User Manual

enepaq
Power In Shape

Tiny BMS

Revision D, 2025-07-04

Contents

Preface

This manual discusses the modules and peripherals of the *Tiny BMS s516* device, and how each is used to build a complete battery pack management system and protection solution. *Tiny BMS s516* device line is splitted into two different versions:

- *Tiny BMS s516-30A* – a low power *Tiny BMS PCB* version
- *Tiny BMS s516-150A/750A* – a high power *Tiny BMS PCB* version.

This document covers both *Tiny BMS* versions.

List of Abbreviations

BMS – Battery Management System
SOC – State-Of-Charge
PCB – Printed Circuit Board
FET – Field Effect Transistor
DFET – Discharge *FET*
CFET – Charge *FET*
NTC – Negative Temperature Coefficient
USB – Universal Serial Bus
UART – Universal Asynchronous Receiver / Transmitter
CAN – Controller Area Network
ASCII – American Standard Code for Information Interchange
AIDO – Analog Input Digital output
DIDO – Digital Input Digital Output
AIHO – Analog Input Hybrid Output
LED – Light emitting diode

List of Tables

List of Figures

Safety

To ensure the security of you and your surroundings, please read these rules and follow all the guidelines contained installation instructions:

- DO NOT use the system where it may cause interference (eg. medical equipment);
- DO NOT use the system in an explosive environment;
- The system is NOT resistant to the chemical environment and mechanical stress;
- All electrical equipment MUST be rated for the voltage of the battery and battery management system;
- Use double-insulated tools;
- DO NOT short-circuit the battery or battery management system terminals, this could cause the damage to the product or the personal property;
- DO NOT lay tools or metal parts on top of the batteries or near the battery management system cable lugs;
- Before establishing connections, make sure to verify polarity.

⚠	**This list is not exhaustive, and it is the responsibility of the system designer / installer to conduct their own failure mode analysis and determine what is required.**

⚠	**Working around batteries is DANGEROUS. Risk associated with improper use of the battery with the *Tiny BMS* device includes: short-circuit, fire or explosion.**

⚠	**Read the ENTIRE documentation to become familiar with the *Tiny BMS* device and its features before operating. Failure to operate the product correctly and safely may result in damage to the product, personal property and cause serious injury.**

⚠	**Throughout the literature the term "Note:" will be used to indicate procedures which, if not properly followed, create a possibility of physical property damage.**

Chapter 1: Introduction

BMS, or *Battery Management System* is an essential component of every lithium battery. *Enepaq Tiny BMS s516* device provides a feature rich battery management solution for *4*-series cells to *16*-series cells battery-pack applications. *Tiny BMS* measures individual voltages of parallel cell groups and manages switching of load and charger. Besides, *BMS* measures battery-pack current and estimates its *State-of-Charge* and *State-of-Health* (from 2.3.251 firmware version). Battery capacity from sub-*1 Ah* to *655 Ah* can be managed easily. Various safety mechanisms are built in: disconnection of load and charger if any of the cells go out of allowed range of voltage, current or temperature. User-upgradeable firmware allows quick updates, bringing bugfixes, new features and other improvements, such as client-specific functionality. Integrated communication bus allows easy reconfiguration and streaming of real time data, while extension *I/Os* enables operation with various external peripherals: current sensors, contactors, external switches, *SOC* gauges and more.

This document covers all aspects of using the *Tiny BMS s516* to provide the user with an in-depth knowledge about its hardware and software functionality and possible uses, as well as with information about its proper installation and configuration together with connection diagrams, examples, and recommendations – all in order to help utilize all of its capabilities and prevent most common faced issues.

Note: Using *Tiny BMS s516* requires at least basic knowledge in electronics and electrical engineering. Any use of the device that does not comply with the instructions given in this document is considered improper and no warranty is provided. *Enepaq* will not be held responsible for damage to the battery or any other consequences in case the *Tiny BMS s516* is used improperly, and reserves the right to not provide any technical support in such case.

Chapter 2: System hardware structure overview

2. Introduction

Enepaq Tiny BMS is a very small dimensions *PCB* board, which monitors main battery-pack parameters and performs all control, safety, protection and *SOC* gauging tasks. *Tiny BMS* is equipped with the onboard connectors for easy configuration and different setup, which is suitable for many types of battery management applications. *Tiny BMS* is suitable for both dual port operation mode (separate control switches for load and charger) and for single port operation mode (the same main control switch for load and charger). For small loads, *BMS* can use internal *FET* control switch and, if higher current capability is needed, external current sensor and relays / contactors, or another compatible circuitry can be used. The relays / contactors can be connected to any of these programmable extension *I/Os*: AIHO1, AIHO2, AIDO1, AIDO2, DIDO1, DIDO2. All of these are multi-purpose and can be used either as inputs or outputs. *Tiny BMS* supports up to two external *NTC* temperature sensors or up to *32 Enepaq* multipoint active temperature sensors available in the *Enepaq Cell Modules*. *Tiny BMS* supports *USB-UART*, *CAN*, *BLUETOOTH* communication interfaces for system configuration, *BMS* live data monitoring and firmware updating. This chapter in detail covers all aspects of connecting, configuring and using *Tiny BMS* for different type of battery management applications.

2.1. *Tiny BMS s516* hardware overview

There are two hardware versions of Tiny BMS released: v2.1 and v2.2. The differences between two revisions of Tiny BMS hardware design are summarized in Table 2.1. All the PCB versions of the Tiny BMS has the same firmware features and functionality. More detailed information about each revision is provided in the next chapters respectively. To identify version of Tiny BMS, there is a label on PCB board. Although, each *BMS* hardware has two *BMS* power type variants. Both *Tiny BMS* power types shares the same *PCB*, but low power *BMS* version lacks some hardware and firmware features. The differences between *Tiny BMS* power modes are summarizes in *Table 2.2*.

Table 2.1: Tiny BMS hardware overview

Tiny BMS hardware	v2.1	v2.2
Max. cells balancing current (*mA*)	*300*	*150*
L+ / *C+* solder-pads	➕	➖
Onboard load precharge circuit	➕ (not used)	➖
5 V power pins on *Digital* / *Analog IO* connector	*4*	*4*
Onboard load switch current (continuous)	*60*	*125*
Onboard charge switch current (continuous)	*30*	*90*

Table 2.2: Differences between Tiny BMS power types

Tiny BMS power type	High Power	Low Power
External current sensor support	➕	➖
Onboard current sensor range for v2.1	*+/- 150 A*	*+/- 60 A*
Onboard current sensor range for v2.2	*+/- 200 A*	*+/- 100 A*
AIDO inputs / outputs	➕	➕ (only inputs)
DIDO inputs / outputs	➕	➕ (only inputs)
AIHO inputs / outputs	➕	➖

Opto-isolated output	⊕ (not used)	⊖
BMS is mounted on cool metal surface	⊕	⊕

2.2. *Tiny BMS s516 PCB* board detailed description
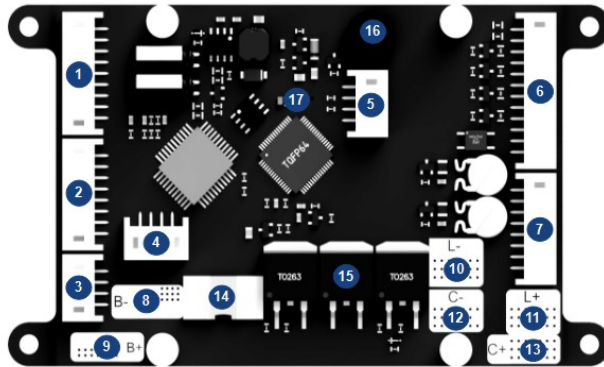
Figure 2.1: Tiny BMS v2.1 high power PCB board
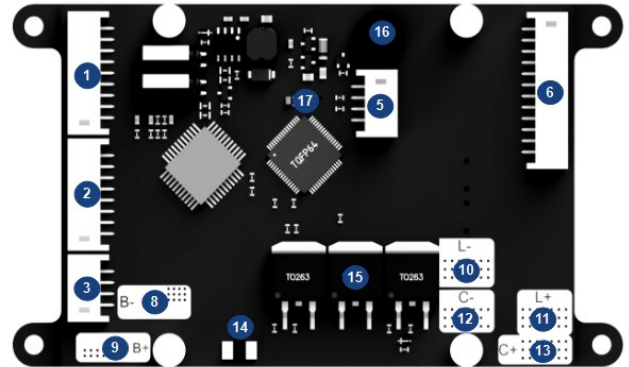
Figure 2.2: Tiny BMS v2.1 low power PCB board
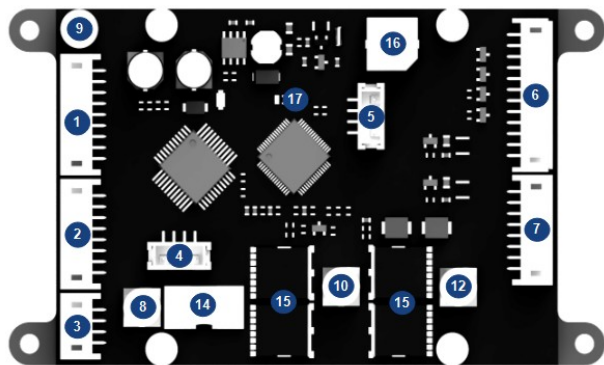
Figure 2.3: Tiny BMS v2.2 high power PCB board
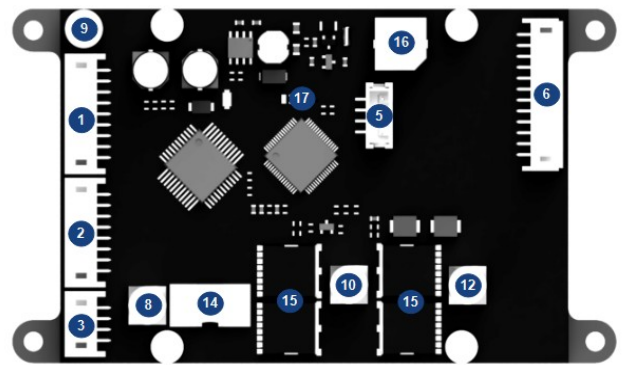
Figure 2.4: Tiny BMS v2.2 low power PCB board

Table 2.3: Tiny BMS PCB boards overview

Tiny BMS PCB boards (Figure *2.1* to Figure *2.4*)
1 Upper *8* to *16* cells connector, used to connect cells to *BMS*, enabling *BMS* to measure and balance individual cells block of battery pack.
2 Lower *1* to *7* cells connector, used to connect cells to *BMS*, enabling *BMS* to measure and balance individual cells block of battery pack.
3 External two channels temperature sensor connector. *BMS* supports one *NTC* sensors or up to sixteen *Enepaq* multipoint active temperature sensors available in the *Enepaq Cell Modules* per channel.
4 External *LEM* current sensor connector. Not available in the low power v2.1 and v2.2 versions of the *Tiny BMS.*
5 *Tiny BMS* communication interface connector. *USB-UART* cable, *UART-CAN* module, *LED SOC-BAR* and *BLUETOOTH* adapter can be connected.
6 Digital or analog inputs and outputs connector. For low power *v2.1 and v2.2* versions of *Tiny BMS* only inputs are available.
7 Hybrid inputs and outputs connector. Not available in the low power *v2.1 and v2.2* versions of the *Tiny BMS.*
8 Solder-pad (or terminal in version 2.2) for battery pack negative terminal (*B-*). Thick wire from battery pack negative terminal should be soldered or screwed (dependent of TinyBMS version), because high discharging and charging current will flow through this solder-pad, if internal *BMS FET* switch is used.

9	Solder-pad (or terminal in version 2.2) for battery pack positive terminal (*B+*). For *Tiny BMS* hardware version *2.1* (*Figure 2.1* to *Figure 2.2*) this solder-pad can be used as high current path from battery pack positive terminal (+) to load and charger positive terminals (+). Solder-pad *B+* is directly connected to *L+* **11** and *C+* **13** solder-pads on the *BMS* board, but beware: only *30 A* continues current can flow through this on board positive current path, otherwise there is a high probability to damage the *BMS* board itself. It is not recommended to use *L+* **11** and *C+* **13** solder-pads on high power versions of *Tiny BMS*. *B+* also supplies current to *BMS* internal circuits, the thinner wire can be used to connect battery pack positive terminal (+) to *BMS* board *B+* solder-pad.
10	Solder-pad (or terminal in version 2.2) for load negative terminal (*L-*).
11	Solder-pad for load positive terminal (*L+*). **Note:** *L+* solder-pad (2.1v) only can be used if discharging current is under *30 A*!
12	Solder-pad (or terminal in version 2.2) for charger negative terminal (*C-*).
13	Solder-pad for charger positive terminal (*C+*). **Note:** *C+* solder-pad (2.1) only can be used if charging current is under *30 A*!
14	Internal current sensor.
15	Internal *FET* switch for load and charger.
16	*Tiny BMS* activity sound indicator.
17	*Tiny BMS* activity *LED* indicator.

2.3. *Tiny BMS* connectors and wiring

Tiny BMS onboard connectors and internal *FET* power switch wiring are shown in *Figure 2.5*. and *Figure 2.6*. All connectors on the *Tiny BMS* board are compatible for *JST PH* series *2.0 mm* pitch type connector or any other compatible connector.
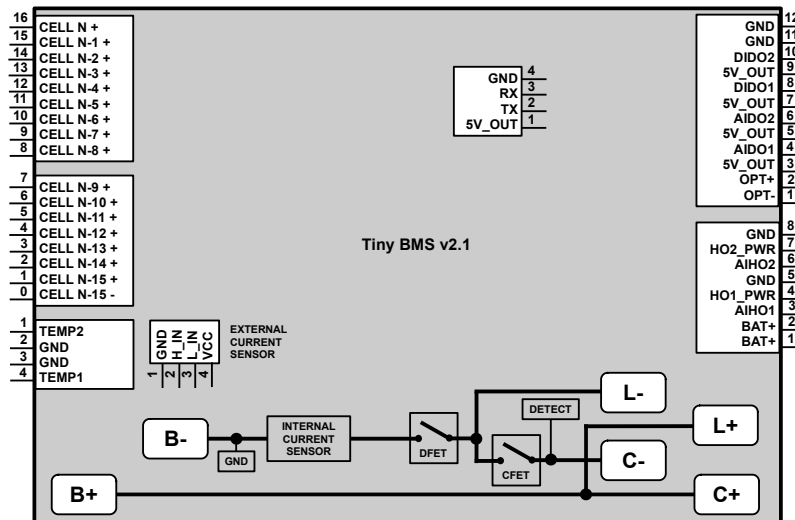
Figure 2.5: Tiny BMS v2.1 onboard connectors pinout
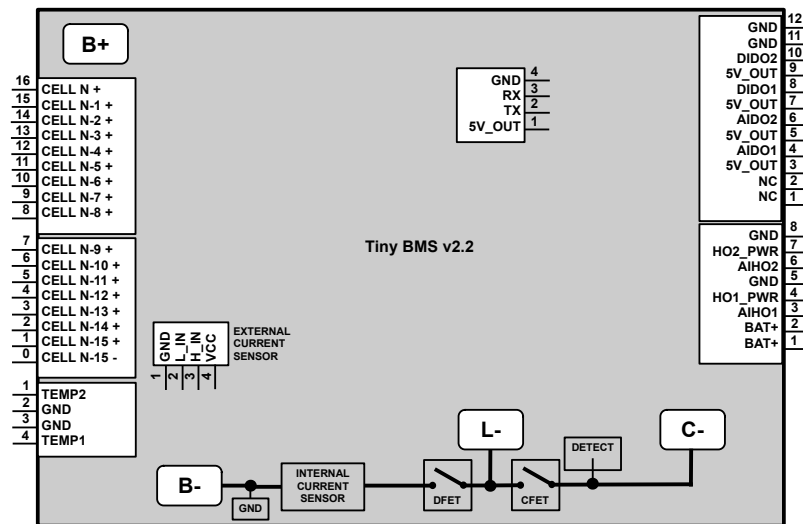
Figure 2.6: Tiny BMS v2.2 onboard connectors pinout

2.3.1. Cells voltages measurement and balancing connectors

There is a *9* pin connector ❶ for uppermost battery pack cells and the *8* pin connector ❷ for lowermost battery pack cells. Cell count starts from most lowest cell of the battery pack negative terminal (-). In *Figure 2.5* and *2.6* cells connector numbering is shown for battery pack made of *16* in series connected cells. If less in series connected cells are needed, the lowest cell connector pins should be skipped and left unconnected. The positive terminal of the highest cell in battery pack must be connected to *16-th* pin on the *Tiny BMS* connector *Pin 16 CELL N +*. Below is a formula to determine which *Tiny BMS* cell connector pin should be used to connect cell's positive terminals (+):

$$Pin\,Number = 16 - N + n;$$

where N – count of series cells in the battery pack, n – cell number for which the *BMS* connector pin is calculated (numbering starts from bottom battery most lower cell). For the battery pack most lower cell positive terminal (+) – $n=1$, for the battery pack most lower cell negative terminal (-) – $n=0$. For an example lets take that battery pack, assembled from *10* cells connected in series. According to the above formula *1-st* cell's (-) should be connected to the pin *6* of the *Tiny BMS* cells connector (*16-10+0=6*), *1-st* cell's (+) should be connected to the pin *7* of the cells connector (*16-10+1=7*) and so on, until you reached cell's *10-th* (+), which should be connected to the pin *16* of the cells connector (*16-10+10=16*). Be aware, that connection sequence is important. For *Tiny BMS* connection sequence please refer to the next chapter: *2.3.2. Tiny BMS battery pack connection sequence.*

Note: *Tiny BMS* cells connector pin *0* is not directly connected to battery pack negative terminal (-), *B*-solder-pad on the *Tiny BMS* board, therefore if lowest cell (-) / battery pack (-) is not connected to *Tiny BMS* cells connector, battery pack lowest cell voltage will be measured lower by *0.6 – 0.7 V*.

2.3.2. *Tiny BMS* battery pack connection sequence

Tiny BMS connection sequence to the battery pack is important. *Tiny BMS* is supplied from *B-* and *B+* terminals on the board and incorrect connection can lead to permanent damage of the board. Main rule is to avoid balancing wire connection without *B-* and *B+* terminals being connected. When disconnecting from battery, disconnect balancing wires first, only then followed by *B+* and *B-*. To properly connect battery pack to *Tiny BMS* should be followed by these steps.

Step 1: Preparing battery pack wiring harness.

Firstly, battery pack wiring harness should be prepared while nothing is connected to *BMS*. For proper *BMS* connection to battery pack several different cables required:

- *N* (number of series cells in the battery pack) + *1* copper wires *35 AWG* (*9* and *8* pin wire assembly kits);
- one copper wire *35 AWG* for *BMS* supply solder-pad *B+*;
- one main high copper current wire from battery pack positive terminal (+);
- two high current copper wires to branch battery pack positive terminal (+) to charger and load positive terminals (+);
- one high current copper wire from battery pack negative terminal (-) to *Tiny BMS* board *B-* solder-pad.

If battery pack assembled from *8* or less in series connected cells, only *9* pin wire assembly kit is needed, otherwise one *9* pin and one *8* pin wire assembly kits are needed.

Step 2: Remove unused wires from wire assembly.

For less than *16* cells, connected in series, there are few unused cells wires and it is recommended to remove them. *Figure 2.7* depicts how to remove unused wires from connector housing. Before doing following steps make sure that wiring assembly isn't connected to anything. Using tweezers carefully lift plastic fin, that holds contact in place. While holding fin lifted – pull wire out of connector housing.
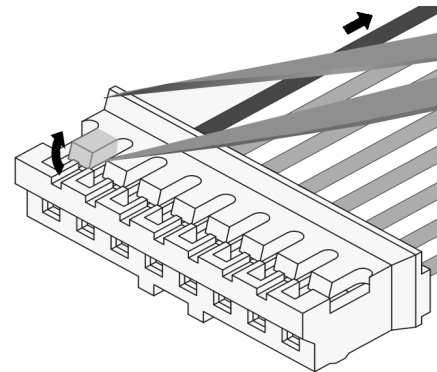
Step 3: Connect loose end of wiring assemblies to battery pack.

Make sure cells wire assemblies are disconnected from *BMS* cells connectors and solder loose wires of wiring assemblies to battery pack cells.

Figure 2.7: Remove wires from wire assembly

Step 4: Check, if cells connected in correct sequence in wire assembly connectors.

When in series cells groups of the battery pack has been connected to wire assembly connectors, it is strongly recommended to check, if the cells connected in right order. Connector in wire assemblies has small contacts exposed and the multimeter can be used to measure cells voltages on these contacts. Be very careful not to short multimeter probes while measuring, this could lead to damage of battery. Voltage on adjacent connector pins should be equal to cell voltage and always same sign (must to keep multimeter probes with the same polarity). If the measured voltage between adjacent pins are two or more times the cell voltage, or with different sign, cells connection order is mixed up and must be fixed, before connected to *BMS* device.

Step 5: Solder high current wires to battery pack.

Solder/screw thick cooper wire to battery pack negative terminal (-). Also connect one thin wire (*35 AWG* is enough) to battery pack positive terminal (+). Now it should contain two wires connected to battery pack negative terminal (-): one thin wire for cell voltage measurement and one thick wire for charger and load negative terminals (-). Also there should be two thin wires connected to battery pack positive terminal (+): one for cell voltage measurement and one for *Tiny BMS* supply voltage *B+* solder-pad/terminal. Later it should be added thick wires from battery positive terminal (+) to charger and load positive terminals (+). Example of wiring harness for *12* cells in series battery pack is shown in *Figure 2.8*.
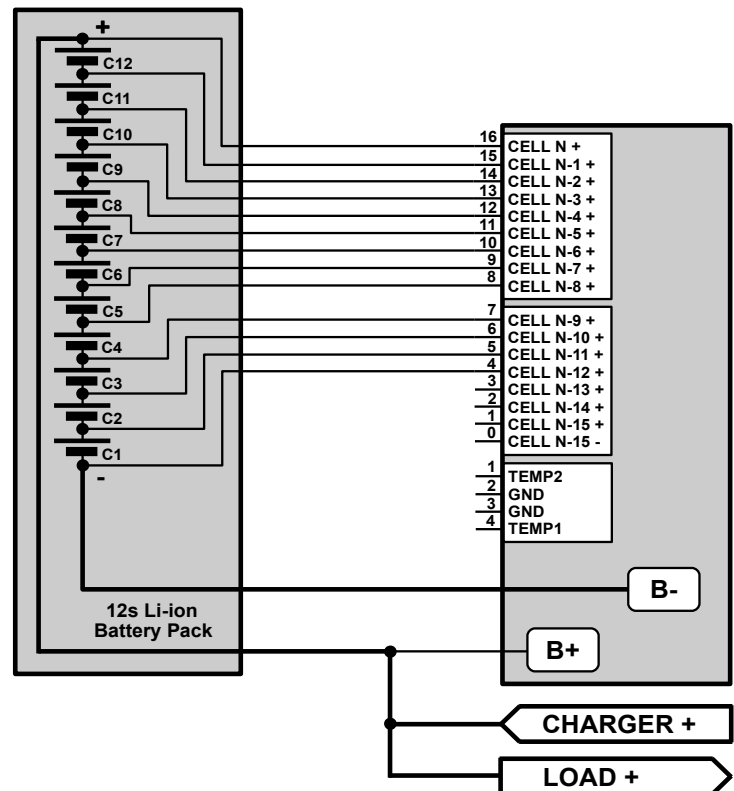
Figure 2.8: A typical 12s battery pack connection diagram

Step 6: Connect battery pack negative terminal (-) to *BMS*.

Solder or screw loose end of thick wire, coming from battery pack (-) to *Tiny BMS B-* solder-pad or terminal on the board.

Step 7: Connect battery pack positive terminal (+) to *BMS*.

Solder or screw loose end of thin wire, coming from battery pack (+) to *Tiny BMS B+* solder-pad or terminal on the board. *BMS* should beep and blink the on board *LED*. When connecting battery pack with voltage, higher than *50 V*, *BMS* may be damaged by high voltage spike in moment of connecting to *B+* solder-pad. It is strongly recommended first to connect battery pack positive terminal (+) to *Tiny BMS B+* solder-pad or terminal through *50 - 100* Ω resistor and when *BMS* starts, resistor must by bypassed with direct connection and then removed, leaving direct connection to the battery pack. Make sure main power lines connections are strong and contain no fuses or switches.

Step 8: Plug 8 pin connector of lower cells wiring assembly.

If battery pack is assembled from more than *8* cells in series there should be prepared two cells wiring assemblies: *9* pin for higher cells (relative to battery pack negative terminal (-)), and one *8* pin for lower cells. If *8* or less in series connected cells are used, it is only be needed *9* pin wiring assembly. If *8* pin wire assembly is used, plug connector of this assembly into *8* pin socket on *Tiny BMS* board first. If *8* pin wire assembly is not used, leave *8* pin socket on *Tiny BMS* board empty and go directly to *Step 9*.

Step 9: Plug *9* pin connector of higher cells wiring assembly.

Plug *9* pin connector of higher cells wire assembly in to *Tiny BMS 9* pin cells socket on the board.

2.3.3. External temperature sensor connector

Tiny BMS has one on board *NTC* temperature sensor and supports two more external *NTC* temperature sensors (each per channel) or up to sixteen *Enepaq* multipoint active temperature sensors per *BMS* channel available in the *Enepaq Cell Modules*. Used temperature sensor type should be configured on *Battery Insider* windows application (see *Chapter 4*). Note that temperature sensor type configuration applied to both *BMS* temperature channels and different temperature sensor types cannot be used at the same time on different channels. Unused temperature sensor channel should be left unconnected. Once temperature sensor connected to *BMS*, it is recognized automatically and after few seconds measured temperature values are displayed on the *Battery Insider* application *Live Data* tab (see *Chapter 4*).

2.3.4. Dual *NTC* temperature sensor

Supported *10 K @ 25 °C* (*Beta* value *3977 K*) *NTC* temperature sensors. *NTC* thermistor should be connected between signal and ground pins of external temperature sensor connector.
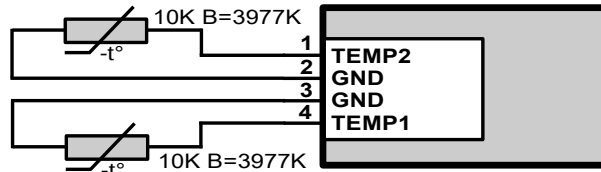
Figure 2.9: Dual NTC temperature sensor

2.3.5. *Enepaq* multipoint active temperature sensor

Enepaq multipoint active temperature sensor available in the *Enepaq Cell Modules* has up to four *NTC* sensors inside. Innovative analogue signal *OR'ing* technique allows all *4* sensor signals to be read with two-wire acquisition system (one *Tiny BMS* temperature channel): output acts as a hot spot detector and reports only the maximum temperature. When battery is operating within safe limits, all four sensors report similar temperatures and such measurement accurately represents overall temperature of the module. However, in case of failure event, hot spot is very quickly noticed. The sensor is a special-made temperature-variable voltage shunt reference. In simple words, it acts as a zener diode, whose voltage drop depends on temperature. For v2.1 it requires a pull-up resistor to operate at *BMS* 5 V output level. As already mentioned, *Tiny BMS* can support up to *16 Enepaq* multipoint active temperature sensors per channel (in case that one sensors contains all *4* sensor nodes). It means that one *BMS* temperature channel is capable to measure altogether *64* temperature sensing nodes (hotspots). To find out how temperature sensing nodes contains each *Enepaq Cell Module* (one multipoint temperature sensor) and determine how many modules can be connected to *BMS* temperature channels please refer to *Enepaq Cell Moules* datasheet. A typical *Enepaq* multipoint active temperature sensor connection diagram is shown in *Figure 2.10*. and *2.11.*

Temperature sensor pull-up resistors *R1* and *R2* values *330 R* (*Figure 2.10*) has been calculated for *16* multipoint sensors with all *4* temperature sensor nodes (altogether *64* temperature nodes) per *BMS* external temperature channel and temperature range *-40 °C* to *120 °C*. When a smaller number of sensors is connected to the *BMS* it is suggested to adjust the pull-up resistors accordingly. Below is a formula to determine maximum allowed *R1* and *R2* pull - up resistors resistance:

$$R_{max} = \frac{2.5}{10^{-4} \cdot N};$$

where, *N* – number of temperature sensing nodes, connected to one *BMS* temperature measurement channel. It is recommended to use calculated resistors *R1* and *R2* values with resistance decreased by *30 %*. As an example, module *Li4P25RT* has *2* temperature sensor nodes and let say *12* such modules need to be connected to *BMS*. In this case *6* cell modules (*12* temperatures sensor nodes) should be connected to each *BMS* external temperature sensor channels. Calculated *R1* and *R2* resistors values *R=2.5/(10⁻⁴·12)=2038 Ω*. It's safe to select *1500 Ω* or *1800 Ω* resistors values. Lower resistance will lead to

more current consumption by the temperature measurement nodes. Maximal current, that flows through pull-up resistor:

$$I = N \cdot \frac{3.8}{R};$$ and the *R1* and *R2* resistors must be rated for power: $$P = \frac{14.44 \cdot N^2}{R}.$$
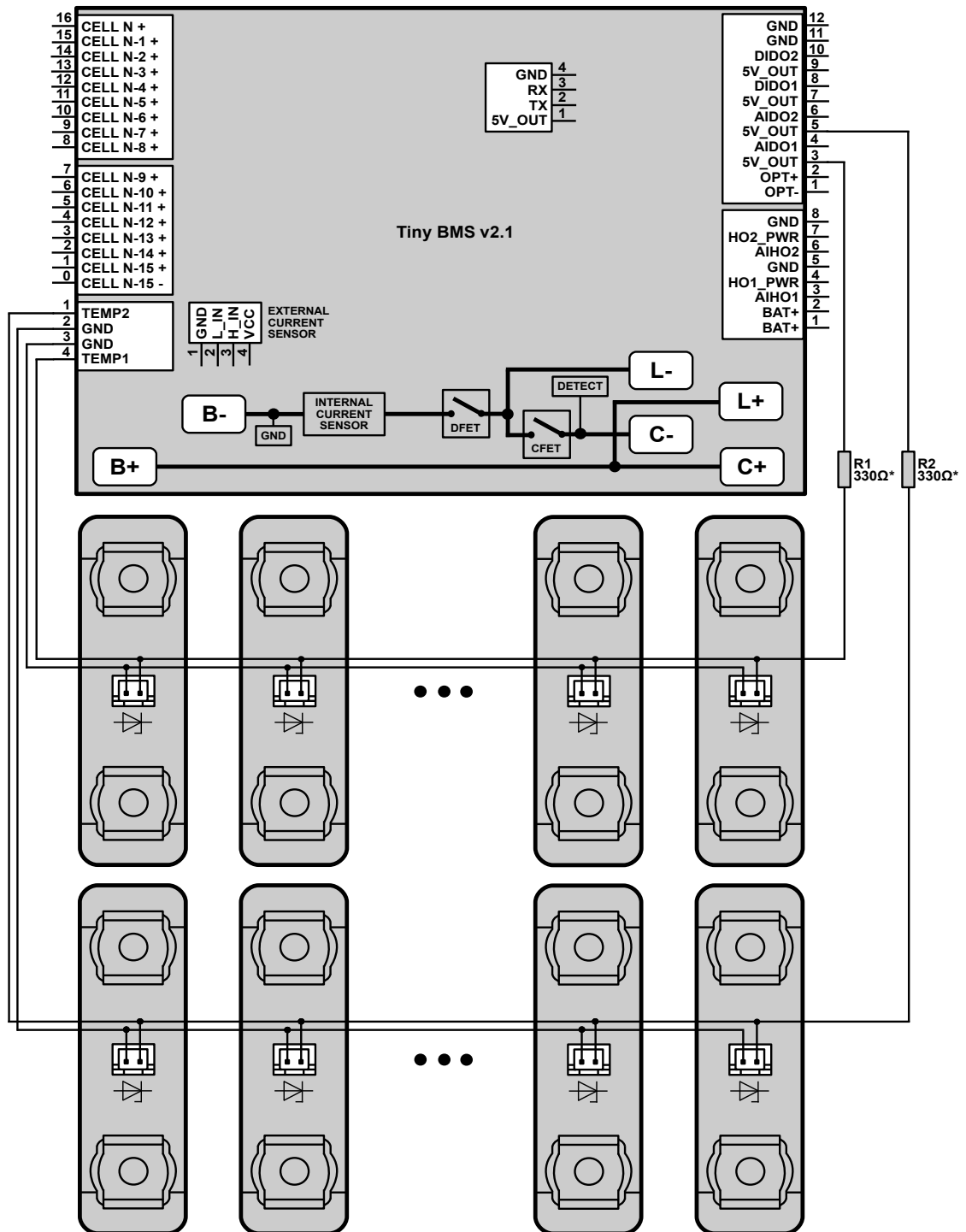
Figure 2.10: Enepaq multipoint active temperature sensor connection diagram for v2.1
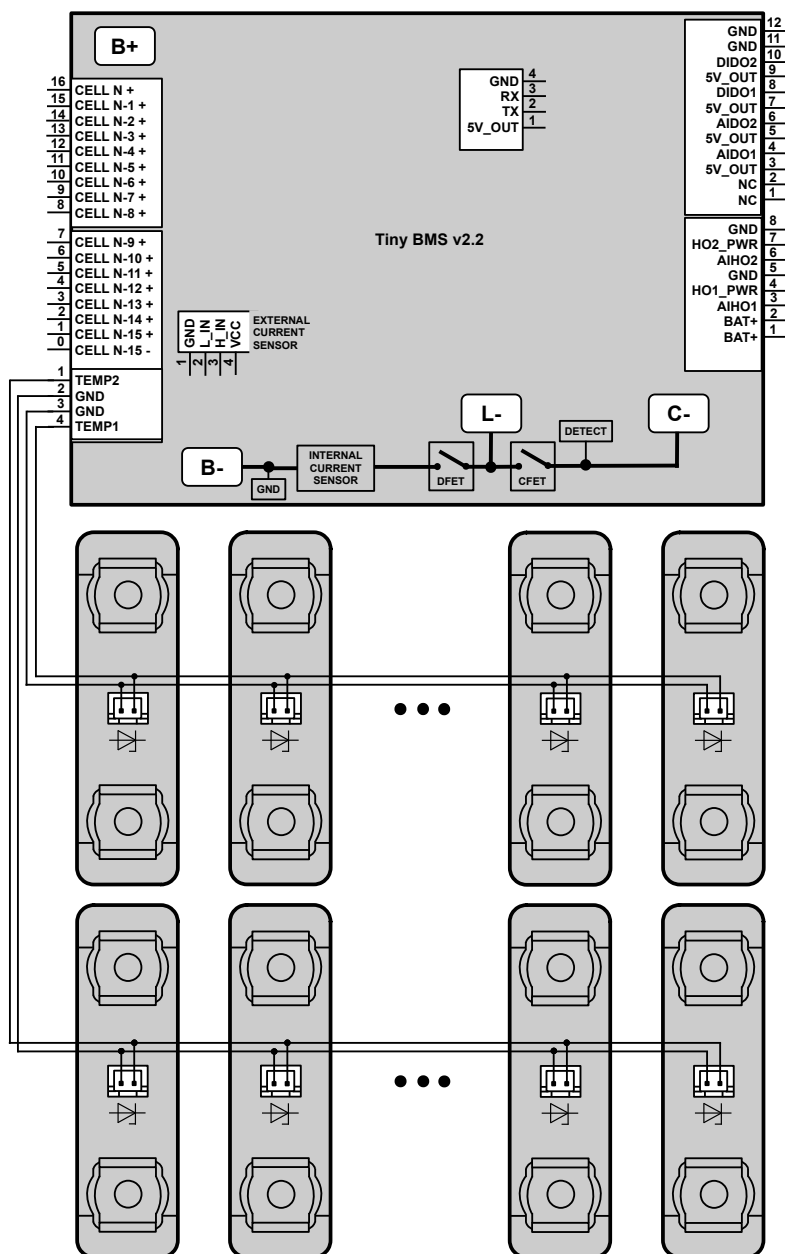
Figure 2.11: Enepaq multipoint active temperature sensor connection diagram for v2.2

2.3.6. External current sensor connector

Tiny BMS v2.1 can hold *150 A* discharging current just for very short pulse current spikes (*Table 2.1*).To support higher current, use version 2.2 or add an external current sensor and contactors. Currently supported current sensor is *LEM DHAB S- 133*, which is a dual-range sensor, with measurement range up to *±750 A*. Lower range measures up to *±75 A* and is used for higher accuracy measurement of low current. Switching of ranges is done automatically by *Tiny BMS* itself. External current sensor to *BMS* system should be connected once at the setup stage of the battery pack.

Figure 2.12: External LEM current sensor

Note: External current sensor is critical *BMS* and battery application component. After external current sensor has been connected to *BMS*, *BMS* must be restarted by the user to confirm that external current sensor is used for battery application, otherwise *BMS* internal current sensor will be used for current measurement. For complete guide how to properly connect and use external current sensor refer to *Chapter 4.*

A typical external *LEM* current sensor connection diagram is shown in *Figure 2.13.* External current sensor should be installed in such way that all currents going in and out of the battery would flow through it (i.e. behind all load, charging, or utility devices that are connected to the battery). If *Enepaq LEM* current sensor kit is used, simply plug current sensor connector into *Tiny BMS* external current sensor connector, otherwise when custom wiring assembly is needed, wiring should be built according to *Figure 2.13* and *LEM DHAB S-133* datasheet.
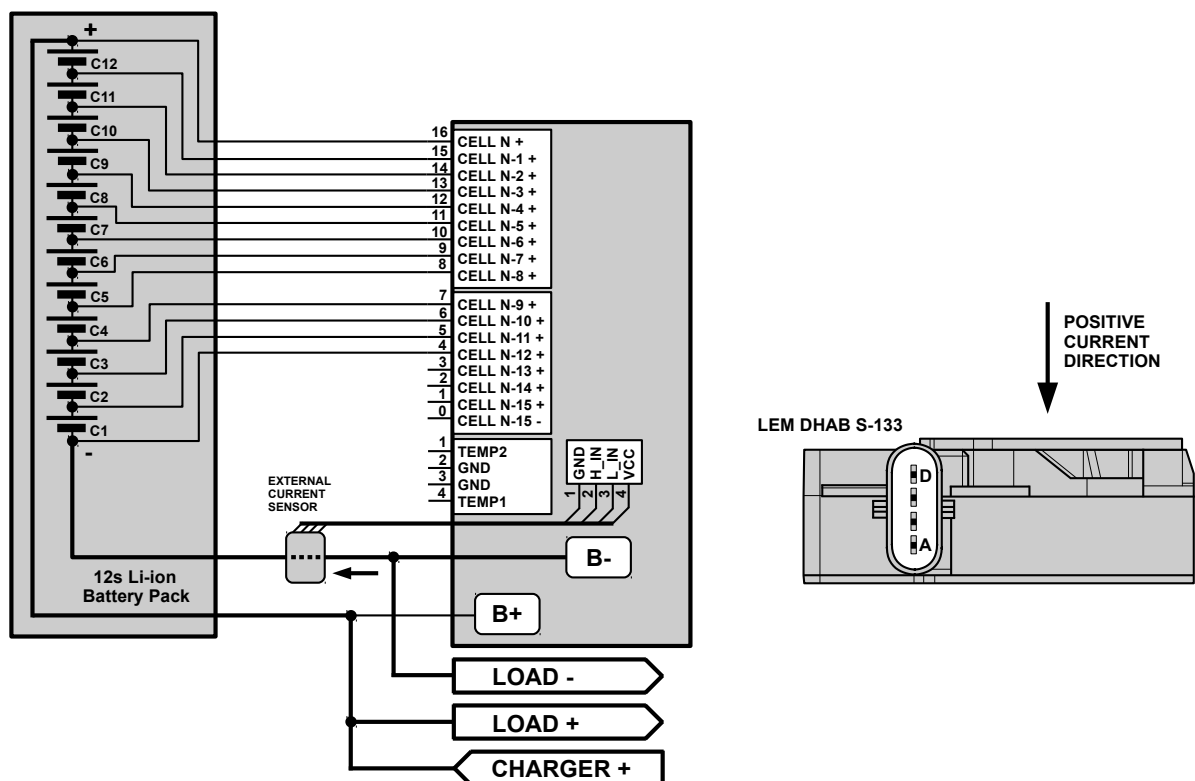
Figure 2.13: External LEM current sensor connection diagram

Connector part number, that mates with *Tiny BMS* board is *PHR-4* made by *JST* and connector, that mates with *LEM* current sensor, part number is *1-1456426-5* made *Tyco Electronics*. When connecting external current sensor attention must be taken to the positive current direction, providing that high current wire is passed through current sensor in correct direction. *Tiny BMS* treats positive current from battery positive terminal to load / charger positive terminal.

Note: External current sensor should only be used along with external relay / contactor for load or charger control, when high current capability is needed. When external relay / contactor is used for load / charger switching high current path bypasses the *Tiny BMS* board and the internal current measurement is not possible.

2.3.7. Communication connector

Tiny BMS includes a single multi-purpose *UART* interface. In combination with various converters, the interface is compatible with *USB*, *Bluetooth* and *CAN* communication interfaces. Various proprietary commands are available for quick and straightforward connection to a *PC Battery Insider* application when configuration, diagnostics, or maintenance is needed. Also the main *MODBUS* protocol commands are supported for rapid integration to existing industrial systems. For more information about communication protocols refer to *Tiny BMS communication protocols* documentation.

Figure 2.14: Tiny BMS USB–UART communication cable

UART communication interface is the main *Tiny BMS* communication, enabled by default. Using *Tiny BMS UART* commands it is possible to setup all settings, read back and monitor live data, events, statistics, wakeup *BMS* from sleep mode and all other tasks that *BMS* are intended to do for *PC* and custom systems. If *USB-UART* cable is used for *Tiny BMS* connection to a *Windows Battery Insider* application, it is no need any external steps of configuration, just plug *USB-UART* cable to *Tiny BMS* communication connector and *Battery Insider* automatically connects to *BMS*, even if *BMS* is in deep sleep mode. *Tiny BMS USB-UART* converter is based on genuine *Silicon Labs CP2102* chipset. All recent *Windows* operating systems already include the drivers and cable will work straight away. If there is a problem, it is possible to download a driver version that is confirmed as compatible with *Tiny BMS* from *Enepaq* web store. Alternatively, driver can be downloaded directly from *Silicon Labs* support website, just make sure to get *v6.7.3* or newer if you are running *Windows 7* or later *OS*. When cable is connected, it will appear as a virtual *COM* port and *Battery Insider* application will connect to it automatically. When *USB- UART* communication is not necessary, any other communication module or *SOC-BAR LED* indicator can be connected to *BMS* communication interface connector. Only one *Tiny BMS* communication extension module can be connected to *Tiny BMS* at the same moment.

Note: The *USB-UART* cable is not galvanically isolated. Take spacial care and give your full attention to avoid ground loops and potential damage to equipment. Alternatively, use *USB* isolator.

Note: .From June 2024, the USB-UART cable includes an additional USB isolator.

Tiny *BMS Bluetooth* communication module is a wireless direct solution for replacement of wired *USB-UART* cable for parameter viewing, configuration and data streaming to *Android* dashboard application (*CAV3* protocol). All standard *UART* commands are supported directly, any extra configuration is not required. *CAV3* protocol data streaming can be enabled in the *Battery Insider* application. For more detail about how to configure *BMS* data broadcast refer to *Chapter 4*.
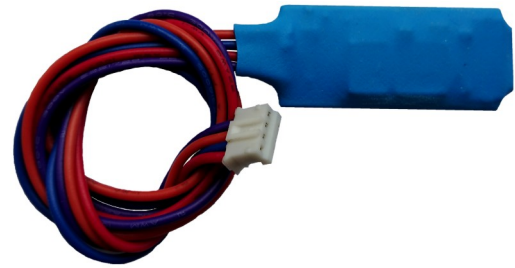
Figure 2.15: Tiny BMS Bluetooth communication module

Note: *Tiny BMS* currently not supported firmware update over *Bluetooth* communication feature. *BMS* firmware can only be updated over *USB-UART* cable using *Battery Insider* application.

Note: *Tiny BMS Bluetooth* module, *SOC-BAR LED* indicator and *CAN* communication module only works when *BMS* is in active mode (*Idle*, *Charing* and *Discharging*), because in sleep mode *BMS* turns off its internal *5 V* power supply to achieve the lowest power consumption and there is no *5 V* output on the *Tiny BMS* communication connector. It means that *BMS* can not wake up from sleep mode by *Bluetooth* or *CAN* communication commands.

Note: All communication modules and indicators once configured and enabled can be changed on the fly, there is no need to restart the *BMS* or any other action required, just unplug unused module (for example *USB-UART* cable) and connect the new module instead (for example *SOC-BAR LED* indicator).

SOC-BAR LED indicator is useful to monitor the battery pack *State-Of-Charge* (*SOC*) in action. Indicator connects to the same *Tiny BMS* communication interface connector. It shows the percentage representation of *BMS* calculated *SOC* value of the battery pack. Before using *SOC-BAR LED* indicator *Tiny BMS SOC-BAR* broadcast protocol must be configured in the *Battery Insider* application. How to properly enable and configure the *Tiny BMS* for data broadcasting refer to *Chapter 4*.
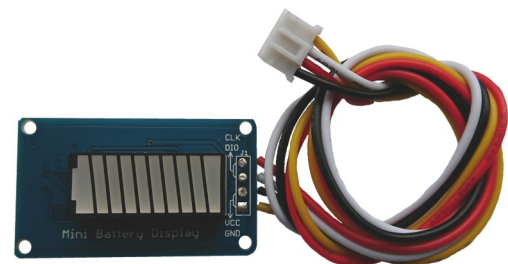
Figure 2.16: Tiny BMS SOC-BAR LED indicator

Enepaq UART-CAN communication module is a very compact isolated *Tiny BMS UART* signals to *CAN* bus converter. It gives a possibility to connect *Tiny BMS* with other industrial equipment, robotics, solar controllers, where the live data from *BMS* is necessary. The module consumes ultra low power in sleep mode, when no data received on *CAN* bus or *UART* interface and instant wake up when data received. Like it was mentioned before, communication only works when *BMS* is in active state, otherwise *BMS* itself turns off *UART-CAN* converter to achieve lowest power consumption of all battery system.

Figure 2.17: Tiny BMS UART-CAN communication module

For more about *UART-CAN* module specifications refer to product datasheet. *UART-CAN* module firmware is user upgradeable and allows quick updates, bug fixes, new features and other improvements such as client-specific functionality, which allows to connect *Tiny BMS* to any other embedded system and gives an instant *CAN* connectivity. How to update *UART-CAN* firmware using *Battery Insider* refer to *Chapter 5*. Like

any other *CAN* bus device it needs *120 R* termination resistor at the end of the *CAN* bus. Also it is very important that *UART RX* and *TX* signals from *Tiny BMS* communication connector are cross wired to *UART-CAN* converter *RX* and *TX* pins. All possible *Tiny BMS* communication extension modules connection diagram is shown in *Figure 2.18*.
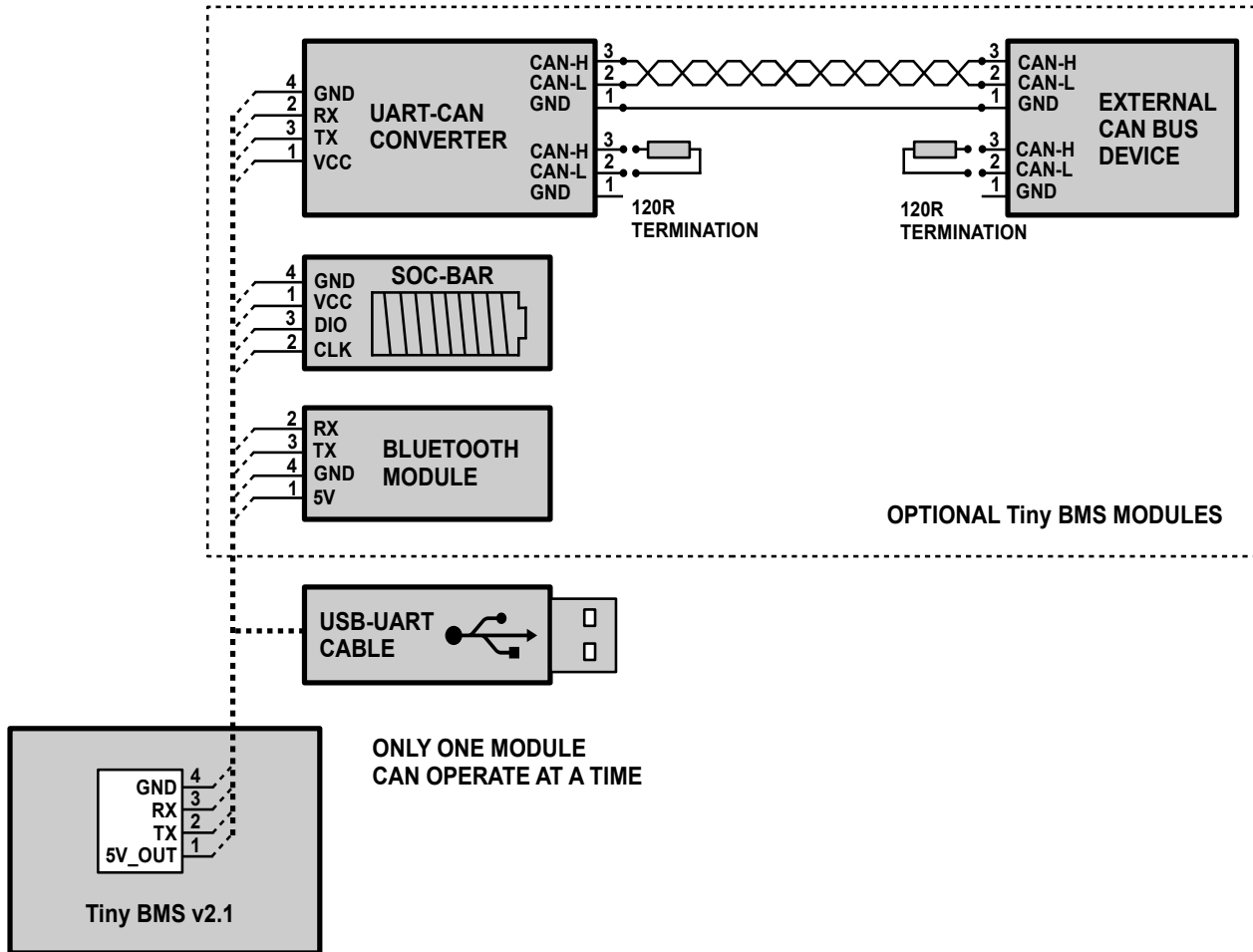
Figure 2.18: Tiny BMS communication modules connection diagram

2.3.8. Analog / Digital and Hybrid *IO* connectors

Tiny BMS has external analog / digital and hybrid input / output pins, which are multi-purpose and can be used either as inputs or outputs (*Figure 2.5* and *Figure 2.6 Tiny BMS* pinouts). Each pin can be mapped to different *BMS* features. External *BMS I/O* pins as inputs can be programmed to act as ignition switch, external charger detection input or speed measurement input from motor or other sensor. The *BMS I/O* pins as outputs can be programmed to control external load, charger and precharge relays / contactors or another compatible circuitry for switching of high current. These external *I/O* are divided into three groups: analog *I/O* (*AIDO 1/2*), digital *I/O* (*DIDO 1/2*) and hybrid *I/O* (*AIHO 1/2*). *AIDO I/O*'s can measure analog input voltage from interval: battery minus *B-* to battery plus *B+*. If *AIDO I/O* is configured as output, it can sink up to *250 mA* current. *DIDO I/O*'s can tolerate input voltages from interval *B-* to *B+*, but logic state voltage levels are same as *3.3 V CMOS* (for actual levels refer to *Tiny BMS* datasheet). *DIDO* also can sink up to *250 mA* current, if configured as output. *AIHO I/O*'s can measure input voltage from interval *B-* to *B+*. If configured as output, *AIHO I/O* can supply up to *250 mA* current from *AIHO* supply pin to *AIHO I/O* pin. *AIHO* may be supplied with voltages from interval *B-* to *B+*. Because of *AIHO* supply pin, *AIHO* output voltage levels depends on supplied voltage on *AIHO* supply pin. *AIHO* also can sink up to *250 mA* current. *Table 2.3* shows which *BMS* features can be mapped with each *BMS* exernal *I/O* pin. More information

about how to setup external pins for specific features refer to *Chapter 4*. Also all *BMS* hardware versions have optocoupler output left for future features or client specific functionality. Optocoupler output collector and emiter of optocoupler's transistor side are available. By using common emitter or emitter repeater topology selected signal polarity may be designed. External circuit should have pull-up resistor and should supply collector current.

> **Note:** Currently optocoupler output is not implemented and is not assigned to any firmware features. It is reserved for future or client specific functionality.
>
> **Note:** External *AIDO*, *DIDO* and *AIHO I/O*'s are not galvanically isolated. Their voltages are related to battery negative terminal *B-*. *Tiny BMS* or its accessories may be damaged if non isolated *I/O*'s are used with other device of which ground potential is different to battery negative terminal *B-*.
>
> **Note:** Only one firmware feature can be assigned to the same external pin at a time. Special care must be taken when external pin is used as input. Before the ignition, charger detection or speed input pin wiring setup is being implemented always check if used pin has not been configured as output before.

2.3.9. *Tiny BMS Ignition* connection

Tiny BMS Ignition feature can be used for two main reasons: to control (turn on or turn off) load output and to enter deep sleep mode to achieve the lowest *Tiny BMS* current consumption. *Ignition* software feature can be assigned to one of the available *Tiny BMS* analog (*AIDO1/2*), digital (*DIDO1/2*) or hybrid (*AIHO1/2*) external input pins, if it is not assigned to any other firmware features. *Figure 2.19* shows *Ignition* connection diagram example when *AIDO1* pin is configured to use as *Ignition* feature. When *AIDO1* is connected to *BAT+* pin (ignition key is on and a high logic level is presented on the *AIDO1* pin), *Ignition* is activated and *BMS* wakesup from sleep mode immediately and turns on the load output.
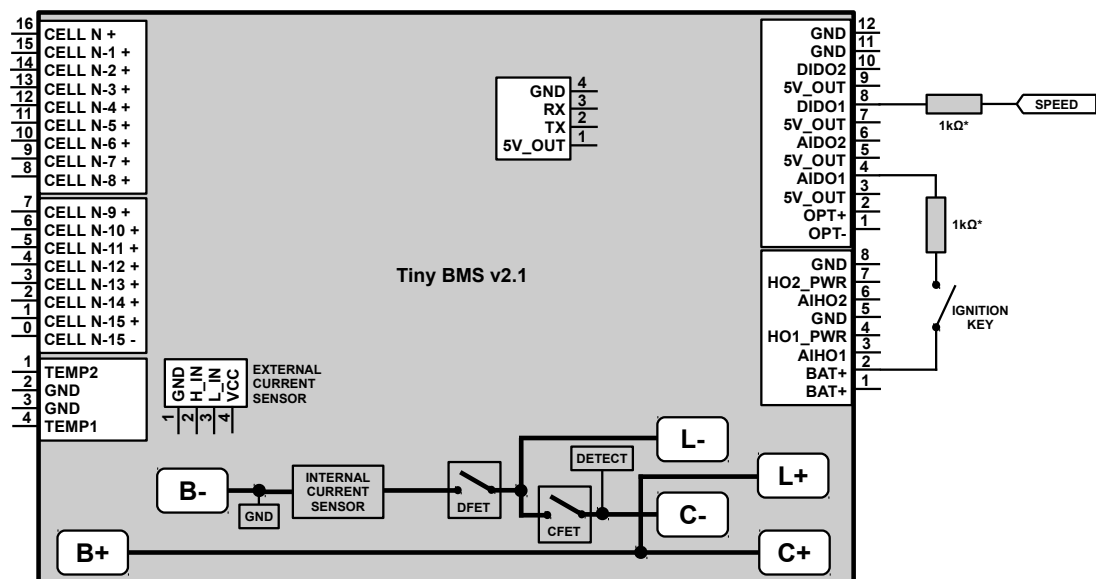
Figure 2.19: Tiny BMS Ignition feature (configured on AIDO1 pin) and Speed Sensor input (configured on DIDO1 pin) connection diagram

Otherwise, when *AIDO1* is disconnected from *BAT+* pin (ignition key is off and a low logic level is presented on the *AIDO1* pin), *Ignition* is deactivated, *BMS* turns off load output and the *BMS* enters deep sleep mode after *10* seconds to achieve the lowest possible *Tiny BMS* current consumption. For safety reasons it is highly recommended to use about a *1 kΩ* resistor in series between configured *Ignition* pin and *BAT+* pin. How to properly configure and use *Ignition* feature and other aspects and limitations about this feature refer to *Chapter 4*.

2.3.10. *Tiny BMS* speed sensor input connection

Tiny BMS supports electric vehicle speed measurement feature, when speed sensor signal is applied to the *Tiny BMS* configured external input pin. Measured vehicle speed is displayd on the *Ebyke Analyzer Android* application, furthermore, *BMS* calculates traveled distance and estimates how many distance wehicle can travel   depending on the available battery state of charge. Speed sensor input can be assigned only to *BMS* digital external input pins (*DIDO1/2*), if it is not assigned to any other firmware features. For supported *DIDO1/2* signal levels refer to *Tiny BMS* datasheet. External speed sensor connection diagram are shown in the *Figure 2.19*. For safety reasons it is highly recomended to use about a *1 kΩ* resistor in series between configured *BMS* speed sensor input pin and speed sensor signal source. How to properly configure and use *Tiny BMS* speed sensor feature refet to *Chapter 4*.

2.3.11. *Tiny BMS* external charger detection connection

Tiny BMS has internal circuitry for charger / load detection, but it is only used when internal *FETs* are used to control charger / load. When charger is controlled with external relay / contactor along with the external current sensor for high currents battery application, the external charger detection feature also should be used. The external charger detection feature is recommended to use only for *Tiny BMS Dual Port* operation mode setup. For *Single Port* operation mode due to specific wiring and for the need to detect both connected load and charger, the external charger detection is not necessary (refer to *Chapter 2.4 Tiny BMS Single Port operation mode reference designs*). The external charger detection feature bassically works as folow: when configured charger detection pin is in a high logic level state, *BMS* thinks the charger is connected and turns on the charger relay / contactor. Otherwise when configured charger detection pin is in a low logic level state, *BMS* thinks the charger is disconnected and turns off charger relay / contactor. External charger detection feature can be assigned to one of the available *Tiny BMS* analog (*AIDO1/2*), digital (*DIDO1/2*) or hybrid (*AIHO1/2*) external input pins if it is not assigned to any other firmware features. Also external charger detection feature can be used to imitate the fake charger even when physically the charger is not connected to *BMS*. It is usefull in some cases when user needs to balance battery cells all the time without charger connected. It can be done by connecting configured external charger detection pin to *BAT+* through a *1 kΩ* resistor (*Figure 2.23*). Also, it is still a possibility to use *BMS* internal charger detection circuitry even the external relay / contactor is used for charger control. This option is possible if the charger relay / contactor is used as a low side switch (*Figure 2.22*). How to properly configure and use *Tiny BMS* external charger detection feature refet to *Chapter 4*.
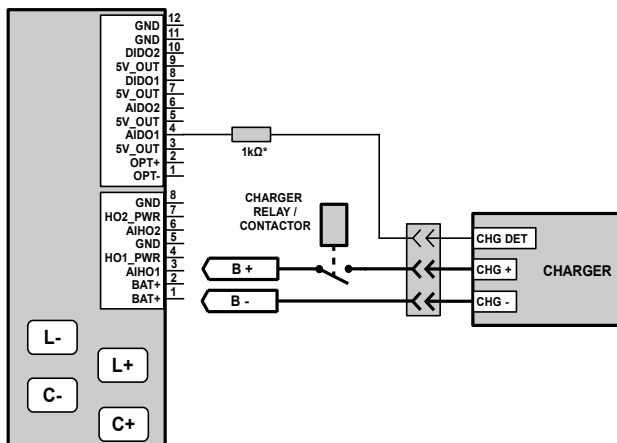
Figure 2.20: Tiny BMS external charger detection (configured on AIDO1 pin) connection diagram when charger has charger detection output
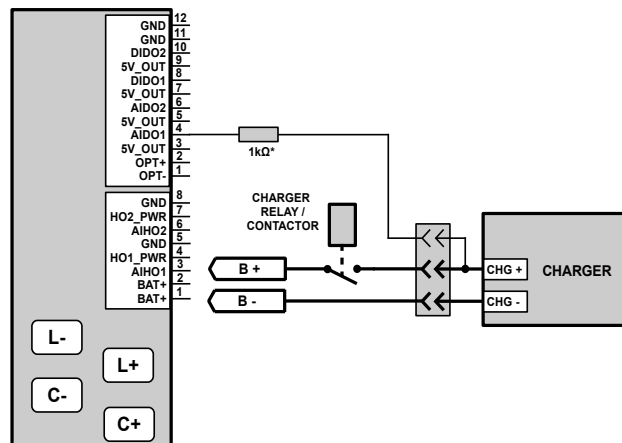
Figure 2.21: Tiny BMS external charger detection (configured on AIDO1 pin) connection diagram when charger has not charger detection output
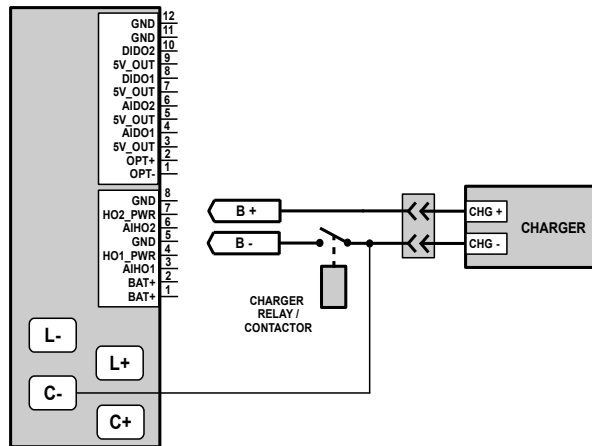
Figure 2.22: Tiny BMS internal charger detection connection diagram only when charger relay / contactor is used as low side switch
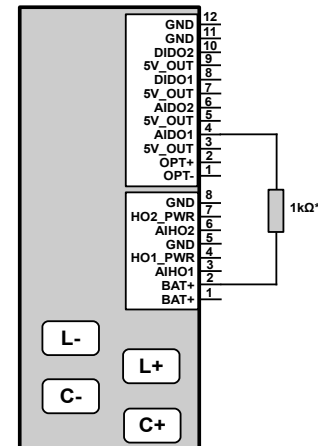
Figure 2.23: Tiny BMS fake charger imitation when only battery cells balancing is required

2.3.12. *Tiny BMS* load / charger external relay / contactor connection

Each *Tiny BMS* analog (*AIDO1/2*), digital (*DIDO1/2*) or hybrid (*AIHO1/2*) external pins can be configured as output to control both charger and load relays / contactors or another analog circuitry for high current battery application. *AIDO1/2* and *DIDO1/2* outputs are open drain and can sink up to *250 mA* continous current. *AIHO1/2* output is splitted into two groups: *AIHO1/2 Active Low* and *AIHO1/2 Active High*. When *AIHO1/2* is configured as *Active Low*, the output works the same way like analog *AIDO1/2* and digital *DIDO1/2* outputs. The relay / contactor connection diagram to *AIDO1/2*, *DIDO1/2* or *AIHO1/2 Active Low* configurations is shown in *Figure 2.24*. When relay / contactor voltage is equal to battery pack nominal voltage and rated current is no more than *250 mA* no aditional circuitry is needed and relay / contactor coil terminals can be connected directly between battery pack positive terminal and *Tiny BMS* external output as shown in *Figure 2.24*. Otherwise, it may require a *DC-DC* converter or another suitable solution. Also when relay / contactor working voltage does not match the battery pack nominal voltage *AIHO1/2* outputs configured as *Active High* can be used. If configured as output, *AIHO* output can supply up to *250 mA* current from *AIHO* supply pin to *AIHO I/O* pin. *Figure 2.25* shows *12 V* relay / contactor connection diagram to *AIHO1/2 Active High* configuration *BMS* external output for *12s Li-ion (48 V)* battery pack and external *12 V* system power supply for relay / contactor control, connected to *AIHO1/2* power pin. The choice of relay / contactor for load and charger, its electrical and mechanical characteristics, dimensioning depends on various aspects of the system in which the battery and the *BMS* are used and is out of the scope of this document. How to properly configure and use *Tiny BMS* external outputs for load and charger control refet to *Chapter 3* and *Chapter 4* .

Note: If the rated current of the relay / contactor coil is more than *250 mA*, additional signal relay should be used to drive the contactor.

Note: *Tiny BMS* only supports non-latching type relays / contactors.

Note: Protection diodes are necessary between relay / contactor coil terminals to protect *Tiny BMS* from high voltage spikes induced while deactivating the relay / contactor.
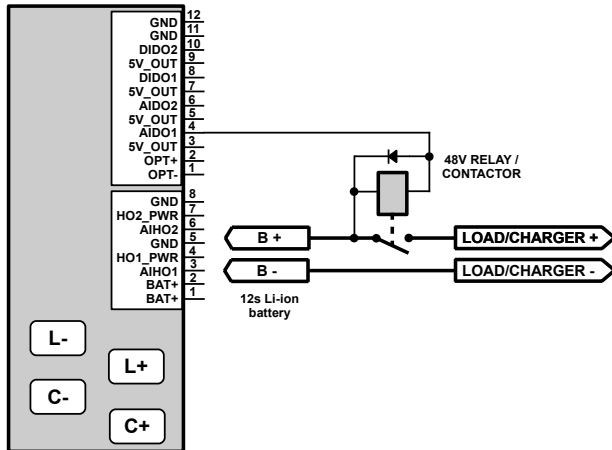
Figure 2.24: Tiny BMS external relay / contactor (configured on AIDO1 pin) connection diagram
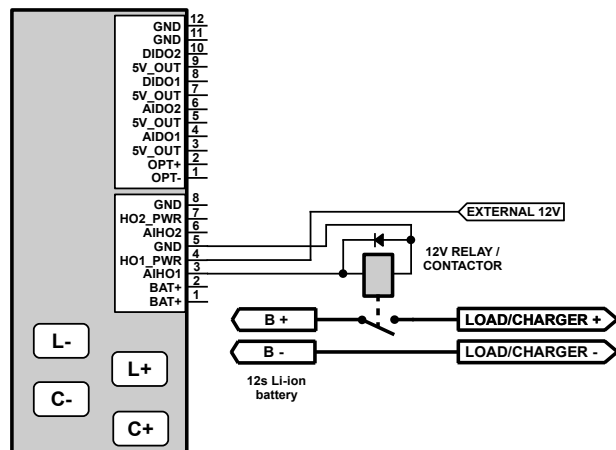
Figure 2.25: Tiny BMS external relay / contactor (configured on AIHO1 Active High pin) connection diagram

2.3.13. *Tiny BMS* load contactor *Precharge* connection

Somethimes *Tiny BMS* is used to manage battery applications where load has a substantial input capacitance and often it is desirable to limit the high in-rush current that occurs when the load relay / contactor is closed in order to reduce stress to the internal components of the load device and prevent the contactor from welding. In such case *Tiny BMS* offers a very straightforward load relay / contactor precharge feature. The load contactor precharge sub-circuit should consist of a relay in series with a precharge resistor (or another current limiting device), and should be connected in such way that it would bypass the load contactor. The *Tiny BMS* precharge reference connection diagram is showned in the *Figure 2.26*. The *BMS* load *Precharge* feature can be assigned to one of the available *Tiny BMS* analog (*AIDO1/2*), digital (*DIDO1/2*) or hybrid (*AIHO1/2*) external output pins, if it is not assigned to any other firmware features. The load precharge relay is controlled in the same way like the main load relay / contactor and this allows the precharge relay can be connected to any available *Tiny BMS* external output pin according to *Figure 2.24* or *Figure 2.25*. The precharge timing diagram is displayed in the *Figure 2.27*. Basically, this feature works by activating configured *Precharge* output pin for a *Precharge Duration* period of time whenever the load contactor needs to be closed and opened again (i.e. system power-up, ignition toggle, or when a protection is deactivated). It is possible use *BMS* internal *FETS* as precharge switch, if both load and charger are controlled by relay / contactor and load relay is connected as low side switch. In that case load precharge resistor can be connected directly to *C- BMS* solder-pad or therminal. How to properly configure and use *Tiny BMS Precharge* feature refet to *Chapter 4*.

Note: *Tiny BMS Precharge* feature is used together with load relay / contactor. Do not be confused with *Pre-Charging* stage that occurs during the charging process.
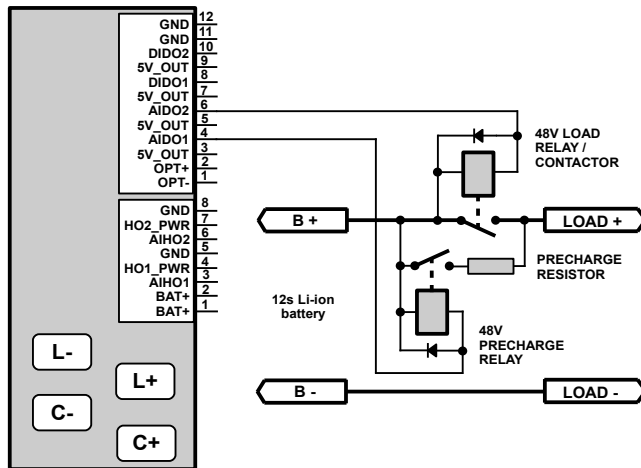
Figure 2.26: Tiny BMS Precharge reference connection diagram (Precharge relay control configured on AIDO1, Load contactor - on AIDO2)
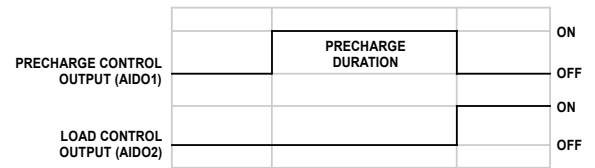
Figure 2.27: Tiny BMS Precharge and load signals timing diagram

2.4. *Tiny BMS* reference designs

This section provides the most common *Tiny BMS* connection diagrams used in the various battery management systems, depending on the *BMS* operating mode (*Dual Port* or *Single Port* mode), load / charger switch configuration and covers all aspects of proper installation and wiring recomendations – all in order to prevent most commonly faced issues with *Tiny BMS* hardware features and limitations.

2.4.1. *Tiny BMS Dual Port* mode reference designs

Tiny BMS by default was designed to work in a *Dual Port* operation mode, where different switches for load and charger control should be used. *BMS* has integrated internal switches (*FETs*) for load and charger and it can be used for low current battery applications: up to *150 A* discharging peak and up to *30 A* charging peak current. *Figure 2.28* shows a typical *Tiny BMS* connection diagram when internal *FETs* are used for load and charger control.
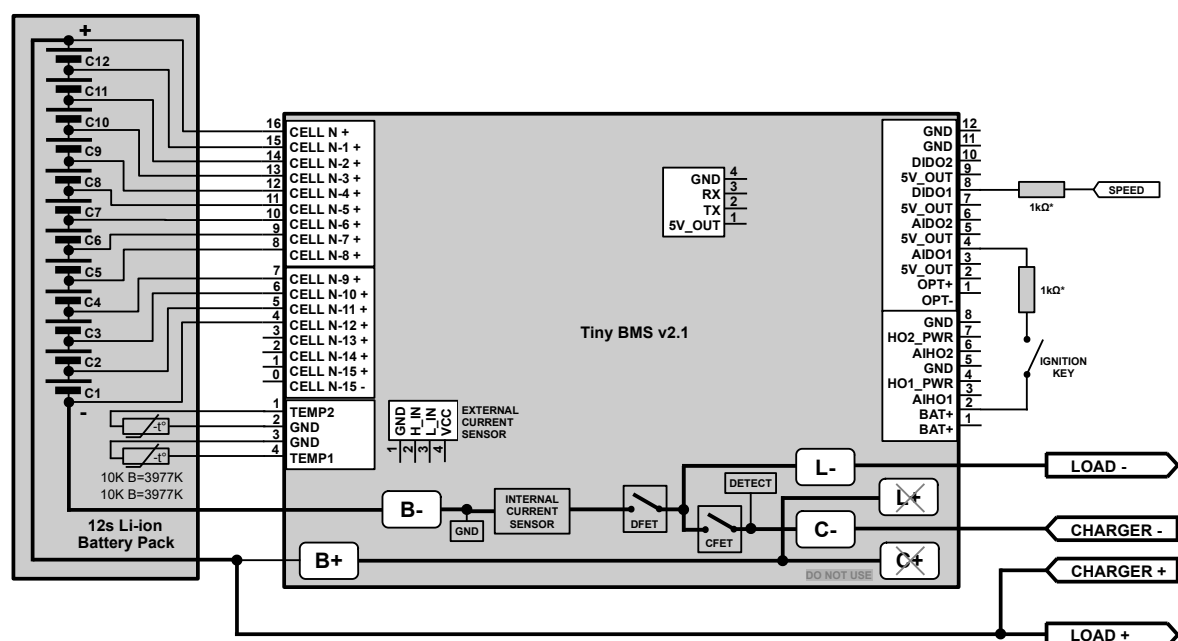
Figure 2.28: A typical Tiny BMS Dual Port connection diagram: low current application with internal FETs, ignition and speed sensor input features

In *Figure 2.28* reference design *Tiny BMS* is connected to a *12s Li-ion* Battery pack. Battery pack has integrated two *NTC* temperature sensors, which can be connected to *Tiny BMS* external temperature sensor channels. Load and charger are controlled (turned on and off) using *BMS* internal *FET* switches. Take a note, that *BMS* internal *DFET* and *CFET* are connected in series, that's why charger always has higher priority against load or ignition signal. If the charger is connected to the *BMS*, both *BMS* internal *DFET* and *CFET* switches are turned on, even if the load should be disconnected by the ignition signal. Use of *Ignition* feature is always recomended to enter *BMS* device deep sleep mode to achievie lowest *BMS* power consumption. *Tiny BMS v2.1* solder-pads *L+* and *C+* are recommended only for low power *30 A Tiny BMS* version. For high power *150 A Tiny BMS 2.1* version *L+* and *C+* solder-pads are not recommended and should be left unconnected and load / charger positive terminal should bypass *Tiny BMS* board and connect directly to battery pack positive terminal. Only thin wire should be connected to *Tiny BMS* board *B+* solder-pad only for powering internal *BMS* circuitry.
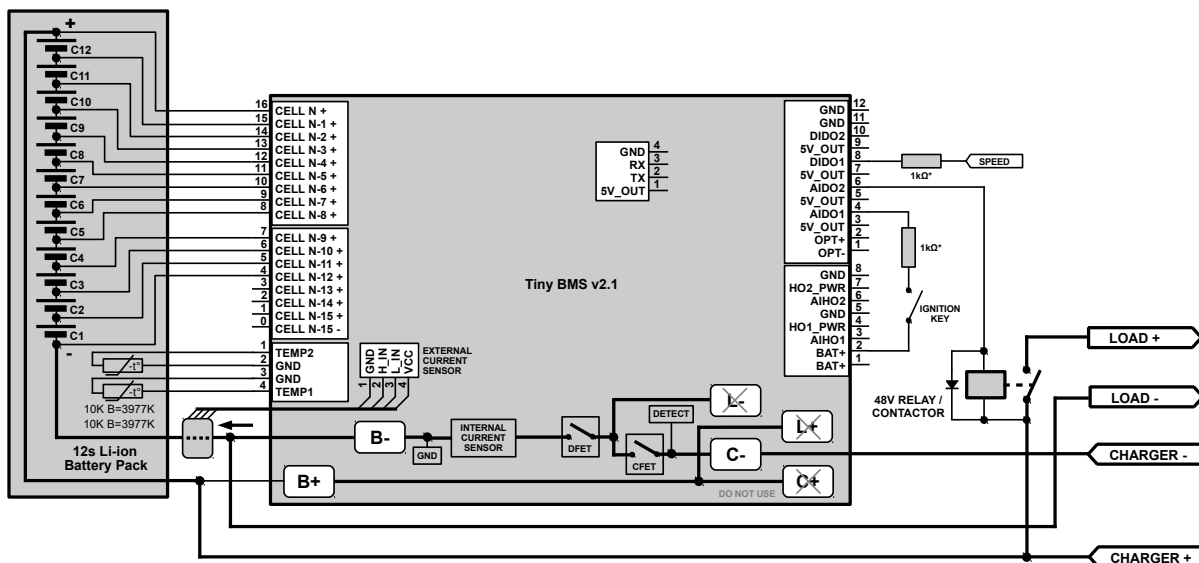
Figure 2.29: Tiny BMS Dual Port connection diagram: high current application with relay / contactor for load control, internal FET switch for charger control, ignition and speed sensor input features
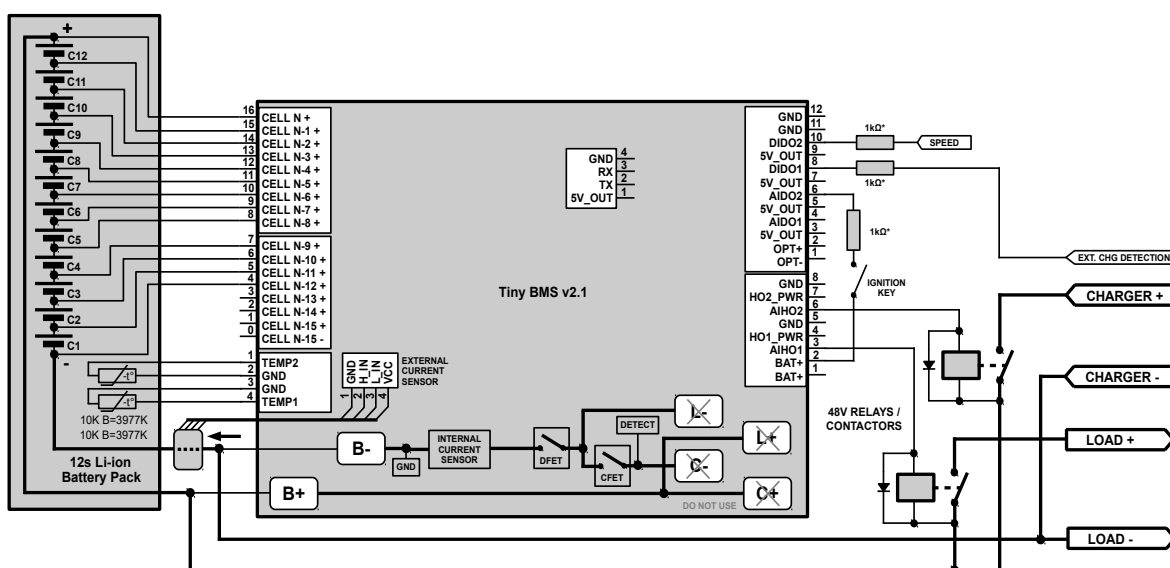
Figure 2.30: Tiny BMS Dual Port connection diagram: high current application with relays / contactors for load and charger control, external charger detection, ignition and speed sensor input features

Figure 2.29 to *Figure 2.30* shows *Tiny BMS* connection diagram when load, charger or both are controlled with external relay / contactor. In this configuration external current sensor is necessary for *BMS* current measurement, because high current path bypasses the *BMS* board and the internal current measurement is not possible. Also, in this *BMS* configuration, when charger is controlled by external relay / contactor, external charger detection signal is needed, to detect whenever charger is connected or not. If load is controlled by external relay / contactor (the feature is enabled by *Battery Insider* application), *BMS* turns on its internal *DFET*, and during *BMS* idle, charging and discharging states *DFET* always stays turned on and *L- BMS* solder-pad or terminal should be left unconnected as well as *C-* solder-pad or terminal, when charger is controlled with external relay / contactor. If load and charger are controlled by external relay / contactor and load precharge feature is needed it is possible to use internal *BMS FET* switches to control precharge. This allows to save extra relay and simplifies the design. To use this *Precharge* feature load relay / contactor must be connected as a low side switch, because the same way is connected *BMS* internal *FETs* and *Precharge* circuitry should be connected to bypass load relay contacts. *Figure 2.31* shows how to connect external load relay / contactor and precharge resistor using internal *BMS FET* switches to control load precharge. How to properly configure *Tiny BMS* for *Dual Port* operation mode refer to *Chapter 4*.
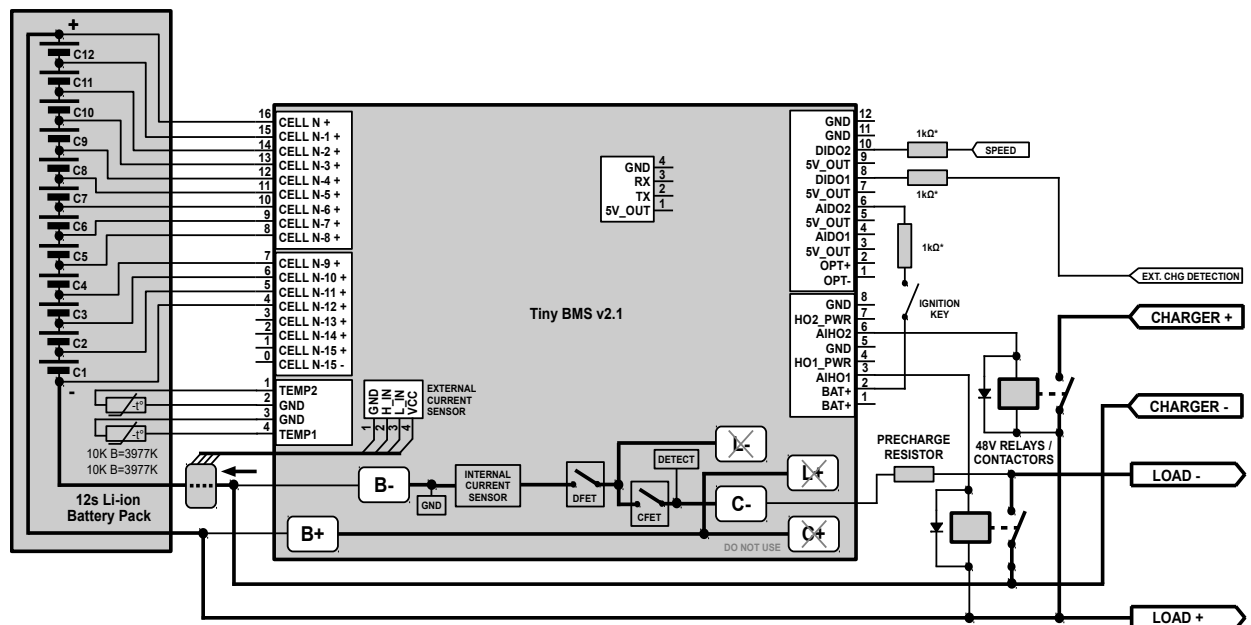
Figure 2.31: Tiny BMS Dual Port connection diagram: high current application with relays / contactors for load and charger control, external charger detection, ignition and speed sensor input, load precharge features

2.4.2. *Tiny BMS Single Port* mode reference designs

For the special customers needs in the battery applications where load and charger needs to be connected together to *BMS* and the battery system or in the applications where some type of inverters (like solar inverters), which acts as load and charger, *Tiny BMS* was adopted to work in this manner by implementing the *Tiny BMS Single Port* operation mode. In the *BMS Single Port* operation mode the load or / and charger should be connected together to the same *Tiny BMS* charging / discharging port. *Figure 2.32* shows a typical *Tiny BMS Single Port* connection diagram for a low current applications, using *BMS* internal *FET* switches for load / charger control. Pay attention, that load / charger negative terminal in *Single Port* operation mode should be connected only to *C- BMS* solder-pad or terminal, leaving *L- Tiny BMS* solder-pad or terminal unconnected. In this mode max. charging / discharging current that *Tiny BMS 2.1 version* can hold is only *30 A*. If a higher current sustainability is needed external relay / contactor should be used. *Tiny BMS* connection diagram for a high current application in a *Single Port* operation mode are shown in *Figure 2.33*. In *Single Port* operation mode load / charger external relay contactor only can be connected

as a low side switch and load / charger negative terminal should be connected to *Tiny BMS C-* solder-pad for internal load and charger detection. In a *Single Port* operation mode use of external charger detection feature is not recommended. How to properly configure *Tiny BMS* for a *Single Port* operation mode refer to *Chapter 4.*

Figure 2.32: Tiny BMS Single Port connection diagram: low current application with internal FETs, ignition and speed sensor input features

Figure 2.33: Tiny BMS Single Port connection diagram: high current application with relay / contactor for load / charger control, internal load / charger detection, ignition and speed sensor input features

Note: *Tiny BMS v2.1* in a *Single Port* operation mode due to load / charger connection to *C- BMS* solder-pad, can hold only up to *30 A* continuous current!

Note: In a *Single Port* operation mode load / charger relay / contactor should be connected as a low side switch along with the *Tiny BMS* internal charger / load detection.

Chapter 3: Firmware features overview

3. Introduction

The *Tiny BMS* device provides a feature-rich battery management and protection solution for *4*-series cell to *16*-series cell battery-pack applications. The *Tiny BMS* device has extended firmware features and capabilities, including:

- Fully integrated battery pack manager and protection;
- Low side protection *FET* drive;
- Integrated cell balancing;
- Ultra low power mode;
- Full array of programmable protection features: voltage, current, temperature;
- Lifetme statistics data monitor and event recorder;
- Supports two external temperature sensor channels and one dual-range current sensor;
- Connectivity: *UART*, *MODBUS*, *CAN*, *Bluetooth* and *BMS* data broadcasting;
- Programmable inputs and outputs;
- User-upgradeable firmware for quick updates, bringing bugfixes, new features and other improvements, such as client-specific functionality.

This chapter in detail covers all aspects of *Tiny BMS* firmware features and capabilities. This docummentation is based on firmware version *BMS.v2.3.254.bms* and later releases. The *Tiny BMS* is being constantly improved and evolved, new features are built on top of the base firmware, but the basic protection concept and logic, described in this document, is still the same in the updates.

3.1. *Tiny BMS* protections

The *Tiny BMS* safety mechanisms disconnects load, charger or both to prevent battery if any of the cells go out of allowed range of voltage or current, also up to two external temperature sensor channels along with *BMS* internal onboard temperature sensor are measured and can be used of emergency disconnect or for disabling of charging if cell temperatue is too low. The *Tiny BMS* provided protections can be configured as recoverable or non-recoverable protection. Recoverable protection means, that *Tiny BMS* automatically resumes charging and discharging once the protection is recovered. In non-recoverable protection mode charging and discharging process are resumed only by the user (when charger is reconnected, or ignition signal is toggled). It can be controlled by *Tiny BMS Automatic Recovery* settings. How to configure the *Tiny BMS* refer to *Chapter 4*. *Tiny BMS* protections are divided into two levels: *Warning* and *Fault*. When the *Warning* condition occurs only one switch (load or charger) is turned off. Otherwise, when the *Fault* condition occurs both load and charger switches are turned off.

3.1.1. *Over-Temperature Fault*

The *Tiny BMS* device provides over-temperature protection for cells under charge, discharge or idle states, based on cell temperature measurement (two external temperature sensor channels) and onboard temperature sensor measurement.

Table 3.1: Tiny BMS Over-Temperature Fault

Status	Condition	Action
Normal	**If one or both external temperature sensor channels are connected and in use:** ***Ext. Temp #1 ≤ Over-Heat Cutoff*** threshold ***and*** ***Ext. Temp #2 ≤ Over-Heat Cutoff*** threshold ***and*** ***Onboard Temp ≤ 90 °C.*** **If no external temperature sensor channels are connected:** ***Onboard Temp ≤ Over-Heat Cutoff*** threshold.	***BMS Dual Port* mode:** ***Load Switch*** – turned on, if *Ignition* feature disabled. ***Load Switch*** – depending on *Ignition* signal, if *Ignition* feature enabled. ***Charger Switch*** – turned on, if charger detected, otherwise turned off. ***BMS Single Port* mode:** ***Single Switch*** – turned on, if *Ignition* feature disabled. ***Single Switch*** – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Fault	**If one or both external temperature sensor channels are connected and in use:** ***Ext. Temp #1 > Over-Heat Cutoff*** threshold ***or*** ***Ext. Temp #2 > Over-Heat Cutoff*** threshold ***or*** ***Onboard Temp > 90 °C.*** **If no external temperature sensor channels are connected:** ***Onboard Temp > Over-Heat Cutoff*** threshold.	*BMS* device generates *Over-Temperature Cutoff Occurred* event (*0x04*). ***BMS Dual Port* mode:** **Load Switch** – turned off. **Charger Switch** – turned off. *BMS* switches to sleep mode after *10* seconds time interval, if no charger, *Ignition* signal or ongoing *UART* communication connected. *BMS* in sleep mode on *Over-Temperature Fault (*if no *Under-Voltage Fault occurred* at the same moment*)* measures temperature and, if *Fault* condition recovers, wakes up from sleep mode and turns back on **Load Switch,** if *Ignition* feature disabled. If *Ignition* feature enabled, *BMS* **Load Switch** state depends on *Ignition* signal. ***BMS Single Port* mode:** **Single Switch** – turned off. *BMS* switches to sleep mode after *10* seconds time interval, if no charger, *Ignition* signal or ongoing *UART* communication connected. *BMS* in sleep mode on *Over-Temperature Fault (*if no *Under-Voltage Fault* occurred at the same moment*)* measures temperature and, if *Fault* condition recovers, wakes up from sleep mode and turns back on **Single Switch,** if *Ignition* feature disabled. If *Ignition* feature enabled, *BMS* **Single Switch** state depends on *Ignition* signal and the charger detection.
Recovery	***Automatic Recovery > 0 s*:** *Fault* condition recovers after *BMS* measured temperatures meets normal status conditions and *Automatic Recovery* time interval has been passed.	*BMS* device generates *Recovered From Over-Temperature Fault Condition* event (*0x73*). *BMS* goes back to normal status operation.

	Automatic Recovery = 0 s: *Fault* condition recovers after *BMS* measured temperatures meets normal status conditions and the charger was connected to *BMS,* or *Ignition* signal was toggled by the user (if *Ignition* feature was enabled), or *BMS* was restarted.	

3.1.2. Charge Under-Temperature Warning

The *Tiny BMS* device provides charge under-temperature protection for cells under charge state, based on cell temperature measurement (two external temperature sensor channels) or onboard temperature sensor measurement.

Table 3.2: Tiny BMS Charge Under-Temperature Warning

Status	Condition	Action
Normal	**If one or both external temperature sensor channels are connected and in use:** ***Ext. Temp #1 ≥ Low Temperature Charger Cutoff*** threshold ***and*** **Ext. Temp #2 ≥** ***Low Temperature Charger Cutoff*** threshold. **If no external temperature sensor channels are connected:** ***Onboard Temp ≥ Low Temperature Charger Cutoff*** threshold.	***BMS Dual Port*** **mode:** ***Load Switch*** – turned on, if *Ignition* feature disabled. ***Load Switch*** – depending on *Ignition* signal, if *Ignition* feature enabled. ***Charger Switch*** – turned on, if charger detected, otherwise turned off. ***BMS Single Port*** **mode:** ***Single Switch*** – turned on, if *Ignition* feature disabled. ***Single Switch*** – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Warning	**If one or both external temperature sensor channels are connected and in use:** ***Ext. Temp #1 < Low Temperature Charger Cutoff*** threshold ***or*** **Ext. Temp #2 <** ***Low Temperature Charger Cutoff*** threshold. **If no external temperature sensor channels are connected:** ***Onboard Temp < Low Temperature Charger Cutoff*** threshold.	*BMS* device generates *Low Temperature Charging Cutoff Occurred* event (*0x37*). ***BMS Dual Port*** **mode:** ***Load Switch*** – turned on, if *Ignition* feature disabled. ***Load Switch*** – depending on *Ignition* signal, if *Ignition* feature enabled. ***Charger Switch*** – turned off. *BMS* switches to sleep mode after *10* seconds time interval, if no charger, *Ignition* signal or ongoing *UART* communication connected. *BMS* in sleep mode on *Charge Under-Temperature Warning (*if no *Under-Voltage Fault* occurred at the same moment)* measures temperature and, if the *Warning* condition recovers, wakes up from sleep mode and turns back on ***Charger Switch,*** if the charger was detected*. ***BMS Single Port*** **mode:** ***Single Switch*** – turned on, if the load connection is detected and *Ignition* feature disabled. ***Single Switch*** – depends on *Ignition* signal, if *Ignition*

		feature enabled, and on the load detection.
		BMS switches to sleep mode after *10* seconds time interval, if no charger, ignition signal or ongoing *UART* communication connected. *BMS* in sleep mode on *Charge Under-Temperature Warning (*if no *Under-Voltage Fault* occurred at the same moment*)* measures temperature and, if the *Warning* condition recovers, wakes up from sleep mode and turns back on **Single Switch,** if *Ignition* feature disabled. If *Ignition* feature enabled, *BMS* **Single Switch** state depends on *Ignition* signal and the charger detection.
Recovery	***Automatic Recovery > 0 s***: *Warning* condition recovers after *BMS* measured temperatures meets normal status conditions, *Automatic Recovery* time interval has been passed and, if the second level *Discharge Under-Temperature Fault* protection has been recovered, if it was detected. ***Automatic Recovery = 0 s***: *Warning* condition recovers after *BMS* measured temperatures meets normal status conditions, the second level *Discharge Under-Temperature Fault* protection has been recovered, if it was detected, and the charger was connected to *BMS,* or *Ignition* signal was toggled by the user (if *Ignition* feature was enabled), or *BMS* was restarted.	*BMS* device generates *Recovered From Low Temperature Warning Condition* event (*0x74*). *BMS* goes back to normal status operation.

3.1.3. Discharge Under-Temperature Fault

The *Tiny BMS* device provides discharge under-temperature protection for cells under charge and discharge states, based on cell temperature measurement (two external temperature sensor channels) or onboard temperature sensor measurement. *Discharge Under-Temperature Fault* is the *BMS* second level protection and always detected after the first level protection C*harge Under-Temperature Warning* has been detected*.*

Table 3.3: Tiny BMS Discharge Under-Temperature Fault

Status	Condition	Action
Normal	**If one or both external temperature sensor channels are connected and in use:** ***Ext. Temp #1 ≥ -40 °C*** threshold **and** **Ext. Temp #2 ≥ -40 °C** threshold. **If no external temperature sensor channels are connected:** ***Onboard Temp ≥ -40 °C*** threshold.	**BMS Dual Port** mode (no *Charge Under-Temperature Warning* detected*)*: **Load Switch** – turned on, if *Ignition* feature disabled. **Load Switch** – depending on *Ignition* signal, if *Ignition* feature enabled. **Charger Switch** – turned on, if charger detected, otherwise turned off. **BMS Single Port** mode (no *Charge Under-Temperature Warning* detected*)*:

		Single Switch – turned on, if *Ignition* feature disabled. *Single Switch* – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Fault	**If one or both external temperature sensor channels are connected and in use:** *Ext. Temp #1* < *-40 °C* threshold *or* *Ext. Temp #2* < *-40 °C* threshold. **If no external temperature sensor channels are connected:** *Onboard Temp* < *-40 °C* threshold.	*BMS* device generates *Low Temperature Cutoff Occurred* event (*0x0A*). ***BMS Dual Port* mode:** **Load Switch** – turned off. **Charger Switch** – turned off. *BMS* switches to sleep mode after *10* seconds time interval, if no charger, *Ignition* signal or ongoing *UART* communication connected. *BMS* in sleep mode on *Charge Under-Temperature Warning and Discharge Under-Temperature Fault (*if no *Under-Voltage Fault* occurred at the same moment)* measures temperature and, if *Fault* condition recovers, wakes up from sleep mode and turns back on **Load Switch,** if *Ignition* feature disabled. If *Ignition* feature enabled, *BMS* **Load Switch** state depends on *Ignition* signal. **Charger Switch** state depends on *Charge Under-Temperature Warning* status. ***BMS Single Port* mode:** **Single Switch** – turned off. *BMS* switches to sleep mode after *10* seconds time interval, if no charger, *Ignition* signal or ongoing *UART* communication connected. *BMS* in sleep mode on *Charge Under-Temperature Warning and Discharge Under-Temperature Fault (*if no *Under-Voltage Fault* occurred at the same moment)* measures temperature and, if *Fault* condition recovers, wakes up from sleep mode and controls **Single Switch** according to *Charge Under-Temperature Warning* status.
Recovery	***Automatic Recovery > 0 s*:** *Fault* condition recovers after *BMS* measured temperatures meets normal status conditions and *Automatic Recovery* time interval has been passed. ***Automatic Recovery = 0 s*:** *Fault* condition recovers after *BMS* measured temperatures meets normal status conditions and the charger was connected to *BMS,* or *Ignition* signal was toggled by the user (if *Ignition* feature was enabled), or *BMS* was restarted.	*BMS* device generates *Recovered From Low Temperature Fault Condition* event (*0x75*). *BMS* goes back to normal status operation, if *Charge Under-Temperature Warning* also recovers, or goes to *Charge Under-Temperature Warning* status operation*,* if it was not recovered yet*.*

3.1.4. Over-Voltage Fault

The *Tiny BMS* device provides over-voltage protection for cells under charge, discharge and idle states, based on each cell in series voltages measurement.

Table 3.4: Tiny BMS Over-Voltage Fault

Status	Condition	Action
Normal	**Max. Cell Voltage ≤ Over-Voltage Cutoff** threshold.	**BMS Dual Port** mode: **Load Switch** – turned on, if *Ignition* feature disabled. **Load Switch** – depending on *Ignition* signal, if *Ignition* feature enabled. **Charger Switch** – turned on, if charger detected, otherwise turned off. **BMS Single Port** mode: **Single Switch** – turned on, if *Ignition* feature disabled. **Single Switch** – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Fault	**Max. Cell Voltage > Over-Voltage Cutoff** threshold.	*BMS* device generates *Over-Voltage Cutoff Occurred* event (*0x03*). **BMS Dual Port** mode: **Load Switch** – turned on, if *Ignition* feature disabled. **Load Switch** – depending on *Ignition* signal, if *Ignition* feature enabled. **Charger Switch** – turned off. Cells balancing process is activated. Due to ongoing cells balancing process *BMS* does not switch to sleep mode unless *Fully-Discharged Warning* or *Under-Voltage Cutoff Fault* was detected. **BMS Single Port** mode: **Single Switch** – turned on, if the load connection is detected, if *Ignition* feature disabled. **Single Switch** – depends on *Ignition* signal, if *Ignition* feature enabled and if the load connection is detected. Cells balancing process is activated. Due to ongoing cells balancing process *BMS* does not switch to sleep mode unless *Fully-Discharged Warning* or *Under-Voltage Cutoff Fault* was detected.
Recovery	**Automatic Recovery > 0 s**: *Fault* condition recovers after *BMS* measured cells voltages meets normal status conditions and *Automatic Recovery* time interval has been passed.	*BMS* device generates *Recovered From Over-Voltage Fault Condition* event (*0x79*). *BMS* goes back to normal status operation.

	Automatic Recovery = 0 s: *Fault* condition recovers after *BMS* measured cells voltages meets normal status conditions and the charger was connected to *BMS,* or *Ignition* signal was toggled by the user (if *Ignition* feature was enabled), or *BMS* was restarted.	

3.1.5.   Fully-Discharged Warning

The *Tiny BMS* device provides battery fully-discharged protection for cells under charge, discharge and idle states, based on each cell in series voltages measurement.

Table 3.5: Tiny BMS Fully-Discharged Warning

Status	Condition	Action
Normal	***Min. Cell Voltage ≥ Fully Discharged Voltage*** threshold.	***BMS Dual Port* mode:** ***Load Switch*** – turned on, if *Ignition* feature disabled. ***Load Switch*** – depending on *Ignition* signal, if *Ignition* feature enabled. ***Charger Switch*** – turned on, if charger detected, otherwise turned off. ***BMS Single Port* mode:** ***Single Switch*** – turned on, if *Ignition* feature disabled. ***Single Switch*** – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Warning	***Min. Cell Voltage < Fully Discharged Voltage*** threshold.	*BMS* device generates *Fully Discharged Cutoff Occurred* event (*0x31*). ***BMS Dual Port* mode:** ***Load Switch*** – turned off. ***Charger Switch*** – turned on, if charger detected, otherwise turned off. *BMS* switches to sleep mode after *10* seconds time interval, if no charger, *Ignition* signal or ongoing *UART* communication is connected. ***BMS Single Port* mode:** ***Single Switch*** – turned on, if the charger connection is detected, otherwise turned off. When charging is detected, but current decreases to a minimum threshold level, ***Single Switch*** are turned off again, to prevent battery from discharging. *BMS* switches to sleep mode after *10* seconds time interval, if no charger, *Ignition* signal or ongoing *UART* communication is connected.

| Recovery | **_Automatic Recovery > 0 s_:**

Warning condition recovers after _BMS_ measured cell voltages meets normal status conditions, _Automatic Recovery_ time interval has been passed and if the second level _Under-Voltage Fault_ protection has been recovered, if it was detected.

Automatic Recovery = 0 s:

Warning condition recovers after _BMS_ measured cells voltages meets normal status conditions, the second level _Under-Voltage Fault_ protection has been recovered, if it was detected, and the charger was connected to _BMS,_ or _Ignition_ signal was toggled by the user (if _Ignition_ feature was enabled), or _BMS_ was restarted. | _BMS_ device generates _Recovered From Fully Discharged Warning Condition_ event (_0x7A_).

BMS goes back to normal status operation. |

3.1.6. _Under-Voltage Fault_

The _Tiny BMS_ device provides battery Under-Voltage protection for cells under charge, discharge and idle states, based on each cell in series voltages measurement. _Under-Voltage Fault_ is the _BMS_ second level protection and always detected after the first level protection _Fully Discharged Warning_ has been detected_.

Note: _Under-Voltage Fault_ never recovers automatically even cells voltages back to normal level after _Automatic Recovery_ timeout. The only way to recover this fault is to connect the charger or restart the _BMS_, if the cells voltages are higher than _Under-Voltage Cutoff_ threshold. If cells voltages are lower than _Under-Voltage Cutoff_ threshold level, the user must to evaluate the situation and if the battery can be charged, the _Under-Voltage Cutoff_ threshold should be configured respectively to allow charging the battery. In the normal battery usage cycle the _BMS_ should be configured to never reach _Under-Voltage Fault_ condition.

Table 3.6: Tiny BMS Under-Voltage Fault

Status	Condition	Action
Normal	**_Min. Cell Voltage_ ≥ _Under-Voltage Cutoff_** threshold.	**_BMS Dual Port_ mode (no _Fully Discharged Warning_ detected_)_:** **_Load Switch_** – turned on, if _Ignition_ feature disabled. **_Load Switch_** – depending on _Ignition_ signal, if _Ignition_ feature enabled. **_Charger Switch_** – turned on, if charger detected, otherwise turned off. **_BMS Single Port_ mode (no _Fully Discharged Warning_ detected_)_:** **_Single Switch_** – turned on, if _Ignition_ feature disabled. **_Single Switch_** – depending on _Ignition_ signal and the charger detection, if _Ignition_ feature enabled.

Fault	*Min. Cell Voltage* < *Under-Voltage Cutoff* threshold.	*BMS* device generates *Under-Voltage Cutoff Occurred* event (*0x02*). **BMS Dual Port mode:** **Load Switch** – turned off. **Charger Switch** – turned off. **BMS Single Port mode:** **Single Switch** – turned off. *Tiny BMS* switches to sleep mode after *10* seconds time interval, if no ongoing *UART* communication is connected. *BMS* wakes up from sleep mode, if charger was connected to *BMS*, to check if the *Fault* condition can be recovered. If the *Fault* condition cannot be recovered at the moment, *BMS* goes back to sleep mode after the *Automatic Recovery* time interval has been passed, because charging is not allowed, even the charger is still connected to *BMS*. From *Ignition* signal *BMS* does not wake up from sleep mode, because discharging is also not allowed.
Recovery	*Fault* condition recovers after *BMS* measured cells voltages meets normal status conditions, the charger was connected to *BMS,* or *BMS* was restarted.	*BMS* device generates *Recovered From Under-Voltage Fault Condition* event (*0x7B*). *BMS* goes back to normal status operation, if *Fully Discharged Warning* also recovers, or goes to *Fully Discharged Warning* status operation*,* if it was not recovered yet*.*

3.1.7.   Charging Over-Current Fault

The *Tiny BMS* device provides over-current protection for cells under charging, based on charging current measurement.

Table 3.7: Tiny BMS Charging Over-Current Fault

Status	Condition	Action
Normal	*Charging Current ≤ Charge Over-Current Cutoff* threshold.	**BMS Dual Port mode:** **Load Switch** – turned on, if *Ignition* feature disabled. **Load Switch** – depending on *Ignition* signal, if *Ignition* feature enabled. **Charger Switch** – turned on, if charger detected, otherwise turned off. **BMS Single Port mode:** **Single Switch** – turned on, if *Ignition* feature disabled. **Single Switch** – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Fault	*Charging Current* > *Charge Over-Current Cutoff* threshold.	*BMS* device generates *Charging Over-Current Cutoff Occurred* event (*0x06*).

		BMS Dual Port mode: **Load Switch** – turned on, if *Ignition* feature disabled. **Load Switch** – depending on *Ignition* signal, if *Ignition* feature enabled. **Charger Switch** – turned off. *BMS* does not switch to sleep mode unless *Fault* is recovered, or *Fully-Discharged Warning,* or *Under-Voltage Cutoff Fault* was detected. **BMS Single Port** mode: **Single Switch** – turned off. *BMS* does not switch to sleep mode unless *Fault* is recovered, or *Fully-Discharged Warning,* or *Under-Voltage Cutoff Fault* was detected.
Recovery	**Automatic Recovery > 0 s**: *Fault* condition recovers after *BMS* measured charging current meets normal status conditions and *Automatic Recovery* time interval has been passed. **Automatic Recovery = 0 s**: *Fault* condition recovers after *BMS* measured charging current meets normal status conditions and the charger was reconnected to *BMS,* or *Ignition* signal was toggled by the user (if *Ignition* feature was enabled), or *BMS* was restarted.	*BMS* device generates *Recovered From Charging Over-Current Fault Condition* event (*0x76*). *BMS* goes back to normal status operation.

3.1.8. Discharging Over-Current Fault

The *Tiny BMS* device provides over-current protection for cells under discharging, based on discharging current measurement.

Table 3.8: Tiny BMS Discharging Over-Current Fault

Status	Condition	Action
Normal	**Discharging Current ≤ Discharge Over-Current Cutoff** threshold.	**BMS Dual Port** mode: **Load Switch** – turned on, if *Ignition* feature disabled. **Load Switch** – depending on *Ignition* signal, if *Ignition* feature enabled. **Charger Switch** – turned on, if charger detected, otherwise turned off. **BMS Single Port** mode: **Single Switch** – turned on, if *Ignition* feature disabled.

		Single Switch – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Fault	***Discharging Current* > *Discharge Over-Current Cutoff* threshold.**	*BMS* device generates *Discharging Over-Current Cutoff Occurred* event (*0x05*). ***BMS Dual Port* mode:** ***Load Switch*** – turned off. ***Charger Switch*** – turned off, if *Tiny BMS* internal *FETs* are configured as ***Load Switch*** and ***Charger Switch***. ***Charger Switch*** – turned on, if charger detected, and only if relays / contactors are used for ***Load Switch*** or ***Charger Switch***. *BMS* does not switch to sleep mode unless *Fault* is recovered, or *Fully-Discharged Warning,* or *Under-Voltage Cutoff Fault* was detected. ***BMS Single Port* mode:** ***Single Switch*** – turned off. *BMS* does not switch to sleep mode unless *Fault* is recovered, or *Fully-Discharged Warning,* or *Under-Voltage Cutoff Fault* was detected.
Recovery	***Automatic Recovery > 0 s*:** *Fault* condition recovers after *BMS* measured discharging current meets normal status conditions and *Automatic Recovery* time interval has been passed. ***Automatic Recovery = 0 s*:** *Fault* condition recovers after *BMS* measured discharging current meets normal status conditions and the charger was reconnected to *BMS,* or *Ignition* signal was toggled by the user (if *Ignition* feature was enabled), or *BMS* was restarted.	*BMS* device generates *Recovered From Discharging Over-Current Fault Condition* event (*0x77*). *BMS* goes back to normal status operation.

3.1.9. Regeneration Over-Current Fault

The *Tiny BMS* device provides over-current protection for cells under regeneration, based on regeneration current measurement.

Table 3.9: Tiny BMS Regeneration Over-Current Fault

Status	Condition	Action
Normal	***Regeneration Current ≤ Discharge Over-Current Cutoff* threshold.**	***BMS Dual Port* mode:** ***Load Switch*** – turned on, if *Ignition* feature disabled. ***Load Switch*** – depending on *Ignition* signal, if *Ignition* feature enabled. ***Charger Switch*** – turned on, if charger detected,

		otherwise turned off. ***BMS Single Port* mode:** ***Single Switch*** – turned on, if *Ignition* feature disabled. ***Single Switch*** – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Fault	***Regeneration Current* > *Discharge Over-Current Cutoff*** threshold.	*BMS* device generates *Regeneration Over-Current Cutoff Occurred* event (*0x07*). ***BMS Dual Port* mode:** ***Load Switch*** – turned off. ***Charger Switch*** – turned off. *BMS* does not switch to sleep mode unless *Fault* is recovered, or *Fully-Discharged Warning,* or *Under-Voltage Cutoff Fault* was detected. ***BMS Single Port* mode:** There is no *Regeneration* state in the *BMS Single Port* mode, available only *Charging* and *Discharging* states.
Recovery	***Automatic Recovery > 0 s*:** *Fault* condition recovers after *BMS* measured regeneration current meets normal status conditions and *Automatic Recovery* time interval has been passed. ***Automatic Recovery = 0 s*:** *Fault* condition recovers after *BMS* measured regeneration current meets normal status conditions and the charger was reconnected to *BMS,* or *Ignition* signal was toggled by the user (if *Ignition* feature was enabled), or *BMS* was restarted.	*BMS* device generates *Recovered From Regeneration Over-Current Fault Condition* event (*0x78*). *BMS* goes back to normal status operation*.*

3.1.10. Disconnected External Current Sensor Fault

The *Tiny BMS* device provides protection from accidentally disconnection of in use external current sensor.

Table 3.10: Tiny BMS Disconnected External Current Sensor Fault

Status	Condition	Action
Normal	External current sensor is connected and in use.	***BMS Dual Port* mode:** ***Load Switch*** – turned on, if *Ignition* feature disabled. ***Load Switch*** – depending on *Ignition* signal, if *Ignition* feature enabled. ***Charger Switch*** – turned on, if charger detected, otherwise turned off.

		BMS Single Port **mode:**
		Single Switch – turned on, if *Ignition* feature disabled. ***Single Switch*** – depending on *Ignition* signal and the charger detection, if *Ignition* feature enabled.
Fault	External current sensor is accidentally disconnected.	*BMS* device generates *External Current Sensor Disconnected (BMS restart required)* event (*0x0E*). ***BMS Dual Port*** **mode:** ***Load Switch*** – turned off. ***Charger Switch*** – turned off. ***BMS Single Port*** **mode:** ***Single Switch*** – turned off. Tiny BMS onboard buzzer generates beeping sound signal to warm the user about disconnected external current sensor.
Recovery	External current sensor is connected again, or *BMS* was manually restarted. After *BMS* restart, if external current sensor was not connected to *BMS*, for current measurement internal *BMS HALL* sensor is used.	*BMS* device generates *External Current Sensor Connected* event (*0x7c*), if external current sensor was connected again. *BMS* goes back to normal status operation*.*

3.2. Tiny BMS operation states

During operation *Tiny BMS* can enter one of the following operation states: *Sleep*, *Idle*, *Charging*, *Discharging*, *Regeneration* and *Warning / Fault*. The relationship between these states are shown in Figure *3.1*.

BMS is in *Idle* state when there is no charging, discharging or regeneration activity and the *Tiny BMS* measured current is lower than activity threshold. This current threshold value is hard-coded and dependent on *Tiny BMS* hardware versions. Due to internal *Shunt* resistor for a low range current measurement on a *Low Power 30 A Tiny BMS* version this activity threshold value is *±80 mA*. On the *High Power 150 A Tiny BMS* version this activity threshold value is *±800 mA*, because for high current range measurement internal *HALL* or external current sensor is used. After *10* seconds in *Idle* state without any activity and if all other *Sleep* state conditions are met *BMS* enters *Sleep* state to reduce its power consumption. *Tiny BMS* in *Sleep* state performs basic measurements and triggers wake up from *Sleep* mode action if necessary. *Tiny BMS* woken up from *Sleep* state can be in any other *Idle*, *Fault*, *Charging*, *Discharging* or *Regeneration* state. It depends on what event triggers *BMS* to wake up. More about *Tiny BMS Sleep* mode refer to *Chapter 3.3: Tiny BMS Sleep mode.*
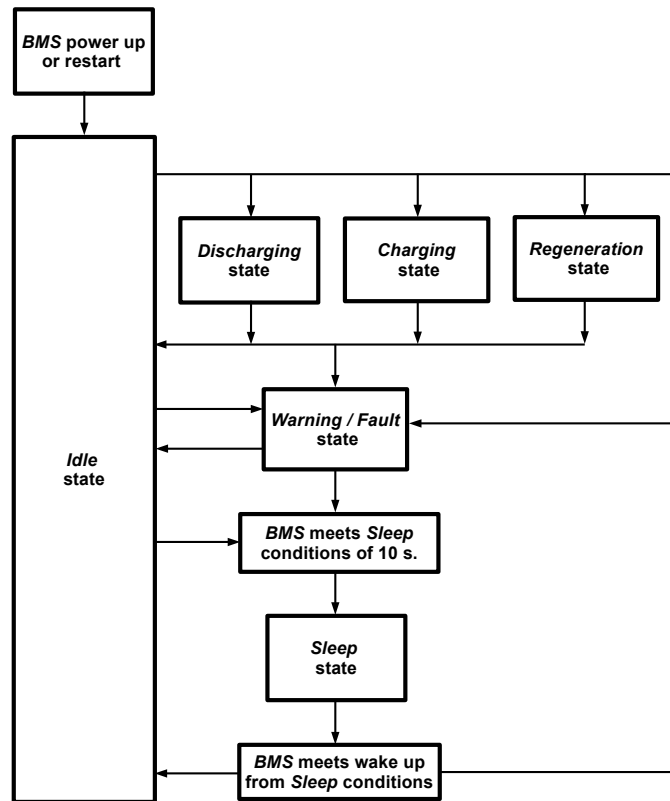
Figure 3.1: Tiny BMS operation states

BMS enters *Discharging* state when measured current is negative and lower than *-80 mA* on the *Low Power 30 A Tiny BMS PCB* version or *-800 mA* on the *High Power 150 A Tiny BMS PCB* version. When charger is connected to *BMS* (*BMS* generates *Charger Connected* event) and measured current is positive and higher than *+80 mA* on the *Low Power 30 A Tiny BMS PCB* version or *+800 mA* on the *High Power Tiny BMS PCB* version, *BMS* enters the *Charging* state. Otherwise if charger is not connected to *BMS*, but measured current is positive and higher than activity threshold, *BMS* enters to *Regeneration* state (possible only in *Dual Port* mode). When *BMS* is in *Charging*, *Regeneration* or *Over-Voltage Fault* condition states cells balancing is performing. *Tiny BMS Warning / Fault* state is discussed in detail in *Chapter 3.1: Tiny BMS protections*.

3.3. *Tiny BMS* sleep mode

To enhance battery life, the *Tiny BMS* device supports several power modes to minimize power consumption during operation. In normal mode, the device takes voltage, current and temperature readings every *100 ms*, performs protection and *SOC* gauging calculations, updates live, statistics and events data and makes status decisions. When the sleep conditions are met, the device goes to *Sleep* mode with periodic wakeups for current, voltage or temperature measurements to reduce its power consumption. The

Tiny BMS device returns to normal active mode if any exit *Sleep* condition is met. To know what is *Tiny BMS* current consumption in different device operation modes and setups refer to *Tiny BMS* datasheet. It's important to note, that *Tiny BMS* device can achieve the lowest current consumption (<1*00 μA*) in *Sleep* mode when *Ignition* feature is enabled, independently of the *Tiny BMS PCB* hardware or power version. The table below shows what conditions triggers *Tiny BMS* to enter or exit the low power *Sleep* state.

Table 3.11: Tiny BMS Sleep mode conditions

Status	Condition	Action
Enter *Sleep* state	***Ignition* feature disabled:** No *UART* communication *and* Charger is disconnected *and* No *Over-Current Fault and* *(*No *Over-Voltage Fault or* *Under-Voltage Fault* occurred *or* *Fully-Discharged Fault* Occurred*) and* *(-80 mA < Low Power BMS* current *< 80 mA or* *-800 mA < High Power BMS* current *< 800 mA) and* *10 s. Sleep Timeout* passed. ***Ignition* feature enabled:** (No *Ignition* signal connected (low logic level) *or* *Ignition* signal connected (high logic level), but *Fully Discharged Warning* or *Under-Voltage Fault* occurred*) and* No *UART* communication *and* Charger is disconnected *and* *10 s. Sleep Timeout* passed.	***Ignition* feature disabled (no *Warnings / Faults*):** *BMS* goes to *Sleep* state. Internal *5 V* power supply is turned off (no *5 V* output on *UART* and *External I/O* connectors). On *Dual Port* mode discharging *DFET* or external relay / contactor is turned on, charging *CFET* or external relay / contactor is turned off. On *Single Port* mode *DFET*, *CFET* or external relay / contactor is turned on. *BMS* periodically wakes up every *1 s.* time period to perform current measurements, charger and *UART* communication detections.Cells voltage is measured every *1 min*. time period. ***Ignition* feature enabled (no *Warnings / Faults*):** *BMS* goes to *Sleep* state. Internal *5 V* power supply is turned off (no *5 V* output on *UART* and *External I/O* connectors). On *Dual Port* mode discharging *DFET* or external relay / contactor and charging *CFET* or external relay / contactor are turned off. On *Single Port* mode *DFET*, *CFET* or external relay / contactor is turned off. *BMS* periodically wakes up every *1 s.* time period to perform, charger, ignition and *UART* communication detections.
Exit *Sleep* state	***Ignition* feature disabled:** Ongoing *UART* communication *or* Charger is connected *or* *(Low Power BMS* current *< -80 mA or Low Power BMS* current *>* 80 mA*) or* *(High Power BMS* current *< -800 mA or High Power BMS* current *> 800 mA) or* *Warning / Fault* state recovered in *Sleep* mode. ***Ignition* feature enabled:**	***Ignition* feature disabled:** *BMS* wakes up from *Sleep* mode. Internal *5 V* power supply is turned on. On *Dual Port* mode discharging *DFET* or external relay / contactor is turned on, charging *CFET* or external relay / contactor is turned on, if Charger is connected. On *Single Port* mode *DFET*, *CFET* or external relay / contactor is turned on.

Ongoing *UART* communication or Charger is connected or (Ignition signal connected (high logic level) and No Fully-Discharged Warning occurred and No Under-Voltage Fault occurred)	***Ignition* feature enabled:** *BMS* wakes up from *Sleep* mode. Internal *5 V* power supply is turned on. On *Dual Port* mode discharging *DFET* or external relay / contactor state depends on Ignition signal, charging *CFET* or external relay / contactor is turned on, if charger is connected. On *Single Port* mode *DFET*, *CFET* or external relay / contactor state depends on *Ignition* signal and charger detection.

As it was said before *Tiny BMS* device performs some basic parameters measurement in *Sleep* mode to trigger wake up from *Sleep* mode on certain events or on *Warning* / *Fault* recovery conditions. In *Table 3.12* is summarized *Tiny BMS* device measurements in *Sleep* mode. In detail covered when and under what conditions these measurements are taken. *Tiny BMS* power consumption also depends on what conditions and features enabled *BMS* device enters *Sleep* mode and what measurements are taken during *Sleep* state (refer *Tiny BMS* datasheet).

Table 3.12: Tiny BMS Sleep mode measurements

Measurement in *Sleep* mode	Conditions
BMS current	*BMS* current measurement is taken every *1 s*. time period, if *Ignition* feature was disabled and unused, and, if *Load Switch* (*Dual Port* mode), or *Single Port Switch* is turned on. To measure current in *Sleep* mode *BMS* turns on its internal *5 V* power supply for a very short period of time. There is a very short *5 V* pulses on *External I/O* connector 5 V output pins. *BMS* wakes up from *Sleep* mode, if measured current hits wake up from *Sleep* mode current threshold level (*±500 mA* for *Low Power BMS* version*, ±800 mA* for *High Power BMS* version*).*
Temperature	*BMS* temperature measurement is taken every *1 s*. time period, if several conditions are met: if *Over-Temperature Fault*, *Charge Low Temperature Warning* or *Discharge Low Temperature Fault* has been occurred, but no *Fully Discharged Fault* or *Under-Voltage Fault* has occured and *Ignition* feature is disabled. Another case, if no temperature *Fault* has been occurred, but in *Sleep* mode *Load Switch* (*Dual Port* mode) or *Single Port Switch* is turned on. The *BMS* internal temperature is always measured if the above conditions are met, and external sensors are only measured if they were detected before switching to sleep.
Charger detection	Charger detection status in *Sleep* mode *BMS* monitors continuously every *1 s*. time period in both cases when *BMS* internal or external charger detection is used.
Load detection	Load detection status in *Sleep* mode *BMS* monitors every *1 s*. time period only when *BMS* operates as *Single Port* mode and *Over-Voltage Fault* or *Charge Low Temperature Warning* has been occured, *Ignition* feature is disabled and battery discharging must be allowed when load is connected to *BMS*.
Ignition signal detection	*Ignition* signal status in *Sleep* mode *BMS* monitors continuously every *1 s*. time period if *Ignition* feature is enabled.

Cells voltages	Battery cells voltages measurement is taken every *1 min*. time period, if *Ignition* feature was disabled and unused, and, if *Load Switch* (*Dual Port* mode), or *Single Port Switch* is turned on. In such case *BMS* measures current in *Sleep* mode and turns on its internal *5 V* power supply for a very short period of time and current consumption is increased. When *BMS* measures cells voltage and detects that cells voltage drops below *Under-Voltage Cutoff* threshold level, *BMS* turns off all *Switches* and enters lowest possible power consumption mode (< *100 µA*).

3.4. *Tiny BMS* current measurement

Tiny BMS continuously (at a steady rate of *10 Hz*) measures battery pack current in active mode to determine battery pack state (charging, idle or discharging) and estimate it's State-of-Charge level. Current measurement process depends on different *Tiny BMS* hardware versions. On the *30 A* low power *Tiny BMS* version *Shunt* resistor is used to measure current flowing through *Tiny BMS device*. Basically, current sampling is done by the separate *AFE* chip and *Tiny BMS* main microcontroller reads digitized current value from *AFE* chip. On this version current calculation almost does not depend on diferent *Tiny BMS* firmware versions. When current is higher than *80 mA* threshold level *Tiny BMS* detects battery charging action and when current is lower than *-80 mA, Tiny BMS* detects battery discharging action. In all other cases *Tiny BMS* stays in idle or sleep state. On the *150 A* high power *Tiny BMS* version internal *HALL* current sensor is used for current calculation. Alternatively, external current sensor can be used. There is not possible to use both sensors at the same tame, when external current sensor is connected and in use, internal *HALL* current sensor is deactivated. For both internal *HALL* and external current sensors charging / discharging state detection threshold level is the same. When current is higher than *800 mA* threshold level *Tiny BMS* detects battery charging and when current is lower than *-800 mA* discharging is detected. For high power *BMS* device the measurement of battery current comprises of continuous, simultaneously executed process: the *BMS* main microcontroller continuously measures the analog signals from internal or external current sensor using a high sample rate analog to digital converter peripheral and periodically stores averaged measurement results for further processing. The averaging is needed in order to mitigate the effects of thermal, quantization and other noises. The most recent results of measurement process are taken and used for calculating the battery current value and storing the result for use in higher level algorithms. Because whole current measurement process for *150 A Tiny BMS* version is controled by the main microcontroller, the current measurement algorithms are constantly improved and can slightly differ between firmware versions.

3.5. *Tiny BMS SOC* estimation

State-of-Charge (*SOC*) estimation in *Tiny BMS* device is mainly based on the coulomb counting technique, with the exception where the *SOC* value is adjusted to 100 % on *Fully Charged* event of battery pack in order to mitigate its drawbacks. The coulomb counting is done in sync with the current update process: every newly determined momentary current value is multiplied by the current value update period, and the result, regardless of its sign, is accumulated and compared to battery capacity value in order to determine the *SOC* value expressed in percentage. The *SOC* value in any way is not adjusted by cells voltage, that's why *Tiny BMS* after initial setup process uses user defined *SOC* setting value. This setting (*Set SOC manually*) is located in the *Battery Insider Cell Settings* tab. When battery pack is fully charged (*BMS* shows the *Fully Charged* state), *Tiny BMS* sets its *SOC* level to 100% and the user defined *SOC* value becomes unnecessary and is no longer used. For more details, when the *Tiny BMS* device enters *Fully Charged* state refer to *Chapter 3.7. Tiny BMS charging process*. The *SOC* value is stored in the backup *SRAM* data memory and when *Tiny BMS* reset condition occurs the *SOC* value is retained from backup

memory and is recorded to the non-volatile flash memory. When *Tiny BMS* is disconnected from the battery and then powered up again, the *SOC* value is reset to the last known *SOC* value stored in the Flash memory or user defined initial *SOC* value.

3.6. *Tiny BMS SOH* estimation

In an ideal situation, a battery cycle means the battery is fully discharged to 0% and then fully charged back to 100% battery capacity. But in real-life use, this rarely happens — batteries are often only partially used (for example, discharged to 50% and then recharged). These are called partial cycles, and they accumulate over time. To track how much the battery is actually being used, Battery Management System (BMS) uses a technique called coulomb counting. This measures how much energy (in ampere-hours, or Ah) is discharged from the battery over time. For example:

- If a battery is rated at 3Ah, and the BMS sees that 3Ah has been discharged (even if it happened in small parts), we count this as one full cycle.
- So if the BMS sees a total of 30Ah discharged, it assumes around 10 full cycles have occurred.

Every battery has a maximum number of charge/discharge cycles it can handle before its capacity starts to drop significantly. This is usually defined in the battery manufacturer's datasheet. The system, includes this as a configurable setting called Battery Maximum Cycles Count.

Example

To see this feature in action, you can try the following:

- Set the Battery Capacity to 1Ah.
- Set the Maximum Cycle Count to 10.
- Discharge the battery at 1A for 1 hour (or 2A for 30 minutes) and then recharge it at 1A for 1 hour.
- This simulates one full cycle, and the SOH (State of Health) will drop from 100% to 90%, assuming linear degradation.

This feature helps estimate how much life is left in the battery based on real usage, which is crucial for long-term reliability and planning.

3.7. *Tiny BMS* charger detection

Due to different *Tiny BMS* configurations and use in various battery pack applications, *Tiny BMS* supported several charger detection modes. Internal charger detection is mainly used when *BMS* device internal *CFET* is used for charger control, otherwise when external relay / contactor is used for charger control, external charger detection signal connected to one of the configurable external input (*AIDO*, *DIDO*, *AIHO*) should be used (*Chapter 2.3.11*). Alternatively, various combinations of internal / external charger detection and internal *FET* or external relay / contactor can be used depending on the needs of battery application configuration. For safety reasons, charger switch (internal *FET* or external relay / contactor) is kept turned off when charger is not detected and, if the charger is detected, charger switch is turned on and the battery is allowed to charge.

Note: Due to smart internal *Tiny BMS* charger detection logic somethimes there are some isues with smart chargers, which does not apply voltage, until it detects battery connected. The smart charger can not detect battery connected due to closed *Tiny BMS* internal *FETs* and *Tiny BMS* cannot detect the charger connected, because it does not apply charging voltage and we get two smart devices toggling each other, struggle in a loop and missing the window to turn on each other.

Below in the *Table 3.13* is a detiled description how internal and external charger detection works for different *Tiny BMS* device operation modes.

Table 3.13: Tiny BMS charger detection

Charger detection	Operation description
Internal	***Dual Port* operation mode:** *Charger switch* is turned off, if charger not detected. *Charger switch* is truned on, if charger is detected. Charger is detected, when charger is connected to charging port (*C-*) and charger voltage is higher than battery pack voltage. When *Tiny BMS* detects connected charger, event *Charger Connected (0x64)* is generated, charging switch is turned on and *Tiny BMS* state becomes *Charging*. Charger disconnection is detected by the current. When charging current drops below *80 mA* (*30 A* low power *BMS*) or *800 mA* (*150 A* high power *BMS*), the *Tiny BMS* finds that the charger has been disconnected, turns off charger port and generates event *Charger Disconnected (0x65)*. If he charger is disconnected until the current from the charger starts to flow, the event *Charger Disconnected (0x65)* will only be generated after *~20* seconds, otherwise, if the charging current has been raised before the charger is disconnected, the event *Charger Disconnected (0x65)* will be generated after *~2* seconds. The charger is detected again when it is physically disconnected from the *BMS* and reconnected again. ***Single Port* operation mode:** *Single Switch* is turned on independently to charger detection. Charger is detected, when charger is connected and the charging current starts to flow. *Tiny BMS* generates *Charger Connected (0x64)* event and *BMS* state becomes *Charging*. If charging current drops below *80 mA* (*30 A* low power *BMS*) or *800 mA* (*150 A* high power *BMS*), event *Charger Disconected (0x65)* is generated and BMS state becomes *Idle* or *Discharging*. *Single Switch* stays turned on. Charger is detected again, if charging current starts flowing again and the whole cycle is restarted.
External (*AIDO / DIDO / AIHO*)	***Dual Port* operation mode:** *Charger switch* is turned off, if charger not detected. *Charger switch* is truned on, if charger is detected. Charger is detected, when a high logic level voltage is connected to configured external charger detection pin. When *Tiny BMS* detects external charger detetcion signal *Charger Connected (0x64)* event is generated, charging switch is turned on and the *Tiny BMS* state becomes *Charging*. Charger disconnection is detected when there is low logic level on the configured external charger detection pin. In this case, the BMS does not check the charging current and does not disconnect the *Charger Switch* when the charging current drops. ***Single Port* operation mode:** *Single Switch* is turned on independently to charger detection. If the *BMS* status is *Idle* and *Single Switch* is turned on, the *BMS* detects the charger according to the current, independently the external charger

	detection pin is connected or not. If the *BMS* is in the *Fault* state or *Single Switch* is turned off, the charger is detected according to the external pin status. If there is a high logic level on the external detetction pin, *BMS* generates *Charger Connected (0x64)* event, accordingly the *BMS* status becomes *Charging* and *Single Switch* is turned on. When the charging current is depleted, *Charger Disconnected (0x65)* event is generated and the *BMS* status becomes *Idle* or *Discharging*, even if the charger detetion pin is still connected. When the charging current starts flowing again, the charger is detected again and the whole cycle is restarted.

3.8. *Tiny BMS* charging process

Currently *Tiny BMS* device supports only generic *CC/CV* chargers, which is connected directly to *Tiny BMS* charging port. *Tiny BMS* controls the charging process only by turning on or off charging port (internal *CFET* or external relay / contactor). During charging process cells groups connected in series are balanced. *Tiny BMS* charging and cells balancing process starts when charger is detected, *Charger Connected (0x64)* event is generated, charger switch is turned on and *BMS* state becomes *Charging*. During battery charging process, if at least one cell voltage reaches *Fully Charged Voltage*, *Charger Switch* (*Dual Port* mode) or *Single Port Switch* (*Single Port* mode) is turned off (battery charging is stopped) and only cell balancing is performed. When the cells imbalance becomes less than the *Allowed Disbalance* setting, the *Charger Switch* (*Dual Port* mode) or *Single Switch* (*Single Port* mode) is restarted and the battery charging continues. This cycle continues until the battery fully charged conditions are met. The battery is fully charged, the *Charging Done (0x63)* event is generated, *BMS* goes to the *Fully Charged* state and the *SOC* value is set to the *100 %* value only if all three conditions are met:
- The voltage of the all cells is at the *Fully Charged* voltage limit;
- The cells imbalance is less than the *Allowed Disbalance* setting;
- The charging current becomes lower than *Charge Finished Current* setting value.

When the *BMS* is in the *Fully Charged* state, the battery charging process is restarted only by physically disconnecting the charger from the *BMS* and reconnecting it again, or at least one of the cells voltages reaches *Fully Discharged Voltage* level. If the charging voltage is too high (cells voltages is not balanced correctly and exceeds *Fully Charged Voltage* threshold level, and the charging current is still higher than *Charge Finished Current* limit), *BMS* status becomes also *Fully Charged*, but *SOC* is not adjusted to *100 %* and event *Charging Done (charger voltage too high) (0x38)* is generated. Also, in *Tiny BMS Dual Port* operation mode, if the external charger detection feature is used and cells voltages have reached *Fully Charged Voltage* at least once, but after cells balancing stage, charging current drops below *Charge Finished Current* level, but the cells voltages does not reach the *Fully Charged Voltage* limit – *BMS* state becomes *Fully Charged*, but *SOC* is not adjusted to *100 %* and the event *Charging Done (charger voltage to low) (0x39)* is generated. Alternatively, if the charger detection is set to *Internal* and the charger voltage is too low and the charging current drops below *Fully Charged Current* limit, *Charger Disconnected* (*0x65*) event is generated.

3.9. *Tiny BMS* cells balancing process

The cell balancing algorithm used in the *Tiny BMS* device decreases the differences in imbalanced cells in a fully charged state gradually, which prevents fully charged cells from becoming overcharged, causing excessive degradation. This increases overall pack energy by preventing premature charge termination. Tiny BMS activates the cells balancing process when the battery is in the charging state, or in the case of *Over-Voltage Fault* or regeneration (*Dual Port* mode only) states. During charging and regeneration states,

the cells balancing process continues until the cells imbalance becomes less than the *Allowed Disbalance* setting, while in the *Over-Voltage Fault* state, balancing process continues until the fault state is cleared, even if the cells imbalance is lower than the *Allowed Disbalance* setting. During cells balancing process each cell can be balanced, if its voltage is higher than *Early Balancing Threshold* voltage level. Due to *Tiny BMS* hardware limitation only every third cell can be balanced at a time (adjacent cells can not be balanced at a time). Before the *Tiny BMS* switches itself to sleep mode, the cells balancing process is always stoped.

3.10. *Tiny BMS Ignition* feature

Tiny BMS Ignition feature can be used to control (turn on or off) load output (internal *FET* or external relay / contactor) and to enter deep sleep mode to achieve the lowest *Tiny BMS* power consumption. More about *Tiny BMS Ignition* setup examples and wiring diagrams refer to *Chapter 2.3.9*. Before using *Tiny BMS Ignition* feature, it must be enabled by using the *Battery Insider* windows application. Once *Ignition* feature is enabled, the load output can be controlled by configured external *I/O* pin connected to the logic high voltage level (*3 V* to *75 V*) or low logic voltage level (*0 V* to *1.5 V* or left unconnected). The summary of the *Ignition* control signals behaviour is provided in the *Table 3.14*.

Table 3.14: Tiny BMS Ignition feature summary

Ignition signal	Operation description
Ignition disabled	*Load Switch* (*Dual Port* mode), or *Single Switch* (*Single Port* mode) is always turned on, if there are no *Warnings* or *Faults*. *Tiny BMS* can wake up from sleep mode due to discharging current flow. Increased energy consumption in sleep mode, because *Tiny BMS* device measures current each second time period.
Ignition enabled, high logic level on configured *Ignition* input pin.	*Load Switch* (*Dual Port* mode), or *Single Switch* (*Single Port* mode) is turned on, if there are no *Warnings* or *Faults*. *Tiny BMS* never enters the sleep mode until the high logic level is present to *Ignition* pin, except one condition when *Under-Voltage Cutoff Fault* has been occurred.
Ignition enabled, low logic level on configured *Ignition* input pin.	*Load Switch* (*Dual Port* mode), or *Single Switch* (*Single Port* mode) is turned off. *Tiny BMS* can not wake up from sleep mode due to discharging current flow. *Tiny BMS* achieve the lowest power consumption in sleep mode. **Note:** In *Dual Port* mode, if *Load Switch Type* is configured as *Discharge FET* and *Charger Switch Type* is configured as *Charge FET,* and charger is connected to *BMS, Load Switch* is turned on, regardless of *Ignition* signal state. In *Single Port* mode, if charger is connected in all cases *Single Switch* is turned on, regardless of *Ignition* signal.

3.11. *Tiny BMS* load *Precharge* feature

The *Tiny BMS* load *Precharge* feature is useful in a high current battery applications, where load contactor is used to control load and where high in-rush current occurs during switching the load contactor in order to reduce stress to the internal components of the load device and prevent contactor from welding. More about *Tiny BMS* load *Precharge* setup examples and wiring diagrams refer to *Chapter 2.3.13*. Load *Precharge* feature can be used only when contactor (driven by external *Tiny BMS I/O* pin) is used for load control. Once the load *Precharge* feature is activated by the *Battery Insider* application, the two *Tiny BMS* external *I/O* pins are dedicated for load contactor control. One external pin (*Load Switch Type*) controls contactor, the other (*Precharge*) – controls a relay connected in series with a resistor whitch bypasses the main load contactor. Once the load contactor should be turned on, at first for a *Precharge Duration* time

period the *Precharge* output (relay) is turned on while keeping main contactor turned off. After the *Precharge Duration* time interval has been passed, main load contactor is turned on and *Precharge* relay is turned off. In the *Tiny BMS Dual Port* operation mode if both the load and the charger is controlled by the relays / contactors internal *Tiny BMS FETS* can be used as *Precarger* switch.

3.12. *Tiny BMS* events

Tiny BMS device during operation records a log of the most recent events in its internal non-volatile memory. It consists of 49 circular buffer entries, where each of which stores an event occurrence timestamp and event code. Once all 49 entries are filled, the oldest entry is replaced with the newest system generated event. All events are described in the Table 3.15.

Note: *Clear Events* button in the *Battery Insider* application only clears all events from BMS internal memory events list, but not recovers BMS from warning or fault states. BMS recovers from warning or fault states only when all conditions that caused BMS to enter this state are cleared.

Table 3.15: Tiny BMS events

Importance	Code	Event message	Event description
Fault	0x02	*Under-Voltage Cutoff Occurred*	Permanent second level cell under-voltage protection has been ativated. At least one cell voltage droped below *Under-Voltage Cutofl* threshold.
Fault	0x03	*Over-Voltage Cutoff Occurred*	Recoverable cell over-voltage protection has been activated. At least one cell voltage is hifher than *Over-Voltage Cutofl* threshold.
Fault	0x04	*Over-Temperature Cutoff Occurred*	Recoverable over-temperature protection has been activated. Battery or *BMS* temperature is higer than *Over-Heat Cutoff* threshold.
Fault	0x05	*Discharging Over-Current Cutoff Occurred*	Recoverable over-current protection has been activated. Discharging current is higher than *Discharge Over-Current Cutoff* threshold.
Fault	0x06	*Charging Over-Current Cutoff Occurred*	Recoverable over-current protection has been activated. Charging current is higher than *Charge Over-Current Cutoff* threshold.
Fault	0x07	*Regeneration Over-Current Cutoff Occurred*	Recoverable over-current protection has been activated (possible only in *Dual Port* operation mode). Regeneration current is higher than *Discharge Over-Current Cutoff* threshold.
Fault	0x0A	*Low Temperature Cutoff Occurred*	Recoverable under-temperature protection has been activated. Battery temperature is lower than *-40°C.*
Fault	0x0B	*Charger Switch Error Detected*	*Charger Switch* (*Dual Port* mode) should be turned off, but *BMS* still measures charging current. Possible *BMS* internal *FETs* or relay / contactor damage.
Fault	0x0C	*Load Switch Error Detected*	*Load Switch* (*Dual Port* mode) should be turned off, but *BMS* still measures discharging / regeneration current. Possible *BMS* internal

			FETs or relay / contactor damage.
Fault	0x0D	*Single Port Switch Error Detected*	*Single Switch* (*Single Port* mode) should be turned off, but *BMS* still measures charging / discharging current. Possible *BMS* internal *FETs* or relay / contactor damage.
Fault	0x0E	*External Current Sensor Disconnected (BMS restart required)*	External current sensor disconnected. *BMS* restart is required to properly use *Tiny BMS* internal *HALL* current sensor.
Fault	0x0F	*External Current Sensor Connected (BMS restart required)*	External current sensor connected, but not initialized. *BMS* restart is required to properly use external current sensor.
Warning	0x31	*Fully Discharged Cutoff Occurred*	Recoverable under-voltage protection has been activated. At least one cell voltage droped below *Fully-Discharged Voltage* threshold.
Warning	0x37	*Low Temperature Charging Cutoff Occurred*	Recoverable charging under-temperature protection has been activated. Battery temperature is lower than *Low Temperature Charger Cutoff* threshold.
Warning	0x38	*Charging Done (Charger voltage too high)*	Charging process is stopped, due to excessive charger output voltage. *SOC* is not adjusted to *100 %.*
Warning	0x39	*Charging Done (Charger voltage too low)*	Charging process is stopped, due to low charger output voltage. *SOC* is not adjusted to *100 %.*
Info	0x61	*System Started*	*Tiny BMS* device powered up or restarted.
Info	0x62	*Charging Started*	Charger is connected and *Charger Switch* is turned on.
Info	0x63	*Charging Done*	Battery is fully charged. *SOC* adjusted to *100 %*
Info	0x64	*Charger Connected*	Charger is connected.
Info	0x65	*Charger Disconnected*	Charger is disconnected.
Info	0x66	*Dual Port Operation Mode Activated*	*Tiny BMS* is switched to *Dual Port* operation mode.
Info	0x67	*Single Port Operation Mode Activated*	*Tiny BMS* is switched to *Single Port* operation mode.
Info	0x73	*Recovered From Over-Temperature Fault Condition*	Over-temperature protection has been deactivated.
Info	0x74	*Recovered From Low Temperature Warning Condition*	Charging under-temperature protection has been deactivated.
Info	0x75	*Recovered From Low Temperature Fault Condition*	Under-temperature protection has been deactivated.
Info	0x76	*Recovered From Charging Over-Current Fault Condition*	Over-temperature protection has been deactivated.
Info	0x77	*Recovered From Discharging Over-Current Fault Condition*	Discharging over-current protection has been deactivated.
Info	0x78	*Recovered From Regeneration Over-Current Fault Condition*	Regeneration over-current protection has been deactivated (possible only in *Dual Port* operation mode).

Info	0x79	*Recovered From Over-Voltage Fault Condition*	Over-voltage protection has been deactivated.
Info	0x7A	*Recovered From Fully Discharged Warning Condition*	Under-voltage (*Fully-Discharged Warning*) protection has been deactivated.
Info	0x7B	*Recovered From Under-Voltage Fault Condition*	Permanent under-voltage protection has been deactivated.
Info	0x7C	*External Current Sensor Connected*	External current sensor connected, initialized and in use.
Info	0x7D	*External Current Sensor Disconnected*	External current sensor disconnected, *BMS* internal *HALL* current sensor is used.

3.13. *Tiny BMS* statistics

Tiny BMS device collects some useful data (statistics) for analysis purpose during normal operation over the life time of the battery. During *Tiny BMS* restart or at the crytical moments of *Tiny BMS* operation recorded statistics data is saved to *BMS* internal non-voltaile memory. All the *Tiny BMS* device statistics data is listed under the *Battery Insider Statistics* tab, alternativelly, it can be read out directly from *Tiny BMS* internal memory over *USB-UART,* or *CAN* bus interfaces using a user side integrated device.

3.14. *Tiny BMS* data broadcast

Tiny BMS provides data broadcasting feature, when all basic *BMS* and battery data are transmited over the *UART* communicaton interface without data requesting at each broadcast duration time period. Currently there are predefined three data broadcast protocols *CA V3*, *ASCII* and *SOC BAR*. *CA V3* data protocol bassically is used for data transmision over bluetooth communication between *Tiny BMS* device and *Cycle Analyst* android application. How to properly configure and connect *Tiny BMS* device with *Ebike Analyzer* android application, refer to *Chapter 4*. *ASCII* data protocol is usefull for battery data logging. *SOC BAR* broadcast protocol is only used wit *Enepaq SOC-BAR LED* indicator. Once *Tiny BMS* broadcast protocol and broadcast duration is enabled by *Battery Insider* application, it starts transmitting data when there is no other ingoing *UART* commands. When other *UART* protocol command is send to *Tiny BMS*, it stops the data broadcasting, responses to a received command and starts data broadcasting again. Currently, data broadcasting is not possible over *CAN* interface when *Enepaq UART-CAN* converter is used. The summary of the broadcast protocols is provided in the *Table 3.16*.

Note: The *Tiny BMS* data broadcasting over the CAN bus interface is not implemented at the current *Tiny BMS* and *Enepaq UART-CAN* converter firmware versions.

Table 3.16: Tiny BMS broadcast protocols

Protocol	Description
CA V3	Data in float type is being sent as tab delimited *ASCII* text format: [*Value 1*][Tab][*Value 2*][Tab][*Value 3*][Tab][*Value 4*][Tab][*Value 5*][Tab][*Value 6*][Tab][*Value 7*][Tab][*Value 8*][Tab][*Value 9*][Tab][*Value 10*][Tab][*Value 11*][Tab][*Value 12*][Tab][*Value 13*] [*Value 1*] – Remaining charge of battery pack until empty (Ah); [*Value 2*] – Battery pack voltage (*V*); [*Value 3*] – Battery pack current (*A*); [*Value 4*] – Speed (*km/h*); [*Value 5*] – Total traveled distance (*km*);

[*Value 6*] – Battery pack max temerature (°C);
[*Value 7*] to [*Value 13*] – all zeros.

Real exmaple:
"*0.690000 49.048000 -3.013000 0.000000 0.000000 23.400000 0 0 0 0 0 0 0*"
Left battery pack capacity until empty (*0.69 Ah*);
Battery pack voltage (*49.048 V*);
Battery pack current (*-3.013 A*);
Speed (*0 km/h*);
Total traveled distance (*0 km*);
Battery pack max temerature (*23.4 °C*);

ASCII	Data is being sent as tab delimited *ASCII* text format. **Dual Port** operation mode: BPV:[*Value 1*][Tab]MINCV:[*Value 2*][Tab]MAXCV:[*Value 3*][Tab]BPC:[*Value 4*][Tab] SOC:[Value5][Tab]BMST:[*Value 6*][Tab]EXT1:[*Value 7*][Tab]EXT2:[*Value 8*][Tab] OS:[*Value 9*][Tab]CB:[*16 Bits*][Tab]LST:[*1 bit*][Tab]LSS:[*1 bit*][Tab]CST:[*1 bit*][Tab]CSS:[*1 bit*]

BPV – Battery pack voltage (*V*);
MINCV – Min cell voltage (*V*);
MAXCV – Max cell voltage (*V*);
BPC – Battery pack current (*A*);
SOC – Battery pack *SOC* (%);
BMST – Internal temperature sensor temperature value (°C);
EXT1 – External temperature sensor *#1* temperature value (°C);
EXT2 – External temperature sensor *#2* temperature value (°C);
OS – BMS online status;
CB – Real cells balancing bits (*1* – cell is balanced);
LST – *Load Switch* type bit (*0* – *FET*, *1* – External *I/O*);
LSS – *Load Switch* status bit (*0* – OFF, *1* – ON);
CST – *Charger Switch* type bit (*0* – *FET*, *1* – External *I/O*);
CSS – *Charger Switch* status bit (*0* – OFF, *1* – ON).

Single Port operation mode:
BPV:[*Value 1*][Tab]MINCV:[*Value 2*][Tab]MAXCV:[*Value 3*][Tab]BPC:[*Value 4*][Tab]
SOC:[*Value5*][Tab]BMST:[*Value 6*][Tab]EXT1:[*Value 7*][Tab]EXT2:[*Value 8*][Tab]
OS:[*Value 9*][Tab]CB:[*16 Bits*][Tab]ST:[*1 bit*][Tab]SS:[*1 bit*]

BPV – Battery pack voltage (*V*);
MINCV – Min cell voltage (*V*);
MAXCV – Max cell voltage (*V*);
BPC – Battery pack current (*A*);
SOC – Battery pack *SOC* (%);
BMST – Internal temperature sensor temperature value (°C);
EXT1 – External temperature sensor *#1* temperature value (°C);
EXT2 – External temperature sensor *#2* temperature value (°C);
OS – BMS online status;
CB – Real cells balancing bits (*1* – cell is balanced);
ST – *Switch* type bit (*0* – *FET*, *1* – External *I/O*);
SS – *Switch* status bit (*0* – OFF, *1* – ON);

	Real exmaple: "*BPV:49.05 MINCV:3.766 MAXCV:3.776 BPC:-3.03 SOC:80 BMST:23.60 EXT1:24.35 EXT2:24.63 OS:151 CB:0000000000000000 LST:0 LSS:1 CST:0 CSS:0*" Battery pack voltage (*49.05 V*); Min cell voltage (*3.766 V*); Max cell voltage (*3.776 V*); Battery pack current (*-3.03 A*); Battery pack *SOC* (*80 %*); Internal temperature sensor temperature value (*23.6 °C*); External temperature sensor #1 temperature value (*24.35 °C*); External temperature sensor #2 temperature value (*24.63 °C*); *BMS* online status (*151 – Idle*, refer to *Tiny BMS* communication protocols document); Real cells balancing (cells are not balanced); *Load Switch* type (*FET*); *Load Switch* status (*ON*); *Charger Switch* type (*FET*); *Charger Switch* status bit (*OFF*).
SOC BAR	Used only for *Enepaq SOC-BAR LED* indicator. Only *SOC* value is send in a specific format. No readable data is transmitted over the *UART* interface.

3.15. *Tiny BMS* speed measurement

One of the *Tiny BMS* external digital inputs (*DIDO*) can be mapped with vehicle speed measurement feature of the *BMS* device, which eliminates the need for a separate electronics for these functions and allows to simplify the electrical system of the vehicle. Once feature enabled, and signal from speed sensor is connected to configured external input (refer to *Chapter 2.3.10*), *BMS* device measures the pulses and the duration between them of the vehicle's speed signal. Measured duration between pulses is used to calculate pulses per second, which is divided by the value of *Pulses Per Unit* in kilometers in order to determine the momentary vehicle speed. In the *Battery Insider* user can enter *Pulses per Unit* and *Distance Unit* parameters not necessarily in kilometers, but also in other distance units, but *BMS* itself all distance units converts to kilometers and speed is calculated in *km/s*. Calculated speed value at the moment is displayed only on *Ebike Analyzer* android application. Alternatively, this value can be read out using *UART,* or *CAN* communication interfaces directly by the user system. The speed measurement is not essential feature to the *Tiny BMS* functions, but it is useful for keeping track of the total driven distance and estimating the remaining distance left to the empty battery. These two parameters also are calculated and stored in the *BMS* internal memory, but currently nor *Ebike Analyzer,* nor *Battery Insider* does not shows this information. This information can be read out only by the user via *UART* or *CAN* communication commands (refer to the *Tiny BMS communication protocols* documentation). How to properly configure the speed measurement feature using *Battery Insider*, refer to *Chapter 4*.

3.16. *Tiny BMS* activity indicators

Tiny BMS device is equipped with visual *LED* and audible buzzer indicators on board. *LED* indicator status depends on the *Tiny BMS* state:

- when *BMS* is in *Idle* or *Fully Charged* state – *LED* indicator blinks *1* time every *2* seconds;
- when *BMS* is in the *Fault* state – *LED* indicator blinks *3* times every *2* seconds;
- when *BMS* is in *Charging* or *Discharging* state – *LED* blinks *1* time every *500* ms;
- when *BMS* is in *Sleep* mode – the *LED* indicator is turned off.

The *Tiny BMS* on board buzzer indicator generates an audible signal when the *BMS* device is powered up or restarted (*3* times snaps). The indicator also continuously generates an audible signal by disconnecting an external current sensor (as long as the *BMS* is restarted or the external current sensor is reconnected again) and detecting a possible damaged *Load Switch*, *Charger Switch,* or *Single Port Switch* (if the switch is to be closed, but *BMS* continues to capture the flowing current).

Chapter 4: *Tiny BMS* configuration

4. Introduction

Tiny BMS device can be configured using *Battery Insider Windows* application. Alternatively all the *Tiny BMS* configuration can be done using *UART*, *MODBUS* or *CAN* communication commands (refer to *Tiny BMS Communication Protocols* documentation) from the user side integrated device or existing industrial system.
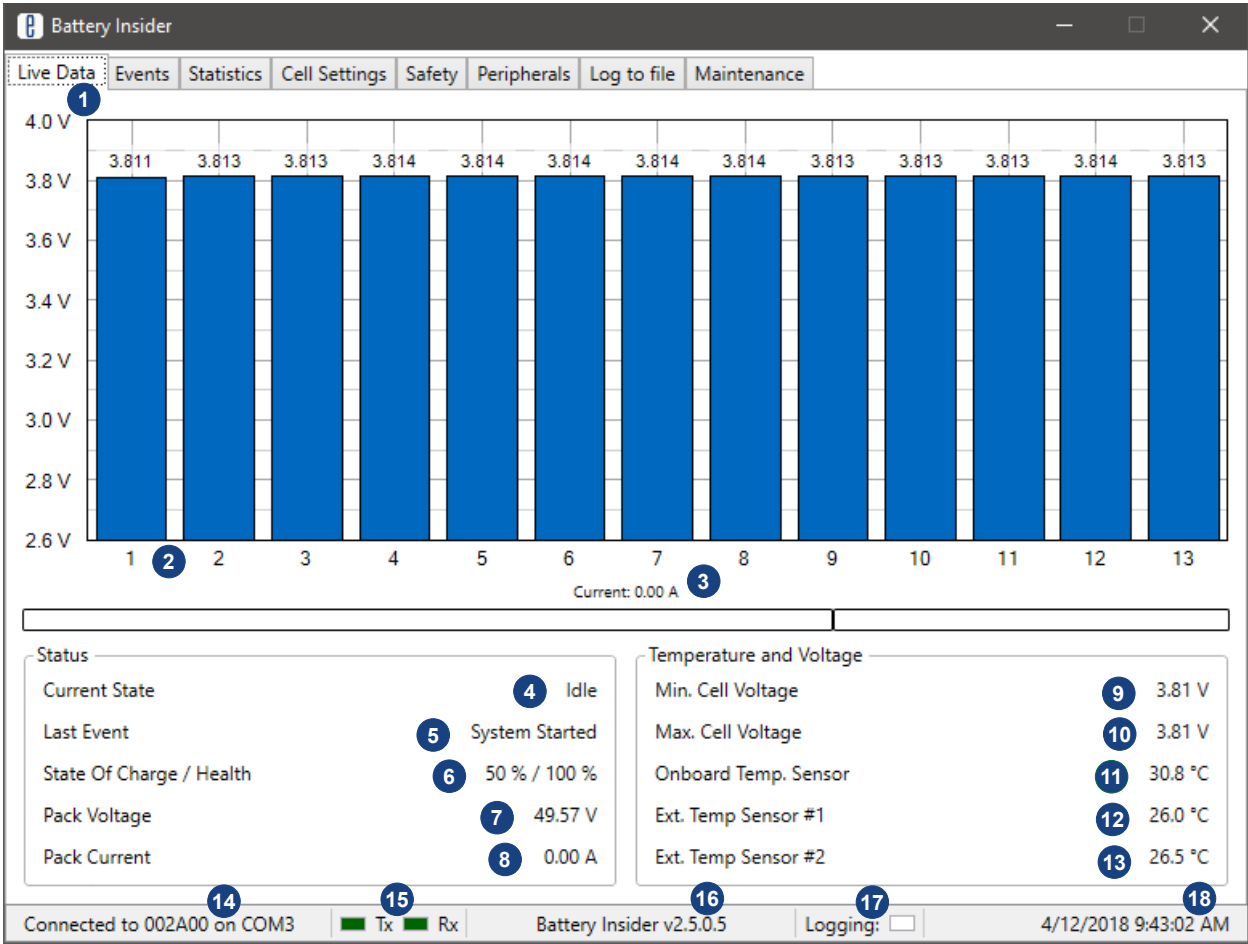
4.1. *Battery Insider* application overview

Figure 4.1: Battery Insider BMS Live Data tab screenshot

Table 4.1: Battery Insider Live Data tab information

1	*BMS Live Data* tab
2	*BMS* cells voltages graph. The number of showing cells is equal to *Number of Series Cells* **35** parameter in the *Cell Settings* **29** tab. When cell is in balancing state, the orange color bar is shown on top of the blue cell voltage column.
3	Charging / discharging current status bar. Blue color bar to the left means discharging current (negative), orange color bar to the right (positive) means charging or regeneration current.
4	*BMS* current operation state.
5	*BMS* last event record from the all *BMS Events* list **19**.

| 6	*BMS* estimated State-of-Charge (*SOC*) and State-of-Health value.
7	Battery pack voltage is calculated as voltages sum of all battery pack cells connected in series. For correct battery pack voltage calculation the correct *Number of Series Cells* 35 parameter should be set in the *Cell Settings* 29 tab.
8	Battery pack curent.
9	Minimal battery pack cell voltage.
10	Maximal battery pack cell voltage.
11	*BMS* onboard *NTC* temperature sensor temperature value.
12	External *NTC* or *Active Multipoint* temperature sensor #1 temperature value.
13	External *NTC* or *Active Multipoint* temperature sensor #2 temperature value.
14	*Battery Insider* connection to *BMS* status. Links to *Device Manager* when mouse clicked.
15	*Battery Insider* communication status on *USB-UART TX* and *RX* lines.
16	*Battery Insider* version. Links to *Maintenance* 82 tab when mouse clicked.
17	*Battery Insider BMS* data logging status. Links to *Log to file* 76 tab when mouse clicked.
18	*Battery Insider* current date time.
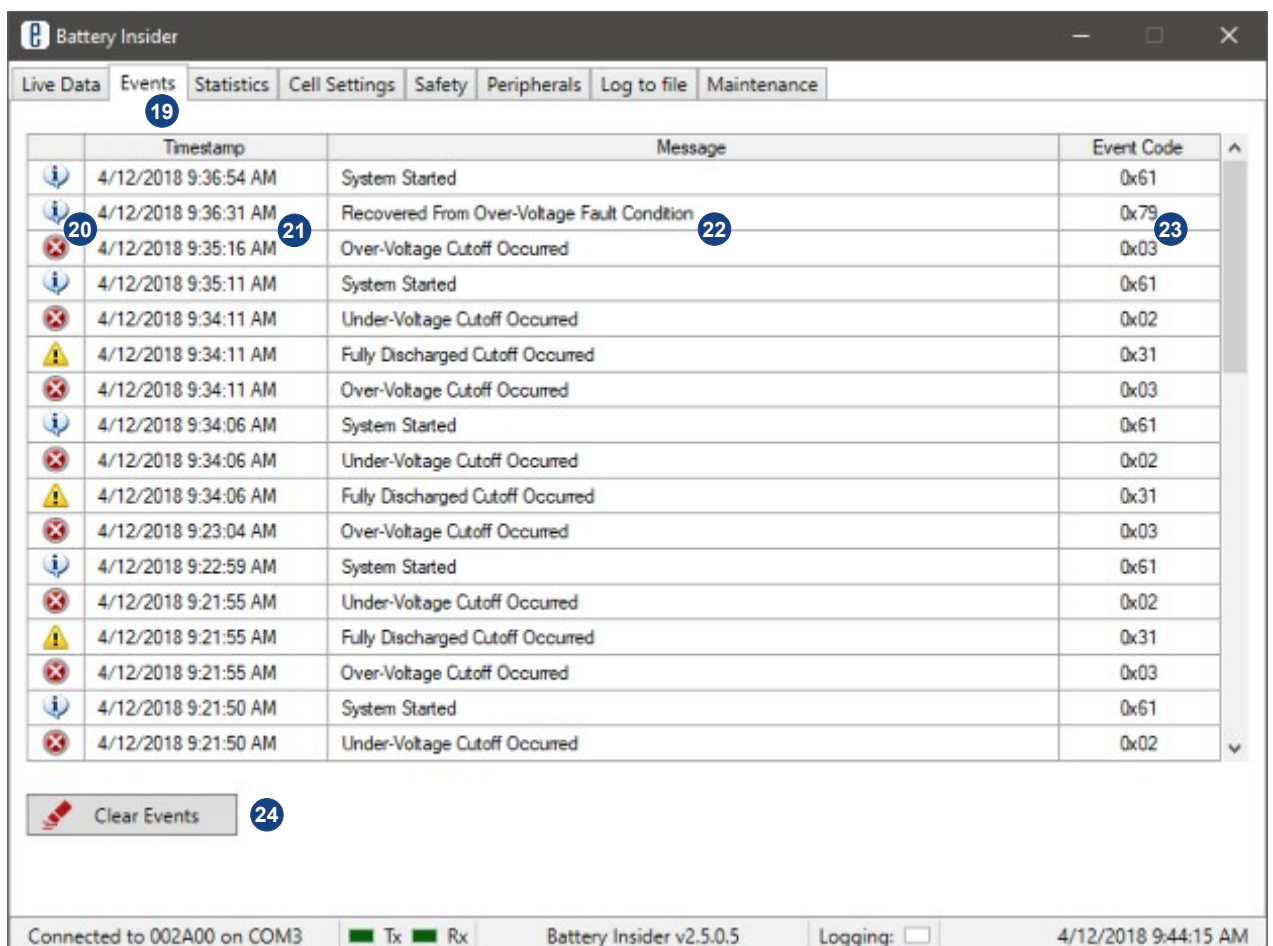
Figure 4.2: Battery Insider BMS Events tab screenshot

Table 4.2: Battery Insider Events tab information

19	*BMS Events* list tab
20	*BMS* events types:

|	
	ⓘ - *Information* event; ⚠ - *Warning* event; ❌ - *Fault* event.
21	*BMS* event timestamp.
22	*BMS* event description.
23	*BMS* event *HEX* code.
24	*BMS Clear Events* button. **Note**: *Clear Events* button only clears all events from *BMS* internal memory events list, but not recovers *BMS* from warning or fault states. *BMS* recovers from warning or fault states only when all conditions that caused *BMS* to enter this state are cleared. *BMS* recovers automatically after automatic recovery timeout (*Automatic Recovery* **50** setting in the *Safety* **41** tab) except one *Under-Voltage Cutoff Fault*. *BMS* from *Under-Voltage Cutoff Fault* state recovers only when all cells voltages are higher than *Under-Voltage Cutoff* **43** threshold and charger was connected to *BMS* or *BMS* was manually restarted **87**. When *BMS* automatic recovery function is disabled (zero value in the *Automatic Recovery* **50** setting in the *Safety* **41** tab), *BMS* recovers from warning or fault condition only when charger was connected to *BMS*, *BMS* was manually restarted or ignition signal was toggled, if *BMS* ignition feature was enabled before (*Ignition* **58** setting in the *Peripherals* **54** tab).
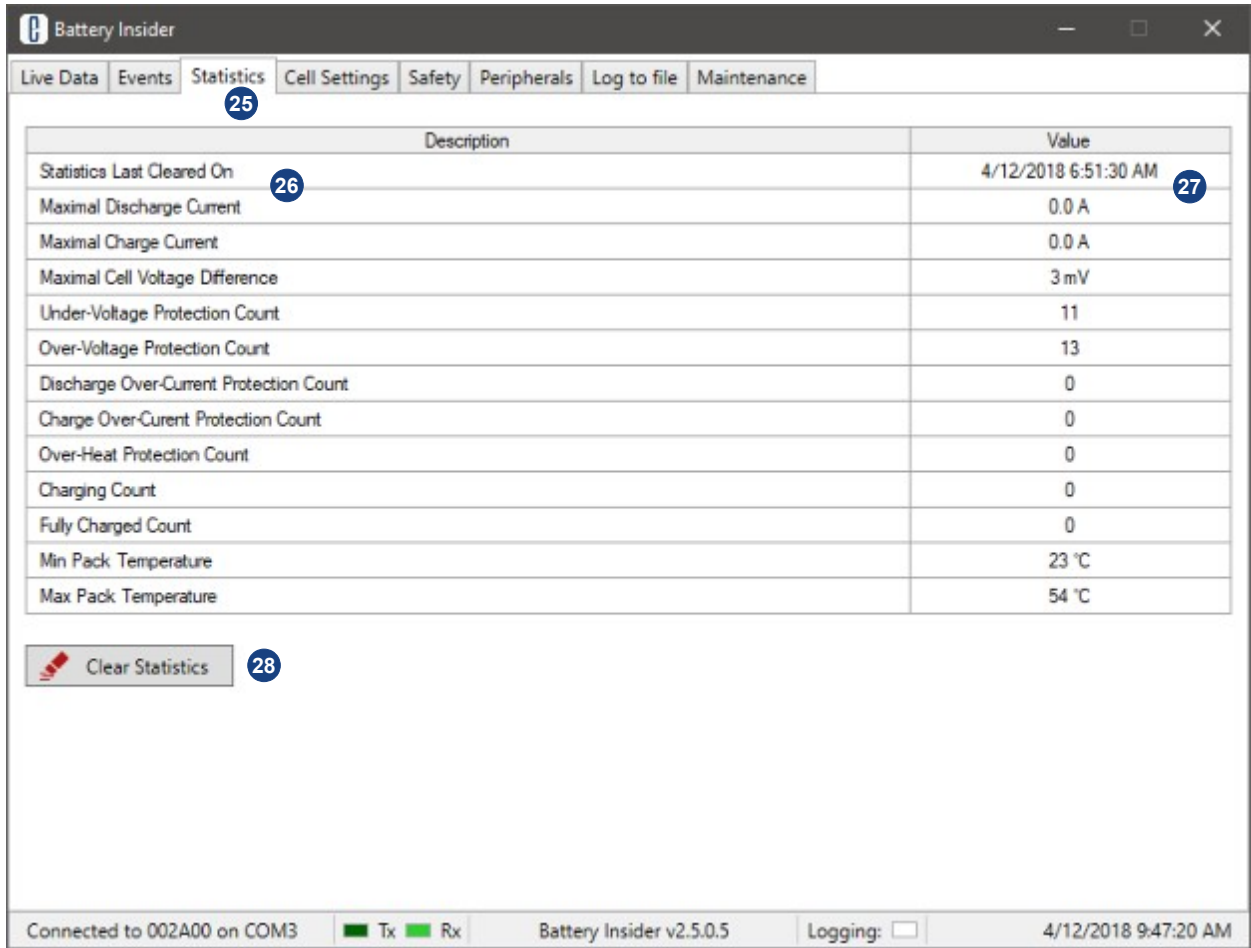
Figure 4.3: Battery Insider BMS Statistics tab screenshot

Table 4.3: Battery Insider Statistics tab information

25	*BMS Statistics* tab
26	*BMS* lifetime statistics records list.

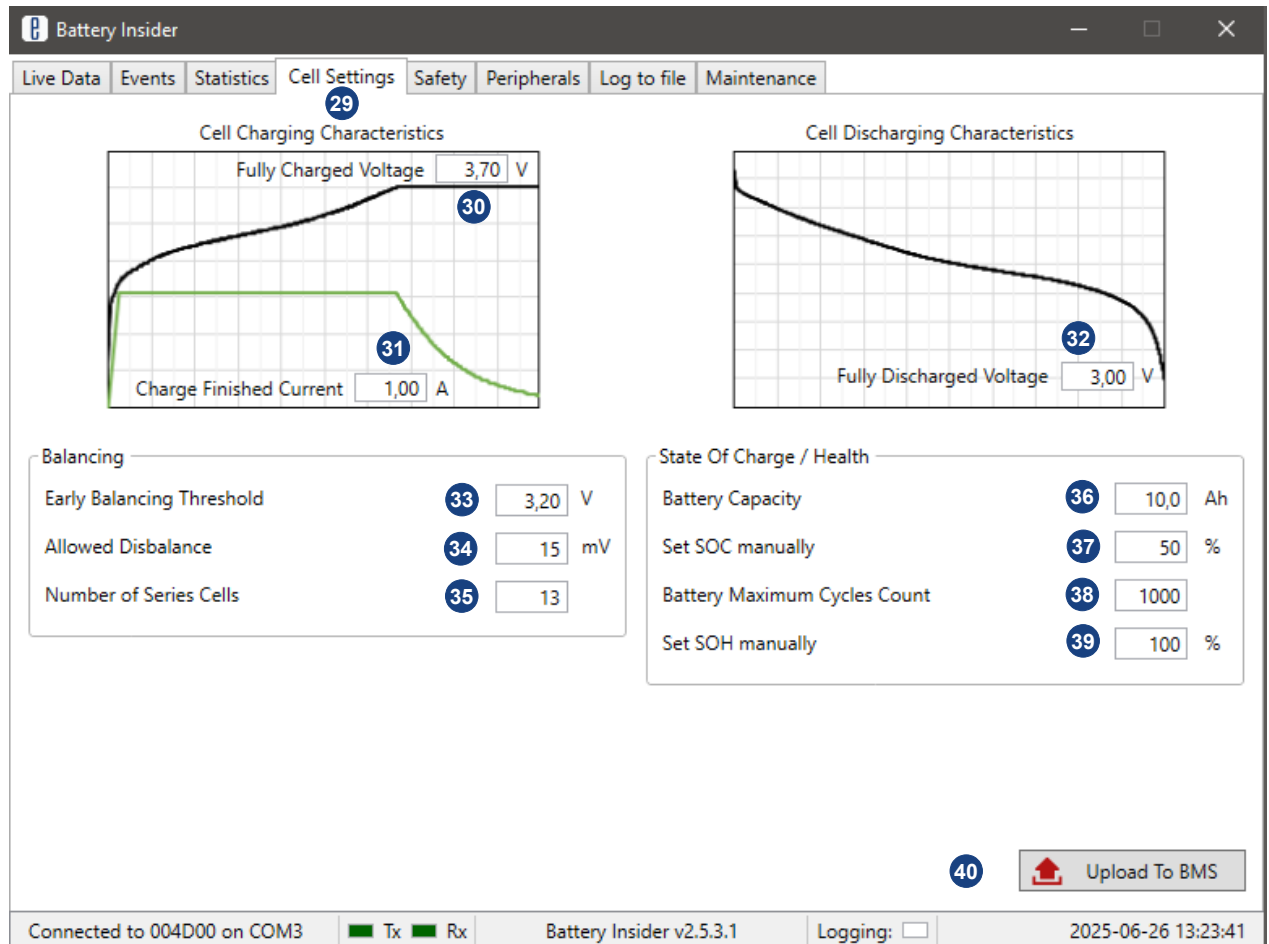|	
27	*BMS* lifetime statistics records values. Statistics records are stored in the *BMS* non-volatile internal memory and the data is not lost even if the battery has been disconnected from the *BMS* device.
28	*BMS Clear Statistics* button. *Clear Statistics* button erases all statistics records values from *BMS* internal memory.

Figure 4.4: Battery Insider BMS Cell Settings tab screenshot

Table 4.4: Battery Insider Cell Settings tab information

29	*BMS Cell Settings* tab
30	*Fully Charged Voltage*: Min. value – (*Fully Discharged Voltage* **32** + *10 mV*); Max. value – (*Over-Voltage Cutoff* **42** – *10 mV*); **Note**: *Fully Charged Voltage* must be always lower than *Over-Voltage Cutoff* and greater than *Fully Discharged Voltage*. If the *Fully Charged Voltage* needs to be set out of range, it is recommended at first set correct *Fully Discharged Voltage* **32** and *Over-Voltage Cutoff* **42** thresholds and then set *Fully Charged Voltage* value in this range.
31	*Charge Finished Current* min. and max. values are *BMS PCB* power type, used current sensor, *BMS* mode and charger switch type settings dependent.
32	*Fully Discharged Voltage*: Min. value – (*Under-Voltage Cutoff* **43** + *10 mV*); Max. value – (*Fully Charged Voltage* **30** – *10 mV*); **Note**: *Fully Discharged Voltage* must be always lower than *Fully Charged Voltage* and greater than *Under-Voltage Cutoff*. If the *Fully Discharged Voltage* needs to be set out of range it is recommended at first set correct *Fully Charged Voltage* **30** and Under-*Voltage Cutoff* **43** thresholds and then set *Fully Discharged Voltage* value in this range.

33	*Early Balancing Threshold* (voltage threshold at which cells starts balancing): Min. value – *1 V*; Max. value – *4.5 V*.
34	*Allowed Disbalance*: Min. value – *15 mV*; Max. value – *100 mV*.
35	*Number of Series Cells*: Min. value – *4 cells*; Max. value – *16 cells*.
36	*Battery Capacity*: Min. value – *0.1 Ah*; Max. value – *655 Ah*.
37	*Set SOC manually*: Min. value – *0 %*; Max. value – *100 %*.
38	Charge/discharge cycles it can handle before its capacity starts to drop significantly
39	*Set SOH manually*: Min. value – *0 %*; Max. value – *100 %*.
40	*Upload To BMS* button writes all changed settings to *BMS* internal memory. After new settings was successfully uploaded to *BMS*, *Battery Insider* reads back from *BMS* all newest settings information, min. and max. settings values.

Table 4.5: Battery Insider Safety settings tab information

41	*BMS Safety* settings tab
42	*Over-Voltage Cutoff*: Min. value – (*Fully Charged Voltage* **30** *+ 10 mV*); Max. value – *4.5 V*. **Note**: *Over-Voltage Cutofl* must be always greater than *Fully Charged Voltage*. If the *Over-Voltage Cutoff* needs to be set out of range it is recommended at first set correct *Fully Charged Voltage* **30** and then set *Over-Voltage Cutofl* value in this range.
43	*Under-Voltage Cutoff*: Min. value – *0.8 V*; Max. value – (*Fully Discharged Voltage* **32** *- 10 mV*). **Note**: Under-*Voltage Cutofl* must be always lower than *Fully Discharged Voltage*. If the Under-*Voltage Cutoff* needs to be set out of range it is recommended at first set correct *Fully Discharged Voltage* **32** and then set Under-*Voltage Cutofl* value in this range.
44	*Discharge Over-Current Cutoff* min. and max. values are *BMS PCB* power type, used current sensor, *BMS* mode and load switch type settings dependent.
45	The Discharge Over-Current Cutoff timeout defines the maximum time duration that the discharge current is allowed to remain above the configured threshold (Discharge Over-Current Cutoff) before the BMS disconnects the load.
46	The Discharge Peak-Current Cutoff defines the maximum instantaneous current that the battery is allowed to deliver during discharge, regardless of duration. If the discharge current exceeds this peak threshold even momentarily, the BMS will immediately disconnect the discharge circuit to prevent potential damage. Discharge Peak-Current Cutoff value must be higher than Discharge Over-Current Cutoff. While Discharge Over-Current Cutoff handles sustained moderate overcurrent (with an optional timeout). Discharge Peak-Current Cutoff reacts to sharp spikes without any delay.
47	*Charge Over-Current Cutoff* min. and max. values are BMS PCB power type, used current sensor, *BMS* mode and charger switch type settings dependent.

|	
㊽	*Over-Heat Cutoff*: Min. value – *20 °C*; Max. Value – *90 °C*.
㊾	*Low Temperature Charger Cutoff*: Min. value – *-40 °C*; Max. Value – *10 °C*.
㊿	*Automatic Recovery*: Min. value – *0 s* (*BMS* warning / fault state automatic recovery feature disabled); Max. value – *30 s*. **Note**: when *BMS* automatic recovery function is disabled (zero value in the *Automatic Recovery* setting), *BMS* recovers from warning or fault condition only when charger was connected to *BMS*, *BMS* was manually restarted, or ignition signal was toggled, if *BMS* ignition feature was enabled (*Ignition* ㊽ setting in the *Peripherals* ㊾ tab).
㉛	Inverted current readings value.
㉜	Disable or enable load/charger switch diagnostics.
㉝	*Upload To BMS* button writes all changed settings to *BMS* internal memory. After new settings was successfully uploaded to *BMS*, *Battery Insider* reads back from *BMS* all newest settings information, min. and max. settings values.
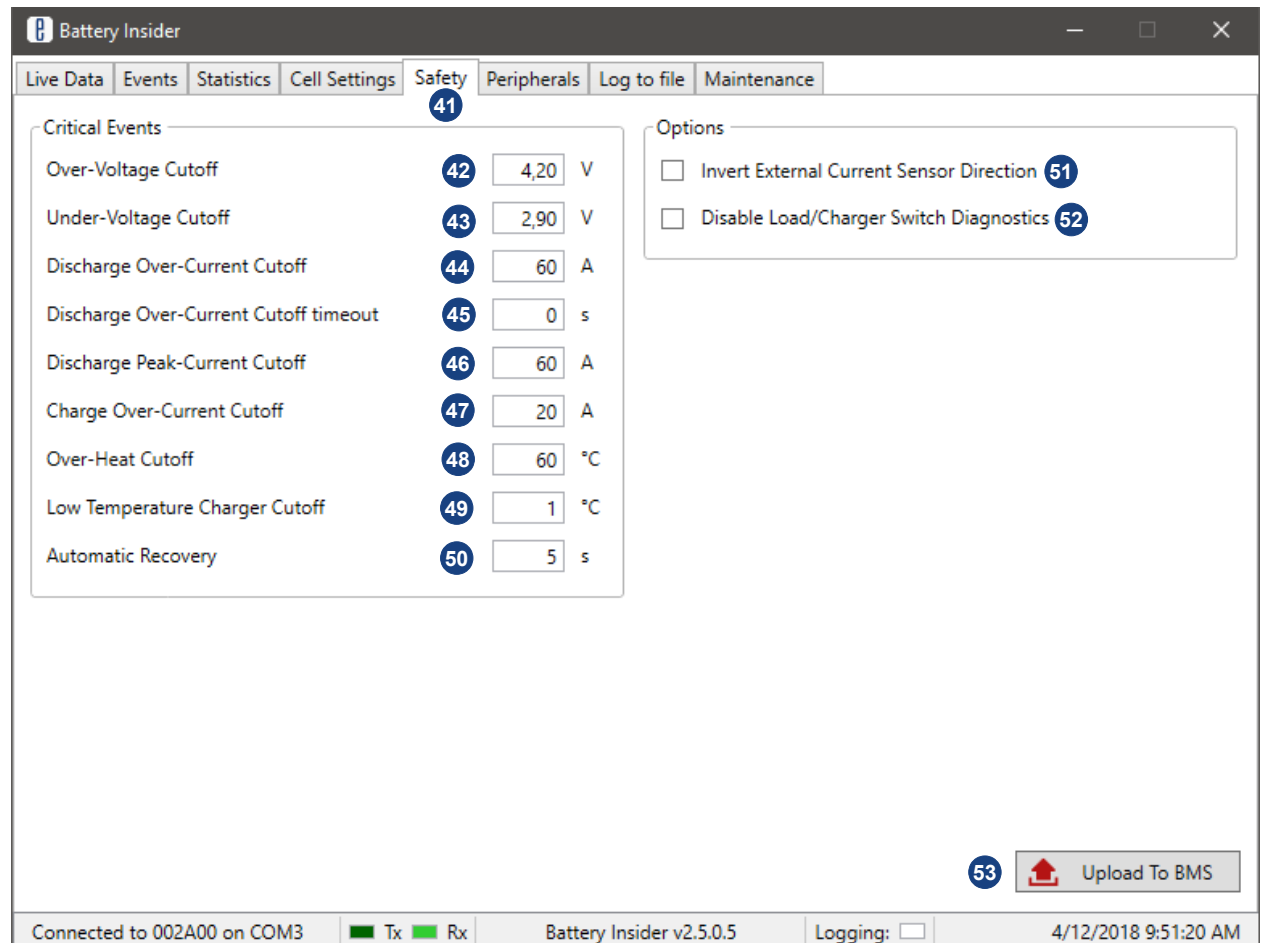
Figure 4.5: Battery Insider BMS Safety settings tab screenshot

Table 4.6: Battery Insider Peripherals settings tab information

�54	BMS *Peripherals* settings tab
�55	*BMS Operation Mode*: *Dual Port* – Separate switches for load and charger (*L-* and *C- BMS* ports);

Single Port – Same switch for load and charger (only *C- BMS* port must be used).
Note: when *BMS* mode has been changed and settings successfully uploaded to *BMS*, it automatically restarts itself and reconnects again with *Battery Insider*.

56 *Single Port Switch Type* setting are active and allowed to change only when *BMS Mode* **55** has been changed to *Single Port*):
Internal FET – Only *C- BMS* port is used (*L- BMS* port must be left unconnected). *Internal FET* cannot be selected when *Precharge* **59** feature has been enabled;
AIDOx / DIDOx / AIHOx – External output for relay / contactor control (only available on high power 150 A BMS). External relay / contactor must be connected as low side switch. Negative load / charger contact must be connected to relay / contactor together with BMS *C-* port for internal load / charger detection.
Note: External *I/O* pins *AIDOx / DIDOx / AIHOx* are active and allowed to change only when pins has not been selected in other settings.

57 *Load Switch Type* setting are active and allowed to change only when *BMS Mode* **55** has been changed to *Dual Port*):
Discharge FET – *L- BMS* port. *Discharge FET* cannot be selected when *Precharge* **59** feature has been enabled;
AIDOx / DIDOx / AIHOx – External output for load relay / contactor control (only available on high power *150 A BMS*);
Note: External *I/O* pins *AIDOx / DIDOx / AIHOx* are active and allowed to change only when pins has not been selected in other settings.

58 *Ignition* feature:
Disabled – *BMS Ignition* feature is disabled;
AIDOx – Ignition feature enabled (low power *30 A BMS*).
AIDOx / DIDOx / AIHOx – *Ignition* feature enabled (high power *150 A BMS*).
Ignition on – High logic level on selected *AIDOx / DIDOx / AIHOx* input pin!
Ignition off – Low logic level on selected *AIDOx / DIDOx / AIHOx* input pin!
Note: External *I/O* pins *AIDOx / DIDOx / AIHOx* are active and allowed to change only when pins has not been selected in other settings.

59 *Precharge* feature only available for high power *150 A BMS*:
Disabled – *BMS* load *Precharge* feature is disabled;
Discharge FET – Option is inactive and is not allowed when *Discharge FET* has been selected as *Load Switch Type* **57** or *Charge FET* has been selected as *Charger Switch Type* **70** or *Single Port* has been selected as *BMS Mode* **55**;
AIDOx / DIDOx / AIHOx – External output pin for load precharge control.
Note: External *I/O* pins *AIDOx / DIDOx / AIHOx* are active and allowed to change only when pins has not been selected in other settings.

60 *Precharge Duration*:
Possible values are *0.1 s*, *0.2 s*, *0.5 s*, *1.0 s*, *2.0 s*, *3.0 s*, *4.0 s*.

61 *Speed Sensor Input*:
Disabled – *BMS* speed calculation feature is disabled;
DIDOx – *BMS* speed calculation feature is enabled.
Note: External *I/O* pins *DIDOx* are active and allowed to change only when pins has not been selected in other settings.

62 *Distance Unit*:
Possible values are *Meters*, *Kilometers*, *Feet*, *Miles*, *Yards*.

63 *Pulses Per Unit*:
Battery Insider automatically recalculates entered value, min. and max. thresholds according to selected *Distance Unit* **62** parameter.

64 Broadcast *Protocol*:
CA V3 – Used to communicate with *Ebike Analyzer Android App;*
ASCII – Used to broadcast main *BMS* data in text format;
SOC BAR – Used to communicate with *LED SOC-BAR* indicator.

65 *Broadcast*:

Disabled – BMS data broadcast disabled;
0.1 s - 10.0 s – BMS data broadcast enabled.
Note: *Broadcast* values *0.1 s - 0.5 s* are not allowed when *SOC BAR* has been selected as broadcast *Protocol* (64).

(66)	*Charger Type*: Currently only *Generic CC/CV* charger type is supported and can be selected.
(67)	*Charger Detection* feature: *Internal – BMS* internal charger detection; *AIDOx / DIDOx / AIHOx* – External input pin for charger detection. *Charger connected* state – High logic level on selected *AIDOx / DIDOx / AIHOx* input pin! *Charger disconnected* state – Low logic level on selected *AIDOx / DIDOx / AIHOx* input pin! **Note**: External *I/O* pins *AIDOx / DIDOx / AIHOx* are active and allowed to change only when pins has not been selected in other settings.

(68)	The Charger Startup Delay defines the maximum amount of time the BMS will wait for charging current to begin after detecting a charger connection. If no charging current is detected within this period, the BMS will turn off the charge FET, effectively disconnecting the charger from the battery.
	
	Example Operation-(Charger Startup Delay = 20 sec)
	      - Charger is connected (BMS detects voltage at input).
	      - BMS enables charge FET and waits up to 20 seconds.
	      - If charging current is detected → normal charging continues.
	      - If no current flows for 20 seconds → charge FET is turned OFF.
	
	Note: You can tune this delay based on how long your charger typically takes to initialize and deliver current after connection.

(69)	The Charger Disable Delay defines the amount of time the BMS waits before disabling the charger circuit after the charger has been unplugged (i.e., charger presence signal is lost).
	When the charger is detected, the BMS enables the charge circuit.
	      - If the charger is then unplugged, the BMS does not immediately turn off the charge FET.
	        Instead, it waits for the Charger Disable Delay duration (e.g., 5 seconds).
	      - If the charger is reconnected within this time, charging resumes uninterrupted.
	      - If the charger remains disconnected after the delay expires, the BMS disables the charging
	        path.

(70)	*Charger Switch Type* setting are active and allowed to change only when *BMS Mode* (55) has been changed to *Dual Port*): *Charge FET* – C- BMS port. *Charge FET* cannot be selected when *Precharge* (59) feature has been enabled; *AIDOx / DIDOx / AIHOx* – External output for load relay / contactor control (only available on high power *150 A BMS*); **Note**: External *I/O* pins *AIDOx / DIDOx / AIHOx* are active and allowed to change only when pins has not been selected in other settings.

(71)	The Charge Restart Level defines the minimum State of Charge (SoC) at which the BMS will automatically re-enable charging after it has been disabled due to full charge conditions.This setting applies only when the charger remains continuously connected to the system.
	
	When "Enable Charger Restart Level" is Checked:
	      - Restart is based on SoC
	      - Charging is paused once the SoC reaches 100%.
	      - Charging resumes automatically only after SoC drops below the configured restart threshold,
	        e.g., 90%.
	
	When "Enable Charger Restart Level" is Not Checked-
	      - The BMS does not monitor SoC for restart.
	      - Once charging is disabled due to full charge, it waits until the battery self-discharges
	        significantly to the Fully Discharge Voltage parameter — before charging resumes.
	      - This allows the battery to fully discharge before recharging begins again.

(72)	*Charge Restart Level*:

	Min. value – *60%*; Max. Value – *95%*.
73	*Temperature Sensor Type*: *Dual 10K NTC Sensor* – One *NTC* temperature sensor on each *BMS* temperature channels can be connected. Supported *NTC* sensor 10 K @ 25 °C, *Beta* value 3977 K; *Multipoint Active Sensor* – Special *Enepaq Multipoint Active NTC* temperature sensor for max. temperature detection.
74	External Current Sensor Type.
75	*Upload To BMS* button writes all changed settings to *BMS* internal memory. After new settings was successfully uploaded to *BMS*, *Battery Insider* reads back from *BMS* all newest settings information, min. and max. settings values.

Figure 4.6: Battery Insider BMS Peripherals settings tab screenshot

Table 4.7: Battery Insider Log to file settings tab information

76	*BMS Log to file* settings tab
77	*BMS* data which will be included in the log file.
78	Logging time interval. Min. value – *1 s*; Max. value – *3600 s*.
79	Log file size: Min. value – *2 MB*; Max. value – *1024 MB*.
80	Start logging from file beginning when file size exceeds max. file size.

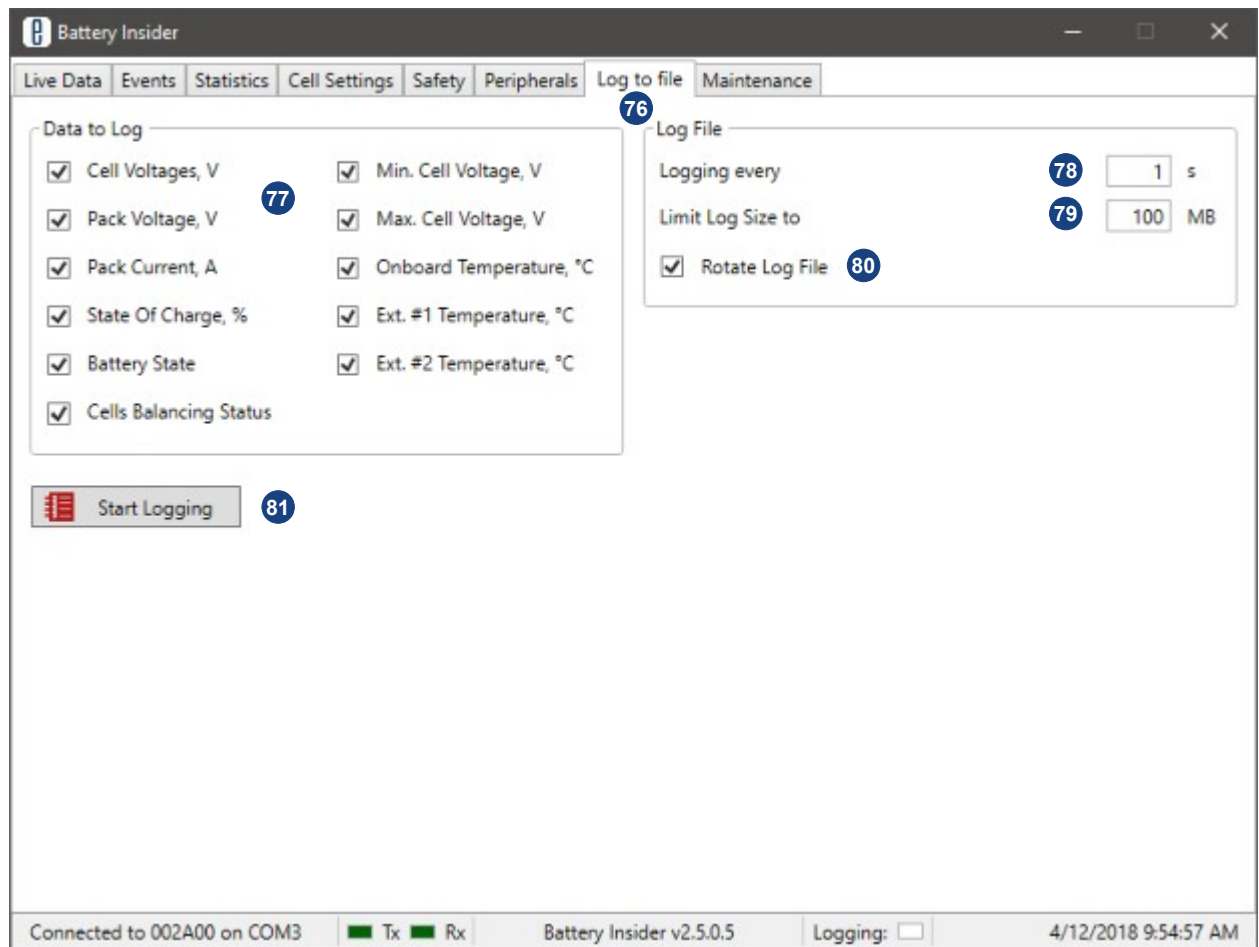| 81	Start / Stop logging.

Figure 4.7: Battery Insider Log to file settings tab screenshot

Table 4.8: Battery Insider Maintenance tab information

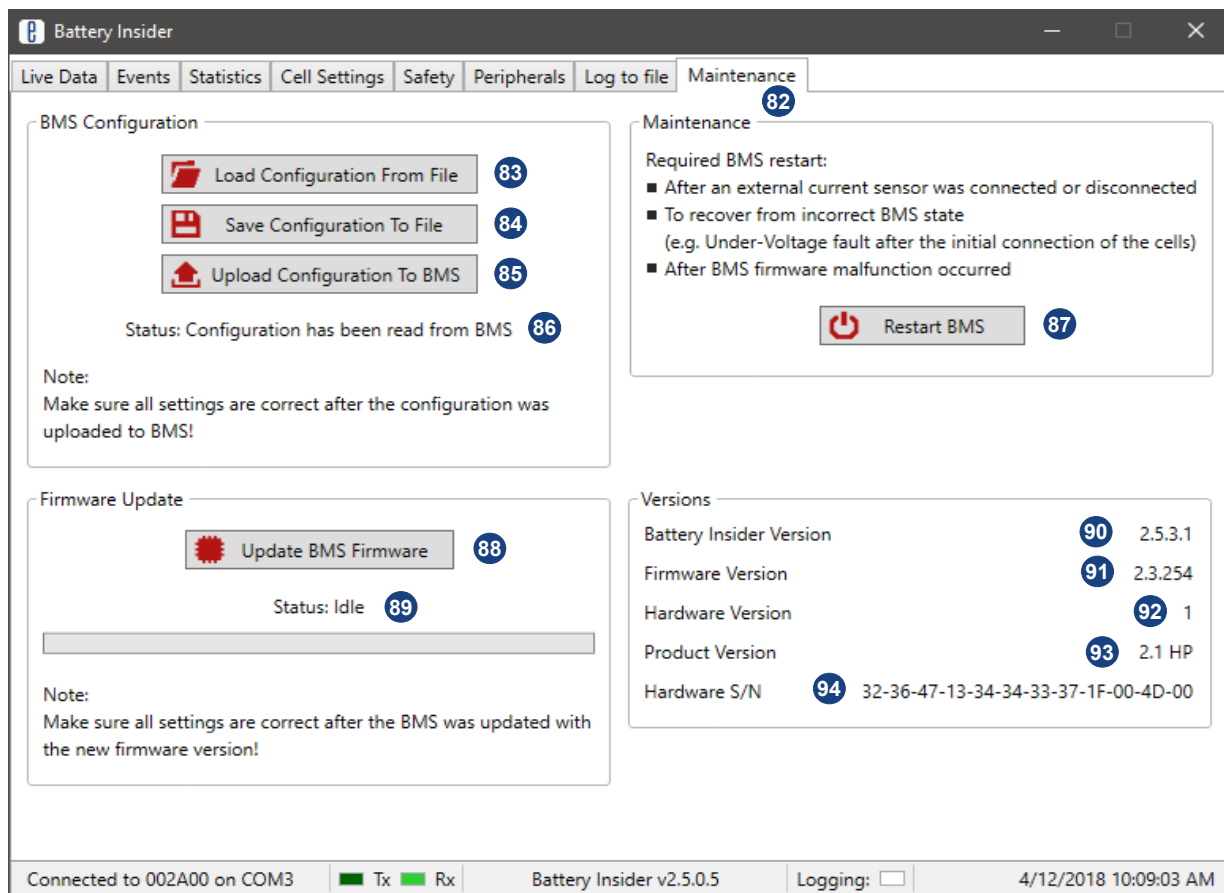| 82	*BMS Maintenance* tab
83	Load *BMS* settings from configuration file.
84	Save *BMS* settings to configuration file.
85	*Upload Configuration To BMS* button writes all settings to *BMS* internal memory. After new settings was successfully uploaded to *BMS*, *Battery Insider* reads back from *BMS* all newest settings information, min. and max. settings values.
86	*Load* / *Save* / *Upload BMS* configuration settings status.
87	*BMS* safe system restart button. *BMS* restart is required after an external current sensor was connected or disconnected, to recover *BMS* from incorrect state (e.g. *Under-Voltage Fault* after the initial connection of the cells), after *BMS* firmware malfunction occurred.
88	*BMS* firmware update button.
89	Firmware update status.
90	Current *Battery Insider* version.
91	*BMS* firmware version.
92	*BMS* hardware version.
93	BMS product version
94	*BMS* hardware serial number.

Figure 4.8: Battery Insider Maintenance tab screenshot

4.2. *Battery Insider* communication settings

The key feature of the *Battery Insider* and the *Tiny BMS* is that there is no need to configure any communication parameters on the *Battery Insider* side. The *Enepaq USB-UART* cable is based on genuine *Silicon Labs CP2102* chipset and all recent *Windows* operating system already include the drivers, therefore the cable will work straight away. However, if there is a problem, download a driver version that is confirmed as compatible from *Enepaq* website, or alternatively, driver can be downloaded directly from *Silicon Labs* support website, just make sure to get *v6.73* or later and running on at least *Windows 7 OS*. When the cable is connected, it will appear as a virtual *COM* port and the *Battery Insider* will connect to it automatically. Communication between *BMS* device and *Battery Insider* using *USB-UART* cable is always ongoing, even, if the *BMS* is in deep sleep mode, the *Battery Insider* will wake up the device. More attention should be taken, if the bluetooth connection is used between *Tiny BMS* and *Battery Insider*. First of all bluetooth module must be connected to the *Tiny BMS* and the *BMS* must be in active state (in sleep mode bluetooth module is not working due to its internal *5 V* shutdown). This means, that charger or load should be connected to wakeup the device from sleep mode, or alternatively, *Ignition* feature must be enabled before, using *USB-UART* cable. Once the *BMS* device is in active state and its bluetooth module is up and running, the *Tiny BMS* bluetooth module should be paired on the *PC* side (*Enepaq* bluetooth module pin code is always *0516*). After the *PC* and *Tiny BMS* is successfully paired, only then the *Battery Insider* application can be launched and it will automatically finds the bluetooth virtual *COM* port and connects to it.

Note: Keep in mind, that using bluetooth connection *Battery Insider* may require much more time to find the correct virtual *COM* port and connect to *Tiny BMS* device. Bescides, to update BMS firmware using

Bluetooth communication is not possible at the moment.

When the *Battery Insider* successfully connects to the *Tiny BMS* device, the connection status is displayed in the lower left corner of the *Battery Insider* application window (Figure *4.9*).

Connected to 002A00 on COM3   ■ Tx ■ Rx

Figure 4.9: Battery Insider connection status

4.3. *Battery Insider* cell settings

The correct order of setting the battery and cells parameters would be at first adjust the settings under *Balancing* and *State of Charge* group boxes in the *Cell Settings* tab of the *Battery Insider* (Figure *4.10*). Upload the settings to the *Tiny BMS*. Then on *Safety* tab adjust the cells critical parameters *Over-Voltage Cutoff* and *Under-Voltage Cutoff*. Upload the settings again to the device. Only then go back to *Cell Settings* tab and adjust *Fully Charged Voltage* and *Fully Discharged Voltage* parameters acordinglly. These settings has a relationship *Under-Voltage Cutoff < Fully Discharged Voltage < Fully Charged Voltage < Over-Voltage Cutoff* and the *Battery Insider* will not allow the user to set these values in any other way which would contradict that parameters relationship.

Balancing
Early Balancing Threshold   3.20 V
Allowed Disbalance   15   mV
Number of Series Cells   13

Step 1

State of Charge
Battery Capacity   10.0 Ah
Set SOC manually   50   %

Step 2 ⬆ Upload To BMS

Critical Events
Over-Voltage Cutoff   4.20 V
                Step 3
Under-Voltage Cutoff   2.90 V

Step 4 ⬆ Upload To BMS

Cell Charging Characteristics
Fully Charged Voltage   4.00 V

Step 5

Charge Finished Current 1.00 A

Cell Discharging Characteristics

Step 6

Fully Discharged Voltage 3.00 V

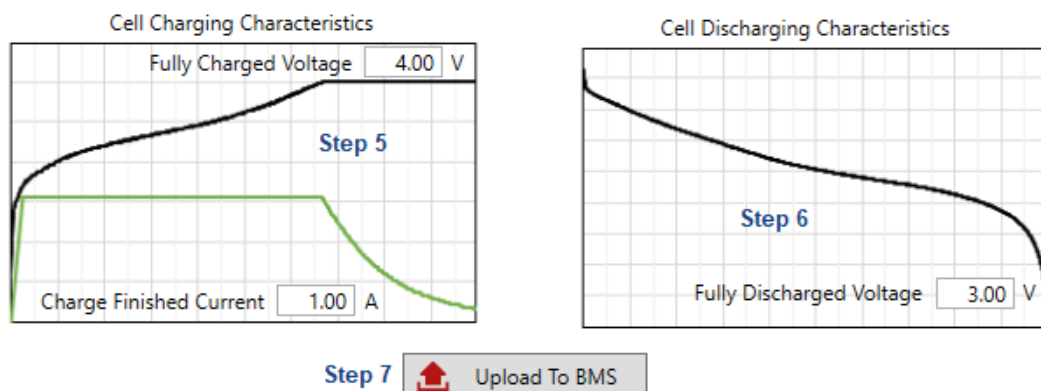**Step 7** ⬆ Upload To BMS

Figure 4.10: Battery Insider cell settings configuration sequence

4.4. *Battery Insider* peripherals and safety settings

Some *Tiny BMS* safety parameters such as all over-current settings depends on configured peripherals and current sensor used: *BMS* operation mode, *Load Switch Type, Charger Switch or Single Port Switch Type* parameters. These main parameters are located under *Mode*, *Load* and *Charger* group boxes in the *Peripherals* tab of the *Battery Insider*. When *Tiny BMS* is used in the high current applications where

relays / contactors should be used and it is needed to set over-current thresholds at a much higher level, it can be done only by setting *Load Switch / Charger Switch* as external *I/O* pins and properly connecting the external current sensor at first. Only then *Battery Insider* allows the user to set higher overcurrent limits. The recommended way to adjust *Tiny BMS* peripherals and safety settings is shown in Figure *4.11*. How to properly connect and configure *Tiny BMS* external current sensor refer to the next chapter.

Note: When *BMS* mode has been changed and settings successfully uploaded to *BMS*, it automatically restarts itself and automatically reconnects again with the *Battery Insider*.

Figure 4.11: Battery Insider peripheral and safety settings configuration sequence

4.5. *Tiny BMS LEM* external current sensor configuration

External current sensor can be connected only to the *150 A* high power *Tiny BMS* hardware version. Current sensor is the essential component of the battery application, therefore due to the safety reasons there is an important sequence how to connect and use *LEM* external current sensor.

4.5.1. External current sensor connection sequence:

After *Tiny BMS* power up or restart, if the external current sensor is not connected yet, an internal *BMS HALL* current sensor is used. Allowed battery charging and discharging as usual. After the external current sensor was connected, the *BMS* generates event on connected sensor *External Current Sensor Connected (BMS restart required) (0x0F)*. After this event, the *BMS* needs to be restarted (Figure *4.12*). After restarting, an external current sensor will be used to measure current. If, after connecting an external current sensor, it is disconnected again and *BMS* is not restarted, *BMS* generates event *External Current*

Sensor Disconnected (0x7D). In this case, the internal *HALL* current sensor is used as before and no *BMS* restart action is required.

	Timestamp	Message	Event Code
❌	7/19/2018 4:14:23 PM	External Current Sensor Connected (BMS restart required)	0x0F
ⓘ	7/19/2018 4:14:11 PM	System Started **Step 1**	0x61

Maintenance

Required BMS restart:
- After an external current sensor was connected or disconnected
- To recover from incorrect BMS state
  (e.g. Under-Voltage fault after the initial connection of the cells)
- After BMS firmware malfunction occurred

Step 2 ⏻ Restart BMS

Figure 4.12: External current sensor connection sequence

4.5.2.   External current sensor disconnection sequence:

After external current sensor was disconnected from the *BMS* device, it generates the event *External current sensor Disconnected (BMS restart required) (0x0E)*. The *BMS* switches to the fault state, the battery charging and discharging is restricted (*Load Switch*, *Charger Switch,* or *Single Port Switch* is turned off). *BMS* generates an audible signal to alert the user about the lost current sensor connection.To confirm that external current sensor is unused and needs to be disconnected the user must restart the *BMS* (Figure *4.13*). After the *BMS* is restarted, the internal *HALL* current sensor is used again to measure the current. When the external current sensor is disconnected, but the *BMS* is not restarted and the external current sensor is reconnected again, the *BMS* automatically goes out of the fault state, the event *External Current Sensor Connected (0x7C)* is generated and the current measurement is continued using an external current sensor.

	Timestamp	Message	Event Code
❌	7/19/2018 4:28:14 PM	External Current Sensor Disconnected (BMS restart required)	0x0E
ⓘ	7/19/2018 4:27:59 PM	System Started **Step 1**	0x61

Maintenance

Required BMS restart:
- After an external current sensor was connected or disconnected
- To recover from incorrect BMS state
  (e.g. Under-Voltage fault after the initial connection of the cells)
- After BMS firmware malfunction occurred

Step 2 ⏻ Restart BMS

Figure 4.13: External current sensor disconnection sequence

4.6. *Tiny BMS* configuration to use with *Ebike Analyzer* android application

To use *Tiny BMS* with *Ebike Analyzer* android application *Enepaq* bluetooth adapter is needed. *BMS* device must stay in the active mode to establish connection to *Ebike Analyzer* application. For that purpose it is highly recommend to use the *Tiny BMS Ignition* feature, otherwise the charger or load should be connected to the *BMS* to keep the device in active state. Therefore, at first time *Tiny BMS* must be configured using *USB-UART* cable and the *Battery Inside*r windows application and then *USB-UART* cable replaced by the bluetooth module. In the *Battery Insider* the *Ignition* feature, *CA V3* broadcast protocol and duration, and *Speed Sensor Input* feature (*Ebike Analyzer* can show the vehicle speed value) should be enabled and

configured properly (Figure *4.14*). On the android device side, before launching *Ebike Analyzer* application for the first time, the android device must be properly paired with the *Tiny BMS* (*Enepaq* bluetooth module pin code is always *0516*). Once, pairing completed successfully and bluetooth connection is established, after launching the *Ebike Analyzer* application the connection status on the top left of the screen should be *Connected* and all battery parameters should be displayed on the dashboard. Next time the *Ebike Analyzer* app can be launched directly skipping the pairing process.

Figure 4.14: Tiny BMS configuration to use with Ebike Analyzer application

4.7. *Tiny BMS* configuration to use with *SOC-BAR* indicator

To use the *Tiny BMS* device with the *Enepaq SOC-BAR LED* indicator it is required to configure *SOC BAR* broadcast protocol and the broadcast duration to *1* second or a higher value (Figure *4.15*). *SOC-BAR LED* indicator is working only when *BMS* is in active state, otherwise indicator is kept shutdown. *SOC-BAR LED* indicator represents the *Tiny BMS* estimated battery pack *State-Of-Charge* level. *SOC-BAR* indicator scale: one line corresponds to *10 %* of *SOC* vale, but some rows are hardware level grouped by two, therefore when *SOC* level falls below *90 %* - the top row goes out, when it falls below *70 %* - 3 rows goes out, when it falls below *50 %* - 5 rows goes out, when it falls below *40 %* - 6 rows goes out, *30 %* - 7 rows goes out, below *20 %* - 8 rows goes out, below *10 %* - the bottom two rows starts to blink. When *SOC* level drops to *0 %* level, the *SOC-BAR* indicator shuts down.

Figure 4.15: Tiny BMS SOC-BAR indicator configuration

4.8. *Tiny BMS* temperature sensor configuration

When *10 K @ 25 °C* (*Beta* value *3977 K*) *NTC* thermistors are used, the *Temperature Sensor Type* parameter should be configured to *Dual 10K NTC Sensor*. Otherwise, *Multipont Active Sensor* value should be configured when *Enepaq* multipoint sensors available in the *Enepaq Cell Modules* are used.
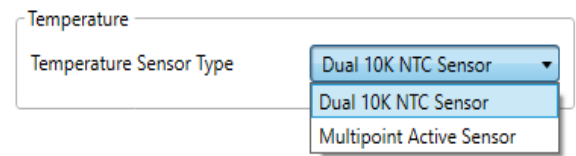
Figure 4.16: Tiny BMS temperature sensor configuration

Chapter 5: *Tiny BMS* maintenance

5. Introduction

This section contains information and step-by-step guides on how to perform certain *Tiny BMS* maintenance operations. These operations cover battery pack data logging, firmware update, *BMS* restart, etc.

5.1. *Battery Insider* live *BMS* data logging

Battery Insider can log basic *Tiny BMS* and battery live data values to a file, which is useful to monitor and analyze the battery processes like charging and discharging and determine battery behavior in different conditions. Also it is very useful to determine some problems of the *BMS* or the battery application itself, when the system malfunction or the unforeseen behavior has been occur. Unfortunately, due to the lack of the *Tiny BMS* internal memory, the data logging is only available until *Tiny BMS* is directly connected to the *Battery Insider* application through *USB-UART* or bluetooth interface. The user can choose which *BMS* and battery data to log. The log file is created in the same directory from which the Battery Insider application was launched. How to control *Battery Insider* logging feature is shown in Figure *5.1*.
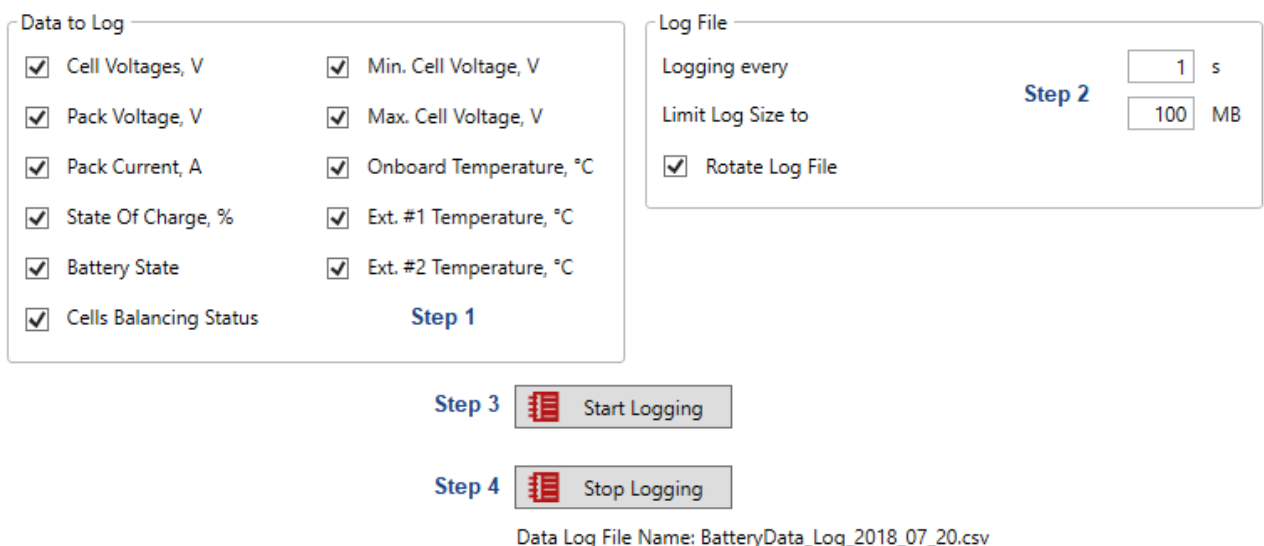
Figure 5.1: Battery Insider data logging control

5.2. *Tiny BMS* device restarting

Tiny BMS device can be restarted in a safe way during operation when the system malfunction has been occurred using the *Battery Insider* application or *Enepaq* proprietary communication commands via *UART*,

bluetooth or *CAN* bus interfaces. Take a note that the *CAN* bus and bluetooth connection does not work when the *BMS* is in sleep mode, so *BMS* device restarting with CAN bus or bluetooth connection is possible only with the device in active mode. Before restarting process *Tiny BMS* device turns off *Load* and *Charger Switches* (*Dual Port* mode), or *Single Switch* (*Single Port* mode), saves all settings, records and events and after the restart operation is completed all modules and peripherals are reinitialized again. *Tiny BMS* also recalibrates the current sensor after restarting. Besides, there are several cases where the restart of the *Tiny BMS* device is highly recommended or even desirable. For example, to recover from incorrect *BMS* state, which occurs after the initial connection of the cells or after an external current sensor was connected or disconnected. The restart button is located in the *Maintenance* tab of the *Battery Insider* (Figure *5.2*). How to restart the *Tiny BMS* device using *Enepaq* communication commands refer to *Tiny BMS communication protocols* documentation.
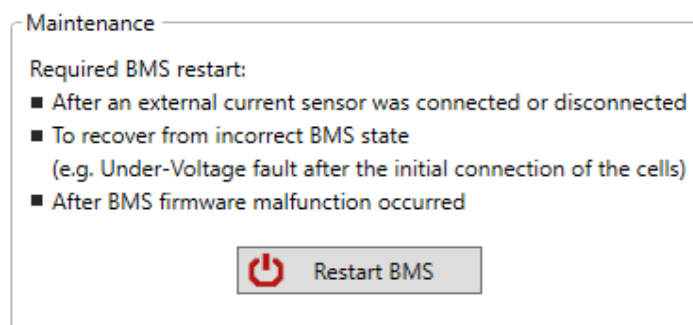

Figure 5.2: Tiny BMS restarting feature

5.3. *Tiny BMS* importing and exporting settings

Both export and import *Tiny BMS* settings options can be reached from *Battery Insider Maintenance* tab (Figure *5.3*). Special care must be taken, when loading settings from file, especially when different *Tiny BMS* power versions or different *150 A* high power *Tiny BMS* configurations are used, because some of the parameters are dependent on different control configurations (internal *FETs* or external relays / contactors) and current sensor used. It is highly recommended to double check the settings, loaded from file, before directly uploading these settings to *Tiny BMS* device.
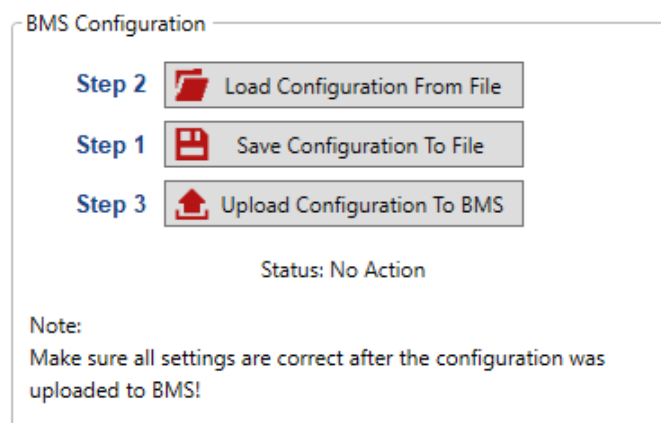

Figure 5.3: Battery Insider importing and exporting Tiny BMS settings

5.4. Tiny BMS firmware update

In order to ensure that *Tiny BMS* would have all the latest features and bug-fixes, it is recommended to periodically check the *Tiny BMS* firmware release notes online at *https://enepaq.com* and perform firmware update procedure. The firmware image can be downloaded upon request. However, firmware update procedure also can be revert back to an older firmware version if necessary. Officially, for use with the *v2.5.3.0 Battery Insider* version, described in this document, and later versions, the *v2.3.254* or higher *Tiny BMS* firmware versions are recommended. *Tiny BMS* firmware image is an *Enepaq* proprietary encrypted binary file with the extension *.bms*. Normally, in most cases during firmware update process all the *Tiny BMS* settings are saved in the internal non-volatile memory and after the update process is completed successfully, all settings are restored. When the firmware update process fails or the *Tiny BMS* is updated to the out of date or unsupported firmware version, the settings are set to factory defaults values. Therefore, it is highly recommended to save all the settings to the file, that tey can be restored in case of an error. If the firmware update process fails to complete successfully, *Tiny BMS* device enters the bootloader mode and the update process can be restarted again. In this case it is recommended to close the *Battery Insider* application and launch it again to be able to detect *Tiny BMS* device trapped in the bootloader mode (Figure *5.5*).

> **Note:** Before selecting desired *Tiny BMS* firmware file in the opened *Update BMS Firmware* dialog of the *Battery Insider* make sure, that firmware file is unzipped and the extension is *.bms* (Figure *5.4*).

> **Note:** It is unsafe to leave the *Tiny BMS* device trapped in the bootloader mode for a longer period of time connected to the battery, because it can drain the battery.

> **Note:** The same *Battery Insider* application can be used to update the *Enepaq UART-CAN* converter firmware. Just connect *UART-CAN* module to a *PC* using *Enepaq USB-UART* cable, wait for connection with *Battery Insider*. After *Battery Insider* detects connected module, the update process can be started. Special care must be taken in order to avoid tthe situation when *UART-CAN* module and *Tiny BMS* device firmware versions are swapped. In this case, the *Tiny BMS* device or *UART-CAN* converter can be damaged permanently.
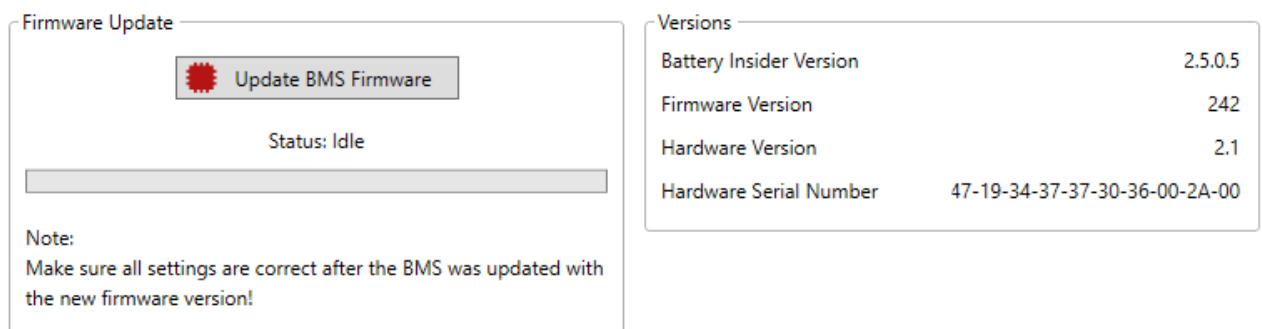

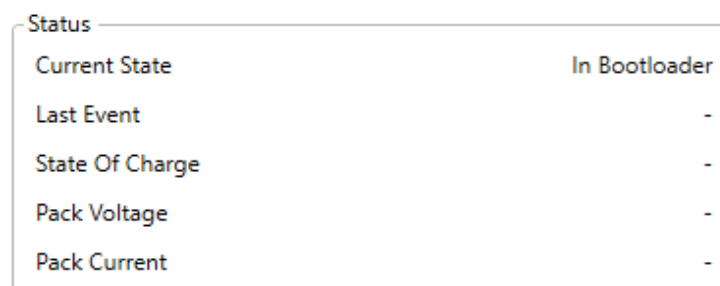*Figure 5.4: Battery Insider firmware update dialog*

Figure 5.5: Tiny BMS trapped in the bootloader after the failed update process

Chapter 6: *Tiny BMS* troubleshooting

6. Introduction

This section contains the frequently asked questions about the detected *Tiny BMS* problems and covers some possible cause of the problem giving a hint about the solution to the problem. *Tiny BMS* device malfunction and possible causes are listed in *Table 6.1*.

Table 6.1: Most frequently detected Tiny BMS problems and possible causes

Tiny BMS malfunction	Possible cause
Tiny BMS device does not start up after the power (*B+* and *B-* solderpads) was applied (no buzzer or *LED* activity indication).	• *Tiny BMS* is trapped in the bootloader mode after firmware update process failed. Restart firmware update process; • *Tiny BMS* is damaged. Contact the distributor for repair.
Tiny BMS device started up successfully, but there is no LED activity indication after some time.	• *Tiny BMS* is in sleep mode. To wake up from sleep mode *USB-UART* communication cable or the charger needs to be connected.
Tiny BMS buzzer emits an audible signal.	• *Tiny BMS* is restarted; • External current sensor was disconnected; • Possible load and / or charger switch damage. The swich should be turned off, but the *BMS* device detects the current flowing.
There is no communication between *Tiny BMS* device and *Battery Insider* application (*USB-UART* cable).	• Unsupported *USB-UART* cable drivers (download supported version of driver from *Enepaq* website); • Virtual *COM* port of the *USB-UART* cable is used by the other application; • *USB-UART* cable is damaged (most likely ground loop problem) . Contact the distributor for repair; • *Tiny BMS* is damaged (most likely ground loop problem). Contact the distributor for repair.
There is no communication between *Tiny BMS* device and *Battery Insider* application (bluetooth module).	• *Tiny BMS* bluetooth module is not paired successfully on the *PC* side; • Battery Insider cannot find the virtual *COM* port of the *PC* bluetooth adapter (need to wait longer it requires much more time to scan); • *Enepaq* bluetooth module is not connected to *Tiny BMS* properly; • *Tiny BMS* device is in sleep mode (no *LED* activity indication).
There is no communication between *Tiny BMS* device and *Ebike Analyzer* android application.	• *Tiny BMS* bluetooth module is not paired successfully on the android device side; • *Enepaq* bluetooth module is not connected to *Tiny BMS* properly; • *Tiny BMS* device is in sleep mode (no *LED* activity indication); • *Tiny BMS Broadcast* protocol was not configured to *CA V3*, and / or *Broadcast* time duration is *Disabled*.
There is no communication between *Tiny BMS* device (*Enepaq UART-CAN* converter) and the user side *CAN* controller.	• *Enepaq UART-CAN* converter is not connected to *Tiny BMS* properly; • *CAN* bus termination is not connected to *UART-CAN* converter or user side *CAN* controller; • Wrong *CAN* bus baudrate (only supported *CAN* baudrate

	is *500 kbit/s*).
	• Wrong *UART-CAN* node id (default node id is *0x01*);
	• *Tiny BMS* device is in sleep mode (no *LED* activity indication);
	• *UART-CAN* converter is damaged. Contact the distributor for repair.
The *Tiny BMS SOC-BAR LED* indicator is always switched off.	• *Tiny BMS Broadcast* protocol was not configured to *SOC BAR*, and / or *Broadcast* time duration is *Disabled;*
	• *Tiny BMS* estimated *SOC* value is 0%;
	• *Tiny BMS* device is in sleep mode (no *LED* activity indication).
One or more cells voltage measured by *BMS* is considerably lower than the real cell voltage.	• Bad *Tiny BMS* cells wiring connection;
	• The lowest battery pack cells negative terminal is not connected to *Tiny BMS* cells connector.
	• *Tiny BMS* device *AFE* chip is damaged (most likely due to wrong connection seuence). Contact the distributor for repair.
The *Tiny BMS* load output is disconnected (*Load Switch* is turned off).	• *Tiny BMS* is in the *Warning* or *Fault* state;
	• The *Ignition* feature was enabled, and there is a low logic level on the configured ignition input pin.
The charger is connected, but the charging was not started (internal charger detection).	• The charging current does not start to flow within *20* seconds time interval (most likely the issue with smart charger, which starts charging after the battery is detected);
	• The charger output voltage too low;
	• The charging current too low;
	• *Tiny BMS* firmware malfunction (*Tiny BMS* restart should solve the problem);
	• *Tiny BMS* device is damaged. Contact the distributor for repair.
The charger is connected, but the charging was not started (external charger detection).	• No high logic level is applied to the configured external charger detection input pin;
	• The charger output voltage too low;
	• *Tiny BMS* firmware malfunction (*Tiny BMS* restart should solve the problem);
	• *Tiny BMS* device is damaged. Contact the distributor for repair.
The battery never fully charges.	• The charger output voltage too high;
	• The charger output voltage too low;
	• The *Tiny BMS Charge Finished Current* threshold too low;
	• Cells imbalance is bigger than *Tiny BMS Allowed Disbalance* parameter (*Tiny BMS* stops charging and enters only cells balancing stage).
The *Tiny BMS* does not wake up from sleep mode even the charger is connected.	• *Under-Voltage Cutoff Fault* occured. At least one cell voltage stays lower than *Under-Voltage Cutoff* threshold;
	• The charging current does not start to flow within *20* seconds time interval and the fault cannot recover (most likely the issue with smart charger, which starts charging only after the battery is detected).
The *Tiny BMS* does not wake up from sleep mode even the high logic level voltage was applied to configured *Ignition* input pin.	• *Under-Voltage Cutoff Fault* occured;
	• The *Ignition* feature was not configured correctly.
Fully Charged Voltage threshold is not allowed to be adjusted.	• Entered value is off the allowed range. *Fully Discharged Voltage* and *Over-Voltage Cutoff* parameters must be configured at first to match the relationship *Fully Discharged Voltage < Fully Charged Voltage < Over-*

	Voltage Cutoff.
Fully Discharged Voltage threshold is not allowed to be adjusted.	• Entered value is off the allowed range. *Fully Charged Voltage* and *Under-Voltage Cutoff* parameters must be configured at first to match the relationship *Under-Voltage Cutoff < Fully Discharged Voltage < Fully Charged Voltage*.
Under-Voltage Cutoff threshold is not allowed to be adjusted.	• Entered value is off the allowed range. *Fully Discharged Voltage* parameter must be configured at first to match the relationship *Under-Voltage Cutoff < Fully Discharged Voltage*.
Over-Voltage Cutoff threshold is not allowed to be adjusted.	• Entered value is off the allowed range. *Fully Charged Voltage* parameter must be configured at first to match the relationship *Fully Charged Voltage < Over-Voltage Cutoff*.
Discharge Over-Current Cutoff threshold is not allowed to be adjusted to a higher value than 3*0 A*.	• *BMS* mode is configured as *Single Port* mode and *Single Port Switch* is configured as *Internal FET*.
Discharge Over-Current Cutoff threshold is not allowed to be adjusted to a higher value than *60 A*.	• *30 A* low power *Tiny BMS* version is used; • *150 A* high power *Tiny BMS* version is used, *BMS* mode is configured as *Single Port* mode and *Single Port Switch* is configured as *Internal FET*.
Discharge Over-Current Cutoff threshold is not allowed to be adjusted to a higher value than 150 *A*.	• *30 A* low power *Tiny BMS* version is used; • *Load Switch Type* or *Single Port Switch Type* is configured as *internal FET* and external current sensor was not connected and used properly.
Charge Over-Current Cutoff threshold is not allowed to be adjusted to higher value than 30 *A*.	• *30 A* low power *Tiny BMS* version is used; • *Charger Switch Type* or *Single Port Switch Type* is configured as *internal FET* and external current sensor was not connected and used properly.
Tiny *BMS* does not enter the sleep mode.	• Ongoing *UART*, bluetooth or *CAN* bus communication; • Charger is connected; • Ongoing cells balancing process; • *Ignition* is enabled and a high logic level voltage is applied to the configured *Ignition* pin; • Charging or discharging current flowing (*Ignition* feature disabled)
Excessive *Tiny BMS* energy consumption in sleep mode (*AFE* chip heats up).	• *Tiny BMS* is damaged. Contact the distributor for repair;
Tiny BMS device always detects charging / discharging current flowing.	• *Tiny BMS* firmware malfunction (*Tiny BMS* restart should solve the problem); • *Tiny BMS* is damaged. Contact the distributor for repair.
Tiny BMS always measures negative temperature with *Enepaq Multipoint active sensor.*	• Pull up rezistor between *Tiny BMS* temperature input pin and *5 V* output pin is not connected; • Bad connection between *Enepaq* battery modules temperature sensors and *Tiny BMS;* • *Enepaq* battery module is damaged. Contact the distributor for repair.

Note: For product warranty repair please contact your distributor.

Note: If you did not find the solution in the table above, please contact your distributor.

Document revision history

Revision	Date	Description
A	2018-07-30	Initial release.
B	2020-09-14	Fixed external current sensor pinout marking.
C	2022-03-24	Company rebranded to Enepaq.
D	2025-07-01	New features description. New v2.2 version discription. Fixed documentation issue