ALBATROS Power Control and Logging Specification

Eamon Egan
2019-02-05
rev. 5

Introduction

Scope

This document describes the power control and logging system for the ALBATROS solar system, which accompanies the Victron Energy charge controller.

Conventions

Italics indicate text that is merely intended as commentary or suggestion, or which describes functions that are not going to be implemented right away.

Hardware

Arduino Mega2560

The logging and control functions are implemented in an Arduino (ATMega2560) with the following hardware peripherals and interfaces (to be further detailed in a later section):

- SD card (accessed by SPI)
- RTC with coin cell backup
- Serial ports connected to front panel 3.5mm stereo jack and Victron
- GPIO control of SSR to power on/off the main system
- Bi-color status LED red/green (to indicate logging status; possibly other functions also if useful)

Serial Ports

The Mega2560 provides 4 hardware serial ports which are used as described in the table below.

Port name	**Connection**	**Baud rate**	**Arduino pin**	
			Tx	Rx
Serial	USB connection, opens as monitor in Arduino IDE not used in firmware (except for development)	9600	1	0
Serial1	Front panel stereo jack serial port	9600	18	19
Serial2	Victron controller	19200	16	17
Serial3	Unused		14	15

Datalogging shield

The datalogging shield (Adafruit product 1141) provides the SD interface including level translation from 5V to 3.3V, the RTC (check accuracy vs Chronodot), and a backup battery. The UART interface receives

the signal directly from the Victron charge controller. No level translation is needed from Victron to Arduino as both are 5V based.

The datalogging shield also includes a prototyping area where are mounted the following:

- SSR (Toshiba TLP3543A) with current limiting resistor (390 ohm) on its input LED circuit
- 6 pin single inline Molex header to connect to the two serial ports (3.5mm jack, Victron)
- 3 pin single inline Molex header to connect to the front panel LED, with two current limiting resistors (1.2k)

SSR Current Capacity

The TLP3543A SSR (photorelay) contains two MOSFETS with a common source connection, in its output circuit. We use it in its “C connection” configuration, with both drain pins connected together. In this configuration, it is rated for 10A output current at 25C ambient temperature.

Using it in is C configuration means that it is sensitive to polarity. It must be connected so that (positive) current flows from drain to source. As it is connected in the circuit, the positive supply (from the battery) is connected to the positive SSR connection (MOSFET drains), and the load is connected to the negative SSR connection (MOSFET common source pin).

Do not connect the SSR in reverse. If this is done, it will fail to block current. Further, when the SSR input is not activated, conduction will occur through MOSFET body diodes, resulting in a diode voltage drop and greater, possibly damaging, heat dissipation.

Firmware Description

The firmware performs the following functions:

- Receive, validate, condense, timestamp and log information received from the Victron charge controller, on a regularly sampled basis.
- Log controller actions and exceptions (to a separate file)
- Power up and down the main system, by controlling a GPIO line connected to an SSR (solid-state relay), according to the sensed battery voltage.
- Allow setting of the RTC, and programming of power up/down parameters, via serial port.

Victron Data Reception

The Victron sends a data packet once per second. The format is described in the VE.DIRECT. The firmware decodes the packet and validates the checksum.

Logging

The SD card is assumed to be pre-formatted FAT32.

Logging occurs to two files at a time: one is the timestamped Victron data, the other is event related messages. The file is opened and closed on each log message.

The code is timezone-agnostic, so the time may be set to UTC or local and will be logged in the same way it is set.

Periodic Logging

The Victron data packet is parsed, and selected fields are accumulated over a user-determined period, then written out in CSV format. Each log record includes a timestamp in "unix time" (seconds since 1970/01/01). Time will wrap to negative in 2038. Victron packets include a checksum is validated prior to logging. If the checksum fails, the data is not accumulated.

The logging file, a new one each day, is named YYYYMMDD.log. The following variables are logged:

CSV header entry / description	**Type of accumulation**	**Units**	**Note**
BatVolt Battery voltage	average/min/max	mV	
PVVolt PV voltage	average/min/max	mV	This field often shows small values e.g. 20mV, when there is no connection on the PV lines
PVPwr PV power	average/min/max	W	
BatCur Battery current	average/min/max	mA	Note that since the MPPT 150/35 does not have a load port, this is the net charger current into the battery. It does not register negative when no charging is occurring, despite the fact that the unit takes some current to run.
Error Error code	sample last value		0 = No error 2 = Battery voltage too high 17 = Charger temperature too high 18 = Charger overcurrent 19 = Charger current reversed 20 = Bulk time limit exceeded 21 = Current sensor issue 26 = Terminals overheated 33 = Input voltage (PV) too high 34 = Input current (PV) too high 38 = Input shutdown (from excessive battery voltage) 116 = Factory calibration data lost
State charger state	bitmap		0 = off 2 = fault 3 = bulk 4 = absorption 5 = float
MPPT MPPT tracking state	bitmap		0 = off 1 = voltage or current limited 2 = MPPT tracker active
Load on/off Load	bitmap		

Bitmap accumulation means that for each status number seen within the sample period, the corresponding bit is set in the logged value. For example, if the value 0 is seen during the entire time, the value 1 (bit 0 set) is logged. If the values 0 and 1 were seen, the value 3 (bits 0 and 1 set) is logged.

Some error states will not reset automatically – for details see the Appendix.

Event Logging

Discrete events are logged in a separate file, as they occur. A separate log file is used for this (to record events and triggering criteria to allow reconstruction of algorithm logic in case debugging is needed), but it may also be good to have a field in the main power log – if only because the Victron will not be logging load current, and this is obviously part of understanding what is going on from the perspective of the solar/battery system.

The filename is named status.log (may be broken into multiple ones on a weekly? Monthly? Basis)

Main System Power Control

Implemented

The Arduino controls one GPIO line that connects or cuts power to the main system, to allow for power conservation by turning off the load under defined conditions.

The load will be turned off if any of the following conditions are true:

1. The sensed battery voltage is below the turn-off threshold.
2. The sensed battery voltage is below the turn-on threshold, and the low-voltage condition is asserted
3. The time-of-day is between day start and day end
4. A timeout occurs in receiving Victron data updates (this is normally once per second). This likely indicates failure of the Victron unit; in any case, we don't know what the voltage is.

Note that conditions 1 and 2 together implement hysteresis on the low-voltage off condition. The turn-on voltage is to be set higher than the turn-off voltage.

Extended Power-control Function (daytime running)

The following functionality is not yet implemented.

The firmware reads the voltage during a prescribed period, when the load is on but charging current is not present (i.e. at night). Using a combination of the data acquired overnight, or a priori data stored in EEPROM as to the discharge rate, or simply using a different voltage threshold applied at a specific time in the morning, a decision is made as to whether to run during the following day. Two schemes, the first simpler than the second, can be described as follows:

1. Compare the sensed voltage with a "DAY_RUN_MINIMUM" EEPROM voltage threshold variable, at a given time at night, and make the decision for the following morning.
2. Use acquired data to compute a slope and an intercept. Using the intercept, plus a combination of the computed slope and a slope value stored in EEPROM (the "slope" variable, which may be updated immediately or gradually to follow acquired data), determine the span in hours after which the voltage would fall below the turn-off threshold if the battery were not charged. If this is less than the "Reserve" EEPROM variable, decide to stay off for the following day.

Note: Battery State of Charge

This note provides guidance in choice of low-voltage cutout thresholds.

Typically, for lead-acid batteries, the battery should never be discharged below 0% state of charge, and ideally (to preserve battery cycle life) never or seldom below 50%. Voltages corresponding to these levels can be read off of discharge curves published for a generic battery of the type used, or, preferably, for the particular battery model being used. Note that voltage is also affected by temperature. The expected temperature should be factored in if possible.

Often, many discharge curves are provided, for different rates of discharge. Bear in mind that the discharge rate in this application is quite low relative to battery capacity, likely lower than that corresponding to any of the curves published: a 2A discharge rate on a 200Ah battery corresponds to a C/100 discharge condition, and can almost be considered an open-circuit condition. If the internal resistance of the battery is known or can be estimated, it can be used to adjust voltage thresholds (downwards) by 2A x R_{INT}, from an open circuit voltage. Open circuit voltages corresponding to different states of discharge may not be published, so measurement or creative estimation may be required.

The firmware does not explicitly estimate state of charge, but instead allows setting of voltage thresholds which correspond to states of charge. This works, under the assumptions that:

1. The battery is in a steady state when the voltage reading is taken (i.e. no recent changes in battery current flow);
2. The steady battery current flow is taken into account in estimating state of charge from the battery terminal voltage.

Serial Monitor

The serial monitor runs at 9600 baud through the stereo jack mounted on the front panel.

It should be used with any PC serial port program, as follows:

- Set to use the USB serial port (note: the FTDI driver is known in Windows, so should not be a problem to install)
- Set to 9600 baud (8 bits, no parity)
- Set to send a CR upon line entry (this is the terminating character of each command as parsed by the monitor)

The pinout (compatible with FTDI p/n TTL-232R-5V-AJ) is as follows:

Tip: Arduino Rx / Plug (PC) Tx
Ring: Arduino Tx / Plug (PC) Rx
Shield: GND

Note that the interface cable also comes in a 3.3V version; the 5V version should be used in this application.

The serial monitor implements the following commands. In the following, *italic font* indicates a variable element in the command, while normal text is literal.

Command	**Meaning**	**Description / note**
help	Display help	Any unrecognized command displays the help.
def	Dump EEPROM fields	List all EEPROM fields (Note: the in-memory version of the fields)
inv	Invalidate EEPROM	Overwrite a header byte in the EEPROM so that it will be reinitialized on the next restart. Used for testing or to set all fields back to defaults.
set *index value*	Set the configuration variable indicated by the specified *index* (0 and up) to the specified *value*	Values can be integers or times (depending on the variable). Hint: display the fields first with def as a guide to the indexes. For times, minutes and seconds are optional and will default to zero.
com	Commit in-memory fields to EEPROM	All fields at once are written to EEPROM and will then persist through power-off. This must be done after any changes are made, unless the desire is to test these changes without making them permanent.
rtc	Read current RTC setting	
rtc time *time*	Set the RTC time buffer to *time* (hh:mm:ss format)	This **does not immediately set the RTC**, but inputs a time (to be written out to the RTC with the adj subcommand). Minutes and seconds are optional and default to zero.
rtc date *date*	Set the RTC date buffer to *date* (yyyy:mm:dd format)	This **does not immediately set the RTC**, but inputs a date time (to be written out to the RTC with the adj subcommand).
rtc adj	RTC adjust. Write the values previously input as *time* and *date*, to the RTC	**Note that both time and date must be entered before adjusting the RTC.** Even if the RTC already has the correct date, it should be set using the rtc command, before adjusting it (to, say, simply adjust the time).
vs	Victron status.	Display the latest (numeric) data fields from the Victron.
ls	As in unix ls	Display the contents of the SD card root folder.
cat *file*	Dump contents of specified *file*	Filename is case-insensitive. File contents are sent to the monitor port. Command cannot be interrupted.
rm *file*	Erase specified *file*	This would not normally be done.
reset	Reset the firmware	Jump to address zero. This does not do a hard reset and so may not work if hardware is somehow stuck.
reinit	Reinitialize variables from EEPROM	Reread the variables from EEPROM (overwriting any local changes). If EEPROM has been invalidated, firmware defaults will be installed.

EEPROM Configuration Variables

Name	Type	Default	Note
VBAT_TURNOFF_MV	integer	25000	25.0 volts
VBAT_TURNON_MV	integer	25100	25.1 volts
MEASURE_START	time	23:00:00	Start of battery state measurement (not currently implemented)
MEASURE_END	time	03:00:00	End of battery state measurement (not currently implemented)
DAY_START	time	06:00:00	Start of daytime (currently unconditional turn-off of load)
DAY_END	time	18:00:00	End of daytime
HOURS_RESERVE	integer	48	Hours of no-charge reserve if to run in daytime (not currently implemented)
SLOPE_UV_PER_HOUR	integer	36360	Assumed discharge rate (220Ah, 50% discharge, 2V span, 2A)
SECS_PER_LOG	integer	60	1 minute between log entries
MAX_VIC_DATA_AGE	integer	1000	Declare a Victron data timeout and shut down the load if 1000 seconds (almost 17 minutes) go by without getting a valid Victron data packet (this is a long time, but not very long compared to battery life)

Status Interface

Status interface denotes a visual indication of system status using the front panel LED.

This interface is still under definition

A bi-colour (green / red) LED is mounted on the front panel to indicate system status as follows.

It should indicate the following conditions:

- logging / normal operation vs exception state
- load on/off

Exception conditions that can be detected:

- Victron error state
- Victron data timeout
- SD, RTC failure

Loose Ends

Documentation TODO

- EEPROM format and upgrade paths
- Hardware implemented on shield (schematic)

Implementation TODO

(This may not be a complete list)

Power saving (hardware and firmware)

Integration of shield (make a PCB for future manufacturing)

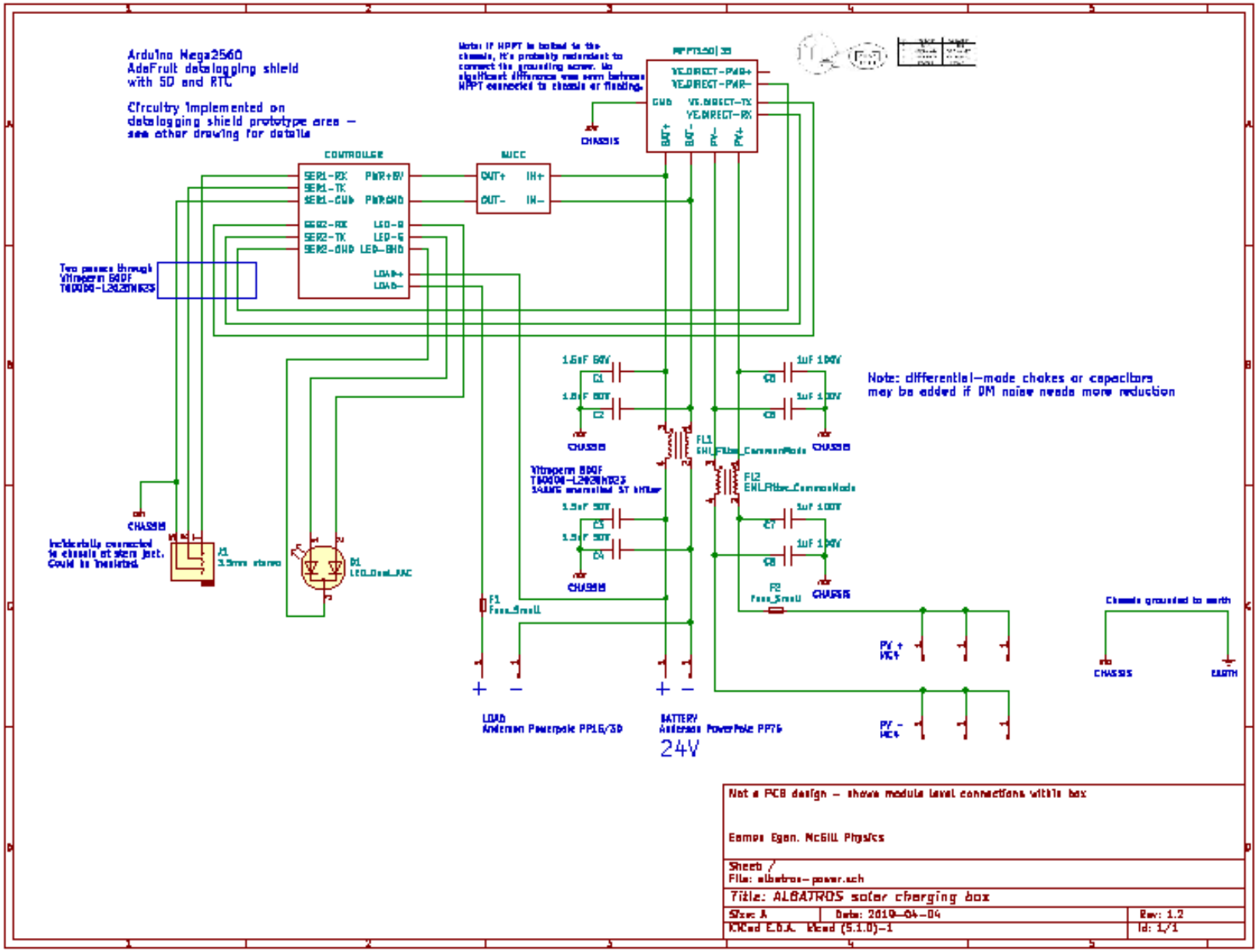
Ability to configure Victron parameters via hex protocol. This is a convenience instead of having to use a smartphone.

TBDs, Random Notes

It is possible to monitor battery voltage using the ATMEGA 10 bit ADC. However the Victron monitors battery voltage, so this is unlikely to be useful. TBD if we want this for battery protection insurance.

Warning to system of impending power down? From prior experience, this is not necessary.

Appendix – System Diagram



Appendix – MPPT Error Codes

Transcribed from https://www.victronenergy.com/live/mppt-error-codes. All entries are left in, including ones that are not listed in the MPPT 150/35 manual, just in case the manual is not up to date.

Err 2 - Battery voltage too high

This error will auto-reset after the battery voltage has dropped. This error can be due to other charging equipment connected to the battery or a fault in the charge controller. This error can also occur if the battery voltage is set to a lower voltage than the actual battery voltage.

Err 3, Err 4 - Remote temperature sensor failure

Check if the T-sense connector is properly connected to a remote temperature sensor. Most likely cause: the remote T-sense connector is connected to the BAT+ or BAT- terminal. This error will auto-reset after proper connection.

Err 5 - Remote temperature sensor failure (connection lost)

Check if the T-sense connector is properly connected to a remote temperature sensor. This error will not auto-reset.

Err 6, Err 7 - Remote battery voltage sense failure

Check if the V-sense connector is properly connected to the battery terminals. Most likely cause: the remote V-sense connector is connected in reverse polarity to the BAT+ or BAT- terminals.

Err 8 - Remote battery voltage sense failure (connection lost)

Check if the V-sense connector is properly connected to the battery terminals.

Err 17 - Controller overheated despite reduced output current

This error will auto-reset after charger has cooled down. Check the ambient temperature and check for obstructions near the heatsink.

Err 18 - Controller over-current

This error will auto-reset. If the error does not auto-reset disconnect the charge controller from all power-sources, wait 3 minutes, and power up again. If the error persists the charge controller is probably faulty. A cause for this error can be switching on a very large load on the battery side.

Err 20 - Maximum Bulk-time exceeded

This error can only occur when the maximun bulk-time protection is active. This error will not auto-reset. This error is generated when the battery-absorption-voltage is not reached after 10 hours of charging.

This protection is default disabled in all Solar Chargers.

It is default enabled on the Skylla-i and the Skylla IP44.

Our advice is to not enable it on solar chargers.

Err 21 - Current sensor issue

This error will not auto-reset.

Disconnect all wires, and then reconnect all wires. Also, make sure the minus on the MPPT charge controller (PV minus/Battery minus) is not bypassing the charge controller.

If the error remains, please contact the dealer, there might be a hardware defect.

Err 26 - Terminal overheated

Power terminals overheated, check wiring, including the wiring type and type of strands, and/or fasten bolts if possible.

This error will auto-reset.

Err 28 - Power stage issue

This error will not auto-reset.

Disconnect all wires, and then reconnect all wires. If the error persists the charge controller is probably faulty.

Err 33 - PV over-voltage

This error will auto-reset after PV-voltage has dropped to safe limit. This error is an indication that the PV-array configuration with regard to open-circuit voltage is critical for this charger. Check configuration, and if required, re-organise panels.

Err 34 - PV over-current

The current from the solar-panel array has exceeded 75A. This error could be generated due to an internal system fault. Disconnect the charger from all power-sources, wait 3 minutes, and power-up again. If the error persists the controller is probably faulty, contact your dealer.

Err 38, Err 39 - PV Input shutdown

To protect the battery from over-charging the panel input is shorted.

Possible reasons for this error to occur:

- The Battery voltage (12/24/48V) is set, or auto-detected, incorrectly. Use VictronConnect to disable auto-detect and set the Battery Voltage to a fixed voltage.
- There is another device connected to the battery, which is configured to a higher voltage. For example a MultiPlus, configured to equalise at 17 Volts, while in the MPPT this is not configured.
- The battery is disconnected using a manual switch. Ideally the charger should be switched off before disconnecting the battery, this avoids a voltage overshoot on the charger output. If necessary the voltage trip-level for the PV Short protection can be increased by raising

the Equalization voltage set-point (note: equalization does not have to be enabled in this case).
- The battery is disconnected using a Lithium charge relay connected to the “allow-to-charge” output of a BMS. Consider wiring this signal to the Remote terminal of the charger instead. This shuts down the charger gracefully without creating a voltage overshoot.

Error recovery:

- Error 38: First disconnect the solar panels and disconnect the battery. Wait for 3 minutes, then reconnect the battery first and next the panels.
- Error 39: The charger will automatically resume operation once the battery voltage drops below the float level for 1 minute.

If the error persists the charge controller is probably faulty.

Information 65 - Communication warning

Communication with one of the paralleled controllers was lost. To clear the warning, switch the controller off and back on

Information 66 - Incompatible device

The controller is being paralleled to another controller that has different settings and/or a different charge algorithm.

Make sure all settings are the same and update firmware on all chargers to the latest version

Err 67 - BMS Connection lost

The charger is configured to be controlled by a BMS, but it does not receive any control messages from a BMS. The charger stopped charging, as a safety precaution.

Check the connection between the charger and the BMS.

How to reset the charger, to de-couple it from the BMS

When the charger needs to operate in stand-alone mode again, not controlled by a BMS, it needs to be reset:

- VE.Can solar chargers, go into the setup menu, and change setting ‘BMS’ from ‘Y’ to ‘N’ (setup item 31).
- VE.Direct solar chargers, reset the charger to factory defaults with VictronConnect, and then reconfigure it.

Note that (solar-)chargers automatically configure themselves to be BMS-controlled when they are connected to one; either direct or via a Color Control GX or Venus GX.

Err 114 - CPU temperature too high

This error will reset after the CPU has cooled down. If the error persists, check the ambient temperature and check for obstructions near the air inlet and outlet holes of the charger cabinet.

Check manual for mounting instructions with regard to cooling. If error persists the controller is probably faulty.

Err 116 - Calibration data lost

If the unit does not work and error 116 pops up as the active error the unit is faulty, contact your dealer for a replacement.

If the error is only present in the history data and the unit operates normally this error can be ignored safely. Explanation: when the units power up for the very first time in the factory, it does not have calibration data and an error 116 is logged. Obviously this should have been cleared, but in the beginning units left the factory with this message still in the history data.