VREGS

Victron Energy B.V. | De Paal 35 | 1351 JG Almere | The Netherlands
General phone: +31 (0)36 535 97 00 | Fax: +31 (0)36 535 97 40
E-mail: sales@victronenergy.com | **www.victronenergy.com**

Introduction

1.1 Overview

Although basic runtime information will be made available on the NMEA 2000 bus, it is undoable to define NMEA messages for all settings / special values and unwanted since that means every setting must be approved by NMEA 2000 before the product can be shipped. NMEA 2000 allows proprietary message on the bus, which will be used for these purposes. This document defines how proprietary Victron messages can be used to request and alter settings / values over the NMEA 2000 network.

1.2 Proprietary messages

NMEA 2000 defines the following proprietary messages:

PGN	Description
0xEF00[1]	Addressable single frame
0xFF00-0xFFFF	Broadcast single frame
0x1EF00[1]	Addressable transport
0x1FF00-0x1FFFF	Broadcast transport

1) The PGN itself is commonly referred to as 0xEF00 or 0x1EF00. Since the least significant byte (the PDU specific part) contains the destination address for destination specific messages, its actual value is 0xEFnn on the CAN bus.

The first two bytes of all proprietary messages are used to identify the manufacturer. Proprietary messages should be sent with priority level 7. Victron' manufacturer code of NMEA 2000 is 358. The format of the first two bytes is defined in NMEA 2000 standard, Appendix A, page 12. The fields are Manufacturer Code (11bit), Reserved(b11), Industry Group(4, Marine). The value is implemented as a single un16 so it can easily be added / compared. Since the bus is Little Endian, the values are reversed. The complete value for Victron is 0x9966, which is send over the bus as 0x66 0x99.

1.3 Victron registers

In order to allow a large amount of settings, all functionality is grouped into registers. Registers can be used for data storage or perform a function (similar to VE.Net external command register). The Victron registers (VREGs) uses different PGNs depending on the transport.

	SF	Transport (Fastpacket)
Destination specific	0xEFnn	0x1EFnn
Broadcast	0xEFFF	0x1EFFF

All fields are sent in Little Endian order. Both messages share the same format:

0x66	0x99	regId.L	regId.H	data	data	data…	…

The data of vreg can consist of multiple (related) fields, but should only be defined as such if there is a strong need to. Settings which seem related for some device might be partially relevant for another device. With a separate vreg per setting, checking if a setting is supported is as simple as checking if the vreg exists and interpretation / creation of the messages is straightforward. The downside is more traffic overhead, but since most vregs are not high frequency traffic it is not considered a problem.

The regId is an un16 leaving 4 bytes for the data for the single frame. Proprietary Fast Packets can contain up to 219 bytes of data. Since Fast Packets might need to be dropped in the software due to a lack of resources (NMEA 2000 defines two fast packets as a minimum requirement, but there is no control of the actual number of Fast Packets on the bus), single CAN messages are preferred.

Synchronizing Fast Packet vregs

Since the first PDU of the Fast Packet contains the required bytes and the changed vreg, it is quite possible to implement some fallback in the data consumers to drop the message, but still remember that it must be updated later. The question is when however. Since there are at such a moment many fast-packets on the bus, instantly is not a brilliant idea, since it could lead to a Fast Packet storm on the bus, making things even worse. Some rough idea: when the Tx Fast Packet processing is idle + some optional delay (to prevent similar device sending allot of similar requests). As an additional rule: after some timeout and when there is space for reconstruction (to catch the case that the reconstruction does not get idle soon enough). Such a solution does not pose any additional condition for producer, or consumers with many resources. It only requires thought for consumers with little resources, which are typically interested in a specific value, and can therefore track if the update of that value is dropped, which takes fixed amount of resources. Since this is currently not used, it is not defined though.

Requests

The meaning of the regId shall not depend on the transport protocol being used. Responses to valid requests are broadcasted; this allows all data consumers to see the value, reducing the total number of requests. This ensures furthermore that there is no request specific data (the address of the function the request originated from) associated with the request. This allows marking the message as requested instead of handling the request directly, which allows for an implementation where requests cannot fail, even with little resources. The request itself can be destination specific or broadcasted, but no Nacks are returned when broadcasting.

Commands

Changing values is destination specific and targeted to the device where the change is intended. Like a request, the new values are broadcasted after a successful change (directly syncing all interested devices and cannot fail), which also serves as an acknowledgement.

Negative acknowledgement

Invalid request / commands are negative acknowledged by a destination specific message addressed to the originator. Since this does require allocation, it can fail if many NACK have to be sent back. There is therefore a broadcast NACK defined to indicate that the target specific NACK could not be transmitted.

Non-bus changes

Changes to the VREGs from within the device or another interface, should also notify the CAN-bus when a value is changed. Rate limiting is implemented to prevent excessive bus traffic.

In summary: There is always a reply to a request or command. Successful request / changes will always sync with all data consumers. Non successful request / commands will always send a NACK, in the worst case a NACK is sent that the NACKs could not be sent.

Settings / static data

Settings and other static data shall not be sent periodically. Data consumers can easily implement a request for all settings / setting of interest and periodically poll them. Since VREGs are broadcasted on change requesting them can be done in a slow pace.

Dynamic values

VREGs consisting of dynamic values which are of common interest can also be sent periodically, albeit in a slow pace.

Padding

Unused bits should be set to 0 (and bytes thus to 0x00). Fast Packets only pad the bits if needed. Padding is implied for all frames and not mentioned explicitly in the definition of the registers.

Minimum requirement

All Victron devices should at least send a 0x8000 (no VReg support) error as a response to target specific request / command.

VREG definition

1.4 Grouping

The registers are grouped into pages identified by the high byte. The last pages are reserved (0xF0-0xFF) for future use, leaving the possibility to use a 5 byte payload, in case there is a real need to do so. RegId 0x0000 should never be used; it is reserved for internal use by application.

Pages from 0x00 up to and including 0x7F have a devices independent meaning. The pages from 0x80 up to and including 0xEE are product specific and can be used for product calibration / test / specific settings etc. The boundary is simple chosen to divide in roughly equal pieces. It is strongly recommended though to define the application specific top-down (i.o.w. starting at 0xEEFF) and the common bottom up. If there is a need to enlarge one region, the boundary can then still be moved without problems.

VReg page	description
0x00	VReg commands
0x01	Product information / Update
0x02	Device Control
…	
0x7F	
0x80	Product specific, message depends on the device sending them.
…	
0xEE	
0xF0	Reserved
…	
0xFF	Reserved

1.5 Page 0x00: VREG commands

The first page is used for requesting register etc.

0x0000	INTERNAL USE	-	-
This ID should never be sent over the bus; it is reserved for internal use.			

0x0001	VREQ	-	
Request for register. Request all supported registers with ids AND-ed with the mask being equal to regId. A single register can therefore be requested by setting the mask to 0xFFFF; a whole page by 0xFF00 and requesting e.g. 0x0100. A request is normally ACKed by sending the requested data. An ack should be sent targeted to the originator of the request only if it could not be acked with the data and the request was destination specific.			
	un16	regId	The regId to request (or match the AND mask)
	un16	mask	Mask to request multiple register at once. 0xFFFF is single register.
error codes:			
0x8000	Invalid request, the request did not match any vreg.		
0x8500	broadcasted, could not sent a Nack, out of resources.		

0x0002	VACK	-	
Ack/Nack, indicates that a request/command is successfully executed or indicates an error.			
	un16	regId	The regId to request (or start)
	un16	ackCode	**High byte**
			0x00-0x3F: Common acks
			0: ACK
			1: …
			2: …
			0x40-7F: register specific acks
			…
			…
			0x80-0xBF: common errors/nacks
			0x80 vregs not supported
			0x81 request not supported
			0x82 command not supported
			…
			0xC0-0xFF: register specific errors/nacks
			Reg specific error
			Low Byte
			The low byte can contain additional error specific information. Normally 0.

0x0003	VPING		-
This not a required message, but used for testing if the device can keep up with full bus load. Not to be used for checking the presence of the device!			
	un32	data	Arbitrary data returned in the reply.
error codes:			
0x8500	broadcasted, could not sent a Nack, out of resources.		

0x0004	Restore defaults		-
Restore all settings to their default except the network address being used.			
	-	-	no data
error codes:			
0x0000	ok.		

Page 0x01: Product information

Some nodes will reply with multiple ProductID PGN's. For example the BMV to NMEA2000 interface can report the version of the interface itself. And on top of that the firmware version of the BMV behind it as well. The parameter used to identify what information you are looking at is called Identifier.

0: local product.

BMV to NMEA2000 interface
1: BMV

0x0100	ProductID		read only
Number indentifying the product			
	un8	Identifier	See above
	un16	Product Id	Number uniquely defining a Victron Product. See velib code for the list.
	un8	Flags	0x02 – 0x80: Reserved 0x01: VUP Support

0x0101	Revision		read only
Number indentifying the hardware revision. **Defined but not used at the moment!**			
	un8	Identifier	
	un16	Hardware revision	Used for identifying hardware changes / incompatibilities. (update related)

0x0102	Firmware version		read only
The firmware version, in BCD format: textual representation of the version numbers uses the hexadecimal value (which allows byte operations for representation instead of division by 100, 10000 etc) . Leading zero's can be removed till the 3rd decimal, others should remain. This definition allows products to use 2 or 3 bytes for the version. Most or even all products currently use 2 bytes. Examples:			
	un8	Identifier	see product ID
	un24	Firmware version	0xFFFFFF is reserved, no firmware at all

Bytes on the bus	Value	Textual
0x66 0x99 0x02 0x01 0x00 0x01 0x02 0x03	0x030201	v3.02.01
0x66 0x99 0x02 0x01 0x00 0x01 0x02 0x00	0x000201	v2.01
0x66 0x99 0x02 0x01 0x00 0x01 0x00 0x00	0x000001	v0.01
0x66 0x99 0x02 0x01 0x00 0x00 0x00 0x00	0x000000	v0.00
0x66 0x99 0x02 0x01 0x00 0x01 0xC2 0x00	0x00C201	vC2.01 (release candidate C for v2.01)
0x66 0x99 0x02 0x01 0x00 0x01 0x02 0xB3	0xB30201	vB3.02.01 (release candidate B for v3.02.01)

0x0103	Minimum version		read only
The minimum software version the product can accept. Products should not be downgraded below this version.			
	un8	Identifier	see product ID
	un24	Firmware version	

0x0104	GroupID		read only
Number identifying the product group, used to group similar devices (e.g. for pairing chargers when operating in parallel charge mode).			
	un8	Group Id	0x00 – 0xFD: Product Group 0xFE: Reserved 0xFF: Not set

0x0105	HardwareRevision		read only
Number identifying the hardware revision of the unit.			
	un8	Hw Rev	0x00 – 0xFD: Hardware revision 0xFE: Reserved 0xFF: Not set

0x010A	Serial Number		read only
	string32	Serial number	ASCIIZ representation of the serial number

0x010B	Model Name		read only
	string32	Model name	ASCIIZ representation of the model name

0x010C	Installation description 1		read write
	string	Description	ASCIIZ user definable installation description

Note that the maximum length of the description is defined by the application.

0x010D	Installation description 2		read write
	string	Description	ASCIIZ user definable installation description

Note that the maximum length of the description is defined by the application.

0x010E	Identify		read write
Used for identification in a network			
	un8	identify	0=normal operation (default) 1=identification mode (blink/beep)

0x0110	Udf version		read only
Version of the bootloader / program which can update the firmware / setting / files , behind in, or around the device..			
	un24	UDF version	
	un8	Flags	0x02 – 0x80: Reserved 0x01: UDF Active

0x0120	Uptime		read only
time since boot			

	un32	uptime	uptime in seconds

0x0130	Can hardware rx overflows	read only	
Can hardware reception overflow counter			
	un32	rx overflows	

0x0131	Can software rx overflows	read only	
Can software reception overflow counter			
	un32	rx overflows	

0x0132	Can error passive counter	read only	
Can error passive counter			
	un32	error passive counts	

0x0133	Can bus off counter	read only	
Can bus off counter			
	un32	bus off counts	

0xED9F	Can select	read write	
Used for identification in a network			
	un8	indentify	0=normal operation (default) 1=identification mode (blink/beep)

NOTE: Deprecated, use register 0x010E instead

Examples

The PGN for single frame proprietary message is 7.0.0.EF.tg.src, where src is the sending CAN node and tg the intended recipient. When tg is 0xFF the message is intended for all nodes on the network.

In all examples our address is 0x40.

Request firmware version from one node

Victron Request (0x0001) for Firmware Version (0x0102) of node at address 0x40 from address 0x20:

```
7.0.0.EF.40.20   0x66 0x99 0x01 0x00 0x02 0x01 0xFF 0xFF
```

The reply, firmware version v3.02.01

```
7.0.0.EF.FF.40   0x66 0x99 0x02 0x01 0x00 0x01 0x02 0x03
```

If the request was not supported for this register, the reply will be an ACK (0x0002) with code 0x8000, invalid request:

```
7.0.0.EF.20.40   0x66 0x99 0x02 0x00 0x02 0x01 0x00 0x80
```

Request firmware version from all nodes

Request can be broadcasted to ask all Victron equipment for responses. Devices not having the register will not NACK in reply to a broadcasted request

Request the firmware version:

```
7.0.0.EF.FF.40   0x66 0x99 0x01 0x00 0x02 0x01 0xFF 0xFF
```

Reply from node at 0x20. Firmware version v3.02.01:

```
7.0.0.EF.FF.20   0x66 0x99 0x02 0x01 0x00 0x01 0x02 0x03
```

Reply from node at 0x30. Firmware version v8.03.01:

```
7.0.0.EF.FF.30   0x66 0x99 0x02 0x01 0x00 0x02 0x03 0x08
```

Changing a value

While data is always broadcasted (tg = 0xFF), commands have to be addressed to a specific node.

Request the current value of VREG 0xEEFF of the node at 0x20:

```
7.0.0.EF.20.40   0x66 0x99 0x01 0x00 0xFF 0xEE 0xFF 0xFF
```

Reply that the value is 0x01.

```
7.0.0.EF.FF.20   0x66 0x99 0xFF 0xEE 0x01 0x00 0x00 0x00
```

Write the new value, 0x02, to VREG 0xEEFF.

```
7.0.0.EF.20.40   0x66 0x99 0xFF 0xEE 0x02 0x00 0x00 0x00
```

Reply, acknowledging the updated data. This is always broadcasted so all nodes are aware of the change:

```
7.0.0.EF.FF.20   0x66 0x99 0xFF 0xEE 0x02 0x00 0x00 0x00
```

If the new value is not valid the reply will be a VACK (0x0002) with, for example, code 0x8300 incase the command was not allowed.

```
7.0.0.EF.40.20   0x66 0x99 0x02 0x00 0xFF 0xEE 0x00 0x83
```

Page 0x02: Controlling device(s)

Terminology

Product: Multi, Charger, Quattro, etcetera.
Control: Display, Remote control, Victron Global Remote, Canbus interface

Changing the mode, for example on/off/charger only, and input current limits is a common user action for Victron products.

Requirements for control are:

1. If all connected controls are removed, and the product is restarted, it should operate in Stand Alone mode and not be current limited / stay off. This is for fault-finding reasons.
2. A product is completely powered down after being switched off by a software command. When it is then instructed to start up again by a control, it should start with the last remote values. Not the standalone or default settings.
3. Multiple controls can be connected to one product.
4. One control could change a parameter while another control is off. The control should therefore never send its last known value on power up. Instead it should ask the network what the latest value is.

Implementation:

1 and 2: Since there are several features which can be overwritten (e.g. AC1 till AC4 input limits) and there will be a different levels of support for them in panels, a bitmask is defined with the settings using the standard, non-remote values. After startup the device will announce on the bus that it is using the internal defaults for all features. Devices wanting to control specific values instead of having the internal defaults shall send the same bitmask back with the features they can control marked as set. The product will change to and report the last stored remote values (see 4) corresponding to these fields (or or the internal defaults if none was ever set).

3: This one is straightforward on the CAN bus under the condition that panels should only send commands when there is an actual change and not periodically sent commands.

4: See requirement.

0x0200	Device Mode		read / write
	un8	mode	The mode of the device / VE.Bus system. Possible values are: 1: Charger Only (rw) 2: Inverter Only (rw) 3: On (rw) 4: Off (rw)
error codes in reply to a write:			
0x8300	Invalid, value is out of range / not applicable for the device / system.		
0xC001	VE.Bus specific error, the system cannot be controlled since another panel controls the mode. This will happen when a Digital Multi Control is connected		

0x0201	Device State		read only
	un8	state	The state of the device / VE.Bus system. Possible values are: 0x00: Off 0x01: Low Power Mode, 0x02: Fault, 0x03: Bulk, 0x04: Absorption, 0x05: Float, 0x06: Storage, 0x07: Equalize, 0x08: Passthru, 0x09: Inverting, 0x0A: Assisting, 0x0B: Power Supply Mode, 0x0C-0xFA: Reserved, 0xFB: Test, 0xFC: Hub-1, 0xFD-0xFE: Reserved, 0xFF: Not Available

Note: The device state is broadcasted on change. There is no need to request it.

Notes specific to the VE.Bus to NMEA2000 interface:

1. Although the value of Device Mode is changed instantly, the corresponding change in the Device State can take up to a minute since the system might be queried for supported functions and/or the device might have to validate the AC input.
2. VREG 0x0201 is supported from firmware version v1.00, released end of june 2012. All firmware versions prior to this might reply with data different from the state list above.

0x0202	Remote Control Used		read / write
	bit0	ACIN1 current limit	This message will report, per feature, if the internal or the remote settings are used. 0: internal value 1: remote value Device wanting to control the inverter/charger in question should check if the features they can control are set. If not, this message shall be sent to device in question, with all values they want to control remotely or with the missing. The inverter/charger shall change the setpoints to the last remote version and update the mask. The VE.Bus device will restore all values to the internal defaults on startup.
	bit1	on/off control	
	bit2	ACIN2 current limit	
	bit3	reserved	
	bit4	reserved	
	bit5	reserved	
	bit6	reserved	
	bit7	reserved	
	bit8	Send panel Leds	
	bit9	reserved	
	bit10	reserved	
	bit11	reserved	
	bit12	reserved	
	bit13	reserved	
	bit14	reserved	
	bit15	reserved	
	bit16	Send cell voltages	
	bit17	reserved	
	bit18	reserved	
	bit19	reserved	
	bit20-31	reserved	

0x0203	AC IN Current Limit		read only
	un16	limit	The active current limit in units of 100mA. 0xFFFF: not available

0x0204	AC active input		read only
	Un8	Active input	The AC input being used (or the last used one). 0xFFFF: not available

1.6 AC IN 1 Current Limit

0x0210	AC IN1 Current Limit		read only
	un16	limit	The ac1 current limit in units of 100mA. 0xFFFF: not available

0x0211	AC IN1 Current Limit Min		read only
	un16	limit	The minimum acceptable current limit in 100mA. 0xFFFF: not available

0x0212	AC IN1 Current Limit Max		read only
	un16	limit	The current limit in steps of 100mA. 0xFFFF: not available

0x0213	AC IN1 Current Limit Internal		read only
	un16	limit	The current limit in 100mA. 0xFFFF: not available

0x0214	AC IN1 Current Limit Remote		read / write
	un16	limit	**read:** The last remote current limit set in units of 100mA. 0: If Power Assist is disabled (Multi setting) the device goes to bypass. 0: If Power Assist enabled -> The device will change to invert or turn off when in charger only mode. 0xFFFF: using internal current limit (read) **write:** Set the current limit to the passed value. 0: use minimum current, see above.
error codes:			
0x8300 (w)	Value is out of range [not within min - max nor 0]		
0x8600 (w)	Not yet initialized (obtaining min/max)		
0xC001 (w)	VEBus specific error, system cannot be controlled since another panel controls the limit. This will happen when a Digital Multi Control is connected		

1.7 AC IN 2 Current Limit

0x0220	AC IN2 Current Limit		read only
	un16	limit	The ac in 2 current limit in units of 100mA. 0xFFFF: not available

0x0221	AC IN2 Current Limit Min		read only
	un16	limit	The minimum acceptable current limit in 100mA. 0xFFFF: not available

0x0222	AC IN2 Current Limit Max		read only
	un16	limit	The current limit in steps of 100mA. 0xFFFF: not available

0x0223	AC IN2 Current Limit Internal		read only
	un16	limit	The current limit in 100mA. 0xFFFF: not available

0x0224	AC IN2 Current Limit Remote		read / write
	un16	limit	**read:** The last remote current limit set in units of 100mA. 0: If Power Assist is disabled (Multi setting) the device goes to bypass. 0: If Power Assist enabled -> The device will change to invert or turn off when in charger only mode. 0xFFFF: using internal current limit (read) **write:** Set the current limit to the passed value. 0: use minimum current, see above.
error codes:			
0x8300 (w)	Value is out of range [not within min - max nor 0]		
0x8600 (w)	Not yet initialized (obtaining min/max)		
0xC001 (w)	VEBus specific error, system cannot be controlled since another panel controls the limit. This will happen when a Digital Multi Control is connected		

1.8 Battery Information

0x0370	BMS Flags		read only
	bit0	Battery charged	
	bit1	Battery almost charged	
	bit2	Battery discharged	
	bit3	Battery almost discharged	
	bit4	Battery is charging	
	bit5	Battery is discharging	
	bit6	Battery is balancing	
	bit7	Safety contactor enabled	
	bit8	Not used	
	bit9	Over-voltage alarm	
	bit10	Over-voltage warning	
	bit11	Under-voltage alarm	
	bit12	Under-voltage warning	
	bit13	Over-current charge warning	
	bit14	Over-current discharge warning	
	bit15	Over-temperature alarm	
	bit16	Over-temperature warning	
	bit17	Under-temperature charge warning	
	bit18	Under-temperature charge alarm	
	bit19	Under-temperature discharge warning	
	bit20	Under-temperature discharge alarm	
	bit21-31	Reserved	

NOTE: This message is only implemented in the Lynx Ion

0x0371	BMS State		read only
	un8	state	The state of the BMS as a number 0-8=initializing 9=running 10=error 11=standby 12=shutting down

NOTE: This message is only implemented in the Lynx Ion

0x0372	BMS Error flags		read only
	bit0	Reserved	
	bit1	Battery initialization error	
	bit2	No batteries connected	
	bit3	Unknown battery	
	bit4	Different battery types connected	
	bit5	Incorrect number of batteries	
	bit6	Lynx Shunt not found	
	bit7	Battery measure error	
	bit8	Internal calculation error	
	bit9	Incorrect number of batteries in series	
	bit10	Incorrect number of batteries	
	bit11	Hardware error	

	bit12	Watchdog error	
	bit13	Over-voltage error	
	bit14	Under-voltage error	
	bit15	Over-temperature error	
	bit16	Under-temperature error	
	bit17-21	Not used	
	bit22	ADC timeout	
	bit23	Slave error	
	bit24	Slave warning	
	bit25	Pre-charge error	
	bit26	Contactor error	
	bit27	Over-current on external output	
	bit28-31	Reserved	

NOTE: This message is only implemented in the Lynx Ion

0x0380	Battery Configuration		read only
	un8	batteries	The number of batteries in the system.
	un8	cells	The number of cells per battery. 0xFF: Not available/Invalid
	un8	parallel	The number of batteries in parallel. 0xFF: Not available/Invalid
	un8	series	The number of batteries in series. 0xFF: Not available/Invalid

NOTE: This message is only implemented in the Lynx Ion

0x0381	Cell Voltage		read only
	un8	battery	The battery instance
	un8	cell	The cell instance
	un16	voltage	Cell voltage in 0.01V 0xFFFF: Not available

NOTE: This message is only implemented in the Lynx Ion

0x0382	Mid-point Voltage		read only
	un16	voltage	Mid-point voltage in 0.01V 0xFFFF: Not available

0x0383	Mid-point Voltage Deviation		read only
	sn16	voltage	Mid-point voltage deviation in 0.1% relative to the expected mid-point voltage. E.g. a battery bank with an actual value of 48.2V has an expected mid-point voltage of 24.1V. 0x7FFF: Not available

0x0384	Cell Voltage min max (non-volatile)		read only
	un16	Minimal voltage	Minimal cell voltage in 0.01V

			0xFFFF: Not available
	un16	Maximal voltage	Maximal cell voltage in 0.01V 0xFFFF: Not available

1.9 Battery settings

The settings below relate to the battery charging algorithm. Note that availability of settings depends on the features of a specific charger model (e.g. a Solar charger does not have the storage mode, so storage mode related vregs will not be implemented).

Some charger models have physical switches to control the charging algorithm, for these models the vregs can only be queried to find the actual settings (e.g. the Skylla-i charger has a rotary switch and dip switches, when the rotary is set to any position other than CAN control most of the vregs will be read-only).

Some settings are battery type related (e.g. absorption voltage, float voltage, etc.). The corresponding vregs can only be modified when the battery type is set to 255 (=user defined type).

Battery voltage settings (absorption voltage, float voltage, etc.) will be scaled automatically when a different charger voltage is selected (only supported on specific charger models).

0xEDFF	Battery safe mode			read write
	un8	battery safe mode	0=off, 1=on (default)	

0xEDFE	Battery adaptive mode			read write
	un8	adaptive mode	0=off, 1=on (default)	

0xEDFD	Battery automatic equalisation mode			read write
	un8	auto equalisation mode	0=off (default), 1=on	

0xEDFC	Battery bulk time limit			read write
	un16	bulk time limit	0=off, time in 0.01 hours	

0xEDFB	Battery absorption time limit			read write
	un16	absorption time limit	0=off, time in 0.01 hours	

0xEDFA	Battery float time limit			read write
	un16	float time limit	0=off, time in 0.01 hours	

0xEDF9	Battery repeated absorption time duration			read write
	un16	rep. abs. time duration	time in 0.01 hours (default 1 hour)	

0xEDF8	Battery repeated absorption time interval			read write
	un16	rep. abs. time interval	time in 0.01 days (default 7 days)	

0xEDF7	Battery absorption voltage level		read write
	un16	absorption voltage	voltage in 0.01V

0xEDF6	Battery float voltage level		read write
	un16	float voltage	voltage in 0.01V

0xEDF5	Battery storage voltage level		read write
	un16	storage voltage	voltage in 0.01V

0xEDF4	Battery equalisation voltage level		read write
	un16	equalisation voltage	voltage in 0.01V

0xEDF3	Battery discharge voltage level (lower alarm boundary)		read write
	un16	discharge voltage	voltage in 0.01V

0xEDF2	Battery temperature compensation setting		read write
	sn16	temperature comp.	voltage in 0.01 mV / degree centigrade

0xEDF1	Battery type		read write
	un8	battery type	255=user defined, others are predefined, refer to the charger manual for the predefined battery types.

0xEDF0	Battery maximum current		read write
	un16	charge current limit	current in 0.1A

0xEDEF	Battery voltage selection		read write
	un8	battery voltage	0=automatic, 12/24/36/48. multiple voltages are only supported on specific charger models. For fixed voltage chargers this register will be read-only.

Note: reading this register always returns the actual battery voltage (i.e. when set to 0=automatic, it will return the detected battery voltage). Read register 0xEDEA to determine if the voltage selection is set to automatic mode.

0xEDEE	Battery storage mode		read write
	un8	storage mode	0=off, 1=on

0xEDED	Battery intelligent mode		read write
	un8	intelligent mode	0=off, 1=on

0xEDEC	Battery temperature	read only

	un16	battery temperature	temperature in 0.01K

0xEDEB	Battery overcharge voltage level (upper alarm boundary)		read write
	un16	overcharge voltage	voltage in 0.01V

0xEDEA	Battery voltage setting		read only
	un8	battery voltage	0=automatic, 12/24/36/48. multiple voltages are only supported on specific charger models.

0xEDE9	Battery power supply voltage		read write
	un16	voltage	voltage in 0.01V. voltage set-point used when the charger operates in power-supply mode.

0xEDE8	Battery BMS present		read write
	un8	bms	0=absent 1=present

1.10 Battery Monitor settings

0x0FFE	Time-to-go		read only
	un16	TTG	Time-to-go in minutes

0x0FFF	State of Charge		read write
	un16	SoC	State of Charge in 0.01% Range: 0.00 – 100.00%

0x1000	Battery Capacity		read write
	un16	Capacity	Battery capacity in Ah

0x1001	Charged Voltage		read write
	un16	Voltage	Charged voltage of the battery in 0.1V

0x1002	Charged Current		read write
	un16	Tail current	Charged current as percentage of the battery capacity (0x1000) in 0.1%

0x1003	Charged Detection Time		read write
	un16	Time	The time in minutes that setting 0x1000 and 0x1001 must be met before the monitor is automatically synchronized

0x1004	Charge Efficiency		read write
	un16	Efficiency	The charge efficiency of the battery in % Range: 0 – 100%

0x1005	Peukert Coefficient		read write
	un16	Peukert coefficient	The Peukert coefficient in steps of 0.01 Range: 1.00 – 1.50

0x1006	Current Threshold		read write
	un16	Current	Current threshold in 0.01A. Everything below this threshold is considered 0A.

0x1007	Average Time-to-go		read write
	un16	Average TTG	The time window in minutes that the averaging filter works with

0x1008	Low State-of-Charge	read write
0x1009	Low State-of-Charge clear	read write

	un16	Low SoC set/clear	Percentage in 0.1%. Used for time-to-go calculation and alarms

0x1034	Current offset		read only
	un16	offset	ADC counts

0x1029	Zero current command		write only
	N/A	N/A	No payload

0x102C	Synchronize monitor command		write only
	N/A	N/A	No payload

1.11 Charger settings

0xEDDF	Charger maximum current		read only
	un16	charger current	current in 0.1A

0xEDDE	Charger number of physical outputs		read only
	un8	outputs	1=single output, 3=fet splitter output module

0xEDDD	Charger system yield		read only
	un32	system yield	yield in 0.01kWh 0xFFFFFFFF = not available

0xEDDC	Charger user yield		read write
	un32	user yield	yield in 0.01kWh, can be reset by writing a 0 to this vreg. 0xFFFFFFFF = not available

0xEDDB	Charger internal temperature		read only
	sn16	internal temperature	temperature in 0.01 degrees centigrade

0xEDDA	Charger error code		read only
	un8	error code	0=no error 1=battery temperature too high 2=battery voltage too high 3=battery temperature sensor miswired (+) 4=battery temperature sensor miswired (-) 5=battery temperature sensor disconnected 6=battery voltage sense miswired (+) 7=battery voltage sense miswired (-) 8=battery voltage sense disconnected 9=battery voltage wire losses too high 17=charger temperature too high 18=charger over-current 19=charger current reversed 20=bulk time limit reached 21=charger current sensor issue 22=charger temperature sensor miswired 23=charger temperature sensor disconnected 24=charger fan missing 25=charger fan over-current 26=charger terminal overheated 27=charger short circuit 33=input voltage too high 34=input current too high 35=input power too high 36=input polarity reversed 37=input voltage absent

			49=load temperature too high 50=load over voltage 51=load over current 52=load current reversed 53=load over power 65=link device missing 66=link incompatible device (settings) 67=link bms connection lost 113=non-volatile storage write error 114=CPU temperature too high 116=user settings corrupt/lost 117=incompatible firmware 118=incompatible hardware 119=factory settings corrupt/lost

0xEDD9	Charger relay mode		read write
	un8	relay mode	0=unused (relay always open) 1=input voltage too high (if applicable) 2=temperature too high (limiting or off) 3=battery voltage too low (default) 4=equalisation active 5=charger in error state 6=defrost mode (<-20C) 7=battery voltage too high 8=charger in float or storage 9=dark/light feedback (solar only) 255=remote control Note: refer to the charger manual for the available options.

0xEDD8	Charger relay state (deprecated)		read write
	un8	relay state	0=open, 1=closed Note: only writable when the relay mode is set to remote control.

NOTE: Deprecated, use register 0x034E instead

0xEDD7	Charger current		read only
	un16	actual current	current in 0.1A (defined for the HEX protocol, charger data is normally broadcast using a regular NMEA2000 PGN).

0xEDD6	Charger power		read only
	un16	actual power	power in 0.01W (defined for the HEX protocol, charger data is normally broadcast using a regular NMEA2000 PGN).

0xEDD5	Charger voltage		read only
	un16	actual voltage	voltage in 0.01V (defined for the HEX protocol, charger data is normally broadcast using a regular NMEA2000 PGN).

0xEDD4	Charger additional state information		read write
	bit0	safe mode (r)	This message will report, per item if it is active or not. 0: not active 1: active Reserved items always read as 0.
	bit1	automatic equalization (r)	
	bit2	repeated absorption (r)	
	bit3	reserved (r)	
	bit4	temperature dimming (r)	
	bit5	sense wire dimming (r)	
	bit6	input current dimming (r)	
	bit7	low power mode (rw)	Write a 1 to toggle low power mode

0xEDD3	Charger yield today		read only
	un32	yield today	yield in 0.01kWh 0xFFFFFFFF = not available

0xEDD2	Charger maximum power today		read only
	un32	maximum power today	power in 1W 0xFFFFFFFF = not available

0xEDD1	Charger yield yesterday		read only
	un32	yield yesterday	yield in 0.01kWh 0xFFFFFFFF = not available

0xEDD0	Charger maximum power yesterday		read only
	un32	maximum power yesterday	power in 1W 0xFFFFFFFF = not available

1.12 DC input settings

0xEDBF	Input maximum current		read only
	un16	maximum input current	current in 0.1A

0xEDBD	Input current limit		read write
	un16	input current limit	current in 0.1A

0xEDBD	Input current		read only
	un16	actual current	current in 0.1A

0xEDBC	Input power		read only
	un32	actual power	power in 0.01W 0xFFFFFFFF = not available

0xEDBB	Input voltage		read only
	un16	actual voltage	voltage in 0.01V

0xEDBA	Input voltage maximum set		read write
	un16	maximum input voltage	voltage in 0.01V, writing this register is relevant when the relay is set to the mode switch on when the input voltage is too high. relay switches when the input voltage rises above this level.

0xEDB9	Input voltage maximum clear		read write
	un16	maximum input voltage	voltage in 0.01V, writing this register is relevant when the relay is set to the mode switch on when the input voltage is too high. relay switches off when the input voltage drops below this level.

1.13 DC output settings (e.g. load output)

The following vregs are only relevant for specific charger models that have a load output.

0xEDAF	Load output maximum current			read only
	un16	maximum current	current in 0.1A	

0xEDAE	Load output current limit			read write
	un16	actual current	current in 0.1A	

0xEDAD	Load output actual current			read only
	un16	actual current	current in 0.1A	

0xEDAC	Load output offset voltage			read only
	un8	voltage offset	voltage in 0.01V	

0xEDAB	Load output control mode			read write
	un8	operation mode	0=always switched off 1=automatic mode 2=alternative settings 1 3=alternative settings 2 4=always switched on	

0xEDAA	Load output power			read only
	un16	actual power	power in 0.01W	

0xEDA9	Load output voltage			read only
	un16	actual voltage	voltage in 0.01V	

0xEDA8	Load output status			read only
	un8	Load status	0=off 1=on	

1.14 DC channel settings

The following vregs are only relevant for specific charger models that have bi-directional ports (e.g. battery monitor) or a unit containing a FET splitter.

0xED8F	Channel 1 current		read only
	sn16	actual current	current in 0.1A

0xED8E	Channel 1 power		read only
	sn16	actual power	power in 1W

0xED8D	Channel 1 voltage		read only
	sn16	actual voltage	voltage in 0.01V

0xED7F	Channel 2 current		read only
	sn16	actual current	current in 0.1A

0xED7E	Channel 2 power		read only
	sn16	actual power	power in 1W

0xED7D	Channel 2 voltage		read only
	sn16	actual voltage	voltage in 0.01V

0xED6F	Channel 3 current		read only
	sn16	actual current	current in 0.1A

0xED6E	Channel 3 power		read only
	sn16	actual power	power in 1W

0xED6D	Channel 3 voltage		read only
	sn16	actual voltage	voltage in 0.01V

1.15 History values (deprecated)

The following registers are for reading the historic values. Note that these registers are deprecated and are used for the following products.

- BMV to N2K interface (firmware < 1.06)

For future designs use the registers described in 0.

0xEE00	H1 (deepest discharge)		read only
	sn32	deepest discharge	amp hours 0.1Ah

NOTE: Deprecated, use 0x300 for future designs

0xEE01	H2 (last discharge)		read only
	sn32	Last discharge	amp hours 0.1Ah

NOTE: Deprecated, use 0x301 for future designs

0xEE02	H3 (average discharge)		read only
	sn32	average discharge	amp hours 0.1Ah

NOTE: Deprecated, use 0x302 for future designs

0xEE03	H4 (number of charge cycles)		read only
	sn32	nr of charge cycles	

NOTE: Deprecated, use 0x303 for future designs

0xEE04	H5 (number of full discharges)		read only
	sn32	nr of discharges	

NOTE: Deprecated, use 0x304 for future designs

0xEE05	H6 (cumulative Ah)		read only
	sn32	cumulative Ah	amp hours 0.1Ah

NOTE: Deprecated, use 0x305 for future designs

0xEE06	H7 (minimal voltage)		read only
	sn32	minimal voltage	voltage in 0.01V

NOTE: Deprecated, use 0x306 for future designs

0xEE07	H8 (maximal voltage)		read only
	sn32	maximal voltage	voltage in 0.01V

NOTE: Deprecated, use 0x307 for future designs

0xEE08	H9 (seconds since last full charge)		read only
	sn32	secs since last charge	

NOTE: Deprecated, use 0x308 for future designs

0xEE09	H10 (number of automatic synchronizations)		read only
	sn32	nr of syncs	

NOTE: Deprecated, use 0x309 for future designs

0xEE0A	H11 (number of low voltage alarms)		read only
	sn32	nr of alarms	

NOTE: Deprecated, use 0x30A for future designs

0xEE0B	H12 (number of high voltage alarms)	read only

	sn32	nr of alarms	

NOTE: Deprecated, use 0x30B for future designs

0xEE0C	H13 (number of low starter voltage alarms)		read only
	sn32	nr of alarms	

NOTE: Deprecated, use 0x30C for future designs

0xEE0D	H14 (number of high starter voltage alarms)		read only
	sn32	nr of alarms	

NOTE: Deprecated, use 0x30D for future designs

0xEE0E	H15 (minimal starter voltage)		read only
	sn32	minimal voltage	voltage in 0.01V

NOTE: Deprecated, use 0x30E for future designs

0xEE0F	H16 (maximal starter voltage)		read only
	sn32	maximal voltage	voltage in 0.01V

NOTE: Deprecated, use 0x30F for future designs

1.16 History values (new)

The following vregs are replacing the vregs described in 1.15.

0x1030	Clear history command		write only
This command clears all history values			
	N/A	N/A	No payload

0x0300	The depth of the deepest discharge		read only
	sn32	deepest discharge	amp hours 0.1Ah

0x0301	The depth of the last discharge		read only
	sn32	Last discharge	amp hours 0.1Ah

0x0302	The depth of the average discharge		read only
	sn32	average discharge	amp hours 0.1Ah

0x0303	The number of charge cycles		read only
	sn32	nr of charge cycles	

0x0304	The number of full discharges		read only
	sn32	nr of discharges	

0x0305	Cumulative number of Amp hours drawn from the battery		read only
	sn32	cumulative Ah	amp hours 0.1Ah

0x0306	The minimum main battery voltage		read only
	sn32	minimal voltage	voltage in 0.01V

0x0307	The maximum main battery voltage		read only
	sn32	maximal voltage	voltage in 0.01V

0x0308	The number of seconds since the last full charge		read only
	sn32	secs since last charge	

0x0309	Number of times the monitor has autom. synchronized		read only
	sn32	nr of syncs	

0x030A	The number of low main voltage alarms		read only
	sn32	nr of alarms	

0x030B	The number of high main voltage alarms		read only

	sn32	nr of alarms	

0x030C	The number of low auxilary voltage alarms		read only
	sn32	nr of alarms	

0x030D	The number of high auxilary voltage alarms		read only
	sn32	nr of alarms	

0x030E	The minimum auxilary voltage		read only
	sn32	minimal voltage	voltage in 0.01V

0x030F	The maximum auxilary voltage		read only
	sn32	maximal voltage	voltage in 0.01V

0x0310	The amount of discharged energy		read only
	un32	discharged energy	energy in 0.01kWh

0x0311	The amount of charged energy		read only
	un32	charged energy	energy in 0.01kWh

1.17 Alarm settings

0x031F	Alarm acknowledge command		write only
	N/A	N/A	This command is used to acknowledge an active alarm. Note that the alarm remains active until the alarm condition disappears. This command has no payload.

0x0320	Low voltage alarm set		read write
0x0321	Low voltage alarm clear		read write
	sn16	Alarm set/clear value	Voltage in 0.01V Alarm is disabled if both registers are 0

0x0322	High voltage alarm set		read write
0x0323	High voltage alarm clear		read write
	sn16	Alarm set/clear value	Voltage in 0.01V Alarm is disabled if both registers are 0

0x0324	Low auxiliary voltage alarm set (e.g. starter battery)		read write
0x0325	Low auxiliary voltage alarm clear (e.g. starter battery)		read write
	sn16	Alarm set/clear value	Voltage in 0.01V Alarm is disabled if both registers are 0

0x0326	High auxiliary voltage alarm set (e.g. starter battery)		read write
0x0327	High auxiliary voltage alarm clear (e.g. starter battery)		read write
	sn16	Alarm set/clear value	Voltage in 0.01V Alarm is disabled if both registers are 0

0x0328	Low State of Charge alarm set		read write
0x0329	Low State of Charge alarm clear		read write
	un16	Alarm set/clear value	SoC in 0.1% Alarm is disabled if both registers are 0

0x032A	Low battery temperature alarm set		read write
0x032B	Low battery temperature alarm clear		read write
	un16	Alarm set/clear value	Temperature in 0.01K Alarm is disabled if both registers are 0

0x032C	High battery temperature alarm set		read write
0x032D	High battery temperature alarm clear		read write
	un16	Alarm set/clear value	Temperature in 0.01K Alarm is disabled if both registers are 0

0x032E	High internal temperature alarm set		read write
0x032F	High internal temperature alarm clear		read write
	un16	Alarm set/clear value	Temperature in 0.01K Alarm is disabled if both registers are 0

0x0330	Fuse blown alarm		read write
	un8	Alarm enable	0: Disabled 1: Enabled

0x0331	Mid-point voltage alarm set		read write
0x0332	Mid-point voltage alarm clear		read write
	un16	Alarm set/clear value	Mid-point voltage deviation in 0.1% Alarm is disabled if both registers are 0

1.18 Relay settings

0x034D	Relay invert		read write
	un8	Invert	0: Not inverted 1: Inverted Not applicable when 0x034F is set to 2 or 3

0x034E	Relay control		read write
	un8	Control	0: open 1: closed Only applicable when 0x034F is set to 2

0x034F	Relay mode		read write
	un8	Mode	0: alarm mode 1: charger mode 2: remote control 3: always open

0x100A	Relay minimal enabled		read write
	un16	time	The minimal time in minutes the relay remains closed.

0x100B	Relay disable delay		read write
	un16	time	The time in minutes the relay remains closed after the relay condition has disappeared.

0x0350	Low voltage relay set		read write
0x0351	Low voltage relay clear		read write
	sn16	Relay set/clear value	Voltage in 0.01V Relay is disabled if both registers are 0

0x0352	High voltage relay set		read write
0x0353	High voltage relay clear		read write
	sn16	Relay set/clear value	Voltage in 0.01V Relay is disabled if both registers are 0

0x0354	Low auxiliary voltage relay set (e.g. starter battery)		read write
0x0355	Low auxiliary voltage relay clear (e.g. starter battery)		read write
	sn16	Relay set/clear value	Voltage in 0.01V Relay is disabled if both registers are 0

0x0356	High auxiliary voltage relay set (e.g. starter battery)		read write
0x0357	High auxiliary voltage relay clear (e.g. starter battery)		read write
	sn16	Relay set/clear value	Voltage in 0.01V Relay is disabled if both registers are 0

0x0358	Low State of Charge relay set		read write
0x0359	Low State of Charge relay clear		read write
	un16	Relay set/clear value	SoC in 0.1% Relay is disabled if both registers are 0

0x035A	Low battery temperature relay set		read write
0x035B	Low battery temperature relay clear		read write
	un16	Relay set/clear value	Temperature in 0.01K Relay is disabled if both registers are 0

0x035C	High battery temperature relay set		read write
0x035D	High battery temperature relay clear		read write
	un16	Relay set/clear value	Temperature in 0.01K Relay is disabled if both registers are 0

0x035E	High internal temperature relay set		read write
0x035F	High internal temperature relay clear		read write
	un16	Relay set/clear value	Temperature in 0.01K Relay is disabled if both registers are 0

0x0360	Fuse blown alarm		read write
	un8	Relay enable	0: Disabled 1: Enabled

0x0361	Mid-point voltage relay set		read write
0x0362	Mid-point voltage relay clear		read write
	un16	Relay set/clear value	Mid-point voltage deviation in 0.1% Relay is disabled if both registers are 0

Examples

Example to control a VE.Bus product:

0x20 is the product we are controlling
0x40 is our address.

Initialization:
Instruct the product (or the VE.Bus to NMEA2000 interface) that we want to control the current limit, and that we want to receive the LED frames.

```
7.0.0.EF.20.40   0x66 0x99 0x02 0x02 0x01 0x01 0x00 0x00
```

Reply that the features are enabled.

```
7.0.0.EF.FF.20   0x66 0x99 0x02 0x02 0x01 0x01 0x00 0x00
```

The product will send you the panel LED state periodically. This is a binary switch event; see the VE.Bus to NMEA2000 interface manual for more information.

```
6.0.1.F2.0E.20   0x01 0x01 0x10 0x00 0x00 0xFF 0xFF 0xFF
```

Note**:**
Make sure to constantly monitor the bus for VREG Remote Control Used (0x0202) after setting it for the first time. This value is stored in volatile memory in the VE.Bus to NMEA2000 interface. So in case the cable is reset or restarted, without the display being restarted, you need to re-write VREG 0x0202.

Ask for the minimal input current limit (VREG 0x0211):

```
7.0.0.EF.20.40   0x66 0x99 0x01 0x00 0x11 0x02 0xFF 0xFF
```

Reply that the minimal input current limit is 4 Ampere:

```
7.0.0.EF.FF.20   0x66 0x99 0x11 0x02 0x28 0x00 0x00 0x00
```

Ask for the maximal input current limit (VREG 0x0212):

```
7.0.0.EF.20.40   0x66 0x99 0x01 0x00 0x12 0x02 0xFF 0xFF
```

Reply that the maximal input current limit is 16 Ampere:

```
7.0.0.EF.FF.20   0x66 0x99 0x12 0x02 0xA0 0x12 0x00 0x00
```

Ask for the active input current limit (VREG 0x0210):

```
7.0.0.EF.20.40   0x66 0x99 0x01 0x00 0x10 0x02 0xFF 0xFF
```

Reply that the active input current limit is 12 Ampere:

```
7.0.0.EF.FF.20   0x66 0x99 0x10 0x02 0x78 0x00 0x00 0x00
```

Displays should show 0x0210, the actual value for the shore current limit. To change the value write to 0x0214, AC IN1 Current Limit Remote. After writing to 0x0214, displays should look for nacks addressed to them and the broadcasted nack, 0x8500.

Displays should not check (and resend) if the new value equals the value they want to set, since two displays might end up fighting on two different values and continuously pushing their intended value.

Ask for the current state (VREG 0x0201):
`7.0.0.EF.20.40  0x66 0x99 0x01 0x00 0x01 0x02 0xFF 0xFF`

Reply that the current state is power assist:
`7.0.0.EF.FF.20  0x66 0x99 0x01 0x02 0x0A 0x00 0x00 0x00`

Normal operation (initialisation finished)

Change the current limit to 10 Ampere:
7.0.0.EF.20.40 | 0x66 0x99 0x14 0x02 0x64 0x00 0x00 0x00

Reply by VREG 0x0214, acknowledging the change:
7.0.0.EF.FF.20 | 0x66 0x99 0x14 0x02 0x64 0x00 0x00 0x00

[delay]

Second reply, indicating that the input current limit has changed:
7.0.0.EF.FF.20 | 0x66 0x99 0x10 0x02 0x64 0x00 0x00 0x00

Turn the product off via VREG Device Mode (0x0200):
7.0.0.EF.20.40 | 0x66 0x99 0x00 0x02 0x04 0x00 0x00 0x00

Reply, indicating that it is switching off:
7.0.0.EF.FF.20 | 0x66 0x99 0x00 0x02 0x04 0x00 0x00 0x00

[delay]

Reply, indicating that it is switched off (VREG 0x0201, Device State)
7.0.0.EF.FF.20 | 0x66 0x99 0x01 0x02 0x00 0x00 0x00 0x00

Changes

v1 - 03-11-2011: Corrected mistake in ACIN1 enabling remote limit. The first bit is 0x01 not 0x80.

v2 - 22-02-2012: Changed example in firmware version frame (0x0212 = v2.12, and not v2.18)

v3 - 28-03-2012: Added VReg command to enable test mode

v4 - 01-05-2012: Updated description of Product ID and UDF Version VREG's

v5 - 01-06-2012: Added example table to Firmware version VREG

v6 - 13-06-2012: Document split in VE.Can registers – private.docx and VE.Can registers.docx.

v7 - 26-06-2012: Rewording and reordering

v8 - 12-07-2012: Added Battery Information registers and updated Remote Control Used

v9 - 22-08-2012: Updated Battery Configuration VREG

v10 - 28-08-2012: Added charger related VREGs

v11 - 10-09-2012: Updated Battery Configuration VREG

v12 - 17-10-2012: Added on/off control bit in VREG 0x0202

v13 - 22-01-2013: Added ac in2 current limit and active input

v14 - 04-06-2013: Added group id, battery temperature and dc channel vregs

v15 - 17-06-2013: Added history vregs

v16 - 16-08-2013: Improved wording of version bytes

v17 - 11-11-2013: Up to date with velib

v18 - 19-08-2014: Added hardware revision (0x0105), added battery vregs (0xEDE9 and 0xEDE8), updated error list (0xEDDA), added/modified input voltage maximum (0xEDBA and 0xEDB9)

v19 - 13-01-2015: Added Hub-1 and Test to Device State (0x0201); Added error 26 & 27 to Charger Error Code (0xEDDA); Added low power mode flag to Charger additional state information (0xEDD4)

v20 - 28-04-2015: Added BMS registers 0x370-0x372, resp. BMS Flags, BMS State and BMS Error Flags