CAN bus protocol for the BMS16, 12-16 Cell Battery Management System

Ian Hooper, 4 April 2016

Introduction

This document is intended to assist people integrating the ZEVA BMS16 with their own master controller or other devices over CAN bus. It describes the required specifications for the CAN bus, packet format and communications protocols.

Standard bus settings

The standard bus speed is 500kbps. Packet format is CAN 2.0A, which uses 11-bit packet IDs. Please contact us if you require modules programmed to suit a different bus speed or packet format.

All packets are the Data frame type, though some require no actual data. Remote frames were not used for request packets since they are not supported by some CAN devices and protocols which may be on the same bus.

Physical Layer

The CAN port on the BMS16 has four screw terminals. Two are for the CAN H and CAN L signals and two for the 12V supply and Ground. The 12V output may be used for powering external devices, but is limited (by internal resettable fuse) to 200mA max.

We recommend using Shielded Twisted Pair (STP) cable, with two pairs of conductor – one pair for CAN H and CAN L, the other for Ground and +12V supply. Conductors should be AWG22 or larger for sufficient mechanical strength and current rating. This type of cable can be difficult to source so you can purchase the cable from us, or from large vendors such as Digikey, Mouser, RS Components, or Element14.

CAN buses work best as a single chain of devices, without any branching, and with 120Ω termination resistors at both ends to prevent signal reflection. The BMS16 has a removable termination resistor for when it is not installed at one end of a CAN bus.

Packet Summary

The BMS16 broadcasts four different CAN packets containing data, can receive configuration and commands from five packets, and broadcasts a further four packets for SMA integration.

CAN bus was invented as a realtime, low latency bus so the standard only supports small data packets up to 8 bytes in length. This sometimes results in the need for strange bit-packing and numerical scaling schemes in order to fit the required data within the 8-byte limit.

CAN ID 500 (0x1F4): BMS status (4Hz)

	Description	Units	Minimum	Maximum
Byte 0	Errors	Enum	0	8
Byte 1	State of Charge (SoC)	%	0	100
Byte 2	Voltage (low byte)	x0.1V	0	800
Byte 3	Voltage (high byte)			
Byte 4	Current (low byte)	x0.1A	22768 (-1000A)	42768 (+1000A)
Byte 5	Current (high byte)			
Byte 6	Temperature	˚C	0	120

Current value is offset by 32768 to allow signed values. (Larger numbers are considered positive and represent discharge current, smaller numbers represent charge current.)

Enumerated errors: 0 = No error, 1 = Corrupt EEPROM, 2 = Load disabled (undervoltage cell), 3 = Charge disabled (overvoltage cell), 4 = Low SoC, 5 = Overcurrent warning, 6 = Overcurrent shut-down, 7 = Overtemperature, 8 = Critical voltage.

CAN ID 501-503 (0x1F5-0x1F7): Cell Voltages, in 3 parts (4Hz)

	Description	Units	Minimum	Maximum
Byte 0	Cell voltage 1 (low byte)	x0.01V	0	500
Byte 1	Cell voltage 2	x0.01V	0	500
Byte 2	Cell voltage 3	x0.01V	0	500
Byte 3	Cell voltage 4	x0.01V	0	500
Byte 4	Cell voltage 5	x0.01V	0	500
Byte 5	Cell voltage 6	x0.01V	0	500
Byte 6	Ninth bits		000000	111111

Packet ID 501 shown. 502 contains cell voltages 7-12, and 503 contains cell voltages 13-16. In all cases cell voltages are represented with 9 bits, with the top bits stored in byte 6 (e.g bit 0 has the 9th bit of cell voltage 1).

CAN ID 504 (0x1F8): Receive config, part 1

	Description	Units	Minimum	Maximum
Byte 0	Number of cells		12	16
Byte 1	Pack capacity	x10 Ah	1	100
Byte 2	SoC warning	%	0	100
Byte 3	Full voltage	V	0	100
Byte 4	Current warning	x10 A	0	100
Byte 5	Current trip	x10 A	0	100
Byte 6	Temperature limit	˚C	0	100

Byte 7	Minimum cell voltage	x0.05 V	20	100

Received values will be automatically restricted to the ranges shown.

CAN ID 505 (0x1F9): Receive config, part 2

	Description	Units	Minimum	Maximum
Byte 0	Maximum cell voltage	x0.05 V	20	100
Byte 1	Hysteresis on cell voltage trips	x0.05 V	0	20
Byte 2	Shunt balance threshold voltage	x0.05 V	20	100
Byte 3	Shunt size	Enum	0	3

Shunt sizes are enummerated as follows: 0 = No shunt, 1 = 100A, 2 = 200A, 3 = 500A.

CAN ID 506 (0x1FA): Acknowledge error (no data)

CAN ID 507 (0x1FB): Reset SoC (to 100%, no data)

CAN ID 508 (0x1FC): Power off (no data)

CAN ID 849 (0x351): SMA integration, overall battery limits (4Hz)

	Description	Units	Minimum	Maximum
Byte 0	Maximum pack voltage (low byte)	x0.1V	0	800
Byte 1	Maximum pack voltage (high byte)			
Byte 2	Maximum charge current (low byte)	x0.1A	Always 10000	
Byte 3	Maximum charge current (high byte)			
Byte 4	Maximum discharge current (low byte)	x0.1A	Always 10000	
Byte 5	Maximum discharge current (high byte)			
Byte 6	Minimum pack voltage (low byte)	x0.1V	0	800
Byte 7	Minimum pack voltage (high byte)			

Maximum charge and discharge current are fixed at 1000A because the inverter is almost always the limiting factor, rather than battery capability.

CAN ID 853 (0x355): SMA integration, state of charge and health (4Hz)

	Description	Units	Minimum	Maximum
Byte 0	State of Charge	%	0	100
Byte 1				
Byte 2	State of Health	%	100	

The BMS16 does not perform state of health calculations for cells so always transmits 100%.

CAN ID 854 (0x356): SMA integration, battery details (4Hz)

	Description	Units	Minimum	Maximum
Byte 0	Battery voltage (low byte)	x0.01V	0	8000
Byte 1	Battery voltage (high byte)			
Byte 2	Battery amps (low byte)	x0.1A	-10000	10000
Byte 3	Battery amps (high byte)			
Byte 4	Battery temperature (low byte)	x0.1 ˚C	0	120
Byte 5	Battery temperature (high byte)			

Battery amps is a signed value. Combining bytes 2 and 3 into a 16-bit unsigned number then casting to signed should yield correct values.

CAN ID 858 (0x35A): SMA integration, error flags (4Hz)

	Description
Byte 0	Bit 2: Overvoltage cell detected Bit 3: No overvoltage cells Bit 4: Undervoltage cell detected Bit 5: No undervoltage cells Bit 6: Overtemperature detected Bit 7: Temperature OK
Byte 1	Bit 6: Overcurrent detected Bit 7: No overcurrent