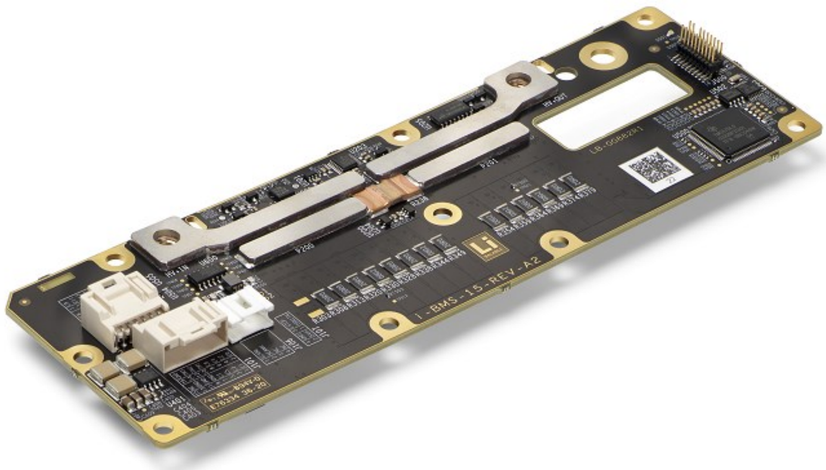
i-BMS15 User Manual

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



LiTHIUM BALANCE
BATTERY MANAGEMENT SYSTEMS

www.lithiumbalance.com LiTHIUM BALANCE A/S
contact@lithiumbalance.com
Tel: +45 5851 5104

1 Safety Information

1.1 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.

1.2 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 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.

- 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 and/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:
    - The load draws an excessive current from the battery which could lead to rapid overheating of the battery cells.
    - The battery is accidentally short-circuited.

1.4 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 of all safety features listed in the previous section 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.

2 Document Purpose

The purpose of this manual is to provide the user of the Lithium Balance i-BMS15 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.

3 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.

4 Copyrights & Patents 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

5 Abbreviations

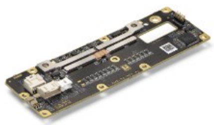
BPU	Battery Protection Unit
DoD	Depth of Discharge
BMS	Battery Management System
SoC	State-of-Charge
SoH	State-of-Health
VCU	Vehicle Control Unit
DLC	Data Length Code (of CAN Frame)
NTC	Negative Temperature Coefficient (thermistors)
GPIO	General Purpose Input Output
PSU	Power Supply Unit

6 Glossary

- **Battery Module :** A battery module consists of multiple cells connected in series/parallel. Multiple battery modules together make one battery pack.
- **Battery Protection Unit (BPU) :** It is a unit comprising the BMS, battery connect/disconnect functions and current measurement sensors.
- **Contactor :** It is a physical switch, controlled by the BMS, in the path of energy flow from a battery pack. The BMS opens or closes these contactors based on the operating mode of the battery system. However, in the case of i-BMS15, being a single-board & compact solution, this functionality is performed by a set of MOSFETs which are present on-board. The contactor term will be used throughout the manual for the user to understand and relate to the concept of switching.
- **Controller Area Network (CAN)** : It is a serial communication bus standard designed to allow microcontrollers/ECUs to communicate with each other's applications without a host computer.
- **Hot Swap :** Hot Swap technology enables connecting/disconnecting/replacing the battery modules while the battery pack is in operation in such a way that there is no interruption in power supply and the overall system. It is important to understand that the hot swap can be performed by the end-users themselves i.e. no professional service personnel is required to swap the battery.
- **Proportional Integral Derivative (PID) :** It is a feedback-based control mechanism used to regulate process variables like speed, pressure or charge current (in i-BMS15 context).

7 Document Structure

Introduction : LiBAL BMS Overview & Specifications



Hardware Guide : Connecting BMS to the battery pack & other auxiliaries

Functionality : Sensing, Algorithms, Techniques & Controls

Software Guide : Boot Loading, Configuring & Troubleshooting the BMS

Communication : Setting up CAN Bus

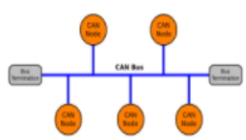


Table of Contents

8 Introduction

8.1 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 form factors. The system addresses three critical aspects of battery management:

- **Safety** - The BMS monitors 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 (for diagnostics, configurations, updates and information exchange) and controls safety related system elements like switches and chargers (for battery connect/disconnect.

8.2 i-BMS15 : Product Rundown & Specifications

The i-BMS15 is a compact Battery Monitoring System (BMS) with monitoring and management functions integrated into a single board. Engineered to monitor upto 15 cells in series and cover applications upto 60 V, the iBMS15 can also be individually connected in parallel architecture for upto 6 battery packs. On-board current measurement, pre-charge circuit, dc-dc supply and battery connect/disconnect mosfets makes this BMS ideal for application in electric 2Ws/3Ws, industrial robots or ESS.

8.2.1 System Block Diagram :

Figure 1 below depicts the block diagram of a typical i-BMS15 system. The important points to note are :

- The i-BMS15 printed circuit board is connected to auxiliary equipment with the help of dedicated cables.
- The BMS communicates with other equipment/ECUs/Users over CAN. In order to interface personal computers (PCs) with the i-BMS15, a PEAK CAN adaptor is required.
- A plethora of i-BMS15 parameters can be configured in the Lithium Balance BMS Creator Configuration Software (Windows PC Only).
- Operating parameters (e.g., current and voltage) can be read from the BMS via a standard CAN bus interface software (PCAN View is recommended).

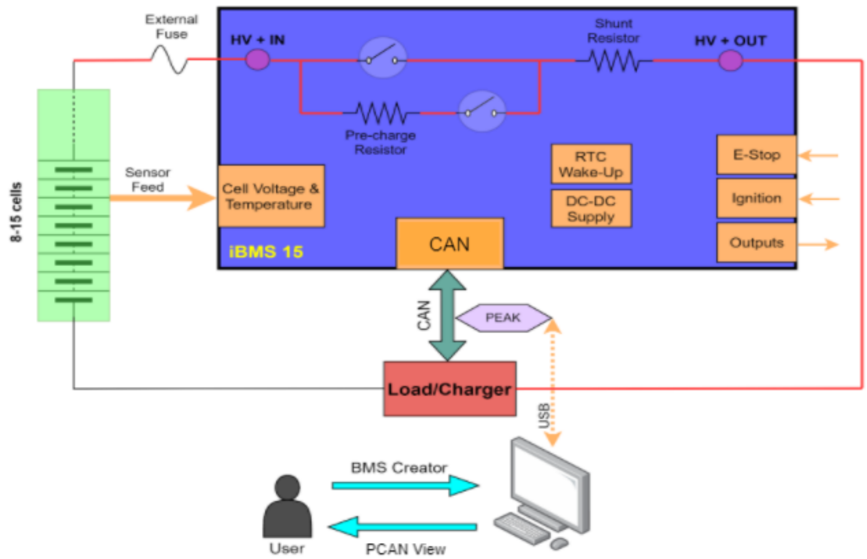


Figure 1 : i-BMS15 Block Diagram

8.2.2 i-BMS15 Key Specifications

Table 1 summarises the key specifications of i-BMS15. It is recommended to refer the i-BMS15 datasheet in order to get an overview of the features, specifications, functionalities and applications of i-BMS15.

Battery Voltage	25-60 Vdc
Number of Voltage Channels	8-15
Current Measurement	On-Board Shunt
Temperature Measurement	NTC
Number of Temperature Channels	4 (on-board) , 6 (external)
Cell Voltage, Current & Temperature Sampling Frequency	100 ms
Power Consumption (Active Mode)	< 75 mW @ 48 V
Power Consumption (Sleep Mode)	< 0.5 mW @ 48 V
Balancing Strategy	Passive
Cell Balancing Current	200 mA @ 4.2 V
Communication	CAN 2.0 A/B
Charger Control Interface/ Diagnostics Interface	CAN / CAN UDS

“Hot Swap” Functionality / Parallel Pack	Yes, Upto 6 Packs
Dimensions (L x W x H)	200 x 65 x 18 mm
Weight	93 g

Table 1 : i-BMS15 Specifications

Important Note : The allowed battery voltage range supported by i-BMS15 is 25-60 Vdc. In case the upper or lower limit for battery voltage is exceeded for some applications, it is recommended to consult LiBAL.

9 i-BMS15 Hardware

The purpose of this chapter is to give a general overview to the i-BMS15 board and also provide the user with an easy to understand guide for connections between the battery pack and connectors on the i-BMS15 board.

9.1 Hardware Overview : Know Your BMS

The i-BMS15 is designed to be extremely compact and can be easily interfaced with the battery pack as well as other equipment. In order to achieve this, a basic know-how of the hardware accessories present on the BMS becomes pivotal. Figure 2 presents an overview of the i-BMS15 printed circuit board :

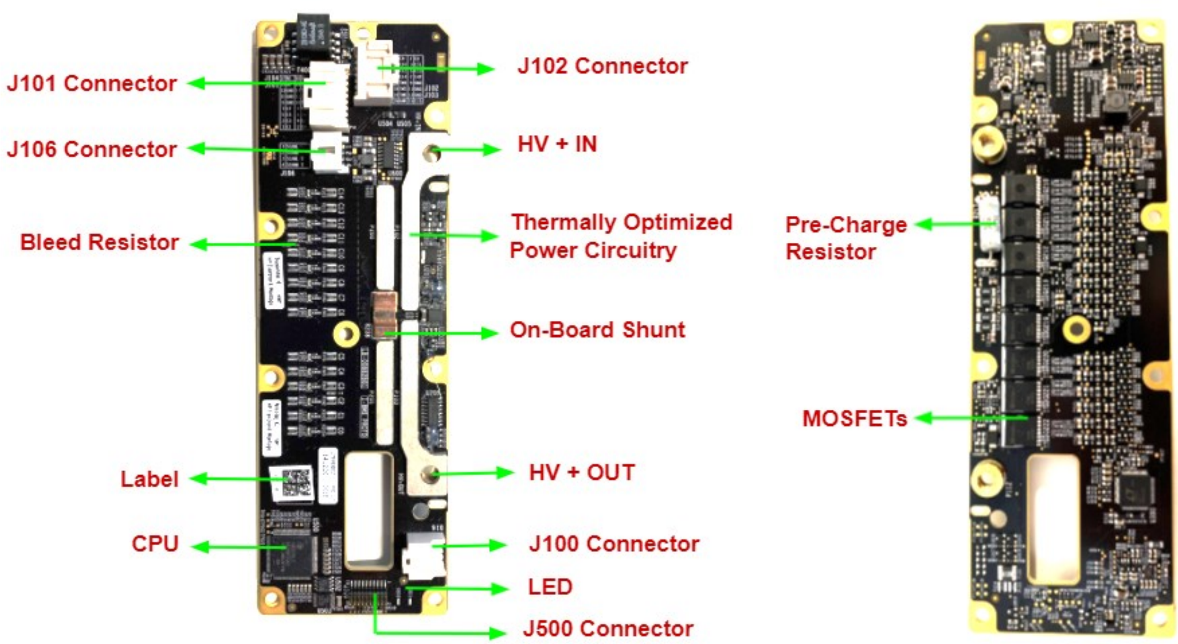


Figure 2 : i-BMS15 Hardware Overview; Front (L) , Rear (R)

9.2 LEDs

The i-BMS15 board comprises of 2 light emitting diodes (LEDs) :

- The red LED indicates 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.
- The orange LED indicates operating modes of the i-BMS15 :
  - LED Flashing 3 times implies Main Firmware CRC Error
  - LED Flashing 4 times implies Configuration Data CRC Error

9.3 Label

The QR label 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 using dedicated QR readers. In case smart phones are used the "QR Reader from TapMedia Ltd" has been tested successfully.

9.4 Fitment Guidelines

This section deals with guidelines for i-BMS15 mounting and connection on the battery pack in order to avoid resistance/heat build up in the power line. **It is extremely important that the user follows the below mentioned guidelines for fitment of the i-BMS15 on the battery pack.**

9.4.1 BMS Mounting Guidelines

The sequence for assembling/integrating the battery pack with i-BMS15 is as following :

1. Prepare the Harness for Battery Pack
2. Connect pack harness to cells based on requirement
3. Mount the BMS on battery Pack
4. Refer to Mounting Guidelines in Section 9.4.2
5. Connect the LV connectors i.e. J100, J101, J102 and J106 to the BMS
6. Connect of HV+OUT
7. Connect HV+IN
8. Refer HV connection guidelines in Section 9.4.3

9.4.2 Mounting Guidelines :

1. In order to mount the i-BMS15 on the battery pack we recommend that the i-BMS15 should be mounted on at least 4 positions (Refer to Figure 3).

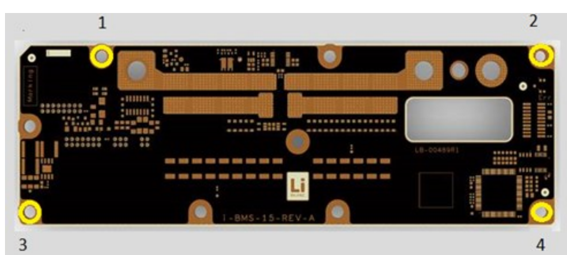


Figure 3 : i-BMS15 mounting positions

2. Support Pad / Spacer : The BMS should be mounted on support pads / spacer which will ensure distance between mounting surface and BMS is 20mm (minimum).

3. Bolt :
    a. It is important to remember that the bolt is mounted directly into the PCB copper material.
    b. Head Type – Socket / Button Headed cap screw
4. Torque : The tightening torque to be decided based on the screw selected.
5. Vibration : In very harsh environments it is recommend to use multiple mounting points to reduce the impact of vibration

9.4.3 HV Connection Guidelines

1. Bolt / Screw Details :
    a. Material : The two 6mm bolts (M6) in the high current connections are made of brass with a thread length of 6mm the Busbar material is copper.
    b. It is important that the cable lug is clean and flat so that the impedance between the cable lug and the bus bar is minimized.
2. Length : With assembled HV connection, the bolt should not protrude out of the boss on BMS.
3. Thread Details : The two high Current connections are supplied with a 6mm thread.
4. Use of washer is not recommended.
5. Torque requirement : The recommended torque is 5-6 Nm.
6. HV+IN and HV+OUT Connection :
    a. With Busbar : If the user is using a busbar to connect HV+IN and HV+OUT, then vibration transfer through the busbar to BMS to be arrested.
    b. With Wire Lug :
        i. If the user is using wire lug, then usage of sleeve on wire lug connection is required.
        ii. Wire Tension : Wire tension should be designed to avoid any stress on the BMS.
    c. Wire lug / Busbar orientation : Should be perpendicular to the BMS and wire / busbar should not cross the BMS.

9.5 Connectors & Wire Harness

Lithium Balance provides two sets of wire harness kits (p/n 100980 & 100981) , one each for parallel pack and regular pack architecture, and the same can be ordered together with the BMS board depending upon the application. Thus, providing a complete solution for integration of the battery pack with i-BMS15.

9.5.1 J101 Connector

The J101 Connector provides an interface for cell voltage (Cell 1 to 7) and temperature measurement (Channel 1 to 3). Figure 4 depicts a typical J101 connector and pin descriptions for the same.

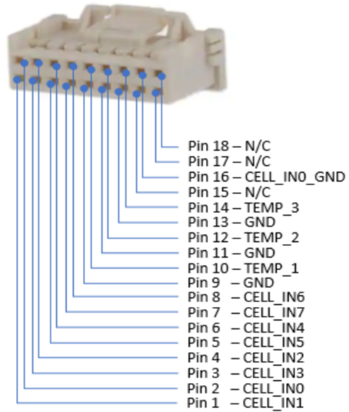


Figure 4 : J101 Connector

Table 2 summarises the J101 connector pin numbers, pin description and the colour codes associated with each pin for easy connection of the connector with the wire harness.

Pin No.	Pin Name	Wire Colour	Pin Description
1	CELL_IN1		Cell 1 + Voltage Channel
2	CELL_IN0		Cell 1 - Voltage Channel
3	CELL_IN3		Cell 3 + Voltage Channel
4	CELL_IN2		Cell 2 + Voltage Channel
5	CELL_IN5		Cell 5 + Voltage Channel
6	CELL_IN4		Cell 4 + Voltage Channel
7	CELL_IN7		Cell 7 + Voltage Channel
8	CELL_IN6		Cell 6 + Voltage Channel
9	GND		Cell Temperature Channel 1 Ground
10	TEMP_1		Cell Temperature Channel 1
11	GND		Cell Temperature Channel 2 Ground
12	TEMP_2		Cell Temperature Channel 2
13	GND		Cell Temperature Channel 3 Ground
14	TEMP_3		Cell Temperature Channel 3

15	N/C	-	Not Connected
16	CELL_IN0_GND		Power Supply Ground for BMS from Cell 1 -
17	N/C	-	Not Connected
18	N/C	-	Not Connected

Table 2 : J101 Connector Pinout Description

Important Note : For temperature measurement, one end of each thermistor must be connected to a temperature input (Pins 10, 12 & 14) and the other end to the corresponding ground (Pins 9, 11 & 13). It is not important which end of the thermistor is connected to the ground pin and to the input pin.

Manufacturer Code : Molex p/n 5016461000

9.5.2 J102 Connector

The J102 Connector provides an interface for cell voltage and temperature measurement. Figure 5 depicts a typical J103 connector and pin descriptions for the same.

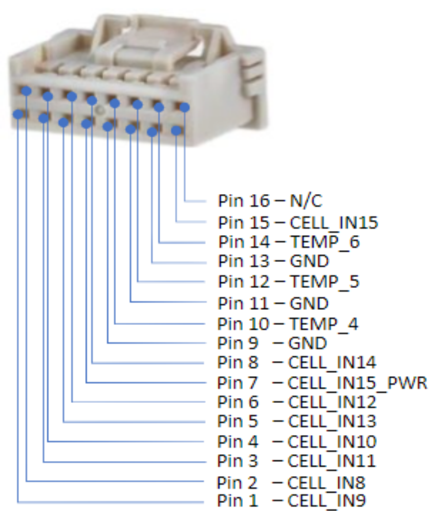


Figure 5 : J102 Connector

Table 3 summarises the J102 connector pin numbers, pin description and the colour codes associated with each pin for easy connection of the connector with the wire harness.

Pin No.	Pin Name	Wire Colour	Pin Description
1	CELL_IN9		Cell 9 + Voltage Channel
2	CELL_IN8		Cell 8 + Voltage Channel
3	CELL_IN11		Cell 11 + Voltage Channel

4	CELL_IN10		Cell 10 + Voltage Channel
5	CELL_IN13		Cell 13 + Voltage Channel
6	CELL_IN12		Cell 12 + Voltage Channel
7	CELL_IN15_PWR		Power Supply for BMS from Cell 15 +
8	CELL_IN14		Cell 14 + Voltage Channel
9	GND		Cell Temperature Channel 4 Ground
10	TEMP_4		Cell Temperature Channel 4
11	GND		Cell Temperature Channel 5 Ground
12	TEMP_5		Cell Temperature Channel 5
13	GND		Cell Temperature Channel 6 Ground
14	TEMP_6		Cell Temperature Channel 6
15	CELL_IN15		Cell 15 + Voltage Channel
16	N/C	-	Not Connected

Table 3 : J102 Connector Pinout Description

Manufacturer Code : Molex p/n 5016461000

9.5.3 J100 Connector

The J100 Connector provides an interface for General Purpose Output, E-Stop and Ignition signals. The Ignition Interface is present on J100 Connector to "wake-up" the i-BMS15 from sleep mode. The 5V output on pin 2 can be used to power auxiliaries and is limited to 250 mA current consumption (thermally fused). Figure 6 depicts a typical J100 connector and pin descriptions for the same.

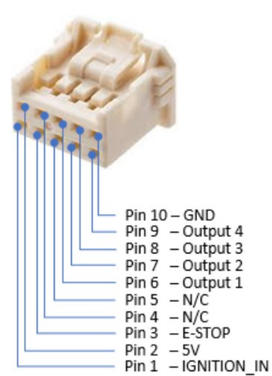


Figure 6 : J100 Connector

Table 4 summarises the J100 connector pin numbers, pin description and the colour codes associated with each pin for easy connection of the connector with the wire harness.

Pin No.	Pin Name	Wire Colour	Pin Description
1	IGNITION_IN		Input Ignition Signal (Wake-Up)
2	5 V		5 V Aux. Output
3	E-STOP_IN		Input Emergency Stop Signal
4	IGNITION_POWER	-	Power to Ignition
5	N/C	-	Not Connected
6	OUTPUT 1		General Purpose Output 1
7	OUTPUT 2		General Purpose Output 2
8	OUTPUT 3		General Purpose Output 3
9	OUTPUT 4		General Purpose Output 4
10	GND		Ground

Table 4 : J100 Connector Pinout Description

Manufacturer Code : Molex p/n 5016461000

9.5.4 J106 Connector

The J106 Connector provides an interface for Serial CAN Communication. Figure 7 depicts a typical J106 connector with and without termination and pin descriptions for the same. CAN Wiring with termination ensures that the CAN signals, being a two-way communication, are not reflected from the cable ends. The reflection of CAN signals increases with the length of the cables as well as the CAN bus baud rates. This is why it is recommended to add proper termination in larger CAN networks.

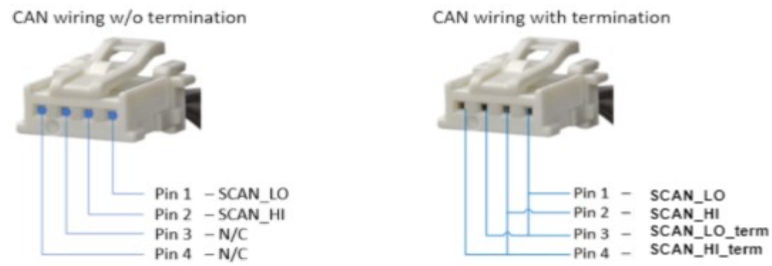


Figure 7 : J106 Connector without termination (L) and with termination (R)

Table 5 summarises the J106 connector (CAN without termination) pin numbers, pin description and the colour codes associated with each pin for easy connection of the connector with the wire harness.

Pin No.	Pin Name	Wire Colour	Pin Description
1	SCAN_LO		s-CAN Low Data Signal
2	SCAN_HI		s-CAN High Data Signal
3	N/C	-	Not Connected
4	N/C	-	Not Connected

Table 5 : J106 Connector (CAN without Termination)

Table 6 summarises the J106 connector (CAN with termination) pin numbers, pin description and the colour codes associated with each pin for easy connection of the connector with the wire harness.

Pin No.	Pin Name	Wire Colour	Pin Description
1	SCAN_LO		s-CAN Low Data Signal
2	SCAN_HI		s-CAN High Data Signal
3	SCAN_LO_term		s-CAN Low Termination
4	SCAN_HI_term		s-CAN High Termination

Table 6 : J106 Connector (CAN with Termination)

Figure 8 shows the wiring diagram for CAN with termination. The wires from Pin 1 shall be shorted with the wire from Pin 3 for the termination of s-CAN Low data signal. The wires from Pin 2 shall be shorted with the wire from Pin 4 for the termination of s-CAN High data signal.

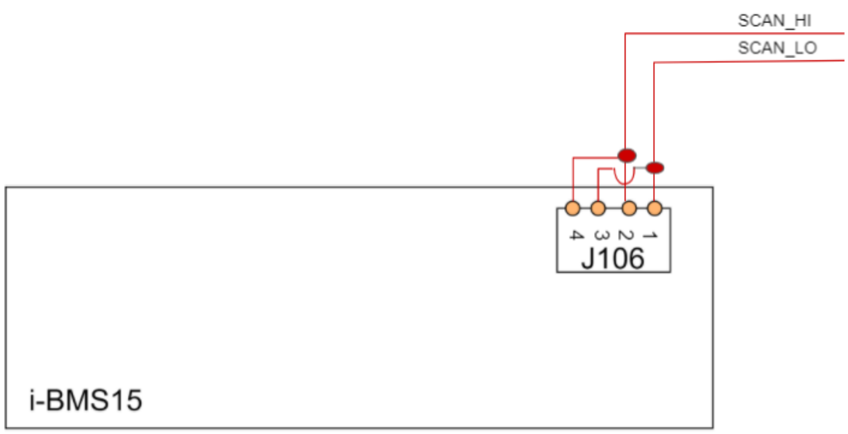


Figure 8 : Wiring Diagram for CAN with termination

For wiring of CAN without termination, dedicated wires for s-CAN High and s-CAN Low data signals shall be connected to Pin 2 and Pin 1 respectively.

Important Note : If the wire harness kit for i-BMS15 integration with the battery pack is ordered from Lithium Balance (p/n 100980 & 100981), it is important to note that wire harness kit for parallel pack architecture will include CAN Wiring without termination and the wire harness kit for standalone packs will include CAN Wiring with termination.

9.5.5 J500 Connector

J500 Connector has been allocated to JTAG (named after the Joint Test Action Group which codified it) which is an industry standard for verifying designs and testing printed circuit boards after manufacture.

9.5.6 Ignition

The Ignition Interface is present on J100 Connector to "wake-up" the i-BMS15 from sleep mode. In order to wake up the i-BMS15 an active signal of at least 5V must be supplied to J100 pin 1 and after a maximum of one second the i-BMS15 will power up. The Active signal can be applied to pin 1 by connecting/shorting J100 pin 4 to pin 1 for at least one second.

9.5.7 Connections for Cell Voltage Monitoring

While connecting the i-BMS15 to a battery, it is very important to ensure proper wired connections in order to monitor cell voltages as well as supply power to the BMS. Figure 9 below shows how to connect the cell voltage monitoring wires for a battery pack with 15 cells in series. The wires from cell voltage monitoring channels on J103 Connector (Pin no 7 & 15) must be connected together onto the positive terminal of the 15th cell. The wires from cell voltage monitoring channels on J104 Connector (Pin no 2 & 16) must be connected together onto the negative terminal of the 1st cell. The blue wires in Figure 9 depict power supply lines and the user must ensure proper connections for the same.

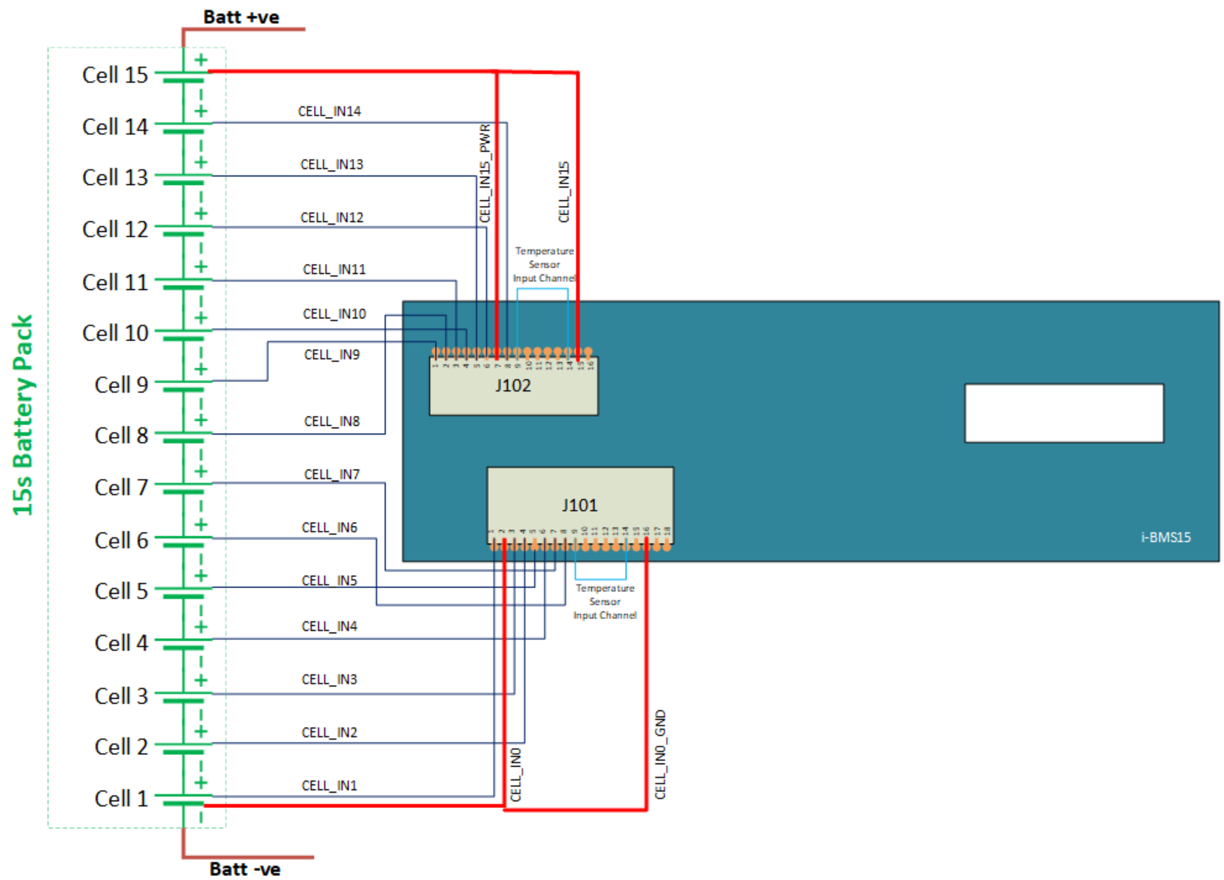


Figure 9 : Wiring Schematic for 15 cells in series

In case the battery connected to the i-BMS15 has fewer cells than 15 (for e.g. 10 cells as depicted in Figure 10, the wires from cell voltage monitoring channels on J102 Connector (Pin nos 3, 4, 5, 6, 7, 8 & 15) must all be connected together onto the positive terminal of the 15th cell. The wires from cell voltage monitoring channels on J101 Connector (Pin nos 2 & 16) must be connected together onto the negative terminal of the 1st cell. The red lines in Figure 10 depict power supply lines and the user must ensure proper connections for the same.

Important Note : When more than two wires from different channels are shorted together, it is advisable to short them as close as possible to the point where they are connected to the particular cell.

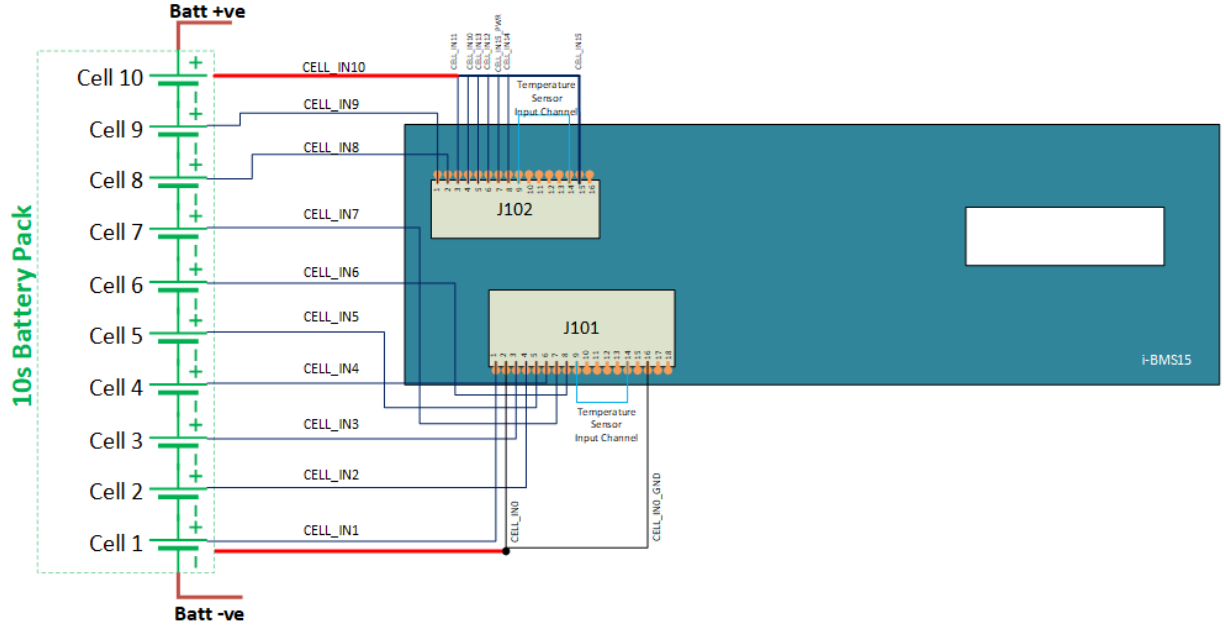


Figure 10 : Wiring Schematic for 10 cells in series

10 i-BMS15 Functionalities

The purpose of this chapter is to provide the user with an overview of the functionalities associated with i-BMS15 and procedures to enable them for use. The functionalities include active measurements (current, voltage and temperature), operational limits, estimated parameters, operating modes and state machine, controls and error handling.

10.1 Safety & Threshold 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. i-BMS15 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 i-BMS15 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.

Important Note : Setting these thresholds correctly are 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.

10.2 Current Measurement

i-BMS15 has an on-board shunt resistor which measures the overall battery pack current. The shunt sensor is calibrated during production of the i-BMS15 and no further configuration is needed for the same.

Shunt is a low-value high-precision resistor placed in series with the battery pack to measure the current flow. This is done by measuring the voltage drop across the shunt resistor and making use of $I = V_{shunt}/R_{shunt}$. Figure 11 depicts the general operating principle of a typical shunt based current sensor.

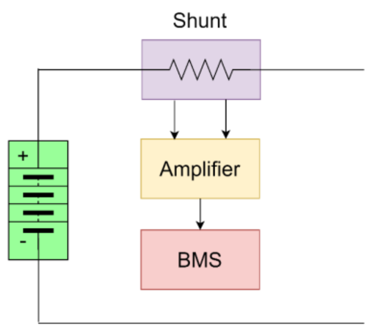


Figure 11 : Shunt Operating Principle

10.3 Temperature Measurement

The i-BMS15 is designed to accommodate 4 on-board and 6 external temperature sensors. The 4 on-board temperature measurements include : MOSFET temperature, Balancing Circuitry temperature and PCB Ambient temperature. The 6 external temperature sensors can be placed inside the battery pack based on results obtained from the thermal analysis of the pack and can be interfaced with i-BMS15 through J103 and J104 Connectors.

i-BMS15 supports NTC (Negative Temperature Coefficient) thermistors for external temperature measurements. Thermistors are semiconductor based temperature sensing elements that are sintered to exhibit large variations in resistance in proportion to small changes in temperatures. This resistance can be calculated by measuring the voltage drop when a small (known) current is passed through the thermistor. This resistance tends to decrease non-linearly with an increase in temperature for NTC thermistors. The nature of this non-linear decrease in resistance is related to a constant called ß-value. ß-value represents the relationship between the resistance and temperature of the thermistor.

The temperature sensor detection system for i-BMS15 is designed for 10 kΩ NTC sensors with a default ß-value of 3900. However, for increased support and flexibility, the resistance-to-temperature characteristics are user-configurable through a set of parameters. A separate calculation tool will be made available to the user for determination of parameters (Steinhart-Hart coefficients) necessary to obtain the best resistance to temperature relationship for the particular NTC thermistor used by the user.

Important Note : The measuring range for the on-board sensors is -40°C to 120°C. The user shall make sure the limits set in the configuration software (Refer to Chapter 4) is within the above mentioned range. The bleeding (balancing) temperature limit is hard-coded at 90°C and the bleeding will be turned on as soon as the board temperature again gets below 90°C.

10.4 Cell & Pack Voltage Measurement

To achieve reliable and robust operation out of a battery system, it is essential to always keep the voltages of all lithium cells within its safety 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 i-BMS15 is designed to measure voltages from a minimum of 8 cells in series to a maximum of 15 cells depending upon the user application. The i-BMS15 is capable of measuring the pack voltage at the battery HV terminals and the same is used for :

- Validation against sum of individual cell voltage measurements
- Determination of the pre-charge voltage
- Measuring the pack voltage during paralleling

10.5 Operating Modes & State Machine

The BMS is designed as a state machine which can be in several different modes as shown in Figure 12. Before entering a new state, an error checklist is verified to allow access to the new state:

- If no errors are present, the BMS will move to the next level in the state machine.
- If errors on the relevant checklist are present, the BMS will enter the error mode.

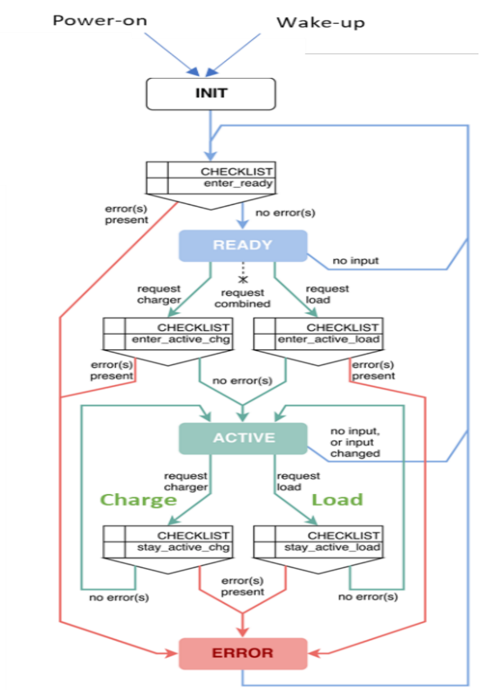


Figure 12 : i-BMS15 state machine

The i-BMS15 is started up by applying main supply voltage (nominal 12 V). This will now bring the BMS out of sleep mode. After starting-up, the BMS now enters into the basic initialisation (INIT) mode. If no critical errors are present, it will immediately move on to the READY mode.

10.5.1 Sleep Mode

The BMS supports sleep mode, which is entered by sending a CAN message to the i-BMS15 (Refer to section 11.10.5). Waking up the BMS can be done in two ways:

- Via the ignition-pin (Refer to section 9.5.6).

- Via a CAN wake-up frame. Any communication on the CAN bus will wake-up the i-BMS15.

10.5.2 Boot/Configuration Mode

The BMS boot-mode indicates a mode outside of the state diagram of Figure 12. In this mode the system is ready to receive new firmware or new configuration (*.XML) files. Once these have been successfully installed the system will leave boot mode and move into the so-called “application mode”, which corresponds to entering the state diagram of Figure 12.

Important Note : In Boot/Configuration mode all contactors are open!

10.5.3 Ready mode

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

Important note : If BOTH the ‘Request Charge’ or the ‘Request Load’ signals are active, the system will by default enter the ‘Active Charge’ mode.

10.5.4 Active mode

In the ACTIVE mode, contactors will be closed 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 Charge’ and the ‘Request Load’ signals are inactivated in which case the BMS will go back to Ready mode
- If a critical error appears and the system enters error mode

In the Active State, there are two operating modes - “Active Charge” mode and “Active Load” Mode. It is important to understand that there are no limitations in terms of direction of current flow in either “Active Charge” or “Active Load” modes. This implies that :

- The BMS can accept charge current or the battery pack can undergo charging in the “Active Load” mode.
- The BMS will allow current draw from the battery in the “Active Charge” mode.

It is due to this reason that the battery pack can be charged during regeneration while driving.

Important Note : It is important for the user to understand the difference between “Active Charge” and “Active Load” modes. The error checklists involved are different for the BMS to enter into one of these modes. Also, in “Active Charge” mode charge requests are made through CAN messages which is not the case in “Active Load” mode.

10.5.5 Error Mode

In Error mode, the BMS will set the battery in a passive state by ensuring 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 'Debounce time' before it can re-enter the active modes (close contactors) after a critical error has forced the contactors to open.

10.6 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.

10.6.1 Error Checklists & Assigning Actions to Errors

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 (Refer to Section 10.5), which in turn controls the contactors. These checklists control the BMS state machine transitions and are thus one of the most important 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 desirable in all applications and situations. 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 cause the BMS to exit that mode.

The first step is to consider what battery related errors should be acted upon 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.

10.6.2 Error Behaviors

With i-BMS15, the user can configure a default error behavior as well as create custom error behaviors that can be assigned to specific errors. These error behaviors are basically a filtering mechanism for accurate reporting of the errors, lasting time of the errors and removal/erasing of the errors based on user inputs in BMS Creator.

The idea is to associate a counter with each error in order to avoid collision but at the same time follow a typical behavior assigned to it by the user. For example, if an error related to cell voltage is occurring, the counter will count up or down according to the behavior assigned to it. But if there is another error related to communication after some time, it will follow the same behavior but this will happen on an individual memory location with each error having its own counter.

In order to understand the parameters associated with configuration of Error Behavior, refer to Section 11.10.11.

10.7 Operational Current Limits

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 four current limits that are used for thresholds:

- Continuous Current Limits : The current limits while the BMS is in operation
  - Maximum current into the battery.
  - Maximum current out of the battery.
- Advanced Current Limits : The current limits “Max i2t” are used to allow short periods of peak currents beyond the continuous current limits
- Maximum i2t sum for peak currents during charge mode.
- Maximum i2t sum for peak currents during discharge mode.

If the BMS is not in active state, the default current limits are:
- Current in (Charge): 0.5 A
- Current out (Discharge): 0.1 A

10.7.1 Continuous Current Limits

The i-BMS15 defines two configurable data arrays for continuous current limits:
- Dynamic current limit in (DCLI)
- 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 i-BMS15 uses two 2-D lookup tables, one for DCLI and another for DCLO. The tables have two dimensions: one dimension being the SOC, and the other being the cell temperatures. There are 7 custom temperatures that can be defined by the user and 11 slots for the SOC which is fixed (not configurable) at an interval of 10%, starting at 0 % SOC. In this section, the DCLI concept is explained with an example.

Table 7 depicts a typical DCLI configuration table. For e.g. Current limit for 80% SOC and at 25°C is 150A. :

SOC/Temp	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
-20ºC	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A
-10ºC	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A
-5ºC	5A	5A	5A	5A	5A	5A	5A	5A	5A	5A	5A
0ºC	50A	100A	100A	100A	100A	100A	100A	100A	100A	80A	10A
15ºC	100A	100A	100A	100A	100A	100A	100A	100A	100A	80A	10A
25ºC	150A	150A	150A	150A	150A	150A	150A	150A	150A	90A	10A
45ºC	100A	100A	100A	100A	100A	100A	100A	100A	100A	80A	10A

Table 7 : A DCLI Configuration Table Example

A graphical representation of the DCLI datasets is shown in Figure 13.

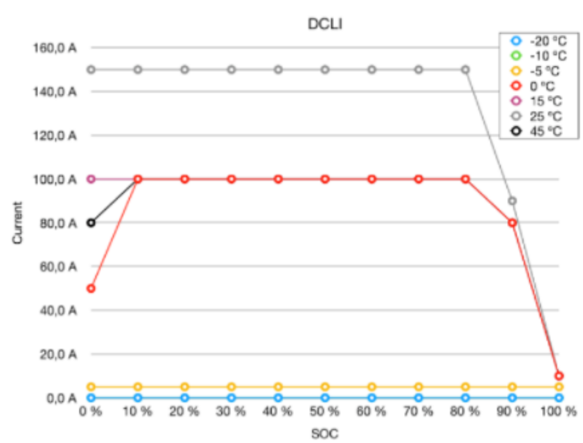


Figure 13 : DCLI 2-D Graph

The i-BMS15 has been programmed to interpolate between data points in order to ensure maximum accuracy, for e.g. When SOC is at 85 % and measured temperature is 12°C, the 4 data points that undergo interpolation are as shown in Figure 14. Interpolation then returns the result of DCLI = **99.0 A**. 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.

SOC/Temp	80%	90%
15ºC	100A	80A
25ºC	150A	90A

Figure 14 : Cross-section of interpolation example (DCLI)

Figure 15 shows the graphical representation of the DCLI interpolation example :

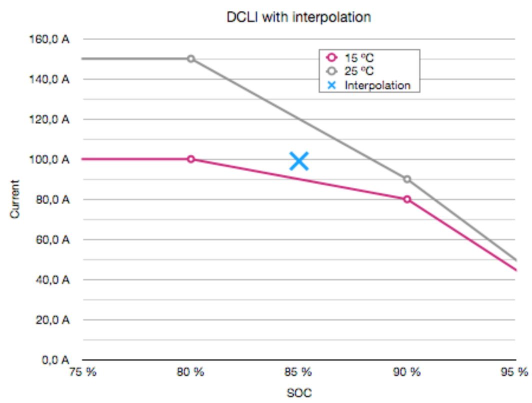


Figure 15 : DCLI Interpolation Graph

Important 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 also taking dynamic current limit IN (DCLI) data into account. The charge current request calculated by BMS follows the limits of DCLI, which is a function of temperature and SOC. In case the user defines DCLI as 0 A for 95% SOC and 45°C, the BMS will request 0 A from the charger. It is because the charge current request algorithm takes into account what DCLI limits are. The charge current request cannot be higher than what DCLI is.

10.7.2 Advanced Current Limits

The BMS calculates the "i^2t-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. The excess power shall be understood in the following way :

- 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 is considered acceptable for 2.5 seconds. Consequently, the i^2t threshold is set to $(200A)^2$ x 2.5s = 100,000 A^2s. The forklift is now performing the following actions: (Refer to Figure 16)

Case 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.
Case 2 : A load is now lifted to a shelf which requires 200A for a few seconds. This exceeds the Max Current Out and the i^2t parameter therefore increases.
Case 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.
Case 4 & 5 : A load requiring 200 A is again lifted but this time the driver mistakenly lowers the load. In this process 125A of regenerative power is generated. This exceeds the Max Current In of 75A and the i^2t parameter therefore continues to increase, though at a lower slope.
Case 6 : The forklift picks up a new load requiring 20 A and the i^2t parameter decreases to 0.
Case 7 : A heavy load is now lifted to a higher shelf which requires 300A for several seconds. The i^2t parameter therefore increases significantly. In this case the i^2t threshold of 100,000 A^2s is breached and the “2010 ERROR_SYS_LIM_PACK_I2T” error is activated by the BMS.

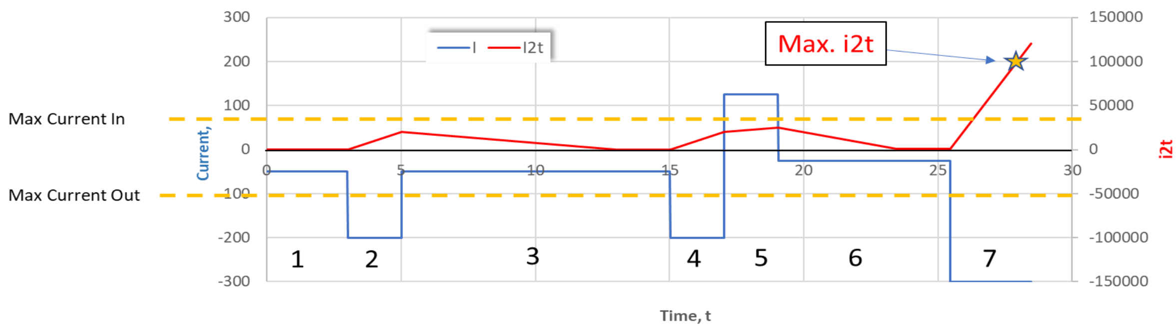


Figure 16 : Graphical Representation of the Forklift Example

10.8 Pre-Charge & Contactor Control

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. 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.

The i-BMS15 has on-board contactors and pre-charge circuitry. The following are the contactor on and off sequences followed in i-BMS15 :

- Contactor activation sequence

  When the i-BMS15 receives a request for Load/Charge, the pre-charge contactor will close at least for the configured 'pre-charge settling' time and maximum for the 'pre-charge timeout' time. Within this time window, the current and voltage are measured and both must drop below the defined limits for either Load or Charge mode as configured.

  As soon as the pre-charge conditions are met, the main contactor will close and the pre-charge contactor will open. The i-BMS15 will now enter active mode. If the pre-charge conditions are not met, the pre-charge contactor will open and retry after the configured 'contactor retry timeout' (Refer to Appendix 4).

- Contactor off sequence

  When requested to open contactors (Contactors Off), the BMS will, in order to protect the contactors, not open them until either (Refer to Appendix 4):
  - The current is below the "Max contactor break current."
  - The time passed since the 'Contactors Off' request exceeds the "Contactors off timeout".

  To override the contactor off sequence the "Emergency Stop (E-Stop)" must be used.

10.9 E-Stop

The i-BMS15 ensures precise contactor control in order to ensure the safety of li-ion cells, connectors or load/charge equipment. But in some critical situations, it might be necessary to perform an emergency opening of the contactors. In the i-BMS15, this functionality is called E-Stop and there are two ways of implementing E-Stop in i-BMS15 :

- Hardwired E-Stop : When activated, the power supply to the iBMS is cut off resulting in the opening of contactors. Pin no. 3 of J100 Connector can be used for E-Stop signalling. This signal has been pulled up internally in i-BMS15, meaning that the signal is active low and can be left unconnected if not intended to be used.

  In order to activate the E-Stop functionality, pin no. 3 on J100 and the ground signal from pin no. 10 on J100 can be used together with a relay. This relay must be controlled externally and when closed the E-Stop functionality is enabled, thus shutting the i-BMS15 down.

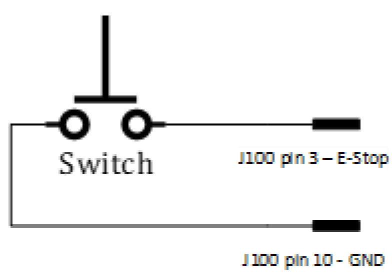


Figure 17 : Hardwired E-Stop

- CAN controlled E-Stop : The i-BMS15 state machine will continuously check for the CAN E-Stop message. When activated, the i-BMS15 contactors will be opened despite request for load or charge but also ignoring the current and voltage measured. For configuring CAN-based E-Stop (Refer to Section 11.10.6).

10.10 Calculated & Estimated Parameters

The BMS makes use of measured battery data to calculate and estimate some important operational parameters of the battery.

10.10.1 Calculated Parameters from Cell Voltage Measurement

Every time the cell voltages are measured and updated, update interval is configurable, the BMS will calculate certain voltage related parameters for every measured cell, which are (Refer to Appendix 5) :

- 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)

10.10.2 State of Charge (SOC) Estimation

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 SOC is expressed in percentage points (0% = empty; 100% = full).

Figure 18 : SOC equivalence with Fuel Gauge

The i-BMS15 makes use of a hybrid estimation algorithm in order to ensure maximum accuracy of SOC prediction. We must understand that by hybrid approach we mean,

- SOC calculated from Coulomb Counting approach, SOC_{CC}
- SOC based on SOC-OCV relationship, SOC_{OCV}

In the Coulomb Counting method, SOC is calculated from the ratio of capacity removed (or added) to the nominal (original) capacity of the battery. This is given by :

$$\mathbf{SOC_{CC} = SOC_{initial} \pm \{(\int I\,dt) / Q_{calib}\}}$$

where,
$SOC_{initial}$ = SOC determined from calibration or a previously stored value
I = Pack Current
Q_{calib} = Calibrated Capacity of the pack/cell

Due to integration of this measured current over time, errors (due to sensor noise/leakage/self-discharge) tend to accumulate when the cell/battery is charged and discharged for multiple cycles. Because of this, relying on just the coulomb counting approach can lead to inaccurate SOC estimations.

The i-BMS15 algorithm corrects this by constantly comparing the SOC_{CC} to the SOC obtained from a reference SOC-OCV relationship of the battery/cell in application, SOC_{OCV} (Refer to Figure 19). The Open Circuit Voltage (OCV), which is a parameter dependent on li-ion cell chemistry, cannot be measured directly and is calculated from an advanced battery model. As we know that SOC is a function of OCV and temperature, the calculated OCV is used to determine the corresponding SOC_{OCV} with the help of a 2-D Lookup Table (OCV-Temperature-SOC).

The i-BMS15 advanced SOC estimation algorithm has inbuilt conditions which helps it to make decisions on when to calibrate the SOC_{CC}. These conditions rely on the inputs received from the user depending upon the cell chemistry and use case. Thus, ensuring that the estimated SOC is as close as possible to the true SOC of the battery/cell.

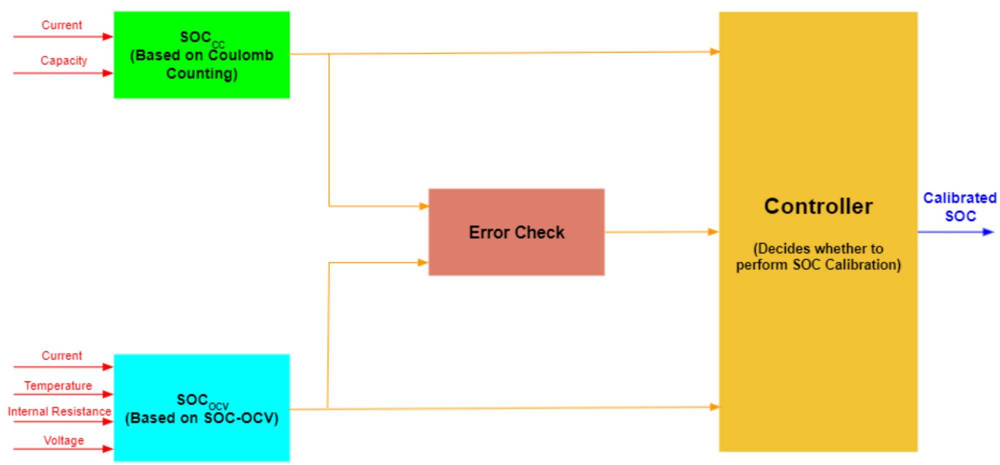


Figure 19 : i-BMS15 SOC Calibration Algorithm

10.10.3 State of Health (SOH) Estimation

State of health (SOH) is a measure of performance degradation of a battery pack. As Li-ion batteries age, their performance will decrease. The performance degradation can be primarily attributed to two important factors :

- Capacity fade, SOH_Q
- Internal resistance increase, SOH_R

The capacity fade phenomenon is caused by active materials/sites 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, $Q_{available}$.In order to quantify the contribution of capacity fade to the overall SOH of the battery, the following relationship is used :

$$\mathbf{SOH_C = Q_{available}/Q_{nom}}$$

where, Q_{nom} is the nominal or initial capacity of the battery.

The increase in internal resistance of the battery can be attributed to unwanted side reactions taking place inside the cell and structural deterioration of the active materials. This leads to a reduced overall ionic/electronic conductivity of the battery. The i-BMS15 algorithm takes this into account by measuring the real time resistance of the battery and validates it based on charge/discharge cycle number and the cell chemistry.

10.10.4 State of Power (SOP) Estimation

State of Power (SOP) is a measure of constant power available from the battery during a particular time interval “t”. SOP is a function of temperature and SOC of the cells. This has many applications, such as determining whether a quick overtaking is feasible with an electric vehicle (e.g. power available the next 10 seconds).

The i-BMS15 algorithm calculates the SOP for a time interval of “t” seconds for charge as well as discharge and the same can be requested from the BMS at any point of time. The algorithm makes sure the following conditions are met while calculating the SOP :

- The maximum allowed power should not exceed the maximum/minimum cell voltages, P_V.
- The maximum allowed power should not exceed the “i^2t” parameter, P_{i2t}.

The minimum of the two i.e. P_V and P_{i2t} is the maximum available power for the next “t” seconds. This is given by the relationship :

$$\mathbf{SOP(t)_{chg/dschg} = Min\ \{P_V,\ P_{i2t}\}}$$

Important Note : The interval for SOP is fixed at 1 second and 10 seconds respectively currently (Refer to Data IDs 1077 & 1078 in Appendix 5)

10.10.5 State of Energy (SOE) Estimation

State of Energy (SOE) is a measure of the residual energy available in a battery at any point of time. It is expressed in W-s (Watt Seconds). SOE can be best understood when we compare it to SOC. SOC provides us with the information of how much current is available over a specific time interval. Whereas, SOE provides us with the information that at what voltage this current is available. Let us consider an example to realize the importance of SOE :

A 4 Ah cell with an average voltage of 3.7 V in the SOC range 90%-100% and an average voltage of 2.6 V in the SOC range of 0-10% (Refer Figure 20). From SOC 100% to 90%, energy available for use is 13.3 Wh and from SOC 10% to 0% energy available for use is 1.04 Wh. Although, the SOC is 10% in both the cases, there is a huge difference in SOE. This is typically the reason why we exhaust our mobile battery much more quickly when the SOC is low as compared to when the SOC is high.

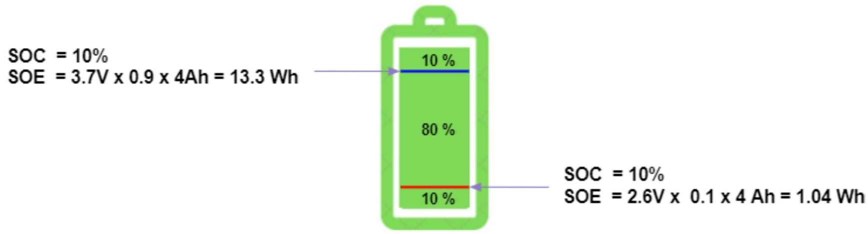


Figure 20 : SOC & SOE Example

The i-BMS15 algorithm calculates the SOE by taking into account the averaged OCV, Initial Capacity, SOC based on Coulomb Counting and the SOH (capacity degradation) of the battery.

10.11 Charging

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 to stop charging:

- The primary method for stopping or modifying charging is via a communication protocol from the BMS, as described in this section.
- 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 via a CAN message, 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, depending on configuration.

10.11.1 CAN-based 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 i-BMS15 has several built-in CAN charger formats that may be used instead of a user-configured CAN frame.

10.11.2 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 cell with highest measured voltage (V_{max}), i.e.

$$\mathbf{I_{charge} = PID(V_{target} - V_{max})}$$

The current requested by the PID will always adhere to the dynamic current limitations defined in Section 10.7.1. 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 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 DC/DC converter or similar load is connected to the BMS during charging.

Important 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 the user specified as IN current limit (charging current limit) for various temperature and SOC conditions. The user must set DCLI settings properly so BMS will not end up requesting 0 A unintentionally. For example, if the user defined DCLI as 0 A for 5°C and all SOC% values, BMS will not ask any current higher than 0 A when the temperature input is 5°C.

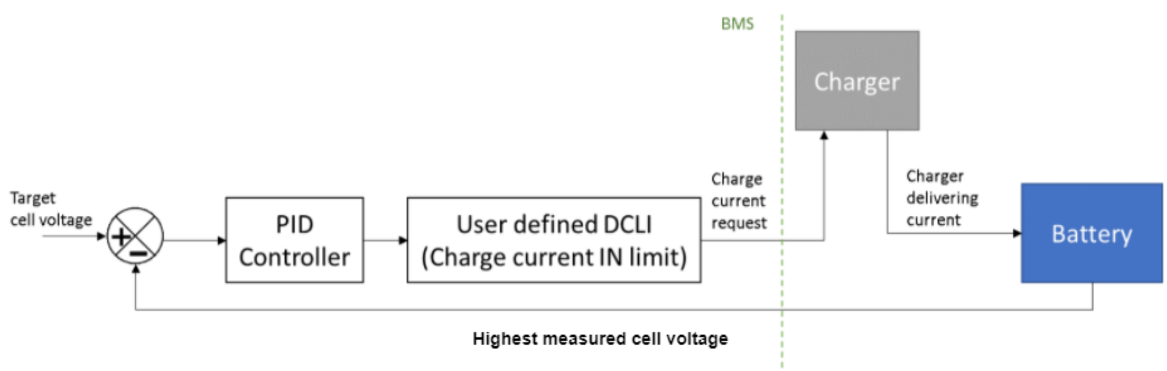


Figure 21 : Block diagram showing the relation of charge current request with PID controller and dynamic current limit IN (DCLI).

10.12 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 when requested by a specific CAN message. While the configured CAN message is received, 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 CAN message is received. 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.

10.12.1 Balancing based on Voltage & Current Measurements

When charging, the system adjusts charger and balancing current levels to assure that at the end of charging all cell voltages are at the “Cell Target Voltage” with an accuracy defined by the “Charge Complete dead band.” The “Balancing Limit” is the lower voltage limit for balancing to be active. It is a constant parameter which gives flexibility to the user to optimize (when do they want to activate balancing and when not) the battery energy based on their application.

“Balancing Target” is used (internally) to identify the cells which need balancing and is controlled by the BMS. The value of Balancing Target is conditional and is given by:
Case 1 : Balancing Target = Average cell voltage ; If the charge complete deadband has not been reached by any of the cells while charging.
Case 2 : Balancing Target = Lowest cell voltage (V_{min}) + balancing deadband ; If either charge complete deadband band for current or charge complete deadband for voltage has been reached by any one of the cells in the charging process.

This can be understood with the help of an example. Let us consider 5 li-ion cells undergoing the charging and balancing process managed by i-BMS15 (Refer Figure 22 & Figure 23)

- **Case 1 :** (Refer Figure 22) In this case, none of the cells have reached the ‘Charge Complete Deadband’ for voltage or current. Therefore, the balancing limit here is equal to the ‘Cell Average Voltage’. Cells 2, 4 & 5 will bleed (marked by red arrows) while cells 1 & 3 will undergo normal charging (marked by green arrows). The bleeding will continue until all the cells reach the ‘Balancing Deadband.’

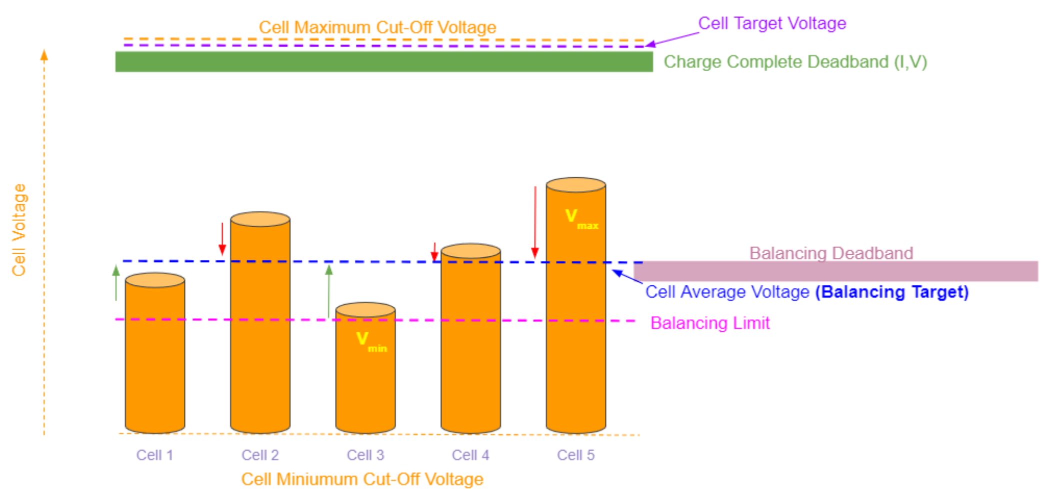


Figure 22 : Balancing algorithm for Case 1

- **Case 2 :** (Refer Figure 23) In this case, cells 2 & 5 have reached the ‘Charge Complete Deadband’ (voltage or current). Therefore, the balancing target is equal to (V_{min} + ‘Balancing Deadband’). Cells 2, 4 & 5 will bleed (marked by red arrows), cell 3 will undergo normal charging (marked by green arrows) while cell 1 will remain as it is because it is already in the ‘Balancing Deadband’. The bleeding will continue until all cells reach the ‘Balancing Deadband.’

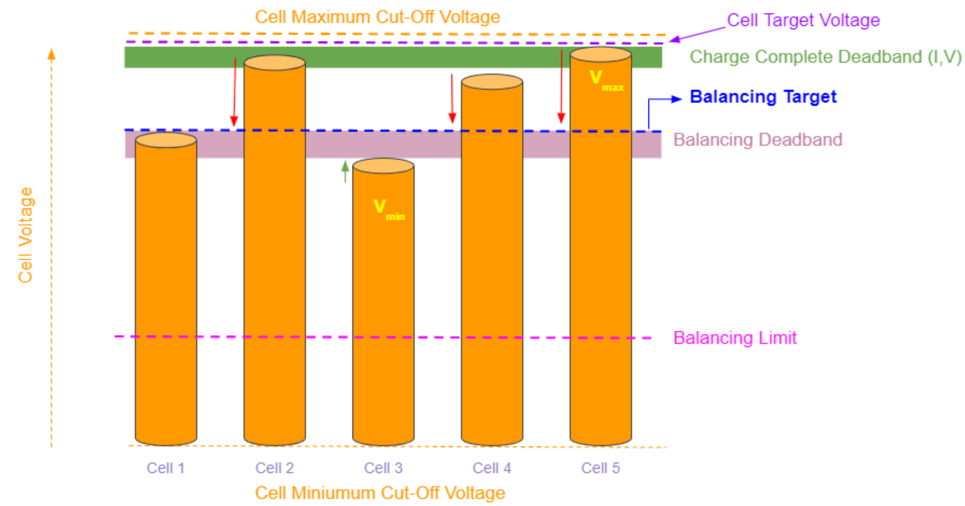


Figure 23 : Balancing Algorithm for Case 2

- **Case 3 :** The balancing process is complete once all the cell voltages are within the 'Balancing Deadband'. Figure 24 depicts the stage when all the cells have successfully undergone the balancing process.

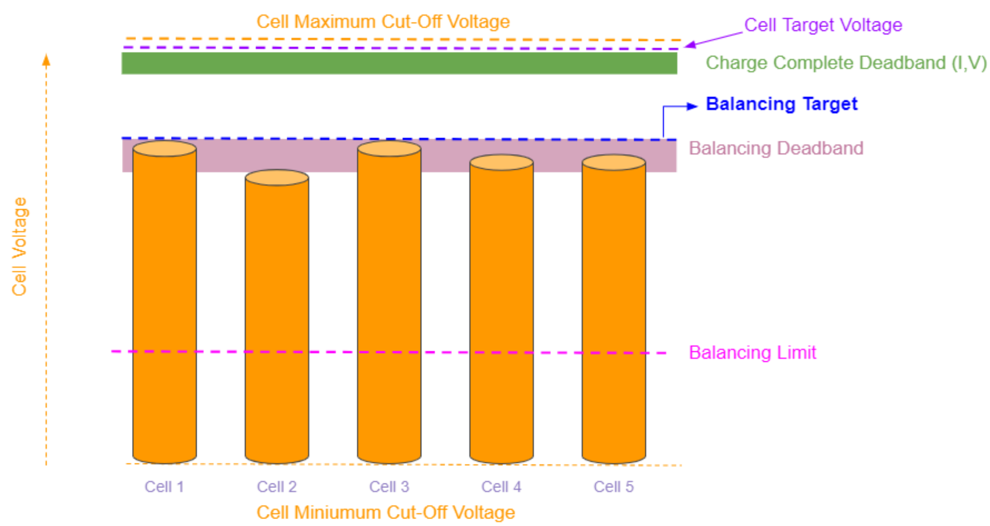


Figure 24 : Balancing Completed

Important Note : The above example depicts the balancing process when the pack is in 'Charge Mode.' It becomes important to understand that the balancing process is not only restricted to 'Charge Mode', the cells will also undergo balancing if the pack is outside of 'Charge Mode' (for e.g. 'Ready Mode'). In that case, the algorithm remains the same but the cells will only bleed and not undergo charging simultaneously. In order to understand this better Refer Figure 22 & 23, the green arrows (depicting charging) will not be there in case of any mode outside of 'Charge Mode.'

10.12.2 Balancing based on SOC

The i-BMS15 is also equipped with algorithms to incorporate balancing based on SOC_{OCV} (SOC derived from SOC-OCV relationship). This can be enabled/disabled by the user depending upon the need and application. The purpose of balancing based on SOC is to bring the SOC of each individual cell very close to the average SOC of all the cells. Therefore, the average SOC becomes the balancing target for each and every cell.

The balancing target is expressed in Ampere-seconds (A-s). Once calculated, the balancing target is stored in memory and needs to be updated regularly when balancing is performed. This update can be configured by the user (Refer to Section 11.10.9) and depends on the accuracy of the SOC_{OCV}. If the SOC accuracy is good, the balancing target is updated and if the accuracy is bad, the balancing target is not updated and the last stored value is used for performing balancing.

10.13 Paralleling

i-BMS15 can support paralleling of upto 6 similar battery modules and all of them being connected to one common battery module. Each of these modules are controlled and monitored by individual i-BMS15 systems.

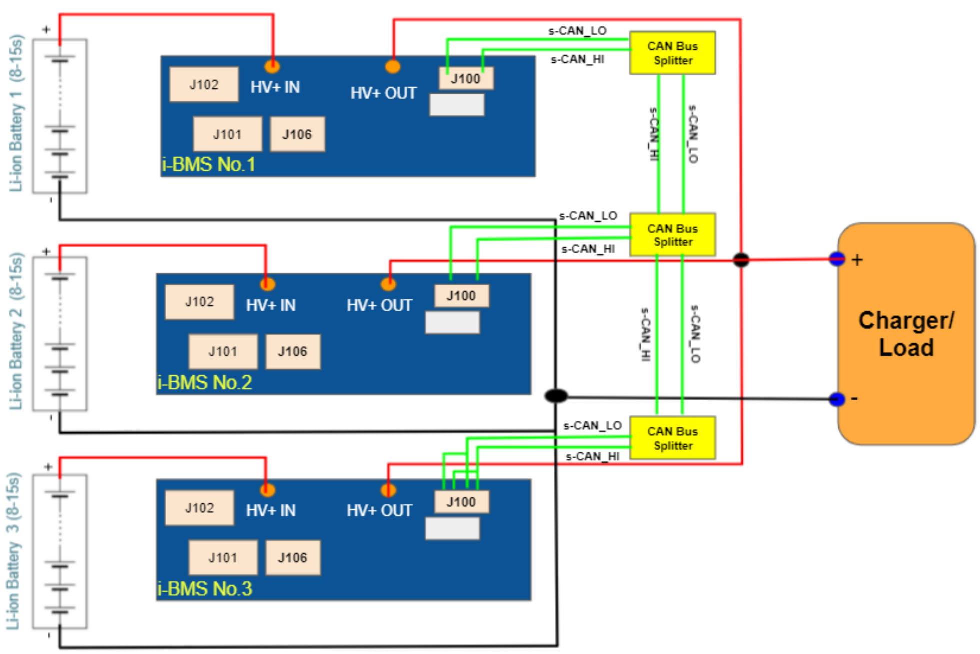


Figure 25 : 3 battery modules in parallel using i-BMS15 system

Each and every i-BMS15s monitoring and controlling respective modules in a battery pack perform internal synchronization via dedicated CAN messages. For this internal communication, a block of 24 CAN IDs is reserved. Since all i-BMS15s in a battery pack have information about all other paralleled modules, no “master” is needed in this system.

- Request load mode

When a battery pack receives a request for load mode, the i-BMS15 controlling & monitoring the module with the highest voltage will close its contactors, while all other i-BMS15s will monitor the pack “load” voltage. When the “load” voltage reaches the level of the next module voltage, the i-BMS15 controlling & monitoring this module will also close its contactors. A dead band of OCV (for e.g. +/- 1 V) as well as terminal voltage (for e.g. +/- 2 V) can be configured by the user for the packs to connect to the paralleled system. This process continues until all modules in the pack are connected to the load.

- Request charge mode

When a battery pack receives a request for charge mode, the i-BMS15 controlling & monitoring the module with the lowest voltage will close its contactors, while all other i-BMS15s will monitor the pack “charge” voltage. When the “charge” voltage reaches the level of the next module voltage, the i-BMS15 controlling & monitoring this module will also close its contactors. A dead band of OCV (for e.g. +/- 1 V) as well as terminal voltage (for e.g. +/- 2 V) can be configured by the user for the packs to connect to the paralleled system while charging. This process continues until all modules in the pack are connected to the charger.

These procedures ensure safe coupling of the pack but also ensures all modules are balanced on pack level. Thus, each i-BMS15 in such a system not only has full control over the battery modules monitored by them, but each of the i-BMS15’s in such a system is also operating autonomously as a battery pack controller.

10.14 General Purpose Input Output (GPIOs)

The i-BMS15 accommodates 4 independent open drain General Purpose Outputs (GPOs). The specifications being : Max. Drain Voltage of 36 V (Max. 30 V) and Max. Guaranteed Continuous Drain Current of 0.7 A (Max. 0.5 A).

In order to understand the GPO functionality, let us consider an example (Refer Figure 26). The GPO1 functionality is being used in thermal management of the battery pack in an EV application. The switching is controlled by dedicated transistors. Since the temperature measurements are performed by the auxiliary temperature sensors and the same is being reported to the BMS. Based on certain conditions such as : If the temperature of the pack exceeds 45^{0}C, the BTMS should cool the battery pack and if the temperature of the battery pack goes below -10^{0}C, the BTMS should switch on the heating element. If one of the above conditions are met, the BMS will switch on/off the respective GPO and ultimately switch on/off the thermal management system.

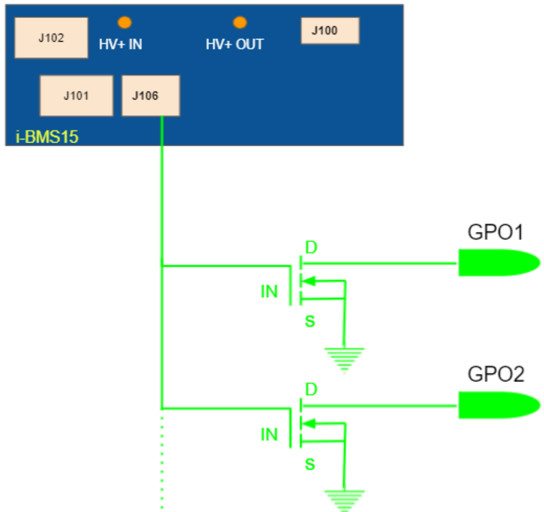


Figure 26 : GPO Circuit

On the i-BMS15 boards, the temperature channel inputs (6 NTC Inputs) can be used as General Purpose Inputs (GPI) depending on the use case and application. The General Purpose Inputs can also be enabled by i-BMS15 memory ID locations. In order to understand the configuration process for enabling the General Purpose Inputs using both the above mentioned methods, refer to Section 11.10.12.

11 BMS Creator

The Lithium Balance PC tool, BMS CREATOR (TM), is used to configure, boot load and service the i-BMS15. It is important to note that the BMS Creator only supports CAN adapters from PEAK SYSTEMS available from www.peak-system.com. The recommended models are

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

11.1 Requirements

- Windows 10, Windows 8, Windows 7 or Windows Vista
- .NET 4.5 framework (or latest)
- 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.

11.2 Connecting the PC to the BMS

The points (a,b,c..) have been marked (in red square boxes) in Figure 28. Refer to Figure 28 while going through each point in order to get a better understanding.

a) In order to connect the i-BMS15 to your PC, plug in the PEAK CAN adapter (Refer Figure 27). Once this is done, open BMS Creator which has been already installed on your PC. By default the first page of the BMS Creator configuration software will be the 'Connection' tab.

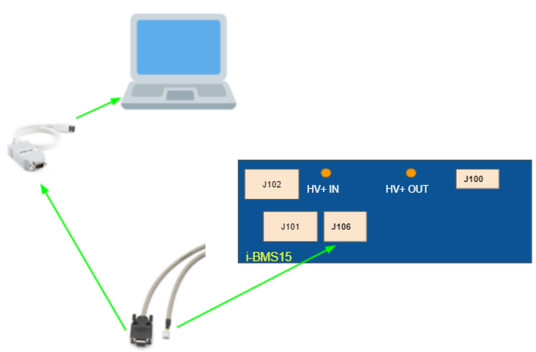


Figure 27 : Hardware Connection between PC and i-BMS15

Communication between BMS Creator on a PC and the i-BMS15 is typically needed in different situations like :

- During upload of firmware or configurations.
- During design of a battery system where fine tuning of BMS configurations or troubleshooting is needed.

The following are the steps in which the PC communicates with the BMS (Refer Figure 27):

- On the PC side, BMS Creator is using a USB driver for communication with the PEAK CAN adapter.
- The PEAK CAN adapter converts the PC USB communication to CAN communication with the help of DP9 interface.
- On the i-BMS15 side, CAN communication is used for all external communication. i-BMS15 communication is different depending on the following modes of the i-BMS15 :
  - Bootloader Mode : If a valid firmware or configuration is not available in the i-BMS15, it will by default enter into bootloader mode. The i-BMS15 must be in bootloader mode for the user to be able to upload new firmware or new configuration. When in bootloader mode, the communication speed on the CAN bus is fixed and other communications which might happen from the PC side (Apps like PCAN-View) or from ECUs (charger/VCU/Display) at different communication speeds will prevent the upload from succeeding. In that case, the user must disconnect all other devices and redo the process.
  - Application Mode : When in Application mode, the BMS Creator can send a CAN message to i-BMS15 and command it to enter into bootloader mode in case new firmware or configuration needs to be uploaded. In Application mode, the BMS Creator can adapt to the CAN channel communication speed and can co-exist with other communications like application data from PCAN-View or communications with other ECUs (Charger/VCU/Display).

b) 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.

c) By 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, and the same can be selected by the user.

d) 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.

e) 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.

f) The BMS Mode status textbox will display the state of the BMS at the point of connection. Note that the BMS state only changes when the ‘Connect’ button, ‘Upload BMS configuration’, ‘Reset BMS’, ‘Boot load BMS’ or ‘Read error log’ button is pressed.

g) 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.

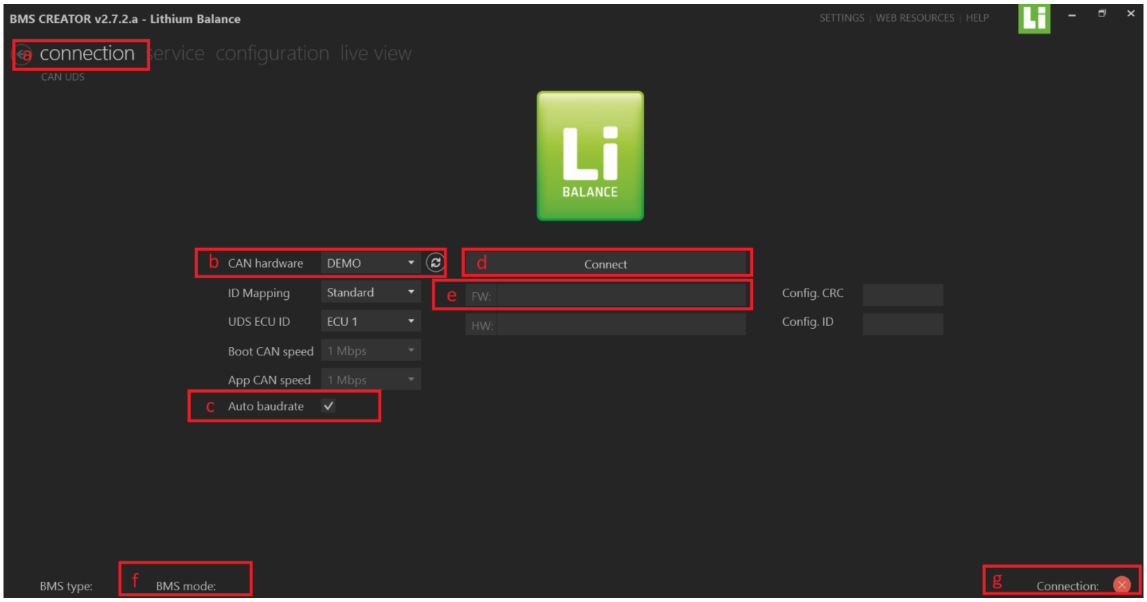


Figure 28 : Guide to connecting the PC to the BMS

11.3 Boot loading the BMS

a) The service/UPGRADE tab contains the interface to the BMS Boot load feature.

b) 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 finishing the upload process.

c) 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.

d) When the boot loading sequence is done there will be a small white check on the 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.

e) Refer to Section 11.6.

f) 'Reset BMS' commands the BMS to restart. This process is completed in less than 1 second. With this command, the up-time (time since the last boot) of the BMS is also reset to 0.

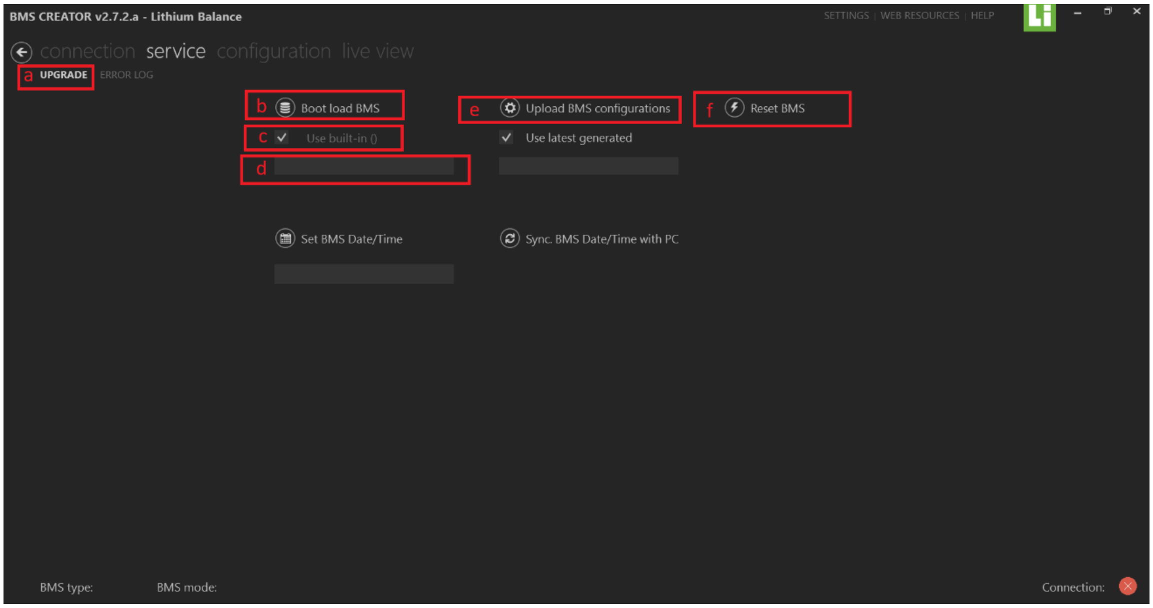


Figure 29 : Guide to Boot Loading (Service) the BMS

11.4 Error Log Readout

a) The service/ERROR LOG page contains functionality to read out a history of logged error in the system.

b) 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 on the same page.

c) 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.

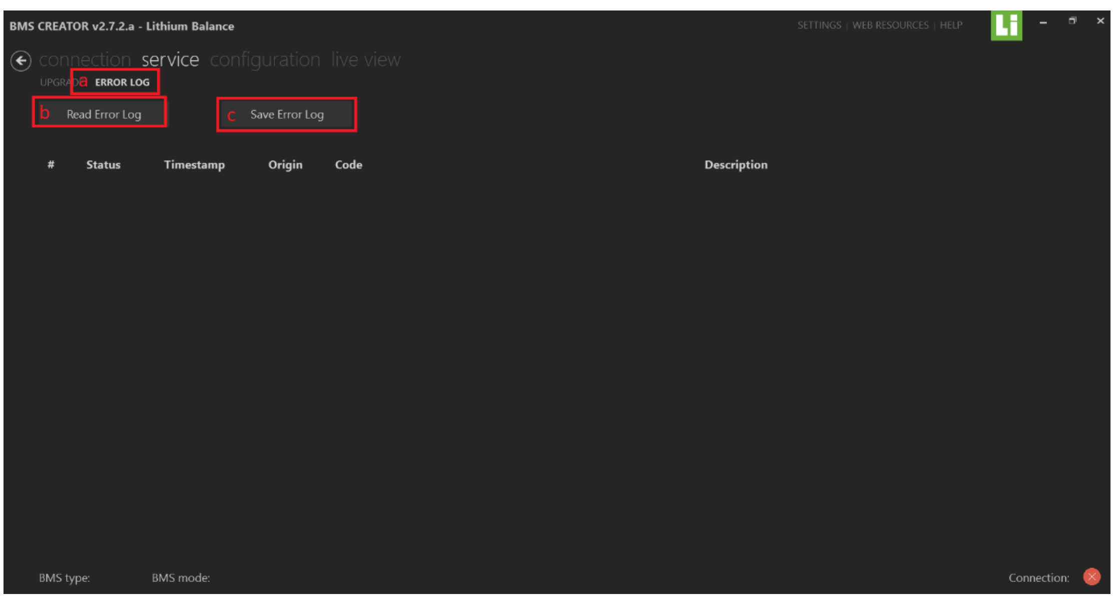


Figure 30 : Guide to Error Log Readout

11.5 Configuring the BMS

Warning : The user should always evaluate the Errors tabs for correct BMS action in case of any application critical error conditions. (Refer to Appendix 3)

a) 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 therefore the first step is to open a XML file.

b) If no file exists, one can be created by pressing the 'New' button.

c) If one already exists, press the 'Open' button, and a file dialog will allow selection of already existing file.

d) When done changing the XML file, the 'Save' button can be used to open a dialog of where to save the file.

e) 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 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.

f) 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.

g) After generation of the configuration file, a small indication is given on the right of the 'Generate BMS configuration' button.

Important Note : Refer to Appendix 4 for an overview and short explanation of the different parameters that can be configured.

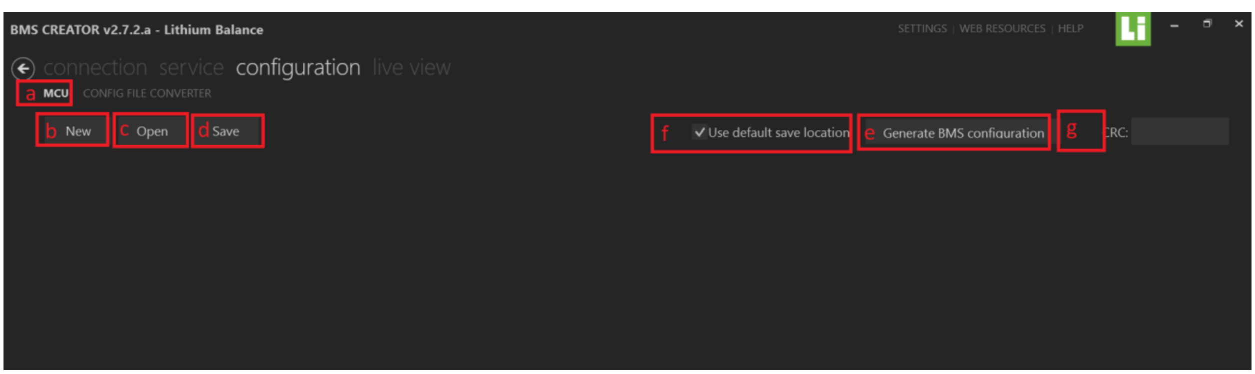


Figure 31 : Guide to start BMS Configuration

11.6 Upload BMS Configurations

a) The last step of the BMS configuration process is to navigate to the service/UPGRADE page and press the 'Upload BMS configuration'.

b) 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.

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

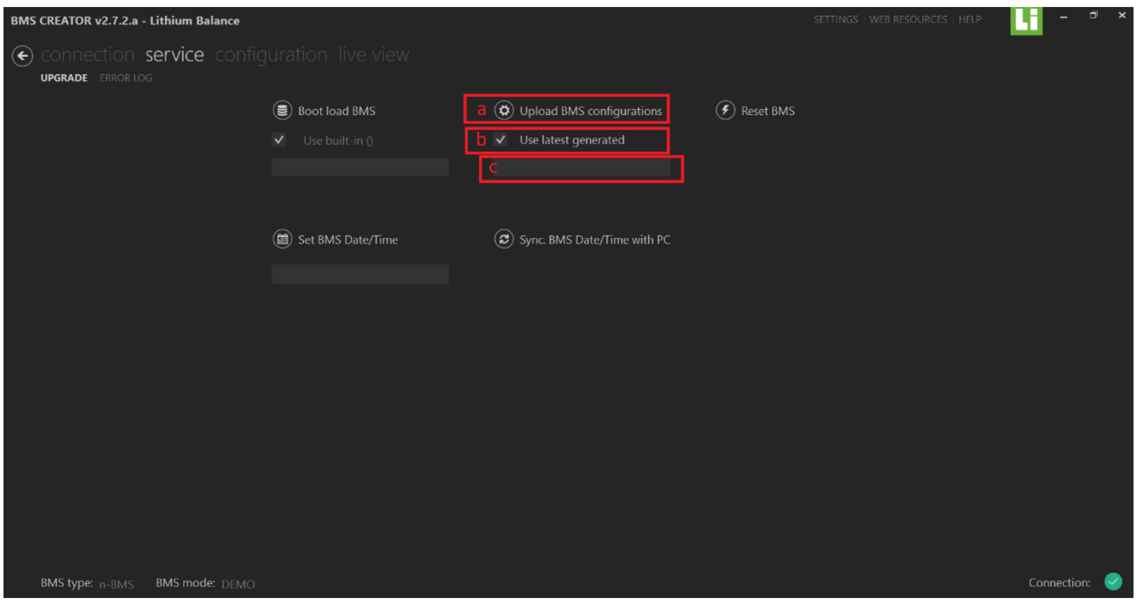


Figure 32 : Guide to Uploading BMS Configuration Files

11.7 Config File Converter

a) 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 of the same BMS platform.

The input file path implies the configuration file which needs to be converted into a compatible file for the latest release. A dialog will appear allowing the user to select the previous configuration.

b) The output file path implies the destination to which the converted file will be saved.

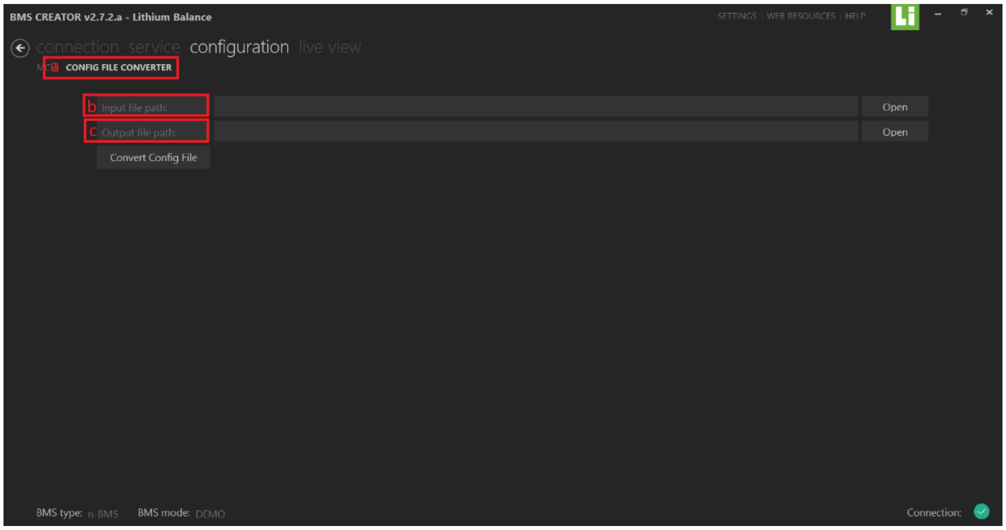


Figure 33 : Guide to convert configuration files

11.8 Live View

a) The live view tabs contain functionality to monitor and request BMS data.

b) The DASHBOARD page gives an overall indication of system status and displays the system current, voltage & temperature with the help of interactive graphs.

c) The MCU DATA page contains information regarding the master controller like alarms, internal measurements, and control data.

d) The CMU DATA page contains data regarding the cell measurement front-end like cell voltages and temperatures.

e) The CUSTOM page allows the user to select upto 8 built-in BMS parameters at a time and view the corresponding status/values of the same. This can be used by the user to build their own dashboard while troubleshooting a particular scenario/functionality.

f) 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 second interval. When pressing the 'start log' button, the user is prompted for a file save location and name.

Figure 34 : BMS Creator : Live View

11.9 Troubleshooting

- Not able to find your USB Adapter on Creator?

It is possible to choose the USB Adapter (in application) under the CAN Hardware dropdown field under the Connection pane of BMS Creator. If your USB is not listed, check whether the PEAK CAN adapter is connected and functional (red LED on the adapter). This problem can occur due to another situation where multiple BMS Creator applications are opened by the user at the same time. The user is recommended to close all tabs and try to re-establish connection.

- Communication not possible error after clicking ‘Connect’?

Once the CAN hardware (USB) has been selected and the user is ready to ‘Connect’, an error might pop up stating that communication with BMS is not possible. This problem typically arises when another program/app/software on the user’s PC has connected to this CAN channel. The solution proposed here is to restart the PC or BMS Creator (based on whichever solves the issue successfully), which will ultimately resolve this problem. Another possible issue can be that the CAN adapter (or communication cable) enters into failure mode. The solution proposed here is to replace the CAN Adapter/communication cable.

- Error in connection when trying to upload new configuration?

In certain cases the user might see that the communication is working perfectly fine and datasets are available in the ‘Live View’ pane of BMS Creator. But, as soon as the user tries to upload a new configuration, an error message pops up and the connection status changes to ‘Disconnected’. This can typically happen when another program like PCAN-View is connected or connects itself to the CAN channel.

11.10 Configuration Examples

11.10.1 Temperature Sensor Activation

The actual temperature inputs used by i-BMS15 needs to be activated in the BMS Creator. This is done by assigning one bit per sensor. If the input is used, set the bit to 1 and if it is not used, set the bit to 0.

For example, to activate temperature sensor 6, 3 and 2, the relevant binary number is 100110 (Refer Figure 35) 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 (Refer Figure 36).

Temperature Sensor No.	6	5	4	3	2	1
Sensors in Use	Yes	No	No	Yes	Yes	No
Corresponding Binary	1	0	0	1	1	0
Corresponding Decimal	38					

Figure 35 : Temperature Activation Guide

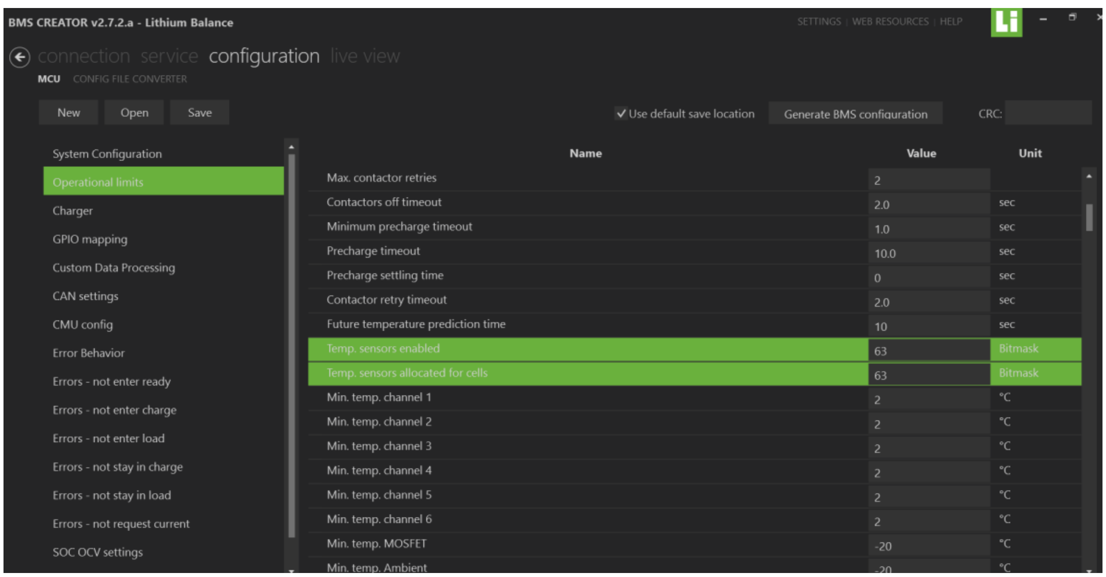


Figure 36 : BMS Creator : Temperature Sensor Activation

It is important for the user to understand the difference between 'Temp. sensors enabled' and 'Temp. sensors allocated for cells' parameters in BMS Creator. The user can define how many out of 6 available temperature sensors are assigned for cell temperature, and how many of them are not assigned for cell temperatures but assigned to other units, for example shunt or inverter temperature. If a temperature channel is allocated for cells, BMS will take the cell temperature settings into account when evaluating the temperature for that channel. Otherwise, the BMS will take "Min/Max. temp. channel 1-6" limits set by the user in BMS creator.

Important Note : It is necessary to have at least one cell temperature configured to be able to make use of features depending on cell temperatures, such as dynamic current limits, state of power estimation etc.

11.10.2 Voltage Sensor Activation

The i-BMS15 is engineered to measure a configurable number of cell voltages. This is done by enabling or disabling the corresponding cell number in BMS creator in CMU Config menu (Refer Figure 37). The i-BMS15 is designed to measure up to 15 cell voltages. The BMS can also work with less than 15 cells connected and therefore needs to be configured via the BMS Creator to know the actual number of cell voltages to read. The user can select the checkbox below the corresponding cell numbers depending upon the architecture of the particular battery pack.

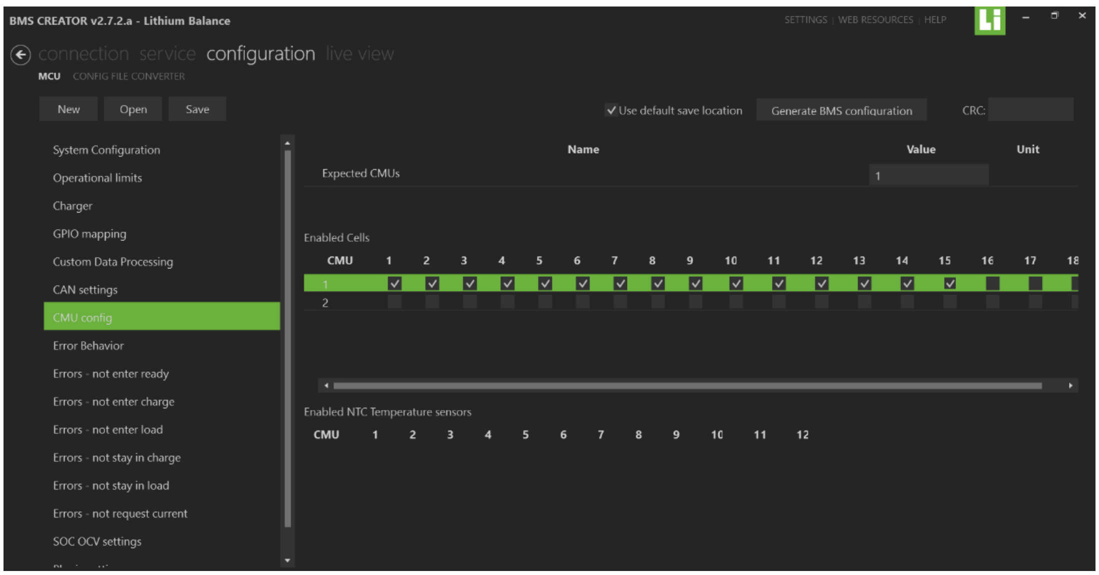


Figure 37 : BMS Creator : CMU Configuration (15 cells in series)

11.10.3 Charge Complete

The charging process will end when two 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.
- The measured current must be within the “charge complete deadband for current” from 0 A , e.g., +/- 100 mA from 0 A.

The same needs to be defined in the BMS Creator in the “Charger” pane (Refer Figure 38)

Once these two criteria are met, the BMS will stop the charge process and calibrate the remaining capacity to 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 0.1 Ah being discharged from the battery pack.

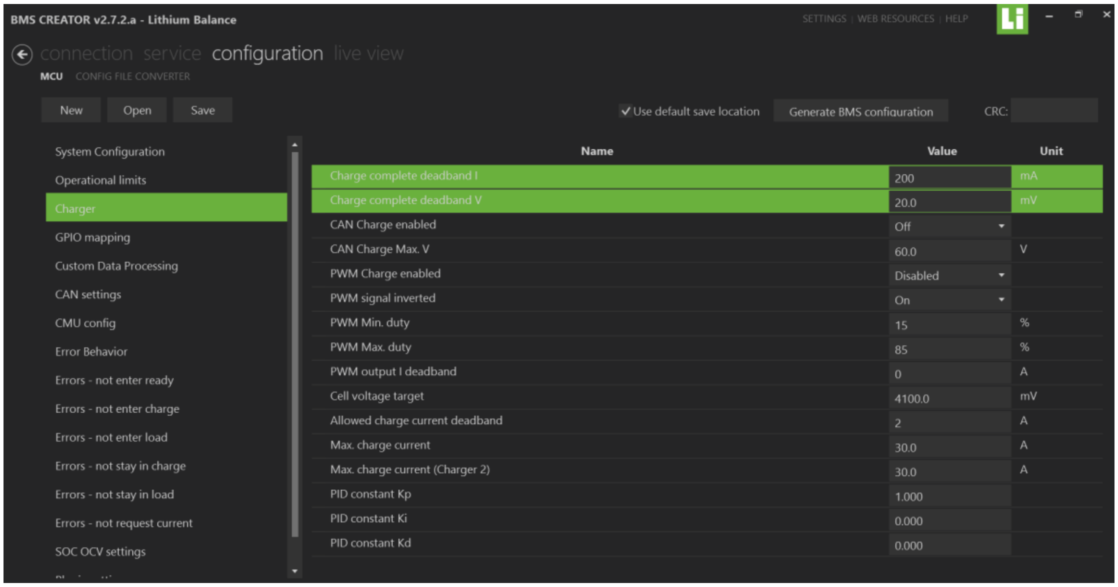


Figure 38 : BMS Creator : Charge Complete Parameters

11.10.4 Selecting PID Constants

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 depicted in Figure 39.

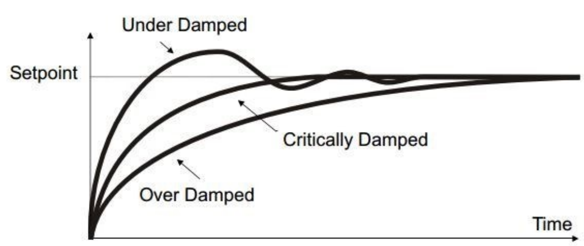


Figure 39 : PID Damping

By BMS Creator default settings, the PID controller is overdamped. In order to ensure the fastest charge response, it is advisable to tune the PID coefficients to achieve a critically damped response. The user should be aware that using underdamped responses will cause the cell voltages to increase beyond the target cell voltage, thus creating a potentially unsafe scenario.

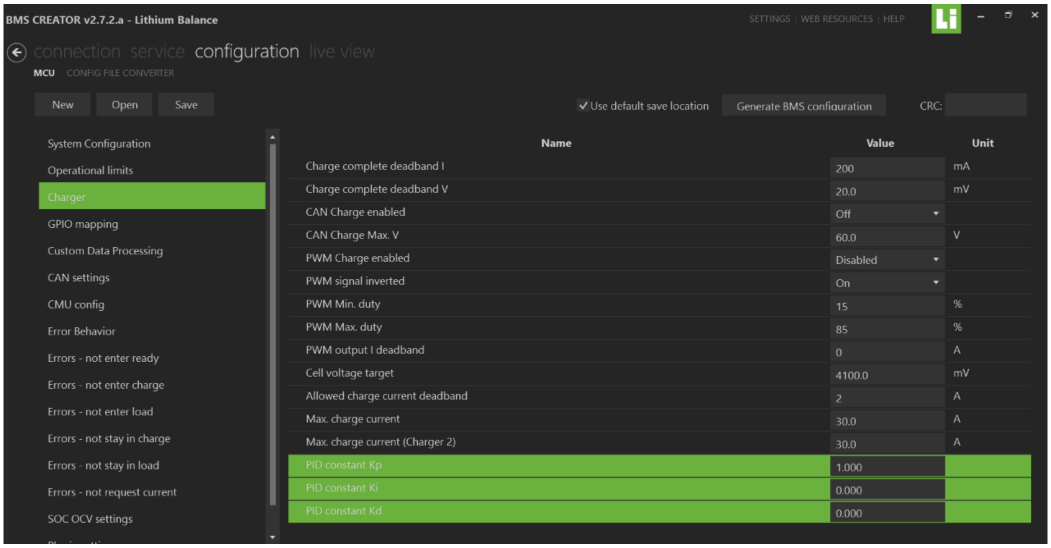


Figure 40 : BMS Creator : PID Constants Configuration

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.

Lithium Balance provides the user with an excel tool to obtain optimized values for Kp, Ki & Kd and the same can be made available to the user upon request.

11.10.5 Activation of Sleep Mode

The i-BMS15 implements a sleep mode function that can be activated via CAN. Figure 41 & Figure 42 below shows how the i-BMS15 can configure a CAN message that will request the BMS to power off.

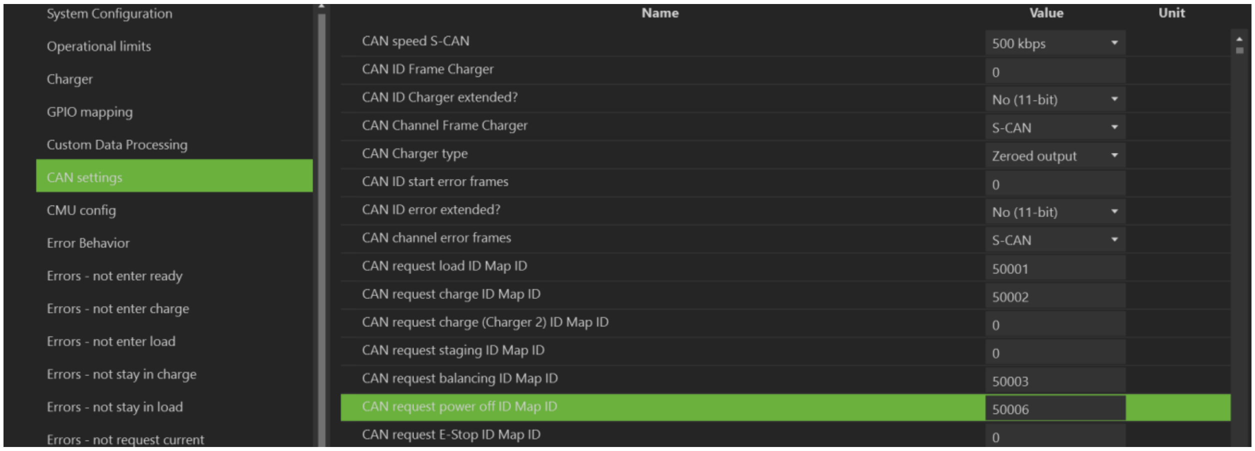


Figure 41 : BMS Creator : CAN Request Power-Off

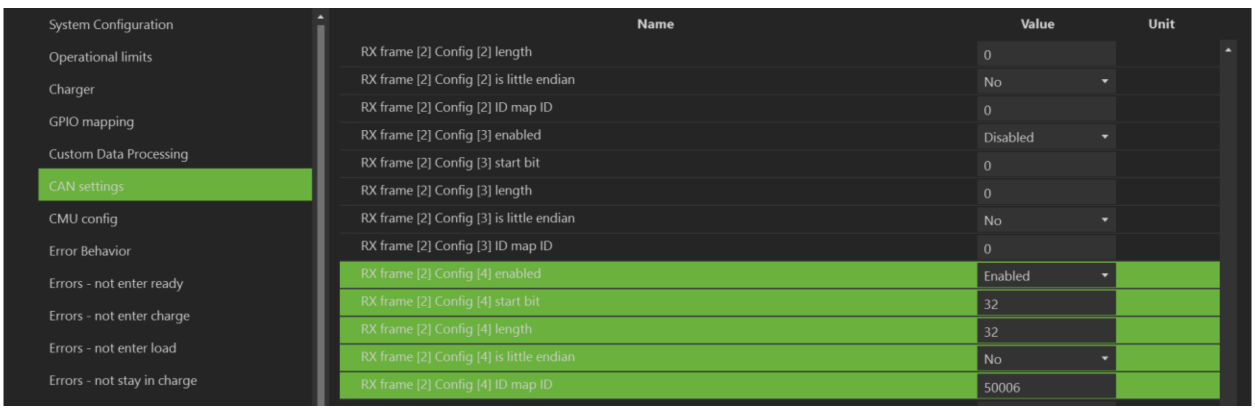


Figure 42 : BMS Creator : Example CAN RX Message for Power Off

Because the i-BMS15 has a CAN wake-up functionality it is important that no other communication is active on the CAN when the BMS is in sleep mode. The example above must be sent only once to request the i-BMS15 to power off.

11.10.6 Enabling E-Stop with CAN

In order to enable E-Stop, the user needs to first configure a CAN RX message to store the E-Stop status in a Data ID. The configuration below shows an example of how this can be done. (Refer Figures 43)

- Enable CAN Rx frame [1]
- The timeout interval 10 corresponds to 1 second meaning the message is expected every 1 second
- Define the length of the frame to be 8 bytes (DLC)
- Setup a CAN Rx frame with CAN ID 512
- Enable Configuration [1] of CAN Rx frame [1]
- Configuration [1] has start bit 32 in CAN Rx frame [1]
- Configuration [1] has length of 32 bits in CAN Rx frame [1]

- Configuration [1] of CAN Rx frame [1] will be stored in CAN Data ID 50005

The next step would be to enable the CAN E-Stop functionality by configuring the CAN Data ID (in this example 50005)

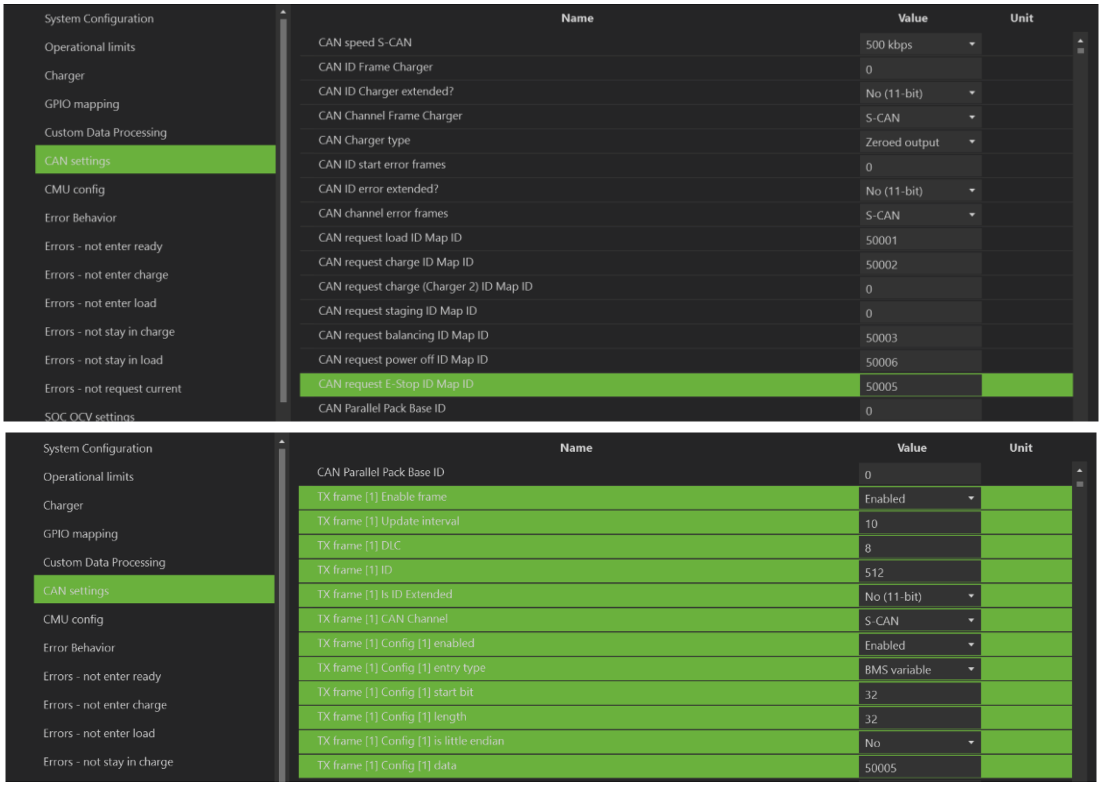


Figure 43 : BMS Creator : CAN RX Frame Configuration

In order to activate the CAN E-Stop functionality as configured above a CAN message must be sent to the iBMS CAN ID 512 where the value at bit 0 is set to 1. In PCAN View it would look like: (Refer Figure 44)

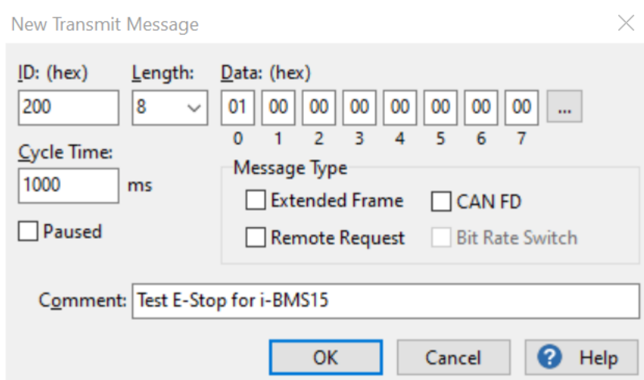


Figure 44 : PCAN-View Configuration for E-Stop

11.10.7 Configuring the windows for SOC Calibration

The hybrid SOC estimation algorithm requires some input from the user in order to ensure accurate results for the particular application. The user is required to identify the windows in the SOC-OCV curve (of the cells used in their application) and specify the quality corresponding to it (Refer to Figure 45). The quality factor here refers to the trust the user is willing to put on the SOC-OCV datasets in that particular window. This quality factor will depend upon the li-ion cell chemistry and the use case. These parameters are useful in calibration of SOC based on Coulomb Counting method with the SOC based on SOC-OCV curve. It is important to understand the following relationship in order to get the idea of quality factor :

$$\textbf{Quality}_{\textbf{CC}} + \textbf{Hybrid Calibration Sensitivity} > \textbf{Quality}_{\textbf{SOC-OCV}}$$

where,
Quality_{CC} = Accuracy of coulomb counter measured by the internal algorithm which increases as the number of charge/discharge cycles increase. It is important to remember that the accuracy will increase only if the calibration of SOC based on SOC-OCV curve does not happen. As soon as this calibration happens this accuracy of coulomb counter is reset to 0. In case of i-BMS15, generally, the accuracy of coulomb counter increases by 1% after one complete charge/discharge cycle.
$\text{Quality}_{SOC\text{-}OCV}$ = Quality specified by the User while configuration by taking into account the SOC-OCV data of the li-ion cell used
Hybrid Calibration Sensitivity = A factor to exaggerate the accuracy of coulomb counter when compared with the user configured accuracy of SOC-OCV window.

The calibration of SOC based on coulomb counter with SOC based on li-on cell SOC-OCV data happens only when the above mentioned condition is satisfied. Once the calibration is done, the accuracy of coulomb counter (Quality_{CC}) is reset to 0. The SOC windows and the quality for each one of the window has been configured in BMS Creator for demonstration purpose (for LFP chemistry) : (Refer to Figure 46)

- Window 1 (w1) : w1 has been configured as 0 to 5% SOC.
- Quality 1 (q1) : q1 has been configured as 20% which means that ± 10% is the expected accuracy of the SOC based on SOC-OCV curve.
- Window 2 (w2) : w2 has been configured as 5% to 15% SOC.
- Quality 2 (q2) : q2 has been configured as 5% which means that ±2.5% is the expected accuracy of the SOC based on SOC-OCV curve.
- Window 3 (w3) : w3 has been configured as 15% to 93% SOC.
- Quality 3 (q3) : q3 has been configured as 50% which means that ±25% is the expected accuracy of the SOC based on SOC-OCV curve. This implies that the trust on the SOC based on the SOC-OCV curve is poor in this window as the curve is more or less flat in this region.
- Window 4 (w4) : w4 has been configured as 93% to 98% SOC.
- Quality 4 (q4) : q3 has been configured as 5% which means that ±2.5% is the expected accuracy of the SOC based on SOC-OCV curve.
- Window 5 (w5) : Is any SOC value from the last window (as configured by the user) to 100% SOC and therefore, the user need not configure this window.
- Quality 5 (q5) : q5 has been configured as 20% which means that ± 10% is the expected accuracy of the SOC based on SOC-OCV curve

Figure 45 : SOC-OCV Windows & corresponding Quality Factor

Name	Value	Unit
Data set [5] voltage [96]	0	mV
Data set [5] voltage [97]	0	mV
Data set [5] voltage [98]	0	mV
Data set [5] voltage [99]	0	mV
Data set [5] voltage [100]	0	mV
Hybrid SOC w1	5.0	
Hybrid SOC q1	20.0	
Hybrid SOC w2	15.0	
Hybrid SOC q2	5.0	
Hybrid SOC w3	93.0	
Hybrid SOC q3	50.0	
Hybrid SOC w4	98.0	
Hybrid SOC q4	5.0	
Hybrid SOC q5	20.0	
Hybrid SOC Calibration Sensitivity	5.0	
Initial Pack Resistance	13.0	mOhms
EOL Resistance Factor	2.0	

Figure 46 : BMS Creator : SOC Windows and Quality factor Configuration

11.10.8 Configuring the parameters for SOH_C Estimation

SOH_C implies the contribution of capacity fade to the overall SOH estimation. The algorithm to calculate the SOH_C parameter requires input from the user which can be configured in BMS Creator. The parameters (Refer to Figure 48) involved in the SOH_C calculation can be understood as below :

- **Minimum OCV Accuracy for SOH_C :** In order for the algorithm to start measuring the SOH_C, the quality (accuracy) of each SOC window must be below than the quality (accuracy) configured here.
- **Minimum Delta SOC for SOH_C :** It is the delta value for which the SOH_C will be calculated. This implies that the difference between two measurement points (a and b) for SOH_C calculation can be configured via this parameter.

- **Maximum Delta Temperature for SOH_C :** This parameter specifies the maximum allowed difference in temperature of the cells while measuring SOH_C between the points 'a' and 'b'.

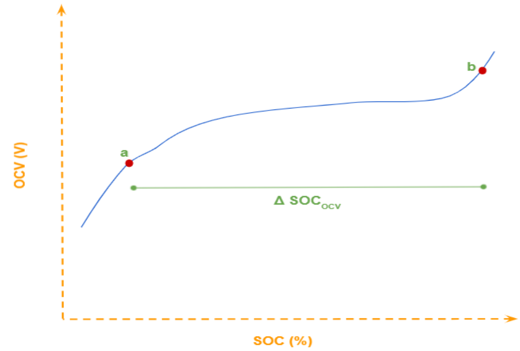
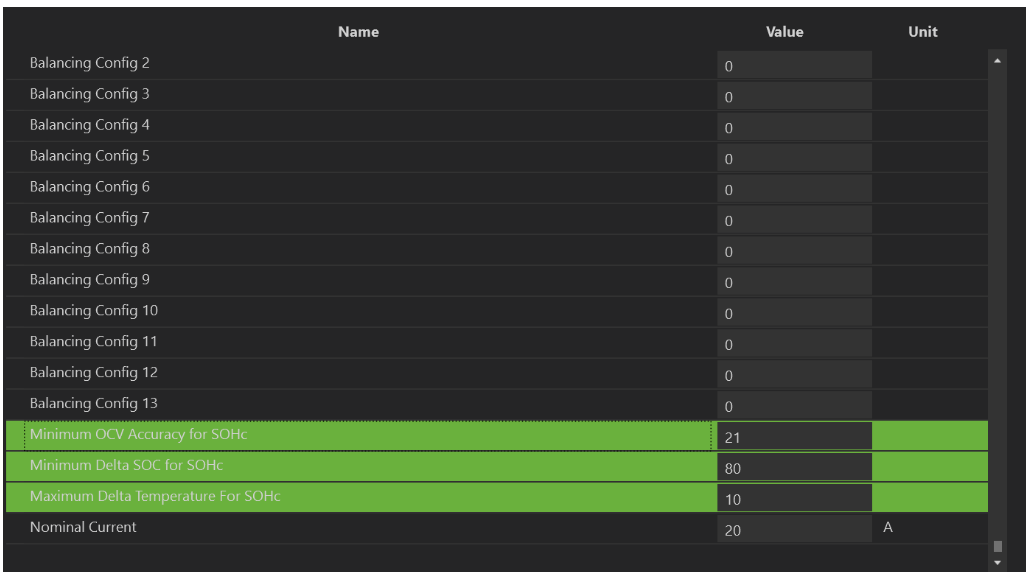


Figure 47 : Parameters for SOHc calculation



Name	Value	Unit
Balancing Config 2	0	
Balancing Config 3	0	
Balancing Config 4	0	
Balancing Config 5	0	
Balancing Config 6	0	
Balancing Config 7	0	
Balancing Config 8	0	
Balancing Config 9	0	
Balancing Config 10	0	
Balancing Config 11	0	
Balancing Config 12	0	
Balancing Config 13	0	
Minimum OCV Accuracy for SOHc	21	
Minimum Delta SOC for SOHc	80	
Maximum Delta Temperature For SOHc	10	
Nominal Current	20	A

Figure 48 : BMS Creator : Configuration Parameters for SOHC Calculation

11.10.9 Enabling Balancing based on SOC

The user is required to configure the following parameters in order to enable balancing based on SOC : (Refer to Figure 49)

- **Balancing Config 0 :** It is used to enable/disable balancing based on SOC. '0' can be used for disabling and '1' can be used for enabling.
- **Balancing Config 1 :** It is the lower limit for updating the balancing target. This implies that if we input 10 here, the balancing target will be allowed to update between 0% and 10% SOC. If we input 0 here, it will mean that this configuration has been disabled.
- **Balancing Config 2 & 3 :** It helps the user to provide a range of SOC values for updating the balancing target. This implies that if we input Balancing Config 2 as 20 and Balancing Config 3 as 50, the balancing target will be allowed to update between 20% and 50% SOC. If we input 0 in both Balancing Config 2 & 3, it will mean that this configuration has been disabled.

- **Balancing Config 4 :** It is the upper limit for updating the balancing target. This implies that if we input 75 here, the balancing target will be allowed to update between 75% and 100% SOC. If we input 0 here, it will mean that this configuration has been disabled.
- **Balancing Config 5 (Charge)** : It is the lower limit for allowing the balancing based on SOC during charging. This implies that if we input 12% here, the balancing based on SOC will be allowed between 0% and 12% SOC. If we input 0 here, it will mean that this configuration has been disabled.
- **Balancing Config 6 & 7 (Charge) :** It helps the user to provide a range of values for allowing balancing based on SOC during charging. This implies that if we input Balancing Config 6 as 30 and Balancing Config 7 as 55, the balancing based on SOC will be allowed between 30% and 55% SOC. If we input 0 in both Balancing Config 6 & 7, it will mean that this configuration has been disabled.
- **Balancing Config 8 (Charge) :** It is the upper limit for allowing the balancing based on SOC during charging. This implies that if we input 80% here, the balancing based on SOC will be allowed between 80% and 100% SOC. If we input 0 here, it will mean that this configuration has been disabled.
- **Balancing Config 9 (Discharge) :** It is the lower limit for allowing the balancing based on SOC during discharging. This implies that if we input 12% here, the balancing based on SOC will be allowed between 0% and 12% SOC. If we input 0 here, it will mean that this configuration has been disabled.
- **Balancing Config 10 & 11 (Discharge) :** It helps the user to provide a range of values for allowing balancing based on SOC during discharging. This implies that if we input Balancing Config 10 as 30 and Balancing Config 11 as 55, the balancing based on SOC will be allowed between 30% and 55% SOC. If we input 0 in both Balancing Config 10 & 11, it will mean that this configuration has been disabled.
- **Balancing Config 12 (Discharge) :** It is the upper limit for allowing the balancing based on SOC during discharging. This implies that if we input 80% here, the balancing based on SOC will be allowed between 80% and 100% SOC. If we input 0 here, it will mean that this configuration has been disabled.
- **Balancing Config 13 :** It is the quality threshold for disabling balancing based on SOC. It is expressed in Ampere-seconds (A-s). This implies that if we input 20 A-s here, it will mean that if the difference between the average SOC of all cells and the SOC of the particular cell is less than 20 A-s, balancing will be stopped.

Name	Value	Unit
SOHC Update Ratio	1	
Balancing Config 0	0	
Balancing Config 1	0	
Balancing Config 2	0	
Balancing Config 3	0	
Balancing Config 4	0	
Balancing Config 5	0	
Balancing Config 6	0	
Balancing Config 7	0	
Balancing Config 8	0	
Balancing Config 9	0	
Balancing Config 10	0	
Balancing Config 11	0	
Balancing Config 12	0	
Balancing Config 13	0	
Minimum OCV Accuracy for SOHc	21	
Minimum Delta SOC for SOHc	80	

Figure 49 : BMS Creator : Configuring Balancing based on SOC

Let us take up a scenario here, in order to understand the configurations for balancing based on SOC. If the user wants the following :

- Enable balancing based on SOC
- Balancing Target update between 15% and 32%
- Balancing allowed during charging between 0% and 12%
- Balancing allowed during discharging between 84% and 100%
- Quality Threshold for disabling balancing = 13 A-s

The user will use the following configurations for the above mentioned scenario :
Balancing Config 0 = 1
Balancing Config 1 = 0
Balancing Config 2 = 15
Balancing Config 3 = 32
Balancing Config 4 = 0
Balancing Config 5 = 12
Balancing Config 6 = 0
Balancing Config 7 = 0
Balancing Config 8 = 0
Balancing Config 9 = 0
Balancing Config 10 = 0
Balancing Config 11 = 0
Balancing Config 12 = 84
Balancing Config 13 = 13

11.10.10 Parallel Pack Configuration

Parallel packs module uses a block of CAN frame IDs for inter-pack communication. The block starting frame ID is configurable. The number of used frame IDs depends on the maximum number of parallel packs. Each pack takes 4 frame IDs, so for example when using 6 packs, the total frame IDs used is 4 * 6 = 24. Only pack data is sent using those frame IDs. There are no master command frames, as each pack decides whether it is safe to open contactors based on data from all other packs. In order to enable paralleling of battery packs, the following configurations must be performed in a step-by step manner to ensure smooth operation :

- **System Configuration -> Parallel Packs Enable** – Enable only if the parallel pack functionality is used in your application.
- **System Configuration -> Number of Parallel Packs** – The number of packs parallel packs module expected.
- **Operational Limits -> Parallel Pack Connection OCV Deadband** – The OCV deadband within which the paralleled packs will be connected to the system if they are imbalanced or not at the same voltage levels.
- **Operational Limits -> Parallel Pack Connection Ext. Voltage Deadband** – The terminal voltage deadband within which the paralleled packs will be connected to the system if they are imbalanced or not at the same voltage levels.
- **CAN Settings -> CAN ID Parallel Packs Base** – The base CAN frame ID that parallel packs module uses. The number of frame IDs used is the number of packs multiplied by 4. The used frame ID is extended CAN address (29-bit).

Important Note : The whole parallel pack frame ID block should not cross multiple of 256 value boundaries (values that multiply by 256) due to CAN receive mailbox filter settings. This has to be taken into account when choosing CAN ID base address. In other words, there cannot be a frame ID inside the block whose value is multiple of 256, except for the base address itself. For example, using 8192 as base is fine because it is multiple of 256 and it will not cross over the next multiple of 256 value boundary (which is 8192+256=8448). Choosing 8200 is also fine because the block will not have any values that are multiple of 256. But choosing a base at 8190 will not work because it crosses the 256 value boundary at 8192.

- **CAN Settings -> RX frame** – In order to enter charge or discharge mode via CAN, a RX frame must be configured with CAN request load and CAN request charge bits. This configuration must be equal across all packs so that all packs receive the commands.
  An example configuration is shown in Figures 50 & 51, where 50001 is configured to be request load ID and 50002 is configured to be request charge ID.

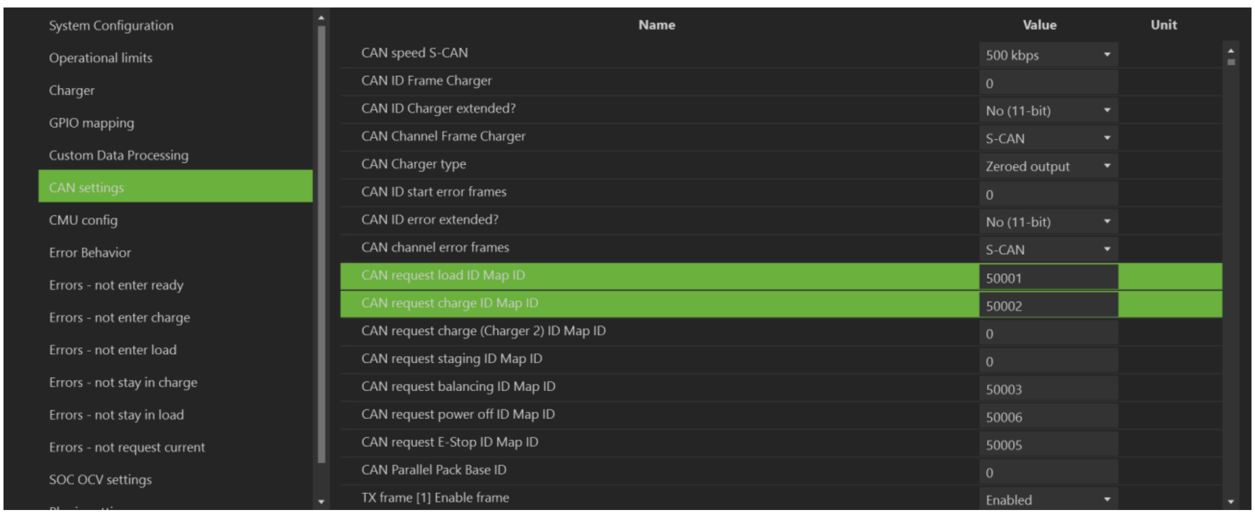


Figure 50 : BMS Creator : Mapping Data ID with CAN

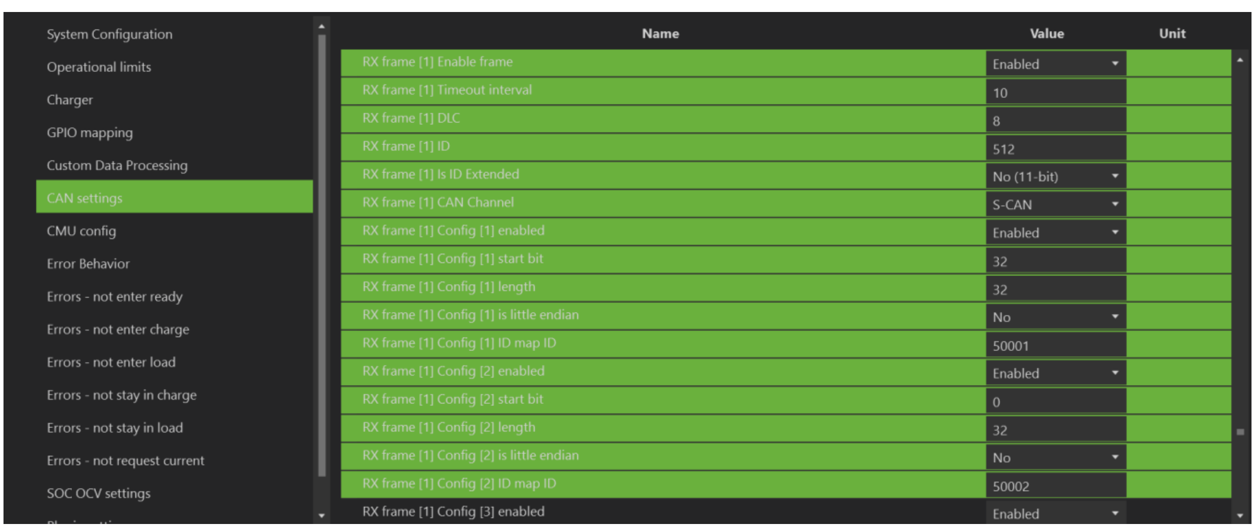


Figure 51 : BMS Creator : CAN RX Frame Configuration for Enabling Parallel Pack Functionality

- Assigning ECU IDs & Configuring Each Pack

Each pack in a configured parallel pack system uses a different UDS CAN frame ID which depends on its module ID. To access different packs and configure them differently, use the UDS ECU ID number in connection settings. ECU 1 refers to pack with module ID 1, ECU 2 to module ID 2 etc. ECU 0 refers to node ID 0 (module ID not initialized).

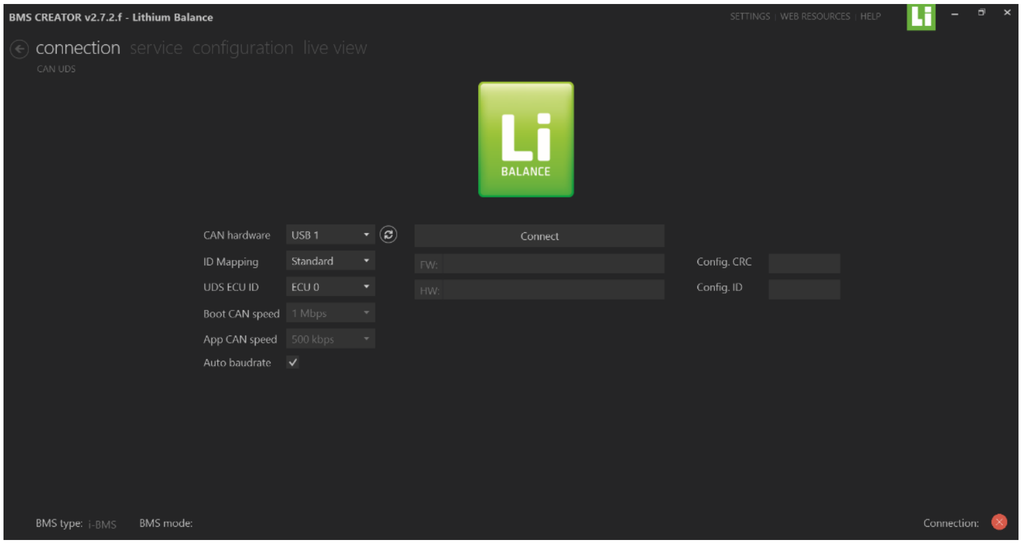


Figure 52 : BMS Creator : ECU ID Setting

Each pack in a parallel packs set must have a unique module ID. The used module IDs must be sequential, starting from 1. Module ID has to be written to a file stored on FRAM inside i-BMS15. This can be done in BMS Creator (psr_xcdp branch only). Please follow the bellow mentioned instructions to read/write ECU IDs and configure the paralleled packs :

a) Boot up BMS, make sure it is in firmware ("App mode").

b) In BMS Creator, for UDS ECU ID, select current module ID ("ECU 0" if module ID is uninitialized). (Refer Figure 53)

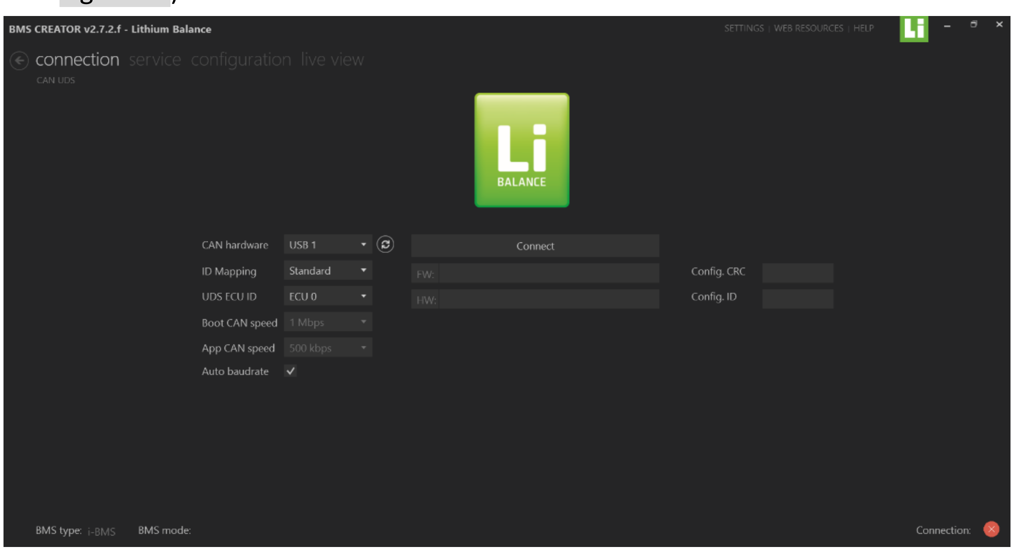


Figure 53 : BMS Creator : Using ECU ID 0 to connect the BMS

c) Click on “Connect” BMS creator

d) When BMS is connected, right-click on the “Connect” button. A popup “Command Window” window appears.

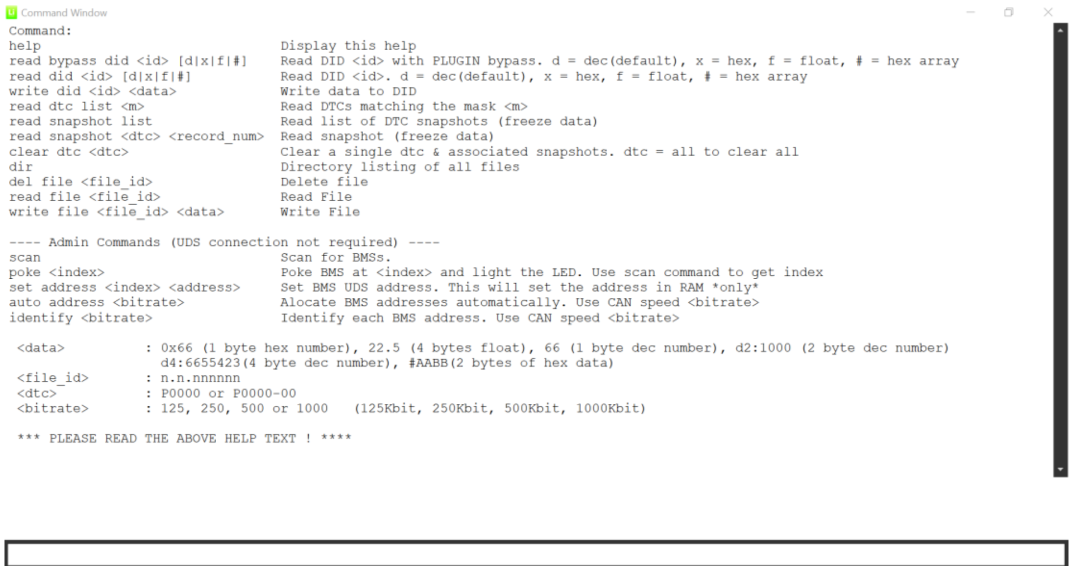


Figure 54 : BMS Creator : Command Window

e) To write module ID, write the following command into the Command Window and press “Enter”:

write id 3142 n
where, n is the actual module ID between 1 and 10.

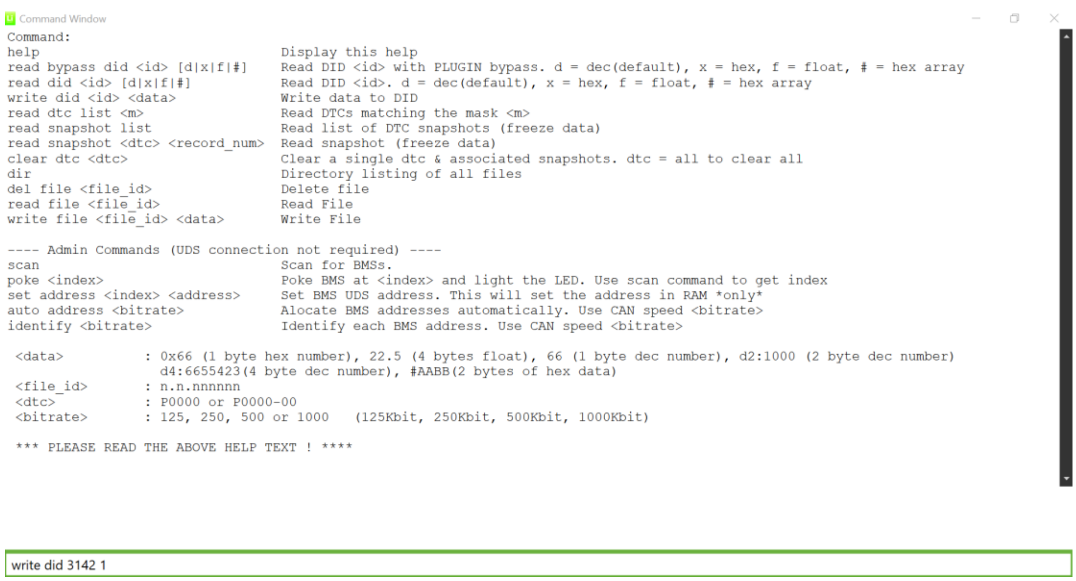


Figure 55 : BMS Creator : Command for Writing Module ID 1

f) The following message will appear on the command window and the Module ID should now be stored in BMS non-volatile memory.

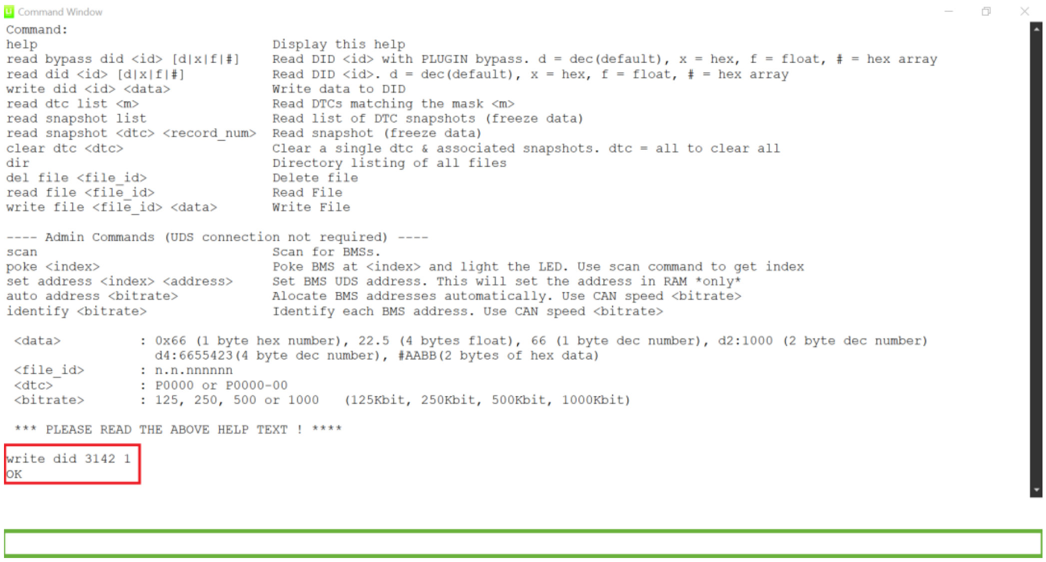


Figure 56 : Write ID Successful

g) Now ECU 1 can be connected via the BMS Creator default page. The user can configure the 1st pack equipped with i-BMS15 and upload configurations as well as firmware onto the BMS.

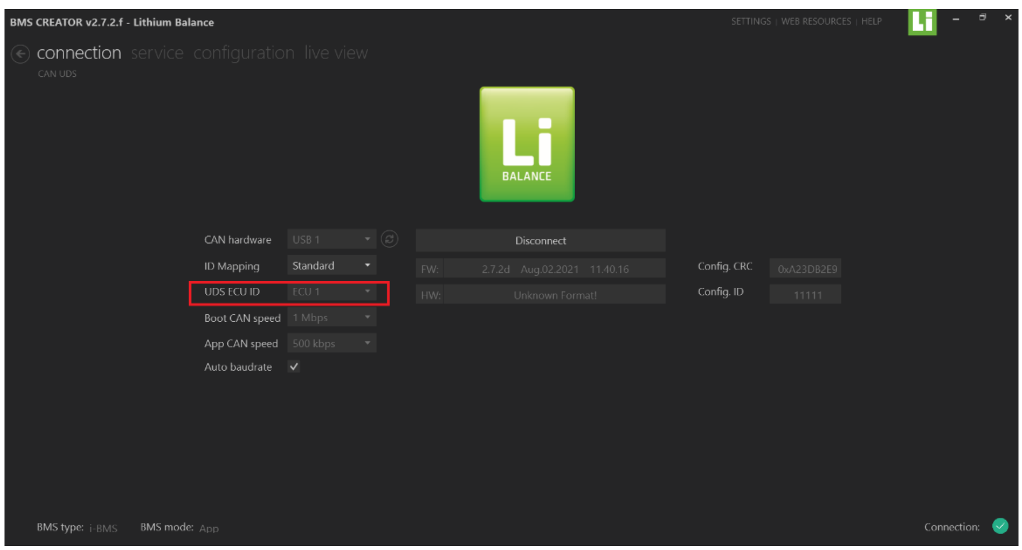


Figure 57 : Connecting and Configuring with ECU1

h) In order to read the data ID of the particular pack, the user can right click on “Disconnect” (after Connecting to ECU 1 in the previous step) and enter the following command in the command window and press “Enter”:

To read the current module ID number use the command
read did 3142 #

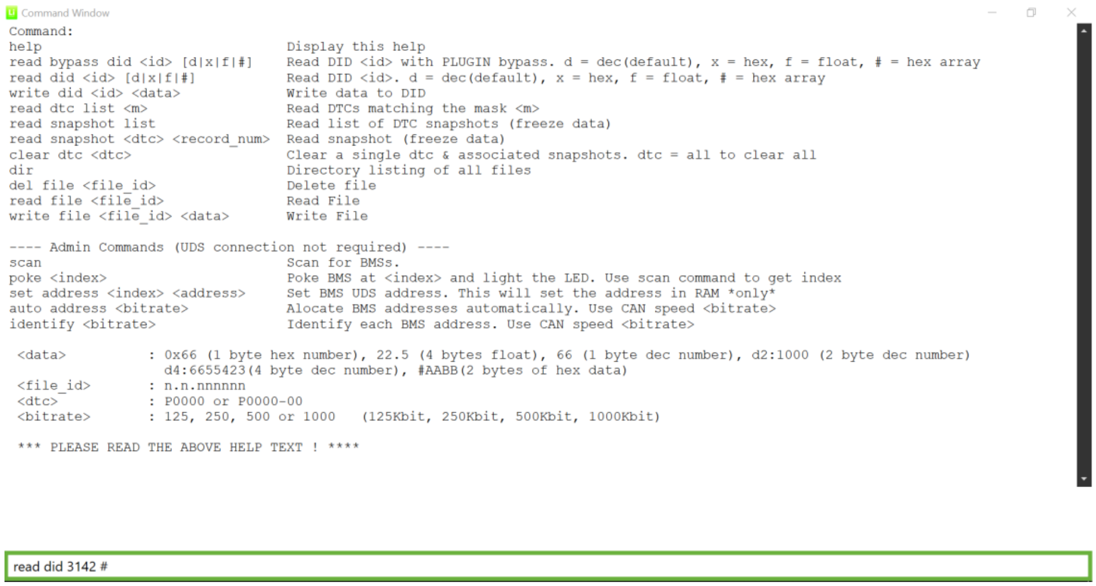


Figure 58 : Command to read data ID of the pack

i) The following message will appear in the command window after entering the read id command

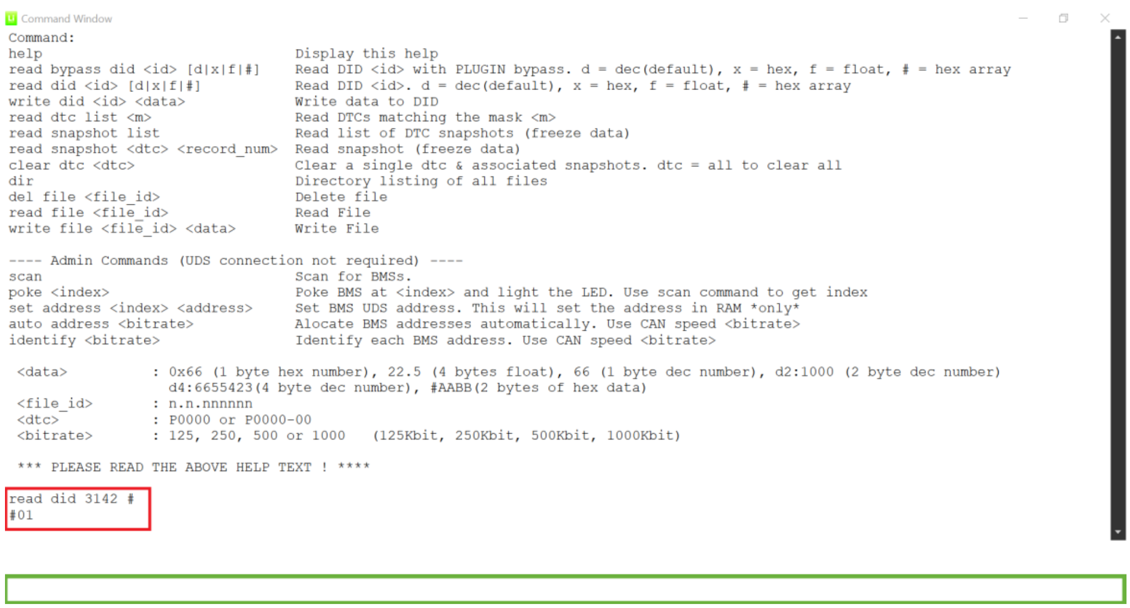
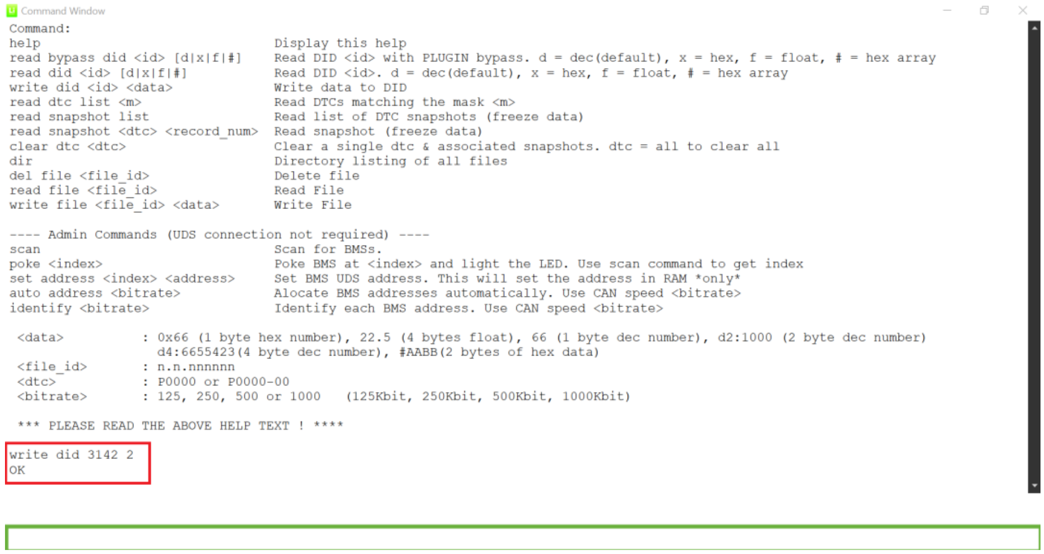


Figure 59 : Reading the data ID of the pack

j) If the user wants to configure the 2nd pack equipped with i-BMS15, the user can enter the following command by right clicking on “Disconnect” while the Creator is connected to ECU 1.



```
Command Window
Command:
help                            Display this help
read bypass did <id> [d|x|f|#]  Read DID <id> with PLUGIN bypass. d = dec(default), x = hex, f = float, # = hex array
read did <id> [d|x|f|#]         Read DID <id>. d = dec(default), x = hex, f = float, # = hex array
write did <id> <data>           Write data to DID
read dtc list <m>               Read DTCs matching the mask <m>
read snapshot list              Read list of DTC snapshots (freeze data)
read snapshot <dtc> <record_num> Read snapshot (freeze data)
clear dtc <dtc>                 Clear a single dtc & associated snapshots. dtc = all to clear all
dir                             Directory listing of all files
del file <file_id>              Delete file
read file <file_id>             Read File
write file <file_id> <data>     Write File

---- Admin Commands (UDS connection not required) ----
scan                            Scan for BMSs.
poke <index>                    Poke BMS at <index> and light the LED. Use scan command to get index
set address <index> <address>   Set BMS UDS address. This will set the address in RAM *only*
auto address <bitrate>          Alocate BMS addresses automatically. Use CAN speed <bitrate>
identify <bitrate>              Identify each BMS address. Use CAN speed <bitrate>

 <data>          : 0x66 (1 byte hex number), 22.5 (4 bytes float), 66 (1 byte dec number), d2:1000 (2 byte dec number)
                   d4:6655423(4 byte dec number), #AABB(2 bytes of hex data)
 <file_id>       : n.n.nnnnnn
 <dtc>           : P0000 or P0000-00
 <bitrate>       : 125, 250, 500 or 1000   (125Kbit, 250Kbit, 500Kbit, 1000Kbit)

 *** PLEASE READ THE ABOVE HELP TEXT ! ****

write did 3142 2
OK
```

Figure 60 : Writing data ID for the second pack

k) The use can now “Connect” to ECU 2 and perform configuration or upload firmware onto the BMS.

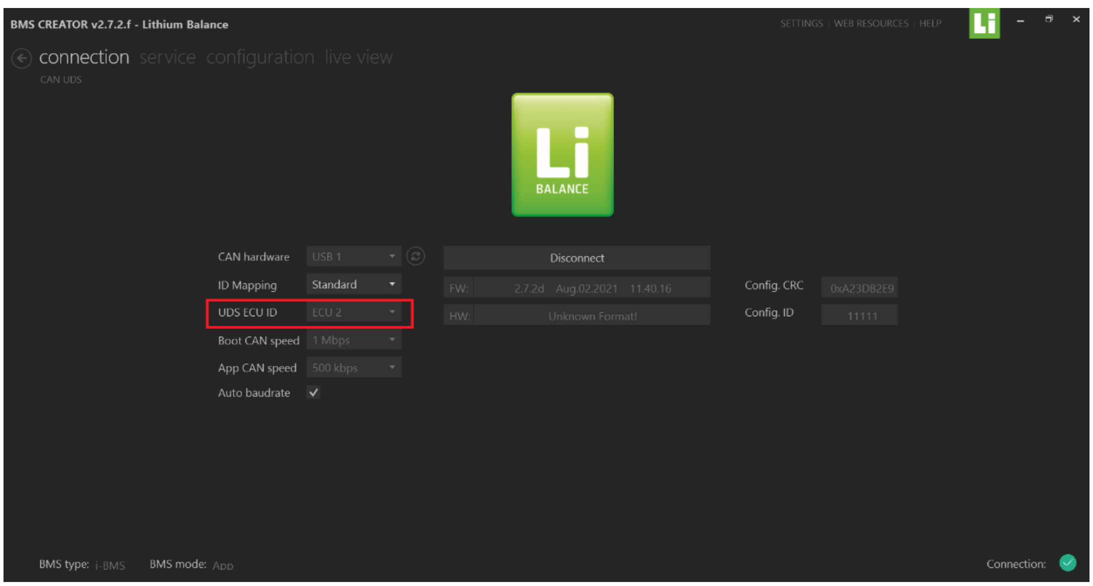


Figure 61 : Connecting to ECU 2

- Example for Paralleling while Charging

Let us take an example of 4 battery packs in parallel in order to understand the concept of parallelizing better. Suppose the individual pack OCV before charging are as following :

Pack 1 : 41 V
Pack 2 : 43.5 V
Pack 3 : 42.5 V
Pack 4 : 39.5 V
Parallel Pack Connection OCV Dead band : 1 V (Primary Connection Criteria)

Parallel Pack Connection Ext. Voltage Dead band : 2 V (Secondary Connection Criteria)

Important Note : It is important to note that both the criteria for connection (primary as well as secondary) should be satisfied in order for the BMS to ensure successful parallel connection.

As soon as charging is started, the pack with the lowest voltage (i.e. Pack 4) will be the first to connect itself to the charging system. As the charging process is continued and the OCV of the Pack 4 reaches > 40 V which is within dead band (configurable by the user), Pack 1 will be connected to the charging system. The system OCV when crosses 41.5 V, Pack 3 will be connected to the system. However, the secondary connection criteria i.e. terminal voltage dead band is also checked while parallelizing. Similarly, pack 2 gets connected to the system as the voltage crosses 42.5 V.

11.10.11 Configuring Error Handling

The parameters involved in configuration of Error Behaviors in BMS Creator are described below (Refer to Figures 62 & 63) :

- **Default Error Behavior Type :** The default error behavior that can be applied to all the error codes until and unless some errors are assigned to custom error behaviors configured by the user. For now, this field should be set to 0.
- **Trigger Level :** It is the upper limit set by the user for the counter. When the trigger level is reached , the error will be reported by the BMS.
- **Debounce Increment :** If a particular error is present consistently while sampling, the counter will count the value up with respect to debounce increment.
- **Debounce Decrement :** If a particular error is not present consistently while sampling, the counter will count the value down with respect to debounce decrement.
- **Heal Decrement :** Once the error is reported by the BMS (i.e. the trigger level is reached), the counter will count down to zero with respect to the heal decrement instead of the debounce decrement. This will enable the BMS to report the error for some filtered time configured by the user so that it is not erased immediately after being presented to the user.
- **Error Behavior (1-9) Type :** The user can create upto 9 custom error behaviors that can be assigned to specific errors depending upon the application. For now, this field should be set to 0.
- **Error Code Config (1-10) :** The user can assign upto 10 errors to the custom error behaviors and the corresponding error code shall be entered in this field.
- **Behavior Index :** It is the Error Behavior Type Number (1 to 9) to which the particular error code has been assigned.
- **DTC (Optional) :** The user can use this field in case of DTC codes available under UDS protocol.

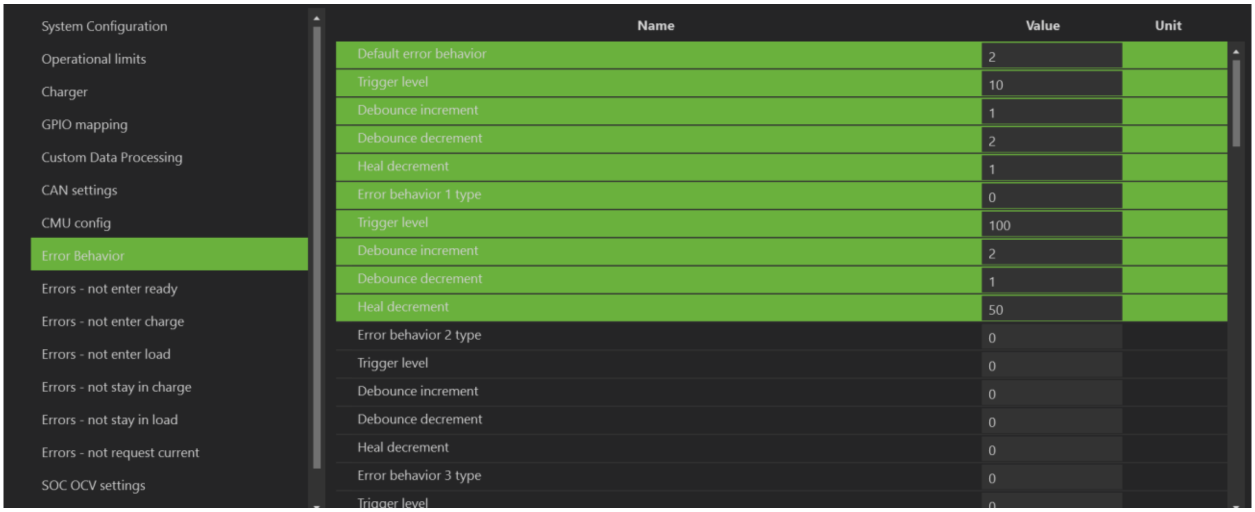


Figure 62 : BMS Creator : Default Error Behavior Type

Figure 63 : BMS Creator : Error Behavior (1-9) Type Configuration

In order to understand the parameters ‘trigger level’, ‘debounce increment’, ‘debounce decrement’ and ‘heal decrement’ better, let us consider an example. Let us assume that the error we are talking about here is related to temperature measurements (sampling frequency of 100ms) the user has defined :

- Trigger Level (TL) = 20
- Debounce Increment (DI) = 4
- Debounce Decrement (DD) = 8
- Heal Decrement (HD) = 2

Table 8 below depicts a typical error behavior scenario. The ‘debounce increment’ and ‘debounce decrement’ are used to count up or down the counter value but at the same time ensure filtering out errors arising for short durations so that they are not reported by the BMS. The error is not reported until the counter reaches the ‘trigger level’. After the ‘trigger level’ is reached, the error is reported until the counter value counts to 0 with respect to ‘heal decrement.’

Sampling	Error Log	Error Report	Counter Value	Remarks
100ms	No Error	Not Reported	0	Counter value remains at 0
200ms	Error	Not Reported	4	Counter value increased by a factor of 4 (DI)
300ms	Error	Not Reported	8	Counter value increased by a factor of 4 (DI)
400ms	Error	Not Reported	12	Counter value increased by a factor of 4 (DI)
500ms	Error	Not Reported	16	Counter value increased by a factor of 4 (DI)
600ms	No Error	Not Reported	8	Counter value decreased by a factor of 8 (DD)
700ms	No Error	Not Reported	0	Counter value decreased by a factor of 8 (DD)
800ms	Error	Not Reported	4	Counter value increased by a factor of 4 (DI)
900ms	Error	Not Reported	8	Counter value increased by a factor of 4 (DI)
1000ms	Error	Not Reported	12	Counter value increased by a factor of 4 (DI)
1100ms	Error	Not Reported	16	Counter value increased by a factor of 4 (DI)
1200ms	Error	Reported	**20**	1. Counter value increased by DI 2. Trigger Level is reached 3. BMS will report the error 4. The counter will now count down with respect to Heal Decrement 5. Error will be reported by the BMS until counter value is 0
1300ms	No Error	Reported	18	Counter Value decreased by a factor of 2 (HD)
1400ms	No Error	Reported	16	Counter Value decreased by a factor of 2 (HD)
1500ms	No Error	Reported	14	Counter Value decreased by a factor of 2 (HD)
1600ms	No Error	Reported	12	Counter Value decreased by a factor of 2 (HD)
1700ms	No Error	Reported	10	Counter Value decreased by a factor of 2 (HD)
1800ms	No Error	Reported	8	Counter Value decreased by a factor of 2 (HD)
1900ms	No Error	Reported	6	Counter Value decreased by a factor of 2 (HD)
2000ms	No Error	Reported	4	Counter Value decreased by a factor of 2 (HD)
2100ms	No Error	Reported	2	Counter Value decreased by a factor of 2 (HD)
2200ms	No Error	Not Reported	**0**	Error Removed

Table 8 : Error Behavior Configuration Example

11.10.12 Enabling General Purpose Inputs

The general purpose inputs in case of i-BMS15 can be configured to associate with either a temperature input channel or with a BMS memory location ID:

- Digital Input Configuration via Temperature Sensor Input Channel

The following steps can be followed in order to enable a General Purpose Input via the temperature sensor input channel. In the example case shown below, the temperature sensor channel number 3 is used as digital input for GPIO Input 8 :

a) The user needs to mention the temperature channel number for the corresponding GPIO Input number as shown in Figure 64.

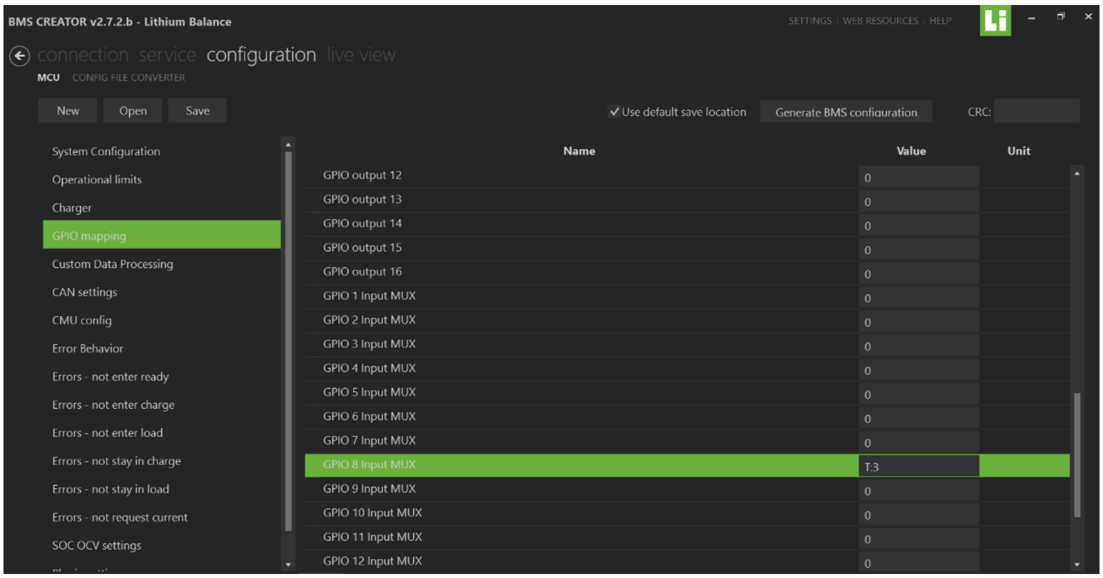


Figure 64 : BMS Creator : Configuring the temperature channel as digital input

b) With the above use of temperature sensor 3 as GPIO digital input, it should be disabled as temperature input as shown in Figure 65. The associated bit position for the particular temperature sensor is set as zero (decimal 59 ~ binary 111011).

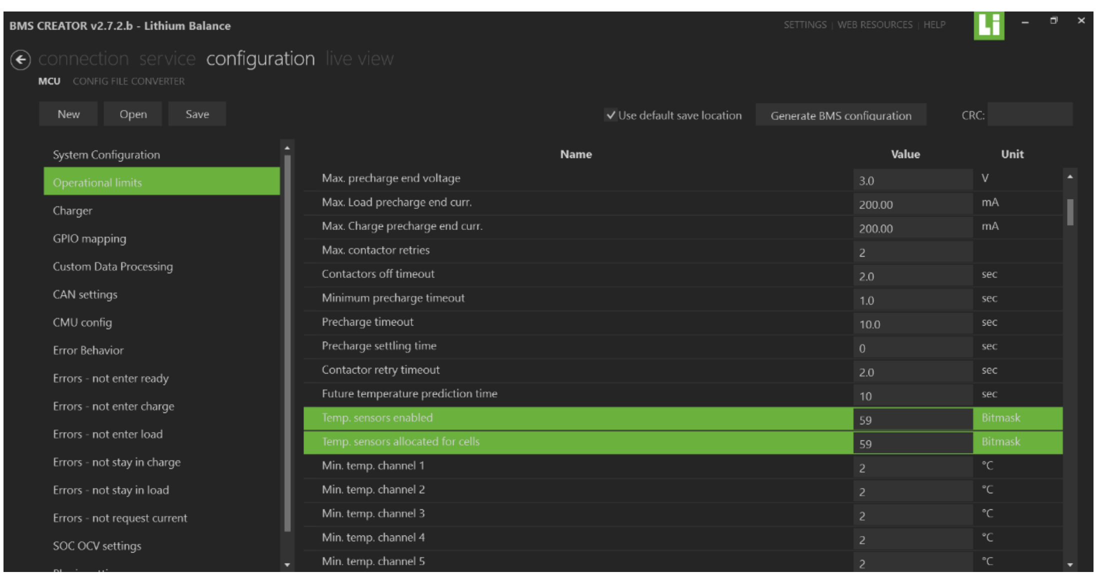


Figure 65 : BMS Creator : Disabling one of the temperature channels

c) Short circuit (grounding) of the temperature sensor input will activate the digital GPIO input port that can be verified in BMS Creator as shown in Figure 66 :

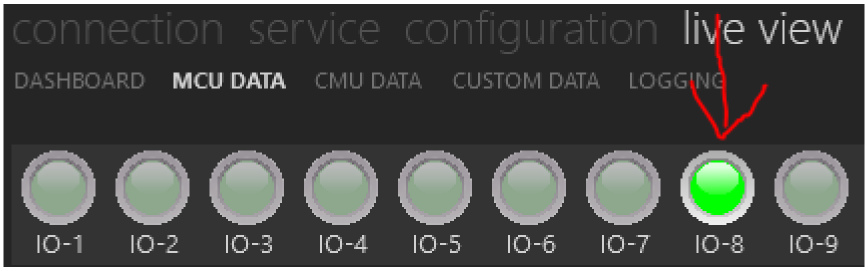


Figure 66 : GPIO Digital Input enabled via temperature sensor input

- Digital Input Configuration via BMS Memory Location ID

The i-BMS15 memory ID locations can also be used as GPIO Digital Inputs. The following steps will help the user to configure the GPIO Digital Input via BMS Memory ID locations. In this example CAN RX memory location ID 50007 is used as digital input for GPIO 1 :

a) The CAN RX ID 50007 needs to be configured in the CAN RX Frame configurations as shown in Figure 67 :

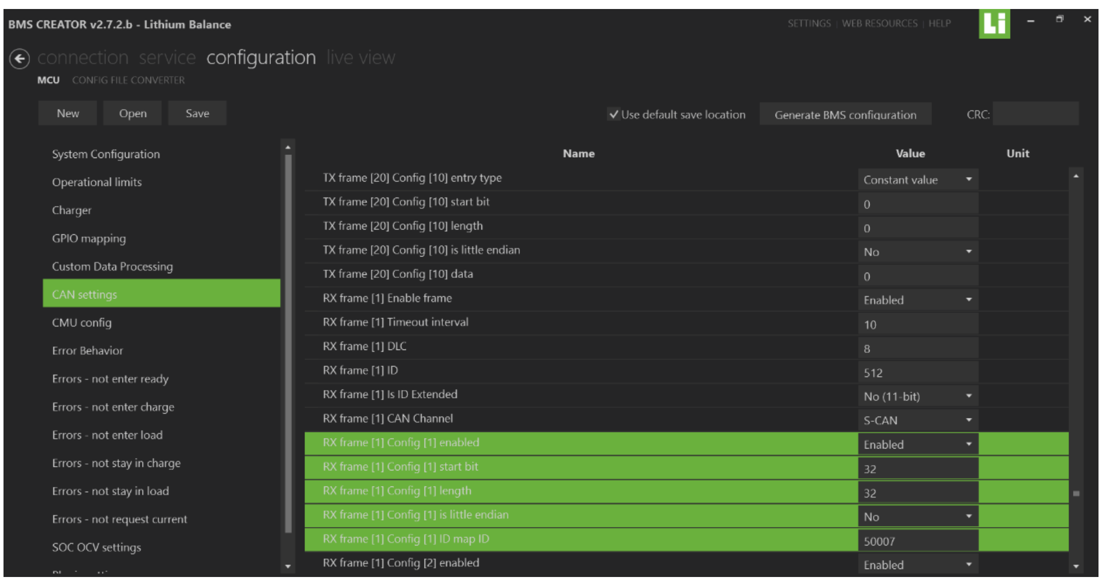


Figure 67 : BMS Creator : CAN RX Frame Configuration

b) The memory ID location needs to be specified in the GPIO Input 1 field as shown in Figure 68.

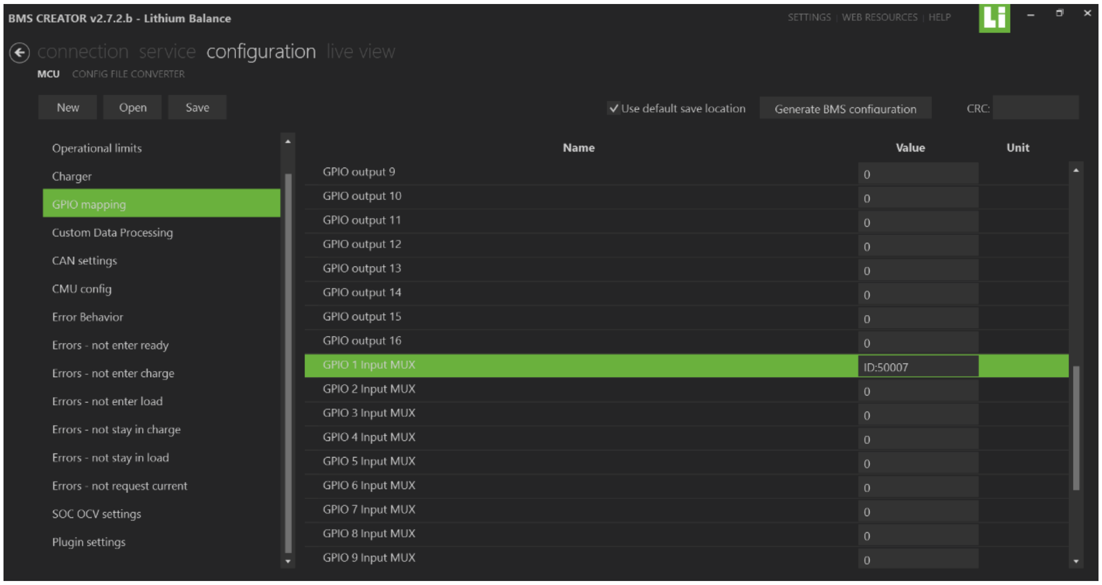


Figure 68 : BMS Creator : Enabling GPIO Digital Input vis BMS Memory ID Location

11.10.13 BMS Active Mode Request via NTC Input (Digital)

Let us understand how to put the BMS inro active mode via NTC Input with the help of an example. In this example we want to use temperature sensor input number 3 as digital input for getting the BMS into "Active Charge" mode. The user needs to do the following configurations for this :

a) Assign GPIO 1 to "Request Active Charge" :

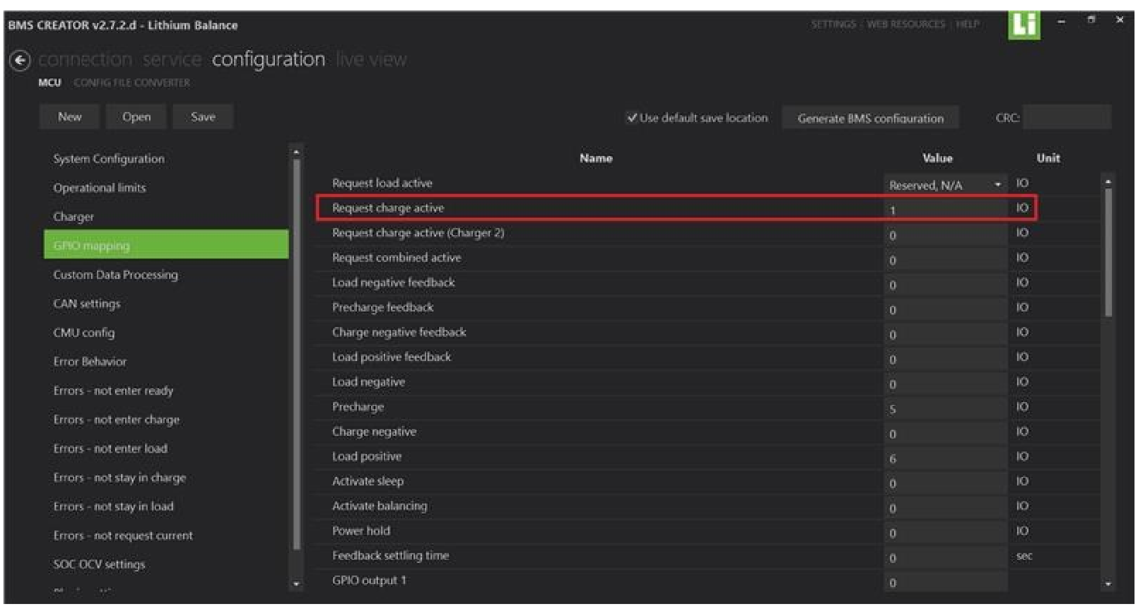


Figure 69 : BMS Creator : Configuring Request Active Charge

b) Assign the temperature sensor number to the "GPIO 1 Input Mux" :

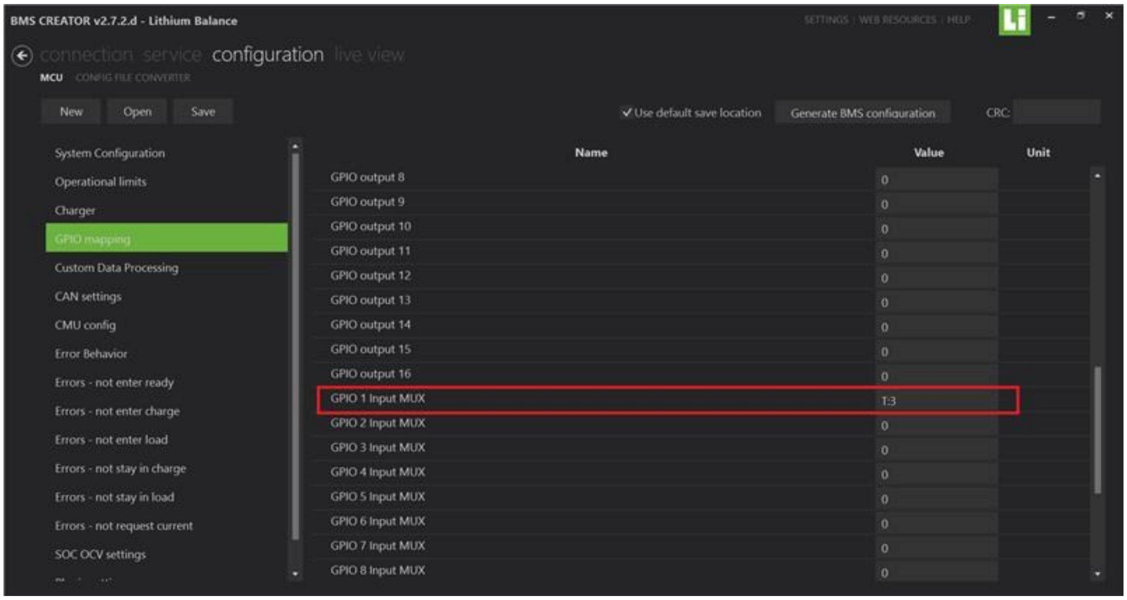


Figure 70 : BMS Creator : Configuring GPIO Input MUX

c) The next step will be to disable the temperature sensor 3 in the "operational limits". If we are using 5 temperature sensors, we will have to make sure that the temperature sensor number 3 (which is used as digital input) is disabled in the bitmask. The corresponding binary ID should be 111011 which corresponds to 59 which needs to be entered in the following two parameters :

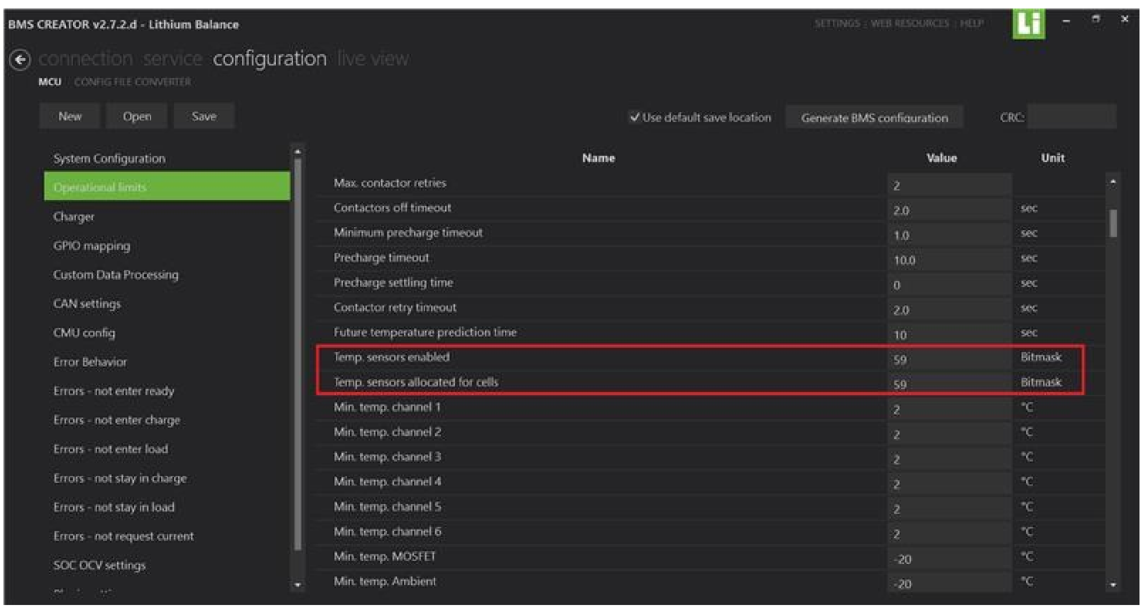


Figure 71 : BMS Creator : Disabling NTC Inputs

d) Short circuit (grounding) of the temperature sensor input will activate the digital GPIO input.

12 CAN Communication

The i-BMS15 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 i-BMS15 is able to run at 4 different CAN speeds: 125 kbps, 250 kbps, 500 kbps, 1000 kbps.

12.1 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 rightmost byte is denoted byte 7.

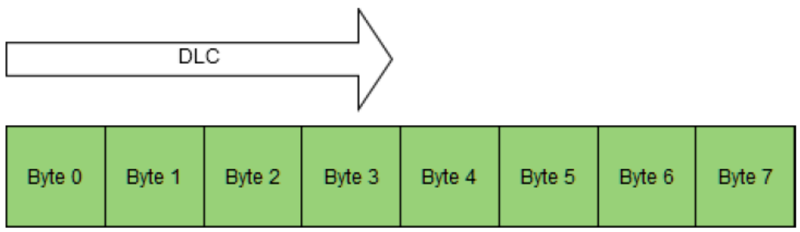


Figure 72 : CAN Frame byte ordering

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 73 .

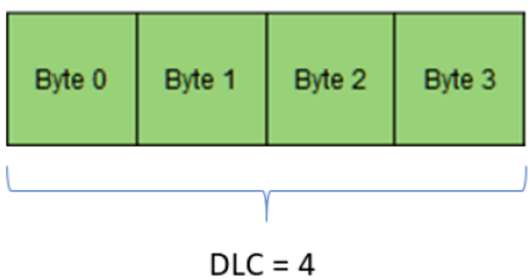


Figure 73 : CAN data byte order, showing a Data Length Code (DLC) of 4 bytes

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.

12.1.1 Big Endian (Motorola) Format

In the big endian bit ordering format, 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 74 : Big endian bit order in CAN frame

A full big endian CAN frame has been depicted in Figure 75.

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 75 : 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 (leftmost byte), but bits count from byte 7 (rightmost byte). Thus, for a DLC of 4 (where bytes 0 to 3 are used as shown in Figure 73), only bits 32-63 are available.

12.1.2 Little Endian (Intel) Format

In the 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 76.

Figure 76 : Little endian bit order in CAN frame

A full little endian CAN frame has been depicted in Figure 77.

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 77 : Little endian bit order for all CAN frame bits

12.2 CAN transmission (TX)

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

- Setting up the frame itself
- Setting up the data to be placed in the frame

12.2.1 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 parameters :

Parameters	Description	Acceptable Values
Enable Frame	The CAN TX frame can be enabled or disabled	Enable/Disable Dropdown
Update Interval	1 : Entering 1 implies that the CAN frame is updated at the maximum rate, i.e. 100 ms 2 : Entering 2 implies that the CAN frame is updated at half the maximum rate, i.e every 200 ms 3 : Entering 3 implies that the CAN frame is updated at one third the maximum frequency, i.e every 300 ms . . ...And so on..	1 - 65535
DLC	The length of the CAN TX frame in bytes.	0 - 8
ID	The CAN TX frame id as a decimal value.	For 29 bit ID : 0 - 536870912 (2^{29}) For 11 bit ID : 0 - 2048 (2^{11})

Is ID Extended	Yes : The frame_id is a 29-bit extended CAN frame ID. No : The frame_id is a 11-bit CAN frame ID.	Yes/No Dropdown
CAN channel	The CAN channel to output the frame on. There is only one CAN channel on i-BMS15 hardware.	S-CAN

Table 9 : Overall CAN TX frame configurable parameters

The parameters mentioned in Table 9 can be configured by clicking on CAN settings under the 'Configuration' pane of BMS Creator (Refer Figure 78).

Figure 78 : BMS Creator : CAN TX frame configuration

12.2.2 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 :

Parameters	Description	Accepted Values
Config [X] enabled	The configuration entry number 'X' (for the CAN TX frame) can be enabled or disabled	Enabled/Disabled Dropdown
Config [X] entry type	The type of the entry data	Constant Value/BMS Variable Dropdown
Config [X] start bit	The bit in the CAN frame, where the first data bit is to be placed	0 - 63
Config [X] length	The amount of bits to write data to; starting from start bit	0 - 32
Config [X] is_little_endian	Yes : Little Endian (Intel format) No : Big Endian (Motorola format)	Yes/No Dropdown

Config [X] data	Either a constant data or an ID allocated to the BMS variable in order to place it in the CAN frame. ID numbers are found in : Appendix 5	0 - 4294967295 ($2^{32} – 1$)

Table 10 : CAN TX Frame Data Configuration Parameters

The parameters mentioned in Table 10 can be configured by clicking on CAN settings under the 'Configuration' pane of BMS Creator (Refer Figure 79).

Figure 79 : BMS Creator : CAN TX Frame Data Configuration

12.3 CAN Reception (RX)

Setting up CAN frames for the BMS to receive them, works similarly as for CAN transmissions. It requires :

- Setting up the frame itself
- Setting up the data contents of the frame and destination IDs

Data in a received frame can be placed in a specific set of Data ID map IDs, from where it can be used by Custom data processing, CAN transmission or any other specific functionality which is set up by the user in the BMS Creator tool as shown in Figure 80.

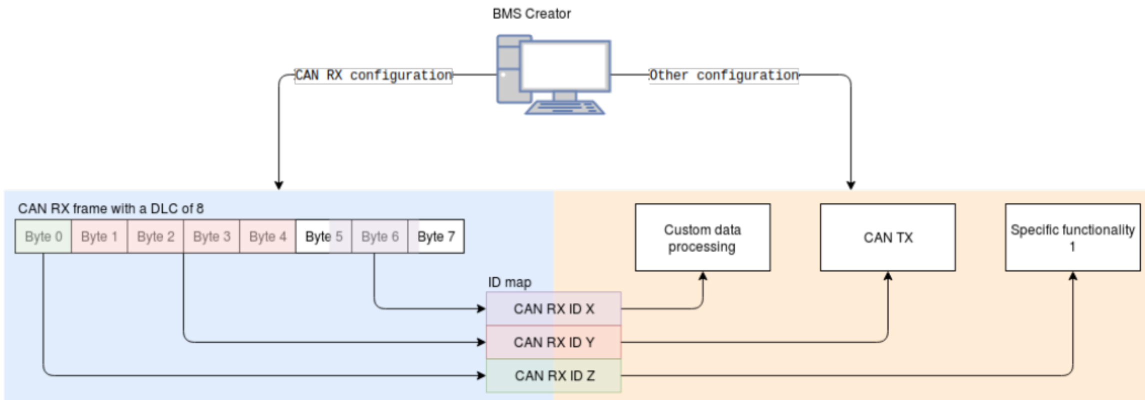


Figure 80 : CAN RX Configuration

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

12.3.1 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 parameters :

Parameters	Description	Acceptable Values
Enable Frame	The CAN RX frame can be enabled or disabled	Enabled/Disabled Dropdown
Update 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 ms. 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...	1 - 65535
DLC	The length of the CAN RX frame in bytes.	0 - 8
ID	The CAN RX frame id as a decimal value.	For 29 bit ID : 0 - 536870912 (2^{29}) For 11 bit ID : 0 - 2048 (2^{11})
Is ID Extended	Yes : The frame id is a 29-bit extended CAN frame ID.	Yes/No Dropdown

	No : The frame id is a 11-bit CAN frame ID.	
CAN channel	The CAN channel to output the frame on. There is only one CAN channel on i-BMS15 hardware.	S-CAN

Table 11 : Overall CAN RX frame configurable parameters

The parameters mentioned in Table 11 can be configured by clicking on CAN settings under the 'Configuration' pane of BMS Creator (Refer Figure 81).

Figure 81 : BMS Creator : CAN RX frame configuration

12.3.2 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 parameters (Refer to Figure 82) :

Parameters	Description	Accepted Values
Config [X] enabled	The configuration entry number 'X' (for the CAN TX frame) can be enabled or disabled	Enabled/Disabled Dropdown
Config [X] start bit	The bit in the CAN frame, where the first data bit is to be placed	0 - 63
Config [X] length	The amount of bits to write data to; starting from start_bit	0 - 32
Config [X] is_little_endian	Yes : Little Endian (Intel format) No : Big Endian (Motorola format)	Yes/No Dropdown
Config [X] data	An ID within a specific range where the data from the CAN frame is placed. ID numbers are found in Appendix 5.	0 - 4294967295 ($2^{32}-1$)

Table 12 : CAN RX Frame Data Configuration Parameters

Figure 82 : BMS Creator : CAN RX Frame Data Configuration

12.4 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:

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

The processed output is then calculated as :

CDP_result = lvalue (operator) rvalue

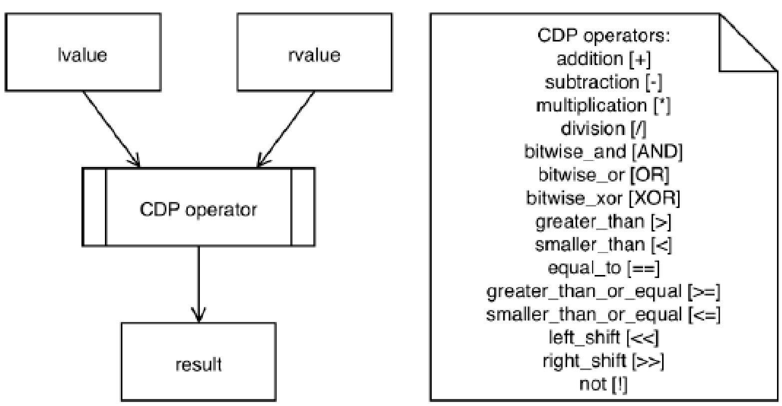


Figure 83 : CDP Operators

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. The intermediate results are stored as regular CDP entries and can be accessed using IDs ID_CDP_OUTPUT1 to ID_CDP_OUTPUT25. 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, refer to Appendix 6.

12.5 CAN Applications

12.5.1 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.

12.5.2 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.

12.5.2.1 EA-PS8200-70 from Elektro-Automatik

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

12.5.2.2 Custom Big Endian Format 1

The frame format is as specified in Figure 84.

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*

Figure 84 : Frame Format 1 for Custom Big Endian

12.5.2.3 Custom Big Endian Format 2

The frame format is as specified in Figure 85.

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*

Figure 85 : Frame Format 2 for Custom Big Endian

Byte 0 uses multiple bits for different purposes as shown in Figure 86.

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*

Figure 86 : Bits Utilisation for Byte 0

12.5.2.4 Custom Big Endian Format 3

The frame format is as specified in Figures 87 & 88 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.

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			

Figure 87 : Format 1 : Requested Charge Voltage

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			

Figure 88 : Format 2 : Requested Charge Current

12.5.3 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.

12.5.3.1 Configuring the Feature

The feature has two configuration parameters :

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

12.5.3.2 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. Refer to Table 13 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 13 : Activating a Request Truth Table (Load & Charge)

For an example of how to use this functionality, Refer to Appendix 7.

12.5.4 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).

12.5.4.1 Configuring the Feature

The feature has one configuration parameter i.e. 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.

12.5.4.2 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. Refer to Table 14 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 14 : Activating a Request Truth Table (Staging)

12.5.4.3 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.

12.5.5 CAN Error Frames

In the "CAN Settings" section on the configuration page of the BMS Creator (Refer to Figure 89), 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. The bit ordering format for CAN Error frames is Motorola (Big Endian).

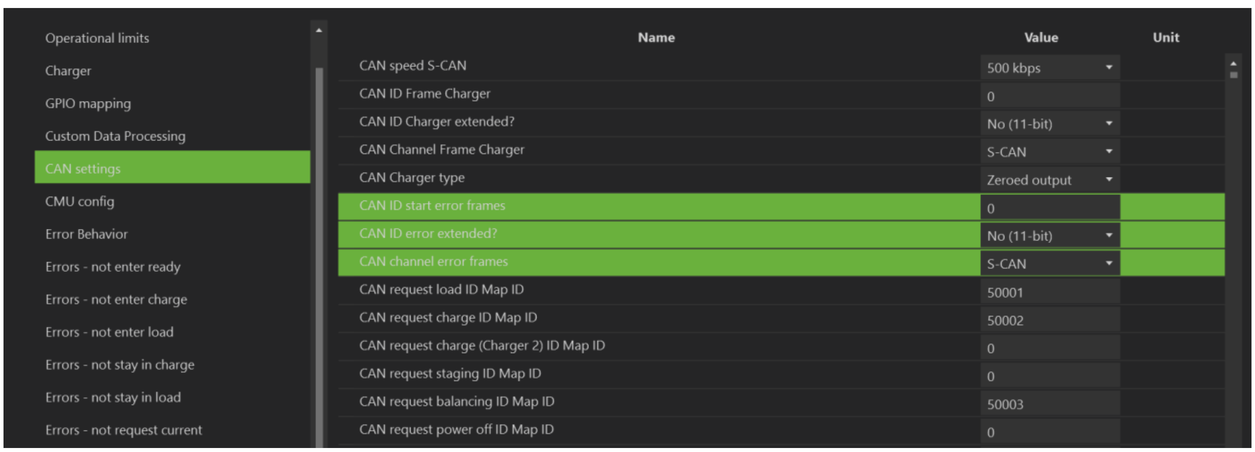


Figure 89 : BMS Creator : CAN Error Frames Configuration

If the CAN Error frames are enabled, the CAN output will consist of the contents displayed in Figure 90 and explained in Table 15 and Table 16. The data format is big endian.

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	

Figure 90 : CAN Error Frame Output

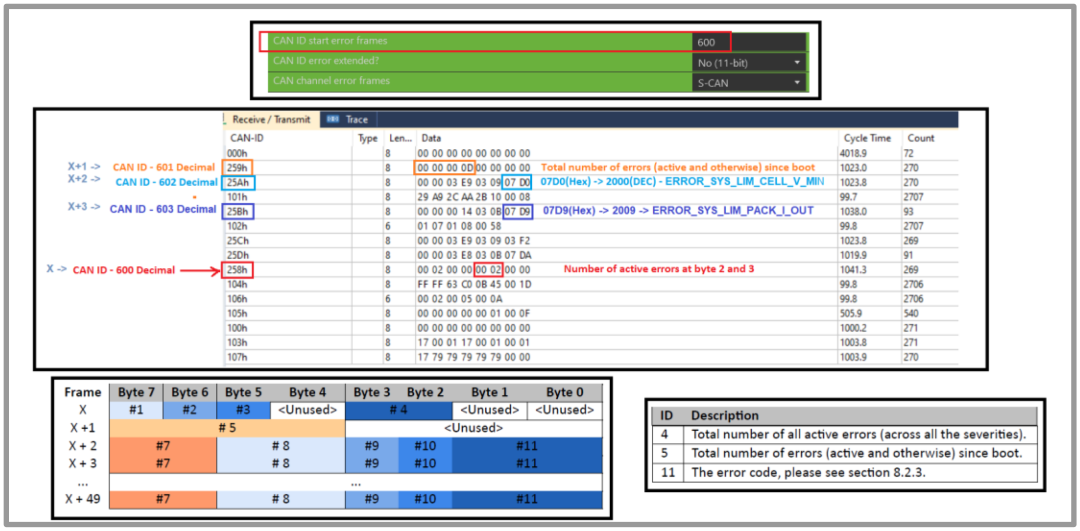
CAN Frame	Description
X (the value entered in the BMS creator field “CAN ID start error frames”)	Overview of the number of active errors
X+1	Error Statistics (Sum of Errors since boot)
X+2 upto maximum X+49	Information about errors. Only active errors are transmitted. E.g. if two errors are present, the X, X+1, X+2 and X+3 CAN frames will be transmitted.

Table 15 : Description of CAN Error 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
6	Internal Lithium Balance A/S variable used for verification
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

Table 16 : Description of CAN error Frame Content

For the configurations made in Figure 89, the following CAN messages can be seen in a particular scenario (Refer to Figure 91). This is just an example in order to provide a better understanding of the CAN Error Frames for the user.



CAN-ID	Type	Len...	Data	Cycle Time	Count
000h		8	00 00 00 00 00 00 00 00	4018.9	72
259h		8	00 00 00 0D 00 00 00 00	1023.0	270
25Ah		8	00 00 03 E9 03 09 07 D0	1023.8	270
101h		8	29 A9 2C AA 2B 10 00 08	99.7	2707
25Bh		8	00 00 00 14 03 0B 07 D9	1038.0	93
102h		6	01 07 01 08 00 58	99.8	2707
25Ch		8	00 00 03 E9 03 09 03 F2	1023.8	269
25Dh		8	00 00 03 E8 03 0B 07 DA	1019.9	91
258h		8	00 02 00 00 00 02 00 00	1041.3	269
104h		8	FF FF 63 C0 0B 45 00 1D	99.8	2706
106h		6	00 02 00 05 00 0A	99.8	2706
105h		8	00 00 00 00 00 01 00 0F	505.9	540
100h		8	00 00 00 00 00 00 00 00	1000.2	271
103h		8	17 00 01 17 00 01 00 01	1003.8	271
107h		8	17 79 79 79 79 79 00 00	1003.9	270

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	

ID	Description
4	Total number of all active errors (across all the severities).
5	Total number of errors (active and otherwise) since boot.
11	The error code, please see section 8.2.3.

Figure 91 : Example of CAN Error Frame in use

13 Appendix 1 : Handling the BMS

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.

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.

Storage

The BMS boards must be stored at temperature and humidity levels that do not exceed specified operating conditions. Refer to 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.

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, like those used for shipping from Lithium Balance.

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.

14 Appendix 2 : Installing BMS Creator

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

15 Appendix 3 : Error Codes & Description

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 occurs at e.g. calculation overflow, wrong indexing, etc. Root cause can be out of bound configuration.
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.
13	D	ERROR_GENERAL_QUEUE_OVERRUN	Overrun in internal software queues like CAN queues or error queues.
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 a RX frame [n] Timeout interval is different than 0, it means that BMS is waiting for 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 the user, this error pops up.
600	258	ERROR_ADC_EXT_GENERAL_ERROR	Analog to Decimal Conversion (ADC) error, e.g. interfacing problems.
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.
700	2BC	LB_ERROR_RTC_LOWBAT	Indicates that the RTC battery is running low
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 setup. 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 sent to CMUs. For example, ADC in MCU after reading the cell voltages is not cleared in it's register.
1006	3EE	ERROR_BALANCING_OPERATION	Write operation of enabled/disabled balancing resistors to the LT device failed.
1008	3F0	ERROR_LTC_VREF2_OUT_OF_RANGE	Reference voltage for cell monitoring ASIC out of range.
1011	3F3	ERROR_CELL_VDELTA	The difference between minimum cell voltage and maximum cell voltage has exceeded the configured difference.
1012	3F4	ERROR_CELL_TDELTA	The difference between minimum cell temperature and maximum cell temperature has exceeded the configured difference.
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 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 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 the 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.
1118	45E	ERROR_TEMP_AUX_SHORTED_CH03	When reading 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.
1119	45F	ERROR_TEMP_AUX_SHORTED_CH04	When reading 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 the 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.

1120	460	ERROR_TEMP_AUX_SHORTED_CH05	When reading 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.
1121	461	ERROR_TEMP_AUX_SHORTED_CH06	When reading 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 the 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.
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 for seeing this error would be a defect in the 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 is not 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 for seeing this error would be a defect in the 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 is not 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 for seeing this error would be a defect in the 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 is not 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 for seeing this error would be a defect in the 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 is not 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 for seeing this error would be a defect in the 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 is not 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 for seeing this error would be a defect in the 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 is not 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.
1170	492	ERROR_TEMP_AUX_MAX_CH07	MOSFET Temperature is above the specified temperature limit.
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.
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.

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 discharge 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".
2011	7DB	ERROR_SYS_CELL_V_NO_VALUE	Special case where the reading of the cell voltage results in extreme values.
2012	7DC	ERROR_SYS_CELL_T_NO_VALUE	Special case where the reading of the cell temperature results in extreme values.
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.
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 sensor type out of allowed range.

2019	7E3	ERROR_SYS_COMM_CMU	When BMS cannot communicate with one or more CMUs.
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	When BMS cannot receive a feedback input for load 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 a feedback intentionally used in the design.
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 precharge contactor. This could be due to missing the actual feedback message, or due to wrong GPIO settings. Users must only define a feedback input in the GPIO menu in BMS Creator, when there is 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 a 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 feedback, but receives feedback input for load positive 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 feedback, but receives 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 a precharge contactor open, and hence does not expect feedback, but receives a feedback input for precharge 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 a charge negative contactor open, and hence does not expect feedback, but receives feedback input for load positive 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 retries of closing contactors exceeds the configured limit. The BMS will retry to close contactors after configured errors are cleared. Number of retries is configured in the Operational limits menu, Max. Contactor retries.
2032	7F0	ERROR_SYS_PROCESS_CMU_DATA	LT device on CMU or i-BMS15 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 i-BMS15.
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.
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. In that case, this error message pops up since 8 A is not within 10 + 1A = [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	ERROR_SYS_OPEN_WIRE	Indicates that there is an open wire in any of the CMUs
2049	801	ERROR_CONFIG_WATERMARKALIGNMEN T_ERRORS	Corrupted xml configuration file
2050	802	ERROR_SYS_ESTOP	Emergency stop activated
2051	803	ERROR_CONTACTOR_PRECHARGE_OPENC	Precharge (or negative contactor) open circuit
2052	804	ERROR_CONTACTOR_PRECHARGE_SHORT	Precharge (or negative contactor) fail
2053	805	ERROR_CONTACTOR_POS_OPENC	Positive contactor is open circuit
2054	806	ERROR_CONTACTOR_SHORT	Indicates a short between hv_1 and hv_2
2055	807	ERROR_CONTACTOR_PRECHARGE_TIMEO UT	Pre-charge has timed out

16 Appendix 4 : Configuration Parameters

Field	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	Parallel Packs Enable	The user can enable the parallel pack functionality using this field.	-	Dropdown
System Configuration	Number of Parallel Packs	Used to enter the number of battery modules connected in parallel.	-	0-6
System Configuration	RTC Wakeup Interval	After this time interval, the RTC wakes up, reads data, checks if there is any error, logs the datasets if any error is there and moves to sleep mode again	mins	0-65535
System Configuration	Current source type	Select sensor to be used for current measurement. Only one sensor can be used at a time. For i-BMS15, the current sensor is shunt only.	-	Shunt
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	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	-	0.00-655.35

		trimmed SoC will indicate the Maximum SoC trim value.		
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
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	Max. cell delta voltage	The maximum allowed difference between the minimum cell voltage and the maximum cell voltage. Error 1011 is generated if this threshold is breached. If the user enters 0 here the error 1011 will be present all the time.	mV	0-6553.5 **(Note :** LiBAL recommended range is 50 - 1000 mV depending upon the application)
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.	degC	-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.	degC	-35.0 - +100.0

Operational Limits	Max. cell delta temperature	The maximum allowed difference between the minimum cell temperature and the maximum cell temperature. Error 1012 is generated if this threshold is breached. If the user enters 0 here the error 1012 will be present all the time	degC	0-6553.5 (**Note :** LiBAL recommended value is anything >5 ^{o}C depending upon the application)
Operational Limits	Max. i2t charge	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.	A^2s	0 – 4294967295
Operational Limits	Max. i2t discharge	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.	A^2s	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 voltage	It is the maximum voltage measured across the precharge resistor i.e. the voltage difference between HV-in and HV-out terminals for the BMS to reach the precharge end condition.	V	0.00 - 42949672.95
Operational Limits	Max. Load 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. Charge 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	Minimum Precharge timeout	It is the minimum time for which the BMS will allow precharge independent of the max. precharge end voltage and max. precharge end current.	sec	0.0 - 6553.5
Operational Limits	Precharge timeout	It is the maximum duration 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	Precharge settling time	Configurable delay from precharge activation until the precharge current starts flowing or until the feedback signals are received.	sec	0 - 255
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	Future Temperature Prediction Time	The BMS will predict the temperature of the battery, based on previous data, for the next "t" seconds configured in this field. It uses the concept of ROCt function.	sec	0-65535 (**Note :** LiBAL recommended range is 1-60 secs)
Operational Limits	Parallel Pack Connection OCV Deadband	The Open Circuit Voltage deadband which will be used for parallel packs connection sequence while charging/discharging.	V	0.0 – 6553.5 (**Note :** LiBAL recommended range is 0-3V)
Operational Limits	Parallel Pack Connection Ext. Voltage Deadband	The Terminal Voltage deadband which will be used for parallel packs connection sequence while charging/discharging. This deadband should always be higher than "Parallel Pack Connection OCV deadband."	V	0.0-6553.5 (**Note :** LiBAL recommended range is 0-3V)
Operational Limits	Temp. sensors enabled	Bitmask of which temperature sensors are enabled. All sensors are enabled by the binary value 00111111 = 63 decimal. Charge. Bit 1 is sensor 1, Bit 2 is sensor 2 etc.	bits	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.	bits	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	Min. temp MOSFET	The minimum allowed temperature for the MOSFET before the error is given.	deg C	(-128) - 127
Operational Limits	Min. temp Ambient	The minimum allowed temperature for Ambient/PCB 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	Max. temp MOSFET	The maximum allowed temperature for the MOSFET before the error is given.	deg C	(-128) - 127
Operational Limits	Max. temp Ambient	The maximum allowed temperature for Ambient/PCB 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	Zero Balancing Deadband	This value determines the balancing performance (BP). For e.g. if the min and max cell voltages are greater than or equal to the xero balancing deadband then the BP is 0%, if the min and max cell voltages are less than balancing deadband then the BP is 100% and if the min and max cell voltages are in between the balancing deadband and zero balancing deadband then the BP is 50%.	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 limit	Not Implemented	V	0 - 65535
Operational Limits	Auto off cell voltage timeout limit	Not Implemented	sec	0 - 65535
Operational Limits	Auto off charge current limit	Not Implemented	%	1 - 100
Operational Limits	Auto off charge timeout limit	Not Implemented	sec	0 - 65535
Operational Limits	Auto off fully charged	Not Implemented	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.	-	On/Off
Charger	CAN Charge Max. V	Maximum output voltage from the CAN charger.		0.0 – 6553.5

Charger	Cell voltage target	The target voltage for all cells when charging. The BMS will use this value as the 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 current and measured current. A value 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 point for the charge current into the battery pack during normal charging.	A	0.0 – 6553.5
Charger	Max. Charge current (Charger 2)	This field gives the user flexibility to configure the maximum charge current for a second charging system apart from the one already configured in Max. Charge Current field above. By second charging system we mean, two different charging systems where the deliverable charge current might be different.	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	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	Request charge active	Input of where to map "Request charge active" functionality for the first charging system. 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 charge active (Charger 2)	Input of where to map "Request charge active" functionality for the second charging system. 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 combined active	This function is not in use in the current version.	IO	0-8
GPIO mapping	Load negative feedback	Feedback for the load negative contactor. If enabled, the BMS expects this feedback signal to follow the state of "Load 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".	IO	0-8

		If there is a mismatch, an error is given and the contactor sequence will fail. 0 = Disabled. 0-8 indicates IO channel.		
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	Load positive feedback	Feedback for the load positive contactor. If enabled, the BMS expects this feedback signal to follow the state of "Load Positive". 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	Load negative	Output to drive Load negative contactor. 0 = Disabled. 5-8 indicates IO channel.	IO	0-8
GPIO mapping	Precharge	Output to drive Precharge contactor. 0 = Disabled. 5 is assigned channel for on-board MOSFET.	IO	0 or 5
GPIO mapping	Charge negative	Output to drive Charge Negative contactor. 0 = Disabled. 5-8 indicates IO channel.	IO	0-8
GPIO mapping	Load positive	Output to drive Load positive contactor. 0 = Disabled. 6 is assigned channel for on-board MOSFET.	IO	0 or 6
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	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	Power hold	Output for the Auto off function.	IO	0 – 16
Custom Data Processing	Slot [N] Operator	The operations to perform : [NOP]:NONE [+] : ADDITION [-] : SUBTRACTION [*] : MULTIPLICATION [/] : DIVISION [&] : BITWISE_AND [|] : BITWISE_OR [^] : BITWISE_XOR [>] : GREATER_THAN [<] : SMALLER_THAN [==] : EQUAL_TO [>=] : GREATER_THAN_OR_EQUAL [<=] : SMALLER_THAN_OR_EQUAL [<<] : LEFT_SHIFT	-	Dropdown

		[>>] : RIGHT_SHIFT [!] : INVERSE		
Custom Data Processing	Slot [N] Right side value	The value to use on the right hand side of the operator.	-	Dropdown
Custom Data Processing	Slot [N] Right side value type	The data type (either constant or data ID).	-	Dropdown
Custom Data Processing	Slot [N] Left side value	The value to use on the left hand side of the operator.	-	Dropdown
Custom Data Processing	Slot [N] Left side value type	The data type (either constant or data ID).	-	Dropdown
CAN settings	CAN speed S-CAN	Input selection for the CAN speed of the application firmware.	-	Dropdown
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. Yes = 29 bit ; No = 11 bit	-	Yes/No
CAN settings	CAN Channel Frame Charger	Selector for what CAN bus to output CAN Charger Frame.	-	S-CAN
CAN settings	CAN Charger type	Selection of predefined legacy CAN charger types: Zeroed output, EA-PS8200-70 from Elektro-Automatik, static charge allowed flag, static charge allowed with latching flag, Powerfinn charger	-	Dropdown
CAN settings	CAN ID start error frames	Start address for a sequential map of all relevant error codes. Select an ID where the following 52 ID's are vacant.	-	0-2047
CAN settings	CAN ID error extended?	Selector to output CAN ID error frames as extended ID. Yes = 29 bit; No = 11 bit	-	Yes/No
CAN settings	CAN channel error frames	The CAN channel	-	S-CAN
CAN settings	CAN request load ID map 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 charge ID map ID	If a CAN rx is setup to receive a charge mode request from the first charging system, this specifies the ID map ID (in CAN rx range) where the request is located.	-	0-65535
CAN settings	CAN request charge ID (Charger 2) map ID	If a CAN rx is setup to receive a charge mode request from the second charging system, this specifies the ID map ID (in CAN rx range) where the request is located.	-	0-65535
CAN settings	CAN request staging ID 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	CAN request balancing ID map ID	If a CAN rx is setup to receive a balancing mode request, this specifies the ID map ID (in CAN rx range) where the request is located.	-	0-65535
CAN settings	CAN request Power Off ID map ID	If a CAN rx is setup to receive activation of sleep mode request, this specifies the ID map	-	0-65535

		ID (in CAN rx range) where the request is located.		
CAN settings	CAN request E-Stop ID Map ID	If a CAN rx is setup to receive a E-Stop functionality 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 can be enabled or disabled	-	Enabled/Disabled
CAN settings	TX Frame [N] Update interval	CAN 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 – 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 a decimal 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	Yes = 29 bit frame; No = 11 bit frame	-	Yes/No
CAN settings	TX Frame [N] CAN Channel	The CAN channel	-	S-CAN
CAN settings	TX Frame [N] Config 1(-10) enabled	The configuration entry number 'X' (for the CAN frame) can be enabled or disabled	-	Enabled/DIsabled
CAN settings	TX Frame [N] Config 1(-10) entry type	The type of the entry data.	-	Constant/BMS Variable
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 amount of bits to write data to, starting from start bit	-	0 to 32
CAN settings	TX Frame [N] Config 1(-10) is little endian	Yes : Little Endian data format ; No : Big Endian data format	-	Yes/No
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] Enable frame	The CAN frame can be enabled or disabled	-	Enabled/Disabled
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	-	1 – 65535

		200 ms 3: The CAN frame is expected within 300 ms ... and so on.		
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	Yes : 29 bit frame ; No : 11 bit frame	-	Yes/No
CAN settings	RX Frame [N] CAN Channel	The CAN channel	-	S-CAN
CAN settings	RX Frame [N] Config 1(-10) enabled	Enabling of CAN Rx frame	-	Enabled/Disabled/Enabled inc. RX timeout error
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	Yes : Little Endian Data Format ; No : Big Endian Data Format	-	Yes/No
CAN settings	RX Frame [N] Config 1(-10) ID map ID	ID to store the received data in.	-	50001-50026
CMU config	Expected CMUs	Setup for number of front end measuring devices.	-	2
CMU config	Enabled cells CMU 1-2	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	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
Error Behavior	Default error behavior type	Reserved (Cannot be configured)	-	-
Error Behavior	Trigger Level	The upper limit for the counter specified by the user where the BMS starts reporting the error.	-	0-65535
Error Behavior	Debounce Increment	The rate at which the counter counts the value up when an error is logged.	-	0-65535
Error Behavior	Debounce Decrement	The rate at which the counter counts the value down when no error is logged.	-	0-65535
Error Behavior	Heal Decrement	The rate at which the counter counts the value down once the trigger value has been reached.	-	0-65535
Error Behavior	Error Behavior (1-9) Type	Reserved (Cannot be configured)	-	0
Error Behavior	Error Code Config (1-10)	The error code to which one of the custom error behaviors (1-9) is assigned.	-	1 – 65535
Error Behavior	Behavior Index	The error behavior index (1 to 9) to be assigned to the particular error code.	-	1-9

Error Behavior	DTC	The user can use this field if the application requires DTCs based on UDS.	-	-
Errors - not enter ready	Code error 1-64	The user can enter the error codes in order to create an error checklist for the BMS before it enters Ready Mode.	-	1 – 65535
Errors - not enter charge	Code error 1-64	The user can enter the error codes in order to create an error checklist for the BMS before it enters Charge Mode.	-	1 – 65535
Errors - not enter load	Code error 1-64	The user can enter the error codes in order to create an error checklist for the BMS before it enters Load Mode.	-	1 – 65535
Errors - not stay charge	Code error 1-64	The user can enter the error codes in order to create an error checklist for the BMS to stay in Charge Mode.	-	1 – 65535
Errors - not stay load	Code error 1-64	The user can enter the error codes in order to create an error checklist for the BMS to stay in Load Mode.	-	1 – 65535
Errors - not request current	Code error 1-64	The user can enter the error codes and create an error checklist so that : 1. The BMS does not go into error mode but stays in the present mode and does not change contactors state. 2. The requested current is set to 0.	-	1 – 65535
SOC-OCV settings	Enable data sets	The user can enter the number of temperature values for which he/she wants to enter the OCV data. 0 implies the feature is disabled. The user must enter at least 2 temperature values here.	-	0-5
SOC-OCV settings	Data set [N] temperature	The temperature at which the data points for dataset N are valid. The user should start entering from lowest temperature to highest temperature.	deg C	(-40)-120
SOC-OCV settings	Data set [N] voltage [X]	The OCV data points for SoC value X (X % SoC) for dataset N	mV	0-5500
SOC-OCV settings	Hybrid SOC w1	The user can specify the first window based on the SOC-OCV curve of the li-ion cell in application.	%	0-100
SOC-OCV settings	Hybrid SOC q1	The quality factor/accuracy for window 1 here refers to the trust the user is willing to put on the SOC-OCV datasets in the first window. The user must enter 0 in this field. (0 implying complete trust on the SOC-OCV datasets)	%	0
SOC-OCV settings	Hybrid SOC w2	The user can specify the second window based on the SOC-OCV curve of the li-ion cell in application.	%	0-100
SOC-OCV settings	Hybrid SOC q2	The quality factor/accuracy for window 2 here refers to the trust the user is willing to put on the SOC-OCV datasets in the second window.	%	0-100

SOC-OCV settings	Hybrid SOC w3	The user can specify the third window based on the SOC-OCV curve of the li-ion cell in application.	%	0-100
SOC-OCV settings	Hybrid SOC q3	The quality factor/accuracy for window 3 here refers to the trust the user is willing to put on the SOC-OCV datasets in the third window.	%	0-100
SOC-OCV settings	Hybrid SOC w4	The user can specify the fourth window based on the SOC-OCV curve of the li-ion cell in application.	%	0-100
SOC-OCV settings	Hybrid SOC q4	The quality factor/accuracy for window 4 here refers to the trust the user is willing to put on the SOC-OCV datasets in the third window.	%	0-100
SOC-OCV settings	Hybrid SOC q5	The quality factor/accuracy for window 4 here refers to the trust the user is willing to put on the SOC-OCV datasets in the third window. The user must enter 0 in this field, thus, implying complete trust in the SOC-OCV datasets.	%	0
SOC-OCV settings	Hybrid SOC Calibration Sensitivity	It is the value added to the accuracy of SOC based on Coulomb Counting in case of SOC calibration.	%	0-100
SOC-OCV settings	Initial Pack Resistance	The measured initial internal resistance of the battery pack.	mΩ	0-6553.5 (**Note :** LiBAL recommended range is 0-25mΩ depending upon the cell manufacturer data)
SOC-OCV settings	EOL Resistance Factor	The factor by which the cell internal resistance will increase as it reached end of life. The user can calculate this value from the cell datasheet.	-	0-6553.5
SOC-OCV settings	EOL Count	The cycle life of the cells used in the application before it reaches end of life.	-	0-6553.5
SOC-OCV settings	SOHC Update Ratio	The user needs to enter the weightage they want to give to the newly calculated SOH (based on capacity). It is recommended by LiBAL to enter 1% here as SOH is something that changes very slowly.	%	0-100
SOC-OCV settings	Balancing Config 0	It is used to enable/disable balancing based on SOC. ‘0’ can be used for enabling and ‘1’ can be used for enabling.	-	0-1
SOC-OCV settings	Balancing Config 1	It is the lower limit for updating the balancing target.	%	0-100
SOC-OCV settings	Balancing Config 2	It helps the user to provide a range (lower limit) of SOC values for updating the balancing target.	%	0-100

SOC-OCV settings	Balancing Config 3	It helps the user to provide a range (upper limit) of SOC values for updating the balancing target.	%	0-100
SOC-OCV settings	Balancing Config 4	It is the upper limit for updating the balancing target.	%	0-100
SOC-OCV settings	Balancing Config 5	It is the lower limit for allowing the balancing based on SOC during charging.	%	0-100
SOC-OCV settings	Balancing Config 6	It helps the user to provide a range (lower limit) of values for allowing balancing based on SOC during charging.	%	0-100
SOC-OCV settings	Balancing Config 7	It helps the user to provide a range (upper limit) of values for allowing balancing based on SOC during charging.	%	0-100
SOC-OCV settings	Balancing Config 8	It is the upper limit for allowing the balancing based on SOC during charging.	%	0-100
SOC-OCV settings	Balancing Config 9	It is the lower limit for allowing the balancing based on SOC during discharging.	%	0-100
SOC-OCV settings	Balancing Config 10	It helps the user to provide a range (lower limit) of values for allowing balancing based on SOC during discharging.	%	0-100
SOC-OCV settings	Balancing Config 11	It helps the user to provide a range (upper limit) of values for allowing balancing based on SOC during discharging.	%	0-100
SOC-OCV settings	Balancing Config 12	It is the upper limit for allowing the balancing based on SOC during discharging.	%	0-100
SOC-OCV settings	Balancing Config 13	It is the quality threshold for disabling balancing based on SOC. It is expressed in Ampere-seconds (A-s).	A-s	0-255
SOC-OCV settings	Minimum OCV Accuracy for SOHc	The user is required to input the point where the quality of SOC based on SOC-OCV curve is maximum.	%	0-100
SOC-OCV settings	Minimum Delta SOC for SOHc	It is the minimum difference in SOC from point 'Minimum OCV Accuracy for SOHc' to another point where the quality of SOC based on SOC-OCV curve is expected to be higher than 'Minimum OCV Accuracy for SOHc.'	%	0-100
SOC-OCV settings	Maximum Delta Temperature for SOHc	This parameter specifies the maximum allowed difference in temperature while measuring the two above mentioned points.	degC	0-255 (Note : LiBAL recommended range is 0-60^{0}C)
SOC-OCV settings	Nominal Current	The rated continuous current for the pack which will indicate the C-rating for the i-BMS15 algorithms.	A	0-65535

17 Appendix 5 : Data ID Maps

ID	Name	Type	Scaling	Unit	Comment
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 with 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 with 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_HV_1	UINT16	0,1	V	Voltage measurement at positive terminal of load
10	PACK_V_HV_2	UINT16	0,1	V	Voltage measurement at positive terminal of charger
11	PACK_V_SUM_OF_CELLS	UINT16	0,1	V	Sum of all cells
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	%	Trimmed 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	-	0 if charger output enabled; 1 if charger output disabled
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	-	0 if charger CAN active; 1 if not active
31	DYN_LIM_I2T_REMAIN	UINT32	1	A^2s	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	-	As soon as the precharge end conditions are met, precharge contactor opens and the main contactors close. This is when ID 35 is set and the BMS moves into "Active" state
36	CONTACTORS_ACTIVATE_CHARGE	UINT8	1	-	The flag raised when there is a request for activation charge mode
37	CONTACTORS_ACTIVATE_LOAD	UINT8	1	-	The flag raised when there is a request for activation of load mode
39	CONTACTORS_CHARGER_ACTIVATED	UINT8	1	-	Once the flag (ID 36) is raised and the error checklists are checked successfully, charge mode is activated
40	CONTACTORS_LOAD_ACTIVATED	UINT8	1	-	Once the flag (ID 37) is raised and the error checklists are checked successfully, load mode is activated
42	CONTACTORS_ACTIVATION_ALLOWED	UINT8	1	-	Contactor activation is allowed as soon as ID 39 or ID 40 are activated
43	CONTACTORS_EMERGENCY_OFF	UINT8	1	-	This flag is raised when E-Stop functionality has been enabled by the user
44	CONTACTORS_CONTACTOR_RETRIES	UINT8	1	-	Count of contactor closing retries
45	BALANCING_ALLOWED	UINT8	1	-	This flag is raised when the cell voltages are above the balancing limit and not within the balancing deadband
46	BALANCING_LIMIT	UINT16	0,1	mV	Lower voltage limit for balancing to be active
47	FLAGS_FULLY_CHARGED	UINT8	1	-	This flag is raised when all the cells are in the 'charge complete deadband' and gets deactivated when the charging process is stopped and capacity calibration takes place.
48	FLAGS_FULLY_CHARGED_LATCHED	UINT8	1	-	This flag is raised when all the cells are in the 'charge complete deadband' and gets deactivated only when at least 0.1 Ah charge has been removed from the cell during discharging

49	FLAGS_LOAD_ACTIVE	UINT8	1	-	This flag is raised when the BMS enters into Active Load mode
50	FLAGS_CHARGER_ACTIVE	UINT8	1	-	This flag is raised when the BMS enters into Active Charge mode
51	FLAGS_PRECHARGE_ACTIVE	UINT8	1	-	This flag is raised when the pre-charge sequence begins
52	FLAGS_BALANCING_ACTIVE	UINT8	1	-	This flag is raised when the balancing is performed on cells
53	FLAGS_CHARGE_REG_ACTIVE	UINT8	1	-	This flag is raised when the BMS enters into Active Charge Regeneration Mode
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
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
144	MOSFET_T	INT8	1	°C	MOSFET Temperature
145	PCB_T	INT8	1	°C	Ambient Temperature
535	CMUS_CMU1_V_REF_5V	UINT16	0.1	mV	5 V Reference Voltage
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]
955	TIME_BETWEEN_BOOTS_S	UINT32	1	sec	Time between the current boot and the last one
956	EPOCH_TIME_S	UINT32	1	sec	System time (boot time plus uptime)
957	UPTIME_S	UINT32	1	sec	System up time since power on or wakeup
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 of cell w. highest 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	UINT32	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
1076	FLAGS_WAITING_FOR_STAGING_REQ	UINT8	1	-	This flag will turn to 1 once the BMS awaits staging request
1077	SOP_INTERVAL_DATA1	UINT16	1	-	SoP data for the next 1 second
1078	SOP_INTERVAL_DATA2	UINT16	1	-	SoP data for the next 10 seconds
1081	SOH	UINT16	0,01	%	SoH

1114	CONFIG_ID	UINT32	-	-	User created ID of the stored configuration
1115	CONFIG_CRC	UINT32	-	-	Checksum calculated for the latest configuration stored in the BMS
1116	DYNAMIC_SOC_0	UINT8	-	-	Data for Dynamic SOC
1117	DYNAMIC_SOC_1	UINT8	-	-	Data for Dynamic SOC
1118	DYNAMIC_SOC_2	UINT8	-	-	Data for Dynamic SOC
1119	SAFETY_OPEN_WIRE_CMU_1	UINT32	-	-	Open wire
1151	DCLI at time 0	UINT16	-	-	Future release
1152	DCLO at time 0	UINT16	-	-	Future release
1153	FUTURE_TEMPERATURE	UINT32	1	°C	Battery Temperature Prediction for the next “t” seconds configured by the user
1200	CMUS_CMU1_CELL_V13	UINT16	0,1	mV	Cell voltage for cmu[0], cell_v[12]
1201	CMUS_CMU1_CELL_V14	UINT16	0,1	mV	Cell voltage for cmu[0], cell_v[13]
1202	CMUS_CMU1_CELL_V15	UINT16	0,1	mV	Cell voltage for cmu[0], cell_v[14]
3000	CMU_VERSION	UINT8	-	-	LiBAL internal value
3002	PACK_RESISTANCE_U	UINT32	0,1	µohm	Calculated Pack Resistance
3003	I_MA	INT32	1	mA	LiBAL internal value
3004	I_CHARGE_MAX	UINT16	0,1	A	Maximum Charge Current
3005	CHG_TIME_ESTIMATE	UINT32	1	sec	Estimated Charging Time
3006	CHARGER_2_ACTIVE	UNIT8	1	-	Second Charging System Active
3007	INITIAL_PACK_RESISTANCE_100U	UINT16	0,1	mohm	User Configured Initial Resistance of Battery Pack
3008	SOE_WS	UINT 32	1	Ws	Available State of Energy of the Battery at any point of time
3009	SOE_FULL_WS	UINT 32	1	Ws	State of Energy of the fully charged battery
3010	RTC_WAKEUP_LATCHED	UINT8	-	-	This flag will occur when the RTC Wakes Up
3011	UDS_DIAGNOSTIC_SESSION	UINT8	-	-	UDS ISO14229 Service
3012	LIM_CELL_V_MIN	UINT16	0,1	mV	Cell Minimum Voltage Limit
3013	LIM_CELL_V_MAX	UINT16	0,1	mV	Cell Maximum Voltage Limit
3014	CONFIG_VERSION_TYPE	UINT8	-	-	The firmware type defines what kind of code release the build is: * a = **alpha** build * b = **beta** build * r = **release** build (also release candidates) * t = **test** build * D = **debug** build (ASCII value needs to considered)
3015	BALANCING_PERFORMANCE	UINT8	0,01	%	The balancing performance based on Balancing Deadband and Zero Balancing Deadband
3101	R_PACK[0]	UINT16	0,1	mohm	Meant for Future Use
3102	PACK_OCV	UINT32	1	mV	The Open Circuit Voltage of the Pack

3140	SHUNT_R_100N	UINT32	0,1	µohm	Shunt Resistance
3141	SHUNT_OFFSET	INT16	1	-	ADC Offset for Shunt
3142	ADDRESS	UINT8	-	-	Node Index Value
4000	IS_MASTER	UINT8	-	-	Flag that indicates that the particular pack is the master. The output will be 0 if it is not the master and 1 if it is the master.
4001	NUM_PACKS	UINT8	-	-	Number of Parallel Packs Connected
4002	NUM_PACKS_ACTIVE	UINT8	-	-	Number of Parallel Packs active currently
4003	UNIFIED_DCLI	UINT32	0,1	A	Total Dynamic Current Limit In for all parallel packs combined
4004	UNIFIED_DCLO	UINT32	0,1	A	Total Dynamic Current Out for all parallel packs combined
4005	UNIFIED _SOC	UINT16	0,01	%	Combined SOC of all paralleled packs
4006	UNIFIED_VOLTAGE	UINT16	0,1	V	Combined Voltage of all paralleled packs
4007	UNIFIED_CURRENT	UINT32	0,01	mA	Combined Current of all paralleled packs
4008	UNIFIED_FULL_CHARGE	UINT16	0,1	Ah	Combined Full Capacity of the paralleled packs
4009	UNIFIED_REMAINING_CHARGE	UINT16	0,1	Ah	Combined Remaining Capacity of the paralleled packs
4010	PACK[0]_STATUS	UINT8	-	-	Status of the pack : 0 - Initializing, 1 - Ready, 2 - Transitioning (Opening/Closing FETs), 3 - Charging, 4- Discharging, 5 - Error
4011	PACK[0]_FET_CLOSED	UINT8	-	-	1 when FET is closed
4012	PACK[0]_PACK_VOLTAGE	UINT16	0,1	V	Pack Voltage
4013	PACK[0]_PACK_CURRENT	UINT32	0,01	mA	Pack Current
4014	PACK[0]_INTERNAL_RESISTANCE_M	UINT16	1	mohm	Internal Resistance of the Pack
4015	PACK[0]_DCLI	UINT16	0.1	A	Dynamic Charge Current Limit
4016	PACK[0]_DCLO	UINT16	0.1	A	Dynamic Discharge Current Limit
4017	PACK[0]_Q_FULL	UINT16	0.1	Ah	Full Capacity of Pack
4018	PACK[0]_Q_REMAIN	UINT16	0.1	Ah	Remaining Capacity of Pack
4019	PACK[1]_STATUS	UINT8	-	-	Status of the pack : 0 - Initializing, 1 - Ready, 2 - Transitioning (Opening/Closing FETs), 3 - Charging, 4- Discharging, 5 - Error
4020	PACK[1]_FET_CLOSED	UINT8	-	-	1 when FET is closed
4021	PACK[1]_PACK_VOLTAGE	UINT16	0,1	V	Pack Voltage
4022	PACK[1]_PACK_CURRENT	UINT32	0,01	mA	Pack Current
4023	PACK[1]_INTERNAL_RESISTANCE_M	UINT16	1	mohm	Internal Resistance of the Pack
4024	PACK[1]_DCLI	UINT16	0.1	A	Dynamic Charge Current Limit

4025	PACK[1]_DCLO	UINT16	0.1	A	Dynamic Discharge Current Limit
4026	PACK[1]_Q_FULL	UINT16	0.1	Ah	Full Capacity of Pack
4027	PACK[1]_Q_REMAIN	UINT16	0.1	Ah	Remaining Capacity of Pack
4028	PACK[2]_STATUS	UINT8	-	-	Status of the pack : 0 - Initializing, 1 - Ready, 2 - Transitioning (Opening/Closing FETs), 3 - Charging, 4- Discharging, 5 - Error
4029	PACK[2]_FET_CLOSED	UINT8	-	-	1 when FET is closed
4030	PACK[2]_PACK_VOLTAGE	UINT16	0,1	V	Pack Voltage
4031	PACK[2]_PACK_CURRENT	UINT32	0,01	mA	Pack Current
4032	PACK[2]_INTERNAL_RESISTANCE_M	UINT16	1	mohm	Internal Resistance of the Pack
4033	PACK[2]_DCLI	UINT16	0.1	A	Dynamic Charge Current Limit
4034	PACK[2]_DCLO	UINT16	0.1	A	Dynamic Discharge Current Limit
4035	PACK[2]_Q_FULL	UINT16	0.1	Ah	Full Capacity of Pack
4036	PACK[2]_Q_REMAIN	UINT16	0.1	Ah	Remaining Capacity of Pack
4037	PACK[3]_STATUS	UINT8	-	-	Status of the pack : 0 - Initializing, 1 - Ready, 2 - Transitioning (Opening/Closing FETs), 3 - Charging, 4- Discharging, 5 - Error
4038	PACK[3]_FET_CLOSED	UINT8	-	-	1 when FET is closed
4039	PACK[3]_PACK_VOLTAGE	UINT16	0,1	V	Pack Voltage
4040	PACK[3]_PACK_CURRENT	UINT32	0,01	mA	Pack Current
4041	PACK[3]_INTERNAL_RESISTANCE_M	UINT16	1	mohm	Internal Resistance of the Pack
4042	PACK[3]_DCLI	UINT16	0.1	A	Dynamic Charge Current Limit
4043	PACK[3]_DCLO	UINT16	0.1	A	Dynamic Discharge Current Limit
4044	PACK[3]_Q_FULL	UINT16	0.1	Ah	Full Capacity of Pack
4045	PACK[3]_Q_REMAIN	UINT16	0.1	Ah	Remaining Capacity of Pack
4046	PACK[4]_STATUS	UINT8	-	-	Status of the pack : 0 - Initializing, 1 - Ready, 2 - Transitioning (Opening/Closing FETs), 3 - Charging, 4- Discharging, 5 - Error
4047	PACK[4]_FET_CLOSED	UINT8	-	-	1 when FET is closed
4048	PACK[4]_PACK_VOLTAGE	UINT16	0,1	V	Pack Voltage
4049	PACK[4]_PACK_CURRENT	UINT32	0,01	mA	Pack Current
4050	PACK[4]_INTERNAL_RESISTANCE_M	UINT16	1	mohm	Internal Resistance of the Pack
4051	PACK[4]_DCLI	UINT16	0.1	A	Dynamic Charge Current Limit
4052	PACK[4]_DCLO	UINT16	0.1	A	Dynamic Discharge Current Limit
4053	PACK[4]_Q_FULL	UINT16	0.1	Ah	Full Capacity of Pack
4054	PACK[4]_Q_REMAIN	UINT16	0.1	Ah	Remaining Capacity of Pack
4055	PACK[5]_STATUS	UINT8	-	-	Status of the pack : 0 - Initializing, 1 - Ready, 2 - Transitioning (Opening/Closing FETs), 3 - Charging, 4- Discharging, 5 - Error

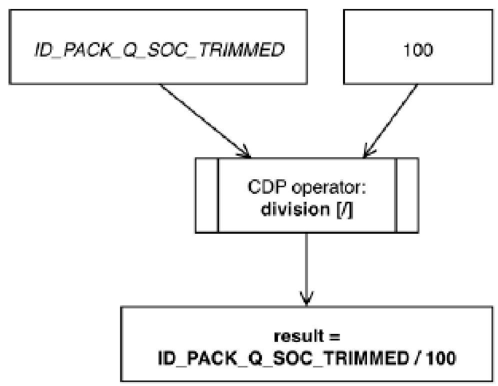
4056	PACK[5]_FET_CLOSED	UINT8	-	-	1 when FET is closed
4057	PACK[5]_PACK_VOLTAGE	UINT16	0,1	V	Pack Voltage
4058	PACK[5]_PACK_CURRENT	UINT32	0,01	mA	Pack Current
4059	PACK[5]_INTERNAL_RESISTANCE_M	UINT16	1	mohm	Internal Resistance of the Pack
4060	PACK[5]_DCLI	UINT16	0.1	A	Dynamic Charge Current Limit
4061	PACK[5]_DCLO	UINT16	0.1	A	Dynamic Discharge Current Limit
4062	PACK[5]_Q_FULL	UINT16	0.1	Ah	Full Capacity of Pack
4063	PACK[5]_Q_REMAIN	UINT16	0.1	Ah	Remaining Capacity of Pack
4064	PACK[6]_STATUS	UINT8	-	-	Status of the pack : 0 - Initializing, 1 - Ready, 2 - Transitioning (Opening/Closing FETs), 3 - Charging, 4- Discharging, 5 - Error
4065	PACK[6]_FET_CLOSED	UINT8	-	-	1 when FET is closed
4066	PACK[6]_PACK_VOLTAGE	UINT16	0,1	V	Pack Voltage
4067	PACK[6]_PACK_CURRENT	UINT32	0,01	mA	Pack Current
4068	PACK[6]_INTERNAL_RESISTANCE_M	UINT16	1	mohm	Internal Resistance of the Pack
4069	PACK[6]_DCLI	UINT16	0.1	A	Dynamic Charge Current Limit
4070	PACK[6]_DCLO	UINT16	0.1	A	Dynamic Discharge Current Limit
4071	PACK[6]_Q_FULL	UINT16	0.1	Ah	Full Capacity of Pack
4072	PACK[6]_Q_REMAIN	UINT16	0.1	Ah	Remaining Capacity of Pack
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
65000	GET_FIRST_ERROR_LOG_ENTRY	UINT8_ ARRAY	-	-	Retrieves the first error log entry (slot 0) and resets "current slot" counter to 0.
65001	GET_NEXT_ERROR_LOG_ENTRY	UINT8_ ARRAY	-	-	Retrieves the next error log entry and sets "current slot" to this entry.
65002	GET_FIRST_ACTIVE_ERROR_ENTR Y	UINT8_ ARRAY	-	-	Retrieves the first active error entry and resets "current index" counter to 0.
65003	GET_NEXT_ACTIVE_ERROR_ENTR Y	UINT8_ ARRAY	-	-	Retrieves the next active error entry and sets "current index" to this entry.
65004	SERIAL_NUMBER	UINT8_ ARRAY	-	-	Either read or write the serial number using this ID.

18 Appendix 6 : Custom Data Processing Examples

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

- lvalue is addressed data from the ID MAP using ID_PACK_Q_SOC_TRIMMED
- rvalue is constant data with the value 100
- The operator is division [\]



18.2 Scaling and Offset of Pack Current

Configure CDP output #1 with :

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

Result is stored in ID_CDP_OUTPUT1

Configure CDP output #2 with :

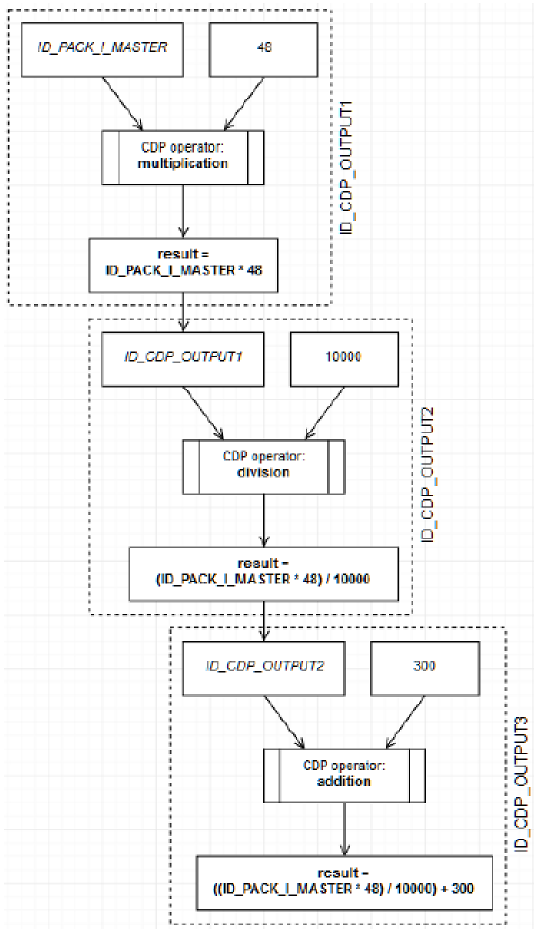
- lvalue is addressed data from the ID MAP using ID: ID_CDP_OUTPUT1 (CDP result from before)
- rvalue is constant data with value 10000
- operator is division [/]

Result is stored in ID_CDP_OUTPUT2

Configure CDP output #3 with :

- lvalue is addressed data from the ID MAP using ID: ID_CDP_OUTPUT2 (CDP result from before) rvalue is constant data with value 300
- operator is addition [+]

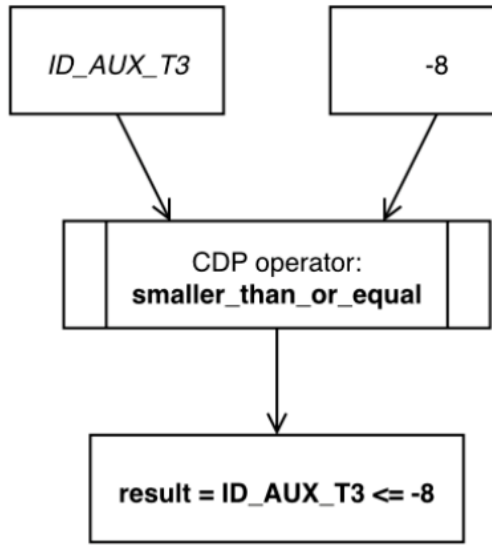
Result is stored in ID_CDP_OUTPUT3, where it can be configured to be broadcasted on the CAN bus.



18.3 Setting Up Custom Warning Bit with CDP :

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

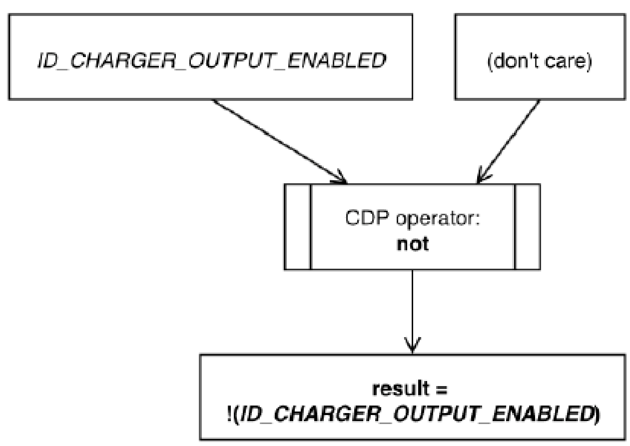
The result is then 1 if ID_AUX_T3 is less than or equal to -8, otherwise the result is 0.



18.4 Inverting Charger Enabled Bit

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

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



19 Appendix 7 : CAN based Mode and Staging Request Examples

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.

19.1 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)

19.2 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 = Enabled
RX frame [1] Timeout interval = 200
RX frame [1] DLC = 8
RX frame [1] ID = 291 (0x123)
RX frame [1]is ID extended= No (11-bit)
RX frame [1] CAN Channel = SCAN
RX frame [1] Config [1] enable = Enabled
RX frame [1] Config [1] start bit = 56
RX frame [1] Config [1] length = 8
RX frame [1] Config [1] is little endian = No
RX frame [1] Config [1] ID map ID = 50001 (ID Map CAN Rx Data 1 ID)
RX frame [1] Config [2] enable = Enabled
RX frame [1] Config [2] start bit = 48
RX frame [1] Config [2] length = 8
RX frame [1] Config [2] is little endian = No
RX frame [1] Config [2] ID map ID = 50002 (ID Map CAN Rx Data 2 ID)
RX frame [1] Config [3] enable = Enabled
RX frame [1] Config [3] start bit = 40
RX frame [1] Config [3] length = 8
RX frame [1] Config [3] is little endian = No
RX frame [1] Config [3] ID map ID = 50003 (ID Map CAN Rx Data 3 ID)
RX frame [1] Config [4] enable=Enabled
RX frame [1] Config [4] start bit=32

RX frame [1] Config [4] length=8
RX frame [1] Config [4] is little endian=No
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.

19.3 CAN TX Configuration Parameters

TX frame [1] Enable frame = Enabled
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 = No (11-bit)
TX frame [1] CAN Channel = SCAN
TX frame [1] Config [1] enabled = Enabled
TX frame [1] Config [1] entry type = BMS Variable
TX frame [1] Config [1] start bit = 56
TX frame [1] Config [1] length = 8
TX frame [1] Config [1] is little endian = No
TX frame [1] Config [1] data = 50001 (ID Map CAN Rx Data 1 ID)
TX frame [1] Config [2] enabled = Enabled
TX frame [1] Config [2] entry type = BMS Variable
TX frame [1] Config [2] start bit = 48
TX frame [1] Config [2] length = 8
TX frame [1] Config [2] is little endian = No
TX frame [1] Config [2] data = 50002 (ID Map CAN Rx Data 2 ID)
TX frame [1] Config [3] enabled = Enabled
TX frame [1] Config [3] entry type = BMS Variable
TX frame [1] Config [3] start bit = 40
TX frame [1] Config [3] length = 8
TX frame [1] Config [3] is little endian = No
TX frame [1] Config [3] data = 50003 (ID Map CAN Rx Data 3 ID)
TX frame [1] Config [4] enabled = Enabled
TX frame [1] Config [4] entry type = BMS Variable
TX frame [1] Config [4] start bit = 32
TX frame [1] Config [4] length = 8
TX frame [1] Config [4] is little endian = No
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.

19.4 Load Mode Cycle

- 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.

- Stage 2: Load Request, Pre-Charging in Progress

ID: 0x123 Data: 01 00 00 00 00 00 00 ID: 0x001 Data: 01 00 00 00 00 00 00

A load request is being broadcasted, and the BMS indicates that it has received a load request.

- Stage 3: Load Request, Pre-Charging is Done

ID: 0x123 Data: 01 00 00 00 00 00 00 ID: 0x001 Data: 01 00 00 01 00 00 00

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.

- Stage 4: Load Request and Staging Request, Staging is Complete

ID: 0x123 Data: 01 00 01 00 00 00 00 ID: 0x001 Data: 01 00 01 00 00 00 00

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.

- Stage 5: Load Request, Staging is Complete

ID: 0x123 Data: 01 00 00 00 00 00 00 ID: 0x001 Data: 01 00 00 00 00 00 00

The user can now remove the staging request and simply keep the system in active load mode by sending only the load request.

- 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 are 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.

19.5 Charge Model Cycle

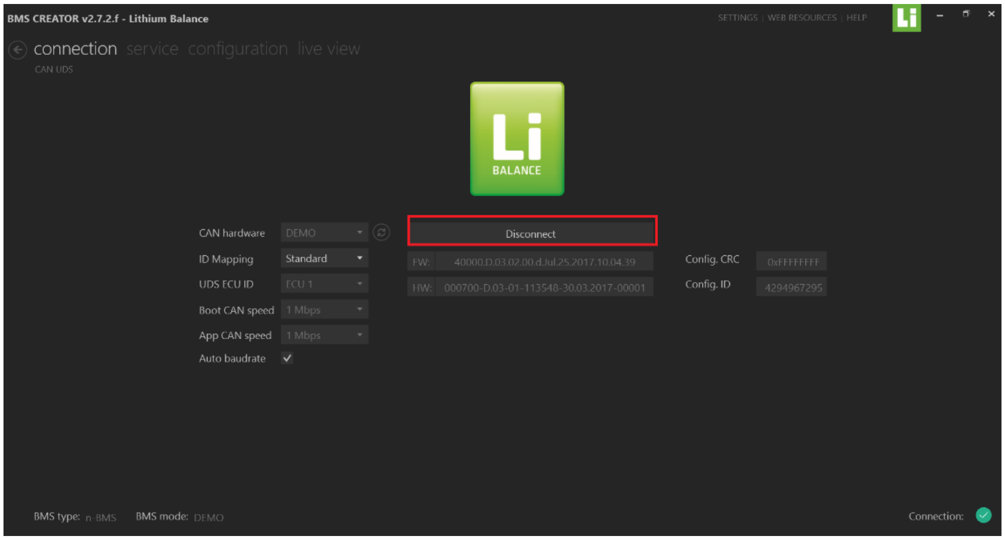
The same above six stages apply for the charge mode request, with the charge mode request data BB and FF indicated instead.

20 Appendix 8 : i-BMS15 Memory Reset

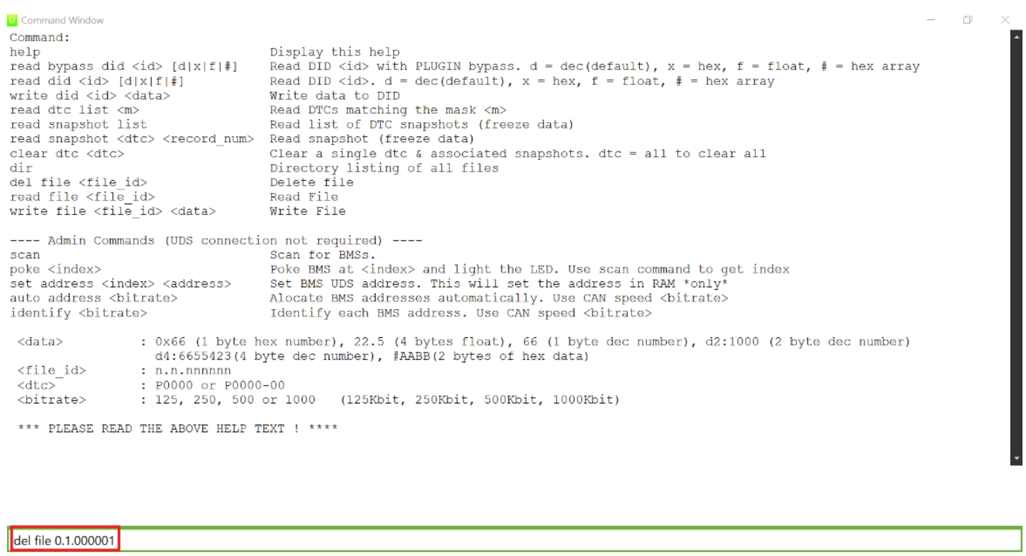
In order to reset the i-BMS15 memory with the help of Creator software, the user needs to follow the below process :

- Connect the Creator software to the BMS by ensuring correct power supply.

- Right Click on “Disconnect”



- A command window will pop up. In the command line, enter “del file 0.1.000001” and press enter and turn off i-BMS15 immediately or click ‘Rest BMS’ available under “Service” tab of BMS Creator.



- The i-BMS15 memory will be reset after this procedure is followed. The user can flash new configuration or make necessary changes to the configuration parameters after this.

21 Appendix 9 : Preliminary Check for BMS Integration

The following basic tests/checks must be performed by the user in order to achieve smooth operation during vehicle/pack level testing. It is important to understand that these checks must be done after the configuration (to be used in actual application) has been uploaded on the BMS :

- SOC Update :
  - Charge the battery fully until both the charge complete flags (Data IDs 47 & 48) are raised (i.e. becomes 1 from 0).
  - The IDs 47 & 48 can be monitored in "Custom Data" tab available under "Live View" pane of BMS Creator.

- Balancing :
  - While charging, once the cell voltages go above the configured "Balancing Lower Limit" and the cell voltages are not within the configured dead band for balancing, the "balancing allowed" flag will be enabled and upon receiving balancing command via CAN or digital input, balancing will start.
  - The user can check this on "Custom Data" tab available under "Live View" pane of BMS Creator on Data ID 52.

Note : It is very important to configure the "balancing lower limit" based on cell chemistry (SOC-OCV curve), charging time and cell capacity.

- Measurements :
  - The user must ensure the measured cell voltages, pack current and temperature are in sync with the BMS Creator configuration.
  - The number of temperature sensors enabled, number of CMUs and number of cells in series must be verified with the Live View (BMS Creator) Data.

- Error Checklist & Handling :
  - The user must observe and verify the critical errors which are configured in the Error Checklists in BMS Creator.
  - This verification can be done by creating an error scenario and observing whether the error reporting and action taken by the BMS after the error was reported are in sync with the behavior configured in BMS Creator configuration.

Note : The user must perform the above checks and prepare the BMS configuration accordingly in order to start with the basic BMS functionalities which will be important for advanced functionalities like – DCLI/DCLO limit check, I2T limit check, SOC Accuracy etc. which the user feels will be necessary for their application.

22 Appendix 10 : Parts and Ordering Codes

Part Number	Component	Comments
100916	i-BMS15 Board	-
100980	Wire harness kit for i-BMS15 without CAN termination	For standalone packs
100981	Wire harness kit for i-BMS15 with CAN termination	For parallel packs
100910.99	i-BMS Creator Software	-
000510.1	Vishay Thermistor (NTCLE100E3103JB0)	150 mm wire
000560	Fuse - 500mA quick blow	For 12 V power fuse
100538.1	Fuel Gauge	CAN - based
100545	PEAK CAN Adapter	-