GenStar MPPT

MODBUS® Specification

Version 04

31 July 2026

Morningstar Corporation

8 Pheasant Run

Newtown, PA 18940

www.morningstarcorp.com

info@morningstarcorp.com

Overview

The **GenStar MPPT** supports the industry standard MODBUS® application protocol via USB, RS-232, RS-485, and Ethernet. This document assumes the user is familiar with the Modbus protocol and its terminology.

MODBUS® is a registered trademark of Modbus-IDA (www.modbus-ida.org)

Parameters

The **GenStar MPPT** supports RTU mode only. MODBUS addresses are 16-bit (per the modbus.org spec)

Serial communication parameters:

BPS	9600 baud
Parity	None
Data bits	8
Stop bits	2
Flow control	None

Default TCP communication parameters:

DHCP	enabled
Port	502
MODBUS ID	1
NetBIOS address	GSMPPT + serial number (no spaces)
LiveView Web address	http://GSMPPTXXXXXXX (where X is the serial number)

If DHCP fails, a link-local IP (a.k.a. auto-IP) will be automatically configured.

Supported Modbus Functions

Read Device Identification (0x2B, subcode 0x0E)

Read Coil (0x01); Read Discrete Inputs (0x02); Write Coil (0x05)

Read Holding Registers (0x03); Read Input Registers (0x04)

Write Single Register (0x06); Write Multiple Registers (0x10)

Any write to EEPROM will set an “EEPROM changed” fault. The control must be reset to clear this fault.

Note: No verify is performed on the write.

Values which span multiple registers (e.g., float32 and uint32 as described below) are stored in little-endian. E.g., if a uint32 of 0x12345678 is stored at SYSTEM_LOAD_AH_TOTAL10_0 (address 0x02e6) and SYSTEM_LOAD_AH_TOTAL10_1 (address 0x02e7), the _0 register will contain 0x5678 (the low word) and the _1 register will contain 0x1234 (the high word).

Terms

float32	The variable is stored as an IEEE748-2008 single-precision number in two adjacent registers. See the Examples section for conversion formulas.
float16	The variable is stored as an IEEE748-2008 half-precision number. See the Examples section for conversion formulas.
uint	The variable is stored as an **unsigned** integer. This means that the value can range from 0 to 65535. To represent larger numbers, a variable may be stored as two or more unsigned integers.
int	The variable is stored as a **signed** integer. This means that the value can range from -32768 to +32768.
norm	A normalized floating-point number between 0.0 and 1.0. If a variable represents a percent value, that means that the percentage is the stored value multiplied by 100. Example: BATT_SOC==0.5 means 50%.

Essential System Information

Addr	Name	Description
0x0000	SOFTWARE_VERSION	Software version. High byte == major version; low byte == minor version. E.g. software version 1.2 looks like (0x0102)
0x0001	SERIAL_NUMBER0	Serial number bytes 0-1
0x0002	SERIAL_NUMBER2	Serial number bytes 2-3
0x0003	SERIAL_NUMBER4	Serial number bytes 4-5
0x0004	SERIAL_NUMBER6	Serial number bytes 6-7
0x0005	SOFTWARE_PATCHLEVEL	Software patch level

Filtered ADC values with gain and offset applied

Addr	Name	Description	Units	Scaling
0x001f	ILOAD_F256	Load current	A	float16
0x0020	IARRAY_F256	Array current	A	float16
0x0021	VBTERM_F256	Battery voltage	V	float16
0x0022	VARRAY_F256	Array voltage	V	float16
0x0023	VLOAD_F256	Load voltage	V	float16
0x0025	VBSENSE_F256	Battery sense voltage	V	float16
0x002a	T_CASTING_F256	Lower aluminium casting temp	C	float16
0x002b	P12_F256	12V power supply	V	float16
0x002c	P3_F256	3V power supply	V	float16
0x002d	P3_EXTBB_F256	3V external power supply	V	float16
0x002e	P12_EXTBB_F256	12V external power supply	V	float16
0x0032	RTS_F256	Remote temperature sensor voltage	V	float16
0x0034	THS_F256	Heatsink temperature sensor voltage	V	float16
0x0035	P5_F256	5V power supply	V	float16
0x0036	V_COIN_F256	Coincell battery voltage	V	float16

0x0038	V_TEMP_F256	Internal temp sensor voltage	V	float16
0x0041	ILOAD_F256_0	Load current	A	float32
0x0042	ILOAD_F256_1			
0x0043	IARRAY_F256_0	Array current	A	float32
0x0044	IARRAY_F256_1			
0x0045	VBTERM_F256_0	Battery voltage	V	float32
0x0046	VBTERM_F256_1			
0x0047	VARRAY_F256_0	Array voltage	V	float32
0x0048	VARRAY_F256_1			
0x0049	VBSENSE_F256_0	Battery sense voltage	V	float32
0x004a	VBSENSE_F256_1			
0x0053	RTS_F256_0	RTS voltage normalized	V	float32
0x0054	RTS_F256_1			
0x00b8	ADC_VBATT_F_1M_FLOAT32_0	Battery voltage filtered 1min	V	float32
0x00b9	ADC_VBATT_F_1M_FLOAT32_1			
0x00ba	VBATT_MIN_DAILY_FLOAT32_0	Minimum battery voltage daily	V	float32
0x00bb	VBATT_MIN_DAILY_FLOAT32_1			
0x00bc	VBATT_MAX_DAILY_FLOAT32_0	Maximum battery voltage daily	V	float32
0x00bd	VBATT_MAX_DAILY_FLOAT32_1			
0x00be	VBATT_REF_CHARGE_FLOAT32_0	Charging target voltage	V	float32
0x00bf	VBATT_REF_CHARGE_FLOAT32_1			
0x00c0	VLOAD_F256_FLOAT32_0	Load voltage	V	float32
0x00c1	VLOAD_F256_FLOAT32_1			
0x0f4a	ADC_VARRAY_F_SLOW_FLOAT32_0	Array voltage, slow filtered	V	float32
0x0f4b	ADC_VARRAY_F_SLOW_FLOAT32_1			

System Currents

Addr	**Name**	**Description**	**Units**	**Scaling**
0x0063	SYSTEM_ICHARGE	System charging current	A	float16
0x0064	SYSTEM_IBATT	System battery current	A	float16
0x0065	SYSTEM_ILOAD	System load current	A	float16
0x0f4c	ADC_IARRAY_F_SLOW_FLOAT32_0	Array current, slow filtered	A	float32
0x0f4d	ADC_IARRAY_F_SLOW_FLOAT32_1			
0x0f4e	ADC_ICHARGE_F_SLOW_FLOAT32_0	Charge current, slow filtered	A	float32

0x0f4f	ADC_ICHARGE_F_SLOW_FLOAT32_1			
0x0f52	SYSTEM_IBATT_FLOAT32_0	System net battery current, slow filtered	A	float32
0x0f53	SYSTEM_IBATT_FLOAT32_1			
0x0f54	LOAD0_INTERNAL_AMPS_FLOAT32_0	Internal load current, slow filtered	A	float32
0x0f55	LOAD0_INTERNAL_AMPS_FLOAT32_1			

Temperature Readings

Addr	Name	Description	Units	Scaling
0x0075	TEMP_BATT	Battery temperature	C	float16
0x0076	TEMP_RTS	Remote temperature sensor temperature	C	float16
0x0077	TEMP_HEATSINK	Heatsink temperature	C	float16
0x0078	TEMP_INDCASTING	Inductor casting temperature	C	float16
0x0079	TEMP_INTERNAL	Internal uC temperature	C	float16
0x00c2	TEMP_BATT_FLOAT32_0	Battery temperature	C	float32
0x00c3	TEMP_BATT_FLOAT32_1			
0x00c4	TEMP_HEATSINK_FLOAT32_0	Heatsink temperature	C	float32
0x00c5	TEMP_HEATSINK_FLOAT32_1			

Charge Setpoints

Addr	Name	Description	Units	Scaling
0x0080	CHARGE_STATE	Charge state 0 = starting 1 = night check 2 = disconnected 3 = night 4 = fault 5 = bulk 6 = absorption 7 = float 8 = equalize		enum
0x0081	vBattRefLimit_0	Maximum regulator voltage	V	float32
0x0082	vBattRefLimit_1			
0x0083	vAbsorb_0	Absorption voltage (25C)	V	float32
0x0084	vAbsorb_1			
0x0085	iBattAbsorbTerm_0	Absorption ends-amps (25C)	A	float32
0x0086	iBattAbsorbTerm_1			
0x008b	tIBattAbsorbTerm	Absorption end-amps time	s	uint

0x008d	vFloat_0	Float voltage (25C)	V	float32
0x008e	vFloat_1			
0x008f	vFloatExit_0	Float exit voltage (25C)	V	float32
0x0090	vFloatExit_1			
0x0091	vFloatCancel_0	Float cancel voltage (25C)	V	float32
0x0092	vFloatCancel_1			
0x0093	vAbsorptionExt_0	Absorption extension threshold	V	float32
0x0094	vAbsorptionExt_1			
0x0097	tFloatExitCum	Float timeout	s	uint
0x0099	vEqualize_0	Equalize voltage (25C)	V	float32
0x009a	vEqualize_1			
0x009b	iEqualizeLimit_0	Equalize max current limit	A	float32
0x009c	iEqualizeLimit_1			
0x00a1	vHVD25_0	High voltage disconnect voltage (25C)	V	float32
0x00a2	vHVD25_1			
0x00a3	vHVR_0	High voltage reconnect voltage (25C)	V	float32
0x00a4	vHVR_1			
0x00a5	tAbsorption	Time to spend in Absorption	s	uint
0x00a6	tEqualize	Time to spend equalizing	s	uint
0x00a7	tEqualizeCalendar	Interval between auto-equalize cycles	d	uint
0x00a8	tEqualizeTO	Equalize timeout	s	uint
0x00a9	vTempComp_0	Temperature compensation coefficient	V/C	float32
0x00aa	vTempComp_1			
0x00ab	TBattMinLim	Minimum compensation temperature	C	float16
0x00ac	TBattMaxLim	Maximum compensation temperature	C	float16
0x00ad	temperatureFoldbackCold0	The cold temperature below which the charge controller current is 0%; NAN=disable	C	float16
0x00ae	temperatureFoldbackCold100	The cold temperature above which the charge controller current can be 100%; NAN=disable	C	float16
0x00af	temperatureFoldbackHot100	The hot temperature below which the charge controller current can be 100%; NAN=disable	C	float16
0x00b0	temperatureFoldbackHot0	The hot temperature above which the charge controller current is 0%;	C	float16

		NAN=disable		
0x00b1	VBATT_REF_CHARGE	Charging target voltage	V	float16
0x00b2	CHARGE_CONTROL_BMS_MODE	Charge Controller BMS Mode; 1 = BMS Mode is active		bool
0x00b4	VBATT_REF_CHARGE_SLAVE	Charging target voltage from external (slave) device	V	float16
0x00b5	HW_MAX_IARRAY	Hardware-specific max array current	A	float16
0x00b6	HW_MAX_ICHARGE	Hardware-specific max charge current	A	float16
0x00b7	HW_MAX_IBATT	Hardware-specific max batt current	A	float16

State of Charge and Days Since Counters

Addr	Name	Description	Units	Scaling
0x00e0	BATT_SOC	Battery state of charge	norm	float16
0x00e1	VBATT_MIN	Minimum battery voltage	V	float16
0x00e2	VBATT_MAX	Maximum battery voltage	V	float16
0x00e5	DAYS_SINCE_ABS	Number of days since absorption	d	uint
0x00e6	DAYS_SINCE_FLOAT	Number of days since float	d	uint
0x00e7	DAYS_SINCE_EQUALIZE	Number of days since equalize	d	uint
0x00e8	DAYS_SINCE_ATTEMPTED_EQUALIZE	Number of days since attempted equalize	d	uint
0x00e9	BATT_SOC_MIN_DAILY	Minimum battery state of charge daily	norm	float16
0x00ea	BATT_SOC_MAX_DAILY	Maximum battery state of charge daily	norm	float16
0x00c6	BATT_SOC_FLOAT32_0	Battery state of charge	norm	float32
0x00c7	BATT_SOC_FLOAT32_1			
0x00c8	BATT_SOC_MIN_DAILY_FLOAT32_0	Minimum battery state of charge daily	norm	float32
0x00c9	BATT_SOC_MIN_DAILY_FLOAT32_1			
0x00ca	BATT_SOC_MAX_DAILY_FLOAT32_0	Maximum battery state of charge daily	norm	float32
0x00cb	BATT_SOC_MAX_DAILY_FLOAT32_1			

PPT

Addr	Name	Description	Units	Scaling
0x00f0	POWER_OUT	Charging power out of control to battery	W	float16
0x00f1	SWEEP_VMP	Vmp found during sweep	V	float16
0x00f2	SWEEP_PIN_MAX	Pmax found during sweep	W	float16
0x00f3	SWEEP_VOC	Voc found during sweep	V	float16

0x00f4	VA_REF_F	Array voltage reference, filtered	V	float16
0x0f50	POWER_OUT_F_SLOW_FLOAT32_0	Charging power out of control to battery, slow filtered	W	float32
0x0f51	POWER_OUT_F_SLOW_FLOAT32_1			

Load

Addr	Name	Description	Units	Scaling
0x0100	LOAD_STATE	Load state 0 = start 1 = norm 2 = lvdwarn 3 = lvd 4 = fault 5 = disc 6 = norm_off 7 = override		enum
0x0101	LOAD_FAULT_0	Load fault bitfield 0: load_software 1: load_overcurrent 2: load_ex_short 3: load_power_supply 4: load_hvd 5: load_heatsink_overtemp 6: load_fet_shorted 7: load_system		bitfield32
0x0102	LOAD_FAULT_1			
0x0103	LOAD_FAULT_DAILY_0	Load fault daily bitfield 0: load_software 1: load_overcurrent 2: load_ex_short 3: load_power_supply 4: load_hvd 5: load_heatsink_overtemp 6: load_fet_shorted 7: load_system		bitfield32
0x0104	LOAD_FAULT_DAILY_1			
0x0105	SELECTED_V_LVD0	LVD voltage the control is running on	V	float16
0x0106	SELECTED_V_LVR	LVR voltage the control is running on	V	float16
0x0107	SELECTED_T_LVD_WARN	LVD warning time the control is running on	s	float16
0x0108	SELECTED_V_HVD_LOAD	HVD voltage for the load that the control is running on	V	float16
0x0109	SELECTED_V_HVR_LOAD	HVR voltage for the load that the control is running on	V	float16
0x010a	SELECTED_TMR_LVD_WARN	LVD warning timer; the time left until LVD	s	uint

0x010b	SELECTED_SOC_LOAD_DISCONNECT	SOC Disconnect the control is running on	norm	float16
0x010c	SELECTED_SOC_LOAD_RECONNECT	SOC Reconnect the control is running on	norm	float16
0x010d	SELECTED_SOC_LOAD_START	SOC start level	norm	float16
0x010f	SELECTED_V_START_LVD0	Voltage at start to go straight into LVD	V	float16
0x0110	SELECTED_LOAD_CURRENT_COMP	Positive load current compensation	ohms	float16

Faults and Alarms

Addr	**Name**	**Description**
0x0212	FAULT_0	System fault bitfield bit 0: EEPROM chip failure bit 2: software fault bit 3: no model number bit 6: EEPROM settings edit bit 8: BMS required bit 10: software reset bit 15: ADC overrun bit 16: ADC no inj bit 17: ADC double inj bit 18: battery shunt missing bit 29: EEPROM settings bad bit 30: hardware failure
0x0213	FAULT_1	
0x0214	FAULT_2	
0x0215	FAULT_3	
0x0216	FAULT_DAILY_0	System fault daily bitfield
0x0217	FAULT_DAILY_1	
0x0218	FAULT_DAILY_2	
0x0219	FAULT_DAILY_3	
0x021a	ALARM_0	Alarm bitfield bit 0: RTS open bit 1: RTS shorted bit 2: RTS disconnected bit 3: heatsink temperature sensor open bit 4: heatsink temperature sensor shorted bit 5: heatsink over-temperature limiting bit 6: current limiting bit 7: battery current offset/error (when FETs are off) bit 8: battery sense error bit 9: battery sense was working, but got disconnected bit 10: control is uncalibrated bit 11: RTS is close to 5v bit 17: high Va current limit

		bit 23: array current offset/error (when fets are off) bit 24: EEPROM or its bus had a failure bit 25: ethernet/controller hardware (initialization) problem. bit 26: entered LVD bit 27: software runtime error bit 29: external flash failure bit 32: I inductor current offset/error (when fets are off) bit 33: coincell battery voltage low bit 35: real-time clock is wrong bit 36: time clock hardware failure bit 37: casting temperature sensor open bit 38: casting temperature sensor shorted bit 39: inductor casting over-temperature limiting bit 41: wireless module is unresponsive or not working bit 42: hardware error post-startup bit 43: unknown block type detected bit 44: block1 general hardware alarm bit 45: block2 general hardware alarm bit 46: block3 general hardware alarm bit 50: generator failed to start within retry limit bit 51: backup generator failed to start within retry limit bit 52: schedule publisher failed to change state bit 53: uP is uncalibrated bit 54: load current offset/error (when fets are off) bit 55: load fet should be closed but is stuck open bit 56: bms sense error bit 57: bms sense disconnected
0x021b	ALARM_1	
0x021c	ALARM_2	
0x021d	ALARM_3	
0x021e	ALARM_DAILY_0	Alarm daily bitfield
0x021f	ALARM_DAILY_1	
0x0220	ALARM_DAILY_2	
0x0221	ALARM_DAILY_3	
0x0222	CHARGE_FAULT_0	Charge fault bitfield bit 0: software bug bit 1: battery HVD bit 2: RTS shorted bit 3: RTS disconnected bit 4: slave mode timeout
0x0223	CHARGE_FAULT_1	
0x0224	CHARGE_FAULT_2	
0x0225	CHARGE_FAULT_3	
0x0226	CHARGE_FAULT_DAILY_0	Charge fault daily bitfield

0x0227	CHARGE_FAULT_DAILY_1	
0x0228	CHARGE_FAULT_DAILY_2	
0x0229	CHARGE_FAULT_DAILY_3	
0x0232	FAULT_POWERSUPPLY_0	Power Supply fault bitfield bit 0: PS 12V bit 1: PS 5V bit 2: PS 3V bit 4: PS Array N3 bit 6: PS Battery N3 bit 7: PS Blockbus P12 bit 8: PS Blockbus P3 bit 9: PS Array P10 bit 10: PS Battery P10 bit 11: Array reverse polarity
0x0233	FAULT_POWERSUPPLY_1	
0x0234	FAULT_POWERSUPPLY_DAILY_0	Power Supply fault daily bitfield
0x0235	FAULT_POWERSUPPLY_DAILY_1	
0x0236	BLOCK_FAULT_0	ReadyBlock subsystem fault bitfield bit 0: ReadyBlock 1 Missing bit 1: ReadyBlock 2 Missing bit 2: ReadyBlock 3 Missing bit 8: ReadyBlock 1 HW Fail bit 9: ReadyBlock 2 HW Fail bit 10: ReadyBlock 3 HW Fail bit 16: ReadyBlock 1 HW Ver bit 17: ReadyBlock 2 HW Ver bit 18: ReadyBlock 3 HW Ver bit 24: ReadyBlock 1 Wiring bit 25: ReadyBlock 2 Wiring bit 26: ReadyBlock 3 Wiring
0x0237	BLOCK_FAULT_1	
0x0238	BLOCK_FAULT_DAILY_0	ReadyBlock subsystem fault daily bitfield
0x0239	BLOCK_FAULT_DAILY_1	
0x023a	FAULT_POWERSTAGE_0	Powerstage fault bitfield 0: powerstage_overcurrent 1: powerstage_fet_short 2: powerstage_dn_short 3: powerstage_batt_hvd 4: powerstage_array_hvd 5: powerstage_software 6: powerstage_bad_setting 7: powerstage_batt_lvd 8: powerstage_power_supply 9: powerstage_H7Rev 10: powerstage_hwrev

		11: powerstage_inductor_temp 12: powerstage_hs_temp 13: powerstage_inductor_offset 14: powerstage_array_offset 15: powerstage_misc
0x023b	FAULT_POWERSTAGE_1	
0x023c	FAULT_POWERSTAGE_2	
0x023d	FAULT_POWERSTAGE_3	
0x023e	FAULT_POWERSTAGE_DAILY_0	Powerstage fault daily bitfield
0x023f	FAULT_POWERSTAGE_DAILY_1	
0x0240	FAULT_POWERSTAGE_DAILY_2	
0x0241	FAULT_POWERSTAGE_DAILY_3	

Daily Log

Addr	Name	Description	Units	Scaling
0x026a	VBATT_MIN_HOURLY	Minimum battery voltage hourly	V	float16
0x026b	VBATT_MAX_HOURLY	Maximum battery voltage hourly	V	float16
0x026c	VBATT_MIN_DAILY	Minimum battery voltage daily	V	float16
0x026d	VBATT_MAX_DAILY	Maximum battery voltage daily	V	float16
0x026e	VARRAY_MAX_DAILY	Maximum array voltage daily	V	float16
0x0271	POUT_MAX_DAILY	Maximum output power daily	W	float16
0x0272	TEMP_BATT_MIN_DAILY	Minimum battery temp daily	V	float16
0x0273	TEMP_BATT_MAX_DAILY	Maximum battery temp daily	V	float16
0x0274	TIME_ABSORB_DAILY_0	Time battery spent in absorption daily	s	float32
0x0275	TIME_ABSORB_DAILY_1			
0x0276	TIME_EQ_DAILY_0	Time battery spent in equalize daily	s	float32
0x0277	TIME_EQ_DAILY_1			
0x0278	TIME_FLOAT_DAILY_0	Time battery spent in float daily	s	float32
0x0279	TIME_FLOAT_DAILY_1			

LED State

Addr	Name	Description
0x02a0	LED_STATE	State of LED indications
0x02a1	SOC_LED_STATE	State of SOC LEDs

Time and Date

Addr	Name	Description	Units	Scaling
0x02a6	HOURMETER_0	How long the control has ever been on	h	uint
0x02a7	HOURMETER_1			
0x02a8	REALTIMECLOCK_TIMEU_0	Current local time as a TimeU value. TimeU is the number of seconds since 2000-Jan-01 00:00:00 (UTC). If you have a time in the Unix epoch (1970-Jan-01 00:00:00 UTC) and want to convert to TimeU, subtract 946 684 800 seconds; to go the other way, add that many seconds.	s	uint
0x02a9	REALTIMECLOCK_TIMEU_1			
0x02aa	REALTIMECLOCK_UTC_OFFSET	UTC offset	min	int

System Counters

Addr	Name	Description	Units	Scaling
0x02d0	SYSTEM_CHARGE_KWH_DAILY_0	System charge kWh daily	kWh	float32
0x02d1	SYSTEM_CHARGE_KWH_DAILY_1			
0x02d2	SYSTEM_CHARGE_KWH_RESETTABLE10_0	System charge kWh resettable	kWh	uint*0.1
0x02d3	SYSTEM_CHARGE_KWH_RESETTABLE10_1			
0x02d4	SYSTEM_CHARGE_KWH_TOTAL10_0	System charge kWh total	kWh	uint*0.1
0x02d5	SYSTEM_CHARGE_KWH_TOTAL10_1			
0x02d6	SYSTEM_CHARGE_AH_DAILY_0	System charge Ah daily	Ah	float32
0x02d7	SYSTEM_CHARGE_AH_DAILY_1			
0x02d8	SYSTEM_CHARGE_AH_RESETTABLE10_0	System charge Ah resettable	Ah	uint*0.1
0x02d9	SYSTEM_CHARGE_AH_RESETTABLE10_1			
0x02da	SYSTEM_CHARGE_AH_TOTAL10_0	System charge Ah total	Ah	uint*0.1
0x02db	SYSTEM_CHARGE_AH_TOTAL10_1			
0x02dc	SYSTEM_BATT_AH_DAILY_0	System battery Ah daily	Ah	float32
0x02dd	SYSTEM_BATT_AH_DAILY_1			
0x02de	SYSTEM_BATT_AH_RESETTABLE10_0	System battery Ah resettable	Ah	int*0.1
0x02df	SYSTEM_BATT_AH_RESETTABLE10_1			
0x02e0	SYSTEM_BATT_AH_TOTAL10_0	System battery Ah total	Ah	int*0.1
0x02e1	SYSTEM_BATT_AH_TOTAL10_1			
0x02e2	SYSTEM_LOAD_AH_DAILY_0	System load Ah daily	Ah	float32

0x02e3	SYSTEM_LOAD_AH_DAILY_1			
0x02e4	SYSTEM_LOAD_AH_RESETTABLE10_0	System load Ah resettable	Ah	uint*0.1
0x02e5	SYSTEM_LOAD_AH_RESETTABLE10_1			
0x02e6	SYSTEM_LOAD_AH_TOTAL10_0	System load Ah total	Ah	uint*0.1
0x02e7	SYSTEM_LOAD_AH_TOTAL10_1			

Internal Counters

Addr	Name	Description	Units	Scaling
0x02e8	INTERNAL_CHARGE_KWH_DAILY_0	Internal charge kWh daily	kWh	float32
0x02e9	INTERNAL_CHARGE_KWH_DAILY_1			
0x02ea	INTERNAL_CHARGE_KWH_RESETTABLE10_0	Internal charge kWh resettable	kWh	uint*0.1
0x02eb	INTERNAL_CHARGE_KWH_RESETTABLE10_1			
0x02ec	INTERNAL_CHARGE_KWH_TOTAL10_0	Internal charge kWh total	kWh	uint*0.1
0x02ed	INTERNAL_CHARGE_KWH_TOTAL10_1			
0x02ee	INTERNAL_CHARGE_AH_DAILY_0	Internal charge Ah daily	Ah	float32
0x02ef	INTERNAL_CHARGE_AH_DAILY_1			
0x02f0	INTERNAL_CHARGE_AH_RESETTABLE10_0	Internal charge Ah resettable	Ah	uint*0.1
0x02f1	INTERNAL_CHARGE_AH_RESETTABLE10_1			
0x02f2	INTERNAL_CHARGE_AH_TOTAL10_0	Internal charge Ah total	Ah	uint*0.1
0x02f3	INTERNAL_CHARGE_AH_TOTAL10_1			
0x02f4	INTERNAL_BATT_AH_DAILY_0	Internal battery Ah daily	Ah	float32
0x02f5	INTERNAL_BATT_AH_DAILY_1			
0x02f6	INTERNAL_BATT_AH_RESETTABLE10_0	Internal battery Ah resettable	Ah	int*0.1
0x02f7	INTERNAL_BATT_AH_RESETTABLE10_1			
0x02f8	INTERNAL_BATT_AH_TOTAL10_0	Internal battery Ah total	Ah	int*0.1
0x02f9	INTERNAL_BATT_AH_TOTAL10_1			
0x02fa	LOAD0_AH_DAILY_0	Internal (load0) Ah daily	Ah	float32
0x02fb	LOAD0_AH_DAILY_1			
0x02fc	LOAD0_AH_RESETTABLE10_0	Internal (load0) Ah resettable	Ah	uint*0.1
0x02fd	LOAD0_AH_RESETTABLE10_1			
0x02fe	LOAD0_AH_TOTAL10_0	Internal (load0) Ah total	Ah	uint*0.1
0x02ff	LOAD0_AH_TOTAL10_1			

Network Information

Addr	Name	Description
0x0f00	NETBIOS_NAME_0	NetBIOS name; The format is <device-prefix><serial-number>
0x0f01	NETBIOS_NAME_1	
0x0f02	NETBIOS_NAME_2	
0x0f03	NETBIOS_NAME_3	
0x0f04	NETBIOS_NAME_4	
0x0f05	NETBIOS_NAME_5	
0x0f06	NETBIOS_NAME_6	
0x0f07	NETBIOS_NAME_7	
0x0f08	CURRENT_IP_ADDR_0	Current network IP address in use expressed as a 32-bit unsigned integer in host byte-order (typically little-endian). Other network addresses follow this word order. E.g., 192.168.10.170 would appear as IP_ADDR_0==0x0AAA, IP_ADDR_1==0xC0A8
0x0f09	CURRENT_IP_ADDR_1	
0x0f0a	CURRENT_NET_MASK_0	Current subnet mask in use
0x0f0b	CURRENT_NET_MASK_1	
0x0f0c	CURRENT_DEFAULT_GW_0	Current default gateway in use
0x0f0d	CURRENT_DEFAULT_GW_1	
0x0f0e	CURRENT_PRIMARY_DNS_0	Current primary DNS in use
0x0f0f	CURRENT_PRIMARY_DNS_1	
0x0f10	CURRENT_SECONDARY_DNS_0	Current secondary DNS in use
0x0f11	CURRENT_SECONDARY_DNS_1	

IoT Status

Addr	Name	Description
0x0f14	IOT_CONNECTION_STATUS	IoT Connection Status 0 = Not Connected 1 = Connected
0x0f15	IOT_REGISTRATION_STATUS	IoT Registration Status 0 = Not Registered 1 = Registered 2 = Unknown registration status
0x0f16	IOT_CERT_STATUS	IoT Certificate Status 0 = No Client Certificate in extflash

		1 = Client Certificate found in extflash

SD Card

Addr	Name	Description
0x0f20	SD_INSERTED	SD card information Set to 1 if SD card is physically in slot, 0 otherwise
0x0f21	SD_SIZE_0	If SD card is init and functioning, size in 512-byte blocks; 0 otherwise. Divide by 2048 to get size in megabytes.
0x0f22	SD_SIZE_1	

Charge Settings

Addr	Name	Description	Units	Scaling
0x1000	EchargevBattRefLimit	Maximum regulator voltage; 0==disable	V	float32
0x1002	EchargevAbsorb	Absorption voltage target	V	float32
0x1004	EchargeiBattAbsorbTerm	Absorption end-amps; The net battery current that will terminate absorption (based on timer)	V	float16
0x1008	EchargetIBattAbsorbTerm	Absorption end-amps time; Absorption termination will occur when the net battery current is below the end-amps threshold for this long	sec	uint
0x100c	EchargevFloatExit	Voltage trip to exit float prematurely	V	float32
0x100e	EchargevFloatCancel	Voltage trip to cancel float	V	float32
0x1010	EchargevAbsorptionExt	Absorption extension threshold	V	float32
0x1012	EchargetAbsorptionExt	Time to spend in Absorption extension	sec	uint
0x1015	EchargevEqualize	Equalize voltage target; 0==disable	V	float32
0x1017	EchargeiEqualizeLimit	Equalize max current limit; 0==disable	A	float32
0x101d	EchargevHVD25	High voltage disconnect voltage (25C)	V	float32
0x101f	EchargevHVR	High voltage reconnect voltage	V	float32
0x1021	EchargetAbsorption	Time to spend in Absorption	sec	uint
0x1022	EchargetEqualize	Time to spend equalizing	sec	uint
0x1023	EchargetEqualizeCalendar	Interval between auto-equalize cycles	days	uint
0x1024	EchargetEqualizeTO	Equalize timeout	sec	uint
0x1025	EchargevTempComp	Temperature compensation coefficient	V/C	float16
0x1026	EchargeTBattMinLim	Minimum compensation temperature	C	float16
0x1027	EchargeTBattMaxLim	Maximum compensation temperature	C	float16

0x1028	EchargetemperatureFoldbackCold0	the cold temperature below which the charge controller current is 0%; NAN==disable	C	float16
0x1029	EchargetemperatureFoldbackCold100	the cold temperature above which the charge controller current can be 100%; NAN==disable	C	float16
0x102a	EchargetemperatureFoldbackHot100	the hot temperature below which the charge controller current can be 100%; NAN==disable	C	float16
0x102b	EchargetemperatureFoldbackHot0	the hot temperature above which the charge controller current is 0%; NAN==disable	C	float16

Misc Settings

Addr	Name	Description
0x105c	Egreen_mode	Ethernet green mode. Enabling this attempts to save power in the Ethernet system when possible, e.g., turning off power to unused ports. 1==enable, 0==disable
0x1060	EBatteryType	Battery Type; See the latest revision manual on our website for battery type definitions
0x1062	EbatteryServiceReminderInterval	Battery service timer: how often to remind the user to service their battery. Setting this to 0 disables the reminder.
0x1063	EBMSRequired	If nonzero, a BMS is required in order to function
0x1068	Emodbus_id	This device's Modbus address (1 to 247)
0x1069	Ebaudrate_mb232	Baudrate for serial Modbus; E.g., 96 means 9600 baud

PPT limits

Addr	Name	Description	Units	Scaling
0x106b	Esweep_va_min_limit	Minimum array voltage to limit MPPT sweeps to; set to V_SWEEP_MIN for normal behavior	V	float16
0x106c	Esweep_va_max_limit	Maximum array voltage to limit MPPT sweeps to; set to INFINITY for normal behavior	V	float16
0x106d	EVa_ref_fixed	if non-zero, runs in fixed Vmp mode at this array voltage	V	float16
0x106e	EVa_ref_fixed_pct	If non-zero, run in fixed %Voc mode at this fraction of open-circuit array voltage; E.g., 0.5 means 50% of Voc	norm	float16

Load Setpoints

Addr	Name	Description	Units	Scaling
0x10a0	Eload0VLVD0	Load low voltage disconnect	V	float16
0x10a1	Eload0VLVR	Load low voltage reconnect	V	float16
0x10a2	Eload0vStartLVD	Voltage at start to go straight into LVD	V	float16
0x10a3	Eload0HVD	The HVD used in the load controller	V	float16
0x10a4	Eload0HVR	The HVR used in the load controller	V	float16
0x10a5	Eload0SOCDisconnect	Load low SOC disconnect	norm	float16
0x10a6	Eload0SOCReconnect	Load low SOC reconnect	norm	float16
0x10a8	Eload0tLVDWarn	Load low voltage disconnect warning timer.	sec	uint
0x10aa	Eload0currentComp	Battery resistance for LVD current compensation; this value will always be positive	ohms	float16

Network Settings

Addr	Name	Description
0x1110	EIPAddr	This device's static IP address, in host (little-endian) order. Only meaningful if DHCP is disabled.
0x1112	ENetMask	This device's static IP netmask. Only meaningful if DHCP is disabled
0x1114	EGateway	This device's static IP gateway. Only meaningful if DHCP is disabled
0x1116	EPrimaryDNSServer	This device's static IP primary DNS server to use. Only meaningful if DHCP is disabled
0x1118	ESecondaryDNSServer	This device's static IP secondary DNS server to use. Only meaningful if DHCP is disabled
0x1122	EIsDHCPEnabled	DHCP status. If DHCP is disabled, the control will use static IP settings. 0==disabled 1==enabled
0x1125	EHTTPPort	HTTP port number
0x1126	EMBIPPort	Modbus IP port number
0x1127	EModbusTcpBridgingEnabled	Modbus bridging status 0==disabled 1==enabled
0x112b	EWirelessEnable	Wireless communications mode. Depending on value, the corresponding wireless radio will be turned on and will advertise for connections, etc. When this value is written, it applies immediately without requiring a reboot.

		0==disabled 1==Bluetooth ON
0x112c	EIoTEnable	Enable/disable IoT. Determines whether the device attempts to contact the cloud. Applied immediately on change.. Note: Invalid values (everything except 0 and 1) will be treated as 0 (disabled) 0==disabled 1==enabled
0x1150	ENTP1_Server	NTP Server 1 (254 (0x7f words) character FQDN)
0x11cf	ENTP2_Server	NTP Server 2 (254 character FQDN)
0x124e	ENTP3_Server	NTP Server 3 (254 character FQDN)
0x12d3	EenableEthernetModbusWrites	Allow/Disallow Modbus writes over ethernet. When enabled, an error will be returned if a user attempts to write modbus via TCP/IP or via HTTP. When this value is written, it applies immediately without requiring a reboot. 1==enabled 0==disabled

SOC Settings

Addr	Name	Description	Units	Scaling
0x12eb	ESocEfficiencyFactor	While coulomb counting, this percent is used to adjust the amp-hours charged and discharged before they are included in the count.	norm	float16
0x12ec	ESocCurrentSetpointPercent	This value, when multiplied by the total battery capacity, determines the net battery current setpoint (in amps) used by the coulomb counter; If the net battery current falls below this setpoint, the battery is considered to be fully charged; The default value is 0.02 (2%)	norm	float16
0x12ed	EDurationUntil100Soc	100% SOC is set when the 100% SOC conditions are met for this amount of time.	s	uint

SNMP Settings

Addr	Name	Description
0x1391	ESnmpEnabledVersions	Bitfield describing which SNMP versions are enabled; 0 means SNMP is disabled bit 0: SNMPv1 bit 1: SNMPv2c bit 2: SNMPv3
0x1392	ESnmpV3DefaultUserEnabled	Whether or not the SNMPv3 default user is enabled. If set to 1, a default user with known credentials will be created on boot

		and/or when a custom Engine ID is set. Its credentials can be changed by the user, but will be reset to the default values if the Engine ID changes. Default credentials: Username: "defaultUser" Auth password: "defaultAuthPwd" auth algorithm: SHA Priv password: "defaultPrivPwd" priv algorithm: AES
0x14a0	ESnmpReadCommunity	SNMP read community string, used for authentication
0x14c0	ESnmpWriteCommunity	SNMP write community string, used for authentication

Read-only Settings

Addr	Name	Description	Units	Scaling
0x2200	Ehourmeter	How long the control has ever been on	h	uint
0x2202	EAhNetResettable10	Net Ah resettable	Ah	int*0.1
0x2204	EAhNetTotal10	Net Ah total	Ah	int*0.1
0x2211	Esw_Version	Stored software version in high-byte-major low-byte-minor format. This is used to determine software upgrades at startup.		
0x2213	EvBatt_min	Vbatt minimum log	V	float16
0x2214	EvBatt_max	Vbatt maximum log	V	float16
0x2215	EvArray_max	Varray maximum log	V	float16
0x2216	Etmr_eqcalendar	Interval between auto eq cycles; 0==disable	d	uint

Local Counters

Addr	Name	Description	Units	Scaling
0x2260	EChargekWhResettable	Local charge kWh counter resettable	kWh	uint
0x2262	EChargekWhTotal	Local charge kWh counter total	kWh	uint
0x2264	EChargeAhResettable	Local charge Ah counter resettable	Ah	uint
0x2266	EChargeAhTotal	Local charge Ah counter total	Ah	uint
0x2268	EBatteryAhResettable	Local battery Ah counter resettable	Ah	int
0x226a	EBatteryAhTotal	Local battery Ah counter total	Ah	int

Aggregated Shunt Counters

Addr	Name	Description	Units	Scaling
0x227c	EAggregatedShuntsChargeAhDaily	Shunt-level (all configured external charge	Ah	float32

		shunts) charge amp-hours, as counted just for today; updated approximately every hour; useful for persisting data across reboots		
0x227e	EAggregatedShuntsChargeAhResettable10	Shunt-level (all configured external charge shunts) charge amp-hours, "resettable" version; updated approximately every hour; useful for persisting data across reboots	Ah	uint*0.1
0x2280	EAggregatedShuntsChargeAhTotal10	Shunt-level (all configured external charge shunts) charge amp-hours, "total" version; updated approximately every hour; useful for persisting data across reboots	Ah	uint*0.1
0x2282	EAggregatedShuntsChargekWhDaily	Shunt-level (all configured external charge shunts) charge energy, as counted just for today; updated approximately every hour; useful for persisting data across reboots	kWh	float32
0x2284	EAggregatedShuntsChargekWhResettable10	Shunt-level (all configured external charge shunts) charge energy, "resettable" version; updated approximately every hour; useful for persisting data across reboots	kWh	uint*0.1
0x2286	EAggregatedShuntsChargekWhTotal10	Shunt-level (all configured external charge shunts) charge energy, "total" version; updated approximately every hour; useful for persisting data across reboots	kWh	uint*0.1
0x2288	EAggregatedShuntsBattAhDaily	Shunt-level (all configured external charge shunts) battery (net) amp- hours, as counted just for today; updated approximately every hour; useful for persisting data across reboots	Ah	float32
0x228a	EAggregatedShuntsBattAhResettable10	Shunt-level (all configured external charge shunts) battery (net) amp- hours, "resettable" version; updated approximately every hour; useful for persisting data across reboots	Ah	int*0.1
0x228c	EAggregatedShuntsBattAhTotal10	Shunt-level (all configured external charge shunts) battery (net) amp- hours, "total" version; updated approximately every hour; useful for persisting data across reboots	Ah	int*0.1
0x228e	EAggregatedShuntsLoadAhDaily	Shunt-level (all configured external charge shunts) load amp-hours, as counted just for today; updated approximately every hour; useful for persisting data across reboots	Ah	float32
0x2290	EAggregatedShuntsLoadAhResettable10	Shunt-level (all configured external charge shunts) load amp-hours, "resettable"	Ah	uint*0.1

		version; updated approximately every hour; useful for persisting data across reboots		
0x2292	EAggregatedShuntsLoadAhTotal10	Shunt-level (all configured external charge shunts) load amp-hours, "total" version; updated approximately every hour; useful for persisting data across reboots	Ah	uint*0.1

SNMPv3 User Management API

Addr	Name	Description
0x5000	SnmpV3DeleteUserResult	Result of the last SNMPv3 user deletion API usage. It contains 2 octets: The first is a counter describing how many times the API has been used, and the second is the result of the last API usage. @see ESnmpUserDeletionErr to understand the second octet (0=success)
0x5001	SnmpV3DeleteUserName_0	Write this array of registers to the SNMPv3 username you want to delete. The username must be null-terminated.
0x5002	SnmpV3DeleteUserName_1	
0x5003	SnmpV3DeleteUserName_2	
0x5004	SnmpV3DeleteUserName_3	
0x5005	SnmpV3DeleteUserName_4	
0x5006	SnmpV3DeleteUserName_5	
0x5007	SnmpV3DeleteUserName_6	
0x5008	SnmpV3DeleteUserName_7	
0x5009	SnmpV3DeleteUserName_8	
0x500a	SnmpV3DeleteUserName_9	
0x500b	SnmpV3DeleteUserName_10	
0x500c	SnmpV3DeleteUserName_11	
0x500d	SnmpV3DeleteUserName_12	
0x500e	SnmpV3DeleteUserName_13	
0x500f	SnmpV3DeleteUserName_14	
0x5010	SnmpV3DeleteUserName_15	
0x5011	SnmpV3NewUserResult	Result of the last SNMPv3 user creation API usage. It contains 2 octets: The first is a counter describing how many times the API has been used, and the second is the result of the last API usage. @see ESnmpUserCreationErr to understand the second octet

		(0=success)
0x5012	SnmpV3NewUserName_0	Write this array of registers to the SNMPv3 username you want to create or modify.
0x5013	SnmpV3NewUserName_1	
0x5014	SnmpV3NewUserName_2	
0x5015	SnmpV3NewUserName_3	
0x5016	SnmpV3NewUserName_4	
0x5017	SnmpV3NewUserName_5	
0x5018	SnmpV3NewUserName_6	
0x5019	SnmpV3NewUserName_7	
0x501a	SnmpV3NewUserName_8	
0x501b	SnmpV3NewUserName_9	
0x501c	SnmpV3NewUserName_10	
0x501d	SnmpV3NewUserName_11	
0x501e	SnmpV3NewUserName_12	
0x501f	SnmpV3NewUserName_13	
0x5020	SnmpV3NewUserName_14	
0x5021	SnmpV3NewUserName_15	
0x5022	SnmpV3NewUserAuthAlgo	Write this register to specify the SNMPv3 authentication protocol for the user you're creating or modifying @see snmpv3_auth_algo_t
0x5023	SnmpV3NewUserAuthPwd_0	Write this array of registers to specify the SNMPv3 authentication password for the user you're creating or modifying. The password must be null-terminated. It will be hashed according to the authentication algorithm and reset after the creation process is complete
0x5024	SnmpV3NewUserAuthPwd_1	
0x5025	SnmpV3NewUserAuthPwd_2	
0x5026	SnmpV3NewUserAuthPwd_3	
0x5027	SnmpV3NewUserAuthPwd_4	
0x5028	SnmpV3NewUserAuthPwd_5	
0x5029	SnmpV3NewUserAuthPwd_6	
0x502a	SnmpV3NewUserAuthPwd_7	
0x502b	SnmpV3NewUserAuthPwd_8	
0x502c	SnmpV3NewUserAuthPwd_9	

0x502d	SnmpV3NewUserAuthPwd_10	
0x502e	SnmpV3NewUserAuthPwd_11	
0x502f	SnmpV3NewUserAuthPwd_12	
0x5030	SnmpV3NewUserAuthPwd_13	
0x5031	SnmpV3NewUserAuthPwd_14	
0x5032	SnmpV3NewUserAuthPwd_15	
0x5033	SnmpV3NewUserAuthPwd_16	
0x5034	SnmpV3NewUserAuthPwd_17	
0x5035	SnmpV3NewUserAuthPwd_18	
0x5036	SnmpV3NewUserAuthPwd_19	
0x5037	SnmpV3NewUserAuthPwd_20	
0x5038	SnmpV3NewUserAuthPwd_21	
0x5039	SnmpV3NewUserAuthPwd_22	
0x503a	SnmpV3NewUserAuthPwd_23	
0x503b	SnmpV3NewUserAuthPwd_24	
0x503c	SnmpV3NewUserAuthPwd_25	
0x503d	SnmpV3NewUserAuthPwd_26	
0x503e	SnmpV3NewUserAuthPwd_27	
0x503f	SnmpV3NewUserAuthPwd_28	
0x5040	SnmpV3NewUserAuthPwd_29	
0x5041	SnmpV3NewUserAuthPwd_30	
0x5042	SnmpV3NewUserAuthPwd_31	
0x5043	SnmpV3NewUserAuthPwd_32	
0x5044	SnmpV3NewUserPrivAlgo	Write this register to specify the SNMPv3 privacy protocol for the user you're creating or modifying. @see snmpv3_priv_algo_t
0x5045	SnmpV3NewUserPrivPwd_0	Write this array of registers to specify the SNMPv3 privacy password for the user you're creating or modifying. The password must be null-terminated. It will be hashed according to the authentication algorithm and reset after the creation process is complete
0x5046	SnmpV3NewUserPrivPwd_1	
0x5047	SnmpV3NewUserPrivPwd_2	
0x5048	SnmpV3NewUserPrivPwd_3	

0x5049	SnmpV3NewUserPrivPwd_4	
0x504a	SnmpV3NewUserPrivPwd_5	
0x504b	SnmpV3NewUserPrivPwd_6	
0x504c	SnmpV3NewUserPrivPwd_7	
0x504d	SnmpV3NewUserPrivPwd_8	
0x504e	SnmpV3NewUserPrivPwd_9	
0x504f	SnmpV3NewUserPrivPwd_10	
0x5050	SnmpV3NewUserPrivPwd_11	
0x5051	SnmpV3NewUserPrivPwd_12	
0x5052	SnmpV3NewUserPrivPwd_13	
0x5053	SnmpV3NewUserPrivPwd_14	
0x5054	SnmpV3NewUserPrivPwd_15	
0x5055	SnmpV3NewUserPrivPwd_16	
0x5056	SnmpV3NewUserPrivPwd_17	
0x5057	SnmpV3NewUserPrivPwd_18	
0x5058	SnmpV3NewUserPrivPwd_19	
0x5059	SnmpV3NewUserPrivPwd_20	
0x505a	SnmpV3NewUserPrivPwd_21	
0x505b	SnmpV3NewUserPrivPwd_22	
0x505c	SnmpV3NewUserPrivPwd_23	
0x505d	SnmpV3NewUserPrivPwd_24	
0x505e	SnmpV3NewUserPrivPwd_25	
0x505f	SnmpV3NewUserPrivPwd_26	
0x5060	SnmpV3NewUserPrivPwd_27	
0x5061	SnmpV3NewUserPrivPwd_28	
0x5062	SnmpV3NewUserPrivPwd_29	
0x5063	SnmpV3NewUserPrivPwd_30	
0x5064	SnmpV3NewUserPrivPwd_31	
0x5065	SnmpV3NewUserPrivPwd_32	
0x5066	SnmpV3NewEngineId_0	Write this array of registers to new SNMPv3 Engine ID you'd like to set After setting this array and SnmpV3NewEngineIdLen, write true to MBCOIL_SNMPV3_ENGINE_ID_CHANGE to apply the new

		Engine ID @see MBVAR_SnmpV3_NewEngineIdLen
0x5067	SnmpV3NewEngineId_1	
0x5068	SnmpV3NewEngineId_2	
0x5069	SnmpV3NewEngineId_3	
0x506a	SnmpV3NewEngineId_4	
0x506b	SnmpV3NewEngineId_5	
0x506c	SnmpV3NewEngineId_6	
0x506d	SnmpV3NewEngineId_7	
0x506e	SnmpV3NewEngineId_8	
0x506f	SnmpV3NewEngineId_9	
0x5070	SnmpV3NewEngineId_10	
0x5071	SnmpV3NewEngineId_11	
0x5072	SnmpV3NewEngineId_12	
0x5073	SnmpV3NewEngineId_13	
0x5074	SnmpV3NewEngineId_14	
0x5075	SnmpV3NewEngineId_15	
0x5076	SnmpV3NewEngineIdLen	Number of bytes in the user-configured SNMPv3 EngineID. The Engine ID must be between 5 and 32 bytes long. If this value is not set within that range, it will be ignored and the default Engine ID (based on the device's MAC address) will be used. The Engine ID must be unique to the device for its administrative domain. It must not be all 0x00 or 0xFF; if it is, the default Engine ID will be used instead. The MBCOIL_SNMPV3_ENGINE_ID_CHANGE coil must be set for the new Engine ID to take effect.

Modbus Coils

Addr	**Name**	**Description**
0x0000	EQTRIG	Manual equalize trigger; true==start equalize, false==stop equalize
0x0001	LOADDISCONNECT	System Load Disconnect (refers to DC load terminals and all external Loads on the battery); true==disconnect, false==reconnect
0x0002	CHARGEDISCONNECT	Charge disconnect; true==stop charge, false==restart charge
0x0014	CLEAR_COUNTERS_RESETTABLE	Resets all resettable counters; true==do action
0x0015	CLEAR_COUNTERS_TOTAL	Reset all total counters; true==do action

0x0016	CLEAR_VBMINMAX	Clear Vbatt min and max saved values; true==do action
0x0017	CLEAR_FAULTS	Clear faults; true==do action
0x0018	CLEAR_ALARMS	Clear alarms; true==do action
0x0020	EE_UPDATE	Forces eeprom update of logged variables (hourmeter/vbmin/vbmax/etc); true== do action
0x0021	LVD_OVERRIDE	Override low voltage disconnect; true==do action
0x00ff	RESET	Resets the controller without power cycling; true==do action
0x1903	SNMPV3_USER_CREATION	Read this coil to check if the SNMPv3 user creation API is being used (some of the modbusRAM registers have been set, but the data has not been processed yet). Write 0 to discard the input data, write 1 to process the data provided via the modbusRAM API and create/modify the user
0x1904	SNMPV3_USER_DELETION	Read this coil to check if the SNMPv3 user deletion API is being used (the username registers are being modified, but the data has not been processed yet). Write 0 to discard the input data, write 1 to process the data provided via the modbusRAM API and delete the user
0x1905	SNMPV3_ENGINEID_CHANGE	Read this coil to check if the SNMPv3 Engine ID change APi is being used (the engine ID registers are being modified, but the data has not been processed yet). Write 0 to discard the input data, write 1 to process the data provided via the modbusRAM API

Examples

C Float32 Conversion to Float16

```
// Convert a float32 (IEEE754 Single precision binary32) to float16 (IEEE754 half precision binary16)

#include <stdint.h>

#include <math.h>


uint16_t F32ConvertToF16(float f32) {
    uint16_t f16 = 0;
    uint32_t f32_u = *(uint32_t*)&f32;
    unsigned sign = (f32_u & 0x80000000) >> 31;
    unsigned exponent = (f32_u & 0x7f800000) >> 23;
    unsigned fraction = (f32_u & 0x007fffff) >> 13;
    if (exponent == 0xFF) {     // check for inf and NaN
        if (fraction == 0) { f16 = (sign == 1) ? 0xfc00 : 0x7c00; }
        else { f16 = 0x7c01; }
    } else {
        // verify the number if within range
        if (((int) exponent - 127) ≥ 15) {
            //overflow, if exponent values are too high, return inf
            f16 = (sign == 1) ? 0xfc00 : 0x7c00; // neg and pos inf
        } else if (((int) exponent - 127) ≤ -14) {
            //underflow
        } else {
            // normal numbers
            f16 = (sign ≪ 15) + ((unsigned)((int)exponent - 127) ≪ 10) + (fraction);
```

```
        }
    }
    return f16;
}
```

C Float16 Conversion to Float32

```
// Convert a float16 (IEEE754 half-precision binary16) to float32 (IEEE754 Single
// precision binary32)
#include <stdint.h>
#include <math.h>

float F16ConvertToF32(uint16_t f16) {
    float f32 = 0;
    unsigned sign = (f16 & 0x8000) >> 15;   // extract out the sign
    unsigned exponent = ((f16 & 0x7C00) >> 10); // extract out the exponent
    float fraction = (f16 & 0x03ff) / 1024.0;   // extract out the fraction

    if (exponent == 0x1f) {
        if (fraction == 0) {
            int positveInf = 0x7f800000;
            int negativeInf = 0xff800000;
            return (sign == 0) ? *(float*)&positveInf : *(float*)&negativeInf;
        } else { return 0.0 / 0.0; }  // use 0.0 to generate NaN
    }
    if (exponent == 0 ) {     //check for 0 or subnormal
        if (fraction == 0 ) { // if it is 0
            if (sign == 1 ) { return -0.0; }
            return 0.0;         // use 0.0 to return a zero in float
        } else {
            f32 = fraction * pow(2.0, -14.0);
            if (sign == 1) { f32 *= -1.0; }
```

```
            return f32;
        }
    }
    // the number is not a NaN or 0 or subnormal
    f32 = (fraction + 1.0) * pow(2.0, ((int)exponent - 15));
    if (sign == 1) { f32 *= -1.0; }
    return f32;
}
```

JavaScript Float16 conversion example

```
// Converts an integer read by modbus to float16 (IEEE754 half-precision binary16)
// Modbus result is likely sign extended but we will ignore that
function ScaleF16(passedVal) {
    var s = 0;      // sign
    var e = 0;      // exponent
    var currentVal = 0;     // mantissa/result

    currentVal = (passedVal & 0x03ff) / 1024.0; // 10 bit mantissa (normalized)
    passedVal >>= 10;
    e = (passedVal & 0x001f);   // 5 bit exponent (stored w/ 15 offset)
    passedVal >>= 5;
    s = passedVal & 0x0001;     // 1 bit sign

    if (e == 0) {   // zero or subnormal
        if (currentVal == 0) { return(0) };
        // subnormal (no leading 1.xxx)
        currentVal *= Math.pow(2.0,-14);
        if (s ≠ 0) { currentVal *= -1.0; }
        return (currentVal);
    }
    if (e == 0x1f) {            // infinity or NaN
        if (currentVal == 0) {
            if (s == 0) { return(Number.POSITIVE_INFINITY); }   // +infinity
            else { return (Number.NEGATIVE_INFINITY); } // -infinity
        } else { return(Number.NaN); } // NaN
```

```
    }
    currentVal += 1.0; // add in leading 1
    currentVal *= Math.pow(2.0,e-15);
    if (s ≠ 0) currentVal *= -1.0;

    return (currentVal);
}
```

Document Revision History

v01: First release
v02: Update doc title and add SOC Settings section
v03: Genstar v4.0 update; Added IoT and SNMP sections, plus additional miscellaneous clean up.
v04: GenStar v4.0.6 update; added float32 versions of some popular float16 variables for PLCs et al which don't natively support parsing float16.