



# Battery Management System

Document version 3.1, Jan 2024

© 2018-2024 Dilithium Design

## Contents

Overview.....	5
Battery Management .....	5
Integrated Charge Control .....	6
BMS Architecture.....	7
Cell Wiring and Cell Groups.....	8
Pack and Cell Numbering.....	10
Example 1: 48 Cell Pack.....	11
Example 2: Two Parallel 24 Cell Packs.....	11
Example 3: 44 Cell Pack, LTCs not wired in pack order .....	12
Cell Discovery and Configuration.....	13
Measurement and Statistics .....	15
High Voltage Cutoff.....	15
Low Voltage Cutoff.....	15
Balance Voltage Cutoff.....	16
High and Low Watermarks and Standard Deviation .....	16
Cell Discharging.....	17
Thermistor Operation .....	17
Alerts .....	18
Alert Processing Summary.....	19
LED Operation Summary .....	19
Using Multiple BMSCs .....	20
Powering.....	20
Charge Control Configuration .....	21
Installation .....	23
BMSC System Connections .....	24
Power (12v, gnd).....	24
Cell Loop (LP1, LP2).....	24
CAN (CANH, CANL) .....	24
Serial Port.....	24
IsoSPI (IPO, IMO) .....	24
BMSS System Connections.....	25
IsoSPI (IPO, IMO, IPI, IMI).....	25
BMSC and BMSS Cell Wiring .....	25

Cell Harness Testing .....26

BMSC and BMSS Thermistor Connections.....26

CAN.....27

IsoSPI .....28

Integrated Charge Control .....28

Charge Control with the ThunderStruck EVCC.....28

Dilithium Design Electric Vehicle Display .....30

Curtis enGage II 3100r Display.....30

Startup Checklist and Troubleshooting Hints .....32

Output Messages.....33

    Startup Banner.....33

    Hardware Discovery and LTC Diagnostics.....33

    IsoSPI Errors .....33

    HVC LVC Messages .....34

    Thermistor Messages.....34

Command Line Interface .....35

    help .....35

    show.....36

        show version .....37

        show config.....38

        show map .....38

        show cells.....39

        show ltc .....40

        show stats .....40

        show thermistors .....41

    set.....42

        set id .....42

        set canbr .....42

        set map .....42

        set hvc .....42

        set hvcdelay.....43

        set hvcc.....43

        set lvc .....43

        set lvcdelay.....43

        set lvcc.....43

        set bvc .....43

set bvmin .....43

set thmax .....44

set maxv .....44

set maxc.....44

set termc .....44

set termt.....44

set flt\_maxv.....44

set flt\_maxc .....44

reset .....44

  reset config.....44

  reset stats.....44

enable | disable .....45

  enable | disable balance.....45

  enable | disable canterm .....45

  enable | disable thermistor.....45

  enable | disable c3100r .....46

  enable | disable loopoc .....46

  enable | disable loopclosedonflt .....46

lock .....46

upgrade .....46

CAN Dictionary .....48

Hardware Revisions .....50

  Revision 2.1 .....50

  Revision 2.2.....50

Warrantee and Support .....50

Document Revision .....50

## Overview

### Battery Management

The Dilithium Design Battery Management System is a modular BMS designed for multicell lithium-ion battery packs. The BMS is implemented as two assemblies: the BMS Controller (BMSC) and the BMS Satellite (BMSS). The BMSC consists of the BMS Processor and a BMS Measurement board in a single enclosure and is a standalone 24 cell BMS. The BMSS contains a BMS Measurement board. Up to three BMSSs may be connected to the BMSC, resulting in 48, 72, and 96 cell systems. Multiple systems may be used together to monitor larger packs.

The BMS measurement board uses the Linear Technologies LTC6811 (or pin compatible LTC6804) multicell battery monitor chips (referred to as LTC). Each LTC can monitor from 4 to 12 cells. All cells can be measured in less than 2.5 ms with a total measurement error of less than 1.2mV. Accuracy is stable over voltage, temperature, and time. Multiple devices are used to monitor large cell count packs and a high-speed datalink is used that allows measurements on multiple devices to be performed simultaneously. The LTC supports diagnostics to identify hardware and wiring problems and is tolerant of common wiring errors such as shorted or open connections. The LTC supports passive cell balancing. The LTC has been designed to minimize power consumption, especially during long-term storage where battery drain is unacceptable. In sleep mode, the LTC6811 draws less than 4 $\mu$ A from the batteries.<sup>1</sup>

A BMS measurement board has two LTC6811 devices: it can measure up to 24 cells and has 10 thermistor inputs. The BMS Processor contains all firmware and provides interfaces for 12V power, CAN bus, cell loop, serial port, and LED. The BMS Processor communicates with the measurement boards using a serial datalink.

The BMS performs diagnostics on startup which verifies proper functioning of measurement circuitry, verifies cell wiring, monitors for communication errors, and does an inventory of how many cells are present in the pack. In operation, all cells in the system are measured approximately 8 times a second with high coherency and accuracy.

The BMS supports several error conditions or “alerts”. An alert is generated if any cell drops below the configurable Low Voltage Cutoff (LVC) threshold or if any cell rises above the High Voltage Cutoff (HVC) threshold. The HVC alert can be used to stop charging. The BMS supports a configurable Balance Voltage Cutoff (BVC) threshold, which, when used with the ThunderStruck EVCC, can be used to reduce charge current when charging.

The BMS supports passive cell balancing. A cell is balanced if its voltage is greater than the average cell voltage for the pack and if its voltage is greater than a configurable minimum threshold.

---

<sup>1</sup> For a product overview and datasheet for the LTC6811, see <http://www.linear.com/product/LTC6811-1>.

BMS configuration is performed using a laptop and a USB to serial cable (which connects to the BMSC using a 3.5mm jack). The serial interface is used for configuration and debugging, but is not required for normal operation. The BMS firmware supports diagnostic commands to verify proper hardware operation and wiring, to trace CAN messages, and to retrieve statistics. The BMS firmware can be upgraded using the serial port should that be necessary.

The most commonly used BMS parameters can also be configured using the Dilithium Design EV display, a touch screen LCD display.

### **Integrated Charge Control**

An integrated charge control feature is available with Dilithium BMSC firmware version v3.1.2 and later. If this feature is desired for BMSC units purchased before January 2024, please contact ThunderStruck Motors to request an update file and instructions.

With this firmware installed, the BMSC is configurable as both a BMS and charge controller. For simpler battery systems, this can be a cost-effective option, because a separate charge controller like the EVCC is not necessary.

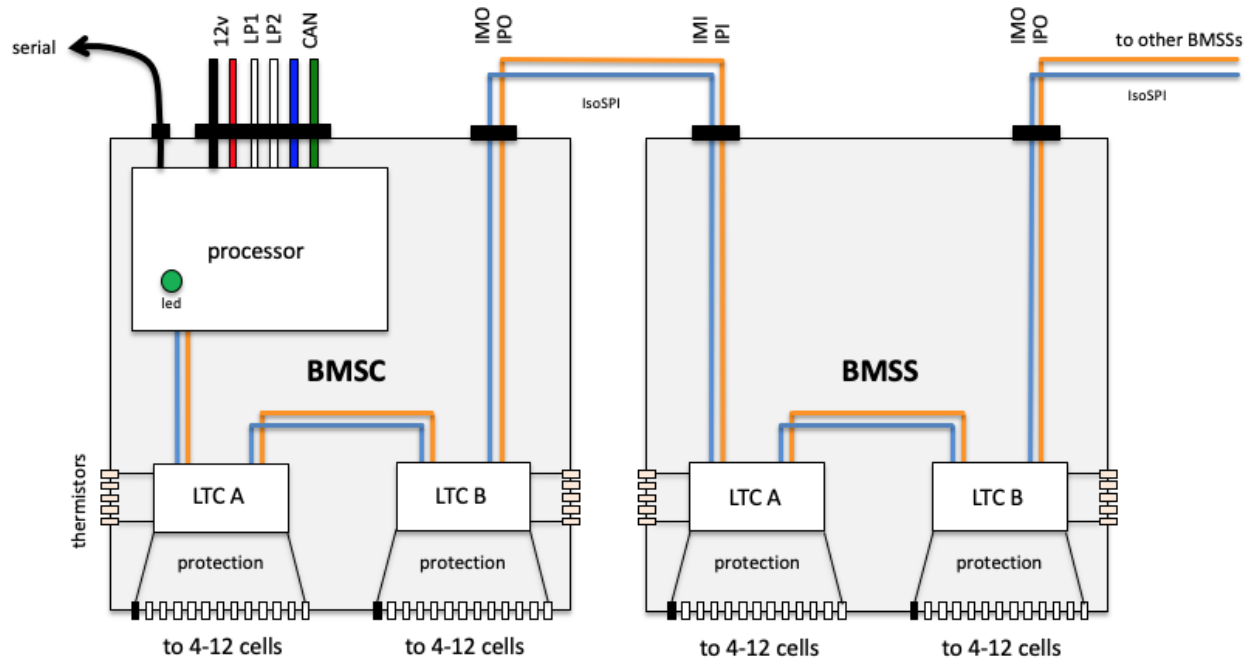
Note that the integrated charge control option has a limited feature set when compared to using the BMS with a separate EVCC. See the following feature notes:

- Communicates with charger over canbus
- Controls a single charger only
- TSM2500 series chargers only
- Charge controller communicates internally with BMS
- Charge system reporting via user interface
- J1772 protocol is not supported

See the *Charge Control Configuration* section for information about configuration and use.

## BMS Architecture

Figure 1, below, shows a block diagram of the BMSC and BMSS. The BMSC contains a processor board and a measurement board. The BMSS only contains a measurement board.



**Figure 1 – BMS Architecture**

The BMSC processor has system interfaces for 12v power, CAN, status outputs (LP1, LP2), and it contains the firmware for the BMS. The BMSC is configured using a serial interface or by CAN messages using the Dilithium Design Display.

Each measurement board has two LTC measurement devices (“LTC A” and “LTC B”). Each LTC can monitor from four to 12 cells, and supports five thermistor inputs.

The LTCs use a two-wire datalink (Linear Technologies IsoSPI) to communicate. IsoSPI can operate at 1 Mbps over 100 meters of cable, supports a Packet Error Code to detect datalink errors, and is highly immune to EMI. Each LTC has two IsoSPI interfaces: an “upstream” interface and a “downstream” interface.

The measurement board contains an internal IsoSPI connection between the two onboard LTC devices. It also contains two IsoSPI interfaces “upstream” and “downstream”. On the BMSC, there is an internal IsoSPI connection between the BMSC processor and the upstream IsoSPI interface of the first LTC. The downstream IsoSPI interface is brought out to a two pin connector. The BMSS devices have two two pin IsoSPI connectors: one for upstream and one for downstream.

BMSC firmware supports up to eight LTC devices, numbered from 1 to 8. LTC A and LTC B on the BMSC are LTCs 1 and 2. LTCs three to eight are on subtending BMSS boards, numbered in

the order of connection. In all, the BMSC can support from zero to three BMS Satellites, resulting in systems that can monitor up to 24, 48, 72 or 96 cells.

### Cell Wiring and Cell Groups

Each LTC can monitor from four and twelve cells wired in series. The cells that are monitored by one LTC is called a **Cell Group**.

Figure 2, below, shows LTC 1 connected to a Cell Group of 12 cells and LTC 2 connected to a Cell Group of 8 cells. It requires 13 connections (or “wires”) to connect to the 12 cells in a Cell Group. The wires are numbered **w0** to **w12** and cells are numbered **c1** to **c12**. All cells in a Cell Group must be wired consecutively: the positive terminal of one cell must connect to the negative terminal of the next cell. If fewer than 12 cells are to be monitored, the unused cell inputs should be at the top, all connected together to w12.

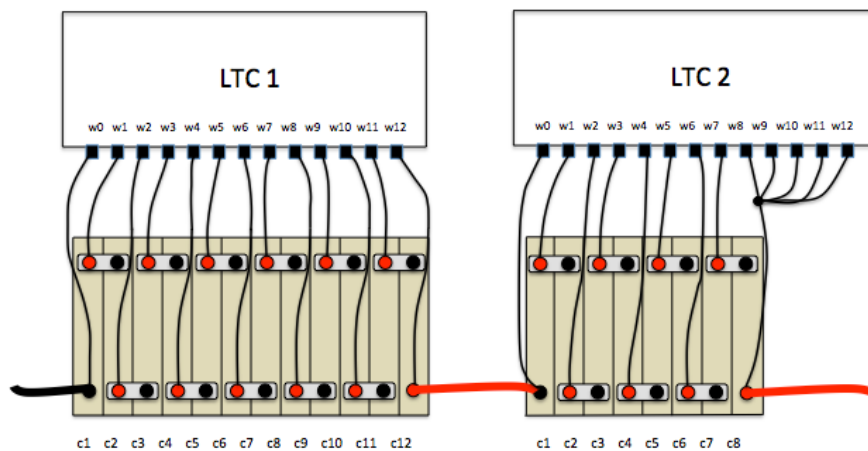


Figure 2 – Cell Group Wiring

A Cell Group should not “span” a circuit protection device (such as a contactor, fuse, or a circuit breaker). See Figure 3, below for an example.

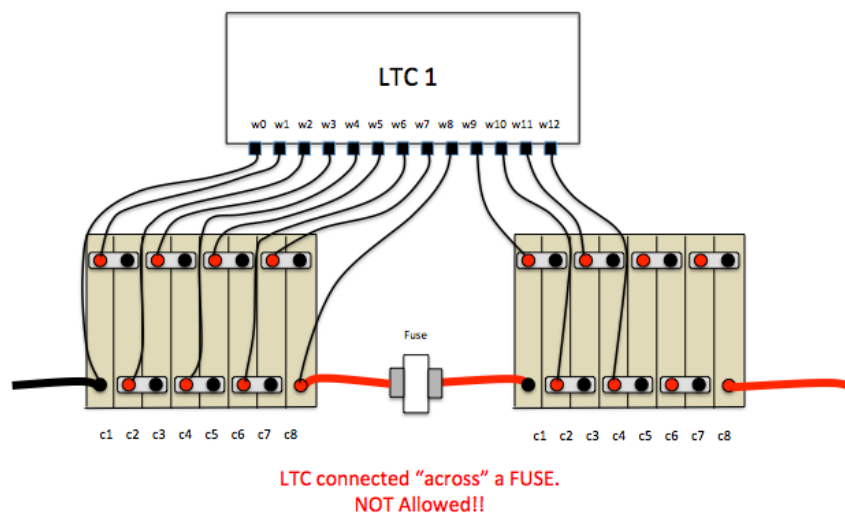


Figure 3 – LTC Wiring Spans A Fuse



The reason for this guideline is that if the fuse blows (or contactor opens, or circuit breaker trips), then the LTC may be destroyed because the LTC will span the newly open circuit and could be damaged by full pack voltage.

For similar reasons, a Cell Group should not span a traction pack jumper (because the jumper might be disconnected for maintenance). If, despite these guidelines, this cannot be avoided, and the pack is wired in this way, then the Cell Group must be disconnected at the BMS before the traction jumper is disconnected.

Figure 4, below, shows a 48-cell pack wired as one series string of cells. In this example, there are four Cell Groups, each with twelve cells.

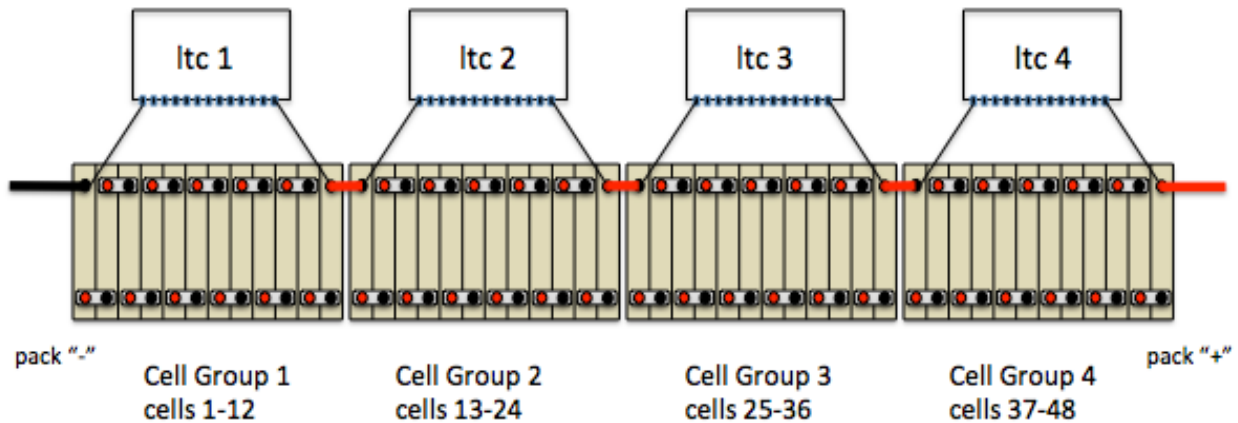


Figure 4 – 48 Cell Pack Wired to the BMS

The BMS supports configurations of multiple parallel strings of cells. In the BMS, each parallel string of cells is called a **Pack**. This is shown in Figure 5, below.

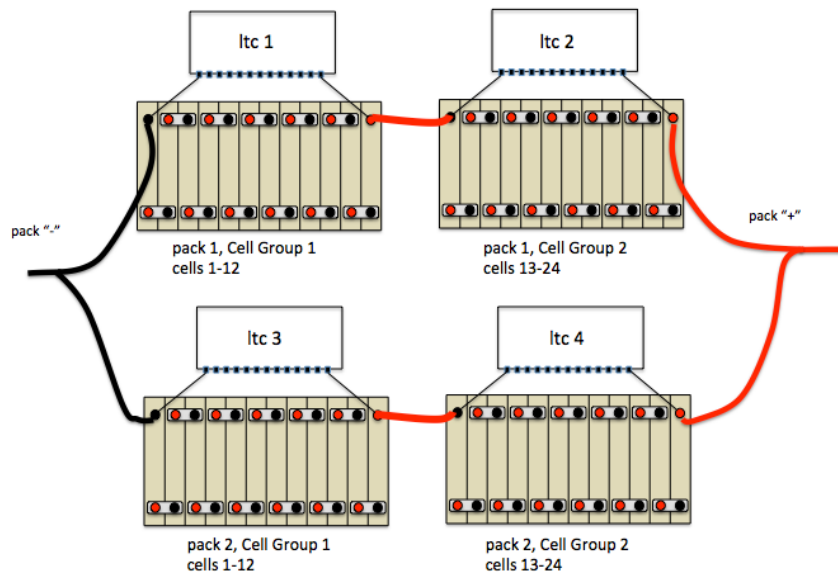
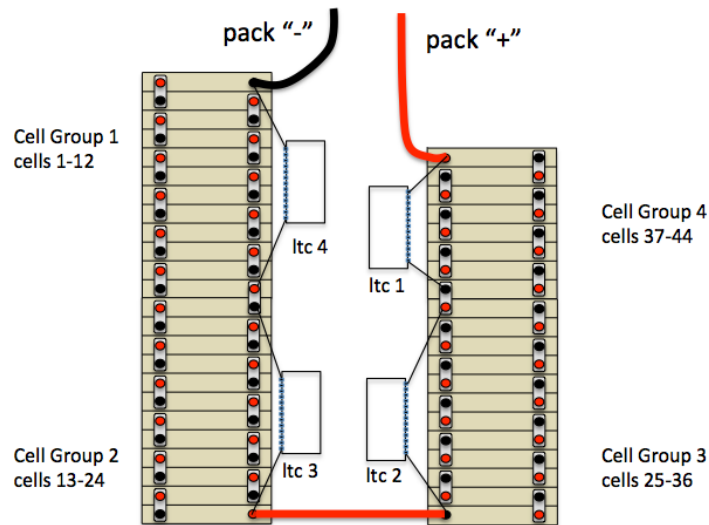


Figure 5 – Two Parallel 24 Cell Packs Wired to the BMS

In the BMS, all LTCs are electrically isolated from each other, even the two LTCs on a single measurement board. As a result, LTCs can be connected to Cell Groups anywhere in the pack and in any order, wherever it is convenient<sup>2</sup>.

Figure 6, below, shows a 44 cell pack where the LTCs are connected to Cell Groups without regard to cell order. Note that LTC 4 monitors Cell Group 1.



**Figure 6 – 44 Cell Pack, LTCs connected where convenient**

### Pack and Cell Numbering

LTCs are numbered from 1 to 8 and each LTC manages a Cell Group from 4 to 12 cells. A cell may be identified by specifying the **LTC** (from 1 to 8) and the **Cell Number** (from 1 to 12) managed by the LTC (e.g., LTC 2, cell 4). This is known as **physical cell numbering**.

As a convenience, **logical cell numbering** is also supported which numbers the cells from most negative to most positive cell, irrespective of Cell Group boundaries<sup>3</sup>. If there is only one pack, then a cell may be identified by its (logical) Cell Number from 1 to 96. If there is more than one pack, then a cell may be identified by specifying the **Pack** (from 1 to 4) and the **Cell** (from 1 to 96).

Logical Cell Numbering is implemented using a “physical to logical” mapping table. By default, the BMS assumes that there is one series string of cells, and that the LTCs are connected to cells in order and so it is not necessary to configure this table. The physical to logical mapping may however be changed to support cases when LTCs are not wired in order or if there is more than one Pack.

<sup>2</sup> It is recommended that LTCs on the same measurement board be wired to consecutive cells if possible, as it results in a lower voltage difference between groups on the measurement board.

<sup>3</sup> “Usually”, but “not always”. Cell groups consist of 12 cells. If there are partially filled cell groups in the pack then there will be corresponding holes in the numbering.

**Example 1: 48 Cell Pack**

This is the 48 cell system shown in Figure 4, above. Below is sample output from the command **show map** and **show cells** commands which shows the default cell ordering. (See the Command Line Interface, below, for a full description of these commands).

The physical to logical mapping can be displayed by using the command **show map**.

```
bmsc> show map
ltc|pack|group| cells
-----|-----|-----|-----
 1 | 1 | 1 | (c1 -c12) . . . . .
 2 |   | 2 | (c13-c24) . . . . .
 3 |   | 3 | (c25-c36) . . . . .
 4 |   | 4 | (c37-c48) . . . . .
```

Figure 4 shows is one pack with four Cell Groups. Note that LTC 1 is mapped to Group 1, LTC 2 to Group 2, and so on. Each group contains 12 cells.

The following is the output of **show cells**.

```
bmsc> show cells
c1 - 3.645v ++      c13- 3.637v ++      c25- 3.252v        c37- 3.144v -
c2 - 3.476v +       c14- 3.479v +       c26- 3.319v        c38- 3.082v -
c3 - 3.547v +       c15- 3.554v +       c27- 3.353v        c39- 3.311v
c4 - 3.478v +       c16- 3.470v         c28- 3.481v +      c40- 3.321v
c5 - 3.535v +       c17- 3.541v +       c29- 3.053v --     c41- 3.366v
c6 - 3.469v         c18- 3.468v         c30- 3.278v        c42- 3.394v
c7 - 3.351v         c19- 3.318v         c31- 3.209v        c43- 3.434v
c8 - 3.233v         c20- 3.226v         c32- 3.328v        c44- 3.308v
c9 - 3.208v         c21- 3.223v         c33- 3.069v -      c45- 3.315v
c10- 3.205v         c22- 3.231v         c34- 3.372v        c46- 3.082v -
c11- 3.217v         c23- 3.230v         c35- 3.068v -      c47- 3.090v -
c12- 3.222v         c24- 3.213v         c36- 3.283v        c48- 3.075v -
```

**Example 2: Two Parallel 24 Cell Packs**

Figure 5, above, gives an example of two parallel packs. LTC 1 and LTC 2 are connected to Pack 1, Cell Groups 1 and 2 and contains cells c1 to c24. LTC 3 and LTC 4 are connected to Pack 2, Cell Groups 1 and 2. In order to define the two Packs, it is necessary to use the **set map** configuration command.

The syntax of this command is:

```
set map <ltc> <pack> <group>
```

Use the following commands to map LTC 3 to Pack 2, Cell Group 1 and LTC 4 to Pack 2, Cell Group 2:

```
bmsc> set map 3 2 1
bmsc> set map 4 2 2
```

Check the configuration with **show map**. Now there are now two packs.

```

bmsc> show map
  ltc|pack|group| cells
-----|-----|-----|-----
  1 | 1 | 1 | (c1 -c12) . . . . .
  2 |   | 2 | (c13-c24) . . . . .
-----|-----|-----|-----
  3 | 2 | 1 | (c1 -c12) . . . . .
  4 |   | 2 | (c13-c24) . . . . .

```

Here is example output of **show cells** with two packs.

```

bmsc> show cells
pack 1 -----
c1 - 3.646v ++      c13- 3.630v ++
c2 - 3.477v +       c14- 3.476v +
c3 - 3.547v +       c15- 3.553v +
c4 - 3.478v +       c16- 3.468v
c5 - 3.534v +       c17- 3.540v +
c6 - 3.469v         c18- 3.467v
c7 - 3.352v         c19- 3.311v
c8 - 3.234v         c20- 3.223v
c9 - 3.209v         c21- 3.222v
c10- 3.206v         c22- 3.230v
c11- 3.216v         c23- 3.229v
c12- 3.222v         c24- 3.212v
pack 2 -----
c1 - 3.249v         c13- 3.143v -
c2 - 3.316v         c14- 3.082v -
c3 - 3.352v         c15- 3.311v
c4 - 3.480v +       c16- 3.321v
c5 - 3.052v --      c17- 3.366v
c6 - 3.278v         c18- 3.394v
c7 - 3.199v         c19- 3.424v
c8 - 3.325v         c20- 3.308v
c9 - 3.068v -       c21- 3.315v
c10- 3.370v         c22- 3.082v -
c11- 3.067v -       c23- 3.090v -
c12- 3.282v         c24- 3.075v -

```

### Example 3: 44 Cell Pack, LTCs not wired in pack order

See the 44 cell example in Figure 6, above. In this case, LTC 1 is connected to Cell Group 4, LTC 2 is connected to Cell Group 3, LTC 3 is connected to Cell Group 2, and LTC 4 is connected to Cell Group 1. Also note that LTC 1 only has 8 cells.

Configuring the Cell Group mapping is accomplished using the following commands:

```

bmsc> set map 1 1 4
bmsc> set map 2 1 3
bmsc> set map 3 1 2
bmsc> set map 4 1 1

```

Check the configuration with **show map**.

```

bmsc> show map
  ltc|pack|group| cells
-----|-----|-----|-----

```

```

4 | 1 | 1 | (c1 -c12) . . . . .
3 |   | 2 | (c13-c24) . . . . .
2 |   | 3 | (c25-c36) . . . . .
1 |   | 4 | (c37-c48) . . . . .

```

Note that LTC 4 is mapped to group 1, LTC 3 to group 2, LTC 2 to group 3 and LTC 1 to group 4.

Example output of **show cells** in this case would be:

```

bmsc> show cells
c1 - 2.866v          c13- 2.971v          c25- 3.770v +       c37- 3.792v +
c2 - 2.789v          c14- 3.010v          c26- 3.581v         c38- 3.585v
c3 - 2.995v          c15- 3.035v          c27- 3.656v         c39- 3.663v
c4 - 3.004v          c16- 3.148v          c28- 3.562v         c40- 3.599v
c5 - 3.044v          c17- 2.760v          c29- 3.635v         c41- 3.662v
c6 - 3.070v          c18- 2.964v          c30- 3.557v         c42- 3.596v
c7 - 3.133v          c19- 2.930v          c31- 3.438v         c43- 3.484v
c8 - 2.996v          c20- 3.018v          c32- 3.321v         c44- 3.335v
c9 - 2.998v          c21- 2.778v          c33- 3.315v         c45- -----
c10- 2.787v          c22- 3.049v          c34- 3.318v         c46- -----
c11- 2.795v          c23- 2.775v          c35- 3.315v         c47- -----
c12- 2.781v          c24- 2.968v          c36- 3.296v         c48- -----

```

### Cell Discovery and Configuration

The BMS firmware automatically detects when an LTC is connected or disconnected. LTCs are “probed” on startup, and periodically afterwards. If the number of detected LTCs change, the BMS performs an LTC diagnostic on all discovered LTCs and then prints a message to the serial port. Below is a typical message on startup:

```

ltc1 detected
ltc2 detected
ltc3 detected
ltc4 detected

```

The BMS also detects when an LTC has been disconnected or is no longer communicating. An example printout is:

```

ltc3 was present, and is now gone
ltc4 was present, and is now gone

```

The BMS automatically discovers how many cells are connected. This can be shown using the command **show map**.

```

bmsc> show map
ltc|pack|group| cells
----|----|-----|-----
1 | 1 | 1 | (c1 -c12) . . . . .
2 |   | 2 | (c13-c24) . . . . .
3 |   | 3 | (c25-c36) . . . . .
4 |   | 4 | (c37-c48) . . . . .

```

The output of **show map** shows a “dot” for every cell detected. In this example, LTCs 1 and 3 have 12 cells connected, LTC 2 has 10 cells connected and LTC 4 has 6 cells connected.

The BMS automatically uses all discovered cells in its HVC and LVC calculations. Other than setting HVC and LVC threshold values, it is not necessary to do any additional configuration in order to use the BMS.

The **lock** command is used to configure LTCs and Cells. Once the **lock** command has been entered, the BMS will generate a Cell Census alert if any locked LTCs or cells are not present.

The **show map** command shows the status of the cell.

```
bmsc> show map
ltc|pack|group| cells
-----|-----|-----|-----
 1 | 1 | 1 | (c1 -c12) . . . . .
 2 |   | 2 | (c13-c24) . . . . .
 3 |   | 3 | (c25-c36) . . . . .
 4 |   | 4 | (c37-c48) . . . . .
bmsc>
```

```
bmsc> lock
bmsc> show map
ltc|pack|group| cells
-----|-----|-----|-----
 1 | 1 | 1 | (c1 -c12) X X X X X X X X X X X X
 2 |   | 2 | (c13-c24) X X X X X X X X X X
 3 |   | 3 | (c25-c36) X X X X X X X X X X X X
 4 |   | 4 | (c37-c48) X X X X X X
```

“Locked” cells are shown as “X”s .

If the cells in LTC 4 are now disconnected (for example, by disconnecting the cell harness), the BMS will output the following messages:

```
bmsc> ltc4 was present, and is now gone
ltc4: not all configured cells detected !
```

And the output from **show map** will look like:

```
bmsc> show map
ltc|pack|group| cells
-----|-----|-----|-----
 1 | 1 | 1 | (c1 -c12) X X X X X X X X X X X X
 2 |   | 2 | (c13-c24) X X X X X X X X X X
 3 |   | 3 | (c25-c36) X X X X X X X X X X X X
 4 |   | 4 | (c37-c48) ? ? ? ? ? ?
```

where “?” indicate a locked but not present cell.

When the configuration is locked, the BMS will still detect new LTCs and cells. Suppose, for example that when the harness for LTC was disconnected, two additional cells got connected. In that case, when the harness gets reconnected, the output would look like:

```
bmsc> show map
    ltc|pack|group| cells
-----|-----|-----|-----
  1 | 1 | 1 | (c1 -c12) X X X X X X X X X X X X
  2 |   | 2 | (c13-c24) X X X X X X X X X X
  3 |   | 3 | (c25-c36) X X X X X X X X X X X X
  4 |   | 4 | (c37-c48) X X X X X X . .
```

Once the configuration has been locked, it can be locked again.

In all, the map shows four possible entries for each cell:

<blank>	cell not locked, cell not detected
.	cell not locked, cell detected
?	cell locked, cell not detected
x	cell locked, cell detected

## Measurement and Statistics

When performing a cell measurement, a request is sent to all LTC devices and measurements are made simultaneously. The BMS retrieves the measurements approximately 5ms later. In a fully populated system, all 96 cells can be measured in less than 10ms and the measurements are taken within 2ms of each other.

When the data is collected, the BMS determines if any cells are beyond the provisioned High Voltage Cutoff, Low Voltage Cutoff, and Balance Voltage Cutoff thresholds.

### High Voltage Cutoff

If a cell voltage exceeds the configured **hvc** value, then an HVC alert will be raised.

The HVC alert also supports an optional **hvcc** (HVC “clear”) parameter. If set, the HVC alert is not cleared until the cell drops lower than the configured **hvcc** parameter. (Example usage: In a solar application, the charger may always be connected to the pack and the HVC alert is used to stop the charger. Without hysteresis, the charger will cycle ON and OFF as a cell exceeds HVC only to drop back quickly once charging current stops).

### Low Voltage Cutoff

If a cell voltage drops below the configured **lvc** value, then an LVC alert will be raised.

The LVC alert also supports an optional **lvcc** (LVC “clear”) parameter. If set, the LVC alert is not cleared until the cell rises above than the configured **lvcc** parameter. (Example usage: The LVC condition may be used to disconnect the load from the pack. Without hysteresis, the load would cycle between connected and disconnected as a cell drops below LVC only to rise back quickly once the load is removed).

### Balance Voltage Cutoff

If a cell voltage rises above the configured **bvc** value, then a BVC alert will be raised. This information is sent in a CAN message and when used with the ThunderStruck EVCC, can be used to lower the charging current.

Note that the **BVMIN** value is used to control cell discharge balancing, a completely different function.

### High and Low Watermarks and Standard Deviation

The **show stats** command lists current cell statistics. However, this feature is not available for BMSC units with firmware supporting the integrated charge control feature (v3.1.2 or later).

Earlier versions record “low watermark” and “high watermark” cell values, which are the lowest and highest values for the cell since reset. (Or since **reset stats**). The BMS calculates the average cell voltage and the Standard Deviation<sup>4</sup>. Standard Deviation is a measure of how well balanced the cells are ... the lower the better. Standard Deviation is used to derive further statistics.

For each cell, the Standard Deviation Multiple (SDM) is calculated. The SDM is a ratio of Variance (the difference between the cell voltage and the mean cell voltage) to Standard Deviation. SDM is stored as a number from -9.9 to +9.9.

Example:

```
if
    SD           = 0.020v           // value of standard deviation
    ave          = 3.800v           // the average cell voltage
    c8           = 3.820v           // current voltage of cell c8
    c9           = 3.790v           // current voltage of cell c9
then
    SDM(c8)     = 1.0               // cell c8 is ave + 1.0*SD
    SDM(c9)     = -0.5              // cell c9 is ave - 0.5*SD
```

Positive values of SDM mean the cell voltage is higher than average, and negative values of SDM means the cell voltage is lower than average.

The minimum and maximum values of SDM are kept over time. The difference between the maximum and minimum SDM (the “delta”) can give an idea of cell performance under load: the larger the “delta” the more nonlinear the cell. The following is example output of **show stats**.

```
bmsc> show stats
total|-mean cell voltage-----|-standard deviation-----
      | 3.414v                    | 0.261v
pack1|-voltage----min---max--|---deviation----min---max--delta-
c1   | 3.447v   3.392v 3.520v | 0.033v +0.1s +0.1s +0.1s 0.0s
c2   | 3.400v   3.345v 3.471v | -0.014v +0.0s +0.0s +0.0s 0.0s
c3   | 3.473v   3.417v 3.546v | 0.059v +0.2s +0.2s +0.2s 0.0s
```

---

<sup>4</sup> Standard Deviation is a concept from statistics that measures “variance”. See Wikipedia for more details.



...

## Cell Discharging

Automatic cell balancing by discharging is enabled using the command **enable balance**.

When enabled, the BMS determines one from each group of six cells to balance. Once a cell has been chosen for discharge, then its associated shunting resistor is enabled. The hardware supports 24 ohm, 1W balancing resistors, which will shunt about 170ma at 4V. The best candidate for balancing will have a cell voltage that is:

- the highest voltage of any other cell in its six cell group.
- higher than the configured **bvmin** value, and
- higher than the pack mean voltage + 2.5mv.

Approximately once a minute, discharge candidates are reevaluated: charge shunting is disabled, cell measurements are retaken, new discharge candidates are determined, and then charge shunting is resumed.

If enabled, cell discharging runs whenever the BMS is turned on, whether the car is being driven or being charged.

## Thermistor Operation

Each LTC device supports five thermistors. By default, thermistor operation is disabled. The **show thermistor** command shows which thermistors are enabled. If a thermistor is connected but not enabled it will show as “.”, and if there is no thermistor connected it will show as a blank entry.

The following example shows two thermistors on LTC 1, 2, and 3.

```
bmsc> show th
ltc|  T1  |  T2  |  T3  |  T4  |  T5
-----|-----
1 |  .   |  .   |      |      |
2 |  .   |  .   |      |      |
3 |  .   |  .   |      |      |
4 |      |      |      |      |
```

The **enable thermistor** and **disable thermistor** commands are used to enable and disable thermistor inputs. Individual thermistors can be enabled and disabled (see commandline interface, below), but if the enable thermistor is used with no options, then all detected thermistors will be enabled. Once enabled, the BMS will show the thermistor readings.

```
bmsc> enable th
bmsc> show th
ltc|  T1  |  T2  |  T3  |  T4  |  T5
-----|-----
1 | 16C  | 18C  |      |      |
2 | 18C  | 17C  |      |      |
3 | 19C  | 18C  |      |      |
4 |      |      |      |      |
```

The thermistor threshold temperature is determined by the configuration parameter **thmax**. If a thermistor is enabled and if the measured thermistor value is greater than **thmax** an alert is raised and a message is printed.

```
ltc2: T3 OVERTEMP!
```

Once the thermistor reading drops below **thmax** the following message is printed:

```
ltc2: T3 temperature OK
```

Once enabled, the BMS periodically performs a thermistor census to make sure that all enabled thermistors are present. If a thermistor is not detected or is shorted, indicating faulty operation, then this will raise a “**not all thermistors detected**” alert.

## Alerts

Error conditions detected by the BMS generate an alert. An alert can affect serial port messages (see *Output Messages*, below), **show** command output (see *Commandline Interface* below), LED operation, the LP1 and LP2 outputs and the CAN interface (see *CAN Dictionary*, below).

There can be multiple Alerts present at once. Alerts are prioritized as only one Alert may be signaled by the LEDs and at most two Alerts can be signaled by the LP1 and LP2 outputs.

The following is a list of Alerts:

- **LTC Fault** - If an LTC fails diagnostics. See *Cell Discovery and Configuration*.
- **Configuration Not Locked** - If the configuration has not been “Locked”. See *Cell Discovery and Configuration*.
- **Cell Census** - If locked cells are not detected. See *Cell Discovery and Configuration*.
- **Thermistor Census** - If enabled thermistors are not detected. See *Thermistor Operation*.
- **Thermistor Overtemp** - If a thermistor reading is above thermistor threshold temperature. See *Thermistor Operation*.
- **High Voltage Cutoff** - If a cell voltage exceeds the configured **hvc** value. See *High Voltage Cutoff*.
- **Low Voltage Cutoff** - If a cell voltage drops below the configured **lvc** value. See *Low Voltage Cutoff*.
- **Balance Voltage Cutoff** - If a cell voltage rises above the configured **bvc** value. See *Balance Voltage Cutoff*.

### Alert Processing Summary

The following table summarizes alert behavior.

Alert	Priority	Serial Port Message	Additional Status	LP Outputs (1)			CAN
				Form C (2)	LP1	LP2 (3)	
Powered Down				open	off	off	
LTC Fault	1	yes	show ltc	fault	on	on	yes
Cell Census Error	2	yes	show map	fault	on	on	yes
Thermistor Census Error	3	yes	show th	fault	on	on	yes
Thermistor Overtemp	4	yes	show th	fault	on	on	yes
HVC	5	yes	show cells	fault	on		yes
LVC	6	yes	show cells	fault		on	yes
Configuration not Locked	7		show map				yes
BVC	8		show cells				yes
Discharge In Progress	9		show cells				
No Alerts				normal	off	off	yes

- (1) The operation of the LP Outputs is configurable to be either the contacts of a “Form C” relay (the default) or as discrete “open collector to ground” outputs.
- (2) A Form C relay can either be “open” or “closed”. The sense of this relay can be configured in firmware to be “open on fault” (the default) or “closed on fault”.

This column has the following options:

**open** – relay is open

**closed** – relay is closed

**normal** – relay is closed (by default)

**fault** – relay is open (by default) or closed if (“closed on fault” is configured).

**blank** – the alert does not affect the operation of the LP outputs

- (3) The LP1 and LP2 outputs, when configured for open collector operation can either be “off” (high impedance), or “on” connected to GND.

This column has the following options:

**off** – high impedance

**on** – connected to GND

**blank** – the alert does not affect the operation of this output

### LED Operation Summary

The BMSC supports a bicolor (red/green) LED; “yellow” is created by turning on both red and green. In all, the following blink patterns are defined. In the diagram, each “square” represents 125ms.



## Charge Control Configuration

The integrated charge control feature is available with Dilithium BMSC firmware version v3.1.2 and later. If this feature is desired for BMSC units purchased before January 2024, please contact ThunderStruck Motors to request an update file and instructions.

Once the firmware is installed and the feature is enabled, charging voltage, charging current and charge time parameters can be configured in the BMSC to control the charging process.

The BMSC supports two step charging. The **bulk** charge is used by all battery cell chemistries, and the **float** (or trickle) charge step is optional and typically only used for lead acid batteries.

It is the responsibility of the user to configure charge parameter values appropriate for the pack.

**WARNING: Batteries can be dangerous if overcharged and it is strongly recommended that the user check with their battery supplier to determine appropriate charging parameters.**

Charging starts when the charger and BMSC are powered on. If the BMSC is powered by the charger 12-volt output, then plugging in the charger to an AC power source will initiate a charge.

Charging normally completes when the charging current drops below a minimum configured value. The BMSC will also stop charging if an error condition occurs, such as: a cell High Voltage Cutoff error, CAN communication is lost between the BMSC and Charger (which will happen if the charger is powered down), or the maximum configured charge time is reached.

The serial interface has commands to monitor charging progress. See the serial port interface information in the *Installation* section below.

To enable the Integrated Charge Control feature, use the serial port interface to set the battery pack maximum charge voltage, maximum charging current, maximum charge time and termination current. In a typical charge, as the pack voltage nears the target charging voltage, the delivered current will drop; the termination current is used to detect when the delivered current is low and terminates the charge.

The example below sets the charging voltage to 56 volts, the charging current to 30 amps, the termination current to 2 amps, and the termination time to 1200 minutes. This is a common configuration for a 48v nominal lithium iron phosphate pack in a 16s configuration.

```
bmsc> set maxv 56
bmsc> set maxc 35
bmsc> set termc 2
bmsc> set termt 1200
```

The **show config** command lists the updated charging parameters as shown below:

```
bmsc> sh config
id      : 1
canbr   : 250kbps
lvc     : 2.800v
hvc     : 3.600v
lvcc    : n/a
hvcc    : n/a
bvc     : n/a
bvmin   : 3.300v
thmax   : 50C
thmin   : 0C
--charging-----
maxv    : 56.0V
maxc    : 35.0A
termc   : 2.0A
termt   : 1200 mins
bmsc>
```

When using a lead acid pack or if the float charge step is desired for lithium cells, enter the following parameters. The charging portion of the **show config** report is also shown.

```
bmsc> set flt_maxv 54
bmsc> set flt_maxc 2
bmsc> show config
...
--charging-----
maxv    : 56.0V
maxc    : 35.0A
termc   : 2.0A
termt   : 1200 mins
flt_maxv : 54.0V
flt_maxc : 2.0A
bmsc>
```

The float step is initiated when the bulk charging step is completed. This happens after **maxv** has been reached and the current has reduced to the **termc** setting.

When float settings have been configured, the charge process will remain active indefinitely, unless power is interrupted or if an error condition is noted by the BMSC.

Note that when the BMSC is powered on for long periods of time, the 12v supply battery must be kept charged. If the battery system is inactive for long periods, monthly status checks should be made, including reviewing pack cell voltages and checking the 12v supply.

## Installation

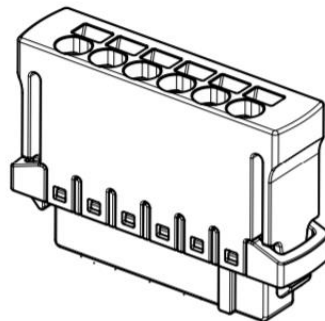
Both the BMSC and BMSS are housed in Serpac WM031 3.3” x 5.4” x 1.1” enclosures. For a detailed specification, see [http://www.serpac.com/userprints/WM031\\_UP.pdf](http://www.serpac.com/userprints/WM031_UP.pdf).

Figure 7, below, shows the BMSC and BMSS system connections.



**Figure 7 – BMSC and BMSS System Connections**

BMS hardware v2.1 uses Harting “push-in” connectors for the system connectors. The Harting connectors accept 20-24 gauge stranded or solid wire; stranded 20-gauge wire is recommended. To make a connection, strip the wire back 1/4”. Twist the wire end or alternately, tin the end of the wire and insert. Be sure that all strands of wire get correctly inserted to prevent shorting between adjacent wires.



**Figure 8 – Harting Push-in Connector**

Removing the wire from the connector requires a removal tool: insert the tool into the associated slot above the wire and wiggle it in. This will collapse the spring holding the wire and the wire can be removed.

The removal tool is Harting 14990000001. Alternately, the Molex KK terminal removal tool, part number W-HT-1884, is widely available and has been found to work well. The BMSC uses one 6-pin connector and two 2-pin connectors. The BMSS uses two 2-pin connectors.

The system connector part numbers are:

- 6p Systems Connector                      Harting 14310613101000
- 2p IsoSPI Connector                        Harting 14310213101000

## BMSC System Connections

Figure 9, below, shows the BMSC system connections.

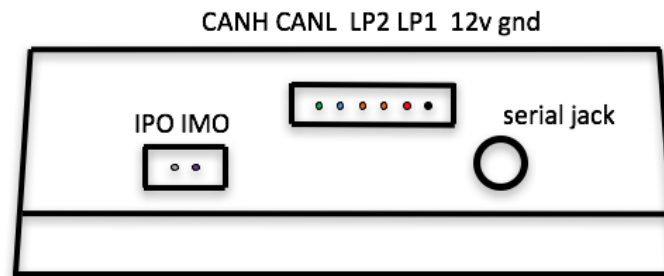


Figure 9 – BMSC System Connections

The BMSC has one 6-pin system connector, a 2-pin IsoSPI connector, and a 3.5mm serial port Jack.

- The serial jack connects to a USB-to-serial cable that is used to configure the BMS.
- The 6-pin connector provides power, LP1 and LP2 discrete outputs and the CAN interface.
- The 2-pin IsoSPI connector is used to connect to the first BMSS, if present.

### Power (12v, gnd)

Power is provided at **12v** and **GND**. The BMS requires approximately 15ma of 10-15V power. The BMS should be powered both when the EV is being operated and when being charged.

### Cell Loop (LP1, LP2)

The **cell loop** outputs are provided at **LP1** and **LP2**. These contacts can either be configured as “dry contacts” (e.g., terminals for a “form C” relay), or “open collector to ground” outputs, also rated to 200ma.

These outputs are rated to 200ma. Protection diodes are provided so that these outputs can be used to operate the coil of an external relay.

### CAN (CANH, CANL)

**CAN** is provided on **CANH**, **CANL**. See below for guidelines on wiring and configuring CAN operation.

### Serial Port

Before using the serial port, host computer drivers and a terminal application must be installed. See the document *Serial Port Utilities Document* on the ThunderStruck Motors BMSC webpage for details, or link to the document directly below:

[https://www.thunderstruck-ev.com/images/companies/1/DD\\_SerialPortUtilities\\_v1.3.pdf](https://www.thunderstruck-ev.com/images/companies/1/DD_SerialPortUtilities_v1.3.pdf)

### IsoSPI (IPO, IMO)

The **IsoSPI datalink** is connected to the **IPO** and **IMO** (IsoSPI Plus Out and IsoSPI Minus Out) signals on the 2 pin connector. These signals are connected to the corresponding **IPI** and **IMI** signals on the first BMS Satellite. If this is a 24 cell BMS, then there is no BMS Satellite and these pins are left unconnected.



### BMSS System Connections

The figure below shows the BMSS system connections.

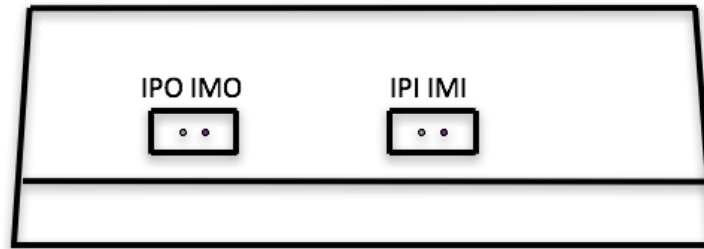


Figure 10 – BMSS System Connections

### IsoSPI (IPO, IMO, IPI, IMI)

The IsoSPI datalink is given on IPO and IMO (IsoSPI Plus Out and IsoSPI Minus Out) and IPI and IMI (IsoSPI Plus In and IsoSPI Minus In). Connect IPI and IMI to the IPO and IMO connections of the “upstream” device and connect IPO and IMO to the IPI and IMI signals of the downstream device, if present.

### BMSC and BMSS Cell Wiring

Cell wiring is identical for the BMSC and BMSS. The Cell Harness connections are shown below.

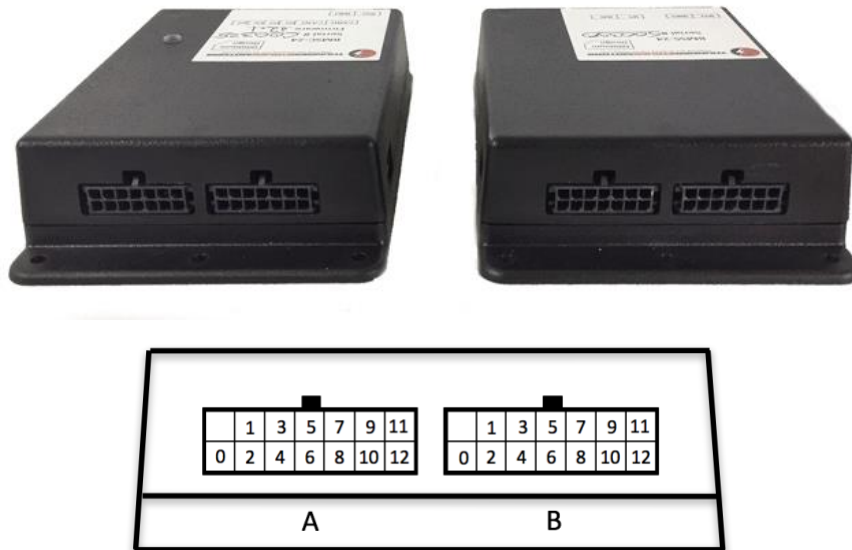


Figure 11 – Cell Harness Connections

The cell harness connector is a 14p Molex 43045-1422, with the matching connector Molex 43025-1408 with crimp terminal 46235-5001. Cell harnesses with pigtail wire connections are provided.

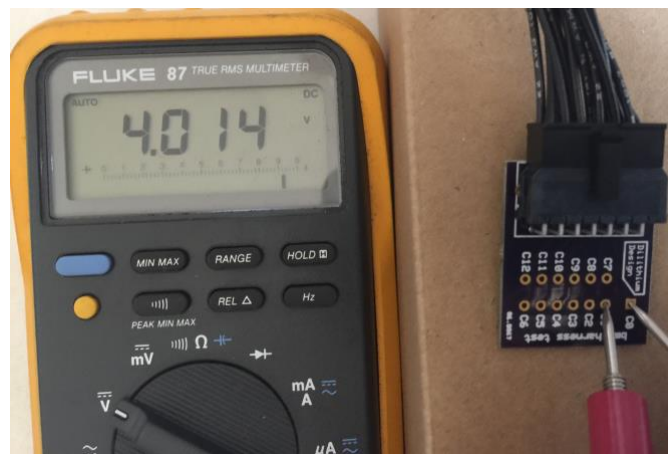
When wiring the cell harness connector, all cells within a cell group must be connected in series. Each cell group has 13 connections to connect to the 12 cells in the group. If fewer than 12 cells are to be monitored, then the unused cell inputs should be at the top, all connected to w12.

## Cell Harness Testing

A Cell Harness tester is provided with BMS. Once the cell group is wired, use a voltmeter to verify that cells are connected with the proper polarity, and that they are in series. Verify that voltage difference between adjacent cells is correct; this value depends on the cell chemistry and the state of charge. Also verify that if fewer than 12 cells are used in the group that the top cells all have identical voltage readings ... indicating that they are correctly connected.

**NOTE: The Harness Tester is connected directly to pack voltage and so there can be up to 60V connected on the harness. Be Careful! While it is connected to the pack, do not place the harness tester on anything metallic!**

The following picture shows a voltmeter measuring the voltage of Cell 1.



**Figure 12 – Using the Cell Harness Tester**

Once the cell harness is verified, it is safe to plug into the BMSC or BMSS. Cell Harness connectors may be plugged in or unplugged in any order. It is not necessary to remove 12V power from the BMS in order to plug or unplug Cell Harness connectors.

## BMSC and BMSS Thermistor Connections

The BMSC and BMSS thermistor connectors are on the sides of the enclosure. Each connector supports five thermistor connections, two connections for each thermistor.

The thermistor connector is a 10p Molex 87833-1031, with the matching connector Molex 51110-1060 with crimp terminal 50394-8054. A harness with pigtail wire connections is available. This be soldered to a wire lead thermistor or a connector may be added to allow it to connect directly to the thermistor connection in an OEM battery pack.

The thermistor harness connectors are shown in Figure 13, below.



Figure 13 – Thermistor Connections

The pinout of this connector is the same on both sides of the enclosure (the Thermistor 1 connections are to the right). A thermistor has two leads: polarity is not important.

The BMS has been designed to work with a 1% 10K NTC thermistor, with a “B value” of 3435K. One possible choice is the Vishay NTCLE413E2103F520.

**CAN**

CAN is a robust communications protocol designed for automotive applications. CAN uses a two-wire interface; the signals are designated CANH (“CAN High”) and CANL (“CAN Low”). A CAN network is a daisy-chain, multi-station network that should be terminated on both ends of the string by 120ohm termination resistors. See below for a simple network diagram.

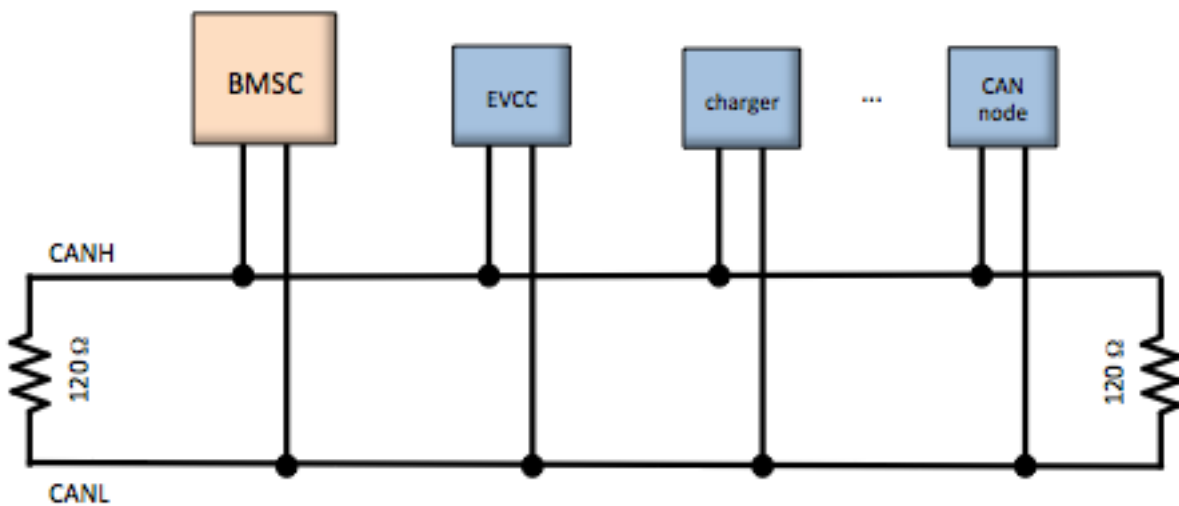


Figure 14 – CAN Network Diagram

CAN wiring should be kept short and the conductors should be twisted. Wiring should be placed away from EMI (Electro-Magnetic Interference) such as the motor and controller, and parallel runs next to the traction cabling should be avoided.

In a simple installation, there will be only a few nodes on the network: the BMSC, the charger, and possibly the EVCC, with short and direct connections between the three. In this case, hand-twisted wiring usually works fine. For longer runs, more nodes, or cases where EMI may be an issue, shielded cable may be used. If a shielded cable is used, the shield should be connected to chassis ground at a single point.

By default, the BMSC supports a CAN data rate of 250Kbps and 29-bit (Extended) CAN addressing. The CAN data rate is configurable.

The BMSC supports a programmable can termination resistor (see **enable canterm**).

### IsoSPI

The wiring guidelines for IsoSPI are similar to those of CAN. Twisted pair is recommended: for short runs, hand twisted wire is fine. However, for longer runs, shielded cable may be used. Note that IsoSPI is a point to point link, not a bus. Two wires are used between nodes. External termination resistors are not required.

### Integrated Charge Control

If the integrated charge control feature has been enabled, then the BMSC installation can be completed as follows:

- Connect BMSC +12V so that it receives power from the 12 volt keyswitch circuit while charging and operating. Optionally, use a relay as described below to power the BMSC.
- Connect BMSC GND to EV chassis ground (12v-).
- Connect BMSC CANH and CANL to the charger CAN network.
- Add a 120 ohm resistor across CAN network at the connection to the charger or other more distant device.
- Enable the BMSC internal termination by entering the command `"bmsc> enable canterm"`.

To keep the BMSC powered both during charging and operation, use a double throw relay controlled by the 12-volt keyswitch to power the BMSC when operating. When the key is off while charging, use the normally closed relay contact to power the BMSC from the 12v output connector on the charger.

### Charge Control with the ThunderStruck EVCC

If the ThunderStruck EVCC is being used instead of the BMSC integrated charge control feature, then the BMSC installation is completed as follows:

- Connect BMSC +12V to the EVCC 12V\_SW output, and BMSC GND to EV chassis ground.
- Connect BMSC CANH and CANL to the EVCC and charger CAN network.
- Make sure the CAN termination resistors are enabled at the ends of the CAN network. The BMSC has a programmable CAN termination resistor.
- Configure the EVCC to use the CAN BMS (For EVCC 3.0, the command is `"evcc> set bms bmsc"`; for EVCC 2.4, the command is `"evcc> set bms can"`).

When using the BMS with the EVCC, the CAN interface is recommended, and the Loop Interface is not necessary.

Up to four BMSC systems may be used in an EV. When using multiple BMSC systems, install the systems individually, and wire all to the EV CAN network.

Each BMSC requires a unique id, from 1 to 4. By default, the BMS has an id of 1.

If the EVCC is being used, enable all BMSCs.

Example, if three BMSCs are being used, with ids 1, 2, and 3, then these may be enabled in the 3.0 EVCC by the command:

```
evcc> set bms bmsc bmsc2 bmsc3
```

If the integrated charge control feature is being used (no EVCC) then enable integrated charge control on only one of the BMSCs on the network.

### Dilithium Design Electric Vehicle Display

The Dilithium Design Electric Vehicle Display is a 3.2” 240x320 pixel LCD capacitive touch display with an attached bezel. This display shows pack State of Charge, Pack Current and Voltage Measurements, Cell Status, Pack Status, and individual cell voltages.

The Display can be used to configure some of the most common BMS parameters. Up to four BMS Controllers may be managed by a single Display; cells may be organized in up to four parallel packs.

If an EVCC is connected to the canbus, then the Display can configure most EVCC parameters, monitor the J1772 plug and charging status, and can show the EVCC charge history.

The Display Pack State of Charge is determined by “charge counting” using an optional Current Sensor.

The Display supports an optional analog Fuel Gauge.

No configuration is required in the BMS to use the EV Display.

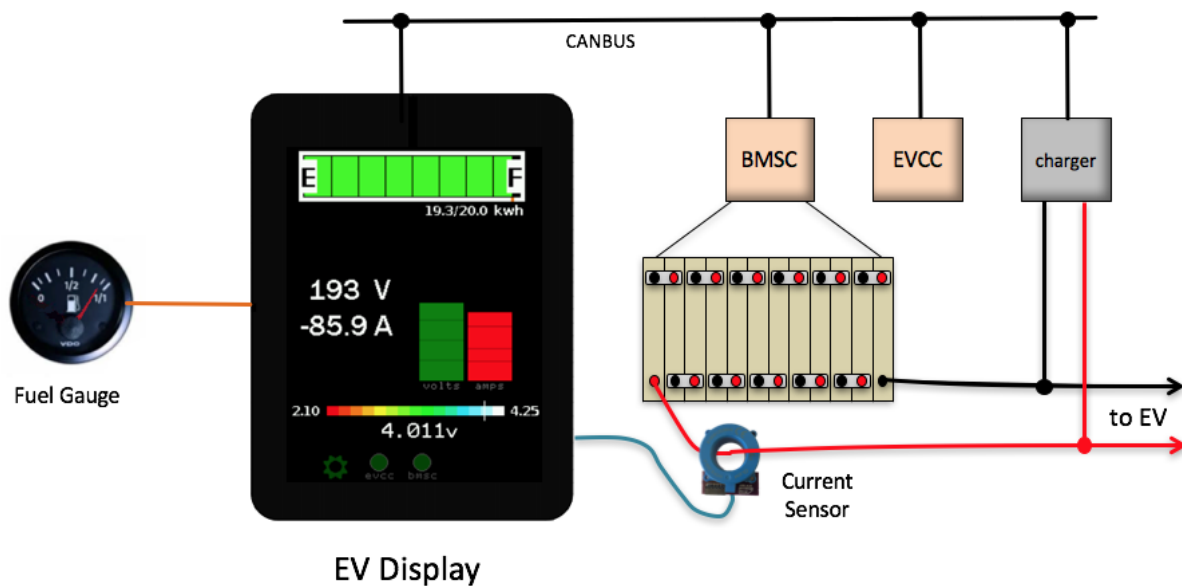


Figure 15 – Dilithium Design EV Display

### Curtis enGage II 3100r Display

The Curtis 3100r display is a 52mm round display that requires power and CAN to operate. This display can display State of Charge, has a 7-segment alphanumeric LCD display, graphical icons, an LED, backlight and two buttons.



**Figure 16 – c3100r Display**

The c3100r must be programmed to operate at a CAN rate of 250kb to match the CAN datarate of the BMS (if this display is purchased from ThunderStruck Motors, the datarate will already be configured). Once the c3100r is connected to power and the CAN network, the display may be enabled in the bms by the command:

```
bmsc> enable 3100r
```

Pack **State of Charge** is shown in the top 10 segment SOC arc. The BMS uses pack voltage measurements and HVC and LVC configuration to compute State of Charge; a current sensor is not required. This approach works best for packs that have a linear discharge curve.

The c3100r **LED** blinks when the pack is in HVC or LVC. The LED is on for Thermistor over-temperature.

The **Wrench** Icon blinks if there is a configuration error: configuration not locked, cell census error, thermistor census error. This corresponds with the RED/GREEN alternating blink on the BMSC.

The **Battery** Icon is on when the pack is balancing.



The left button sequences the alphanumeric display between **pack voltage, minimum cell voltage, maximum cell voltage, average cell voltage, and standard deviation.**

- The pack voltage is shown in tenths of a volt.
- The minimum cell voltage shows the cell number, an "underbar" and then three digits which is the cell voltage in hundredths, but with no period. (So "354" is 3.54v).
- The maximum cell voltage shows the cell number, an "upperbar" and the value.
- The average cell voltage shows three "middle bars" and a value.
- The standard deviation shows "Sd", "space" and then three digits of standard deviation.

The right button toggles the **backlight**.

The c3100r is limited for use in systems with one bmsc.

## Startup Checklist and Troubleshooting Hints

### Planning

Read documentation

Determine the number of packs and cell groups required

Plan system connections, powering strategy, CAN bus and IsoSPI routing

### Physical Installation

Physically install BMSCs and BMSS

Route power, CAN, IsoSPI datalinks, Cell loop wiring

### Wiring Harnesses

Make harness to cell connections

Verify harness with harness verification tool

### System Startup

Install terminal drivers and terminal emulation program. See the document *Serial Port Drivers and Firmware Upgrade*.

Power up BMSC and verify serial port operation

Set BMSC id (if necessary)

Enable CAN termination (if necessary)

Connect cell harnesses

Verify that Cells are "discovered" (show cells)

Configure HVC and LVC

Enable Cell Discharging, if desired (enable balance)

Basic operations are now working. Further system testing depends on the installation. If the EVCC is being used with CAN, then CAN messages can be verified at the EVCC (using "trace can").



## Output Messages

### Startup Banner

When the BMSC is powered up, it prints a banner message with the version number.

```
*****
*                BMS Controller v3.1.2                *
*                Dilithium Design                      *
*****
```

### Hardware Discovery and LTC Diagnostics

On startup (and periodically afterwards), the BMSC probes to see how many LTC devices are present. A diagnostic test is performed on all LTC devices. In addition, the BMSC determines how many cells are present in each LTC.

If the configuration has been “locked”, the BMSC will perform a “census check”. An error is printed if not all “locked” cells are detected. Example output is given below:

#### Initial discovery:

```
ltc1 detected
ltc2 detected
ltc3 detected
ltc4 detected
```

#### Initial discovery, with a failed diagnostic:

```
ltc1 detected
ltc2 detected
ltc3 detected, diagnostic failure=0x1000
ltc4 detected
```

There are several internal diagnostics performed on the LTC devices to verify proper operation.

#### Initial discovery, census error:

```
ltc1: not all configured cells detected !
ltc1: not all configured cells detected !
```

This message will repeatedly print on the serial port.

#### LTC goes absent:

```
ltc1 was present, and is now gone
ltc2 was present, and is now gone
ltc3 was present, and is now gone
ltc4 was present, and is now gone
```

### IsoSPI Errors

IsoSPI messages have a Packet Error Code which is checked for every message and errored packets are printed. If a measurement board loses power or gets disconnected from its pack connection, then there may be a short burst of datalink errors until the BMSC determines that the LTC is no longer connected. Repetitive messages indicate that the datalink is experiencing

errors. (Possible mitigations are: to check the connections, make sure the wiring is tightly twisted, shorten the data link, or add shielding)

Example output looks like:

```
bmsc> 00:01:28.7  0004 03 ffff ffff ffff ffff 4c66
00:01:28.9  0004 03 ffff ffff ffff ffff 4c66
00:01:29.0  0004 03 ffff ffff ffff ffff 4c66
00:01:29.2  0004 03 ffff ffff ffff ffff 4c66
```

### HVC LVC Messages

When a cell voltage exceeds the High Voltage Cutoff (HVC) threshold, or goes below the Low Voltage Cutoff threshold, then a message is printed, which includes the Pack and Cell Number. If the cell recovers then a message is printed. Example output:

```
1/c25: in HVC !
1/c26: in HVC !
1/c25: OK
1/c26: OK
1/c12: in LVC !
1/c11: in LVC !
```

Note that if a cell is declared as HVC and if the `hvc` parameters is being used, then the HVC condition will not clear until it drops below `hvc`.

### Thermistor Messages

When a thermistor exceeds the `tmax` threshold a message is printed. The message includes the LTC number and thermistor number.

```
ltc2: T3 OVERTEMP!
```

And when it recovers a message is printed.

```
ltc2: T3 temperature OK
```

The thermistor threshold temperature is determined by the configuration parameter `tmax`. When a thermistor is overtemp an “alert” is present in the `show` command. The measured value can be determined by the `show thermistor` command.

## Command Line Interface

This section describes the command line interface commands.

### help

The **help** command prints out command help.

```

bmsc> help
  SHow [<>|Version|COnfig|MAP|CELLS|LTC|STATS|Thermistors]
    <>      - status
    version - firmware version
    config  - configuration
    map     - cell group map
    cells   - cell values
    ltc     - measurement chips
    stats   - cell statistics (pre-v3.1 only)
    thermistors - thermistor readings
SEt  [<>|ID|CANBR|MAP|LVC|HVC|HVCDELAY|LVCDELAY|HVCC|LVCC
     |BVC|BVMIN|THMAX|THMIN|MAXV|MAXC]
    <>      - show config
    id      - bmsc ID (1..4)
    canbr   - CAN baud rate (125|250|500)
    map <ltc> <pack> <grp> - Map an LTC to a Pack/Cell Group
    hvc     - High Voltage Cutoff
    lvc     - Low Voltage Cutoff
    hvcdelay - Delay (in seconds) before reporting HVC alert
    lvcdelay - Delay (in seconds) before reporting LVC alert
    hvcc    - High Voltage Cutoff Clear
    lvcc    - Low Voltage Cutoff Clear
    bvc     - Charge Balancing Voltage Cutoff
    bvmin   - Cell Discharge Balancing Voltage Minimum
    thmax   - Thermistor Max Temperature
    thmin   - Thermistor Min Temperature
    maxv    - charge voltage
    maxc    - charge current
    termc   - charge termination current
    termt   - charge termination time
    flt_maxv - float charge voltage
    flt_maxc - float charge current
REset [CONFIG|STATS]
    config  - reset configuration to defaults
    stats   - reset cell statistics
ENable | DISable [BALANCE|CANTERM|C3100R|Thermistor|LOOPCLOSEDONFLT]
    Balance - enable/disable Cell Discharge Balancing
    canterm - enable/disable CAN termination resistor
    c3100r  - enable/disable Curtis 3100R display
    thermistors <ltc> <num> - enable/disable thermistor(s)
    loopoc  - loop is configured as open collector outputs (HVC lp1, LVC lp2)
    loopclosedonflt - loop logic reversed: closed on FLT
    speedhut - SPEEDHUT CAN messages
TRace [CAN|CHARGE|OFF]
    - debug trace options
LOCK - lock configuration
UPGRADE - performs a firmware upgrade
bmsc>

```

In many cases, either a full version or an abbreviated version of a command (or command parameter) can be used. This is shown in the “help” with the use of uppercase and lowercase

letters. For example, the abbreviation for **SHow** is **sh**, the abbreviation for **SHow THERmistors** is **sh th**.

### show

The **show** command displays configured parameters or status. If **show** is entered without parameters, current status is displayed. This includes per-pack and total statistics, alerts, and BMS uptime.

In the following example, there is a single pack. In this case **voltage** is pack voltage, **cells** is the number of cells, **mean** is the average cell voltage, and **std dev** is the cell standard deviation.

```
bmsc> show
pack
  voltage: 150.92v
  cells  : 48
  mean   : 3.144v
  std dev: 0.137v
alerts   : configuration not locked
uptime   : 0 hour(s), 0 minute(s), 43 second(s)
```

For configurations that contain multiple packs, per-pack and total statistics are given.

```
bmsc> show
pack1
  voltage: 78.65v
  cells  : 24
pack2
  voltage: 78.50v
  cells  : 24
total
  cells  : 48
  mean   : 3.274v
  std dev: 0.143v
alerts   : pack in HVC
uptime   : 0 hour(s), 25 minute(s), 47 second(s)
bmsc>
```

For configurations using the integrated charge control feature, the **show** command also returns information about the charge in progress, or the last charge, once the float step is entered.

```
bmsc> sh
pack1
  voltage: 56v
  cells  : 16
total
  cells  : 16
  mean   : 3.515v
  std dev: 0.005v
charge   : COMPLETE/FLOAT
last charge ...
voltage: 56.0v
current: 35.0A
```

```
watthrs:      1250 WHr
reason  : NORMAL
time    :      38 mins
alerts  : none
uptime   : 0 hour(s), 37 minute(s), 53 second(s)
bmsc>
```

The list of BMS alerts includes the following:

```
LTC<n> fault
cell census
thermistor census
thermistor overtemp
HVC
LVC
BVC
balancing active
configuration not locked
```

The parameter **uptime** gives the time since power on.

#### **show version**

The **show version** command displays firmware version number and build date.

```
bmsc> show version
version   :    v3.1.2 ; Oct 17 2023 15:01:16
```

**show config**

The **show config** command displays configuration parameters.

```
bmsc> show config
id          : 1
lvc         : 2.400v
hvc         : 3.400v
lvcc        : n/a
hvcc        : n/a
bvc         : n/a
bvmin       : n/a
thmax       : 50C
options     : balance (cell discharge balancing is enabled)
            : canterm (CAN termination resistor enabled)
            : 3100r (Curtis display enabled)

bmsc>
```

The **id** is the BMSC Id: one of 1, 2, 3, or 4.

The **lvc** is the Low Voltage Cutoff, in volts. The **hvc** is the High Voltage Cutoff, in volts.

The **lvcc**, **hvcc**, **bvc**, and **bvmin** parameters are optional. If they are configured to be 0, they are not used. For these parameters see the text for more details on operation.

- **lvcc** is the Low Voltage Cutoff Clear parameter, in volts, this value affects when the LVC alert is cleared.
- **hvcc** is the High Voltage Cutoff Clear parameter, in volts, this value affects when the HVC alert is cleared.
- **bvc** is the Charge Balancing Voltage Cutoff, in volts
- **bvmin** is the automatic Balancing Minimum Voltage, in volts.

The **thmax** is the maximum allowable thermistor temperature reading, in Centigrade.

The following options may be configured

**canterm** - Enables the programmable CAN termination resistor  
**balance** - Enables cell discharge balancing  
**3100r** - Enables the Curtis 3100r display  
**loopoc** - Configures LP1 and LP2 to be open collector outputs  
**loopclosedonflt** – Reverses loop logic to be closed on fault, not closed when OK.

**show map**

Each LTC is mapped to a Pack and a Cell Group, and the **show map** command shows the mapping.

```
bmsc> show map
ltc|pack|group| cells
-----|-----|-----|-----
1 | 1 | 1 | (c1 -c12) X X X X X X X X X X X X
```

```

2 |      | 2 | (c13-c24) X X X X X X . . . . .
3 |      | 3 | (c25-c36) X X X X X X X X X X X X
4 |      | 4 | (c37-c48) ? ? ? ? ? ?
    
```

In this display, the following letters indicate the state of the cell:

```

<blank>    cell not locked, cell not detected
.          cell not locked, cell detected
?          cell locked, cell not detected
X          cell locked, cell detected
    
```

See the text for examples of this command. Also see the commands **set map** and **lock**.

**show cells**

The **show cells** command lists current cell voltage measurements and status.

If the cell value shown is:

```

-----    then the cell voltage is less than 20mv
??????    the cell is configured, but not present
           otherwise the latest cell voltage is shown.
    
```

After the cell voltage, there may be an Indicator Flag. There is room for only one Indicator Flag, with the following priority:

```

>>DIS    the cell is being discharged
>HVC     the cell voltage is greater than the High Voltage Cutoff threshold
>HVCC    the cell voltage was in HVC and is now greater than the High Voltage
          Cutoff Clear threshold
>BVC     the cell voltage is greater than the Balance Voltage threshold
<LVC     the cell voltage is less than the Low Voltage Cutoff threshold
<LVCC    the cell voltage was in LVC and is now less than the Low Voltage Cutoff
          Clear threshold
    
```

If there is no Indicator Flag, then there may be a Deviation Flag which highlights high or low cells. The Deviation Flag is determined by the Standard Deviation Multiple (SDM). See the text above for an explanation of this field.

```

++        SDM is > 1.5
+         SDM is > 1.0
-         SDM is < 1.0
--        SDM is < 1.5
    
```

If there is no Indicator Flag and no Deviation Flag, then the cell is within 1 Standard Deviation of the mean.

Example output:

```

bmsc> show cells
c1 - 3.042v +      c13- 3.027v +      c25- 2.718v -      c37- 2.713v -
c2 - 3.472v >HVC  c14- 2.995v        c26- 2.918v        c38- 2.706v -
c3 - 3.421v >HVCC c15- 3.054v +      c27- 2.959v        c39- 2.908v
c4 - 3.012v +      c16- 2.993v        c28- 3.080v ++     c40- 2.913v
c5 - 3.052v +      c17- 3.045v +      c29- 2.706v -      c41- ??????
c6 - 3.001v +      c18- 2.991v        c30- 2.910v        c42- ??????
c7 - 2.799v        c19- 2.766v        c31- 2.708v -      c43- ??????
c8 - 2.788v        c20- 2.773v        c32- 2.920v        c44- ??????
c9 - 2.774v        c21- 2.777v        c33- 2.714v -      c45- -----
c10- 2.771v        c22- 2.783v        c34- 2.980v        c46- -----
c11- 2.783v        c23- 2.783v        c35- 2.722v -      c47- -----
c12- 2.788v        c24- 2.770v        c36- 2.917v        c48- -----
bmsc>

```

### show ltc

The **show ltc** command shows information about the LTC measurement chips, and is provided primarily to troubleshoot potential hardware problems.

Example output:

```

bmsc> show ltc
ltc| status | rev| Van  | Vdig | Vref | TMP | SOC | SUM | conf | disc
----|-----|---|-----|-----|-----|-----|-----|-----|-----|-----
1 | ok   | 3 | 5.004v| 3.083v| 3.000v| 10C | 43.48v | 43.49v | 0x0fff | 0x0fff
2 | ok   | 3 | 4.997v| 3.084v| 2.998v| 11C | 43.48v | 43.49v | 0x0fff | 0x0fff
3 | ok   | 3 | 4.986v| 3.090v| 2.998v| 12C | 43.48v | 43.49v | 0x0fff | 0x0fff
4 | ok   | 3 | 4.985v| 3.096v| 3.001v| 12C | 43.48v | 43.49v | 0x0fff | 0x0fff

```

The **status** value should be **ok**. If not then this field will show a hex failure value.

The **rev** shows the ltc chip hardware revision.

The values **Van**, **Vdig**, **Vref** are internal voltage measurements on the chip. The nominal values are be 5v, 3v, and 3v respectively.

The **TMP** is the internal die temperature of the chip.

The values of **SOC** and **SUM** is the total voltage managed by the chip. These numbers are calculated in two different ways: as a total and by adding up individual cell measurements. These numbers should be close to each other.

The **conf** is a hexadecimal representation of which cells have been configured for this LTC using the lock command. The **disc** field is a hexadecimal representation of which cells have been discovered. These should be equal. Note that an equivalent, more user friendly, representation of this data is given in **show map**.

### show stats

The **show stats** command lists current cell statistics. Note, this feature is not available for BMS units with firmware supporting the integrated charge control feature (v3.1.2 or later).



**Average cell voltage** and **std deviation** (standard deviation) are calculated over all cells. For pack data, the **voltage** is the current reading of the cell. The **min** and **max** values are the lowest and highest value of the cell since power on.

The **variance** is the difference, in volts, of the current cell voltage with the average cell voltage. For an explanation of **sdm**, **min**, **max**, and **delta**, see the text.

```
bmsc> show stats
total|-mean cell voltage-----|standard deviation-----
      | 3.414v                      | 0.261v
pack1|-voltage----min----max--|----deviation----min---max--delta-
c1  | 3.447v  3.392v 3.520v | 0.033v +0.1s +0.1s +0.1s 0.0s
c2  | 3.400v  3.345v 3.471v | -0.014v +0.0s +0.0s +0.0s 0.0s
c3  | 3.473v  3.417v 3.546v | 0.059v +0.2s +0.2s +0.2s 0.0s
c4  | 3.414v  3.359v 3.486v | 0.000v +0.0s +0.0s +0.0s 0.0s
c5  | 3.463v  3.407v 3.536v | 0.049v +0.1s +0.1s +0.1s 0.0s
c6  | 3.403v  3.348v 3.475v | -0.011v +0.0s +0.0s +0.0s 0.0s
c7  | 3.175v  3.123v 3.242v | -0.239v -0.9s -0.9s -0.9s 0.0s
c8  | 3.161v  3.110v 3.228v | -0.252v -0.9s -0.9s -0.9s 0.0s
c9  | 3.146v  3.095v 3.212v | -0.268v -1.0s -1.0s -1.0s 0.0s
c10 | 3.143v  3.092v 3.210v | -0.270v -1.0s -1.0s -1.0s 0.0s
c11 | 3.156v  3.105v 3.223v | -0.258v -0.9s -0.9s -0.9s 0.0s
c12 | 3.162v  3.110v 3.229v | -0.252v -0.9s -0.9s -0.9s 0.0s
c13 | 3.443v  3.387v 3.515v | 0.029v +0.1s +0.1s +0.1s 0.0s
c14 | 3.403v  3.348v 3.475v | -0.011v +0.0s +0.0s +0.0s 0.0s
c15 | 3.475v  3.419v 3.549v | 0.061v +0.2s +0.2s +0.2s 0.0s
c16 | 3.401v  3.346v 3.473v | -0.013v +0.0s +0.0s +0.0s 0.0s
c17 | 3.463v  3.407v 3.536v | 0.049v +0.1s +0.1s +0.1s 0.0s
c18 | 3.399v  3.344v 3.471v | -0.015v +0.0s +0.0s +0.0s 0.0s
c19 | 3.146v  3.095v 3.212v | -0.268v -1.0s -1.0s -1.0s 0.0s
c20 | 3.152v  3.101v 3.219v | -0.262v -1.0s -1.0s -0.9s 0.1s
c21 | 3.156v  3.105v 3.223v | -0.258v -0.9s -0.9s -0.9s 0.0s
c22 | 3.163v  3.111v 3.229v | -0.251v -0.9s -0.9s -0.9s 0.0s
c23 | 3.162v  3.111v 3.229v | -0.251v -0.9s -0.9s -0.9s 0.0s
c24 | 3.146v  3.095v 3.212v | -0.268v -1.0s -1.0s -1.0s 0.0s
bmsc>
```

### show thermistors

The **show thermistors** command displays thermistor measurements.

```
bmsc> show th
ltc|  T1  |  T2  |  T3  |  T4  |  T5
----|-----|-----|-----|-----|-----
 1 | 16C  | 18C  |
 2 | 18C  | 17C  |
 3 | 19C  | 18C  |
 4 | 17C  | 17C  |
```

If the thermistor value shown is

blank	the thermistor is not enabled and not connected
.	the thermistor is not enabled but is connected
NC	the thermistor is enabled but not connected

SHORT	the thermistor is enabled but shorted
<0C	the thermistor is is reading a temperature < 0C otherwise, the thermistor temperature is given

After the thermistor temperature, there may be an “+”, which indicates whether the thermistor reading is greater than **thmax**.

### set

The **set** command sets the configurable parameters. If **set** is entered without parameters, **show config** will be displayed.

### set id

The **set id** command sets the bmsc id, a number from 1 to 4. By default the bmsc id is set to 1. Multiple bmsc devices may be used on a single CAN bus, typically to support packs with more than 96 cells. When the bmsc id is 1, then the command prompt is “bmsc>”. If the bmsc id is from 2 to 4, then the command prompt changes (to “bmsc2”, etc).

```
bmsc> set id 2
bmsc2>
```

### set canbr

The **set canbr** command sets the CAN data rate. This can be set to 125, 250 (the default), or 500.

```
bmsc> set canbr 500
changing CAN baud rate!
bmsc>
```

### set map

The **set map** is used to set the ltc to cell group mapping. The syntax of this command is

```
set map <ltc> <pack> <group>
```

where <ltc> is from 1 to 8, <pack> is from 1 to 8 and <group> is from 1 to 8.

### set hvc

The **set hvc** sets the High Voltage Cutoff.

```
bmsc> set hvc 3.5
bmsc> show config
id          : 1
lvc         : 2.800v
hvc         : 3.500v
lvcc        : n/a
hvcc        : n/a
bvc         : n/a
bvmin       : n/a
thmax       : 50C
options     : canterm (CAN termination resistor enabled)
```

**set hvcdelay**

The **set hvcdelay** sets the delay, in seconds, before reporting an HVC alert. The default is 0 seconds.

**set hvcc**

The **set hvcc** command sets the High Voltage Cutoff Clear threshold. When the BMS has detected a cell in HVC then it will not clear the HVC alert until the cell drops below the configured **hvcc** value.

If this value is configured to be 0, then the feature is disabled and its value will display as **n/a**.

**set lvc**

The **set lvc** command sets the Low Voltage Cutoff.

```
bmsc> set lvc 2.75
bmsc> show config
  id      : 1
  lvc     : 2.750v
  hvc     : 3.400v
  lvcc    : n/a
  hvcc    : n/a
  bvc     : n/a
  bvmin   : n/a
  thmax   : 50C
  options : canterm (CAN termination resistor enabled)
```

**set lvcdelay**

The **set lvcdelay** sets the delay, in seconds, before reporting an LVC alert. The default is 0 seconds.

**set lvcc**

The **set lvcc** command sets the Low Voltage Cutoff Clear threshold. When the BMS has detected a cell in LVC then it will not clear the LVC alert until the cell rises below the configured **lvcc** value.

If this value is configured to be 0, then the feature is disabled and its value will display as **n/a**.

**set bvc**

The **set bvc** command sets the Charge Balance Voltage Cutoff. When a cell voltage exceeds this threshold, a CAN message is sent to the EVCC to lower the charge current.

If this value is configured to be 0, then the feature is disabled and its value will display as **n/a**.

**set bvmin**

The **set bvmin** command sets the Cell Discharge Balancing Minimum voltage. This parameter is used during cell discharge balancing. A cell voltage must be higher than this threshold in order to allow discharging.

If this value is configured to be 0, then the feature is disabled and its value will display as **n/a**.

### set thmax

The **set thmax** command sets the Thermistor Maximum temperature, in Centigrade.

```
bmsc> set thmax 42
bmsc> show config
  id      : 1
  lvc     : 2.400v
  hvc     : 3.400v
  lvcc    : n/a
  hvcc    : n/a
  bvc     : n/a
  bvmin   : n/a
  thmax   : 42C
```

### set maxv

The **set maxv** command sets the maximum charge voltage for the integrated charge control feature. This set command enables the feature if v3.1.2 firmware is installed.

### set maxc

The **set maxc** command sets the maximum charge current for the integrated charge control feature.

### set termc

The **set termc** command sets the termination current for the integrated charge control feature.

### set termt

The **set termt** command sets the termination time for the integrated charge control feature.

### set flt\_maxv

The **set flt\_maxv** command sets the float charge voltage for the integrated charge control feature.

### set flt\_maxc

The **set flt\_maxc** command sets the float charge current for the integrated charge control feature.

## reset

### reset config

The **reset config** command sets all configuration parameters to the defaults.

Example:

```
bmsc> reset config
```

### reset stats

The **reset stats** command clears out the statistics. Note, this feature is not available for BMSC units with firmware supporting the integrated charge control feature (v3.1.2 or later).

Example:

```
bmsc> reset stats
```

## enable | disable

### enable | disable balance

The **enable balance** command activates cell discharge balancing. By default, cell discharge balancing is disabled.

The **disable balance** command disables cell discharge balancing.

### enable | disable canterm

The **enable canterm** command enables the programmable CAN termination resistor. By default, the termination resistor is NOT enabled.

The **disable canterm** command disables the programmable CAN termination resistor.

### enable | disable thermistor

The **enable thermistor** is used to enable thermistor operation. The command **disable thermistor** disables thermistor operation. The list of which thermistors are enabled can be determined by the command **show thermistor**.

If a thermistor is enabled then if the measured value of the thermistor is greater than **thmax** then the BMS will generate a **thermistor overtemp** alert.

To enable all thermistors that have been detected, use the command:

```
enable thermistor
```

To enable or disable individual thermistors, use the commands:

```
enable thermistor <ltc> <therm_number>
disable thermistor <ltc> <therm_number>
```

where **<ltc>** is from 1 to 8, and **<therm\_number>** is from 1 to 5.

### Examples:

```
bmsc> en th 1 1
bmsc> en th 1 2
bmsc> en th 2 1
bmsc> en th 2 2
bmsc> en th 3 1
bmsc> en th 3 2
bmsc> en th 4 1
bmsc> en th 4 2
bmsc> show thermistor
```

ltc	T1	T2	T3	T4	T5
1	16C	18C	---	---	---
2	18C	17C	---	---	---
3	19C	18C	---	---	---
4	17C	17C	---	---	---

**enable | disable c3100r**

The **enable c3100r** command enables the Curtis C3100r display.

The **disable c3100r** command disables the Curtis C3100r display.

**enable | disable loopoc**

The **enable loopoc** command configures the LP1 and LP2 outputs to be open collector.

The **disable loopoc** command configures the LP1 and LP2 outputs to be the contacts of a Form C relay.

**enable | disable loopclosedonflt**

The **enable loopclosedonflt** command reverses the logic of the loop contacts. That is, the contacts are open when everything is “good”, and “closed” if there is a fault. Note that this option has no effect if the loopoc open is enabled.

The **disable loopclosedonflt** command sets the logic of the loop contacts to be the default. That is, the contacts are closed when everything is “good”.

**lock**

The **lock** command is used to configure the discovered configuration. See text.

```
bmsc> lock
configuration locked
bmsc>
```

The **lock** command can also be used to temporarily disable BMS functionality when no cell harnesses are connected. This can be useful when testing the integrated charge control feature before connecting cell harnesses to the pack (firmware v3.1.2 or later).

**upgrade**

Firmware upgrade is performed by using the serial port and uses a special bootloader application program, the ds30 PIC bootloader. To begin the upgrade process, use the upgrade command. See the document *Serial Port Utilities Document* on the ThunderStruck Motors BMSC webpage for details, or link to the document directly below:

[https://www.thunderstruck-ev.com/images/companies/1/DD\\_SerialPortUtilities\\_v1.3.pdf](https://www.thunderstruck-ev.com/images/companies/1/DD_SerialPortUtilities_v1.3.pdf)

```
bmsc> upgrade
```

```
***
***                               Starting BMSC Upgrade                               ***
*** 1) Exit from the terminal application                                           ***
*** 2) Start the bootloader and download a new .hex file                          ***
*** 3) Restart the BMSC                                                           ***
```

The BMS performs basic statistic gathering and data reduction so that marginally performing cells can be identified. Statistics include high and low cell watermark data which can be used to track cell performance under load, and Standard Deviation, which measures pack balance.

## CAN Dictionary

```
// Dilithium Design CAN Message Definitions

#if 0
#define byte unsigned char;    // 8 bits
#define word unsigned int;    // 16 bits
#endif

//
// DD_BMS_STATUS_IND
//
// This message is sent once a second by the BMS to indicate BMS health.
//
#define DD_BMS_STATUS_IND      0x01dd0001

// BMS status flag definitions
#define BMS_FLAG_CELL_HVC     0x01 // at least one cell v is >HVC
#define BMS_FLAG_CELL_LVC     0x02 // at least one cell v is <LVC
#define BMS_FLAG_CELL_BVC     0x04 // at least one cell v is >BVC

// BMS flag definitions
#define BMS_FAULT_NOT_LOCKED  0x01 // configuration not locked
#define BMS_FAULT_CENSUS      0x02 // not all cells present
#define BMS_FAULT_OVERTEMP    0x04 // thermistor overtemp
#define BMS_FAULT_THERM_CENSUS 0x08 // not all thermistors present

typedef struct tDD_BMS_StatusInd
{
    byte      bBmsStatusFlags;
    byte      bBmscId;           // bmsc id (0..3)
    byte      bBmscFault;
    byte      bLtcFault;        // bit mask; 1 indicates error
    byte      bLtcCount;        // number of LTCs
} tDD_BMS_StatusInd;

//
// DD_BMSC_TH_STATUS_IND
//
// This message is sent once a second sent by the BMS to report the
// thermistor temperatures. Each report is for one LTC, and so it takes
// 8 seconds to report the temperatures of all thermistors
//
// LTCs are numbered from 0 to 7; bLtcIdx indicates the LTC
//
// Each LTC supports 5 thermistors, numbered 1 to 5
// bThEnabled is a bitmask that indicates which thermistors are enabled
// bThPresent is a bitmask that indicates which thermistors are detected
// bThTempInC is an array of 5 entries that contains the temperature, in C,
// for each thermistor
//
#define DD_BMSC_TH_STATUS_IND   0x01df0e00
#define DD_BMSC2_TH_STATUS_IND  0x01df0e10
#define DD_BMSC3_TH_STATUS_IND  0x01df0e20
#define DD_BMSC4_TH_STATUS_IND  0x01df0e30

typedef struct tDD_BMSCThStatus
{
```



```

    byte          bLtcIdx;           // 0 .. 7
    byte          bThEnabled;        // bitmask - 0b000xxxxx
    byte          bThPresent;        // bitmask - 0b000xxxxx
    byte          bThTemp[5];        // in Centigrade
} tDD_BMSCThStatus;

//
// DD_BMS_CVCUR_REQ
// DD_BMS_CVCUR_C1_TO_C4_RSP
// DD_BMS_CVCUR_C5_TO_C8_RSP
// DD_BMS_CVCUR_C9_TO_C12_RSP
//
// These messages report Current Cell Cell data from the BMSs
//
// The message ID is of the form 0xppppggbl
//
//      pppp  = 01de      - message prefix
//
//      gg    = 08        - request
//              = 09        - reply, cells 1 to 4
//              = 0a        - reply, cells 5 to 8
//              = 0b        - reply, cells 9 to 12
//
//      b     = 0 to 3 (for bmsc 1 to 4)
//
//      l     = 0 to 7 (for LTC 1 to 8)
//
// Example: To request the cell voltage data for bmsc1:ltc3,
//           the <bmsc id> is 0 and the <ltc id> is 2.
// The following id must be sent:
//
//           0x01de0802
//
// The BMS will then reply with three CAN messages with the following ids
//
//           0x01df0902
//           0x01df0a02
//           0x01df0b02
//
// The reply message will have a payload using the structure tDD_BMS_RawData,
// which contains four cell voltage values
//
//
#define DD_BMS_CVCUR_REQ                0x01de0800

#define DD_BMS_CVCUR_C1_TO_C4_RSP      0x01df0900
#define DD_BMS_CVCUR_C5_TO_C8_RSP      0x01df0a00
#define DD_BMS_CVCUR_C9_TO_C12_RSP     0x01df0b00

#define DD_BMSC_MASK                    0x0030 // 2 bits of bmsc idx (0 - 3)
#define DD_LTC_MASK                     0x0007 // 3 bits of ltc idx (0 - 7)

// tDD_BMS_RawData
typedef struct tDD_BMS_RawData
{
    word          wData[4];           // cell voltages in tenths of mv
} tDD_BMS_RawData;

```

## Hardware Revisions

### Revision 2.1

BMS v2.1 hardware uses Harting har-flexicon connectors for the system connections and Molex Milli Grid connectors for the thermistor connections, but is functionally the same and is software compatible with BMS v2.0 hardware.

### Revision 2.2

BMS v2.2 hardware adds the ability to configure the LP1 and LP2 outputs as open collector outputs. It also adds protection diodes to these outputs.

Previous versions of hardware may be updated to v2.2.x firmware but will not have access to these new hardware functions.

## Warranty and Support

The BMSC and BMSS are warranted to be free from defects in components and workmanship under normal use and service for a period of 1 year.

The product is intended for non-commercial use by hobbyists. The warranty does not cover defects arising from miswiring, abuse, negligence, accidents, or reverse engineering. Dilithium Design shall not be responsible for any incidental or consequential damages.

Dilithium Design reserve the right to make changes or improvements in design or manufacturing without assuming any obligation to change or improve products previously manufactured and / or sold.

For errors in this document, or comments about the product, contact [djmdilithium@gmail.com](mailto:djmdilithium@gmail.com)

## Document Revision

Rev 1.0	Feb 2017	In Review
Rev 1.1	Mar 2017	Initial Release
Rev 1.2	July 2018	Updates for v2.1 hardware
Rev 2.2	July 2019	Updates for v2.2 hardware
Rev 3.1	Jan 2024	Updates for Integrated Charging, firmware v3.1.2