byte 0. A DLC of 4, thus means CAN frame byte 0, 1, 2 and 3, as indicated on Figure 4.2 .

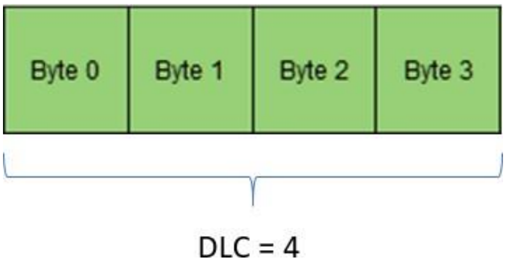


Figure 4.2: CAN data byte order, showing a Data Length Code (DLC) of 4 bytes.

CAN data is structured in one of two formats known as “Big Endian” or “Little Endian”. Which of these formats to use for a given parameter is typically determined by the format requirements of the other CAN-communicating equipment in the system.

4.1.1 Big endian bit (Motorola) format

In big endian bit ordering mode, or Motorola format, the bit ordering starts from the right most part of the CAN frame, with bit 0 being the right most bit, while bit 63 is the left-most bit.

Figure 4.3: Big endian bit order in CAN frame.

A full big endian CAN frame can be seen in Figure 4.4.

Byte 0	63	62	61	60	59	58	57	56
Byte 1	55	54	53	52	51	50	49	48
Byte 2	47	46	45	44	43	42	41	40
Byte 3	39	38	37	36	35	34	33	32
Byte 4	31	30	29	28	27	26	25	24
Byte 5	23	22	21	20	19	18	17	16
Byte 6	15	14	13	12	11	10	9	8
Byte 7	7	6	5	4	3	2	1	0

Figure 4.4: Big endian bit order for all CAN frame bits.

A caveat of using big endian bit ordering mode is that the DLC counts from byte 0 (left most byte), but bits count from byte 7 (right most byte). Thus, for a DLC of 4 (where bytes 0 to 3 are used as shown in Figure 4.2), only bits 32-63 are available.

4.1.2 Little endian (Intel) format

In little endian bit ordering format, or Intel format, the bit ordering is somewhat more complex. The bit ordering within each byte is from right to left, which results in the CAN frame layout seen in Figure 4.5

Figure 4.5: Little endian bit order in CAN frame.

A full little endian CAN frame can be seen in Figure 4.6.

Byte 0	7	6	5	4	3	2	1	0
Byte 1	15	14	13	12	11	10	9	8
Byte 2	23	22	21	20	19	18	17	16
Byte 3	31	30	29	28	27	26	25	24
Byte 4	39	38	37	36	35	34	33	32
Byte 5	47	46	45	44	43	42	41	40
Byte 6	55	54	53	52	51	50	49	48
Byte 7	63	62	61	60	59	58	57	56

Figure 4.6: Little endian bit order for all CAN frame bits.

4.2 CAN transmission (TX)

Each CAN frame to be transmitted by the BMS must be setup individually. Setting up a CAN frame involves

1. Setting up the frame itself
2. Setting up the data to be placed in the frame.

4.2.1 Setting up CAN TX frames

In the BMS Creator up to 20 CAN frames can be configured for transmission. Each available CAN frame has the following overall configurable items:

Item name	Description	Acceptable values
Enable Frame	The CAN TX frame is enabled if set to 1. If set to 0 it is not enabled	0 and 1.
Update Interval	CAN TX frames can be updated at a maximum rate of 100 ms. 1: the CAN frame is updated at the maximum rate 2: the CAN frame is updated at half the maximum rate, i.e every 200 ms 3: The 3 CAN frame is updated at one third the maximum frequency, i.e every 300 ms ...and so on.	1 to 65535.
DLC	The length of the CAN TX frame in bytes.	0 to 8.
ID	The CAN TX frame id as a decimal value.	0 to 536870912 [2^(29) 29 bit ID] 0 to 2048 [2^(11) 11 bit ID]
Is ID Extended	0: the frame_id is an 11-bit CAN frame ID. 1: the frame_id is a 29-bit extended CAN frame ID.	0 and 1.
CAN channel	The CAN channel to output the frame on. 0: S-CAN, the first CAN channel 1: I-CAN, the second CAN channel 2: U-CAN, the third CAN channel	0 to 2, depending on the number of available CAN channels on the BMS hardware.

Table 4.1: Overall CAN TX frame configurable parameters.

4.2.2 Setting up data within a CAN TX frame

Within each CAN frame up to 10 individual data sets can be configured, implying that each frame can contain up to 10 individual parameters. Each 'data' to be contained in the CAN TX frame has the following overall configurable items:

Item name	Description	Acceptable values
Config X enabled	The configuration entry number 'X' (for the CAN TX frame) is enabled if set to 1.	0 and 1.
Config X entry type	The type of the entry data. 0 means constant, 1 means variable BMS data ID.	0 or 1.
Config X start_bit	The bit in the CAN frame, where the first data bit is to be placed	0 to 63.
Config X length	The amount of bits to write data to, starting from start_bit	0 to 32.
Config X is_little_endian	If set to 1, bits are treated as little endian (Intel format). If set to 0, bits are treated as big endian (Motorola format).	0 or 1.
Config X data	Either the constant data or an ID to the BMS variable to place in the CAN frame. ID numbers are found in Appendix: Data ID Mapsection 0	0 to 4294967295 (2^(32) – 1)

Table 4.2: CAN TX frame data configuration parameters.

Note that if data configurations for a CAN frame are set up such that they place data in the same bits of the CAN frame, the result is implementation defined. What this means is that it may work as a user expects (e.g. the top or last data configuration has priority) in a given version, but be different in future versions. ***Do not depend on any given overlap functionality.***

4.3 CAN reception (RX)

Setting up CAN frames for the BMS to receive works similarly as for CAN transmissions in that it requires

1. Setting up the frame itself
2. Setting up the data contents of the frame and destinations IDs

Data in a received frame can be placed in a specific set of ID map IDs, from where it can be used by Custom data processing, CAN transmission or specific functionality which is set up individually in the BMS Creator tool as shown in Figure.

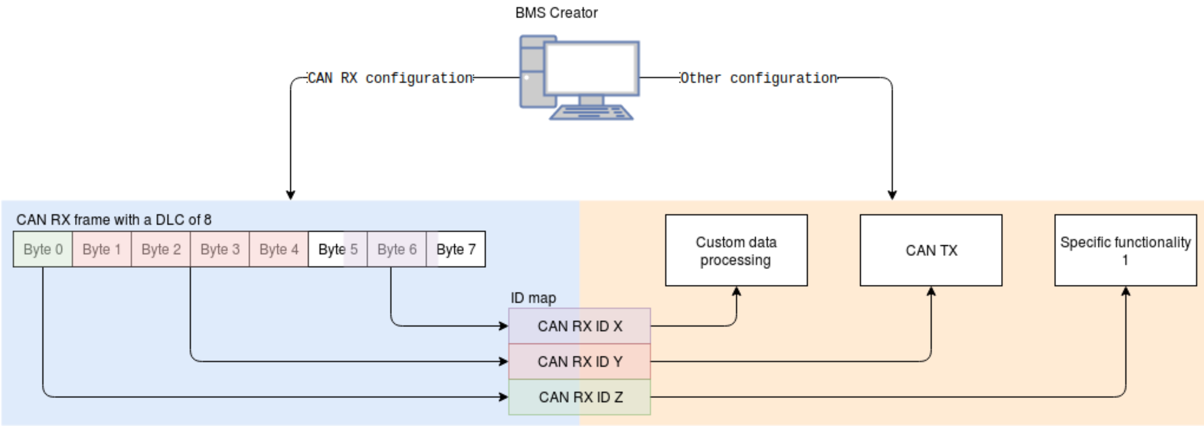


Figure 4.7: CAN RX configuration.

The BMS is designed to handle incoming CAN RX frames at a frequency of up to 100 Hz (every 10 ms). If frames are sent more often than that, some frames may risk being dropped.

4.3.1 Setting up CAN RX frames

In the BMS Creator up to 5 CAN frames can be configured for reception by the BMS. Each available CAN RX frame has the following overall configurable items:

Enable Frame	The CAN RX frame is enabled if set to 1. If set to 0 it i	0 and 1.
Timeout interval	Optional timeout causing a BMS error whenever more than Timeout intervals of 100 ms occur between receptions of the CAN frame. Counting starts as soon as the BMS is fully up and running. 0: disables this feature. 1: Error if CAN frame is not received within ~100[1] ms[1]. 2: Error if CAN frame is not received within ~200 ms. 3: Error if CAN frame is not received within ~300 ms. ...and so on.	0 to 65535.
DLC	The length of the CAN RX frame in bytes.	0 to 8.
ID	The CAN RX frame ID as a decimal value.	0 to 536870912 [2^(29) 29 bit ID] 0 to 2048 [2^(11) 11 bit ID]
Is ID Extended	0: the frame_id is an 11-bit CAN frame ID. 1: the frame_id is a 29-bit extended CAN frame ID.	0 and 1.
CAN channel	The CAN channel to receive the frame on. 0: S-CAN, the first CAN channel 1: I-CAN, the second CAN channel 2: U-CAN, the third CAN channel.	0 to 2, depending on the number of available CAN channels on the BMS hardware.

Table 4.3: Overall CAN RX frame configurable parameters.

4.3.2 Setting up data within a CAN RX frame

Within each CAN frame up to 10 individual data sets can be configured, implying that each frame can contain up to 10 individual parameters. Each 'data' to be contained in the CAN RX frame has the following overall configurable items:

Item name	Description	Acceptable values
Config X enabled	The configuration entry number 'X' (for the CAN RX frame) is enabled if set to 1.	0 and 1.
Config X start_bit	The bit in the CAN frame, where the first data bit is located.	0 to 63.
Config X length	The amount of bits to read data from, starting from start_bit	0 to 32.
Config X is_little_endian	If set to 1, bits are treated as little endian (Intel format). If set to 0, bits are treated as big endian (Motorola format).	0 or 1.
Config X ID Map ID	An ID within a specific range in which to place the data from the CAN frame. ID numbers are found in Appendix: Data ID Map.	50001 to 50026

Table 4.4: CAN RX frame data configuration parameters.

Note that if data configurations for a CAN frame are set up such that they place data in the same bits of the CAN frame, the result is implementation defined. What this means is that it may work as a user expects (e.g. the top or last data configuration has priority) in a given version, but be different in future versions. ***Do not depend on any given overlap functionality.***

4.4 Custom data processing for CAN

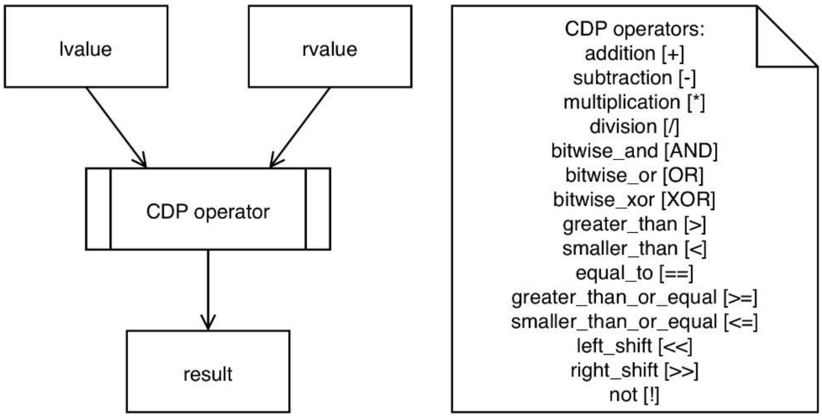
The BMS supports Custom Data Processing (CDP) for up to 25 operations. The CDP is set up using the BMS Creator configuration tool and the processed data is available in the ID MAP with an ID per CDP entry. The CDP functionality requires 3 inputs:

1. The first value, referred to as the lvalue (left value).
2. The second value, referred to as the rvalue (right value).
3. The operator to use between lvalue and rvalue.

The processed output is then calculated as[2]:

CDP_result = lvalue (operator) rvalue

[1]Allowed interval before generating an error may be slightly longer, but will never be shorter.
[2]With the exception of the not [!] operator, this operator applies directly to the lvalue.



The *lvalue* and *rvalue* can either be a constant value (direct data) or data from the ID MAP (addressed data). Typically the *lvalue* will be data from the ID MAP (addressed) and the *rvalue* will be a constant (direct data). The CDP result from processing is available in the ID MAP using IDs ID_CDP_OUTPUT1 to ID_CDP_OUTPUT25. Using the configurable CAN functionality it is then possible to broadcast the value on the CAN bus.

It is possible to have multiple calculations with intermediate results, e.g. if scaling and offset is needed for a value. The intermediate results are then used as input for the next CDP calculation[3].

CDP results are processed successively (one after the other) meaning that CDP entry n will always be processed before *n+1*. Therefore, if using intermediate results, it would be advisable to configure them in the order needed for the calculations.

For examples on how to use custom data processing, see appendix.

4.5 CAN Applications

4.5.1 Wake-up on CAN

The BMS supports remote-wakeup according to ISO 11898-5:2007 which consist of the following pattern:

- a dominant phase of at least 0.5 us, followed by
- a recessive phase of at least 0.5 us, followed by
- a dominant phase of at least 0.5 us.

The complete dominant-recessive-dominant pattern must be received within 0.5 ms to be recognized as a valid wake-up pattern.

As an example, a CAN frame with a Data Length Code (DLC) of 2 bytes with 11-bit ID 0x000 and both data bytes at 0x00, send at 125 kbps to 1000 kbps will wake up the BMS. Here ‘x’ indicates hexadecimal format.

NOTE: Make sure that the I/O used to enter sleep mode is NOT active when trying to wake the BMS. If the I/O is active the BMS will be stuck in a reset loop until the I/O is no longer active.

[3]The intermediate results are stored as regular CDP entries and can be accessed using IDs ID_CDP_OUTPUT1 to ID_CDP_OUTPUT25.

4.5.2 CAN charger support

It is possible to enable a built-in CAN charger frame for certain CAN charger types. If one of the built-in CAN frames is used for controlling the external charger, make sure that the format matches the charging equipment. The built-in CAN charger frames have fixed timings and this is not configurable. If other timings are required a custom CAN frame must be configured.

4.5.2.1 EA-PS8200-70 from Elektro-Automatik

This format includes a special state machine that is only supported on the specified model.

4.5.2.2 Custom big endian format 1

The frame format is as specified in the figure below.

CAN Frame	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Content	Charge enable	100	0	Maximum charge voltage		Requested charge current		0
Description	*1 = charge allowed 0 = charge forbidden*	*Constant*	*Constant*	*0.1 V resolution*		*0.1 A resolution*		*Constant*

4.5.2.3 Custom big endian format 2

The frame format is as specified in the figure below.

CAN Frame	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Content	Charge enable	100	0	Maximum charge voltage		Requested charge current		0
Description	*See byte 0 specification figure*	*Constant*	*Constant*	*0.1 V resolution*		*0.1 A resolution*		*Constant*

Byte 0 uses multiple bits for different purposes as shown in the figure below.

Byte 0	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Content	0	0	0	0	0	0	Clear error flag	Charge enable
Description	*Constant*	*Constant*	*Constant*	*Constant*	*Constant*	*Constant*	*Flag that starts out as 0, becomes 1 after 12 CAN frame transmissions, becomes 0 again after 12 more CAN frame transmissions (24 in total), then remains 0 until BMS is rebooted.*	*1 = charge allowed 0 = charge forbidden*

4.5.2.4 Custom big endian format 3

The frame format is as specified in the figures below. The output in the frame switches between two formats continuously throughout frame broadcasting. The first format outputs the requested charge voltage and the second the requested charge current.

Format 1:

Can Frame	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Content	0x23	0x01	0x24	0x1	Requested charge voltage			
Description	Constant	Constant	Constant	Constant	0.001 V resolution			

Format 2:

Can Frame	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Content	0x23	0x01	0x24	0x2	Requested charge current			
Description	Constant	Constant	Constant	Constant	0.001 A resolution			

4.5.3 CAN-based pack current sensor

The BMS supports using an external pack current sensor that supplies its measurements via CAN, instead of using a directly connected shunt or HALL effect sensor.

The input from the sensor is received via the CAN RX functionality, which stores the relevant data in the ID map. If the BMS is configured to use a CAN-based pack current sensor as its current source, it will then use the data from this ID as the current. The format of the current measurement[4]must be in the same format as the internal current measurements . This can be achieved by the Custom Data Processing functionality.

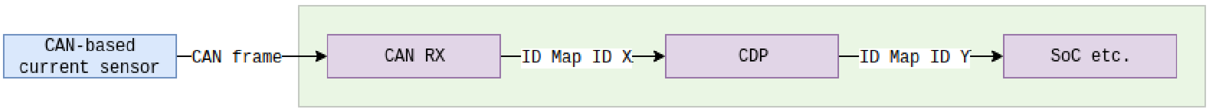


Figure 4.8: Example data flow from CAN-based current sensor to BMS algorithms.

An example flow of data from current sensor to BMS application is shown in Figure 4.8.

4.5.4 CAN-based mode requests

The traditional interface to request load and charge mode is via two digital inputs. The mode request from CAN bus feature implements a secondary interface to issue a load or charge request.

4.5.4.1 Configuring the feature

The feature has two configuration parameters:

- CAN request load ID Map ID
- CAN request charge ID Map ID

[4]32 bit big endian format, resolution of 0.01 mA.

Inputting an ID from the ID map list, will cause the BMS to test this data set for a request. If the functionality is not to be used, the configuration parameter must be zero. This will bypass the request test in the firmware.

If the configuration is not zero, the feature is enabled, and the functionality will test the data location specified for a request.

The most common approach would be to define the CAN request ID MAP ID for a CAN Rx ID Map ID. This indicates to the firmware that it should test the received CAN data for a request.

4.5.4.2 Activating a request

An active request in a data set is defined as a value different from zero. If the data location contains the value zero, this indicates that there is no request in the data location. A value other than zero indicates a request and the functionality will pass on this request to the main state machine. See Table 4.5 for a truth table example.

CAN request ID Map ID (load and charge)	Data in ID 50001 (ID_CAN_RX_DATA1)	Request?
0	0	false
0	1… max value	false
50001 (ID_CAN_RX_DATA1)	0	false
50001 (ID_CAN_RX_DATA1)	1… max value	true

Table 4.5: Truth table for CAN-based mode request

For an example of how to use this functionality, see section 8.8.

4.5.5 Staging

An input which allows user-controlled transition from pre-charge complete to the final stage of the contactor state machine is available. This feature is called a staging request. To support the use of this control input, an output which indicates that pre-charging is complete, and staging is ready, is available (waiting for staging request, ID Map ID 1076).

4.5.5.1 Configuring the feature

The feature has one configuration parameter:

- CAN request staging ID Map ID

Inputting an ID from the ID map list, will cause the BMS to test this data set for a request. If the functionality is not to be used, the configuration parameter must be zero. This will bypass the request test in the firmware. If the configuration is not zero, the feature is enabled, and the functionality will test the data location specified for a request.

The most common approach would be to define the CAN request ID MAP ID for a CAN Rx ID Map ID. This indicates to the firmware that it should test the received CAN data for a request.

4.5.5.2 Activating a request

An active request in a data set is defined as a value different from zero. If the data location contains the value zero, this indicates that there is no request in the data location. A value other than zero indicates a request and the functionality will pass on this request to the main state machine. See Table 4.6 for a truth table example.

CAN request ID Map ID (Staging)	Data in ID 50001 (ID_CAN_RX_DATA1)	Request?
0	0	false
0	1… max value	false
50001 (ID_CAN_RX_DATA1)	0	false
50001 (ID_CAN_RX_DATA1)	1… max value	true

Table 4.6: Truth table for staging request

4.5.5.3 Knowing when the BMS is waiting for a staging request

The BMS ID Map list contains an entry (waiting for staging request, ID Map ID 1076) indicating that pre-charge is complete, and that the contactor state machine is actively waiting for a staging request. It is initially false but whenever a load or charge contactor sequence has reached the pre-charge state, and pre-charge is complete, the value is changed to true. The value remains true until staging is requested, or the pre-charge timer has timed out. Note that if the BMS is indicating that it is waiting for the staging request, and the condition for pre-charge complete is no longer valid, the waiting for staging request flag will be cleared again, until such a time where the pre-charge is again considered complete.

5 BMS Creator

To configure and service the BMS, the Lithium Balance PC tool, BMS CREATOR (TM) is provided.
Note that the BMS Creator only supports CAN adapters from PEAK SYSTEMS available from www.peak-system.com.
Recommended models are

- PCAN-USB IPEH-002022
- PCAN-USB IPEH-002021

5.1 Requirements

- Windows 10, Windows 8, Windows 7 or Windows Vista
- .NET 4.5 framework
- PEAK Systems PCAN USB driver
- Full read/write rights for the installation directory of the BMS Creator to use the 'Use default save location' when generating a configuration, and the 'Use latest generated' feature when uploading BMS configurations.
- 120 Ω termination at the 'far-end' of the CAN bus, please see hardware setup section for details.

5.2 Installation of the BMS Creator Software

1. Run the installation file supplied by Lithium Balance.
2. The installation Wizard will guide the user through the installation process.
3. At the end of the installing process, await confirmation that the installation was completed (this normally takes several minutes).
4. Run the installed software, e.g. by clicking on the newly installed icon on your desktop.
5. A Site Key needs to be entered to active the software. Copy the Site Code and send it to activation@lithiumbalance.
6. A Site Key will be generated for the software by Lithium Balance.
7. The user will receive an e-mail including a Site Key specifically generated for the user PC.
8. When the user receives the Site Key, it should be entered into the site key field, in order to validate the software license.
9. The software is now unlocked and ready to use.

5.3 Connecting the PC to the BMS

The CAN Adapter dropdown box will display available CAN adapters to use for communication with the BMS. If the list is empty, the device(s) is in use by another program or not available. After making the CAN adapter available, click the refresh button next to the CAN Hardware dropdown box to refresh the CAN Adapter list.

As default, the Auto baud rate detection feature is enabled. This feature will automatically detect the BMS baud rate and use this when communicating with the BMS. If the baud rates are known, this feature can be switched off for speed optimization.

The BMS UDS services is only available on the s-CAN bus and hence the CAN adapter should be connected to this bus only.

When CAN speeds and CAN Channel have been configured correctly, click the Connect button to initiate a connection sequence. If no BMS is available the BMS Creator will indicate this with a pop-up box indicating that connection failed.

When connected, the BMS firmware version text field (FW) will contain the firmware version of the connected BMS. The firmware version displayed will be that of the connected state. In Boot load state the version of the Boot loader firmware will be displayed and in Application state the version of the Application firmware.

The BMS Mode status textbox will display the state of the BMS at the point of connection. Note that the BMS state only updates when the connect button, 'Upload BMS configuration', 'Reset BMS', 'Boot load BMS' or 'Read error log' button is pressed.

The BMS connection indication will display a small white check on green background when a connection to the BMS has been established. Like Mode status, the indication is only updated when a button is pressed. If there is no connection, a X on a red background will be displayed. When actively reading data from the BMS, a green process ring is visible around the connection indication.

5.4 Boot loading the BMS

The service/UPGRADE page contains the interface to the BMS Boot load feature. Click the Boot load button to activate a boot load sequence to update the application firmware. The BMS creator will automatically switch the BMS into boot load mode and back after finished upload process.

The BMS Creator will prompt for a file to upload to the BMS if the 'Use built-in' checkbox is not checked. The file format is *.bin and is supplied by Lithium Balance. In a normal Upgrade situation, the built-in should be used by checking the 'Use built-in' checkbox. A progress bar will show the upload progress below the buttons.

When the boot loading sequence is done there will be a small white check on green background indication next to the progress bar. If not, the boot loading sequence has failed and a pop up box with the error code will be displayed. Please note, that the tool will indicate success even though the connected Mode is Boot, and the expectation would be App. This could be because there is no correct and matching configuration on the BMS. In this case continue to the next step of writing configurations.

5.5 Configuration of the BMS

The configuration/MCU page contains the interface to create a configuration for the BMS. The configuration file (*.bin) is always created from a *.XML file and there fore the first step is to open a XML file. If no file exists, one can be created by pressing the 'New' button. If one already exists, press the 'Open' button, and a file dialog will allow selection of already existing file. When done changing the XML file, the 'Save' button can be used to open a dialog of where to save the file.

Change all relevant configuration parameters to match the application the BMS is to be installed. All configuration options are displayed scaled and with a unit next to the value textbox. When all necessary changes are done, press the 'Generate BMS configuration' button. This will generate a *.bin BMS

The 4 digital inputs assigned to:

- Activating load state (IO1)
- Activating charge state (IO2)
- Activating balancing modes (IO3)
- Feedback from a contactor (IO4)

The digital input/outputs are used for:

- Load positive contactor control (IO5)
- Load negative contactor control (IO6)
- Charge negative contactor control (IO7)
- PWM output (IO cBMS/nBMS 8/4) for charger control

configuration file. This file can be used to send to a BMS service user, who should not change anything in the configuration, but only upload it onto a system. When using the tool to change configurations and uploading them in one workflow, it is very useful to have the 'Use default save location' checked. In this way the user will not be prompted for a file location every time a small configuration change is made. After generation of the configuration file, a small indication is given on the right of the 'Generate BMS configuration' button.

Please refer to the section on configuration parameters for an overview and short explanation of the different parameters to be configured.

Please note that this only provides checks of input data which cannot be read by the BMS creator tool. At the moment there is no check on input data which might be meaningless from a BMS point of view, such as configuring a given function for the non-existing IO number 17 or configuring to functions to the same IO.

Please, always evaluate the Errors tabs for correct BMS action in case of any application critical error conditions.

5.5.1 Config file converter

A config file converter exists in the configuration view. This can be used to convert a configuration file from an earlier release into a compatible file for the current release.

After pressing the Convert Config File button, a dialog will appear allowing to select a previous configuration. After that user will be prompted for a place to save the converted configuration.

Note: The configuration converter creates a technically valid configuration, but cannot know the desired configuration values for new configuration fields . The configuration should still be checked manually to ensure that all settings are satisfactory.

5.5.2 IO Configuration

Configuration of the IOs is performed with the BMS Creator "GPIO Mapping" configuration section, where the I/O's are assigned to different functionality. In this section any functionality that has been configured with the I/O as 0 will not be enabled. One example of configuration could be:
In this example, there is only feedback from the load-positive contactor. Consequently, all other feedback mappings are set to 0.

5.5.3 Upload configuration to BMS

The last step of the BMS configuration process is to navigate to the service/UPGRADE page and press the 'Upload BMS configuration'. To use the last generated configuration, be sure to keep the 'Use latest generated' checkbox checked. This will then not prompt for a file location, but directly use the default location which is in the tool install folder. Note that this will only be valuable if the 'Use default save location' was checked during the last configuration generation.

The BMS Creator will automatically switch the BMS from Application mode to Configuration mode, upload the configurations, and switch the mode back to Application.

If the upload succeeded but the BMS is still in Boot mode, it could be because the configuration does not match the firmware, or the configuration values are not valid. In this case the BMS will stay in configuration (Boot) mode.

While the tool is uploading the configurations, a small process ring next the progress bar will indicate activity. Most of the elapsed time associated with this process ring being active but no activity on the progress bar, is while the tool is switching the BMS between App mode and Boot mode or waiting for the hardware to restart.

5.6 Error Log Readout

The service/ERROR LOG page contains functionality to read out a history of logged error in the system. The feature is only available in Application mode and is activated by pressing the 'Read Error Log' button. The tool will read out the log structure and display them in the page. If the user requires the log structure to be saved, the 'Save Error Log' button should be pressed. This will prompt the user for a file location for the CSV file to be saved for later analysis. A successful readout/save of the error log is indicated by a check. During operation a process ring will be visible.

5.7 Live view

The live view pages contain functionality to monitor BMS data. The DASHBOARD page gives an overall indication of system status and displays the system current in a graph view. The MCU DATA page contains information regarding the master controller like IO states, alarms, internal measurements and control data. The CMU DATA page contains data regarding the cell measurement front-end like cell voltages and temperatures. The LOGGING page allows the user to start and stop a log of all live view data to a csv file. The update rate is fixed to 1 [s] interval. When pressing the 'start log' button, the user is prompted for a file save location and name.

5.8 Troubleshooting

- If a communication error occurs, it is likely because the Peak CAN adaptor was not found by the BMS creator tool or is in use by another application. If the "CAN Adapter" field is empty, this is usually the reason. Close any programs that is connected to the CAN adapter or disconnect from the CAN adapter and click the refresh button to refresh the CAN adapter list.
- The LED on the BMS board will flash while in boot loading mode. This information can be useful to help debug communication problems.

5.8.1 c-BMS specific

- Neither firmware boot loading or writing configurations will work unless cells are connected. The cell voltage is required to change state on the BMS board.
- SoC information may be lost if cells are disconnected and BMS board is power cycled.

5.9 CAN error frames

In the "CAN Settings" section one the configuration page of the BMS Creator, it is possible to enable a setting ("CAN ID start error frames").

- A setting of 0 will disable CAN error frames
- Any other setting will start CAN error frames output from the specified CAN frame onwards.

The output of error frames from the firmware onto CAN will start at the "CAN ID start error frames". Note that the specified CAN frame is in decimal, not hexadecimal notation.

It is possible to denote the CAN frame ID as extended and to specify which CAN channel to output to.

5.9.1 CAN error output

If enabled, the CAN output will consist of the contents displayed in Table 5.1 and explained in tables Table 5.2 and Table 5.3. The data is in big endian data format, please see the section discussing configurable CAN for more information.

Frame	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0
X	#1	#2	#3	<Unused>	# 4		<Unused>	<Unused>
X +1	# 5				<Unused>			
X + 2	#7		# 8		#9	#10	#11	
X + 3	#7		# 8		#9	#10	#11	
…	…							
X + 49	#7		# 8		#9	#10	#11	

Table 5.1: CAN frame output

CAN frame	Description
X (the value chosen in the BMS creator field "CAN ID start error frames")	Overview of the number of active errors
X + 1	Error statistics (sum of errors since boot)
X + 2 up to maximum X + 49	Information about errors. Only active errors are transmitted E.g. if two errors are present, the X, X+1, X+2 and X+3 CAN frames will be transmitted.

Table 5.2: Description of error CAN frames

ID	Description
1	Internal Lithium Balance A/S variable used for verification.
2	Internal Lithium Balance A/S variable used for verification.
3	Internal Lithium Balance A/S variable used for verification.
4	Total number of all active errors (across all the severities).
5	Total number of errors (active and otherwise) since boot.
7	Internal Lithium Balance A/S variable used for verification.
8	Internal Lithium Balance A/S variable used for verification.
9	Internal Lithium Balance A/S variable used for verification
10	Internal Lithium Balance A/S variable used for verification.
11	The error code, please see section 8.2.3.

Table 5.3: Description of CAN error frame content

5.9.2 CAN Configuration set-up example

To exemplify how to set-up the CAN configuration of the BMS, the following example shows the procedure for how to add cell voltage 1 to 4 measured on the battery pack to the CAN frames transmitted by the BMS. For information, this example matches the Busmaster set-up example of section 8.11

5.9.2.1 Step 1: Connect BMS creator

5.9.2.2 Step 2: Set up the CAN frame

- Enable frame 1
- Set update interval to 100 ms
- 8 bytes in the frame
- Frame ID set to 200, could be any other unused number
- 11 bit frame

5.9.2.3 Step3 Configure CAN data for cell voltage 1

- Enable 1st parameter in a given frame
- Entry type 1 for ‘BMS variable’ data
- Start bit 48 (equals end bit in byte 1 in big endian format, see section 4.1.1)
- 16 bits (UINT16) for the given parameter, cell voltage 1, (see ID 540 in Appendix: Data ID Map for a list of parameter formats.
- 0 for big endian
- Data ID = 540 for Cell Voltage1 (CMU0 and V0), see Appendix: Data ID Map for a list of Data IDs

5.9.2.4 Step4 Configure data for cell voltage 2 to 4

Repeat step 3 for cell voltage 2 to 4 (Data ID 541 to 543 as listed in Appendix: Data ID Map) moving two bytes = 16 bits ‘down’ for each new entry

Byte 0	63	62	61	60	59	58	57	56	ID 540 Cell Voltage 1
Byte 1	55	54	53	52	51	50	49	48	
Byte 2	47	46	45	44	43	42	41	40	ID 541 Cell Voltage 2
Byte 3	39	38	37	36	35	34	33	32	
Byte 4	31	30	29	28	27	26	25	24	ID 542 Cell Voltage 3
Byte 5	23	22	21	20	19	18	17	16	
Byte 6	15	14	13	12	11	10	9	8	ID 540 Cell Voltage 4
Byte 7	7	6	5	4	3	2	1	0	

Figure 5.1: Allocation of start bits for 2-byte data when using the big endian bit order.

Configurable CAN frame 1

Enable frame	1	Config 3 entry type	1
Update interval	1	Config 3 start bit	16
DLC	8	Config 3 length	16
ID	200	Config 3 is little endian	0
Is ID Extended	0	Config 3 data	542
Config 1 enabled	1	Config 4 enabled	1
Config 1 entry type	1	Config 4 entry type	1
Config 1 start bit	48	Config 4 start bit	0
Config 1 length	16	Config 4 length	16
Config 1 is little endian	0	Config 4 is little endian	0
Config 1 data	540	Config 4 data	543
Config 2 enabled	1	Config 5 enabled	0
Config 2 entry type	1	Config 5 entry type	0
Config 2 start bit	32	Config 5 start bit	0
Config 2 length	16	Config 5 length	0
Config 2 is little endian	0	Config 5 is little endian	0
Config 2 data	541	Config 5 data	0
Config 3 enabled	1	Config 6 enabled	0

Figure 5.2: CAN setting for cell voltage 2 to 4.

5.9.2.5 Step 5: Upload configuration

Once the new CAN set-up has been completed, it should be uploaded to the BMS. Press 'Upload BMS configuration' on the service

1. The program will jump to 'boot mode', upload the configurations to the BMS in Configuration mode and progress bar will indicate upload progress during the operation.
2. When writing is completed, the system will jump back to application mode automatically.

6 Battery system and auxiliary components

To provide the desired functions and ensure safe battery system operation, the BMS must be well integrated with the other components in the battery system. An example of a typical battery system is shown in Figure 6.1 below.

1. Cell voltages and temperatures are monitored directly by the BMS together with the battery current. Please note that if a shunt is used for current monitoring, the shunt must be placed at the negative side of the battery as shown in Figure 6.1.
2. The BMS communicates via the CAN bus with the charger to reduce or stop charging when the desired cell voltages are approached.
3. The BMS communicates via the CAN bus with the load (typically via a Vehicle Control Unit – VCU) to reduce the discharge current if the discharge current exceeds pre-defined limits.
4. Switches are included in the main battery power circuit, between the battery terminals and the load/charger. For safety reasons, at least one switch is MANDATORY to assure that overcharging or discharging can be stopped even if BMS communication (item 2 and 3 in this list) fails.
5. A dedicated soft start or pre-charge switch and pre-charge resistor can be used to reduce potentially damaging current transients during start-up.
6. A fuse in the main battery power circuit is MANDATORY to cut excessive (typically discharge) currents caused by e.g. an external short-circuit or a load requiring too much current for too long.
7. A PC needs to be connected to the BMS during setup for writing configurations. A dedicated CAN adapter (see section c-BMS 2.3.5 is necessary between the BMS and the PC to utilise the Lithium Balance configuration tool.

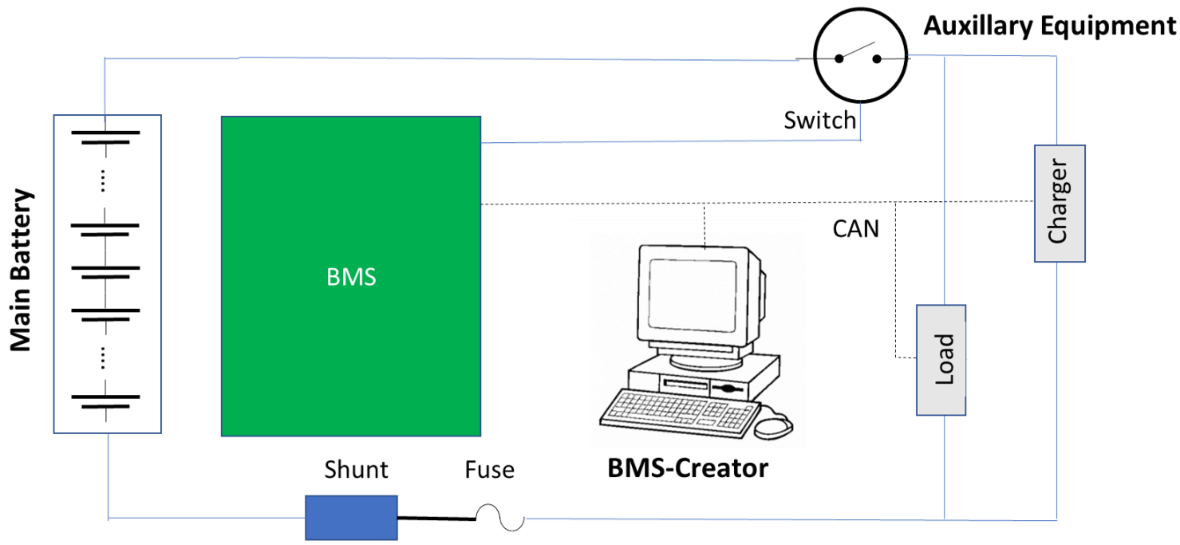


Figure 6.1: Example of a basic battery configuration and the key auxiliary components used

Choosing the right auxiliary components for a battery system can be a complicated task which is beyond the scope of this manual. If you need assistance, please contact Lithium Balance; we may be able to help you directly or can refer you to experts in different areas.

The following section describes the necessary considerations to ensure well-functioning interfaces between the BMS and auxiliary components.

6.1 Battery

Lithium ion batteries are available with different chemistries such as LCO, LMO, NMC, LFP, NCA and LTO. All of these can be managed by the Lithium Balance BMS system. Different chemistries have different operating cell voltages, and these differences have to be taken into account when configuring the BMS. Furthermore, different cell packages and chemistries will vary significantly with regard to maximum sustainable currents and possibly also safe operating temperature range.

It is essential that the selected safety thresholds for cell voltage, temperature and current are based on the requirements of the specific application and on the specific cell information provided by the cell manufacturer, e.g. in datasheets.

In many cases, the battery supplier may specify the maximum value of the battery charge/discharge current in terms of a ‘C-rate’ rather than an ampere value. The ‘C’ (Coulomb) rate reflects “the ‘inverse’ minimum time in hours for a full charge or discharge”. If for example a 200 Ah battery is specified for up to 3C discharge and 2C charge operation, then the

- minimum charge time is 1/2[C] hour = 30 min
- minimum discharge time is 1/3[C] hour = 20 min
- maximum charge current is 200 Ah/30 min = 200 A*2[C] = 400 A
- maximum discharge current is 200 Ah/20 min = 200 A*3[C] = 600 A

6.2 Load

In general, the BMS has no direct control over the load. The BMS needs to be able to communicate via CAN with the central system processor, e.g. the Vehicle Control Unit (VCU). The BMS can then ask the central processor to adjust the load, e.g. to reduce the current if it exceeds the maximum level.

The maximum discharge current (Over Current OUT) threshold can be configured in the BMS Creator. The user can then select how the switches in the system should react if this level is exceeded. Please refer to the BMS Creator manual for threshold setting and switch configuration. The BMS Creator manual also includes detailed information on setting up CAN communication with a VCU.

6.3 Sensors

While cell and battery voltages can be measured directly, sensors are needed to measure current and temperature.

6.3.1 Shunts for current measurements

The battery current (I) can be measured by the voltage drop (over a shunt resistor with a known resistance (R), i.e. ΔV = R * I.

Please note that in the BMS, the shunt detection circuits are internally referenced to the most negative battery voltage (HV-). Consequently, the shunt MUST always be placed on the negative side of the battery pack as shown in Figure 6.1. Otherwise, the BMS may be irreparably damaged.

To provide an accurate measurement, the shunt resistance must be well defined and have a low temperature coefficient. High quality shunt resistors are available with resistance variations below 0.5% at room temperature. Typical temperature coefficients are 0.002%/K.

The accuracy of the shunt resistance will affect the accuracy of the SoC calculations, and it is therefore recommended to use high quality shunts. Typically, an inaccuracy of e.g. 0.5% of the shunt resistance can lead to inaccuracies of up to 0.5% for the SoC values calculated by the BMS.

Shunt resistors have very low Ohmic resistances in the $\mu\Omega$ range. Low resistance reduces the risk of overheating the shunt resistor and reduces the shunt related power loss (P) in the system where P = R x I^2.

The BMS shunt measurement circuit can measure voltages between -250 and +250 mV. When selecting the resistance of a shunt, it is important not to exceed these limits for all specified operating currents in the system. At the same time, it is recommended to use a significant part of the voltage dynamic range in order to obtain a good current resolution.

Lithium Balance can provide shunt resistors for

- nominal currents up to 500 A
- low resistance variations of $\leq$±0.25% at 25 $^{\circ}$C
- operating temperatures between -40 and +60 $^{\circ}$C
- electrical isolation up to 750 V
- temperature coefficients of 0.002%/K

Please refer to section 8.1.1 for more information.

Figure 6.2: Example of shunt resistor .

6.3.2 Hall effect sensors for current measurements

A Hall effect sensor is a transducer that varies its output voltage in response to a magnetic field. When this magnetic field is generated by the battery current, the Hall effect sensor can be used for measuring the main current (I).

The BMS supports Hall sensors powered 5V and providing up to two isolated outputs at voltages between 0 and 5V. Here 2.5V corresponds to 0A. Lithium Balance recommends the use of the DHAB S/34 Hall effect sensors from LEM. These can be provided by Lithium Balance (ordering code 000637)

6.3.3 Temperature sensors

The Lithium Balance BMS is designed for temperature sensing, based on negative temperature coefficient (NTC) thermistors. NTC thermistors are non-linear resistors where the resistance decreases as the temperature increases.

The NTC temperature dependence is typically characterised by the parameter 'β', where

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{\beta} ln(\frac{R}{R_0}) 1/T = 1/T_0 + 1/ln\,(R/R_0) \tag{6.1}$$

Here, R_0 is the reference resistance (typically 25 $^{\circ}$C = 298.15 K) and R is the measured resistance at the temperature T. All temperatures including β are measured in K. Please refer to section 8.1.3 for more information.

6.4 Contactors

Contactors are essential safety mechanisms that provide a redundant means for shutting down the current to or from the battery, if a safety critical event is about to happen. Contactors are used for several different purposes in battery systems. Typical examples are:

- To switch the load in or out
- To switch the charger in and out
- To ensure that the battery in the off-state is galvanically disconnected from the external battery pack terminals
- For precharge/soft-start, as described in section 3.8.

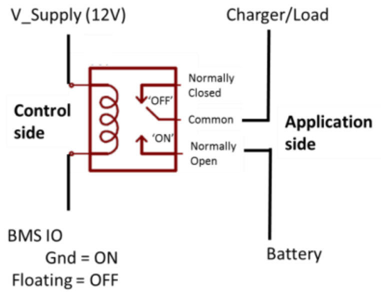


Figure 6.3: Typical configuration of a contactor.

In a contactor, an electromagnet on the 'control side' is used to switch a mechanical contact on the 'application side' between an ON and an OFF position.

The current needed to drive the electromagnet in the contactor is typically provided by coupling the 12 V external power source to ground via the contactor control terminals to the BMS IO as shown in Figure 6.3.

To reduce the steady state power consumption of the contactor, a so-called 'coil economiser' is often included. The coil economiser will reduce the steady state, current but will often also introduce a large capacitance which can lead to significant in-rush currents when the contactor is switched on. This in-rush current level is important to consider, if the supply voltage to the contactor is taken from current limited sources. Typically, a battery can easily provide the required in-rush current, where a DC/DC converter must be selected carefully to be able to handle the in-rush current.

When a contactor is switched off and the current in the magnetic coil on the control side suddenly drops, a voltage peak may appear, since a rapid current change in an inductor will introduce a significant voltage, i.e. $V = L \times dI/dt$[1].

To prevent voltage (V) induced damage to the BMS IO port when the contactor is turned off, it is mandatory to use a so-called flyback diode in parallel with the contactor electromagnet as shown in Figure 6.4. Some contactors, such as those supplied by Lithium Balance come with built-in flyback diodes.

[1] V is the induced voltage, L is the inductance and dI/dt is the change in current

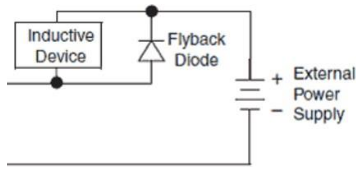


Figure 6.4: Flyback diode

6.5 Fuel gauges

In some systems, it is desired to be able to display the SoC value on a dedicated display, here called a fuel gauge. Dedicated fuel gauge displays are typically used in retrofit systems where the Li-ion battery pack replaces another power source in an already developed application. Examples could be:

- vehicles which are converted from combustion engines to Li-ion powered electrical vehicles
- electrical vehicles which are converted from using lead-acid batteries to using Li-ion batteries

The BMS can interface with SoC fuel gauges via the CAN bus. Lithium Balance recommends the use of CAN-based fuel gauges and can provide these (ordering code 100538).

Please note that the full functionality of the 100538 and possibly other CAN-based fuel gauges are not supported by the BMS rel. 2.0 software.

6.6 DC/DC converter

DC/DC converters are often used in (low-voltage) battery systems where a 12 V power source such as a lead acid battery is not available.

In this case, the 12 V power to the BMS can be provided by a DC/DC converter fed from the main battery as shown in Figure 6.5.

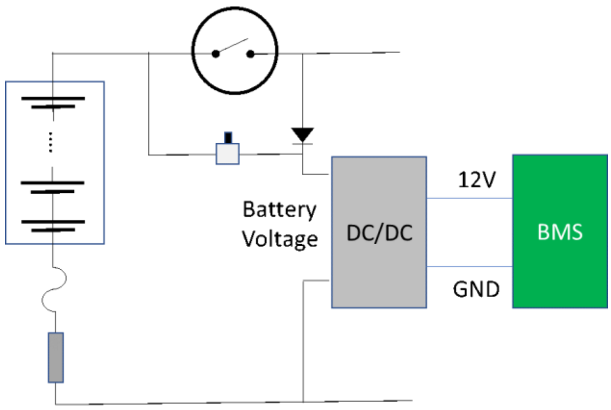


Figure 6.5: Battery pack where the BMS is powered from the main battery via a DC/DC converter.

When selecting a DC/DC converter for a battery pack system, a number of key aspects are important to consider:

- The battery power to the DC/DC converter should be taken after the contactor. Otherwise, the DC/DC converter may drain the battery when the system is off. To start-up the system again after the contactor has been opened, a spring loaded manual push-button switch (in combination with a diode) can be included as shown in Figure 6.5.
- If the BMS and the rest of the battery system needs to be galvanically isolated from the battery, a galvanically isolated DC/DC converter must be used.

- If 12 V powered switches are used in the system, the DC/DC converter must be able to provide sufficient current both for the steady state operation of the switch AND the contactor in-rush current. Alternatively, a contactor powered directly from the battery pack can be used. However, high voltage contactors may compromise a desired galvanic isolation.

7 BMS Quick Installation Guide

The purpose of the “Quick Installation Guide” is – with reference to the manual - to provide an overview of the steps recommended for the hardware installation and the use of the BMS based on configuration via the BMS Creator and monitoring the performance via the Bus-master.

Safety: Please read the safety instructions before installation as described in section 1.3 of the manual.

7.1 Preparation

The following components are ideally required for the full system installation:

- Mandatory for basic setup
  - BMS boards
  - BMS cable kits
  - BMS creator on an PC running Windows operating system
  - Peak PCAN CAN Adapter
  - 500 mA quick blow fuse (for the 12V power supply line)
- Mandatory for systems including batteries
  - Battery Pack (or simulator)
  - Shunt or Hall sensor
  - Contactor(s)
  - Main battery fuse (battery short circuit)
- Optional as per system architecture
  - Charge contactor
  - Pre-charge contactor and pre-charge resistor
- External interfaces
  - Charger
  - Load
- Support equipment and consumables
  - Termination Resistors for CAN (120 ohms)
  - Wire Stripper

7.2 Typical configuration – c-BMS

An example of a typical c-BMS configuration is shown in Figure 7.1 below:

- The c-BMS board is connected via the three cables 100930, 100931 and 100932
- The c-BMS communicates with the charger via CAN
- The c-BMS communicates with a PC on CAN which is adapted to USB in the PEAK adapter
  - Configuration of the BMS is done via the ‘BMS creator’ software licensed from Lithium Balance
  - Reading actual operational parameters (e.g actual cell voltages) are done via a CAN bus monitor tool.

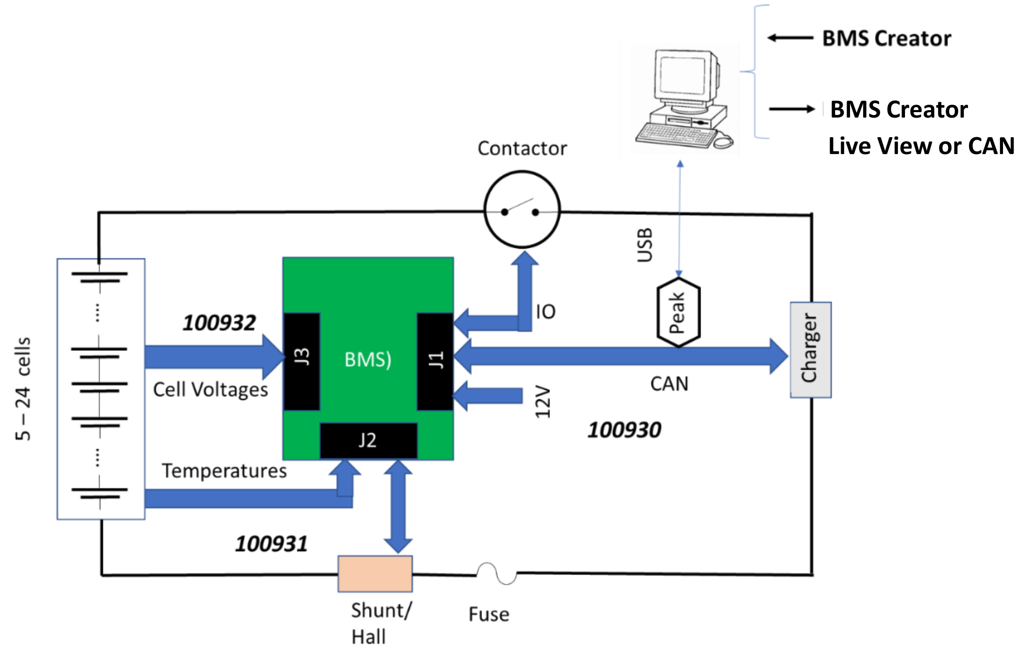


Figure 7.1: Typical c-BMS configuration

7.3 Connect Power and PC to the BMS

The first recommend steps in the BMS installation is to power-up the BMS and connect the BMS to the PC tools, i.e BMS Creator.

1. Connect the board main power (nominally 12V) to the BMS board (for connections see Table 2.1 / Error! Reference source not found.). After successful installation, the red LED will blink once
2. Connect the s-CAN bus to the PC via the Peak CAN adaptor
3. Install the BMS Creator Software as described in section 5.2 and connect the PC to the BMS as described in section 5.3.
   a. Verify that the BMS Creator communicates with the BMS by verifying that clicking connect will change BMS Creator status to “CONNECTED”.

   b. Verify that BMS creator Live View is showing BMS readings. Note that initially you might see some error messages. It must be due to the initial connection. Once you configure your BMS parameters and write a new configuration file to BMS hardware, errors will be cleared. Ideally, you must see no errors after configuring your BMS properly.

Figure 7.2: Typical BMS Creator live view menu.

4. Alternative way for reading Data: Install a CAN bus monitor tool e.g. Busmastermaster as described in section 8.11 Users can configure CAN messages containing BMS variables such as cell voltages, pack voltage, pack current, temperatures, error status, BMS status etc. Likewise, user can also create corresponding dbc or dbf file to use in a CAN bus monitor tool e.g. Busmaster.

7.3.1 Handling CAN data base configuration files (*.dbc or *.dbf)

BMS creator allow users to configure their own CAN messages with configurable CAN ID, DLC, bit and byte location, update rate, endian format. So, everything is configurable and there is no customized or constant CAN IDs embedded. For that reason, there is also no pre-prepared dbc file. There are various tools available, and one example to that is Busmaster. To read the data transmitted form the BMS on the CAN bus, user can create your database set-up (*.dbc or *.dbf) files for data reading programs such as BusMaster. The database files contain the information of the CAN IDs, data formats, location of the useful data etc. in a way that BusMaster or other programs can use to interpret CAN messages and to convert and monitor them as physical values. This physical value can be a cell voltage in mV, or pack current information in amperes etc.

It is possible for the user to change CAN settings in BMS creator e.g. add to a new variable in a CAN frame, change the orientation of the variable, change the location of the variable within a CAN frame, delete a CAN frame, or change CAN ID. Examples of this is given in section 5.9.2.

When any change is done in the BMS Creator CAN setting, please remember also to update the CAN reading database (*.dbc or *.dbf) files accordingly. Examples of how to do this for Busmmaster files are given in section 8.11.

7.4 Connect the BMS to the battery

Step	Connect	Verify
2.1	Connect battery cell voltages to the CMUs as described in section **??**. Start from the bottom and connect Cell1- on the first CMU to the lowest voltage on the battery. When adding a new CMU (e.g. CMU4), then connect this to the previous CMU (e.g CMU3) via the isoSPI connections as decribed in section **??** When all cell voltages have been connected • Configure Voltage Channels (Figure 3.4) • Configure Cell voltage thresholds (Figure 3.10)	Verify on CAN output that all cell voltages are found.
2.2	Connect CMU temperature sensors to the cells, see section **??**. When all cell temperatures have been connected • Configure temperature channels (Figure 3.2). • Configure temperature thresholds (Figure 3.9).	Verify on CAN output that all temperatures are found.
2.3	Connect the current sensing device Shunt or Hall sensor (J2 and 100931) see section **??**. • Configure Shunt/Hall settings, (Figure 3.1) • Configure Current thresholds in the BMS Creator (see chapter 5)	When the BMS is operating for the first time verify that the current measured by the BMS corresponds to the actual current in the application.

7.5 Connect the digital Input and Output signal to contactors and control signals

3.1	Connect output signals to contactors • Configure Contactors	Pending first activation of active mode.
3.2	Connect digital input control signals • Configure input signals	Pending first activation of active mode.

7.6 Connect charger and load

4.1	Connect Charger • Configure Charger communication (CAN or PWM) • Optimize PID settings for 'request current'	Consult charger manufacturer for CAN bus configuration parameters like Current and Voltage request location in frame. Frame ID etc.
4.2	Connect load	Test system and verify current measurement (Hall or Shunt measurement). Remember Current In = Charging = positive current.

7.7 Test and verification

Before the BMS is put into operation, the operation of the BMS within the battery system must be validated under supervision. Specifically, it is recommended to actively verify the key safety aspects listed below during a supervised full charging and balancing of the battery:

- Safe operating limits of voltage, current and temperature.
- The ability of the system to reduce or stop charging when the maximum cell voltage has been reached for any cell.
- A robust and well-tested communication between the BMS and the charger.
- Well-functioning contactors which can disconnect the charger and load from the battery in case direct communication between the BMS and these units fail.

Furthermore, please note that the ability to block excessive current from the battery (preferably by using a fuse) is essential to the safe operation and handling of batteries.

8 Appendices

8.1 Appendix 1: Parts and ordering codes

8.1.1 Shunts

Part number	Description	Comment
100682	200 A/50 mV	Compact
100683	300 A/50 mV	Compact
100684	500 A/50 mV	Compact
100452	150 A/150 mV	
100399	300 A/100 mV	
100400	150 A/150 mV	
100402	500 A/100 mV	

8.1.2 Hall effect Sensors

Part number	Description	Comment
100835	DHAB S/106 Hall effect sensors	Compact
100836	DHAB S/133 Hall effect sensors	Compact

8.1.3 Thermistors

8 These thermistors can be ordered separately. One is part of the 100803 100930 cable accessory R0 = 10 kΩ at 25 ˚C and β = 3900 K. R_0 tolerance = 5%

Part number	Description	Comment
000510.1	150 mm wire	Contains Vishay thermistor NTCLE100E3103JB0

A 5% tolerance corresponds to a potential temperature errors around 1 ˚C, i.e. 0.75 ˚C at -20 ˚C, 1.0 ˚C at 20˚C and 1.7 ˚C at 100 ˚C

8.1.4 Contactors

Part number	Description	Comment
000014	500 A	Main switch
000015	100 A	Typical for pre-charge

8.1.5 Fuse

Part number	Description	Comment
000560	500 mA quick blow	For 12 V power fuse

8.1.6 Pre-charge resistors

Part number	Description	Comment
000018	47 Ω	
000016	100 Ω	
000389	150 Ω	

8.1.7 Fuel gauge

Part number	Description	Comment
100538	CAN based	

8.1.8 DC/DC converters

Part number	Description	Comment
000433 hline 000489	9-36 V input 18-75 V input	12 V and 2.5 A output 12 V and 2.5 A output

8.1.9 Cables and accessories – c-BMS

Part number	Description	Comment
100930	J1 Cable	One 000510.1 thermistor is included
100931	J2 Cable	
100932	J3 Cable	

100802:

000807
000808
000847

100803.300

- Pack monitoring temp sensor 000510.1 - Thermistor
- Temp sensor wire 1000 m
- Serial sensor – ISOSPI cable 300 mm
- Voltage sensor cable

100803.1000

100803.5000

8.2 Firmware error codes

8.2.1 Error code origins

Dec.	Hex	Origin identifier
0	0000	ERROR_ORIGIN_NONE
1	0001	ERROR_ORIGIN_CPU
2	0002	ERROR_ORIGIN_SPI_HAL
3	0003	ERROR_ORIGIN_ADC_HAL
4	0004	ERROR_ORIGIN_SPI_SYS
5	0005	ERROR_ORIGIN_CAN_SYS
6	0006	ERROR_ORIGIN_ADC_EXT
7	0007	ERROR_ORIGIN_MAX14661
8	0008	ERROR_ORIGIN_LTC68XX
9	0009	ERROR_ORIGIN_PACK_AND_CELL
10	000A	ERROR_ORIGIN_CAN
11	000B	ERROR_ORIGIN_STATE_MANAGER
12	000C	ERROR_ORIGIN_PSU_MANAGER
13	000D	ERROR_ORIGIN_ERROR_HANDLING
14	000E	ERROR_ORIGIN_BAT_DIAG
15	000F	ERROR_ORIGIN_CHG_CTRL
16	0010	ERROR_ORIGIN_CMU
17	0011	ERROR_ORIGIN_RTC
18	0012	ERROR_ORIGIN_UDS
19	0013	ERROR_ORIGIN_ID_MAP
20	0014	ERROR_ORIGIN_MAIN
21	0015	ERROR_ORIGIN_M95640
22	0016	ERROR_ORIGIN_BOOT
23	0017	ERROR_ORIGIN_FLASH_SYS
24	0018	ERROR_ORIGIN_CRC
25	0019	ERROR_ORIGIN_GPIO
26	001A	ERROR_ORIGIN_INTERPOLATION
27	001B	ERROR_ORIGIN_QUEUE_HANDLER
28	001C	ERROR_ORIGIN_SRAM_INTERFACE
29	001D	ERROR_ORIGIN_PCF2127
30	001E	ERROR_ORIGIN_TASK_MONITOR
31	001F	ERROR_ORIGIN_MEASURE_AND_MANAGE
32	0020	ERROR_ORIGIN_CDP
33	0021	ERROR_ORIGIN_ERROR_LOG
34	0022	ERROR_ORIGIN_MAIN_APP
35	0023	ERROR_ORIGIN_DEBUG_PORT
36	0024	ERROR_ORIGIN_CAN_RX
37	0025	ERROR_ORIGIN_CAN_TX
38	0026	ERROR_ORIGIN_CURRENT_LIM

8.2.2 Error codes

Dec.	Hex	Error identifier	Explanation
0	0	ERROR_NOERROR	When there is no error.
2	2	ERROR_GENERAL_IOB	Internal values out of bounds. This error occur at e.g. calculation overflow, wrong indexing, etc. Root cause can be caused by out of bound configuration.
4	4	ERROR_GENERAL_RESACC	Resource Access failure, e.g. cannot access config data for enabled CMU cell positions, cannot access input status for ignition and wake-up signals, cannot access and set PID values for charger current output, etc.
7	7	ERROR_GENERAL_PAOB	Parameter out of bounds. Root cause is typically a configuration error in BMS Creator. E.g. CAN settings have wrong numbers for CAN data start bit and length that place data outside the maximum 8 data bytes.(SYSTEM error)
13	D	ERROR_GENERAL_QUEUE_OVERRUN	Overrun in internal software queues like CAN queues or error queue.
15	F	ERROR_GENERAL_COM	Error used to report an error in the internal SPI hardware for communicating with CMUs.(SYSTEM error)
276	114	ERROR_SRAM_WRITE_ERROR	Write error for RTC device
304	130	ERROR_CAN_INVALID_RX_DATA	Internal software detection of unknown received CAN data. Rx CAN frames not configured as Rx frames or UDS frames triggers this error. Note that other CAN frames than the configured RX CAN frames and UDS frames are filtered out before detecting this error.
306	0132	ERROR_CAN_RX_TIMEOUT	When the CAN settings for à RX frame [n] Timeout interval is different than 0, it means that BMS is waiting a CAN message on CAN bus within some time (1 means within 100 ms, 2 means within 200 ms etc.). If BMS does not detect any CAN message according to RX frame set by user, this error pops up.(SYSTEM error)
400	190	ERROR_IO_INPUT_UPDATE	BMS fails when reading the GPIO input states
401	191	ERROR_TASK_MON_EXECUTION_TIME_VIOLATION	states that a task has taken longer to executed than allowed
402	192	ERROR_TASK_MON_TIME_BETWEEN_EXECUTIONS_VIOLATION	states that a task does not execute at the correct intervals
403	193	ERROR_TASK_MON_TASK_DEAD	Detected parts of the software does not run
600	258	ERROR_ADC_EXT_GENERAL_ERROR	Analog to Decimal Conversion (ADC) error, e.g. interfacing problems

601	259	ERROR_ADC_EXT_INIT	Initialization error of the ADC
610	262	ERROR_ADC_EXT_START_SHUNT	Starting of the ADC measurement for the shunt input failed
611	263	ERROR_ADC_EXT_READ_SHUNT	Reading of the ADC measurement result for the shunt failed
620	26C	ERROR_ADC_EXT_START_HVPLUS	Starting of the ADC measurement for the HV input failed
621	26D	ERROR_ADC_EXT_READ_HVPLUS	Reading of the ADC measurement result for the HV failed
630	276	ERROR_ADC_EXT_START_HALL	Starting of the ADC measurement for the HALL input failed
631	277	ERROR_ADC_EXT_READ_HALL	Reading of the ADC measurement result for the HALL failed
901	385	ERROR_ADC_INTERNAL_LOGIC	Is a summation of error 902 and 903
902	386	ERROR_ADC_MCU_FIFO_SIZE	Internal error if the FIFO memory if wrongly sized in the R&D set-up. The FIFO timing is fixed so the error can never be activated
903	387	ERROR_ADC_MCU_TIMING	If the internal ADC is not reporting the conversion completion in time
1001	3E9	ERROR_COMPARISON_FAILED	When communicating with the CMUs, the BMCU reads back data. This error is set if the data does not match the data that was send to CMUs. For example, ADC in MCU after reading the cell voltages is not cleared in it's register. (SYSTEM error)
1006	3EE	ERROR_BALANCING_OPERATION	Write operation of enabled/disabled balancing resistors to the LT device failed
1100	44C	ERROR_TEMP_AUX_NO_VALUE_CH01	Special case where the reading of the temperature sensor results in out of range values.
1101	44D	ERROR_TEMP_AUX_NO_VALUE_CH02	Special case where the reading of the temperature sensor results in out of range values.
1102	44E	ERROR_TEMP_AUX_NO_VALUE_CH03	Special case where the reading of the temperature sensor results in out of range values.
1103	44F	ERROR_TEMP_AUX_NO_VALUE_CH04	Special case where the reading of the temperature sensor results in out of range values.
1104	450	ERROR_TEMP_AUX_NO_VALUE_CH05	Special case where the reading of the temperature sensor results in out of range values.
1105	451	ERROR_TEMP_AUX_NO_VALUE_CH06	Special case where the reading of the temperature sensor results in out of range values.

1116	45C	ERROR_TEMP_AUX_SHORTED_CH01	When reading of temperature sensor resistance is relatively so low, i.e. short circuit. This error indicates that the temperature sensor pins are reading a short circuit. A short circuit may occur on temperature sensor itself, or on the wires poorly implemented, or on the connector or pins if there is a poor implementation. Measured temperature sensor resistance is too low with a resulting temperature above 120°C, i.e. short circuit temperature input.
1117	45D	ERROR_TEMP_AUX_SHORTED_CH02	When reading of temperature sensor resistance is relatively so low, i.e. short circuit. This error indicates that the temperature sensor pins are reading a short circuit. A short circuit may occur on temperature sensor itself, or on the wires or connector or pins. Measured temperature sensor resistance is too low with a resulting temperature above 120°C, i.e. short
1118	45E	ERROR_TEMP_AUX_SHORTED_CH03	When reading of temperature sensor resistance is relatively so low, i.e. short circuit. This error indicates that the temperature sensor pins are reading a short circuit. A short circuit may occur on temperature sensor itself, or on the wires or connector or pins. Measured temperature sensor resistance is too low with a resulting temperature above 120°C, i.e. short
1119	45F	ERROR_TEMP_AUX_SHORTED_CH04	When reading of temperature sensor resistance is relatively so low, i.e. short circuit. This error indicates that the temperature sensor pins are reading a short circuit. A short circuit may occur on temperature sensor itself, or on the wires or connector or pins. Measured temperature sensor resistance is too low with a resulting temperature above 120°C, i.e. short
1120	460	ERROR_TEMP_AUX_SHORTED_CH05	When reading of temperature sensor resistance is relatively so low, i.e. short circuit. This error indicates that the temperature sensor pins are reading a short circuit. A short circuit may occur on temperature sensor itself, or on the wires or connector or pins. Measured temperature sensor resistance is too low with a resulting temperature above 120°C, i.e. short

1121	461	ERROR_TEMP_AUX_SHORTED_CH06	When reading of temperature sensor resistance is relatively so low, i.e. short circuit. This error indicates that the temperature sensor pins are reading a short circuit. A short circuit may occur on temperature sensor itself, or on the wires or connector or pins. Measured temperature sensor resistance is too low with a resulting temperature above 120°C, i.e. short

1132	46C	ERROR_TEMP_AUX_OPEN_CH01	Error pops up when BMS expects a temperature sensor, detects Open Circuit resistance, i.e. cannot detect any temp. sensor. The reason of seeing this error would be a defect temperature sensor (open circuit), or poor connection leading on open circuit, or wrong configuration in BMS Creator Software. Measured temperature sensor resistance is too high with a resulting temperature below -40°C, i.e. open circuit temperature input. Disable the sensor in the configuration if the temperature sensor input not is in use
1133	46D	ERROR_TEMP_AUX_OPEN_CH02	Error pops up when BMS expects a temperature sensor, detects Open Circuit resistance, i.e. cannot detect any temp. sensor. The reason of seeing this error would be a defect temperature sensor (open circuit), or poor connection leading on open circuit, or wrong configuration in BMS Creator Software. Measured temperature sensor resistance is too high with a resulting temperature below -40°C, i.e. open circuit temperature input. Disable the sensor in the configuration if the temperature sensor input not is in use
1134	46E	ERROR_TEMP_AUX_OPEN_CH03	Error pops up when BMS expects a temperature sensor, detects Open Circuit resistance, i.e. cannot detect any temp. sensor. The reason of seeing this error would be a defect temperature sensor (open circuit), or poor connection leading on open circuit, or wrong configuration in BMS Creator Software. Measured temperature sensor resistance is too high with a resulting temperature below -40°C, i.e. open circuit temperature input. Disable the sensor in the configuration if the temperature sensor input not is in use

1135	46F	ERROR_TEMP_AUX_OPEN_CH04	Error pops up when BMS expects a temperature sensor, detects Open Circuit resistance, i.e. cannot detect any temp. sensor. The reason of seeing this error would be a defect temperature sensor (open circuit), or poor connection leading on open circuit, or wrong configuration in BMS Creator Software. Measured temperature sensor resistance is too high with a resulting temperature below -40°C, i.e. open circuit temperature input. Disable the sensor in the configuration if the temperature sensor input not is in use
1136	470	ERROR_TEMP_AUX_OPEN_CH05	Error pops up when BMS expects a temperature sensor, detects Open Circuit resistance, i.e. cannot detect any temp. sensor. The reason of seeing this error would be a defect temperature sensor (open circuit), or poor connection leading on open circuit, or wrong configuration in BMS Creator Software. Measured temperature sensor resistance is too high with a resulting temperature below -40°C, i.e. open circuit temperature input. Disable the sensor in the configuration if the temperature sensor input not is in use
1137	471	ERROR_TEMP_AUX_OPEN_CH06	Error pops up when BMS expects a temperature sensor, detects Open Circuit resistance, i.e. cannot detect any temp. sensor. The reason of seeing this error would be a defect temperature sensor (open circuit), or poor connection leading on open circuit, or wrong configuration in BMS Creator Software. Measured temperature sensor resistance is too high with a resulting temperature below -40°C, i.e. open circuit temperature input. Disable the sensor in the configuration if the temperature sensor input not is in use

1148	47C	ERROR_TEMP_AUX_MIN_CH01	Temperature is below the temperature limit specified in the BMS Creator for this aux sensor.
1149	47D	ERROR_TEMP_AUX_MIN_CH02	Temperature is below the temperature limit specified in the BMS Creator for this aux sensor.
1150	47E	ERROR_TEMP_AUX_MIN_CH03	Temperature is below the temperature limit specified in the BMS Creator for this aux sensor.
1151	47F	ERROR_TEMP_AUX_MIN_CH04	Temperature is below the temperature limit specified in the BMS Creator for this aux sensor.
1152	480	ERROR_TEMP_AUX_MIN_CH05	Temperature is below the temperature limit specified in the BMS Creator for this aux sensor.
1153	481	ERROR_TEMP_AUX_MIN_CH06	Temperature is below the temperature limit specified in the BMS Creator for this aux sensor.
1164	48C	ERROR_TEMP_AUX_MAX_CH01	Temperature is above the temperature limit specified in the BMS Creator for this aux sensor.
1165	48D	ERROR_TEMP_AUX_MAX_CH02	Temperature is above the temperature limit specified in the BMS Creator for this aux sensor.
1166	48E	ERROR_TEMP_AUX_MAX_CH03	Temperature is above the temperature limit specified in the BMS Creator for this aux sensor.
1167	48F	ERROR_TEMP_AUX_MAX_CH04	Temperature is above the temperature limit specified in the BMS Creator for this aux sensor.
1168	490	ERROR_TEMP_AUX_MAX_CH05	Temperature is above the temperature limit specified in the BMS Creator for this aux sensor.
1169	491	ERROR_TEMP_AUX_MAX_CH06	Temperature is above the temperature limit specified in the BMS Creator for this aux sensor.

2000	7D0	ERROR_SYS_LIM_CELL_V_MIN	One or more cell voltages below the limit specified in BMS Creator in Operational Limits, Min. Cell voltage parameter. (PACK error)
2001	7D1	ERROR_SYS_LIM_CELL_V_MAX	One or more cell voltages above the limit specified in the BMS Creator in Operational Limits, Max. Cell voltage parameter. (PACK error)
2004	7D4	ERROR_SYS_LIM_CELL_T_MIN	One or more sensors measure temperature below the limit configured in BMS Creator, Operational Limits menu, Min. cell temperature.
2005	7D5	ERROR_SYS_LIM_CELL_T_MAX	One or more sensors measure temperature above the limit configured in BMS Creator, Operational Limits menu, Max. cell temperature.
2008	7D8	ERROR_SYS_LIM_PACK_I_IN	The charge current floating into the battery is above the limits configured in BMS Creator, Operational limits menu, DCLI parameters. DCLI limits are supervised in both charge and discharge mode.
2009	7D9	ERROR_SYS_LIM_PACK_I_OUT	The charge current floating out of the battery is above the limits configured in BMS Creator, Operational limits menu, DCLO parameters. DCLO limits are supervised in both charge and discharge mode.
2010	7DA	ERROR_SYS_LIM_PACK_I2T	Error pops up when the limit has been exceeded. For more information, refer to the section called "Advanced current limitation, i2t".(PACK error)
2011	7DB	ERROR_SYS_CELL_V_NO_VALUE	Special case where the reading of the cell voltage results in extreme values.(PACK error)
2012	7DC	ERROR_SYS_CELL_T_NO_VALUE	Special case where the reading of the cell temperature results in extreme values.(PACK error)

2013	7DD	ERROR_SYS_CELL_T_SHORTED	Measured cell temperature sensor resistance is too low with a resulting temperature above 120°C, i.e. short circuit temperature input.
2014	7DE	ERROR_SYS_CELL_T_OPEN	Measured cell temperature sensor resistance is too high with a resulting temperature below -40°C, i.e. open circuit temperature input. Disable the cell temperature sensors in the configuration if the sensor inputs not are in use.
2015	7DF	ERROR_SYS_CMU_PCB_T_NO_VALUE	Special case where the reading of the temperature sensor on CMU PCB results in extreme values.(PACK error)
2016	7E0	ERROR_SYS_CMU_PCB_T_SHORTED	Measured PCB temperature sensor resistance is too low with a resulting temperature above 120°C.
2017	7E1	ERROR_SYS_CMU_PCB_T_OPEN	Measured PCB temperature sensor resistance is too high with a resulting temperature below -40°C.
2018	7E2	ERROR_SYS_PACK_I_MASTER _SELECTION	Configuration of Current sensor type out of allowed range
2019	7E3	ERROR_SYS_COMM_CMU	When BMS cannot communicate with one or more CMUs.(SYSTEM error)
2020	7E4	ERROR_SYS_NUM_CMU_MISMATCH	When BMS detects more or less number of CMUs specified in BMS Creator.
2021	7E5	ERROR_SYS_FB_LOAD_NEG_MISSING	The BMS receives wrong feedback input for load negative contactor. This could be because of missing feedback signal, or wrong GPIO configuration. Note that only a separate GPIO can be configured as feedback input per GPIO output function.
2022	7E6	ERROR_SYS_FB_LOAD_POS_MISSING	When BMS cannot receive a feedback input for load positive contactor. This could be due to missing the actual feedback message, or due to wrong GPIO settings. User must only define a feedback input in GPIO menu in BMS Creator, when there is a feedback intentionally used in the design.
2023	7E7	ERROR_SYS_FB_PRECHARGE_MISSING	When BMS cannot receive a feedback input for pre-charge contactor. This could be due to missing the actual feedback message, or due to wrong GPIO settings. User must only define a feedback input in GPIO menu in BMS Creator, when there is a feedback intentionally used in the design.

2024	7E8	ERROR_SYS_FB_CHG_NEG_MISSING	When BMS cannot receive a feedback input for charge negative contactor. This could be due to missing the actual feedback message, or due to wrong GPIO settings. User must only define a feedback input in GPIO menu in BMS Creator, when there is feedback intentionally used in the design.
2025	7E9	ERROR_SYS_FB_LOAD_NEG_WELDED	When BMS expects load negative contactor open, and hence does not expect a feedback, but receive a feedback input for load negative contactor. This could be due to a welded contactor, or wrong connection or short circuit in the circuit.
2026	7EA	ERROR_SYS_FB_LOAD_POS_WELDED	When BMS expects load positive contactor open, and hence does not expect a feedback, but receive a feedback input for load positive contactor. This could be due to a welded contactor, or wrong connection or short circuit in the circuit.
2027	7EB	ERROR_SYS_FB_PRECHARGE_WELDED	When BMS expects pre-charge contactor open, and hence does not expect a feedback, but receive a feedback input for pre-charge contactor. This could be due to a welded contactor, or wrong connection or short circuit in the circuit.
2028	7EC	ERROR_SYS_FB_CHG_NEG_WELDED	When BMS expects charge negative contactor open, and hence does not expect a feedback, but receive a feedback input for charge negative contactor. This could be due to a welded contactor, or wrong connection or short circuit in the circuit.
2031	7EF	ERROR_SYS_CONTACTOR_RETRIES	Occurs when the number of re-tries of closing contactors exceeds the configured limit. The BMS will re-try to close contactors after configured errors are cleared. Number of re-tries is configured in Operational limits menu, Max. Contactor retries.
2032	7F0	ERROR_SYS_PROCESS_CMU_DATA	LT device on CMU or c-BMS board reports internal reference voltage error
2033	7F1	ERROR_SYS_PACK_AND_CELL_INIT	Internal detected error of wrongly configuration of LT temperature sensors for the c-BMS
2036	7F4	ERROR_SYS_PACK_SOC_CALC	The state of charge calculation cannot be completed, normally because the ampere hour summation is much too big. This can be caused by having a floating current measurement input for a while.
2038	7F6	ERROR_SYS_EXEC_STATE_LOGIC	Configured GPIO number is out of range
2039	7F7	ERROR_SYS_VSUPPLY_MIN	This is not currently used.
2040	7F8	ERROR_SYS_VSUPPLY_MAX	This is not currently used.

2041	7F9	ERROR_SYS_PSU_NOT_IN_DIAGNOSTIC _MODE	The MCU expects the PSU chip to be in diagnostic mode upon startup, from where it can be properly configured. This error occurs if that is not the case. This error sometimes can occur right after bootloading and it can be ignored if it disappears after a BMS power recycle. If the error is permanent with okay power to the BMS, then it is related to detection of some kind of PSU fault and the board probably will need to be replaced.
2042	7FA	ERROR_SYS_LIM_CHG_I_OVER_UNDER	When BMS is requesting X amount of current but receiving X+ - [allowed charge current dead band] parameter given in BMS creator. For example, BMs is requesting 10 A, allowed charge current dead band is set to 1 A, however shunt is measuring that there is 8 A charge current. It that case, this error message pops up since 8 A is not within 10 + - 1 A = [9 A, 11 A] range.
2043	7FB	ERROR_SYS_LEAK_ADC	The MCU isolation test circuit has measured an internal circuit fault and cannot complete the isolation test
2044	7FC	ERROR_SYS_LEAK_LOAD	The measured isolation resistance is below the configured level , in load mode
2045	7FD	ERROR_SYS_LEAK_CHARGE	The measured isolation resistance is below the configured level , in charge mode
2046	7FE	ERROR_SYS_LEAK_READY	The measured isolation resistance is below the configured level , in ready mode
2047	7FF	ERROR_SYS_TOO_MANY_BROADCAST _ERRORS	Overflow for CAN Tx'ed error messages. Not all active errors are Tx'ed. NOTE: This error is only used for CAN Tx'ed errors and is not displayed in the creator tool or recorded in the error log.
2048	800	SYS_CMU_OPEN_WIRE	The BMS generally throws this error when there is a loose/open connection in the cell monitoring wires. Re-check/re-do the connections and see if this Error persists. Also, make sure that the cell configuration are correct.
2049	7FF	ERROR_Config_Water mark alignment _ERRORS	Corrupted xml configuration file.
2060	80C	ERROR_SYS_UNKNOWN_VOLTAGE	The BMS will raise this Error while detecting an open wire. From the functionality stand point , this Error is similar to the Error-2048.

8.3 Data ID map c-BMS

For configurable CAN setup and UDS request by ID service reference.

ID	Name	Type	Scaling	Unit	Description
1	CELL_V_MIN_VAL	UINT16	0.1	mV	Lowest cell voltage
2	CELL_V_MIN_ID_CMU	UINT8	1	-	CMU of lowest cell voltage
3	CELL_V_MIN_ID_CELL	UINT8	1	-	Cell number of cell w. lowest voltage
4	CELL_V_MAX_VAL	UINT16	0.1	mV	Highest cell voltage
5	CELL_V_MAX_ID_CMU	UINT8	1	-	CMU of highest cell voltage
6	CELL_V_MAX_ID_CELL	UINT8	1	-	Cell number of cell w. highest voltage
7	CELL_V_AVG	UINT16	0.1	mV	Average cell voltage
8	CELL_V_NUM_AVAILABLE	UINT16	1	-	Number of cell voltages configured
9	PACK_V_EXT_LOAD	UINT16	0.1	V	Not implemented in the current c-BMS
10	PACK_V_EXT_CHG	UINT16	0.1	V	Not implemented in the current c-BMS
11	PACK_V_SUM_OF_CELLS	UINT16	0.1	V	Sum of all cells
12	HALL_V_HALL_V_LO	UINT16	0.1	mV	Voltage level from HALL low channel, master
13	HALL_V_HALL_V_HI	UINT16	0.1	mV	Voltage level from HALL high channel, master
14	PACK_I_HALL	INT32	0.01	mA	HALL current
15	PACK_I_SHUNT	INT32	0.01	mA	Shunt current
16	PACK_I_MASTER	INT32	0.01	mA	Selected current source (shunt or hall) will be the system reference current.
17	PACK_Q_REMAINING_HI_RES	INT32	1	As	Capacity remaining
18	PACK_Q_SOC_INTERNAL	INT16	0.01	%	SoC
19	PACK_Q_SOC_TRIMMED	UINT16	0.01	%	SoC
20	PACK_Q_REMAINING_NOMINAL	UINT16	0.1	Ah	Capacity remaining
21	PACK_Q_DESIGN	UINT16	0.1	Ah	Capacity as designed (nominal capacity from cell manufacturer)
22	PACK_Q_FULL	UINT16	0.1	Ah	Capacity at fully charged state
23	IO_STATE_OUTPUT	UINT8	1	-	Not implemented.
24	IO_STATE_INPUT	UINT8	1	-	Bitmask (b0 = I-1...)
25	CHARGER_OUTPUT_ENABLED	UINT8	1	-	b0 is Boolean
26	CHARGER_OUTPUT_CURRENT	UINT16	0.1	A	Current requested from charger
27	CHARGER_OUTPUT_VOLTAGE	UINT16	0.1	V	Voltage requested from charger
28	CHARGER_CAN_ACTIVE	UINT8	1	-	b0 is Boolean
29	CHARGER_PWM_ACTIVE	UINT8	1	-	b0 is Boolean

30	CHARGER_PWM_ACTIVE_DUTY	UINT8	1	%	PWM duty cycle for charger
31	DYN_LIM_I2T_REMAIN	UINT32	1	A2s	Remaining i2t sum before error is triggered
32	DYN_LIM_I_IN	UINT16	0.1	A	Current limit for incoming current (typically charge or regen)
33	DYN_LIM_I_OUT	UINT16	0.1	A	Current limit for outgoing current (typically discharge)
34	STATUS	UINT8	1	-	BMS state
35	CONTACTORS_ENABLED	UINT8	1	-	b0 is Boolean
36	CONTACTORS_ACTIVATE_CHARGE	UINT8	1	-	b0 is Boolean
37	CONTACTORS_ACTIVATE_LOAD	UINT8	1	-	b0 is Boolean
38	CONTACTORS_ACTIVATE_COMBINED	UINT8	1	-	b0 is Boolean
39	CONTACTORS_CHARGER_ACTIVATED	UINT8	1	-	b0 is Boolean
40	CONTACTORS_LOAD_ACTIVATED	UINT8	1	-	b0 is Boolean
41	CONTACTORS_COMBINED_ACTIVATED	UINT8	1	-	b0 is Boolean
42	CONTACTORS_ACTIVATION_ALLOWED	UINT8	1	-	b0 is Boolean
43	CONTACTORS_EMERGENCY_OFF	UINT8	1	-	b0 is Boolean
44	CONTACTORS_CONTACTOR_RETRIES	UINT8	1	-	Count of contactor closing retries
45	BALANCING_ALLOWED	UINT8	1	-	b0 is Boolean. This bit is equal to ID52 FLAGS_BALANCING_ACTIVE.
46	BALANCING_LIMIT	UINT16	0.1	mV	Lower voltage limit for balancing to be active
47	FLAGS_FULLY_CHARGED	UINT8	1	-	b0 is Boolean
48	FLAGS_FULLY_CHARGED_LATCHED	UINT8	1	-	b0 is Boolean
49	FLAGS_LOAD_ACTIVE	UINT8	1	-	b0 is Boolean
50	FLAGS_CHARGER_ACTIVE	UINT8	1	-	b0 is Boolean
51	FLAGS_PRECHARGE_ACTIVE	UINT8	1	-	b0 is Boolean
52	FLAGS_BALANCING_ACTIVE	UINT8	1	-	b0 is Boolean. This bit is equal to ID45 BALANCING_ALLOWED.
53	FLAGS_CHARGE_REG_ACTIVE	UINT8	1	-	b0 is Boolean
54	B_WU_IGN	UINT8	1	-	b0 is Boolean
55	B_WU_CAN	UINT8	1	-	b0 is Boolean
56	VCP	UINT16	1	mV	PSU charge pump
57	VBAT	UINT16	1	mV	PSU supply voltage (For the future release)
58	MAIN	UINT16	1	mV	PSU regulators band-gap reference
59	VMON	UINT16	1	mV	PSU voltage-monitor band-gap
60	Reserved	UINT8	1	-	
61	Reserved	UINT8	1	-	
62	Reserved	UINT8	1	-	
63	ACTIVE_COUNT	UINT16	1	-	Count of total amount of errors

64	PROJECT	UINT16	1	-	Project version
65	HW_PCB	UINT8	1	-	Hardware PCB version (Interpret as Ascii)
66	HW_BOM	UINT8	1	-	Hardware BOM version
67	FW_MAJOR	UINT8	1	-	Firmware major version
68	FW_MINOR	UINT8	1	-	Firmware minor version
69	TYPE	UINT8	1	-	Firmware type (Interpret as Ascii)
70	BUILD_DATE	UINT8 * 11	1	-	Firmware build date (Ascii, Mmm.dd.yyyy)
71	BUILD_TIME	UINT8 * 8	1	-	Firmware build time (Ascii, hh:mm:ss)
72	CMUS_CMU1_BALANCE_T1	INT8	1	°C	CMU balance circuit temperature. 121 = OC, -41 = SC
73	CMUS_CMU1_BALANCE_T2	INT8	1	°C	CMU balance circuit temperature. 121 = OC, -41 = SC
74	CMUS_CMU2_BALANCE_T1	INT8	1	°C	CMU balance circuit temperature. 121 = OC, -41 = SC
75	CMUS_CMU2_BALANCE_T2	INT8	1	°C	CMU balance circuit temperature. 121 = OC, -41 = SC
138	AUX_T1	INT8	1	°C	AUX temperature 0
139	AUX_T2	INT8	1	°C	AUX temperature 1
140	AUX_T3	INT8	1	°C	AUX temperature 2
141	AUX_T4	INT8	1	°C	AUX temperature 3
142	AUX_T5	INT8	1	°C	AUX temperature 4
143	AUX_T6	INT8	1	°C	AUX temperature 5
538	CMUS_ONLINE_COUNT	UINT8	1	-	Count of online CMUs
539	CMUS_COMM_ERR	UINT32	1	-	Counter for CMU communication errors since power on or wakeup
540	CMUS_CMU1_CELL_V1	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[0]
541	CMUS_CMU1_CELL_V2	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[1]
542	CMUS_CMU1_CELL_V3	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[2]
543	CMUS_CMU1_CELL_V4	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[3]
544	CMUS_CMU1_CELL_V5	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[4]
545	CMUS_CMU1_CELL_V6	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[5]
546	CMUS_CMU1_CELL_V7	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[6]
547	CMUS_CMU1_CELL_V8	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[7]

548	CMUS_CMU1_CELL_V9	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[8]
549	CMUS_CMU1_CELL_V10	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[9]
550	CMUS_CMU1_CELL_V11	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[10]
551	CMUS_CMU1_CELL_V12	UINT16	0.1	mV	Cell voltage for cmu[0], cell_v[11]
552	CMUS_CMU2_CELL_V1	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[0]
553	CMUS_CMU2_CELL_V2	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[1]
554	CMUS_CMU2_CELL_V3	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[2]
555	CMUS_CMU2_CELL_V4	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[3]
556	CMUS_CMU2_CELL_V5	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[4]
557	CMUS_CMU2_CELL_V6	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[5]
558	CMUS_CMU2_CELL_V7	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[6]
559	CMUS_CMU2_CELL_V8	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[7]
560	CMUS_CMU2_CELL_V9	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[8]
561	CMUS_CMU2_CELL_V10	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[9]
562	CMUS_CMU2_CELL_V11	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[10]
563	CMUS_CMU2_CELL_V12	UINT16	0.1	mV	Cell voltage for cmu[1], cell_v[11]
955	TIME_BETWEEN_BOOTS_S	UINT32	1	sec	Time between this boot and the last
956	EPOCH_TIME_S	UINT32	1	sec	System time (Epoch time since 00:00:00, 1970/01/01
957	UPTIME_S	UINT32	1	sec	System up time since power on or wakeup
958	HALL_EXT_V_HALL_V_LO	UINT16	0.1	mV	Voltage measurement fromHALL low channel, external ADC
959	HALL_EXT_V_HALL_V_HI	UINT16	0.1	mV	Voltage measurement fromHALL high channel, external ADC
960	HALL_EXT_V_HALL_V_SUPPLY	UINT16	0.1	mV	Voltage measurement fromHALL supply circuit, external ADC
964	HALL_V_HALL_V_SUPPLY	UINT16	0.1	mV	Voltage level from HALL supply, master
965	CELL_T_MIN_VAL	INT8	1	°C	Lowest cell temperature
966	CELL_T_MIN_ID_CMU	UINT8	1	-	CMU of lowest cell temperature
967	CELL_T_MIN_ID_CELL	UINT8	1	-	Cell number of cell w. lowest temperature
968	CELL_T_MAX_VAL	INT8	1	°C	Highest cell temperature
969	CELL_T_MAX_ID_CMU	UINT8	1	-	CMU of highest cell temperature
970	CELL_T_MAX_ID_CELL	UINT8	1	-	Cell number with highest cell temperature

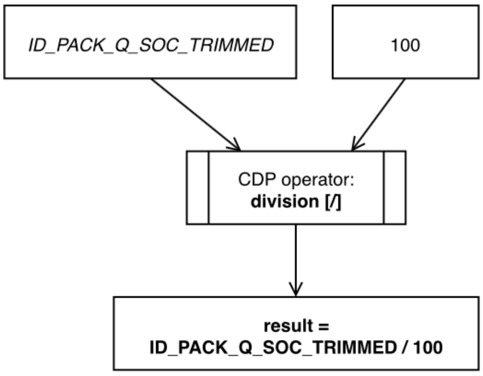
971	CELL_T_AVG	INT8	1	°C	Average cell temperature
972	CELL_T_NUM_AVAILABLE	UINT16	1	-	Number of cell temperature configured
973	BALANCING_BALANCE_SETTING _CMU1_CELL_BITMASK	UINT16	1	-	Cells being balanced as bitmask (b0 = cell 0, ... b11 = cell 11, b12-b15 = always zero)
974	BALANCING_BALANCE_SETTING _CMU2_CELL_BITMASK	UINT16	1	-	Cells being balanced as bitmask (b0 = cell 0, ... b11 = cell 11, b12-b15 = always zero)
1005	CDP_OUTPUT1	UINT32	-	-	Custom Data Processor output number 1
1006	CDP_OUTPUT2	UINT32	-	-	Custom Data Processor output number 2
1007	CDP_OUTPUT3	UINT32	-	-	Custom Data Processor output number 3
1008	CDP_OUTPUT4	UINT32	-	-	Custom Data Processor output number 4
1009	CDP_OUTPUT5	UINT32	-	-	Custom Data Processor output number 5
1010	CDP_OUTPUT6	UINT32	-	-	Custom Data Processor output number 6
1011	CDP_OUTPUT7	UINT32	-	-	Custom Data Processor output number 7
1012	CDP_OUTPUT8	UINT32	-	-	Custom Data Processor output number 8
1013	CDP_OUTPUT9	UINT32	-	-	Custom Data Processor output number 9
1014	CDP_OUTPUT10	UINT32	-	-	Custom Data Processor output number 10
1015	CDP_OUTPUT11	UINT32	-	-	Custom Data Processor output number 11
1016	CDP_OUTPUT12	UINT32	-	-	Custom Data Processor output number 12
1017	CDP_OUTPUT13	UINT32	-	-	Custom Data Processor output number 13
1018	CDP_OUTPUT14	UINT32	-	-	Custom Data Processor output number 14
1019	CDP_OUTPUT15	UINT32	-	-	Custom Data Processor output number 15
1020	CDP_OUTPUT16	UINT32	-	-	Custom Data Processor output number 16
1021	CDP_OUTPUT17	UINT32	-	-	Custom Data Processor output number 17
1022	CDP_OUTPUT18	UINT32	-	-	Custom Data Processor output number 18
1023	CDP_OUTPUT19	UINT32	-	-	Custom Data Processor output number 19
1024	CDP_OUTPUT20	UINT32	-	-	Custom Data Processor output number 20
1025	CDP_OUTPUT21	UINT32	-	-	Custom Data Processor output number 21
1026	CDP_OUTPUT22	UINT32	-	-	Custom Data Processor output number 22
1027	CDP_OUTPUT23	UINT32	-	-	Custom Data Processor output number 23
1028	CDP_OUTPUT24	UINT32	-	-	Custom Data Processor output number 24

1029	CDP_OUTPUT25	UINT32	-	-	Custom Data Processor output number 25
1072	VDD5	UINT16	1	mV	PSU VDD5 regulator output
1073	VDD6	UINT16	1	mV	PSU switch mode pre-regulator
1074	VSFB1	UINT16	1	mV	PSU sensor supply voltage
1075	VBATS	UINT16	1	mV	PSU supply for monitoring and BG2
1076	FLAGS_WAITING_FOR_STAGING_REQ	UINT8	1	-	b0 is Boolean
1077	SOP_INTERVAL_DATA1	UINT16	1	-	SoP data for interval
1078	SOP_INTERVAL_DATA2	UINT16	1	-	SoP data for interval
1079	SOP_INTERVAL_DATA3	UINT16	1	-	SoP data for interval
1080	SOP_INTERVAL_DATA4	UINT16	1	-	SoP data for interval
1081	SOH	UINT16	0.01	%	SoH
1082	VREF2_CMUx	UINT32	0.1	mV	Resolution
1114	CONFIG_ID	UINT32	-	-	-
1115	CHECKSUM_ID	UINT32	-	-	-
50001	CAN_RX_DATA1	UINT32	-	-	CAN RX data storage field 1
50002	CAN_RX_DATA2	UINT32	-	-	CAN RX data storage field 2
50003	CAN_RX_DATA3	UINT32	-	-	CAN RX data storage field 3
50004	CAN_RX_DATA4	UINT32	-	-	CAN RX data storage field 4
50005	CAN_RX_DATA5	UINT32	-	-	CAN RX data storage field 5
50006	CAN_RX_DATA6	UINT32	-	-	CAN RX data storage field 6
50007	CAN_RX_DATA7	UINT32	-	-	CAN RX data storage field 7
50008	CAN_RX_DATA8	UINT32	-	-	CAN RX data storage field 8
50009	CAN_RX_DATA9	UINT32	-	-	CAN RX data storage field 9
50010	CAN_RX_DATA10	UINT32	-	-	CAN RX data storage field 10
50011	CAN_RX_DATA11	UINT32	-	-	CAN RX data storage field 11
50012	CAN_RX_DATA12	UINT32	-	-	CAN RX data storage field 12
50013	CAN_RX_DATA13	UINT32	-	-	CAN RX data storage field 13
50014	CAN_RX_DATA14	UINT32	-	-	CAN RX data storage field 14
50015	CAN_RX_DATA15	UINT32	-	-	CAN RX data storage field 15
50016	CAN_RX_DATA16	UINT32	-	-	CAN RX data storage field 16
50017	CAN_RX_DATA17	UINT32	-	-	CAN RX data storage field 17
50018	CAN_RX_DATA18	UINT32	-	-	CAN RX data storage field 18
50019	CAN_RX_DATA19	UINT32	-	-	CAN RX data storage field 19
50020	CAN_RX_DATA20	UINT32	-	-	CAN RX data storage field 20
50021	CAN_RX_DATA21	UINT32	-	-	CAN RX data storage field 21
50022	CAN_RX_DATA22	UINT32	-	-	CAN RX data storage field 22
50023	CAN_RX_DATA23	UINT32	-	-	CAN RX data storage field 23
50024	CAN_RX_DATA24	UINT32	-	-	CAN RX data storage field 24
50025	CAN_RX_DATA25	UINT32	-	-	CAN RX data storage field 25

8.4 Custom data processing examples

8.4.1 Scaling the SOC from 0.01% resolution to 1 % resolution

1. *lvalue* is addressed data from the ID MAP using ID: ID_PACK_Q_SOC_TRIMMED
2. *rvalue* is constant data with the value 100
3. The operator is division [\]



8.4.2 Scaling and offset of pack current

Configure CDP output #1 with:

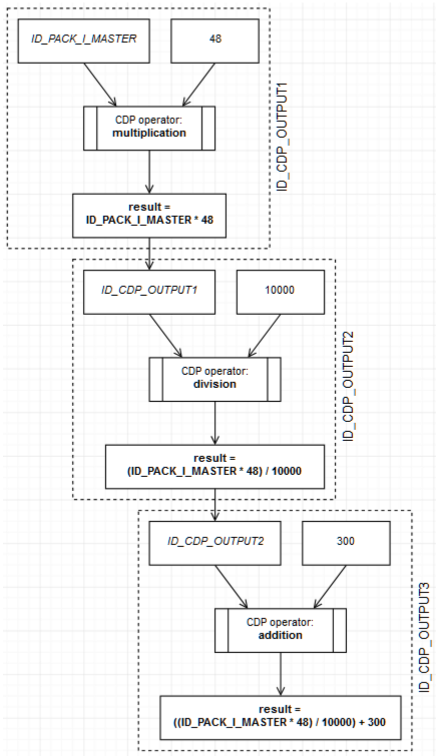
1. lvalue is addressed data from the ID MAP using ID: ID_PACK_I_MASTER
2. rvalue is constant data with value 48
3. operator is multiplication [*]

Result is stored in ID_CDP_OUTPUT1
Configure CDP output #2 with:

4. lvalue is addressed data from the ID MAP using ID: ID_CDP_OUTPUT1 (CDP result from before)
5. rvalue is constant data with value 10000
6. operator is division [/]

Result is stored in ID_CDP_OUTPUT2
Configure CDP output #3 with:

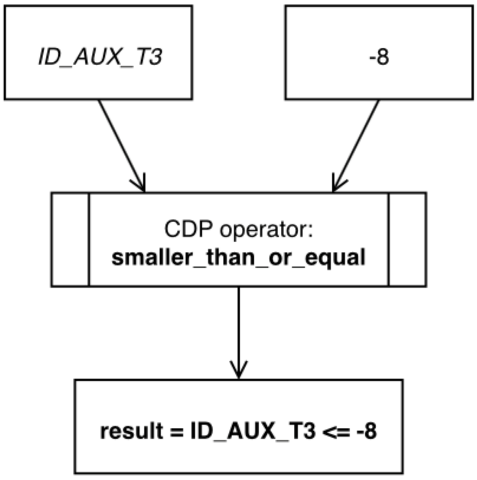
7. lvalue is addressed data from the ID MAP using ID: ID_CDP_OUTPUT2 (CDP result from before)
8. rvalue is constant data with value 300
9. operator is addition [+]
   Result is stored in ID_CDP_OUTPUT3, where it can be configured to be broadcasted on the CAN bus.



8.4.3 Setting up custom warning bit with CDP

1. *lvalue* is addressed data from the ID MAP using ID: ID_AUX_T3
2. *rvalue* is constant data with the value -8
3. The operator is less than or equal [<=]

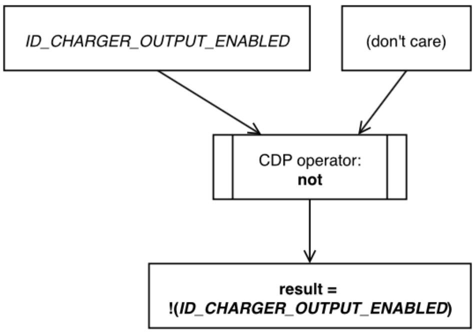
The result is then 1 if ID_AUX_T3 is less than or equal to -8, otherwise the result is 0.



8.4.4 Inverting charger enabled bit

1. *lvalue* is addressed data from the ID MAP using ID: ID_CHARGER_OUTPUT_ENABLED.
2. *rvalue* is not used, but a constant data with the value 0 is used.
3. The operator is not [!].

The result is then the inverse (binary not) of the *lvalue*.



8.5 OCV-SoC experimental battery data example

8.5.1 Introduction

8.5.1.1 Purpose

To use the SoC-OCV calibration functionality available in the BMS-firmware it is necessary to have data sets where OCV is measured against SoC. The purpose of implementing the SoC-OCV calibration functionality into the BMS-firmware is to supplement the BMS SOC-prediction based on coulomb counting.

Such OCV-SOC calibration curves might be obtained from the cell manufacturer or can alternatively be measured by the user. The following sections describes how such measurements have been performed at Lithium Balance.

The objective of the tests performed here has been to measure the relationship between SoC and OCV at constant operating conditions and to investigate the influence of relaxation time on the accuracy of the OCV-values measured.

8.5.1.2 OCV-SOC data requirements

The OCV-SOC input to the BMS should be data set(s) with 101 voltage readings, one for each percentage step from 0 % - 100 %. These data can be provided for up to 5 temperatures. The measurements described here were done at 5 or 10 percentage steps and then the points in between the actual measurements were calculated by linear interpolation from the two nearest data points.

8.5.1.3 The test set-up

To characterize the cell a battery cycler / potentiostat[1]was used to perform full discharge (100% SOC -> 0%) sequences at room temperature and at discharge currents of 0.5 C. The battery cycler is basically a load and a charger combined in one box with the ability to program charge, discharge and rest sequences on several cells and to record data (time, potential, current, temperature).

Four test cells[2]were used, each cell had a nominal capacity on 2450 mAh. 3 test cells were mounted on a prototype board, fused and each connected to a dedicated channel on the battery cycler from the positive and negative terminals on the cells. Further a thermo element was attached on each test cell giving the possibility of measuring the cell temperature along with the charge/discharge sequence. The battery cycler was connected to a PC via ethernet connection and controlled via the software PC-battery lab.

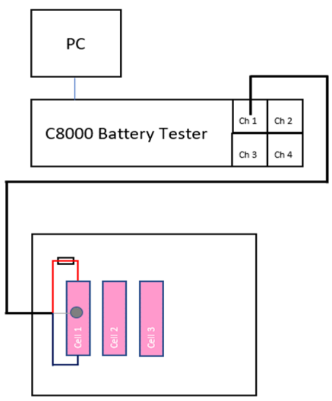


Figure 8.1: Sketch of the test-setup with the test cells mounted on the prototype board connected to the battery tester with cables to the terminals and a magnetic thermo element.

8.6 Experimental results

8.6.1 Nominal capacity

According to the suppliers datasheet, the maximum cell voltage was 4200 mV (SOC = 100%) and the minimum cell voltage was 3000 mV (SOC = 0%).

Initially all test cells were charged to 4200 mV at minimum charge current (which in our case was 250 mA). 20 minutes later the cells were discharged to 3000 mV at 1C. By integrating the (time, current)-curve (coulomb counting) the nominal capacity of each cell (at 1C discharge rate, constant current and 22 °C) was found to be 2196, 2162 and 2217 mAh for the three cells.

8.6.2 OCV-SOC Test sequence

The first step of the test sequence is a full charge as described above. 20 minutes later the discharge steps were initiated. Discharging was done by a constant current at 0.5 C in steps corresponding to SOC percentages (approximately) at 95, 90, 80, 70, 60, 50, 40, 30, 20, 10, 5 and 0. The time to achieve these

[1]Cadex C8000
[2]Panasonic CGR18650DA

SOC steps were calculated from the nominal capacity measured above.

Between each step the cells rested for 20 minutes while the cell voltages were measured.

To avoid too big data files the battery cycler was programmed to log data every second the first and last 30 seconds of a charge or resting period (where the curves are typically showing the highest / lowest slopes) and every 10 seconds in between (where the curves are typically flat).

8.6.3 Results

Graphical representations of the measurements are shown below.

Figure 8.2 shows the voltage response to the current drawn from the 3 cells. The cells are discharged from being fully charged to fully discharged at a C-rate on (approximately) 0.5.

In Figure 8.3, the cell voltages are shown with the SOC calculated from coulomb counting based on the actual measured capacity for each cell.

Figure 8.4 shows the SoC-OCV graph where each cell SoC is plotted against the stabilized OCV (the cell voltages measured after 20 minutes. In Figure 8.4 it is noted that in some regions, 1% change of SOC-scale corresponds to less than 3 mV change of OCV. Consequently, it is essential to assure that the relaxation time when measuring OCV is sufficiently long to assure a complete stabilisation of measured cell voltage. The necessary relaxation time may vary significantly from cell type to cell types and will have to be determined experimentally for each type of cells.

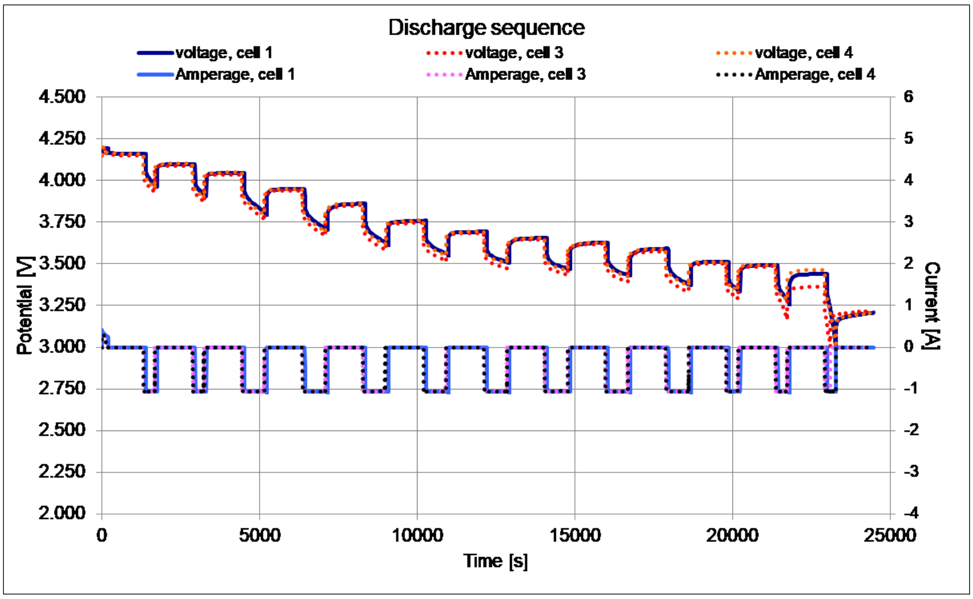


Figure 8.2: The discharge current and the voltage response for each cell during the full discharge sequence is shown. Note the last short discharge step where the cell voltages reach the lower voltage boundary which is made to make sure the cells are fully discharged at the measurement that corresponds to 0 % SoC.

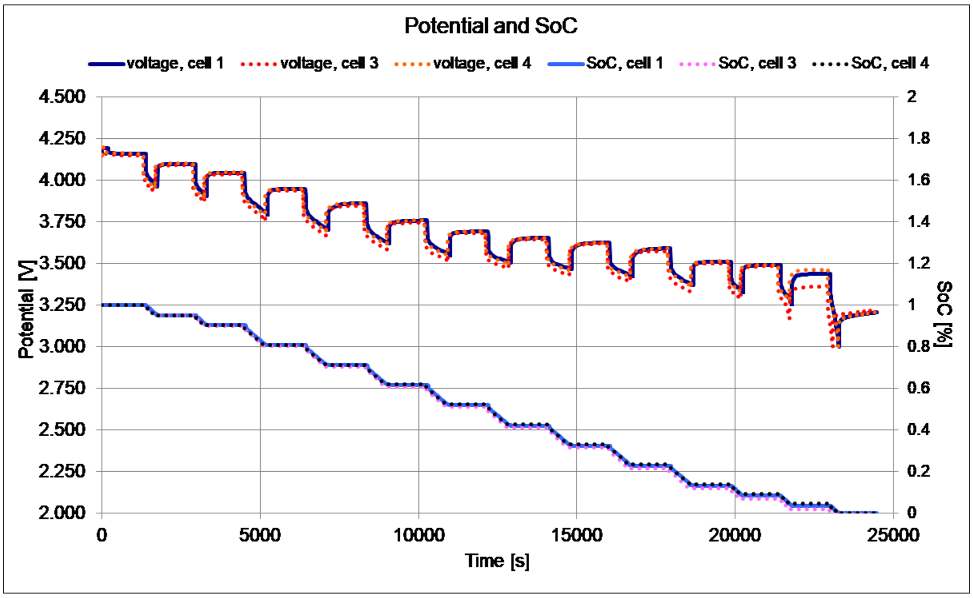


Figure 8.3: Cell voltages and corresponding SOC as calculated individually for each cell based on the actual capacity seen for each cell during discharge. Those are the data that is plotted against each other in the OCV-SoC curve in Figure 8.4.

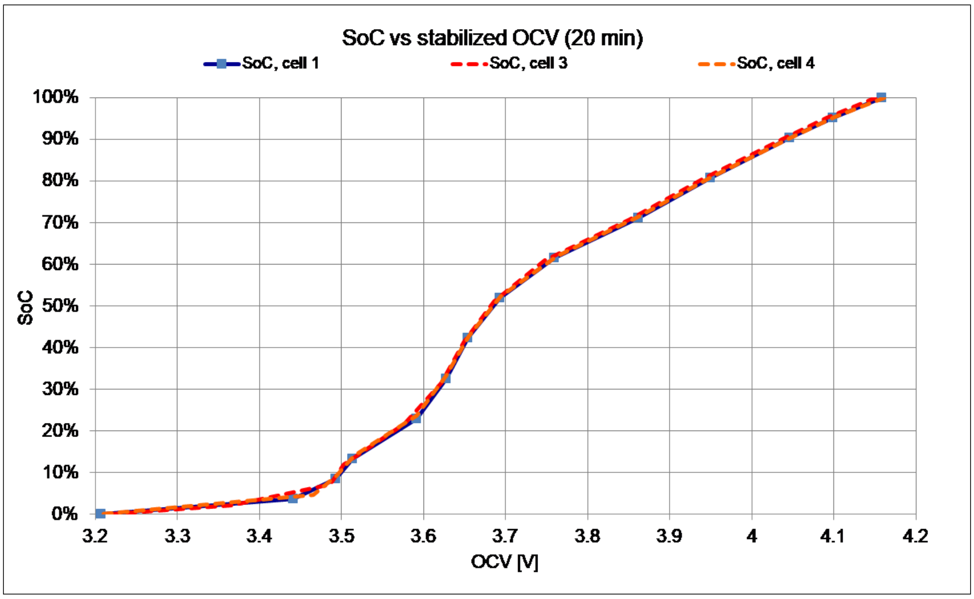


Figure 8.4: SoC-OCV calibration curve. The light blue squares are measurements, lines are interpolated values. The data represented here is what the BMS need for input for the SOC-OCV calibration.

8.7 Conclusion

Based on relatively simple cell cycling experiments it is possible to measure the OCV-SOC calibration curves needed for Lithium Balance SOC-calibration. It should however be noted that these measurements are not trivial and that special attention should be given to:

- Assuring sufficient relaxation time (at 0 current) to allow the OCV to stabilize fully. This may take more than 20 minutes and will have to be measured for each cell type.
- Having a constant temperature during the measurements. It should also be noted, that relaxation time typically will be a function of temperature

8.7.1 Future work

The accuracy of the SoC-OCV relationship depends on a wide range of parameters such as temperature, C-rate, battery chemistry, relaxation time, charge/discharge sequence, voltage limits, etc. as discussed for example in [1]. These needs to be understood for the relevant cells in order to evaluate the expected SOC accuracy for any specific cells.

The characterization methods used here for the SOC-OCV calibration can also be used to evaluate the internal resistance of the battery cells.

8.7.2 Reference documents

[1] Maitane Garmendia: "STATE-OF-CHARGE (SOC) ALGORITHM DESIGN METHODOLOGY FOR IMPLEMENTATION ON BATTERY MANAGEMENT SYSTEMS (BMS) OF INDUSTRIAL LI-ION BATTERY PACKS"

8.8 Example setup for CAN-based mode and staging requests

In this example all four CAN based requests will be configured and used. Furthermore, the feedback on when the system is waiting for a staging request will be read out. Note that reading out the state of inputs is not mandatory for using the feature, but it is useful when configuring the feature for the first time to verify correct setup.

Below is a reference setup to test the requesting of active mode, staging and balancing features.

8.8.1 CAN mode request configuration parameters

CAN request load ID Map ID = 50001 (ID Map CAN Rx Data 1 ID)
CAN request charge ID Map ID = 50002 (ID Map CAN Rx Data 2 ID)
CAN request staging ID Map ID = 50003 (ID Map CAN Rx Data 3 ID)
CAN request balancing ID Map ID=50004 (ID Map CAN Rx Data 4 ID)

8.8.2 CAN Rx configuration parameters

In this example one single CAN frame will be used to receive all four requests. Using four separate frames is also possible. It does however require setting up four separate CAN Rx frames with individual timeouts and IDs. The example will also read out only one bit per request. This is however not mandatory as multiple bits can be received and evaluated for an active request.

RX frame [1] Enable frame = 1
RX frame [1] Timeout interval = 200
RX frame [1] DLC = 8
RX frame [1] ID = 291 (0x123)
RX frame [1]is ID extended= 0 (11 bit identifier)
RX frame [1] CAN Channel = 0 (sCAN)
RX frame [1] Config [1] enable = 1
RX frame [1] Config [1] start bit = 0
RX frame [1] Config [1] length = 8
RX frame [1] Config [1] is little endian = 0
RX frame [1] Config [1] ID map ID = 50001 (ID Map CAN Rx Data 1 ID)
RX frame [1] Config [2] enable = 1
RX frame [1] Config [2] start bit = 8
RX frame [1] Config [2] length = 8
RX frame [1] Config [2] is little endian = 0
RX frame [1] Config [2] ID map ID = 50002 (ID Map CAN Rx Data 2 ID)
RX frame [1] Config [3] enable = 1

RX frame [1] Config [3] start bit = 16
RX frame [1] Config [3] length = 8
RX frame [1] Config [3] is little endian = 0
RX frame [1] Config [3] ID map ID = 50003 (ID Map CAN Rx Data 3 ID)
RX frame [1] Config [4] enable=1
RX frame [1] Config [4] start bit=24
RX frame [1] Config [4] length=8
RX frame [1] Config [4] is little endian=0
RX frame [1] Config [4] ID map ID=50003 (ID Map CAN Rx Data 4 ID)

The setup will expect one frame at least every 200 [ms]:

ID: 0x123 DATA: 00 00 00 00 ww zz yy xx

Where 'xx' is the load request input, 'yy' is the charge request input and 'zz' is the staging request input and 'ww' is the balancing request input. 'xx', 'yy', 'zz' and 'ww' are hexadecimal 8 bit values

8.8.3 CAN Tx configuration parameters

TX frame [1] Enable frame = 1
TX frame [1] Update interval = 1 (x 100 ms)
TX frame [1] DLC = 8
TX frame [1] ID = 1 (0x001)
TX frame [1] is ID Extended = 0
TX frame [1] CAN Channel = 0 (sCAN)
TX frame [1] Config [1] enabled = 1
TX frame [1] Config [1] entry type = 1
TX frame [1] Config [1] start bit = 0
TX frame [1] Config [1] length = 8
TX frame [1] Config [1] is little endian = 0
TX frame [1] Config [1] data = 50001 (ID Map CAN Rx Data 1 ID)
TX frame [1] Config [2] enabled = 1
TX frame [1] Config [2] entry type = 1
TX frame [1] Config [2] start bit = 8
TX frame [1] Config [2] length = 8
TX frame [1] Config [2] is little endian = 0
TX frame [1] Config [2] data = 50002 (ID Map CAN Rx Data 2 ID)
TX frame [1] Config [3] enabled = 1
TX frame [1] Config [3] entry type = 1
TX frame [1] Config [3] start bit = 16
TX frame [1] Config [3] length = 8
TX frame [1] Config [3] is little endian = 0
TX frame [1] Config [3] data = 50003 (ID Map CAN Rx Data 3 ID)
TX frame [1] Config [4] enabled = 1
TX frame [1] Config [4] entry type = 1
TX frame [1] Config [4] start bit = 24
TX frame [1] Config [4] length = 8
TX frame [1] Config [4] is little endian = 0
TX frame [1] Config [4] data = 1076 (CAN ID Map FLAGS_WAITING_FOR_STAGING_REQ)

The setup will make the BMS broadcast one frame every 100 [ms]:

ID: 0x001 DATA: 00 00 00 00 DD CC BB AA

Where AA is the load request data field, BB is the charge request data field and CC is the staging request data field, and DD is the flag that indicates BMS is waiting for staging request.

8.8.4 Load mode cycle

8.8.4.1 Stage 1: No requests

ID: 0x123 Data: 00 00 00 00 00 00 00 ID: 0x001 Data: 00 00 00 00 00 00 00
No request is being requested and the BMS also indicates that its request inputs are empty, and it is not waiting for a staging request.

8.8.4.2 Stage 2: Load request, pre-charging in progress

ID: 0x123 Data: 00 00 00 00 00 00 01 ID: 0x001 Data: 00 00 00 00 00 00 01
A load request is being broadcasted, and the BMS indicates that it has received a load request.

8.8.4.3 Stage 3: Load request, pre-charging is done

ID: 0x123 Data: 00 00 00 00 00 00 01 ID: 0x001 Data: 00 00 00 01 00 00 01
The BMS is done with the pre-charging sequence and is actively waiting for a staging request. Note that it is only possible for the state machine to continue if the pre-charging timer times out or a staging request is indicated. If the user simply stops requesting a mode, the BMS will still be in a waiting position. I.e. the user will have to complete the staging request before cancelling the mode request for the quickest exit.

8.8.4.4 Stage 4: Load request and staging request, staging is complete

ID: 0x123 Data: 00 00 00 00 01 00 01 ID: 0x001 Data: 00 00 00 00 01 00 01
The staging request is given, and the BMS acknowledges the staging by removing the flag that was indicating that it was waiting for the staging request.

8.8.4.5 Stage 5: Load request, staging is complete

ID: 0x123 Data: 00 00 00 00 00 00 01 ID: 0x001 Data: 00 00 00 00 00 00 01
The user can now remove the staging request and simply keep the system in active load mode by sending only the load request.

8.8.4.6 Stage 6: No request

ID: 0x123 Data: 00 00 00 00 00 00 00 ID: 0x001 Data: 00 00 00 00 00 00 00
The user can exit the mode by not requesting any load. Note that the user still needs to send the CAN frame with the request data set to zero. If the user simply stops sending the CAN frame, the data fields used for testing the request is not changed. The BMS will indicate with an error that the frame has not been received within the time limit. The user needs to exit the mode via the error check list if mode

termination by drop out of CAN frame behavior is required.

8.8.4.7 Charge mode cycle

The same six stages apply for the charge mode request, with the charge mode request data BB and FF indicated instead.

8.9 Configuration parameters c-BMS

View	Name	Description	Unit	Accepted values
System Configuration	Configuration ID	Any value can be selected to be used by the user for configuration identification.	-	0 – 4294967295
System Configuration	Current source type	Select sensor to be used for current measurement. Only one sensor can be used at a time. 0 = none, 1 = HALL, 2 = Shunt, 3 = CAN-based sensor.	-	0-3
System Configuration	Shunt resistance	Resistance value of the shunt sensor. This value is usually indicated as a voltage output at an ampere level but can be converted to a resistance value.	uΩ	0.0 – 429496729.5
System Configuration	Shunt sensor offset	Offset value for the shunt sensor measurement. This value is added to the converted current measurement to correct for any offset.	mA	(-32768) – 32767
System Configuration	Hall sensor scaling low	The measured low sensor voltage will to converted to a current value using this factor.	mV/A	0 – 4294967295
System Configuration	Hall sensor scaling high	The measured low sensor voltage will to converted to a current value using this factor.	mV/A	0 – 4294967295
System Configuration	Hall sensor offset	Offset value for the Hall sensor measurement. This value is added to the converted current measurement to correct for any offset.	mA	(-32768) – 32767

System Configuration	Hall sensor low sat. pos.	The low channel upper voltage saturation point, where the firmware switches to the high channel (channel 2) hall sensor. Below this value, the low channel is used, and above, the high channel is used.	mV	0-6563.5
System Configuration	Hall sensor low sat. neg.	The low channel low voltage saturation point, where the firmware switches to the high channel (channel 2) hall sensor. Below this value, the high channel is used, and above, the low channel is used.	mV	0-6563.5
System Configuration	CAN current sensor ID Map ID	If current source type is selected to be CAN sensor, this specifies the ID map ID where the current measurement is located.	-	0-65535
System Configuration	Initial Capacity	The initial capacity is the reference point for the SoC estimation. It indicates to the BMS what the initial capacity of the connected cells was from new.	Ah	0.0-6553.5
System Configuration	Minimum SoC trim	A minimum value indicated by the trimmed SoC output can be setup. This ensures that a SoC value lower than this is never indicated. If the internal value is lower, the trimmed SoC will indicate the Minimum SoC trim value.	-	0.00-655.35
System Configuration	Maximum SoC trim	A maximum value indicated by the trimmed SoC output can be setup. This ensures that a SoC value higher than this is never indicated. If the internal value is higher, the trimmed SoC will indicate the Maximum SoC trim value.	-	0.00-655.35
System Configuration	Aux NTC mantissa A	The significant digits of the Aux A coefficient.	-	0-65535
System Configuration	Aux NTC exponent A	The exponent of the Aux A coefficient.	-	(-128)-127
System Configuration	Aux NTC mantissa B	The significant digits of the Aux B coefficient.	-	0-65535
System Configuration	Aux NTC exponent B	The exponent of the Aux B coefficient.	-	(-128)-127
System Configuration	Aux NTC mantissa D	The significant digits of the Aux D coefficient.	-	0-65535
System Configuration	Aux NTC exponent D	The exponent of the Aux D coefficient.	-	(-128)-127

System Configuration	CMUs NTC mantissa A	The significant digits of the CMUs A coefficient.	-	0-65535
System Configuration	CMUs NTC exponent A	The exponent of the Aux A coefficient.	-	(-128)-127
System Configuration	CMUs NTC mantissa B	The significant digits of the CMUs B coefficient.	-	0-65535
System Configuration	CMUs NTC exponent B	The exponent of the Aux B coefficient.	-	(-128)-127
System Configuration	CMUs NTC mantissa D	The significant digits of the CMUs D coefficient.	-	0-65535
System Configuration	CMUs NTC exponent D	The exponent of the Aux D coefficient.	-	(-128)-127
System Configuration	SOH SOC calibration window 1, lower limit	SoC level for the lower bound of the first window. Window is used to determine where SoH calibration is allowed.	%	0-655.35
System Configuration	SOH SOC calibration window 1, upper limit	SoC level for the upper bound of the first window. Window is used to determine where SoH calibration is allowed.	%	0-655.35
System Configuration	SOH SOC calibration window 1, weight	The weight to apply to the new SoH value, compared to the old one. A value of 1 means use only the newly calculated value. A value of 0 means do not use the new calculation at all.	-	0.00-1.00
System Configuration	SOH SOC calibration window 2, lower limit	SoC level for the lower bound of the second window. Window is used to determine where SoH calibration is allowed.	%	0-655.35
System Configuration	SOH SOC calibration window 2, upper limit	SoC level for the upper bound of the second window. Window is used to determine where SoH calibration is allowed.	%	0-655.35
System Configuration	SOH SOC calibration window 2, weight	This parameter is **not** applicable from the software version v2.5.7. The BMS considers the same value "SOH SOC calibrationwindow 1, weight" for the window-1 and window-2	-	0.00-1.00
Operational Limits	Min. cell voltage	The minimum allowed cell voltage. If a cell voltage is detected to be below this threshold, an error code is indicated (2000). This error will, depending on error settings, limit allowed operation from the BMS.	mV	0.0-4999.9

Operational Limits	Max. cell voltage	The maximum allowed cell voltage. If a cell voltage is detected to be above this threshold, an error code is indicated (2001). This error will, depending on error settings, limit allowed operation from the BMS.	mV	0.0-4999.9
Operational Limits	Min. cell temperature	The minimum cell temperature limit. When the lowest cell temperature gets below this setting is the error code 2004 indicated.	°C	-40.0 - +60.0
Operational Limits	Max. cell temperature	The maximum cell temperature limit. When the highest cell temperature gets above this setting is the error code 2005 indicated.	°C	-35.0 - +100.0
Operational Limits	Max. i2t	The Max. i2t is a digital implementation of a melting fuse. The value set is the limit for the calculated “i2t sum”. Low values mean only a small “i2t sum” will flag an error.	A2s	0 – 4294967295
Operational Limits	Max. contactor break curr.	Before the BMS opens the battery pack main contactors it will test if the current flowing at the given time, is higher than this threshold. If the current is higher it will wait for a maximum time according to the “Contactors off timeout” and then open the contactors.	mA	0.00 - 42949672.95
Operational Limits	Max. precharge end curr.	End criteria for the precharge sequence is if the current is below this limit within the “Precharge timeout” setting. When the current is below the threshold, the precharge sequence will end successfully. If the current does not drop below this threshold within the timeframe, the precharge sequence will fail.	mA	0.00 - 42949672.95
Operational Limits	Max. contactor retries	If a contactor sequence fails, this setting will limit the number of retries to protect the contactors from damage by excessive wear from contactor cycling.	-	0 - 255

Operational Limits	Contactors off timeout	Threshold for the time the BMS is allowed to wait for the system current to drop below the accepted level for contactor opening under load. This test is bypassed if an emergency off command is issued.	sec	0.0 - 6553.5
Operational Limits	Precharge timeout	Time the BMS will wait for the current to drop below the threshold “Max. precharge end curr.” before it will consider the precharge attempt as failed.	sec	0.0 - 6553.5
Operational Limits	Contactor retry timeout	Time the BMS will wait after a failed precharge attempt before trying again.	sec	0.0 - 6553.5
Operational Limits	Temp. sensors enabled	Bitmask of which temperature sensors are enabled. All sensors are enabled by the binary value 00111111 = 63 decimal. charge complete (ignoring current direction). Once Bit 1 is sensor 1, Bit 2 is sensor 2 ect.	-	0 - 63
Operational Limits	Temp. sensors allocated for cells	Allocate temperature channels to cell temperature measurements. Temperature sensors allocated to cell temperature monitoring will follow cell temperature limits and report alarms accordingly.	-	0 - 63
Operational Limits	Min. temp channel 1-6	The minimum allowed temperature on this temperature channel before the error is given.	deg C	(-128) - 127
Operational Limits	Max. temp channel 1-6	The maximum allowed temperature on this temperature channel before the error is given.	deg C	(-128) - 127
Operational Limits	Balancing lower limit	Below this cell voltage limit, no balancing will be any cell.	mV	0.0 - 4999.9
Operational Limits	Balancing deadband	Deadband added to average cell voltage. Above this limit balancing is performed.	mV	0.0 - 4999.9
Operational Limits	DCLI temperature N	Temperature for DCLI row N	deg C	(-128) - 127
Operational Limits	DCLI temperature N, SOC X %	The current limit for ingoing current at X % SOC and N temperature	A	0.0 – 6553.5
Operational Limits	DCLO temperature N	Temperature for DCLO row N	deg C	(-128) - 127
Operational Limits	DCLO temperature N, SOC X %	The current limit for outgoing current at X % SOC and N temperature	A	0.0 – 6553.5

Operational Limits	Auto off cell voltage timeout limit	Time period limit for lowest cell voltage. Lowest cell voltage must constantly be below the "Auto off cell voltage limit" before the Power hold GPIO output is deactivated	sec	0 - 65535
Operational Limits	Auto off charge current limit	Limit for minimum charge current. When below the configured percentage of the requested charge current for the associated timeout period will the Power hold GPIO output be deactivated. Charge mode required.	%	1 - 100
Operational Limits	Auto off charge timeout limit	Time period limit for charge current. Charge current must constantly be below the "Auto off charge current limit" before the Power hold GPIO output is deactivated	sec	0 - 65535
Operational Limits	Auto off fully charged	Time period limit for battery fully charged. Battery status must constantly be fully charged before the Power hold GPIO output is deactivated. Charge mode required.	sec	0 - 65535
Operational Limits	Auto off drain limit	Current out limit. When below the limit for the associated timeout period will the Power hold GPIO output be deactivated. Ready or Load mode required.	mA	0 – 65535
Operational Limits	Auto off drain timeout limit	Time period limit for current out. Current out must constantly be below the "Auto off drain limit" before the Power hold GPIO output is deactivated.	sec	0 - 65535
Charger	Charge complete deadband I (current)	Threshold around 0 A for flagging charge complete. Once all measured current is inside the threshold charge complete may be flagged (if voltage is also within threshold)	mA	0 - 65535
Charger	Charge complete deadband V (voltage)	Threshold around target cell voltage for flagging charge complete. Once all measured cell voltages are inside the threshold charge complete may be flagged (if current is also within threshold)	mV	0.0 – 6553.5

Charger	CAN Charge enabled	Selection to set CAN charger output on. 0 = Off, 1 = On.	-	0 - 1
Charger	CAN Charge Max. V	Maximum output voltage from the CAN charger.		0.0 – 6553.5
Charger	PWM Charge enabled	Selection to set PWM charge signal on. 0 = Off, 1 = On.	-	0 - 1
Charger	PWM signal inverted	Selection to invert PWM signal output. 0 = Off, 1 = On.	-	0 - 1
Charger	PWM Min. duty	Cap PWM output to never output a duty cycle below this threshold.	%	0-100
Charger	PWM Max. duty	Cap PWM output to never output a duty cycle above this threshold.	%	0-100
Charger	Maximum balancing PWM duty cycle	The Max allowed PWM duty cycle during balancing	%	0-100
Charger	PWM output I deadband	If the charge output current is within this deadband, with reference to the setpoint charge current, the PWM output will stop regulating and stay at the previous value.	A	0.0 – 6553.5
Charger	Cell voltage target	The target voltage for all cells when charging. The BMS will use this value asthe end condition for a charge sequence and the calibration point for the capacity sum used for SOC estimation.	mV	0.0 – 4999.9
Charger	Allowed charge current deadband	The maximum allowed deviation between requested charge currentand measured current. Avalue of X means that allowed deviation deadband is -X to +X.	A	0.0 – 6553.5
Charger	Max. Charge current	This threshold is the maximum current set pointfor the charge current into the battery pack during normal charging.	A	0.0 – 6553.5
Charger	PID constant Kp	P coefficient for PID controller	-	(-2000.001)-2000.001
Charger	PID constant Ki	I coefficient for PID controller	-	(-2000.001)-2000.001
Charger	PID constant Kd	D coefficient for PID controller	-	(-2000.001)-2000.001
GPIO mapping	Activate balancing	Input of where to map “Activate balancing” functionality. When the Input is activated the BMS will activate balancing. 0 = Disabled. 0-8 indicates IO channel.	IO	0-8

GPIO mapping	Activate sleep	Input of where to map "Activate sleep" functionality. When the Input is activated the BMS will activate sleep mode. 0 = Disabled. 0-8 indicates IO channel.	IO	0-8
GPIO mapping	Load positive	Output to drive Load positive contactor. 0 = Disabled. 5-8 indicates IO channel.	IO	0-8
GPIO mapping	Charge negative	Output to drive Charge Negative contactor. 0 = Disabled. 5-8 indicates IO channel.	IO	0-8
GPIO mapping	Precharge	Output to drive Precharge contactor. 0 = Disabled. 5-8 indicates Input / Output channel.	IO	0-8
GPIO mapping	Load negative	Output to drive Load negative contactor. 0 = Disabled. 5-8 indicates IO channel.	IO	0-8
GPIO mapping	Load positive feedback	Feedback for the charge negative contactor. If enabled, the BMS expects this feedback signal to follow the state of "Charge negative". If there is a mismatch, an error is given and the contactor sequence will fail. 0 = Disabled. 0-8 indicates IO channel.	IO	0-8
GPIO mapping	Charge negative feedback	Feedback for the charge negative contactor. If enabled, the BMS expects this feedback signal to follow the state of "Charge negative". If there is a mismatch, an error is given and the contactor sequence will fail. 0 = Disabled. 0-8 indicates IO channel.	IO	0-8
GPIO mapping	Precharge feedback	Feedback for the precharge contactor. If enabled, the BMS expects this feedback signal to follow the state of "Precharge". If there is a mismatch, an error is given and the contactor sequence will fail. 0 = Disabled. 0-8 indicates IO channel.	IO	0-8

GPIO mapping	Request combined active	This function is not in use in the current version.	IO	0-8
GPIO mapping	Request charge active	Input of where to map "Request charge active" functionality. When the Input is activated the BMS will activate contactor charge mode. 0 = Disabled. 0-8 indicates IO channel.	IO	0-8
GPIO mapping	Request load active	Input of where to map "Request load active" functionality. When the Input is activated the BMS will activate contactor load mode. 0 = Disabled. 0-8 indicates IO channel.	IO	0-8
GPIO mapping	Power hold	Output for the Auto off function.	IO	0 – 16
GPIO mapping	Feedback settling time	Extra feedback settling time.	50msec	0, 50, 100, 150, 200…
Custom Data Processing	Slot [N] Operator	The operation to perform. 0 = [NOP] NONE 1 = [+] ADDITION 2 = [-] SUBTRACTION 3 = [*] MULTIPLICATION 4 = [/] DIVISION 5 = [&] BITWISE_AND 6 = [|] BITWISE_OR 7 = [^] BITWISE_XOR 8 = [>] GREATER_THAN 9 = [<] SMALLER_THAN 10 = [==] EQUAL_TO 11 = [>=] GREATER_THAN_OR_EQUAL 12 = [<=] SMALLER_THAN_OR_EQUAL 13 = [<<] LEFT_SHIFT 14 = [>>] RIGHT_SHIFT 15 = [!] INVERSE	-	0-15
Custom Data Processing	Slot [N] Right side value	The value to use on the right hand side of the operator (either constant or data ID).	-	-2147483648 - 2147483647
Custom Data Processing	Slot [N] Right side value type	The data type. 0 means a constant value. 1 means data ID.	-	0-1
Custom Data Processing	Slot [N] Left side value	The value to use on the left hand side of the operator (either constant or data ID).	-	-2147483648 - 2147483647

Custom Data Procesing	Slot [N] Left sidevalue type	The data type. 0 means a constant value. 1 means data ID.	-	0-1
CAN settings	CAN speed S-CAN	Input selection for the CAN speed of the application firmware. 0 = 125 kbps, 1 = 250 kbps, 2 = 500 kbps, 3 = 1000 kbps.	-	0-3
CAN settings	CAN ID Frame Charger	ID of the fixed CAN charge frame output. Input is in decimal value.	ID	0 - 2047
CAN settings	CAN ID Charger Extended?	Selector to output CAN ID Frame Charger as extended ID. 1 = 29 bit or 0 = 11 bit.	BOOL	0 or 1
CAN settings	CAN Channel FrameCharger	Selector for what CAN bus to output CAN Charger Frame. 0 = S-CAN or 1 = I-CAN.	-	0 - 1
CAN settings	CAN Charger type	Selection of predefined legacy CAN charger types: 0 = Zeroed output. 1 =EA-PS8200-70 from Elektro-Automatik. 2 static charge allowed flag. 3 = static charge allowed flag,with latching flag. 4 = Powerfinn charger	-	0 - 4
CAN settings	CAN ID start errorframes	Start address for a sequential map of all relevant error codes. Selectan ID where the following 52 ID's are vacant.	ID	0-2047
CAN settings	CAN ID error extended?	Selector to output CAN ID error frames as extended ID. 1 = 29 bit or 0 = 11 bit.	BOOL	0 or 1
CAN settings	CAN channel error frames	The CAN channel. 0: S-CAN	-	0
CAN settings	CAN request balancing ID map ID	If a CAN rx is setup to receive a balancing moderequest, this specifies the ID map ID (in CAN rx range)where the request is located.	-	0-65535
CAN settings	CAN request balancing PWM duty cycle ID map ID	If a CAN rx is setup to receive a balancing PWM duty cycle request, this specifies the ID map ID (inCAN rx range) where the request/data is located.	-	0-65535
CAN settings	CMU1 - CAN request balancing binary OR cell ID Map ID	The CAN frame ID where the forced bleeding bitmask for CMU1 will be received. If it is set to 0, no forced bleeding will be done.	-	0-65535
CAN settings	CMU2 - CAN request balancing binary OR cell ID Map ID	The CAN frame ID where the forced bleeding bitmask for CMU2 will be received. If it is set to 0, no forced bleeding will be done.	-	0-65535

CAN settings	CAN request load IDmap ID	If a CAN rx is setup to receive a load mode request, this specifies the ID map ID (in CAN rx range)where the request is located.	-	0-65535
CAN settings	CAN request chargeID map ID	If a CAN rx is setup to receive a charge mode request, this specifies the ID map ID (in CAN rx range)where the request is located.	-	0-65535
CAN settings	CAN request stagingID map ID	If a CAN rx is setup to receive a staging request, this specifies the ID map ID (in CAN rx range) where the request is located.	-	0-65535
CAN settings	TX Frame [N] Enable frame	The CAN frame is enabled if set to 1. If set to 0 it is not enabled.	BOOL	0 or 1
CAN settings	TX Frame [N] Update interval	CAN frames can be updated at a maximum rate of 100ms. 1: the CAN frame is updated at the maximum rate 2: the CAN frame is updated at half the maximum rate, i.e every 200 ms 3: The 3 CAN frameis updated at one third themaximum frequency, i.e every 300 ms …and so on.	-	1 – 65535
CAN settings	TX Frame [N] DLC	The length of the CAN frame in bytes.	-	0 - 8
CAN settings	TX Frame [N] ID	The CAN frame id as adecimal value.	-	0 to 536870912 [2^(29) 29 bit ID] 0 to 2048 [2^(11) 11 bit ID]
CAN settings	TX Frame [N] Is ID Extended	1: the frame_id is a 29-bit extended CAN frame ID. 0:the frame_id is an 11-bit CAN frame ID.		0 and 1

CAN settings	TX Frame [N] CAN Channel	The CAN channel. 0: S-CAN		0
CAN settings	TX Frame [N] Config 1(-10) enabled	The configuration entry number 'X' (for the CAN frame) is enabled if set to 1.		0 and 1
CAN settings	TX Frame [N] Config 1(-10) entry type	The type of the entry data. 0 means constant, 1 means BMS variable.		0 or 1
CAN settings	TX Frame [N] Config 1(-10) start bit	The bit in the CAN frame, where the first data bit is to be placed		0 to 63
CAN settings	TX Frame [N] Config 1(-10) length	The number of bits to write data to, starting from start bit		0 to 32
CAN settings	TX Frame [N] Config1(-10) is little endian	If set to 1, bits are treated as little endian (Intel format). If set to 0, bits are treated as big endian (Motorola format).		0 or 1
CAN settings	TX Frame [N] Config 1(-10) data	Either the constant data or an ID to the BMS variable to place in the CAN frame. IDd numbers are found in Appendix: Data ID Map		0 to 4294967295 (2^(32) – 1)
CAN settings	RX Frame [N] Enableframe	The CAN frame is enabled if set to 1. If set to 0 it is not enabled.	BOOL	0 or 1
CAN settings	RX Frame [N] Timeout interval	Optional timeout causing a BMS error whenever more than Timeout intervals of 100 ms occurs between receptions of the CAN frame. Counting starts as soon as the BMS is fully up and running. 0 disables this feature 1: the CAN frame is expected within 100 ms 2: The CAN frame is expected within 200 ms 3: The CAN frame is expected within 300 ms ... and so on.	-	1 – 65535

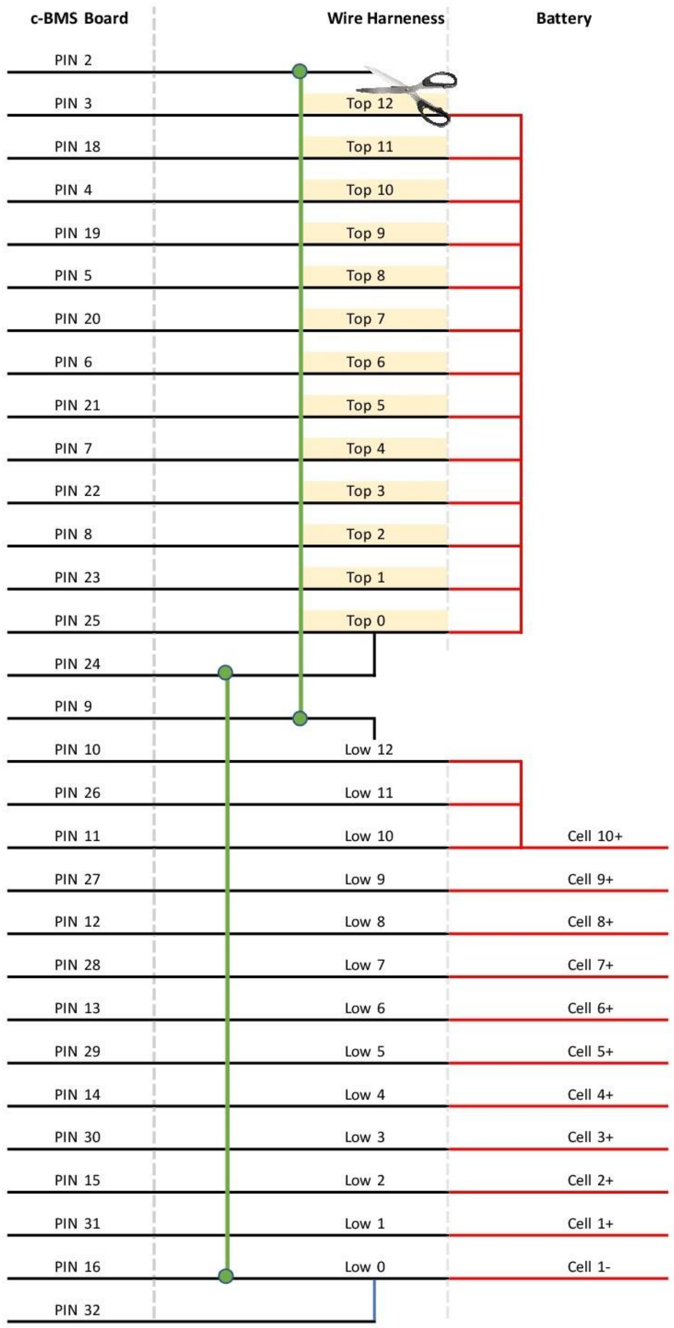
CAN settings	RX Frame [N] DLC	The length of the CAN frame in bytes.	-	0 - 8
CAN settings	RX Frame [N] ID	The CAN frame id as a decimal value.	-	0 to 536870912 [2^(29) 29 bit ID] 0 to 2048 [2^(11) 11 bit ID]
CAN settings	RX Frame [N] Is ID Extended	1: the frame_id is a 29-bit extended CAN frame ID. 0: the frame_id is an 11-bit CAN frame ID.		0 and 1
CAN settings	RX Frame [N] CAN Channel	The CAN channel. 0: S-CAN		0
CAN settings	RX Frame [N] Config 1(-10) enabled	Enabling of CAN Rx frame.0=Rx frame disabled, 1=Rx frame enabled without Rx timeout error, 2=Rx frame enabled including Rx timeout error function		0 to 2
CAN settings	RX Frame [N] Config 1(-10) start bit	The bit in the CAN frame, where the first data bit is to be found		0 to 63
CAN settings	RX Frame [N] Config 1(-10) length	The amount of bits to read, starting from start_bit		0 to 32
CAN settings	RX Frame [N] Config 1(-10) is little endian	If set to 1, bits are treated as little endian (Intel format). If set to 0, bits are treated as big endian (Motorola format).		0 or 1
CAN settings	RX Frame [N] Config 1(-10) ID map ID	ID to store the received data in.		50001-50026
CMU config	Enabled NTC inputs CMU 1-2	Bitmask for enabled NTC inputs on CMU. NTC 1 is bit 1, NTC 2 is bit 2 etc.	Bits	0 - 4095
CMU config	Enabled cells CMU1-2	Bitmask for enabled cells on CMU. Cell 1 is bit 1 (LSB),cell 2 is bit 2 etc.. Input to BMS creator is in decimal.	Bits	0 - 4095
CMU config	Expected CMUs	Setup for number of front end measuring devices.	No.	2
Error Behavior	Default error type	Currently this field must be set to 0	-	0
Error Behavior	Default clear time	This is the minimum time the error will be reported before it is cleared.	s	0.0-100.0

Error Behavior	Code error 1	Contains the error code for an error that required special handling	-	0-65535
Error Behavior	Type	Type of the error: 0 : Normal. The error will be reported for (at least) Clear time 1: Auto Clear. The error will auto-clear after Clear time 2: Reserved. Do not use 3: Reserved. Do not use 4: One-shot. The error will only be reported once between reboots 5: One-shot auto clear. The error will be reported once between reboots and will auto clear after Clear time	-	0-5
Error Behavior	Clear time	Used according to the Type field.	s	0.0-100.0
Error Behavior	Filter type	0: None. No filter is applied to the error 1: Counter. The error is reported after Filter count occurrences. If the error is cleared, the counter is reset. 2: Timer. The error is reported Filter time seconds after it is raised. If the error is cleared the timer is reset.	-	-
Error Behavior	Filter count	Specifies a count to be used with the filter. See Filter type	-	0-255
Error Behavior	Filter time	Specifies a timer to be used with a filter. See Filter type	s	0.0-100.0
Errors - enter ready Errors - enter charge Errors - enter load Errors - stay charge Errors - stay load Errors - request current	Code error 1-64	The specific error code associated with the error. See the full list of all error codes	-	1 – 65535
Errors - enter ready Errors - enter charge Errors - enter load Errors - stay charge Errors - stay load Errors - request current	Timer error 1-64	Delay time for the action	s	0.0-100.0
SOC-OCV settings	Data set [N] voltage [X]	The data point for SoC value X (X % SoC) for data set N	mV	0-5500
SOC-OCV settings	Data set [N] temperature	The temperature at which the data points for data set N are valid	deg C	(-40)-120

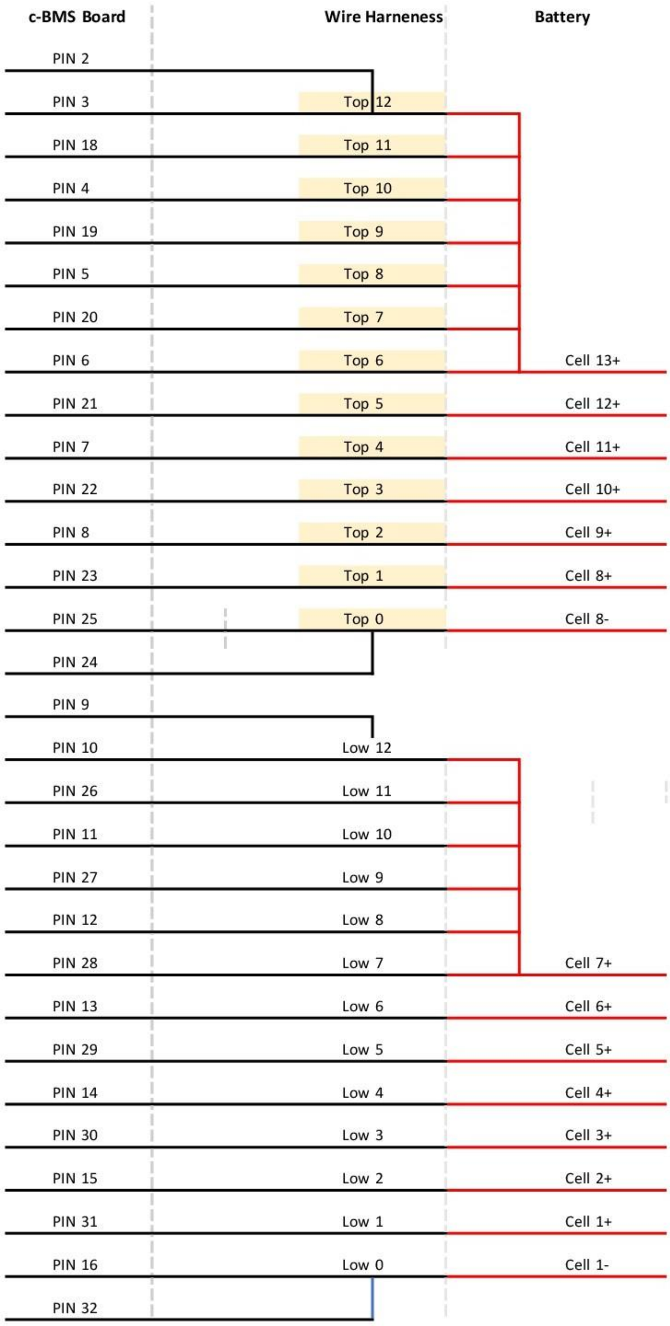
SOC-OCV settings	Maximum off duration for calibration	The maximum off time of the BMS, before skipping SOC-OCV calibration. This is to prevent cell discharge from affecting the calculations.	min	0-65535
SOC-OCV settings	Maximum quality	The maximum weighting to use. A value of 100 % means the OCV-SoC value completely replaces the counted SoC after "Off duration for highest quality" minutes. A value of 90 means the OCV-SoC value is weighted 90 %	%	0-100
SOC-OCV settings	Off duration forhighest quality	The minimum amount of time the BMS must be turned off before considering the OCV-SoC data to be of maximum quality	min	0-65535
SOC-OCV settings	Off duration forlowest quality	The amount of time the BMS must be turned off before considering the OCV-SoC calibrated values	min	0-65535
SOC-OCV settings	Enable data sets	How many data sets to enable for OCV. 0 Means feature is not used	-	0-5
SOP Settings	Temperature X, SoCY, Interval Z	The SOP for interval Z (1-4) when temperature equals the setting for Temperature X (1-7), and SoC equals the setting for SoC Y (1-11)		0-65535
SOP Settings	SoC Y	The SoC value for table column Y	%	0-100
SOP Settings	Temperature X	The temperature value for table row X	deg C	(-40)-120
SOP Settings	Number of SoCcolumns	The number of columns used in table. Used columns must be contiguous (in order)	-	0-11
SOP Settings	Number of temperature rows	The number of rows used in table. Used rows must be contiguous (in order)	-	0-7

8.10 Monitoring wires connection examples c-BMS

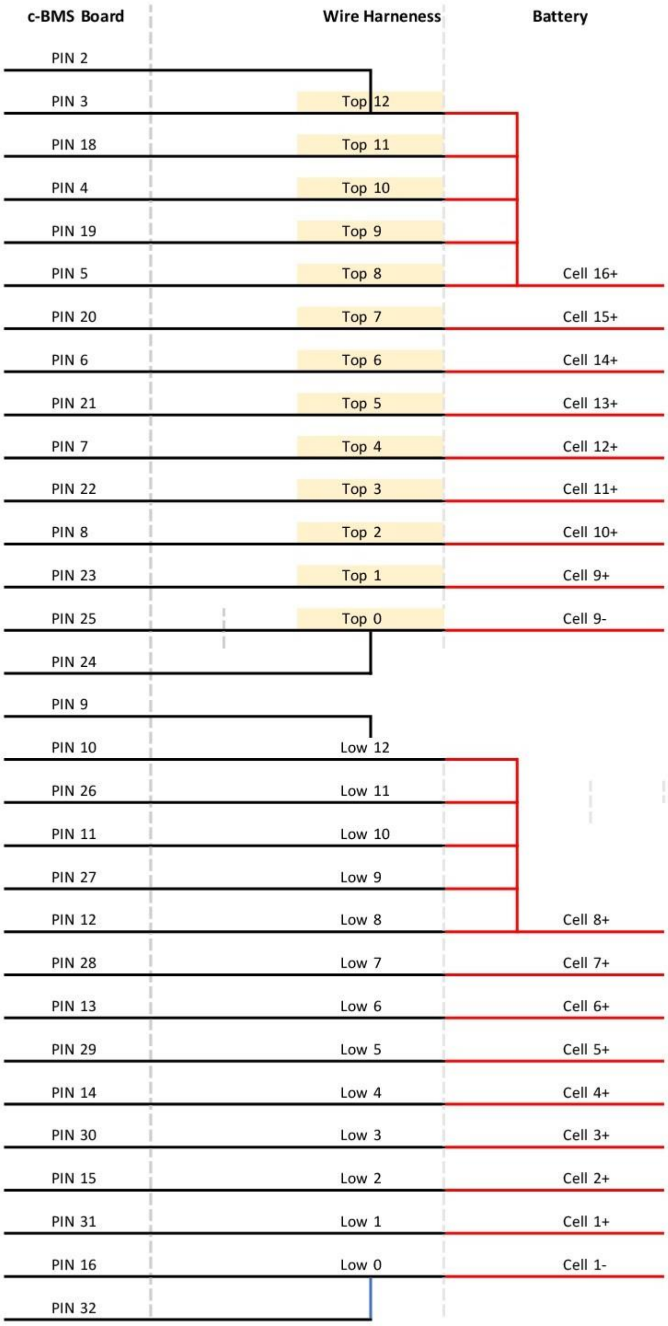
8.10.1 10 cells



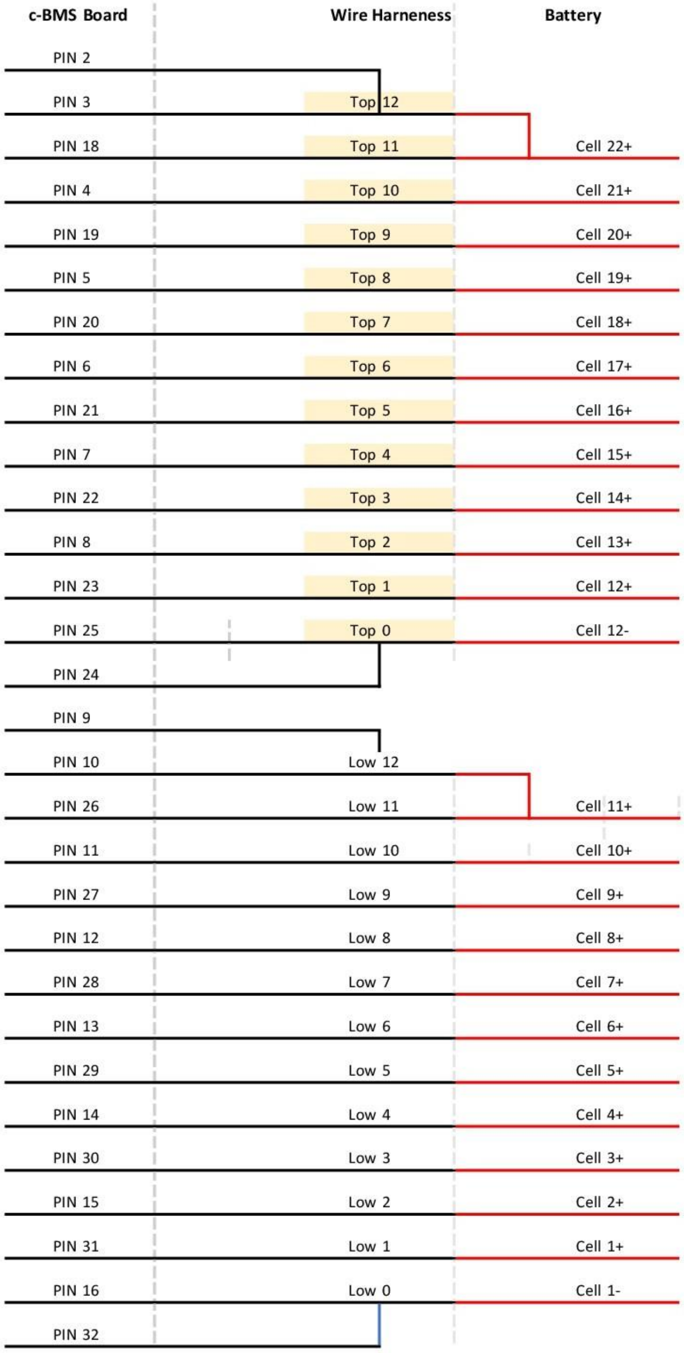
8.10.2 13 cells



8.10.3 16 cells



8.10.4 22 cells



8.11 Busmaster configuration

To read and if needed store the physical values measured by the BMS, a dedicated program must be used. One such program is the freeware program Busmaster. This is available at https://rbei-etas.github.io/busmaster/.

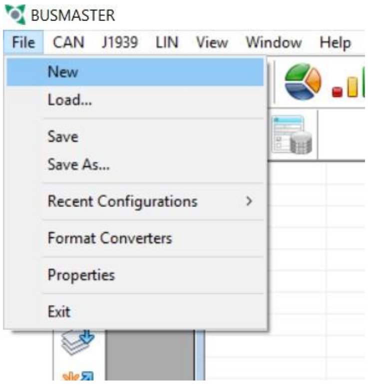
The Busmaster program operates on three levels

- Project Configuration and Database file
  - Message
    - Signals in message

The following sections describes the most important steps in setting up the Bus-master program to read data after Bus-master is installed on the PC connected to the BMS. This is exemplified by showing how to use Busmaster to read the 4 cell voltages configured on the CAN bus in section 5.9.2.

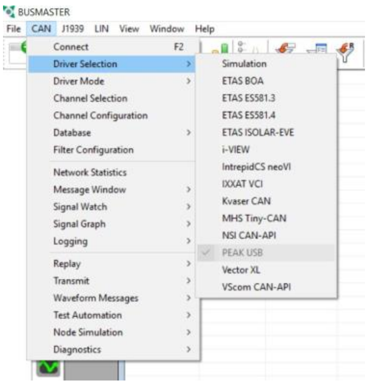
8.11.1 Create new Busmaster project

Step 1.1 Create and save file



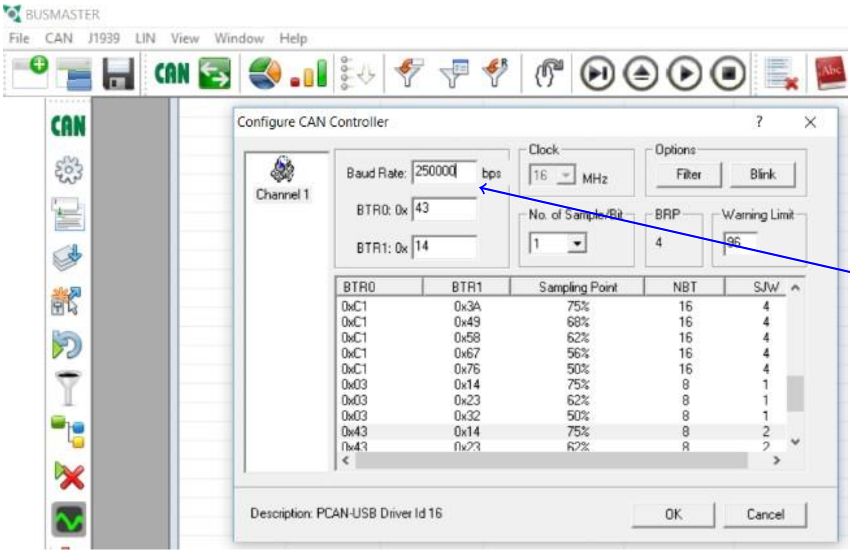
- Create a new project configuration file
- Write a project name and save the file

Step 1.2: Select the relevant adapter driver



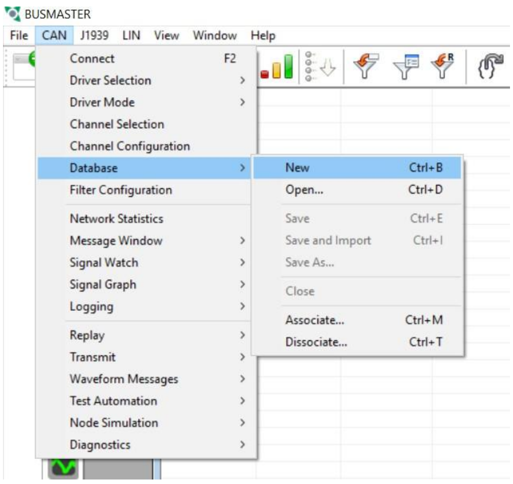
- In the CAN driver selection, select “PEAK USB” for CAN PEAK Adapter

Step 1.3: Select the correct CAN Baud rate



- Baud rate to be matched with value in BMS creator.
- Refer to BMS Creator configuration Setting in section 5.3.

8.11.2 Create a new Database

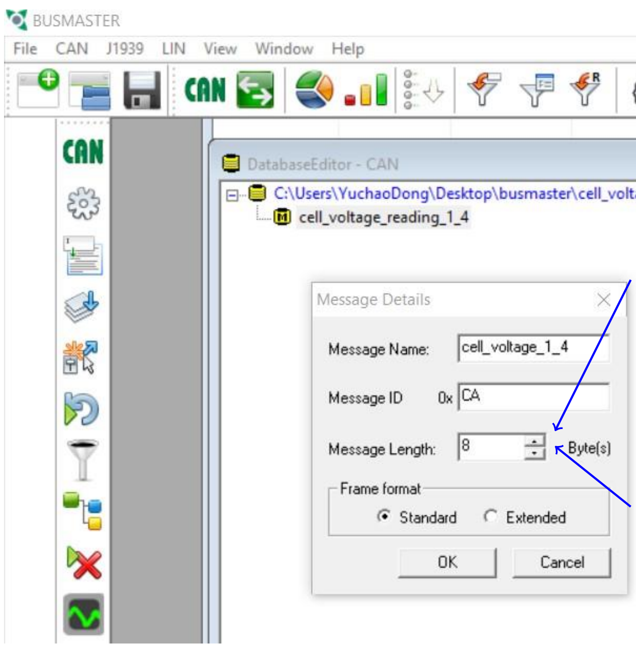


Create a new database file
Save the new file when prompted. In the example in this appendix it is the file "C:\User….\cell_voltage_1_4.DBF"

8.11.3 Create and configure a new "message"

The parameters on the CAN bus are stored in the Busmaster database in individual messages which can each contain one CAN frame corresponding to 8 bytes of information.

Step 2.1 create new message

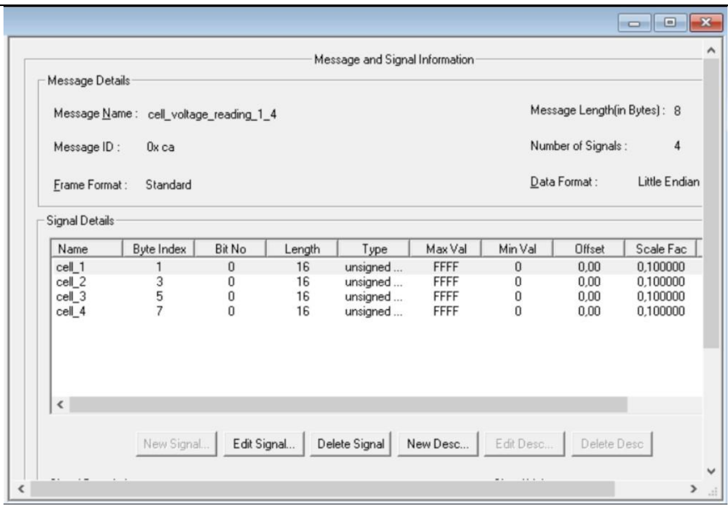
 C-creator CAN frame ID for cell voltage 1: Select message length (Standard is 8 bytes. This can be used for all parameters)	Right-click on database, here “C:User…” to create a new message under the database. Identify the frame to be stored in the message in the Message ID. **Note that Hex-numbers are to be used in the Busmaster for CAN-frame id[3].** Decimal numbers are used in the BMS Creator.

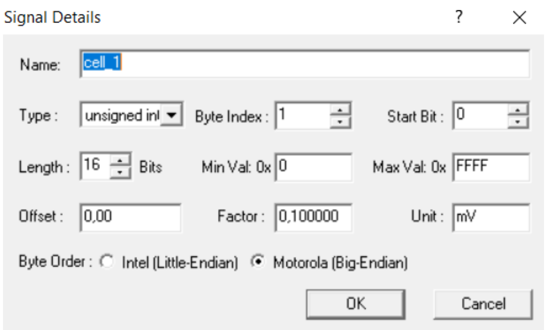
Step 2.2: Create signal contents for the message.

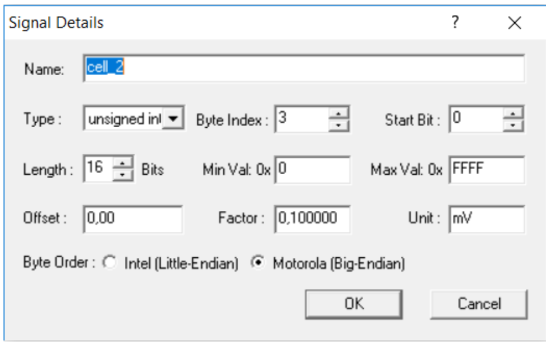
The examples below show how message contents are created for cell voltage 1 to 4 readings.
Please note that for Busmaster, bytes are identified by the byte index, corresponding to the Byte No shown below. the CAN frame for cell voltage 1 can be seen below:

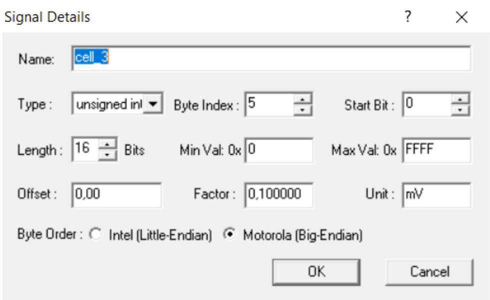
The total CAN frame for the voltages of 4 cells can be seen below:

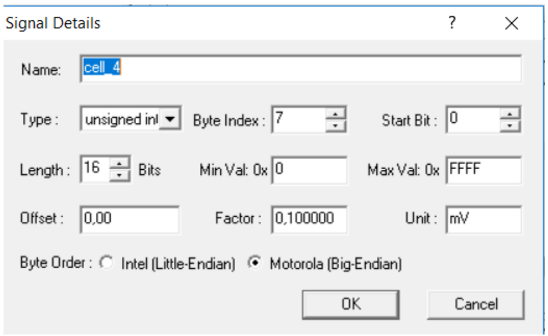
[3]Conversion between decimal and hex numbers can be done for example on http://www.binaryhexconverter.com/decimal-to-hex-converter











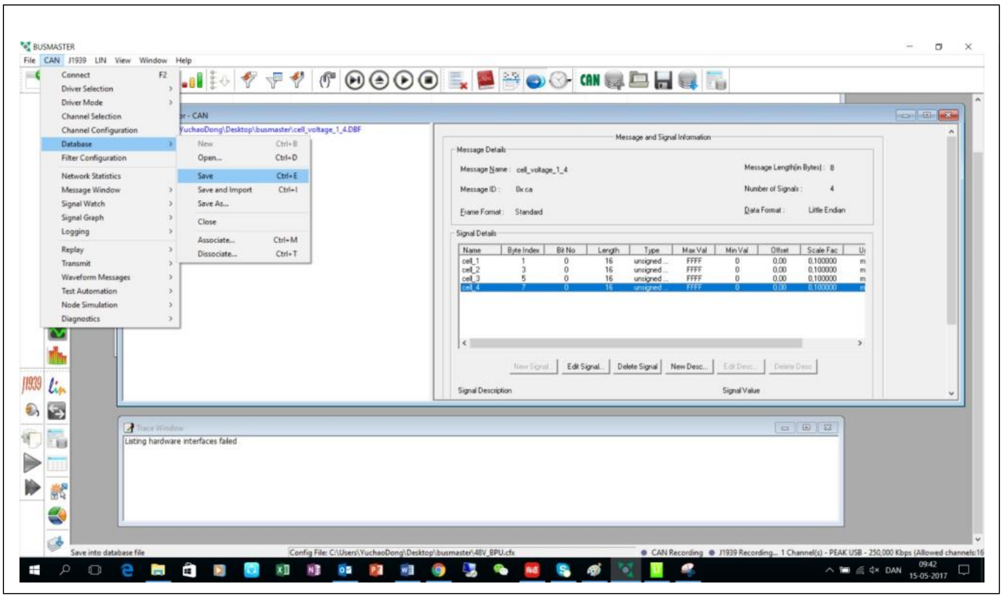
- Name: Select meaningful name to allow the user to recognise the value transmitted by the message
- Byte index: Placement in the frame, please refer to the figure below
- Bit No: Equals the start bit, which is typically 0, when all bits in a byte are used
- Length: Bit lengths as can be found in the "size" column of Appendix: Data ID Map.
- Type: Boolean, Signed integer or unsigned Integer, please refer to the "size" column of Appendix: Data ID Map.
- Max Val, will automatically be filled in
- Offset: to be set to 0
- Factor: please refer to the 'Scaling" column of Appendix: Data ID Map.

Byte No.	0								1							
Bits No.	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
Allocation	Cell voltage 1															

Byte No.	0			1			2			3			4			5			6			7		
Bits No.	63					48	47					32	31					16	15					0
Allocation	Cell voltage 1						Cell voltage 2						Cell voltage 3						Cell voltage 4					

Note that, the green boxes represent the Bits No used in setting up CAN messages in BMS creator and the blue boxes represents the byte index used in setting up the BusMaster.

Step 2.3 Save the new configuration

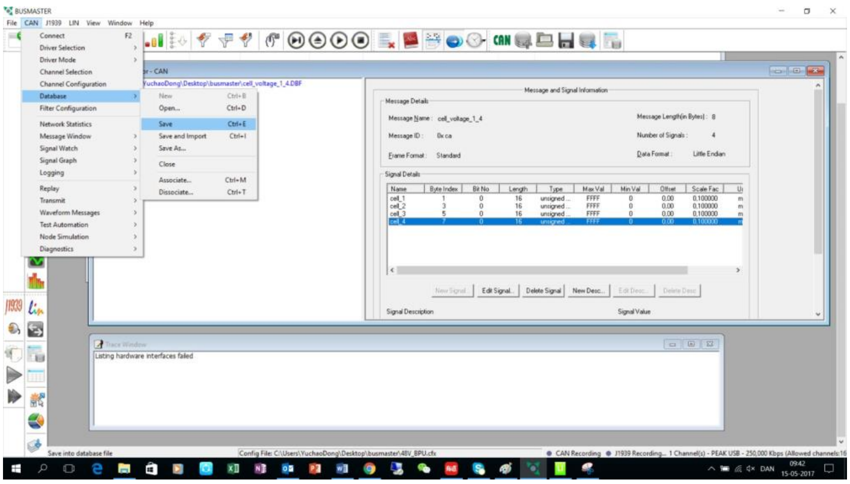


Save the database configuration (the DBF-file) including the set-up of the individual messages

8.11.4 Select Database and connect to BMS

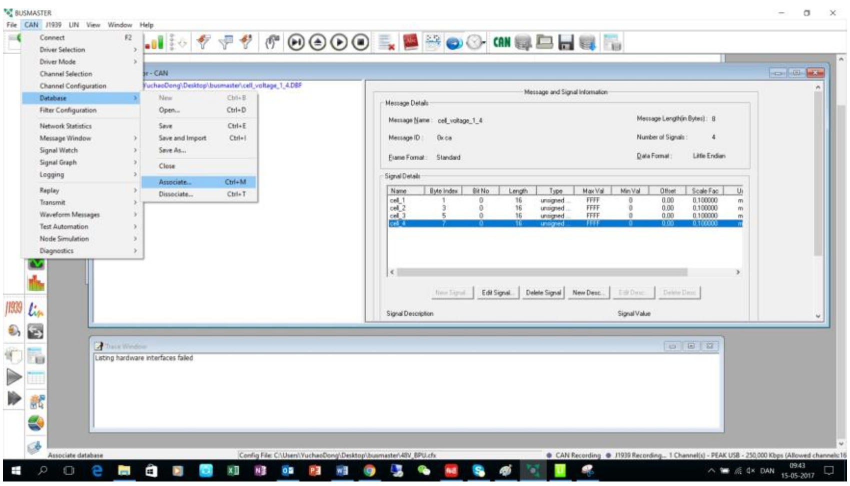
A number of different databases may be used by the Busmater program. To select the relevant configuration for the CAN bus to be read, please use the following steps.

Step 3.1 Associate the relevant database with the present CAN-bus.

	Associate with the database which contains the desired parameters which in this case is the cell voltage 1 to 4.

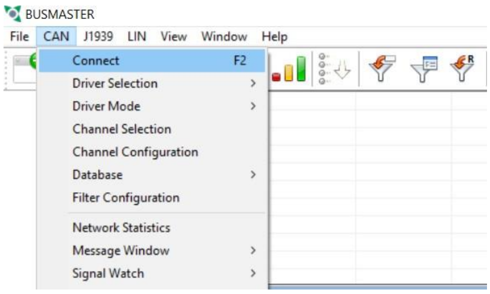
Step 3.2: Connect CAN

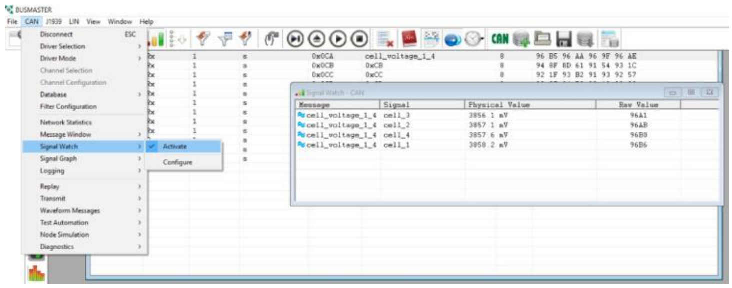
Now Busmaster is ready to read the data from the BMS via CAN and place the read values in the configured messages

	Click on “Connect” to CAN

8.11.5 Select Signals/Parameters to be Watched/Displayed

To have the actual values from the CAN-bus displayed on the Busmaster screen, please use the “Select Signal Watch” function

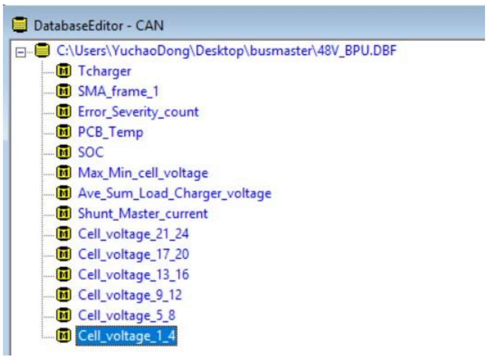




- Activate the Signal watch to read the physical value from the BMS.
- Right click on table to select the messages to be displayed.

8.11.6 Configure more values to be included in messages and displayed

To configure more values to be included in messages and displayed, please repeat



To configure more values to be included in messages and displayed, please repeat

- Section 8.11.3
- Section 8.11.5

8.12 HW functions which are not yet enabled by Software – c-BMS

8.12.1 High Voltage Interlock (HVIL)

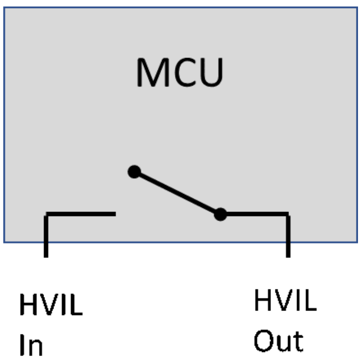


Figure 8.5: HVIL Connection

The MCU provides input and output ports for a High Voltage Interlock (HVIL) signal. This feature can be used by the BMS to communicate a critical error conditions directly to other electronic units where the battery system is used, e.g. in a Vehicle.

Under normal conditions the MCU shorts the HVIL in to HVIL out. In case of emergency conditions which might require the shutdown of other electronic functions, the HVIL switch (relay) will open. This can then be immediately detected by other electronic systems connected to the HVIL system which can act accordingly.

8.12.2 Real Time Clock

A real time clock is available to assure that the BMS has a correct absolute time also when the system enters sleep mode or is powered off. The real time clock is powered by the Cell monitoring ASIC2, which implies that if the BMS has been disconnected from the main battery, the Real Time Clock will no longer show the correct time.

8.13 Converting a PWM signal to an analog control voltage

It is possible to create an analog voltage signal from a PWM signal by Low Pass Filtering the PWM signal. However, almost no current can be provided from the PWM signal provided by the BMS and often chargers needs to be controlled by voltage which is isolated from the BMS ground reference.

An example of a circuit which can be used to convert a PWM signal from the BMS to an analog control voltage is shown in Figure 8.6.

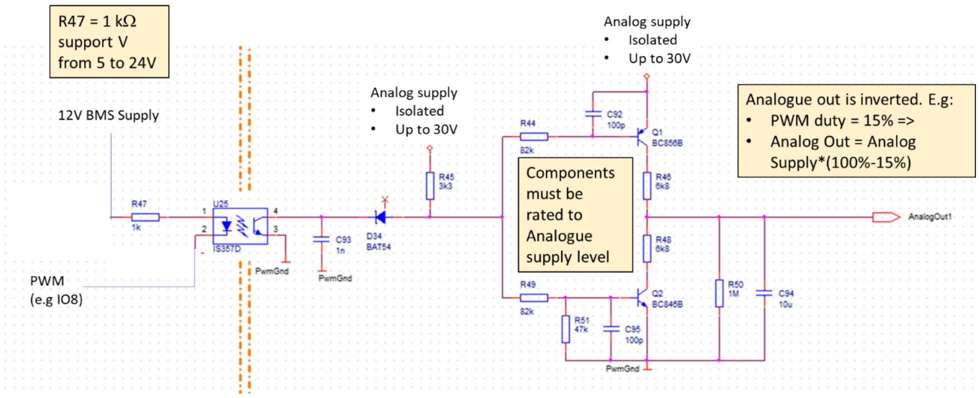


Figure 8.6: Example of a circuit which can be used to convert a PWM signal from the BMS to an analogue control voltage