EMUS G1 BMS documentation

EMUS G1 Control Unit CAN Protocol

Version 2.0.24

Version Notes:

Version	Author	Date	Description
1.0	MM	Jan 18, 2011	Initial version
1.1	MM	Jun 17, 2011	Added SOC and SOE messages
1.2	MM	Aug 25, 2011	Added optional Standard CAN Ids
1.5	MM	Oct 19, 2011	Added support for configurable ID base address and Remote Transfer Request
1.7	MM	Dec 12, 2011	Added Diagnostic Codes message; new charging error codes
2.0.0	JB	Feb 21, 2013	Some messages modified to support multiple parallel cell strings
2.0.1	JB	Apr 25, 2013	Reimplemented periodic cell detail broadcasting, using option A messages
2.0.2	JB	May 6, 2013	Modified Battery Voltage Overall Parameters message to support 4 byte long total voltage
2.0.3	JB	Jul 7, 2014	Added Configuration Parameters, Login, Set Password, Contactor Control messages. Added second byte of protection flags, and battery status flags to Diagnostic Codes message. Some naming fixes.
2.0.4	DP	Jul 7, 2014	Some naming corrections.
2.0.5	JB	Oct 1, 2014	New document title. New template applied. Added format table for each CAN message, separated value description into separate table. Many corrections of inaccuracies in previous version of this document. Many changes in effort to make this document clearer and more readable.
2.0.6	DP	Mar 16, 2015	Overall Parameters: Output Pin status field updated.
2.0.7	JB	Jul 2, 2015	Added description for „Statistics“ and „Events“ messages that were introduced in Control Unit firmware version 2.0.19_RC2.
2.0.8	JB	Nov 26, 2015	Added CAN messages related to cell temperature measurement support which was introduced in Control Unit firmware version 2.0.19_RC9. Updated „Statistics“, „Events“ and „Diagnostic Codes“ message descriptions accordingly. Modified the names of some CAN messages.
2.0.9	JB	Jan 20, 2016	Corrected tables names of “Individual Cell Voltages”, “Individual Cell Module Temperatures”, “Individual Cell Temperatures”, and “Individual Cell Balancing Rate“ Option B CAN messages.
2.0.10	JB	Mar 10, 2017	Added CAN message to set SOC value and updated “Log in/Log Out” and “Set New Password” message descriptions to reflect changes introduced in Control Unit firmware version 2.0.21.0
2.0.11	JB	Mar 22, 2017	Updated “Diagnostic Codes” message according to the changes introduced in Control Unit

			firmware version 2.0.21.2.
2.0.12	MM	Dec 10, 2018	Changed titles according to new company name
2.0.13	HS	Oct 15, 2021	Added J1939 chargers' protocol
2.0.14	HS	Jan 25, 2022	BMS Transmit / Charger Receive message description fix.
2.0.15	HS	Dec 20, 2022	Added Neuro messages, changed total voltage format in the Battery Voltage Overall Params messages.
2.0.16	DR	Mar 5, 2024	Adjusted Contactor Control documentation.
2.0.17	DR	Mar 7, 2024	Adjustment of Diagnostics messages.
2.0.18	DR	Mar 28, 2024	Adjustment of State of Charge messages.
2.0.19	DB	Apr 8, 2024	Fixed minor formatting, grammar as well as extended and standard ID description alignment issues.
2.0.20	DB	Apr 11, 2024	Aligned Diagnostic Codes and their description for extended and standard identifier.
2.0.21	DR	Apr 16, 2024	Neuro message adjustment and reduction values clarification.
2.0.22	DB	Apr 17, 2024	Added a note for the Estimated charge, adjusted the offset for additional values in the Statistics and corrected naming of some bits.
2.0.23	DR	Apr 17, 2024	Adjusted Events description.
2.0.24	DB	Apr 18, 2024	Updated Table of Contents, page numbering and the description of the Contactor Control. Unified the representation of units.

Table of Contents

Overview

EMUS Battery Management System utilizes a CAN interface, which is mainly used for control of a CAN-based charger or for cell communication (when CAN Cell Group Modules are used), but can optionally be used for retrieval of various information, control of certain features, and configuration of EMUS BMS, all done using CAN messages that are described in this document.
Most of the CAN messages that contain various information can be periodically broadcasted by EMUS BMS Control Unit using the display messages period. This period is configurable for active and idle states. Alternatively, the external application may request any of the information containing messages at any time, by sending a request CAN message to the Control Unit. Configuration and control of certain features over CAN is also done using request and response mechanism. Enabling of Remote Transfer Request (RTR) flag is optional.
The identifiers for CAN messages in EMUS CAN Protocol are partially configurable. Each identifier consists of two parts: base addres2 (later mentioned as “Base” in this document), and message Sub-ID. Base address is configurable from EMUS Control Panel.
EMUS CAN Protocol works using both standard 11-bit identifiers and extended 29-bit identifiers, depending on configuration.
If standard 11-bit identifiers are used, message identifier is formed as a sum of base address and message Sub-ID (Base Address + Message Sub-ID). If extended 29-bit identifiers are used, then the message identifier is formed by placing Base Address in upper 13 bits of the identifier, and the message Sub-ID in lower 16 bits.
CAN bus speed is configurable from EMUS Control Panel and can be matched to the one used by the external application. EMUS CAN Protocol messages can be used at any CAN bus speed.

BMS Parameters

Overall Parameters

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x00	0x00			

Response message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	INPUT SIGNALS	OUTPUT SIGNALS	NUMBER OF LIVE CELLS (MSB)	CHARGING STAGE	CHARGING STAGE DURATION (MSB)	CHARGING STAGE DURATION (LSB)	LAST CHARGING ERROR	NUMBER OF LIVE CELLS (LSB)
Base (MSB)	Base (LSB)	0x00	0x00											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 0	0	0	0

Response message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 0	0	0	8	INPUT SIGNALS	OUTPUT SIGNALS	NUMBER OF LIVE CELLS (MSB)	CHARGING STAGE	CHARGING STAGE DURATION (MSB)	CHARGING STAGE DURATION (LSB)	LAST CHARGING ERROR	NUMBER OF LIVE CELLS (LSB)

Description of response values:

Value	Description
INPUT SIGNALS	Input signals encoded by bits: Bit 0: Ignition Key, 1 = ignition is on; Bit 1: Charger Mains, 1 = connected; Bit 2: Fast Charge, 1 = fast charge selected; Bit 3: Leakage Sensor, 1 = leakage detected; Bits 4-7: Reserved.
OUTPUT SIGNALS	Output signals encoded by bits: Bit 0: Charger Enable, 1 = enabled; Bit 1: Heater Enable, 1 = enabled; Bit 2 Battery Contactor, 1 = enabled; Bit 3: Battery Fan, 1 = enabled; Bit 4: Power Reduction, 1 = enabled; Bit 5: Charging Interlock, 1 = enabled; Bit 6: DCDC Control, 1 = enabled; Bit 7: Contactor Pre-Charge, 1 = enabled.
NUMBER OF LIVE CELLS (MSB)	Most significant byte for number of cells detected.
CHARGING STAGE	Charging stage byte with following value meanings: 0 = Disconnected – charger (any type) is disconnected; 1 = Pre-heating – battery is being pre-heated to avoid charging in low temperature; 2 = Pre-charging – battery is being pre-charged with small current; 3 = Main Charging – battery is being charged with Slow or Fast charging current (depending on Fast Charge input state); 4 = Balancing – cells are being balanced to equalize their charge level; 5 = Charging Finished; 6 = Charging Error;
CHARGING STAGE DURATION (MSB)	Time elapsed in current charging stage (in minutes). Most significant byte.
CHARGING STAGE DURATION (LSB)	Time elapsed in current charging stage (in minutes). Least significant byte.

LAST CHARGING ERROR	Last charging error code with following value meanings: 0 = No error; 1 = No cell communication at the start of charging or communication lost during Pre-charging (using CAN charger), cannot charge; 2 = No cell communication (using non-CAN charger), cannot charge; 3 = Maximum charging stage duration expired; 4 = Cell communication lost during Main Charging or Balancing stage (using CAN charger), cannot continue charging; 5 = Cannot set cell module balancing threshold; 6 = Cell or cell module temperature too high; 7 = Cell communication lost during Pre-heating stage (using CAN charger); 8 = Number of cells mismatch; 9 = Cell over-voltage; 10 = Cell protection event occurred, see “Diagnostic Codes” message for determining specific protection reason;
NUMBER OF LIVE CELLS (LSB)	Least significant byte for number of cells detected.

Diagnostic Codes

Request message using extended identifier:

CAN Identifier				**Ext. ID**	**RTR**	**DLC**
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x00	0x07			

Response message using extended identifier (can be periodically broadcasted):

CAN Identifier				**Ext. ID**	**RTR**	**DLC**	**Data0**	**Data1**	**Data2**	**Data3**	**Data4**	**Data5**	**Data6**	**Data7**
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	PROTECTION FLAGS (LSB)	PROTECTION FLAGS (3rd byte)	PROTECTION FLAGS (2nd byte)	PROTECTION FLAGS (MSB)	REDUCTION FLAGS	Reserved	Reserved	BATTERY STA-TUS FLAGS
Base (MSB)	Base (LSB)	0x00	0x07											

Request message using standard identifier:

CAN Identifier	**Ext. ID**	**RTR**	**DLC**
Base + 7	0	0	0

Response message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 7	0	0	8	PROTECTION FLAGS (LSB)	PROTECTION FLAGS (3rd byte)	PROTECTION FLAGS (2nd byte)	PROTECTION FLAGS (MSB)	REDUCTION FLAGS	Reserved	Reserved	BATTERY STA-TUS FLAGS

Description of response values:

Value	Description
PROTECTION FLAGS (LSB -> MSB)	Protection status, encoded by bits for each protection type. If byte value is 0, then system is OK, and no protection flag is active. Other byte values are encoded by bits listed below: Bit 0: Cell Under-Voltage: one of the cells voltages is below minimum. Bit 1: Cell Over-Voltage: one of the cells voltages is over maximum. Bit 2: Discharge Over-Current – discharge current (negative current) exceeds the critical discharge current setting. Bit 3: Charge Over-Current – charge current (positive current) exceeds the critical charge current setting. Bit 4: Cell Module Over-Heat – cell module is above maximum temperature. Bit 5: Leakage fault – leakage signal was detected on leakage input pin. Bit 6: No Cell Communication – loss of communication to cells. Bit 7: Master-Slave Configuration Error – error with master-slave configuration. Bit 8: Master-Slave Internal CAN Bus Error – error with internal master-slave CAN bus. Bit 9: Master-Slave Common CAN Bus Error – error with common master-slave CAN bus. Bit 10: Charger Connected – notice that charger is connected. Bit 11: Cell Over-Heat – one of the cells temperatures is above maximum. Bit 12: No Current Sensor – no current sensor detected. Bit 13: Pack Under-Voltage – battery pack voltage is below minimum. Bit 14: Pack Over-Voltage – battery pack voltage is above maximum. Bit 15: Cell Under-Heat: one of the cells temperatures is below minimum. Bit 16: Cell Voltage Deviation – minimum and maximum cell values deviation. Bit 17: Pack Voltage Deviation – minimum and maximum pack voltage deviation. Bit 18: Cell Module Under-Heat – cell module is below minimum temperature. Bit 19: External Temperature Sensor Loss – loss of external temperature sensors. Bit 20: Wire break – cell wire break detection.

	Bit 21: String Voltage Deviation – detected difference between maximum parallel and minimum parallel voltages. Bit 22: Voltage And External Temperature Validation – voltages deviation between two sources.
REDUCTION FLAGS	Power reduction status, encoded by bits for each warning (requested power reduction) type. If byte value is 0, then no power reduction is requested. Other byte values are encoded by bits listed below: Bit 0: Low Cell Voltage – some cells are below low voltage warning setting. Bit 1: High Discharge Current – discharge current (negative current) exceeds the current warning setting. Bit 2: High Cell Module Temperature – cell module temperature exceeds warning temperature setting. Bit 5: High Cell Temperature – cell temperature exceeds warning temperature setting.
BATTERY STATUS FLAGS	Battery status and cell communication status flags. Bit meanings: Bit 0: Cell voltages validity (1 if valid, 0 if invalid). Bit 1; Cell module temperatures validity. Bit 2: Cell balancing rates validity. Bit 3: Number of live cells validity. Bit 4: Battery charging finished (1 if active, 0 if inactive). This flag is used only when using Non-CAN charger. Bit 5: Cell temperatures validity. Bits 6-7 are reserved.

Battery Voltage Overall Parameters

Request message using extended identifier:

CAN Identifier				**Ext. ID**	**RTR**	**DLC**
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x00	0x01			

Response message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	MIN CELL VOLTAGE	MAX CELL VOLTAGE	AVERAGE CELL VOLT-AGE	TOTAL VOLT-(MSB)	TOTAL VOLT-AGE (3rd byte)	TOTAL VOLT-AGE (2nd byte)	TOTAL VOLT-AGE (LSB)	Reserved
Base (MSB)	Base (LSB)	0x00	0x01											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 1	0	0	0

Response message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 1	0	0	8	MIN CELL VOLTAGE	MAX CELL VOLTAGE	AVERAGE CELL VOLT-AGE	TOTAL VOLT-(MSB)	TOTAL VOLT-AGE (3rd byte)	TOTAL VOLT-AGE (2nd byte)	TOTAL VOLT-AGE (LSB)	Reserved

Description of response values:

Value	Description
MIN CELL VOLTAGE	Lowest cell voltage in the battery pack, encoded in 0.01 V with basis of 2.00 V (for example, decimal value 101 means 3.01 V).
MAX CELL VOLTAGE	Highest cell voltage in the battery pack, encoded in 0.01 V with basis of 2.00 V.
AVERAGE CELL VOLTAGE	Average cell voltage in the battery pack, encoded in 0.01 V with basis of 2.00 V.
TOTAL VOLTAGE (MSB)	Total voltage of all cells in the battery pack, encoded in 0.01 V. (for example, decimal value 70501 from all four bytes means 705.01 V.
TOTAL VOLTAGE (3rd byte)	
TOTAL VOLTAGE (2nd byte)	
TOTAL VOLTAGE (LSB)	

Cell Module Temperature Overall Parameters

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x00	0x02			

Response message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	MIN CELL MODULE TEMPERA-TURE	MAX CELL MODULE TEM-PERATURE	AVERAGE CELL MODULE TEMPERA-TURE	Reserved	Reserved	Reserved	Reserved	Reserved
Base (MSB)	Base (LSB)	0x00	0x02											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 2	0	0	0

Response message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 2	0	0	8	MIN CELL MODULE TEMPERA-TURE	MAX CELL MODULE TEM-PERATURE	AVERAGE CELL MODULE TEMPERA-TURE	Reserved	Reserved	Reserved	Reserved	Reserved

Description of response values:

Value	Description
MIN CELL MODULE TEMPERATURE	Lowest cell module temperature in the battery pack, encoded in 1°C with basis of -100°C (for example, decimal value 115 means +15°C).
MAX CELL MODULE TEMPERATURE	Highest cell module temperature in the battery pack, encoded in 1°C with basis of -100°C.
AVERAGE CELL MODULE TEMPERATURE	Average cell module temperature in the battery pack, encoded in 1°C with basis of -100°C.

Cell Temperature Overall Parameters

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x00	0x08			

Response message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	MIN CELL TEMPERA-TURE	MAX CELL TEMPERA-TURE	AVERAGE CELL TEM-PERATURE	Reserved	Reserved	Reserved	Reserved	Reserved
Base (MSB)	Base (LSB)	0x00	0x08											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 8	0	0	0

Response message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 8	0	0	8	MIN CELL TEMPERA-TURE	MAX CELL TEMPERA-TURE	AVERAGE CELL TEM-PERATURE	Reserved	Reserved	Reserved	Reserved	Reserved

Description of response values:

Value	Description
MIN CELL TEMPERATURE	Lowest cell temperature in the battery pack, encoded in 1°C with basis of -100°C (for example, decimal value 115 means +15°C).
MAX CELL TEMPERATURE	Highest cell temperature in the battery pack, encoded in 1°C with basis of -100°C.
AVERAGE CELL TEMPERATURE	Average cell temperature in the battery pack, encoded in 1°C with basis of -100°C.

Cell Balancing Rate Overall Parameters

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x00	0x03			

Response message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	MIN CELL BALANCING RATE	MAX CELL BALANCING RATE	AVERAGE CELL BAL-ANCING RATE	Reserved	Reserved	Reserved	Reserved	Reserved
Base (MSB)	Base (LSB)	0x00	0x03											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 3	0	0	0

Response message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 3	0	0	8	MIN CELL BALANCING RATE	MAX CELL BALANCING RATE	AVERAGE CELL BAL-ANCING RATE	Reserved	Reserved	Reserved	Reserved	Reserved

Description of response values:

Value	Description
MIN CELL BALANCING	Lowest cell module balancing rate in the battery pack, encoded in units from 0 to 255, that correspond to 0% and 100% (for example, decimal value 127 means that at the moment 50% of maximum possible balancing current is flowing through the balancing resistor(s)).
MAX CELL BALANCING	Highest cell module balancing rate in the battery pack, encoded in units from 0 to 255, that correspond to 0% and 100%.
AVERAGE CELL BALANCING	Average cell module balancing rate in the battery pack, encoded in units from 0 to 255, that correspond to 0% and 100%.

Individual Cell Voltages

Option A

Using this method, individual cell voltages of the battery pack can be retrieved from EMUS BMS in series of CAN messages, each of them containing voltages of a group of 8 cells (last CAN message may be less than 8 bytes long, depending on the size of the last group). Number of the first group can be specified in the request message: if Byte4 in CAN Identifier is 0x00, all cell voltages, starting from the first cell in the battery pack are sent in a series of CAN Messages; if Byte4 in CAN Identifier is 0x01, first 8 cells of the battery pack are skipped, and Control Unit sends cell voltages starting from the 9th cell; etc. Group numbering starts from 0: cells 1 to 8 belong to group 0, cells 9 to 16 belong to group 1, and so on. Since EMUS BMS supports multiple parallel cell strings when using CAN Cell Group Modules, cell string number must be specified in Data0 field of the request message (0x00 if battery pack consists of only one cell string). Before sending response messages with cell voltage values, Control Unit sends one-byte-long CAN message containing cell string number. This message confirms that request was successfully registered and informs which string the cell voltage values in the consecutive CAN messages belong to (when individual cell voltages are broadcasted periodically).
NOTICE! If configured to broadcast, Control Unit will attempt to register individual cell data request of a different parameter each time (voltages, temperatures, balancing rate) at the rate of broadcast period , yet request completion time depends on the number of cells in each string (or total number of cells if there is only one string). Also, request processing may be delayed if individual cell data is actively requested over the RS232/USB interface of the Control Unit, therefore actual broadcast period of individual cell values over CAN is not clearly defined and may vary.
NOTICE! If cell communication is lost and request cannot be processed for more than 5 seconds, Control Unit will send an empty response message (same identifier as normal response message, but data length zero bytes), and unregister the request.

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x01	Group (G)				

Response: Confirmation message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x01	0x00				

Response: Data message(s) using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1-8	CELL 1 VOLT-AGE	CELL 2 VOLT-AGE	CELL 3 VOLT-AGE	CELL 4 VOLT-AGE	CELL 5 VOLT-AGE	CELL 6 VOLT-AGE	CELL 7 VOLT-AGE	CELL 8 VOLT-AGE
Base (MSB)	Base (LSB)	0x01	Group (G)											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 32 + Group (G)	0	0	1	Cell string number

Response: Confirmation message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 32	0	0	1	Cell string number

Response: Data message(s) using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 32 + Group (G)	0	0	1-8	CELL 1 VOLT-AGE	CELL 2 VOLT-AGE	CELL 3 VOLT-AGE	CELL 4 VOLT-AGE	CELL 5 VOLT-AGE	CELL 6 VOLT-AGE	CELL 7 VOLT-AGE	CELL 8 VOLT-AGE

Description of response values:

Value	Description
CELL (N) VOLTAGE	Cell voltage, encoded in 0.01 V with basis of 2.00 V (for example, decimal value 101 means 3.01 V). The cell is identified by 8-cell-group number (G) and Data byte position starting from 0. For example, byte Data2 of packet with (G)=5 shows voltage of 43rd. cell in the string.

Option B

Using this method, individual cell voltages can be retrieved from EMUS BMS using a single CAN identifier, one CAN message at a time, containing group number in Data0 field, and cell voltages of a group of 7 cells in the rest of the Data fields. The last group of the battery pack may contain less than 7 cells, therefore CAN message with values from this group may be less than eight bytes long.
Since EMUS BMS supports multiple parallel cell strings when using CAN Cell Group Modules, cell string number must be specified in Data1 field of the request message (0x00 if battery pack consists of only one cell string). Before sending a response message, Control Unit sends one-byte-long CAN message containing cell string number. This message confirms that the request was successfully registered.
NOTICE! If cell communication is lost and the request cannot be processed for more than 5 seconds, Control Unit will send an empty response message (same identifier as normal response message, but data length zero bytes), and unregister the request.

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	2	Group (G)	Cell string number
Base (MSB)	Base (LSB)	0x00	0x0B					

Response: Confirmation message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x00	0x0B				

Response: Data message(s) using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	2-8	Group (G)	CELL 1 VOLT-AGE	CELL 2 VOLT-AGE	CELL 3 VOLT-AGE	CELL 4 VOLT-AGE	CELL 5 VOLT-AGE	CELL 6 VOLT-AGE	CELL 7 VOLT-AGE
Base (MSB)	Base (LSB)	0x00	0x0B											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	
Base + 11	0	0	2	Group (G)	Cell string number

Response: Confirmation message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 11	0	0	1	Cell string number

Response: Data message(s) using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 11	0	0	2-8	Group (G)	CELL 1 VOLT-AGE	CELL 2 VOLT-AGE	CELL 3 VOLT-AGE	CELL 4 VOLT-AGE	CELL 5 VOLT-AGE	CELL 6 VOLT-AGE	CELL 7 VOLT-AGE

Description of response values:

Value	Description
Group number (G)	Group numbering starts from 0: cells 1 to 7 belong to group 0, cells 8 to 14 belong to group 1, and so on.
CELL (N) VOLTAGE	Cell voltage, encoded in 0.01 V with basis of 2.00 V (for example, decimal value 101 means 3.01 V). The cell is identified by 7-cell-group number (G) and Data byte position starting from 0. For example, byte Data2 of packet with (G)=5 shows voltage of 37^{tph} cell in the string.

Individual Cell Module Temperatures

Option A

Using this method, individual cell module temperatures of the battery pack can be retrieved from EMUS BMS in series of CAN messages, each of them containing cell module temperatures of a group of 8 cells. Request and response CAN messages have the same format as “Individual Cells Voltages” (page 15).

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x02	Group (G)				

Response: Confirmation message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x02	0x00				

Response: Data message(s) using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1-8	CELL MODULE 1 TEMPERA-TURE	CELL MODULE 2 TEMPERA-TURE	CELL MODULE 3 TEMPERA-TURE	CELL MODULE 4 TEMPERA-TURE	CELL MODULE 5 TEMPERA-TURE	CELL MODULE 6 TEMPERA-TURE	CELL MODULE 7 TEMPERA-TURE	CELL MODULE 8 TEMPERA-TURE
Base (MSB)	Base (LSB)	0x02	Group (G)											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 64 + Group (G)	0	0	1	Cell string number

Response: Confirmation message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 64	0	0	1	Cell string number

Response: Data message(s) using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 64 + Group (G)	0	0	1-8	CELL MODULE 1 TEMPERA-TURE	CELL MODULE 2 TEMPERA-TURE	CELL MODULE 3 TEMPERA-TURE	CELL MODULE 4 TEMPERA-TURE	CELL MODULE 5 TEMPERA-TURE	CELL MODULE 6 TEMPERA-TURE	CELL MODULE 7 TEMPERA-TURE	CELL MODULE 8 TEMPERA-TURE

Description of response values:

Value	Description
CELL MODULE (N)	Cell module temperature, encoded in 1°C with basis of -100°C (for example, decimal value 115 means +15°C.
TEMPERATURE	The cell is identified by group number (G) and Data byte position starting from 0. For example, byte Data2 of packet with (G)=5 shows cell module temperature of 43rd cell in the string.

Option B

Using this method, individual cell module temperatures can be retrieved from EMUS BMS using a single CAN identifier, one CAN message at a time. Request and response CAN messages have the same format as “Individual Cells Voltages” (page 18).

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1
Byte1	Byte2	Byte3	Byte4	1	0	2	Group (G)	Cell string number
Base (MSB)	Base (LSB)	0x00	0x0C					

Response: Confirmation message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x00	0x0C				

Response: Data message(s) using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	2-8	Group (G)	CELL MODULE 1 TEMPERA-TURE	CELL MODULE 2 TEMPERA-TURE	CELL MODULE 3 TEMPERA-TURE	CELL MODULE 4 TEMPERA-TURE	CELL MODULE 5 TEMPERA-TURE	CELL MODULE 6 TEMPERA-TURE	CELL MODULE 7 TEMPERA-TURE
Base (MSB)	Base (LSB)	0x00	0x0C											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	
Base + 12	0	0	2	Group (G)	Cell string number

Response: Confirmation message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 12	0	0	1	Cell string number

Response: Data message(s) using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 12	0	0	2-8	Group (G)	CELL MODULE 1 TEMPERA-TURE	CELL MODULE 2 TEMPERA-TURE	CELL MODULE 3 TEMPERA-TURE	CELL MODULE 4 TEMPERA-TURE	CELL MODULE 5 TEMPERA-TURE	CELL MODULE 6 TEMPERA-TURE	CELL MODULE 7 TEMPERA-TURE

Description of response values:

Value	Description
Group number (G)	Group numbering starts from 0: cells 1 to 7 belong to group 0, cells 8 to 14 belong to group 1, and so on.
CELL MODULE (N) TEMPERATURE	Cell module temperature, encoded in 1°C with basis of -100°C (for example, decimal value 115 means +15°C). The cell is identified by 7-cell-group number (G) and Data byte position starting from 0. For example, byte Data2 of packet with (G)=5 shows cell module temperature of 37th cell in the string.

Individual Cell Temperatures

Option A

Using this method, individual cell temperatures of the battery pack can be retrieved from EMUS BMS in series of CAN messages, each of them containing cell temperatures of a group of 8 cells. Request and response CAN messages have the same format as “Individual Cells Voltages” (page 15).

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x08	Group (G)				

Response: Confirmation message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x08	0x00				

Response: Data message(s) using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1-8	CELL 1 TEM-PERATURE	CELL 2 TEM-PERATURE	CELL 3 TEM-PERATURE	CELL 4 TEM-PERATURE	CELL 5 TEM-PERATURE	CELL 6 TEM-PERATURE	CELL 7 TEM-PERATURE	CELL 8 TEM-PERATURE
Base (MSB)	Base (LSB)	0x08	Group (G)											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 256 + Group (G)	0	0	1	Cell string number

Response: Confirmation message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 256	0	0	1	Cell string number

Response: Data message(s) using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 256 + Group (G)	0	0	1-8	CELL 1 TEM-PERATURE	CELL 2 TEM-PERATURE	CELL 3 TEM-PERATURE	CELL 4 TEM-PERATURE	CELL 5 TEM-PERATURE	CELL 6 TEM-PERATURE	CELL 7 TEM-PERATURE	CELL 8 TEM-PERATURE

Description of response values:

Value	Description
CELL (N) TEMPERATURE	Cell temperature, encoded in 1°C with basis of -100°C (for example, decimal value 115 means +15°C). The cell is identified by group number (G) and Data byte position starting from 0. For example, byte Data2 of packet with (G)=5 shows cell temperature of 43rd cell in the string.

Option B

Using this method, individual cell temperatures can be retrieved from EMUS BMS using a single CAN identifier, one CAN message at a time. Request and response CAN messages have the same format as “Individual Cells Voltages” (page 18).

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	2	Group (G)	Cell string number
Base (MSB)	Base (LSB)	0x00	0x0E					

Response: Confirmation message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x00	0x0E				

Response: Data message(s) using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	2-8	Group (G)	CELL 1 TEM-PERATURE	CELL 2 TEM-PERATURE	CELL 3 TEM-PERATURE	CELL 4 TEM-PERATURE	CELL 5 TEM-PERATURE	CELL 6 TEM-PERATURE	CELL 7 TEM-PERATURE
Base (MSB)	Base (LSB)	0x00	0x0E											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	
Base + 14	0	0	2	Group (G)	Cell string number

Response: Confirmation message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 14	0	0	1	Cell string number

Response: Data message(s) using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 14	0	0	2-8	Group (G)	CELL 1 TEM-PERATURE	CELL 2 TEM-PERATURE	CELL 3 TEM-PERATURE	CELL 4 TEM-PERATURE	CELL 5 TEM-PERATURE	CELL 6 TEM-PERATURE	CELL 7 TEM-PERATURE

Description of response values:

Value	Description
Group number (G)	Group numbering starts from 0: cells 1 to 7 belong to group 0, cells 8 to 14 belong to group 1, and so on.
CELL (N) TEMPERATURE	Cell temperature, encoded in 1°C with basis of -100°C (for example, decimal value 115 means +15°C). The cell is identified by 7-cell-group number (G) and Data byte position starting from 0. For example, byte Data2 of packet with (G)=5 shows cell temperature of 37th cell in the string.

Individual Cell Balancing Rate

Option A

Using this method, individual cell balancing rate values of the battery pack can be retrieved from EMUS BMS in series of CAN messages, each of them containing balancing rate values of a group of 8 cells. Request and response CAN messages have the same format as “Battery Individual Cells Voltages” (page 15).

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x03	Group (G)				

Response: Confirmation message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x03	0x00				

Response: Data message(s) using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1-8	CELL 1 BAL-ANCING RATE	CELL 2 BAL-ANCING RATE	CELL 3 BAL-ANCING RATE	CELL 4 BAL-ANCING RATE	CELL 5 BAL-ANCING RATE	CELL 6 BAL-ANCING RATE	CELL 7 BAL-ANCING RATE	CELL 8 BAL-ANCING RATE
Base (MSB)	Base (LSB)	0x03	Group (G)											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 96 + Group (G)	0	0	1	Cell string number

Response: Confirmation message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 96	0	0	1	Cell string number

Response: Data message(s) using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 96 + Group (G)	0	0	1-8	CELL 1 BAL-ANCING RATE	CELL 2 BAL-ANCING RATE	CELL 3 BAL-ANCING RATE	CELL 4 BAL-ANCING RATE	CELL 5 BAL-ANCING RATE	CELL 6 BAL-ANCING RATE	CELL 7 BAL-ANCING RATE	CELL 8 BAL-ANCING RATE

Description of response values:

Value	Description
CELL (N) BALANCING RATE	Cell balancing rate, encoded in units from 0 to 255, which corresponds to 0% and 100% (for example, decimal value 127 means that at the moment 50% of maximum possible balancing current is flowing through the balancing resistor(s)). The cell is identified by group number (G) and Data byte position starting from 0. For example, byte Data2 of packet with (G)=5 shows balancing rate of 43rd cell in the string.

Option B

Using this method, individual cell balancing rate values can be retrieved from EMUS BMS using a single CAN identifier, one CAN message at a time. Request and response CAN messages have the same format as “Battery Individual Cells Voltages” (page 18).

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	2	Group (G)	Cell string number
Base (MSB)	Base (LSB)	0x00	0x0D					

Response: Confirmation message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	Cell string number
Base (MSB)	Base (LSB)	0x00	0x0D				

Response: Data message(s) using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	2-8	Group (G)	CELL 1 BAL-ANCING RATE	CELL 2 BAL-ANCING RATE	CELL 3 BAL-ANCING RATE	CELL 4 BAL-ANCING RATE	CELL 5 BAL-ANCING RATE	CELL 6 BAL-ANCING RATE	CELL 7 BAL-ANCING RATE
Base (MSB)	Base (LSB)	0x00	0x0D											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	
Base + 13	0	0	2	Group (G)	Cell string number

Response: Confirmation message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 13	0	0	1	Cell string number

Response: Data message(s) using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 13	0	0	2-8	Group (G)	CELL 1 BAL-ANCING RATE	CELL 2 BAL-ANCING RATE	CELL 3 BAL-ANCING RATE	CELL 4 BAL-ANCING RATE	CELL 5 BAL-ANCING RATE	CELL 6 BAL-ANCING RATE	CELL 7 BAL-ANCING RATE

Description of response values:

Value	Description
Group number (G)	Group numbering starts from 0: cells 1 to 7 belong to group 0, cells 8 to 14 belong to group 1, and so on.
CELL (N) BALANCING RATE	Cell balancing rate, encoded in units from 0 to 255, which corresponds to 0% and 100% (for example, decimal value 127 means that at the moment 50% of maximum possible balancing current is flowing through the balancing resistor(s)). The cell is identified by 7-cell-group number (G) and Data byte position starting from 0. For example, byte Data2 of packet with (G)=5 shows balancing rate of 37th cell in the string.

State of Charge Parameters

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	Byte2	Byte3	Byte4	1	0	0
Base (MSB)	Base (LSB)	0x05	0x00			

Request message to set SOC using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	0x00	0x00	0x00	0x00	0x00	0x00	NEW STATE OF CHARGE VALUE	0x00
Base (MSB)	Base (LSB)	0x05	0x00											

Response message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	CURRENT (MSB)	CURRENT (LSB)	ESTIMATED CHARGE (MSB)	ESTIMATED CHARGE (LSB)	Reserved	ESTIMATED USER STATE OF CHARGE (MSB)	ESTIMATED USER STATE OF CHARGE (LSB)	ESTIMATED STATE OF HEALTH
Base (MSB)	Base (LSB)	0x05	0x00											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 5	0	0	0

Request message to set SOC using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 5	0	0	8	0x00	0x00	0x00	0x00	0x00	0x00	NEW STATE OF CHARGE VALUE	0x00

Response message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 5	0	0	8	CURRENT (MSB)	CURRENT (LSB)	ESTIMATED CHARGE (MSB)	ESTIMATED CHARGE (LSB)	Reserved	ESTIMATED USER STATE OF CHARGE (MSB)	ESTIMATED USER STATE OF CHARGE (LSB)	ESTIMATED STATE OF HEALTH

Description of response values:

Value	Description
CURRENT (MSB)	Momentary current value, most significant byte (data type is signed 16-bit word). Encoded in 0.1 A units. For example, decimal word value -4098 means -409.8 A (discharging). A decimal value of 173 means 17.3 A (charging).
CURRENT (LSB)	Momentary current value, least significant byte.
ESTIMATED CHARGE (MSB)	Estimated remaining charge in the battery pack, most significant byte. Encoded in 0.1 Ah units (for example, decimal value 1301 means that it is estimated that 130.1 Ah of charge left in battery pack). **NOTE:** If the Cell Capacity is changed in the Control Panel, resetting the BMS is required so that the estimated charge would also change correspondingly.
ESTIMATED CHARGE (LSB)	Estimated remaining charge in the battery pack, least significant byte.
ESTIMATED USER STATE OF CHARGE (MSB)	Estimated User State of Charge, most significant byte. Encoded in 0.01% units (for example, decimal value 04FD means that it is estimated that 12.77% of User State of Charge). **NOTE:** To get real State of Charge you would need to adjust User SoC range in Control Panel to 0% and 100%. Swapping User SoC/SoC functionality does not affect CAN message structure.
ESTIMATED USER STATE OF CHARGE (LSB)	Estimated User State of Charge, least significant byte.
ESTIMATED STATE OF HEALTH	Estimated state of health of the battery pack, encoded in 1% units (for example, decimal value 75 means that it is estimated that battery pack has 75% state of charge).

Configuration Parameters

Using CAN messages described below, it is possible to retrieve and to set EMUS BMS configuration parameter values.
NOTE! If Control Unit is password protected, setting parameter over CAN interface will fail, and Control Unit will respond with one-byte-long “Login” message, containing 0x3F in Data0 field (“?” in ASCII), requesting logging in.
NOTE! For safety, Control Unit will not allow to change configuration parameters when charger is connected and will respond with a message that contains unchanged parameter value.

Request message to retrieve parameter using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	3	PARAMETER ID (MSB)	PARAMETER ID (LSB)	PARAMETER SIZE
Base (MSB)	Base (LSB)	0x04	0x00						

Request message to set parameter using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	PARAMETER ID (MSB)	PARAMETER ID (LSB)	PARAMETER SIZE	PARAMETER DATA (MSB)	PARAMETER DATA (3rd byte)	PARAMETER DATA (2nd byte)	PARAMETER DATA (LSB)	Reserved
Base (MSB)	Base (LSB)	0x04	0x00											

Response message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	PARAMETER ID (MSB)	PARAMETER ID (LSB)	PARAMETER SIZE	PARAMETER DATA (MSB)	PARAMETER DATA (3rd byte)	PARAMETER DATA (2nd byte)	PARAMETER DATA (LSB)	Reserved
Base (MSB)	Base (LSB)	0x04	0x00											

Login request response message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	0x3F (“?” in ASCII)
Base (MSB)	Base (LSB)	0x04	0x02				

Request message to retrieve parameter using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2
Base + 128	0	0	3	PARAMETER ID (MSB)	PARAMETER ID (LSB)	PARAMETER SIZE

Request message to set parameter using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 128	0	0	8	PARAMETER ID (MSB)	PARAMETER ID (LSB)	PARAMETER SIZE	PARAMETER DATA (MSB)	PARAMETER DATA (3rd byte)	PARAMETER DATA (2nd byte)	PARAMETER DATA (LSB)	Reserved

Request message to set parameter using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7

Base + 128	0	0	8	PARAMETER ID (MSB)	PARAMETER ID (LSB)	PARAMETER SIZE	PARAMETER DATA (MSB)	PARAMETER DATA (3rd byte)	PARAMETER DATA (2nd byte)	PARAMETER DATA (LSB)	Reserved

Login request response message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 130	0	0	1	0x3F (“?” in ASCII)

Description of values:

Value	Description
PARAMETER ID (MSB)	Most significant byte of the parameter ID. For list of parameter IDs, see EMUS BMS Serial Protocol.
PARAMETER ID (LSB)	Least significant byte of the parameter ID. For list of parameter IDs, see EMUS BMS Serial Protocol.
PARAMETER SIZE	Parameter size in bytes. Used for checking request validity both when retrieving and setting the parameter. Possible values: 0x01, 0x02, 0x04.
PARAMETER DATA (MSB)	Most significant byte of parameter value (filled with 0x00 if parameter size is less than four bytes)
PARAMETER DATA (3rd byte)	Third byte of parameter value (filled with 0x00 if parameter size is less than four bytes)
PARAMETER DATA (2nd byte)	Second byte of parameter value (filled with 0x00 if parameter size is less than two bytes)
PARAMETER DATA (LSB)	Least significant byte of parameter value.

Log in/Log Out

Using CAN messages described below it is possible to log in and log out if Control Unit is password-protected. To log into level 1 access from level 0 access or into level 2 access from level 1 access, a corresponding password string of four to eight bytes must be sent in the data field of the message. To log out of the current access level, message length must be zero bytes long, and the data field empty. In both cases Control Unit will respond with eight-bytes-long message, containing AUTHENTICATION STATUS in Data0, with value of 0x00, 0x01, or 0x02 which corresponds to statuses “access level 0”, “access level 1”, and “access level 2” respectively.

Log in request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	4-8	CHARACTER 1	CHARACTER 2	CHARACTER 3	CHARACTER 4	CHARACTER 5	CHARACTER 5	CHARACTER 6	CHARACTER 7
Base (MSB)	Base (LSB)	0x04	0x02											

Log out request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x04	0x02			

Response message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	LOGIN STA-TUS	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Base (MSB)	Base (LSB)	0x04	0x02											

Log in request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 130	0	0	8	CHARACTER 1	CHARACTER 2	CHARACTER 3	CHARACTER 4	CHARACTER 5	CHARACTER 5	CHARACTER 6	CHARACTER 7

Log out request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 130	0	0	0

Response message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 130	0	0	8	LOGIN STA-TUS	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

Description of values in login request:

Value	Description
CHARACTER N	N^{th} character of the login password. Example: if password is “12345678” in ASCII, Data2 byte should be 0x32 (“2” in ASCII)

Set New Password

The CAN messages described below are used for setting or clearing EMUS BMS Control Unit passwords. To set a new password, the request message must be four to eight bytes long and contain the new password in the data field. The format of this request is similar to “Log in” request (see “Description of values in login request” in page 30). Level 1 password is set if current access level is 1, and level 2 password is set if current access level is 2. To clear the password, an empty CAN message must be sent while logged into corresponding access level. In both cases Control Unit will respond with eight bytes long message, containing SET PASSWORD STATUS in Data0, with either 0x00 or 0x01 (corresponds to “fail” and “success” respectively).
NOTE! Attempt to set a password while at access level 0 will fail, and along with the response Control Unit will send a one-byte-long “Login” message, containing 0x3F inf Data0 field (“?” in ASCII), requesting logging in.
NOTE! When setting new password, it is advised to use ASCII characters from standard computer keyboard.

Set new password request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	Byte2	Byte3	Byte4	1	0	4-8	FIRST CHAR-ACTER	SECOND CHARACTER	THIRD CHARACTER	FOURTH CHARACTER	FIFTH CHARACTER	SIXTH CHARACTER	SEVENTH CHARACTER	EIGHT CHAR-ACTER
Base (MSB)	Base (LSB)	0x04	0x03											

Clear password request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x04	0x03			

Response message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	SET PASS-WORD STA-TUS	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Base (MSB)	Base (LSB)	0x04	0x03											

Login request response message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	0x3F (“?” in ASCII)
Base (MSB)	Base (LSB)	0x04	0x02				

Set new password request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 131	0	0	8	FIRST CHAR-ACTER	SECOND CHARACTER	THIRD CHARACTER	FOURTH CHARACTER	FIFTH CHARACTER	SIXTH CHARACTER	SEVENTH CHARACTER	EIGHT CHAR-ACTER

Logout request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 131	0	0	0

Response message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 131	0	0	8	SET PASS-WORD STA-TUS	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

Login request response message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 130	0	0	1	0x3F (“?” in ASCII)

Contactor Control

Using CAN message described below it is possible to externally control battery contactor (PF12 pin function) over CAN interface: one-byte-long message must be sent with 0x00 or 0x01 in Data0, which corresponds to “OPEN CONTACTOR” and “CLOSE CONTACTOR”. Control Unit responds with eight-bytes-long message, containing command acknowledge in Data0, with possible values 0x00 and 0x01, that correspond to status “CONTACTOR OPEN” and “CONTACTOR CLOSED”.
NOTE! Contactor control over CAN will have no effect if “External Contactor Control” feature is turned off from the Control Panel (or by changing corresponding parameter over CAN).
NOTE! Contactor control over CAN does not override normal contactor operation: if contactor is open due to an active protection, a request to close it will have no effect.
NOTE! If “External Contactor Control” is enabled, contactor is open by default when Control Unit is powered up, and will be closed when a request to close is received over CAN.

Contactor control request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	Byte2	Byte3	Byte4	1	0	1	OPEN/CLOSE CONTACTOR
Base (MSB)	Base (LSB)	0x04	0x01				

Response message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	CONTACTOR OPEN/CLOS-ED	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Base (MSB)	Base (LSB)	0x04	0x01											

Contactor control request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 129	0	0	1	OPEN/CLOSE CONTACTOR

Response message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 129	0	0	8	CONTACTOR OPEN/CLOS-ED	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

Energy Parameters

Request message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x06	0x00			

Response message using extended identifier (can be periodically broadcasted):

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	ESTIMATED CONSUMP-TION (MSB)	ESTIMATED CONSUMP-TION (LSB)	ESTIMATED ENERGY (MSB)	ESTIMATED ENERGY (LSB)	ESTIMATED DISTANCE LEFT (MSB)	ESTIMATED DISTANCE LEFT (LSB)	DISTANCE TRAVELED (MSB)	DISTANCE TRAVELED (LSB)
Base (MSB)	Base (LSB)	0x06	0x00											

Request message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 6	0	0	0

Response message using standard identifier (can be periodically broadcasted):

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 6	0	0	8	ESTIMATED CONSUMP-TION (MSB)	ESTIMATED CONSUMP-TION (LSB)	ESTIMATED ENERGY (MSB)	ESTIMATED ENERGY (LSB)	ESTIMATED DISTANCE LEFT (MSB)	ESTIMATED DISTANCE LEFT (LSB)	DISTANCE TRAVELED (MSB)	DISTANCE TRAVELED (LSB)

Description of response values:

Value	Description
ESTIMATED CONSUMPTION (MSB)	Estimated momentary energy consumption per distance unit, most significant byte. Encoded in 1 Wh/distance units. For example, decimal word value 214 means that estimated energy consumption is 214 Wh per distance unit.
ESTIMATED CONSUMPTION (LSB)	Estimated momentary energy consumption per distance unit, least significant byte. Encoded in 1 Wh/distance units.
ESTIMATED ENERGY (MSB)	Estimated energy left in battery pack, most significant byte. Encoded in 10 Wh units. For example, decimal value 1296 means that it is estimated that 12.96 kWh of energy left in battery pack.
ESTIMATED ENERGY (LSB)	Estimated energy left in battery pack, least significant byte. Encoded in 10 Wh units.
ESTIMATED DISTANCE LEFT (MSB)	Estimated distance remaining, most significant byte. Encoded in 0.1 of distance units. For example, if distance unit is configured to be a kilometer, then decimal value 1257 means that estimated achievable distance with energy left in the battery pack is 125.7 km.
ESTIMATED DISTANCE LEFT (LSB)	Estimated distance remaining, least significant byte. Encoded in 0.1 of distance units.
DISTANCE TRAVELED (MSB)	Distance traveled since the last full charge of the battery, most significant byte. Encoded in 0.1 of distance units. For example, if distance unit is configured to be a kilometer, then decimal value 362 means that 36.2 km traveled since last full charge.
DISTANCE TRAVELED (LSB)	Distance traveled since the last full charge of the battery, most significant byte. Encoded in 0.1 of distance units.

Statistics

Request message to return all statistics using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x04	0x04			

Request message to return singe statistic using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	STATISTIC IDENTIFIER
Base (MSB)	Base (LSB)	0x04	0x04				

Request message to clear statistics using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	0xFF
Base (MSB)	Base (LSB)	0x04	0x04				

Response message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	STATISTIC IDENTIFIER	DATA TYPE	DATA (MSB)	DATA (3rd byte)	DATA (2nd byte)	DATA (LSB)	Reserved	Reserved
Base (MSB)	Base (LSB)	0x04	0x04											

Request message to return all statistics using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 132	0	0	0

Request message to return single statistic using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 132	0	0	1	STATISTIC IDENTIFIER

Request message to clear all statistics using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 132	0	0	1	0xFF

Response message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 132	0	0	8	STATISTIC IDENTIFIER	DATA TYPE	DATA (MSB)	DATA (3rd byte)	DATA (2nd byte)	DATA (LSB)	Reserved	Reserved

Description of response values:

Value	Description
STATISTIC IDENTIFIER	Identifier of the statistic to which the following data belongs. List of statistic identifier can be found in the table below.
DATA TYPE	Type of the following data. Possible values: 0 = Statistic value; 1 = Additional value; 2 = Statistic timestamp;
DATA (MSB)	Most significant byte of the statistic data. If actual data length is less than four bytes, this byte is equal to 0x00. For more information on particular statistic value, additional value, and timestamp format, see the table below.
DATA (3rd byte)	Third byte of the statistic data. If actual data length is less than four bytes, this byte is equal to 0x00. For more information on particular statistic value, additional value, and timestamp format, see the table below.
DATA (2nd byte)	Second byte of the statistic data. If actual data length is less than two bytes, this byte is equal to 0x00. For more information on particular statistic value, additional value, and timestamp format, see the table below.
DATA (LSB)	Least significant byte of the statistic data. For more information on particular statistic value, additional value, and timestamp format, see the table below.

Description of each statistics value:

Identifier	Value format	Additional value format	Timestamp format
0 – Total discharge	uint32_t offset: 0 multiplier: 1 result: unsigned unit: Ah	-	-
1 – Total charge	uint32_t offset: 0 multiplier: 1 result: unsigned unit: Ah	-	-
2 – Total discharge energy	uint32_t offset: 0 multiplier: 1 result: unsigned unit: Wh	-	-
3 – Total charge energy	uint32_t offset: 0 multiplier: 1 result: unsigned unit: Wh	-	-
4 – Total discharge time	uint32_t offset: 0 multiplier: 1 result: unsigned unit: s	-	-

5 – Total charge time	uint32_t offset: 0 multiplier: 1 result: unsigned unit: s	-	-
6 – Total distance	uint32_t offset: 0 multiplier: 1 result: unsigned unit: pulses	-	-
7 – Master clear count	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences		-
8 – Max Discharge Current	uint16_t offset: 0 multiplier: 0.1 result: unsigned unit: A	-	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
9 – Max Charge Current	uint16_t offset: 0 multiplier: 0.1 result: unsigned unit: A	-	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned

10 – Min Cell Voltage	uint8_t offset: 200 multiplier: 0.01 result: unsigned unit: V	uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
11 – Max Cell Voltage	uint8_t offset: 200 multiplier: 0.01 result: unsigned unit: V	uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
12 – Max Cell Voltage Difference	uint8_t offset: 0 multiplier: 0.01 result: unsigned unit: V	uint32_t LSB – Min cell voltage at the time max cell voltage difference was registered, with following format: uint8_t offset: 200 multiplier: 0.01 result: unsigned unit: V 2nd byte - Max cell voltage at the time max cell voltage difference was registered, with following format: uint8_t offset: 200 multiplier: 0.01 result: unsigned unit: V	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned

		3rd and 4th bytes (word) – ID of cell with min voltage at the time max cell voltage difference was registered, with following format: uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	
13 – Min Pack Voltage	uint32_t offset: 0 multiplier: 0.01 result: unsigned unit: V	-	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
14 – Max Pack Voltage	uint32_t offset: 0 multiplier: 0.01 result: unsigned unit: V	-	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
15 – Min Cell Module Temperature	uint8_t offset: -100 multiplier: 1 result: signed unit: °C	uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned

16 – Max Cell Module Temperature	uint8_t offset: -100 multiplier: 1 result: signed unit: °C	uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
17 – Max Cell Module Temperature Difference	uint8_t offset: 0 multiplier: 1 result: signed unit: °C	uint32_t LSB – Min cell module temperature at the time max temperature difference was registered, with following format: uint8_t offset: -100 multiplier: 1 result: signed unit: °C 2nd byte - Max cell module temperature at the time max temperature difference was registered, with following format: uint8_t offset: -100 multiplier: 1 result: signed unit: °C 3rd and 4th bytes (word) – ID of cell module with min temperature at the time max cell temperature difference was registered, with following format: uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned

18 – BMS starts count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
19 – Under-voltage protection count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
20 – Over-voltage protection count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
21 – Discharge over-current protection count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned

22 – Charge over-current protection count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
23 – Cell module overheat protection count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
24 – Leakage fault protection count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
25 – No cell comm. protection count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned

26 – Low voltage power reduction count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
27 – High current power reduction count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
28 – High cell module temperature power reduction count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
29 – Charger connect count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
30 – Charger disconnect count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-

31 – Pre-heat stage count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
32 – Pre-charge stage count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
33 – Main charge stage count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
34 – Balancing stage count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
35 – Charging finished count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
36 – Charging error count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
37 – Charging retry count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-

38 – Trips count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
39 – Charge restarts count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	-
45 – Cell overheat protection count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
46 – High cell temperature power reduction count	-	uint16_t offset: 0 multiplier: 1 result: unsigned unit: occurrences	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t offset: 0 multiplier: 1 result: unsigned
47 – Min Cell Temperature	uint8_t offset: -100 multiplier: 1 result: signed unit: °C	uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t Offset: 0 Multiplier: 1 Result: unsigned

48 – Max Cell Temperature	uint8_t offset: -100 multiplier: 1 result: signed unit: °C	uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t Offset: 0 Multiplier: 1 Result: unsigned
49 – Max Cell Temperature Difference	uint8_t offset: 0 multiplier: 1 result: signed unit: °C	uint32_t LSB – Min cell temperature at the time max temperature difference was registered, with following format: uint8_t offset: -100 multiplier: 1 result: signed unit: °C 2nd byte - Max cell module temperature at the time max temperature difference was registered, with following format: uint8_t offset: -100 multiplier: 1 result: signed unit: °C 3rd and 4th bytes (word) – ID of cell with min temperature at the time max cell temperature difference was registered, with following format: uint16_t offset: 1 multiplier: 1 result: unsigned unit: cell ID	The timestamp of when the statistic was last updated, in number of seconds since January 1, 2000, time 00:00 uint32_t Offset: 0 Multiplier: 1 Result: unsigned

Events

Request message to return all event using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	0
Base (MSB)	Base (LSB)	0x04	0x05			

Request message to clear events using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	1	0xFF
Base (MSB)	Base (LSB)	0x04	0x05				

Response message using extended identifier:

CAN Identifier				Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Byte1	**Byte2**	**Byte3**	**Byte4**	1	0	8	EVENT ENTRY NUMBER	DATA TYPE	DATA (MSB)	DATA (3rd byte)	DATA (2nd byte)	DATA (LSB)	Reserved	Reserved
Base (MSB)	Base (LSB)	0x04	0x05											

Request message to return all events using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC
Base + 133	0	0	0

Request message to clear all event using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0
Base + 133	0	0	1	0xFF

Response message using standard identifier:

CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
Base + 133	0	0	8	EVENT ENTRY NUMBER	DATA TYPE	DATA (MSB)	DATA (3rd byte)	DATA (2nd byte)	DATA (LSB)	Reserved	Reserved

Description of response values:

Value	Description
EVENT ENTRY NUMBER	Number of the event entry.
DATA TYPE	Type of the following data. Possible values: 0 = Event info; 1 = Event timestamp;
DATA (MSB)	Most significant byte of event data. If data type is 0 (Event info), this byte contains the event ID. List of event IDs: 0 – No event. 1 – BMS Started. 2 – Lost communication to cells. 3 – Established communication to cells. 4 – Cells voltage critically low. 5 – Critical low voltage recovered. 6 – Cells voltage critically high. 7 – Critical high voltage recovered. 8 – Discharge current critically high. 9 – Discharge critical high current recovered. 10 – Charge current critically high. 11 – Charge critical high current recovered. 12 – Cell module temperature critically high. 13 – Critical high cell module temperature recovered. 14 – Leakage detected. 15 – Leakage recovered. 16 – Warning: Low voltage - reducing power. 17 – Power reduction due to low voltage recovered. 18 – Warning: High current - reducing power. 19 – Power reduction due to high current recovered. 20 – Warning: High cell module temperature - reducing power. 21 – Power reduction due to high cell module temperature recovered. 22 – Charger connected. 23 – Charger disconnected. 24 – Started pre-heating stage. 25 – Started pre-charging stage. 26 – Started main charging stage.

	27 – Started balancing stage. 28 – Charging finished. 29 – Charging error occurred. 30 – Retrying charging. 31 – Restarting charging. 42 – Cell temperature critically high. 43 – Critically high cell temperature recovered. 44 – Warning: High cell temperature – reducing power. 45 – Power reduction due to high cell temperature recovered. 46 – No current sensor detected. 47 – Detected current sensor. 48 – Pack voltage critically low. 49 – Critically low pack voltage recovered. 50 – CAN Communication watchdog timed out. 51 – Pack voltage critically high. 52 – Critically high pack voltage recovered. 53 – Cell temperature critically low. 54 – Critically low cell temperature recovered. 55 – Difference between max and min cell voltages critically high. 56 – Critically high difference between max and min cell voltages recovered. 57 – Difference between max and min total voltages critically high. 58 – Critically high difference between max and min total voltages recovered. 59 – Cell Module temperature critically low. 60 – Critically low cell module temperature recovered. 61 – External temperature sensor loss detected. 62 – External temperature sensor loss recovered. 63 – Cell Wire Break detected. 64 – Cell Wire Break recovered. 65 – Voltage difference between parallel battery strings is critically high. 66 – High parallel battery strings voltage difference is recovered. 67 – Voltage difference between two sources is critically different. 68 – Voltage difference between two sources is recovered. If data type is 1 (Event timestamp), this byte contains the most significant byte of the event timestamp. The timestamp of event occurrence is coded in number of seconds since January 1, 2000, time 00:00: uint32_t Offset: 0 Multiplier: 1 Result: unsigned

DATA (3rd byte)	Third byte of the event data. If data type is 0 (Event info), this byte is reserved. If data type is 1 (Event timestamp), this byte contains the third byte of the event timestamp. The timestamp of event occurrence is coded in number of seconds since January 1, 2000, time 00:00: uint32_t Offset: 0 Multiplier: 1 Result: unsigned
DATA (2nd byte)	Second byte of the event data. If data type is 0 (Event info), this byte is reserved. If data type is 1 (Event timestamp), this byte contains the second byte of the event timestamp. The timestamp of event occurrence is coded in number of seconds since January 1, 2000, time 00:00: uint32_t Offset: 0 Multiplier: 1 Result: unsigned
DATA (LSB)	Least significant byte of the event data. If data type is 0 (Event info), this byte is reserved. If data type is 1 (Event timestamp), this byte contains the least significant byte of the event timestamp. The timestamp of event occurrence is coded in number of seconds since January 1, 2000, time 00:00: uint32_t Offset: 0 Multiplier: 1 Result: unsigned

Neuro Messages

Cycle	CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
-	0x521	0	0	8	Reserved	Reserved	Reserved	Depth of discharge (LSB)	Depth of discharge (MSB)	Reserved	Reserved	Reserved

Value	Description
Depth of discharge (LSB)	0.1%/byte offset: 0 e.g., Depth of discharge = 700, its corresponding 70.0%
Depth of discharge (MSB)	

Cycle	CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
-	0x522	0	0	8	Reserved	Reserved	Reserved	Reserved	Min discharge battery voltage (LSB)	Min discharge battery voltage (MSB)	Max discharge battery current (LSB)	Max discharge battery current (MSB)

Value	Description
Min discharge battery voltage (LSB)	0.1 V/byte offset: 0 e.g., Min discharge battery voltage = 700, its corresponding 70.0 V
Min discharge battery voltage (MSB)	
Max discharge battery current (LSB)	0.1 A/byte offset: 0 e.g., Max discharge battery current = 700, its corresponding 70.0 A
Max discharge battery current (MSB)	

Cycle	CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
-	0x523	0	0	8	Max regeneration voltage (LSB)	Max regeneration voltage (MSB)	Max charge battery current (LSB)	Max charge battery current (MSB)	Reserved	Reserved	Reserved	Reserved

Value	Description
Max regeneration voltage (LSB)	0.1 V/byte offset: 0 e.g., Max regeneration voltage = 700, its corresponding 70.0 V
Max regeneration voltage (MSB)	
Max charge battery current (LSB)	0.1 A/byte offset: 0 e.g., Max charge battery current = 700, its corresponding 70.0 A
Max charge battery current (MSB)	

Cycle	CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
-	0x530	0	0	8	Control	Charging status	Average temperature	Temperature Validity	Reserved	Reserved	Reserved	Reserved

Value	Description	
Control	**Bits:**	**Description:**
	Bit 0: Discharge Control	0: Disable. 1: Enable
	Bit 1: Regeneration Control	0: Disable. 1: Enable
	Bit 2: Reserved	-
	Bit 3: Reserved	
	Bit 4: Reserved	

	Bit 5: Reserved	
	Bit 6: Reserved	
	Bit 7: Reserved	
Charging status	**Value**:	**Description**:
	0x00	Not charging
	0x01	Charging in progress
	0x02	Charging completed
	0x03	Charging fault
Average temperature	1°C/byte offset: 40 e.g., Depth of discharge = 40, its corresponding 0°C	
Temperature Validity	**Value**:	**Description**:
	0x00	Temperature valid
	0x01	Temperature invalid

J1939 Chargers Communication Protocol

Communication Specification

The principle for the data link layer.
Communication speed for bus line: 250 Kbps.
The provision for data link layer: Refer to the related regulation of CAN2.0B and J1939.
Use 29 identifiers of CAN extended frame.

Operation Mode

1. The charger sends broadcast messages at intervals of 1s. The display meter can show the status of the charger according to up-to-date information.
2. The BMS after receiving charger broadcast message sends operating information to the charger. Upon receipt of the message, the charger will operate according to the voltage and current set in the message. If the message is not received within 5 seconds, it will enter the connection error status and stop charging.

BMS Transmit / Charger Receive messages

Cycle	CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
1000 ms	0x1806E5F4	1	0	8	Max Allowable Charing Voltage (MSB)	Max Allowable Charing Voltage (LSB)	Max Allowable Charing Current (MSB)	Max Allowable Charging Current (LSB)	Control	Reserved	Reserved	Reserved

Value	Description
Max Allowable Charing Voltage (MSB)	0.1 V/byte offset: 0 e.g., Vset = 3201, its corresponding 320.1 V
Max Allowable Charing Voltage (LSB)	
Max Allowable Charing Current (MSB)	0.1 A/byte offset: 0 e.g., Iset = 582, its corresponding 58.2 A
Max Allowable Charing Current (LSB)	
Control	0: Start Charging, 1: Stop Charging
Reserved	-

BMS Receive / Charger Transmit messages

Cycle	CAN Identifier	Ext. ID	RTR	DLC	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
1000 ms	0x18FF50E5	1	0	8	Charger Output Voltage (MSB)	Charger Output Voltage (LSB)	Charger Output Current (MSB)	Charger Output Current (LSB)	Charger Status	Reserved	Reserved	Reserved

Value	Description	
Charger Output Voltage (MSB)	0.1 V/byte offset: 0 e.g., Vset = 3201, its corresponding 320.1 V	
Charger Output Voltage (LSB)		
Charger Output Current (MSB)	0.1 A/byte offset: 0 e.g., Iset = 582, its corresponding 58.2 A	
Charger Output Current (LSB)		
Charger Status	**Bits:**	**Description**
	Bit 0: Hardware Failure	0: Normal. 1: Hardware Failure
	Bit 1: High Temperature Protection	0: Normal. 1: Over Temperature Protection
	Bit 2: Input Voltage Protection	0: Normal Input Voltage. 1: Incorrect Input Voltage, Charger Stop Working
	Bit 3: Starting State	0: Normal Battery connecting. 1: Battery Disconnecting or Reverse
	Bit 4: Communication State	0: Normal Communication. 1: Communication Receiving Time-out
	Bit 5: Reserved	-
	Bit 6: Reserved	
	Bit 7: Reserved	
Reserved		