

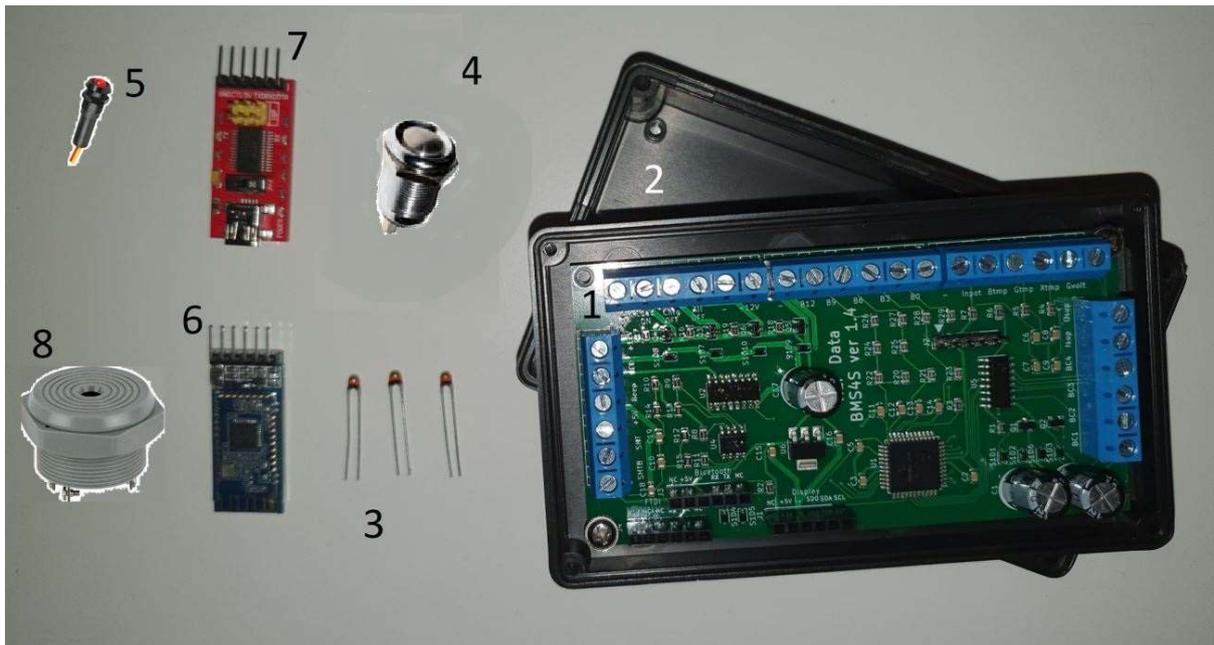
BMS4S user manual 1.4

Valid for software version 1.4.80

This manual is no longer updated, but the new functions added in new software versions that do not require another circuit board are always available also for this version of circuit board. The description of these functions can be read in the manual for the latest version of BMS4S.

BMS4S DIY kit:.....	1
Introduction	2
Normal usage	3
Basic configuration.....	3
Additional Commands.....	9
System Commands.....	13
Installation	14
Equipment connection description.....	16
Testing.....	19
4SBalancer.....	20
I/O description	22

BMS4S DIY kit:



The basic kit contains: 1: a BMS4S PCB, 2: and a box to mount it a in. Beside this you might also need 3: three temperature sensitive resistors, 4: a push-button for temporary charging level setting, 5: a 12 V LED for low SoC early warning, 6: a Bluetooth radio for communication with BMS4S, 7: an adapter for serial to USB connection to BMS4S, and 8: a buzzer/alarm.

There is also an automatic balancer available, 4Sbalancer:



However, we do not think this is a necessity. Most installation will work fine if the battery bank is manually balanced once or twice a year with a 3.6 V LiFePO4 charger.

Introduction

BMS4S is versatile battery management system that can be used in many different configurations. It is designed to supervise four LiFePO4 cells and automatically control that they are kept within specified boundaries. It can be used with default configuration straight out of the box, but can also easily be changed and used in a completely different setup. There are few limitations built into it.

You will find a description on how to install the BMS4S in chapter “Installation”, with more details on exactly how to connect all wires in chapter “Equipment connection description”. After installation you should test that it is working as expected. This is described in chapter “Testing”.

During normal usage there is not much you need to do, but there are some things that can be of interest for most users. This is described in chapter “Normal usage”.

How you use your cellular phone to connect to BMS4S is described in chapter “Normal usage”.

Available commands are described in three places: “Basic configuration”, “Additional commands” and “System commands”. All commands are built from one or two small letters, so in the Basic configuration chapter the commands can be derived from the printout, the capitals in each line is the command to use to change that setting. To give any commands that changes any settings you must first unlock the device with command `du=1234`. Default PIN is 1234, but you should change this after installation to your own one, command `dp=nnnn`. All system commands are protected by another level of protection since you can make BMS4S completely useless by a single bad system command. To unlock system commands, use command: `fu=1632`. Do not use any system command if you have not read the user manual carefully first. But there is always a way to get back to the original settings from factory, see command: `fr`.

And you can always get a command summary when connected to BMS4S with command: `h`.

Normal usage

BMS4S is switched on as long as it has power on terminal +12V. Normal usage is to take a look at the LED every now and then. If it is blinking twice, then there is less than 20% energy left in the batteries. If it is blinking once, it is less than 10%. You can change this to start the blinking at any level of charge. Command `sl=nn`.

Beside the LED you have the push-button. When you are on your way home and the boat will be unused for a while, then your batteries lifetime will increase if they are not left fully charged. Then press it until you see 3 long blinks on the LED. This will decrease the voltage when BMS4S will stop charging with 0.10 V, which equals approx. 80% SoC when using default settings. Or keep in pressed until 4 long blinks on the LED. This to achieve approx. 60% SoC (decreases the stop-voltage with 0.15 V).

When you press the push-button for a single long blink, three things will happen: the charging voltage will be reset, if an alarm is beeping it will be rest and you will wake up the Bluetooth radio (if you set it to switch it off instead of sleep, see command `wm`) that you can connect your cellular phone to. BMS4S has two ways of communication: a socket to plug in a FTDI adapter and a socket to plug in a bluetooth module of type HM-10 or equivalent. You connect with e.g. "Serial Bluetooth" on an android phone and with "BLE Terminal" on an I-phone. These are available for free on Google play and Apple appstore. When installed, scan, find and connect to the Bluetooth device.

If you give the **command p** (print) you will get a printout looking like this:

```
U=3.264 3.268 3.262 3.266 13.060 I=-1.335 SoC=35
```

This (U=...) means that all four cells have 3.26 V each, adding up to 13.06 V. Next, I=..., means that right now there are 1.335 A leaving the batteries (this only works if you have connected BMS4S to a current shunt, but that is not necessary for it to work). And SoC=... means that BMS4S estimates current state of charge to 35 % based on the last 15 minutes (can be changed with `te=nnn`). Using voltage and load gives a very rough estimate on higher SoC levels, but to use it as an early warning system when SoC is getting low works well. And it does not require regular synchronizations like a coulomb-counter based meter does.

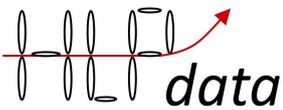
Another thing that is good to check every now and then is **command pe** (print extreme point log). See the description of it in chapter "Additional commands". It will give you information about how your batteries are performing.

And it may be useful to regularly press the button and see that the BMS4S responds with a blink to ensure that the BMS4S is operational. If you want automatic monitoring, install double BMS4S. Then they can monitor each other and alert if one of them stops responding.

Basic configuration

The starting point before changing anything is to take a look at current configuration. You do the with command `ps` (print setup):

```
WorkMode= 1  
BatterySize= 100 Ah  
BatteryUse= 30 Ah  
BatteryCharging= 250 A
```



TimeEstimate= 900 s
TimeBeep= 200 ms
TimeRelay= 300 ms
Timeprecharge= 0 s

VoltBalance= 3.500 V
VoltDifference= 0.000 V
BalanceInit= 0.000 Ah
BalanceAmount= 0 %

StartLedblinking= 20 %
VoltWarn= 3.100 V
VoltLow= 3.050 V
VoltVerylow= 3.000 V
VolswitchOnagain= 3.100 V
TimeLow= 15 s
VolswitchOnagain= 3.100 V

VoltAbsorb= 3.500 V
TimeAbsorb= 1200 s
VoltHigh= 3.600 V
TimeHigh= 5 s
TimeChargeoff= 14400 s
VoltResumecharge= 3.250 V
TimeWarning= 120 s

Templow1= 5 C
Tempwarn2= 50 C
Temphi3= 60 C

Temptarget4= 95 C
Temphi5= 100 C
Tempres6= 90 C
Limit-step1= 40
Limit-step2= 100
Limit-step3= 100

TimePrintout= 0 s
TimeMemlog= 315/600 s

The first block of lines describes some basic setup of BMS4S:

WorkMode= 1: this means that a current shunt is connected.

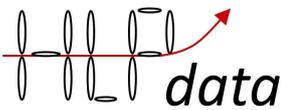
The value to set is calculated by adding the values for the following features:

0 workmode normal

+1 there is shunt connected

+2 measures 12V only

+4 supervised and supervising another BMS4S



- +8 xtemp is used to supervise a battery
- +16 xtemp is used to supervise an alternator
- +32 not used
- +64 Swedish CSV (comma separated values in printouts for import into Swedish excel)
- +128 not used
- +256 only p at tp= instead of both p and po, useful at certain test-cases
- +512 power off when high bat temp, normally only charging will be stopped
- +1024 all off port = charge on for bistable charging relay (no very low volt relay available)
- +2048 precharge on port BC1 (no automatic balancing possible), use if supervised mode

So, if you e.g. have connected a current shunt and have two BMS4S supervising each other you shall give the command `wm=5`.

BatterySize= 100 Ah: size of installed LFP bank. Change with `bs=nnn` as implied by the capital letters. **But before you can start changing anything you must unlock the device by command (device unlock) `du=1234`.** You should change this to your own PIN by command (device pin) `dp=nnnn`. **If you have forgotten your PIN**, you can set a new one within 30 seconds after you have switched on the power to BMS4S. You can remove the PIN by command `dp=0` (not recommended). **(note, all commands are given with small letters)**

BatteryUse= 30 Ah: You use 30 Ah every 24 hour which means that you get out of power after 3 days without charging. Used to estimate SoC if no current shunt is connected. Change with command `bu=nnn`. The same applies to all commands below, the capital letters in the printout imply the command to use when changing the setting. If BatteryUse is set too high, SoC estimates will also be too high.

BatteryCharging= 50 A: the most common charging method will charge with 50 A. Used to estimate unbalance between cells if no current shunt is connected. If you are using automatic balancing using coulomb-counting, you should also install a shunt to ensure that the unbalance-estimates are as good as possible (see chapter "Equipment connection description" below).

TimeEstimate= 900 s: The Soc estimate is based on the last 900 seconds (15 minutes).

TimeBeep= 300 ms: each beep will be 300 ms long. If you have a really loud beeper you might want to shorten this to achieve a bearable sound level.

TimeRelay= 300 ms: when the bi-stable relay is operated it is done with a 300 ms long pulse

TimeprecharGe= 0 s: if you have equipment with high start-up current you might want to pre-charge them before operating the relay. If you set this timer between 1 and 10 s you will get a pulse to use for a small relay with a resistor parallel to the real relay that will soft-start the equipment. This pulse comes on either the port "Osup" or "BC1", see command `wm=nnnn`.

The second block of lines describes how balancing will be handled.

VoltBalance= 3.500 V: There are two ways to use the connected 4sbalancer (sold separately). Both uses this value as the starting point when the balancing procedures are initiated. This value shall normally not be set below 3.4 V. If you do not have a 4sbalancer these settings will be used to calculate how big the unbalance is.

VoltDifference= 0.000 V: This is the first and simplest balancing alternative. If this value is set (recommended value: 0.050 V) then when the first cell has passed the BalanceVolt-value (during charging) and the voltage difference between two cells exceeds the VoltDifference-value the balancing will start and proceed as long as the BalanceVolt-value is exceeded and the unbalance remains. But since the balancing current is low (0.2 A) and that balancing is done only while charging, this method only works when the battery bank is reasonable well matched (all cells have close capacity values). Otherwise, the second method is recommended. (Or you can replace the resistors in the 4sbalancer with ones with lower resistance (the ones installed are on 33 ohm each) to increase the current to be able to handle larger unbalances. But then ensure that the produced heat can be ventilated away)

BalanceInit= 0.000 Ah: This is the second balancing alternative. When the first cell passes BalanceVolt-value during charging (and VoltDifference is set to 0.000) coulomb counting starts and continues until the last cell passes it. If the difference between the best and the worse cell then exceeds the BalanceInit-value (recommended value: 0.5% of the total battery capacity, i.e. 0.500 Ah for a 100 Ah battery bank) the balancing will start and continue until the calculated balancing value is balanced away. It is recommended that you have installed a way to measure the current if you use this method. See chapter “Equipment connection description” how to do that. Note: VoltDifference must be set to 0 (vd=0) to activate this balancing alternative.

You can see the coulomb counters with the sort Coulomb (As) with command po, and all calculated results presented in Ah with command pe. You can use this information to manually balance the cells with a 3.6 V charger.

BalanceAmount= 0 %: when balancing starts, it will aim to reduce the difference with the percentage stated here. The recommended start value 25% is to ensure that it does not overshoot the target. This is especially important if no shunt is connected, since it then bases the coulomb counting on what value is set for BatteryCharging. If you find multiple similar entries in the “extreme point log” (command pe) before an unbalance is cured, you can increase this with e.g. 25% to speed up the process.

The third block of lines describes how and when a low voltage event shall be handled, i.e. when shall the loads be disconnected.

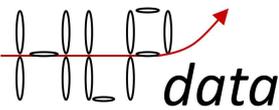
StartLedblinking= 20 %: when SoC gets below 20% the LED will start blinking as an early warning, 2 blink for 10-19 % and 1 blink for 0-9 %.

VoltWarn= 3.100 V: this is the first level of protection. At 3.100 V the buzzer/alarm will start beeping “low voltage”. It is now time to start the engine and start charging to prevent that any loads get disconnected.

VoltLow= 3.050 V: this is when the “not necessary” loads will be disconnected via the bi-stable relay.

VoltVerylow= 3.000 V: this is when all loads will be disconnected. You can use this e.g. if you have a second bi-stable relay isolating the batteries completely. If BMS4S is not disconnected it will go into low power mode and only do the most important tasks in a slow paste, like switching on the system again once charging has brought the voltage up to VoltswitchOnagain.

TimeLow= 15 s: the above voltages must be below the values for 15 seconds before the action is taken. If you use high power things like thrusters and starters for longer continues time than this you might want to extend this time, command tl=nn.



VoltswitchOnagain= 3.100 V: when the voltage has reached this level after a low-voltage event the system will go back to normal again.

The fourth block of lines describes when charging shall be discontinued.

VoltAbsorb= 3.500 V: Normally LFP cells are fully charged at 3.6-3.65 V. Some chargers miss settings adopted for LFP, so they will not manage to get the voltage that high. Then you can set an “absorption” level that it can reach, and let it stay there for a time. Here this time starts at 3.5 V.

TimeAbsorb= 1200 s: and this timer will keep the charger on for 1200 s (20 min) before charging is stopped

VoltHigh= 3.600 V: All charging sources that can achieve 3.6 V will be stopped at that voltage. 3.6 V corresponds to approximately 90 % SoC when charging is done with 0.4 C, and is a good compromise between battery usage and battery life length.

TimeHigh= 5 s: but the first cell must exceed the set voltage in more than 5 s before the action is taken.

TimeChargeoff= 280/14400 s: the total time it will be switched off is 14400 s (4 hours) and so far it has been switched off for 280 s. If it is set to 65500 it means it will never switch on charging, then it is only VoltResumecharg that will do that.

VoltResumecharg= 3.200 V: if the voltage drops down to this level, charging will resume even if TimeChargeoff has not elapsed.

TimeWarning= 120 s: if the voltage has not gone down below the set high voltage 120 s after charging has been switched off, BMS4S will start beeping “high voltage”.

The fifth block of lines describes battery temperature limits.

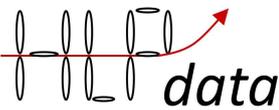
TempLow1= 5 C: if charging is started with battery temperature lower than this a “low temperature charging” will start beeping, and charging will be stopped for a preset time (see command tc). If you do not have any temperature sensors installed on the batteries, their temperature will be reported as 20 degrees.

Tempwarn2= 50 C: if battery temperature exceeds 50 degrees Celsius a “high battery temperature” will start beeping.

TempHi3= 60 C: if battery temperature exceeds 60 degrees Celsius a charging will be switched off for a preset time (see command tc).

The sixth block of lines describes when and what to do when the alternators temperature is getting high.

TempTarget4= 95 C: when the alternator exceeds 95 degrees Celsius, BMS4S will regulate down charging current and keep the temperature around 95. However, the temperature is a slow feedback on how much power the alternator generates, so the temperature will fluctuate a bit around the set value. Therefore, it is important to set a max-value (**TempHi5**) where charging will pause if the



regulation is too slow. This regulation will only work if your alternator has a voltage sense cable, and you have installed a temperature sensor on the alternator.

Temphi5= 100 C: when the alternator temperature exceeds 100 degrees Celsius, all charging will pause. This will only work if you have installed a temperature sensor on the alternator.

Tempres6= 90 C: when the alternator temperature has come down to 90 degrees Celsius, charging will resume if it has been paused.

Limit-step1= 40: Regulation is performed by a PID-regulator (with the integration disabled since it is of no interest). This is the delay time between adjustment-calculations in seconds. It takes some time before an adjustment of current influences the temperature, so too short interval will always generate oscillations, and too long interval will make reaction slow.

Limit-step2= 100: This is the amplification of the P value (the temperature) feedback in percent when the temperature differs more than ± 2 degrees from set value. Below this the feedback is always 0 (the temperature is accepted)

Limit-step3= 100: This is the amplification of the D value (temperature difference) feedback in percent. This is how the regulator stabilizes the temperature when it is within accepted temperature range.

If the temperature oscillates too much, try increasing Limit-step1.

If the regulation is too slow so the temperature hits Temphi5 or Tempres6, try decreasing Limit-step1 or increasing Limit-step2

And the last block contains features useful to understand how your system is performing.

TimePrintout= 0 s: if e.g. command `tp=5` is given, then the command `p` and `po` will automatically be given every 5 s. This is useful when testing, and will prevent that the Bluetooth interface is powered off. So, if you want the Bluetooth interface to stay on all the time, `tp=65500` can be given. This will however increase the power consumption with around 10 mA.

TimeMemlog= 315/600 s: this means that voltage and current will be logged every 600 s (10 min). The log is started by command `tm` and can be printed with command `pm`. 315 is the time since last log entry was saved.

A command summary and description of blinks and beeps are always available with command `h`:

<code>h</code> help	<code>dp=</code> set device pin
<code>l</code> print settings	<code>du=</code> device unlock
<code>p</code> print voltage	<code>fr</code> factory reset
<code>ba=</code> balance amount (%)	<code>fu=</code> factory settings unlock
<code>bi=</code> balance initiate diff (A/100)	<code>pd</code> print debug
<code>bp=</code> button press	<code>pe</code> print extreme-point log
<code>bs=</code> battery size (Ah)	<code>pm</code> print memory log
<code>bu=</code> battery usage (Ah)	<code>po</code> print other info
<code>bt=</code> bluetooth 0=off, 1=sleep, 2=on	<code>ps</code> print settings
<code>ce</code> clear extreme points log	<code>pv</code> print version
<code>cm</code> clear memory log	<code>rb</code> reboot
<code>dl</code> device lock	<code>rp</code> reprogram

sl= set led blink-level (%)	s1-4= set battery volt
wm= set work mode	sc= set current (A)
t1= temp low charging (C)	sf= save factory data
t2= temp high battery warn (C)	sm= set missing current
t3= temp high battery (C)	ss= set float size
t4= temp target (C)	sv= set verbose
t5= temp high pause charging (C)	
t6= temp resume charging (C)	wm=0 workmode normal
ta= time absorb (s)	+1 shunt connected
tb= time beep (ms)	+2 12V only
tc= time of charging pause (s)	+4 supervised
te= time for soc estimate (s)	+8 x is bat temp
th= time high (s)	+16 x is alt temp
tl= time low (s)	+32 not used
tm= time memory log (s)	+64 Swedish CSV
tp= time print volt (s)	+128 not used
tr= time relay (ms)	+256 only p at tp=
tw= time before warning beep (s)	+512 power off when high bat temp
va= volt absorb (V)	+1024 all off port = charge on
vb= volt balance (V)	+2048 precharge on port BC1
vh= volt high (V)	
vo= volt power on again (V)	1 blink SoC 0-9
vr= volt resume charging (V)	2 blink SoC 10-19
vw= volt warning (V)	...
vl= volt low (V)	10 blink SoC 90-99
vv= volt very low (V)	
AT AT command	2 beep low temp charging
	3 beep high battery temp
a1-4= adjust battery volt	4 beep low voltage
l1= initial search step for charge limit	5 beep high voltage
l2= second search step for charge limit	6 beep device failure
l3= third search step for charge limit	

Additional Commands

What you normally do is described chapter “normal usage” and chapter “basic configuration”, so you have the explanation of the important commands there. Here you have a few more commands that can be of use.

AT AT-command

This is the exception to the statement “all commands are written with small letters”. The Bluetooth radio this device was designed for entered AT-mode (mode where it accepts commands) when no device is connected over the radio. So, if you want to change something in the Bluetooth-radio you can give it e.g. a command like this: AT+NAMEMyBMS. As soon as this is sent you will be disconnected which will get the Bluetooth device into AT mode. Half a second after you have given the command, BMS4S will sent it to the radio. When this is done you will see a blink on the LED. After that you have 10 seconds to

reconnect to get the response that the radio gave to your command. Unfortunately, not all bluetooth devices work exactly the same. If this does not work, check the manual for the bluetooth device you have and see if there are any other way to change its settings. But you do not have to change anything, it will work fine anyway.

bp Button Press

Instead of pressing the button to do something, with this command you can do it remotely.
Example: bp=1

ce Clear Extremepointlog

will clear all entries from the log.

cm Clear Memorylog

will clear all entries from the log. You can add a text with 12 characters e.g. to state when the logging was started. Example:
cm=24/12-09:30

p Print

will produce a printout of cell-voltage, total voltage, current and estimated SoC. Example:
U=3.283 3.283 3.283 3.284 13.133 I=0.000 SoC=41

pe Print Extremepointlog

Command pe will give a printout like this:

```
HVE , 3.599 , 3.600 , 3.598 , 3.599 , 14.396 , 77.130 ,
HVA , 3.517 , 3.520 , 3.518 , 3.520 , 14.074 , 14.970 ,
LVE , 2.997 , 2.996 , 2.996 , 2.997 , 11.986 , -1.663 ,
UBC , 3.563 , 3.564 , 3.563 , 3.555 , 14.264 , 75.622 , 0.720 , 0.720 , 0.810 , 0.010 ,
UBI , 3.563 , 3.564 , 3.563 , 3.565 , 14.254 , 75.622 , 0.720 , 0.720 , 0.810 , 0.010
UBE , 3.277 , 3.277 , 3.277 , 3.277 , 13.109 , -1.204 ,
VBI , 3.480 , 3.500 , 3.499 , 3.497 , 13.976 , 75.622,
```

This time the log contained six entries, can store around 50 entries. The log is cleared by **command ce**. The first four numbers in all lines are the voltage of each cell when the event happened. The fifth is the total voltage, and the sixth is the current at the event.

HVE means “high voltage event”, HVA “high voltage absorption event” (timeout has happened in VoltAbsorbtion phase, see command va), LVE “low voltage event”.

UBC means “unbalance calculation” and the four last numbers are the amount of energy that were charged into each cell at the latest cell balance measurement (done by coulomb counting), with the sort Ah. This counting starts when the voltage passes VoltBalance . So right now, there is an unbalance of 0.8 Ah (0.810-0.010) between cell 3 and cell 4, where cell 3 is the best charged one. UBI means “unbalance calculation and balancing initiated” (balancing will only work if a balancing board has been installed). UBE then means that the initiated balancing has ended. If an UBE is missing after an UBI then the system has been switched off before the balancing ended, which is nothing to worry about. If there are enough remining unbalance the process will restart at the next unbalance-calculation. If there are multiple UBI in sequence with the same order of unbalance this indicates that you

can increase how much of the unbalance you try to remove each time. Then you can increase the BalanceAmount (ba=nn, see “Basic configuration”) with steps around 25% each time.

VBI means that voltage based balancing has been initiated. It will stop when the voltage is back under VoltBalance.

You can use this information to manually balance the cells with a 3.6 V LiFePO4 charger. Here you can charge cell 4 with a 5 A 3.6 V charger for 10 minutes to even out the unbalance. One reason not to have an automatic balancer built-in is that if it breaks down, it can start discharging a cell uncontrolled. The second reason is that it takes time to create unbalance, and therefore **doing manual rebalance by charging all four cells individually one or two times a year with the 3.6 V charger is normally sufficient since BMS4S will always keep all cells within set boundaries.**

But there is an automatic balancer available that is controlled by BMS4S. The HLPdata 4Sbalancer. It is built very robust with relays to control balancing. The reason is that if a relay breaks down it normally does so in a disconnected state, which will not harm the batteries. And the risk that it gets stuck in connected state is virtually zero with the low current that flows through it.

pm Print Memorylog

will produce a printout of the memory-log started with command tm. The log can contain 288 entries. Example of log with two entries:

```
interval=600
3.249 , 3.247 , 3.253 , 3.248 , 12.997 , 0.000 ,
3.249 , 3.247 , 3.253 , 3.249 , 12.997 , -4.130 ,
```

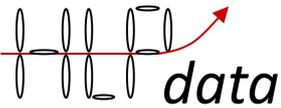
Each line contains voltage on cell 1 to cell 4, system voltage, and current at that time. This log is cleared when BMS4S is switched off, or by command: cm (Clear Memory log).

po Print Other info

will produce a printout of some other information that can be of interest:

volt high adjust=0.000	there is a push-button you can adjust down the end voltage with 0.05, 0,10, 0.15, 0,20 or 2 V. Press the push-button until 2 long blinks to set it to 0,05, 3 blinks for 0,1, 4 blinks for 0,15, 5blinks for 0,2 and 6 blinks for 2 V (which will always switch off charging). Keep it pressed until long blinking stops to reset it to normal. It will also be reset when BMS4S is switched off. This can be useful e.g. if you motor home and want to leave the boat with less than fully charged batteries.
charge is on	Charging is allowed (normal state)
load is on	Power consumption is not limited (normal state)
battery temp=20	Will always show 20 if no sensor is installed
x temp=20	Will always show 20 if no sensor is installed
alternator temp=20	Will always show 20 if no sensor is installed

charge-limit-state=0	there are several reasons charging is limited. When this is 1 to 4 it is limited due to the temperature of the alternator, when 5 it is stopped for a preset time since the battery is either fully charged, or battery temperature is too low or too high for charging. 1 to 4 will of course only happen if you have installed a temperature sensor. . Only printed if verbose not is zero (command sv=1).
push-level=0	A value showing representation of voltage on terminal Gvolt. Lower value means higher voltage. . Only printed if verbose not is zero (command sv=1).
coulomb counters=0 0 0 0	Result of latest coulomb counting, sort is coulomb (As). . Only printed if verbose not is zero (command sv=1).
balancing: off	Will list all cells currently being balanced (or none, as here)
buzzer: 0	State of the buzzer, silent right now
led-blink: 0	State of the LED, SoC level ok right now
pv	Print Version
	will produce a printout of the model and version of the installed software.
pd	Print Debug
	will produce a printout of the device internal memory. Used for debugging only.
rb	ReBoot
	Will force the device into a fault situation that will make the system supervision reboot the device.
sc	SetCurrent
	Used to adjust the values used to calculate actual current. Must be set if you are using a shunt different from the predefined one (50 mv = 500 A), and it doesn't hurt to do it anyway, it will improve the result. First start by stating that there is a shunt connected, command wm=nn, see command wm for what nn shall be). Then definine the zero point: ensure no load or charging sources are connected, and give command sc=0. Then connect a load of known size and give command sc=-15.5 (if the load is 15.5 A). And lastly connect a known charging source and give command sc=20.0 (if the source delivers 20 A).
sm	SetMissingcurrent
	Depending on how far from the battery your shunt is installed some of the first charging current will be missed. I you want a more exact current measurement you need to find out when BMS4S first detects charging, and enter this value here measured in Amps. The default setting will normally work fine, so this is not necessary to do.
ss	SetfloatSize
	Set how many decimal digits you want presented. The resolution of voltage measurements is 0,004 V, so more than 3 digits make no sense, two is normally enough.
sv	SetVerbose
	Set level of information at printouts. Used for debugging.
tm	TimeMemorylog



sets the time between each sample in the log. The log can contain more than 250 entries which is enough for logging every 5 min (300 s) for almost 24 hours.

System Commands

Here are some commands that you should not use unless you know exactly what you are doing. If you have done so anyway, look at command **fr** below to understand how to resolve the problem.

a1(-4) Adjust voltage for cell 1-4

If you are sure that BMS4S reports a faulty voltage on a cell you can adjust the value-calculation with this command.

Example: a4=-0,005

fr Factory Reset

When you have done too much changes and want to come back to a known status, this will reset the system to original settings as it was when it was shipped from the factory.

Example: fr

rp ReProgram

If there are some new functions released, this is used to update the BMS4S. Exact usage is supplied together with the update package.

s1 (-4) Set voltage for cell 1-4

If you are sure that BMS4S reports a faulty voltage on a cell you can by this command set the correct voltage of the actual cell.

Example s4=3.233

You can as an alternative set the accumulated value of that cell, the result will be the same

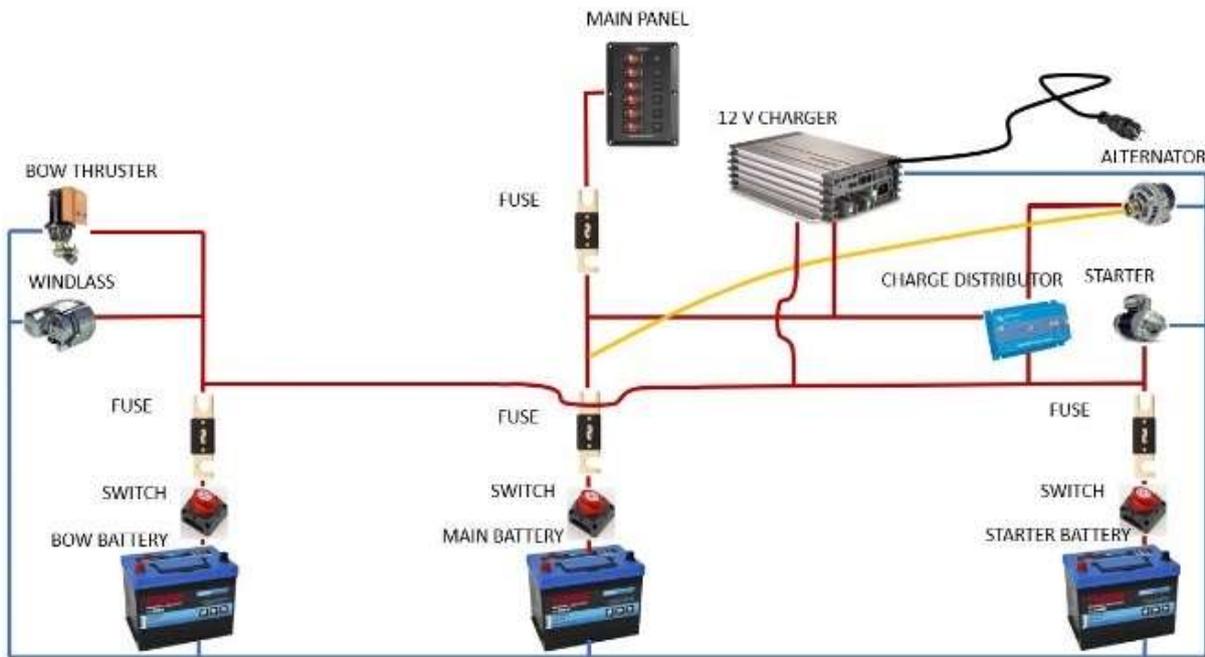
Example s4=13.3

sa Set All voltage

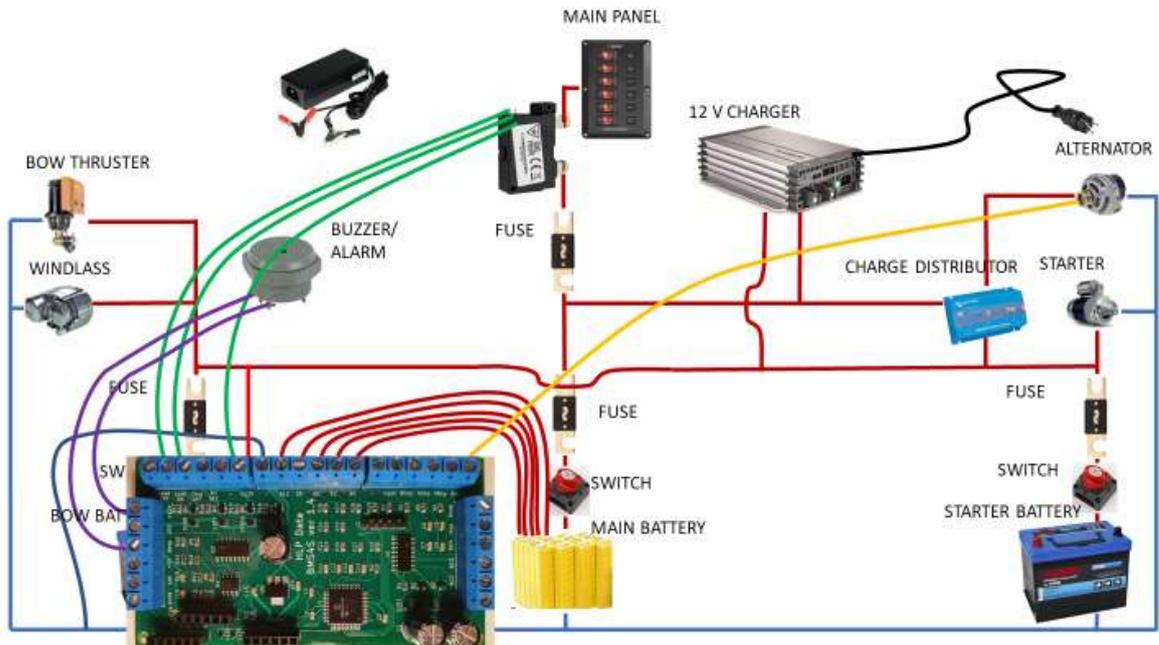
Can be used to set the voltage on all cells if they are perfectly aligned.

Example sa=3.233

Installation



When you start the electrical system looks something like above. You often have a starter battery and a house battery. These are kept separate so that the starter battery is not used by the boat's consumers, and they are charged via a charging distributor or a relay that switches on when the engine is running. If you have bow thruster then you often also have a battery for it, placed in the bow of the boat. All batteries, of course, have switches and fuses installed.

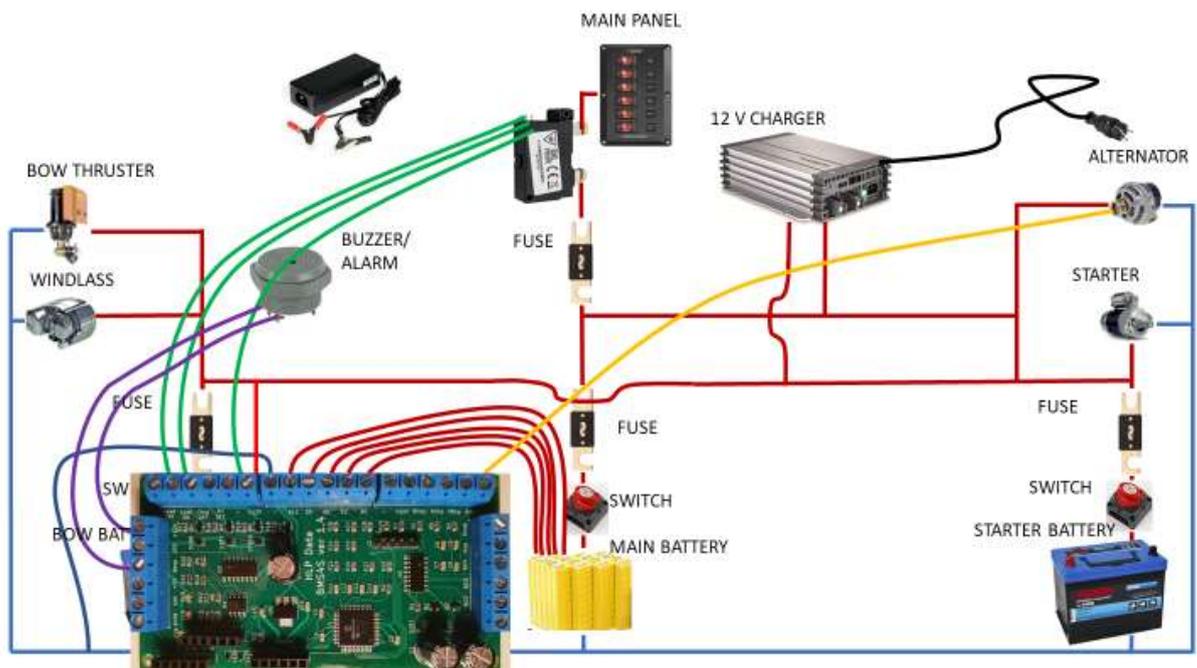


The simplest way is to just replace the main battery with LFP cells, and connect them to BMS4S. You also connect a buzzer/alarm to it. Then you move the voltage sense (often yellow) cable and connect it to BMS4S. You also have to connect a bi-stable relay between the new main-battery and to BMS4S. Then at last you connect the power to supply BMS4S from after the main switch (but before the relay) so it is switched off when all other electricity is switched off.

There are two more items we think you should connect: a push-button and a led. Details on this and where the wires are connected is described in Equipment connection description below.

The disadvantage with this setup is that the lead batteries only get charged when the engine is running, and during most of that time the LFP bank keep the voltage below 13.8 V, which is not enough to get lead batteries fully charged.

Therefore, we recommend one more change: remove the power distributor and connect the three cables together you get a setup looking like this:

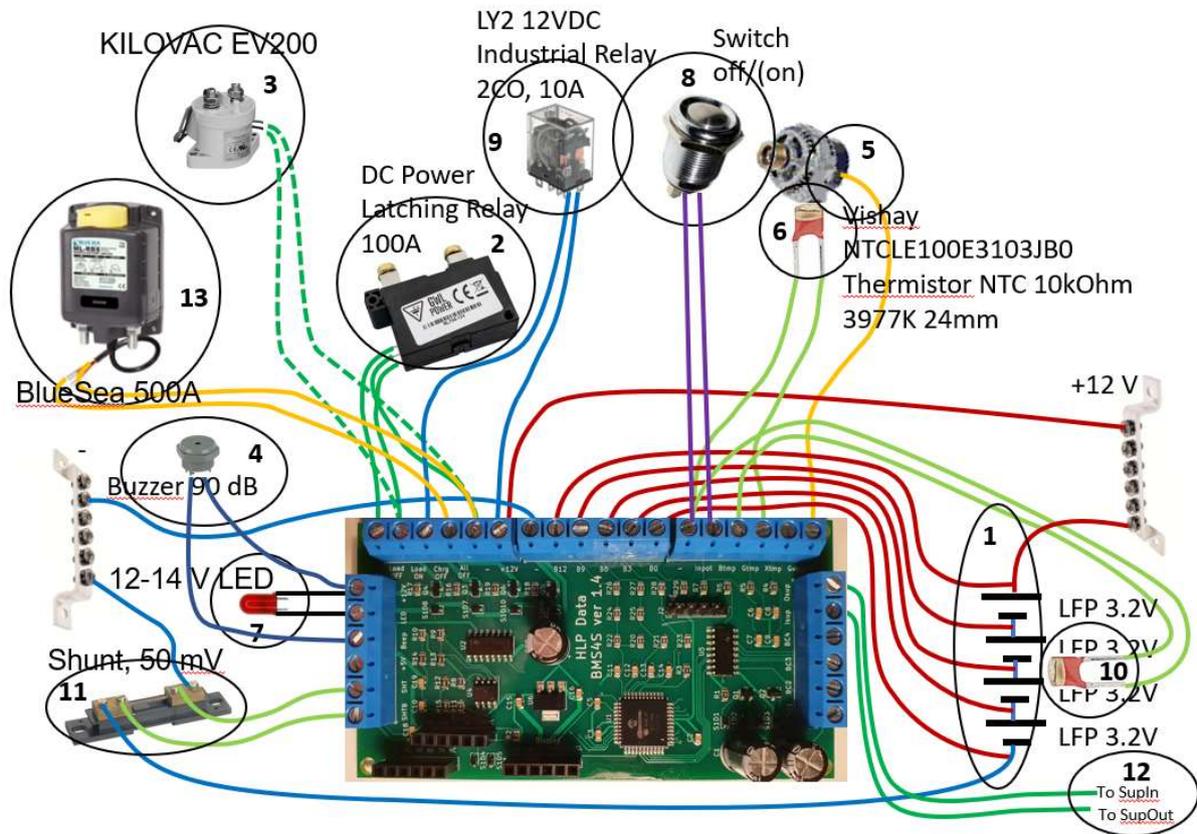


Now the lead batteries have become backup for security and is not used during normal operation. It is then enough to charge them every now and then, which can be done by simply switch them on. They will then get some small current from the LFP that normally are giving 13.2 V, and of course from the alternator when it is running. But since they are not used, they will always be available as backup and fully charge. So how do you actually do this installation.

Here is a step-by-step instruction, but first a warning: NEVER CONNECT +12 V BEFORE – AND B0 ARE CONNECTED. If you want to test the BMS4S before installation, connect – and B0 to battery minus first, and then +12 and B12 to battery plus (and then command wm=2 to tell BMS4S that only 12 V is connected). More details on each connection can be found in the next chapter.

1. Plan where to put the BMS4S, the relay, the buzzer/alarm, the led for low SoC warning and the push-button for simple adjustment of charge level and to wake up the Bluetooth interface.
2. Start with connecting the relay on the cable between the man panel and the LFP cells.
3. Connect the relay to Load OFF and Load ON on the BMS4S, and to - (minus, ground).
4. Connect the buzzer/alarm to 12V. Connect the other terminal Beep on the BMS4S.
5. Connect the 12 V led to BMS4S.
6. Connect the push-button between terminal Inpot and – (minus, ground)
7. Move the (often yellow) voltage sense cable from the generator to the Gvolt terminal on BMS4S.
8. Prepare five cables between the LFP cells and the BMS4S, and press on connectors on the battery side. Mark the cables on the battery side with labels: B12, B9, B6, B3, B0.
9. Connect the cables on the BMS4S side first. Make sure the B12V cable is connected into the B12 terminal etc.
10. Connect the cables to the LFP cells. Make sure you place the cables on the right cell. Start with B0.
11. Now it is just to connect power to BMS4S via the terminals +12V and – (minus/ground)
12. Now move the cables on the current distributor so they all are sitting on the same terminal (unless you intend to keep a separate starter-battery).
13. Test the installation as described in chapter Testing below.
14. Start the engine and run it for a while. Test with a wet finger that the alternator does not exceed 100 degrees. If it does it does probably not contain any protection against overheating. See bullet 6 in chapter “Equipment connection description” on how to mitigate that.
15. To maximize the capacity and life-length of your LFP cells you shall always charge them fully when they have been unused for a while. Do that now with the 3.6 V LFP charger, one cell at the time. (But do not leave them fully charged if they will be unused for a while. Discharge them then to below 80% SoC to maximize life-length.)
16. Now you are ready to go. Enjoy a life with less worries about lack of electricity.

Equipment connection description



Here is a picture showing how equipment shall be connected to BMS4S. Please note that there are no switches or fuses shown in the picture. All wires leaving a battery should have a switch and must be properly fused close to the battery if there is a slightest risk that it can ever create a shortage or overload, but that is outside the scope of this manual and you will have to find descriptions on how to do this elsewhere. So, let us instead discuss how the equipment in the picture shall be connected to BMS4S. We'll take them one by one, but first a warning: NEVER CONNECT +12 V BEFORE – AND B0 ARE CONNECTED. If you want to test the BMS4S before installation, connect – and B0 to battery minus, and then +12 and B12 to battery plus (and then command wm=2 to tell BMS4S that only 12 V is connected).

- 1 You need five cables between BMS and the LFP cells. Mark them in one end with labels: B12, B9, B6, B3 and B0. Start connecting the B0 wire in B0 on BMS4S and then connect it to the battery minus terminal. Continue with B3 which shall be connected to + side of the first cell. B6 to the second cells + terminal. B9 to the third, and B12 to the fourth cells + terminal. Check each wire one extra time before connecting it to the battery.
- 2 Open up the wires between the battery and the main panel and connect the latching relay so that it can disconnect all user equipment when a low voltage situation occurs. Then connect the wires between the relay and BMS as in the picture, i.e. the wire that switches off the relay to the Load OFF terminal, and the one switching it on to the Load On terminal. The common wire is connected to ground. The control terminals for driving the relays on BMS4S are of MOSFET type, and can drive 11 A momentarily and 3 A continuously. You can of course use high power relays like Bluesea 500 A if you have equipment with a high demand for power.

- 3 If you want to use a monostable relay, like KILOWAC EV200, you connect it to the Load on terminal. For mono-stable relays you must also change the time the relay is pulled to 0, which means it is pulled all the time, command: $tr=0$. This also means that the Load off terminal will always be in the opposite position of the Load on terminal. Mono-stable relays will use current all the time it is connected, so bi-stable relays have a definite advantage.
- 4 You can use any 12 V buzzer/alarm to get audio signals when something must be done, as long as it uses less than 100 mA. You simply power it with 12 V and connect the minus wire to the terminal marked Beep. If you want to use higher power equipment you simply connect a normally open relay to the port and let it deliver the power.
- 5 Many modern alternators have a voltage sensing (often yellow) cable used to compensate for the voltage drop in the cables between the alternator and the batteries. If your alternator is missing such a cable it is probably a low power one, and it might be a good idea to replace it to get improved charging since LFP batteries can recharge much faster than lead-batteries does. The voltage sensing cable is simply moved from where it was connected before to the terminal on BMS4S marked Gvolt. But if you still want to use the existing alternator without a voltage sensing cable you have to open it up and identify the two wires that is feeding the rotor with power. They will always go to the two carbon brushes, sometimes built in to the regulator. You have to cut up one of these and extend the two resulting wires outside of the alternator, and connect them to a relay that is of type normally connected (NC). Then connect the relays control wires to terminal Charge OFF and – (ground) on the BMS4S. Now the relay will break up the power feeding of the rotor when BMS4S activates it, and the alternator will stop producing current. This way is also how you connect relays switching off charging from other sources like solarpanels, wind generators etc.
- 6 Many modern alternators are also protected against overheating. A simple test is to run it loaded for a while and then touch it with a wet finger. You hear directly if it is above 100 Celsius since the water then instantly boils. BMS4S can protect the alternator if it doesn't do it itself. Place the thermistor so the it touches the alternator and connect it as shown in the picture, between the Gtemp terminal and ground (minus). Dependent on where you place it you might have to adjust the temperature values: command $t4=nn$, $t5=nn$ and $t6=nn$.
- 7 The LED must be of 14V or 12 V type if you connect it as in the picture. Otherwise, you must insert a 1 kOhm resistor in series with the diod. Note that the LED might be sensitive to the polarity so you connect the right leg toward +12V. As an alternative you can feed the diod from the +5 V terminal, then with a 470 Ohm resistor in serie.
- 8 The push-button is connected between the inpot terminal and – (minus/ground). It is used to temporarily change the settings for charging, and to wake up a bluetooth radio that is switched off.
- 9 If you want to automatically switch off the power to the fixed installed charger when the battery is fully charged you can use a standard industrial relay with sufficient power capability and connect it to terminal Charge OFF and ground (minus).
- 10 If you want to supervise the batteries temperature you connect a thermistor between ports – (minus/ground) and Btemp on BMS4S and place it at the LFP cells.
- 11 If you have a low-side 50 mV shunt for 500 A connected on your minus side, you can connect BMS4S to it as in the picture. Connect the side closest to the battery and

terminal SHTB, and the other at SHT. When it is connected you must inform BMS4S about it, command `wm=1`. You can always calibrate it to adjust for other shunt values. If you have no shunt, connect it to the two sides of the first cable on the battery's minus side if the expected voltage over it never will exceed 100 mV. Or push two needles into the wire with 0.3 m in between if it is a 50 mm² wire, or 0.2 m if it is a 30 mm² wire. Start with an idle system (no current) and zeroize it, command: `sc=0`. Then start charging with a known current, e.g. 20 A, then give the command: `sc=20`. And lastly switch on a known load, e.g. 10 A and give command: `sc=-10`. The current measurement is not high resolution, and is mainly used to roughly estimate SoC and the amount of unbalance between the cells (by coulomb counting during charging).

- 12 If you want redundancy (if you have a very large/expensive LFP bank) you can install a second BMS4S, either exactly the same way as the first one, or simplified by reducing some of the duplicated equipment. The simplest working configuration is to only connect +12V and a wire to the alarm/buzzer so that it can notify you if something fails. Then you connect the two together by two wires going from Osup on one BMS4S to Isup on the other, and vice versa. Then change working mode on both to "normal working mode + is supervised by and supervising a second BMS4S". Command `wm=4` (5 if you have a shunt installed). Make sure the two BMS4S do not share the same fuse to avoid a single source of failure. They are now supervising each other, and if one fails, the other one will start beeping "device failure" while it still is supervising the LFP bank.
- 13 If you intend to keep the batteries connected all the time, also when leaving the boat for long times, it can be wise to install a bi-stable relay that isolates the battery completely in case of a low voltage incident. It shall be connected to terminal All OFF, and BMS4S sends out a 1 s pulse when the voltage of the first cell goes below 2.8 V (can be changed with `vv=n.nnn`). It is recommended that this relay switches off ALL consumers. You might have to connect a capacitor in parallel with the relay of 100-300 μ F to ensure that it finishes switching after the power is cut. This kind of high-current relay can of course also be used as relay in point 2 above if you have equipment like high current inverters. For thrusters and winches it is enough if the control buttons are powered from behind the low-voltage relay, they use very little power.

Testing

Switch on the power and connect to BMS4S with your phone or PC over Bluetooth. Start with checking that all voltages are reported correctly, command `p`. Then take a look at the settings to see if there are anything you want to change, command `ps`. Now it is time to test the system. Firstly, we test that the low voltage protection works:

1. Temporarily change the limits for when power to consumers is turned off and on: command `vo = 3.4` and `vl = 3.4`. To be able to change settings, the device must be unlocked, command `du = 1234`.
2. After 15 seconds the relay shall cut off the electricity to the main panel.
3. If you have a very-low-voltage shutoff relay, temporarily set the very low voltage to 3.4, command `vv=3.4`
4. After 15 seconds the relay shall cut off all power

5. Reset the voltage levels to the desired values ($v_l=3.05$, $v_v=3.0$ and $v_o=3.1$). You might have to switch on the very-low voltage-relay manually to do this, dependent on how you have connected it.
6. The low-voltage relay shall immediately switch the power on again

If this works ok it is time to test the high voltage protection.

1. Start the engine
2. If the batteries are not fully charged, set the high voltage to 3.1 V, command $v_h=3.1$
3. After 15 seconds the engine shall change sound and run more easily since the alternator now has stopped to charge the battery. If you have installed a power shunt you can verify that charging has stopped with command p , that will print voltage and current.
4. Prepare the command to reset the high voltage limit to 3.6 V, command $v_h=3.6$, but do not send it yet.
5. Wait 2 minutes, and the buzzer/alarm for high voltage will start sounding
6. Now send the prepared command to make the alarm stop.
7. Restart charging by pressing the push-button until the LED blinking stops. Will take around 7 seconds.
8. You can also test that the charging stops also by temporarily changing the setting. Press the push-button until you have seen 6 long blinks on the LED. Now you have temporarily decreased the stop charging voltage with 2 volts, which definitely will stop charging within the set delay. Doing it this way will not generate any high voltage alarm.
9. Restart charging by pressing the push-button until the LED blinking stops.
10. Stop the engine.

If this works ok, then test the low voltage warning

1. Set the warning blink level to 100 %, command $sl=100$
2. The led shall start blinking with one blink for each 10% of state of charge in the battery, i.e. if the SoC is 45 there will be 5 blinks, 2 seconds pause, 5 blink etc.
3. Reset the warning blink level to 20 % (or what you think it should be), command $sl=20$

And lastly, we test the charge level adjustment

1. Press the push-button until you see 3 long blinks on the LED.
2. Check that the system has noticed this, command po .
3. Check that in the printout contains: volt high adjust= -0.10 V
4. Do the same thing until you see 4 long blinks, po will give: volt high adjust = -0.15 V
5. Do the same thing until you see 6 long blinks, po will give: volt high adjust = 2.00 V, which means that all charging (that BMS4S controls) will be stopped all the time
6. Press the push-button until it stops blinking and check that the line says: volt high adjust = 0.00 V

4SBalancer

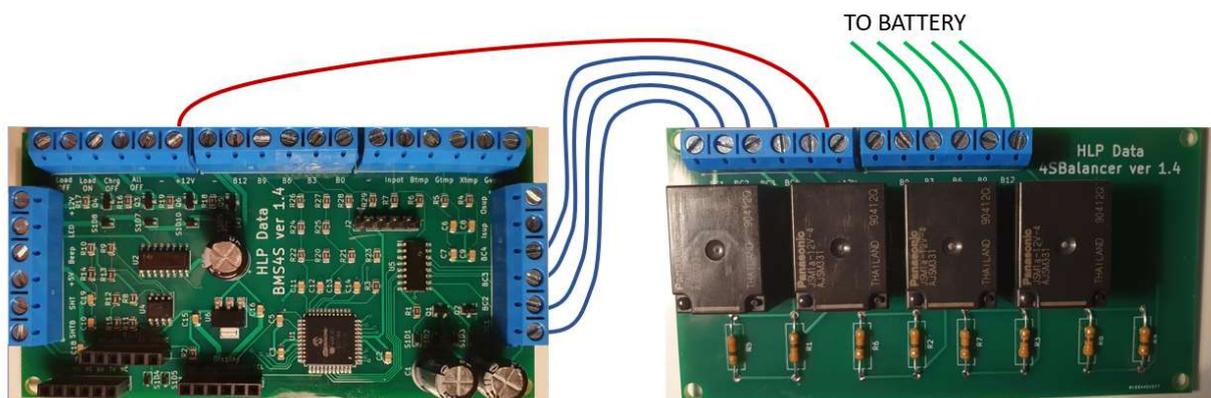
If you have a 3.6V charger it is normally enough if you use it to fully charge all cells one or two times a year with it. Since BMS4S always act on the first cell reaching a boundary, there will be no harm

done even if there is an unbalance and the usage-settings are aggressive using as close to 100% of available capacity as possible.

But if you still want the balancing to be fully automatic you can buy the 4SBalancer and connect it to BMS4S and the cells. It is built very robust with relays instead of transistors to minimize the risk of harming the cells.

This is how it is connected:

1. First prepare the battery cables and mark them with labels on the battery side,
2. Connect them on the balancer first, and last on the battery. Check one time extra that the cables are correctly connected.
3. Terminal BC1 to BC4 shall be connected to the terminals with the same name on BMS4S. Terminal B0 to B12 shall be connected to the cells.
4. Then connect the +12V on the same source that feeds BMS4S.
5. If you have connected a shunt, ensure you have calibrated it, see command `sc=nnn`. Then switch off “balancing only while charging”, command: `vd=0`, and initiate “coulomb-based balancing”, command `bi=nnn` (1.0 is good).
6. If you have no shunt connected, use the model “balancing only while charging”, command `vd=0.nn` (0.05 is good). But if you anyway want to use model “coulomb-based balancing” (i.e. your cells are badly matched), make sure the value for default charging is correctly set, command `bc=nnn`. If you charge with lower current than the set value when the voltage passes the balance voltage (`vb=3.nn`), the unbalance measurement will show a larger unbalance than it is, and the balancing will not be optimal.
7. Set the balancing amount to 25% to start with (command `ba=25`). Then check after a week or so if the extreme-point log (command: `pe`) shows that there is one cell that seem to be constantly having too much charge. If so, you might try to increase the balancing amount.



The balancer can in mode “coulomb-based balancing” even out an unbalance up to 3.5 Ah per day. It is recommended to start with balancing away 25% of the calculated unbalance to ensure that there is no risk of balancing away too much energy. This is especially important if there is no shunt connected, since then the “measurement” then will be based on what you have configured as the main charging source with command `bc=nnn`.

I/O description

In the lower left corner, you can connect the Bluetooth module and a FTDI card (or any serial TTL (5V) serial adapter: Baud rate: 9600, Data bits: 8, Stop bits: 1, Parity: None). Make sure you use the right socket and the right pins in the right holes. TX means that this is where the board is transmitting, shall be connected to the adapters RX.

BMS4S has 24 I/O screw-sockets. From lower left corner clockwise:

SHTB	connect to battery side of a low side current shunt, voltage must never exceed 1 V
SHT	connect to load side of a low side current shunt, voltage must never exceed 1 V
+5V	can be used to support LED and BEEP with current, max load 200 mA
Beep	open collector terminal, used for warning buzzer, max load 100 mA
LED	open collector terminal, used for LED indicating SoC, max load 100 mA
+12V	power supply, can be used to feed the LED and Beep with current
Load OFF	produces a 12 V pulse to switch off the relay for all loads, max 3A constant load
Load ON	produces a 12 V pulse to switch on the relay for all loads, max 3A constant load
Charge OFF	produces 12 V to disconnect all charging sources, max 3A constant load
All OFF	produces a 12 V pulse to disconnect loads when low voltage, max 3A constant load
-	minus, ground, can be used for relays for load and charge.
+12V	power supply, entry point for the power from the battery. BMS4S will use less than 10 mA during normal operations, but this terminal is also driving all equipment on terminals Load OFF, Load ON, Charge OFF, All OFF as well as all loads connected to +12V
-	minus, ground, entry point for the power from the battery
B12	voltage sense cable input, 12 V
B9	voltage sense cable input, 9 V
B6	voltage sense cable input, 6 V
B3	voltage sense cable input, 3 V
B0	voltage sense cable input, must ALWAYS be connected BEFORE +12V is connected
-	minus, ground, can be used to connect the sensors listed below
Inpot	connection point for the push-button, must never exceed 5 V
Btmp	connection point for the temperature sensitive resistor for the batteries, must never exceed 5 V
Gtmp	connection point for the temperature sensitive resistor for the alternator, must never exceed 5 V

Xtmp	connection point for the temperature sensitive resistor for either battery, alternator or other use, must never exceed 5 V
Gvolt	connection point for the voltage sense cable from the alternator
Osup	terminal to connect two BMS4S together for supervision, must never exceed 5 V
Isup	terminal to connect two BMS4S together for supervision, must never exceed 5 V
BC4	battery balancing control, connects to terminal with the same name on HLP data 4Sbalancer if you have one.
BC3	battery balancing control, connects to terminal with the same name on HLP data 4Sbalancer if you have one.
BC2	battery balancing control, connects to terminal with the same name on HLP data 4Sbalancer if you have one.
BC1	battery balancing control, connects to terminal with the same name on HLP data 4Sbalancer if you have one.

Power usage: less than 10 mA during normal operation