eTactica

Know your energy

Modbus Register Map

EM – eTactica Power Meter

Revision history

Version	Action	Author	Date
1.0	Initial document	KP	25.08.2013
1.1	Document review, description and register update	GP	26.08.2013
1.2	Status bits, current noise floor	GP	29.08.2013
1.3	Using EG100 as a Modbus TCP/RTU bridge	GP	28.10.2013
1.4	Separate document for each device	GP	07.07.2014
1.5	Brand changed to eTactica, names of meters changed, the command register was updated.	RE/ÁH	21.06.2016
2.0	Added EM2 extended registers, timing, addressing	KP	26.01.2018
2.1	Clarified definitions of cumulatives	KP	10.04.2018

Introduction

All eTactica hardware devices are standard Modbus/RTU server devices, with a half-duplex RS485 serial interface.

This document covers the following products:
- EM-SC: the eTactica Power Meter with split-core current sensors
- EM-FC: the eTactica Power Meter with flexible-coil current sensors

The eTactica measurement devices implement a register table with both configurable and read-only parameters. These parameter values are accessible via standard Modbus requests.

As the eTactica measurement devices are standard Modbus/RTU, you can use them with any standard Modbus infrastructure.

References

The Modbus protocol specification:
http://modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf

RS485 Serial Settings

All eTactica hardware devices have default settings for the RS485 serial interface:
- 19200 baud rate
- 8 data bits
- Even parity
- 1 stop bit

These settings configurable in devices with firmware version 3.2 or higher.

By default, the Modbus Unit ID is the last byte of the serial number, printed on each device. Eg, for a serial number of "00.04.A3.ED.2B.D1" the Unit ID is 0xD1, or 209 decimal. This can be changed via Modbus register 0x2009

Modbus Supported Functions

All eTactica hardware devices support the following Modbus function codes:
- 0x03 – Read Holding Registers
- 0x10 – Write Multiple Registers

Modbus Timing

Typically, the device will respond in 3-4 milliseconds. There is no limit on back to back requests.

Data Format and Addressing

Unless otherwise noted, each register value is an unsigned 16-bit integer. Signed values are regular 2's Complement Signed.

Data Encoding

According to the Modbus protocol specification the Big-Endian representation of both data and addresses is used. This means that the most significant byte (MSB) is sent first.

Addressing

The addresses used in this document are native register address. Not the register number, nor Modicon formatting with 30000/40000. For example, register 0x2000, the Vendor ID, could also be described as register 8192 (decimal) 8193 (decimal, register number) or 48192 (Modicon holding register format)

See these pages for more of information:

- http://www.csimn.com/CSI_pages/Modbus101.html#mb101_reg1
- http://www.simplymodbus.ca/faq.htm#Map

The Register Map below lists the data addresses to use when forming the Modbus request (ADU message format) to each of the eTactica measurement devices.

Examples

Byte and Register ordering

As specified in Section 4.2 of the Modbus Application Protocol Specification, all values are stored in Big Endian, MSB first order. All register addresses in this document are "PDU Addresses" as per Section 4.4. In other words, the first register (Vendor ID) is accessed at register address 0 (plus the offset of 8192 (0x2000)). Note that some Modbus applications refer to this first register as "Modbus Data Model" register 1, which is then at address 0.

Values marked as 32bit, are *also* stored in Big Endian, MSB first, as would be implied by a sensible reading of section 4.2. 64bit values are also stored Big Endian, MSB first.

Example 16bit value

Read a 16-bit value, Line frequency on EM-xxx, data address 0x200F.

PDU	
Function code	0x03
Starting address	0x200F
Quantity of registers	1
PDU message	0x03 – 0x20 – 0x0F – 0x00 – 0x01

Final Value	Value Stored	Register high byte	Register low byte
50.42 Hz	50420 (Register stores mHz)	0xC4	0xF4

Example 32bit value

Read a 32 bit value, Current on Channel 0 on ES-xxx or EB-xxx, data address 0x2016.

Register 0 = 0x2016, Register 1 = 0x2017

PDU	
Function code	0x03
Starting address	0x2016
Quantity of registers	2
PDU message	0x03 – 0x20 – 0x16 – 0x00 – 0x02

Final Value	Value Store	Register 0 high	Register 0 low	Register 1 high	Register 1 low
320.123 Amps	320123 (Value in mA) (0x4E27B)	0x00	0x04	0xe2	0x7b

Modbus Register Map

Common Registers

Below you find the registers, common to all eTactica measurement devices.

Register Address	R/W	Description
0x2000	R	Vendor id (0x524d)
0x2001	R	Product id
0x2002	R	Firmware version
0x2003	R	Serial number bytes 0..1
0x2004	R	Serial number bytes 2..3
0x2005	R	Serial number bytes 4..5
0x2006	R/W	**Command**
0x2007	R	Total register count
0x2008	R/W	**Serial communication settings**
0x2009	R/W	**Modbus slave ID**
0x200A		Reserved
0x200B		Reserved
0x200C		Reserved
0x200D		Reserved
0x200E	R	CPU Temperature in 0.01°C

Modbus Registers – EM Specific

The following section, continuing on from the "common" block above contains some aggregate readings, and has been available on all EM's

The registers containing Power factor values are signed. Active import of energy is given a positive factor, and active export of energy is negative. The number of decimal places depends on the register itself. Distinction of leading/lagging/capacitive/inductive is provided via separate "sign" registers, that are simply –1 or 1.

Register Address	R/W	Size	Description
0x200F	R	1	Frequency (mHz) (Instantenous from last line cycle)
0x2010	R	2	current phase 1 (mA) (1 second average)
0x2012	R	2	current phase 2 (mA) (1 second average)
0x2014	R	2	current phase 3 (mA) (1 second average)
0x2016	R	2	voltage phase 1 (mV) (1 second average)
0x2018	R	2	voltage phase 2 (mV) (1 second average)
0x201A	R	2	voltage phase 3 (mV) (1 second average)
0x201C			Reserved
0x201D			Reserved
0x201E	R	1	Power factor phase 1 (*100)
0x201F	R	1	Power factor phase 2 (*100)
0x2020	R	1	Power factor phase 3 (*100)
0x2021	R	1	**Status**
0x2022	R/W	1	Nominal full-scale amperage of current sensor (eg, 80 when using 80A CTs, or 3000 if using 3000A Flexible coil)
0x2023			Reserved
0x2024			Reserved
0x2025			Reserved
0x2026			Reserved

Register Address	R/W	Size	Description
0x2027			Reserved
0x2028	R/W	4	**cumulative milli Watt hours** – signed 64bit Sum of each phases active import *less* the sum of each phases active export.
0x202C			Reserved
0x202D			Reserved
0x202E			Reserved
0x202F	R/W	4	**cumulative milli var hours -** signed 64bit Sum of each phases reactive import *plus* the sum of each phases reactive export

The following section contains expanded information and is only available on EM2 hardware revisions,with firmware version 4.0 or greater.

Register Address	R/W	Size	Description
0x2100	R	1	CPU Temperature in 0.01°C
0x2101	R	4	Cumulative micro Watt hours – signed 64bit Cumulative is the sum of each phases active import *less* the sum of each phases active export
0x2105	R	4	Cumulative micro var hours – signed 64bit Sum of each phases reactive import *plus* the sum of each phases reactive export
0x2109	R	2	RMS Voltage in mV – phase 1 – 1 second average
0x210B	R	2	RMS Voltage in mV – phase 2 – 1 second average
0x210D	R	2	RMS Voltage in mV – phase 3 – 1 second average
0x210F	R	2	RMS Current in mA – phase 1 – 1 second average
0x2111	R	2	RMS Current in mA – phase 2 – 1 second average
0x2113	R	2	RMS Current in mA – phase 3 – 1 second average
0x2115	R	1	Power factor phase 1 – 1 second average – *10000

0x2116	R	1	Power factor phase 2 – 1 second average – *10000
0x2117	R	1	Power factor phase 3 – 1 second average – *10000
0x2118	R	1	Frequency in milli Hz (last line cycle value)
0x2119	R/W	4	Active In micro Watt Hours – phase 1
0x211D	R/W	4	Active In micro Watt Hours – phase 2
0x2121	R/W	4	Active In micro Watt Hours – phase 3
0x2125	R/W	4	Active Out micro Watt Hours – phase 1
0x2129	R/W	4	Active Out micro Watt Hours – phase 2
0x212D	R/W	4	Active Out micro Watt Hours – phase 3
0x2131	R/W	4	Reactive In micro VAr hours – phase 1
0x2135	R/W	4	Reactive In micro VAr hours – phase 2
0x2139	R/W	4	Reactive In micro VAr hours – phase 3
0x213D	R/W	4	Reactive Out micro VAr hours – phase 1
0x2141	R/W	4	Reactive Out micro VAr hours – phase 2
0x2145	R/W	4	Reactive Out micro VAr hours – phase 3
0x2149	R	2	RMS Voltage in mV – phase 1 – 60 second average
0x214B	R	2	RMS Voltage in mV – phase 2 – 60 second average
0x214D	R	2	RMS Voltage in mV – phase 3 – 60 second average
0x214F	R	2	RMS Current in mA – phase 1 – 60 second average
0x2151	R	2	RMS Current in mA – phase 2 – 60 second average
0x2153	R	2	RMS Current in mA – phase 3 – 60 second average
0x2155	R	1	Power factor phase 1 – 60 second average – *10000
0x2156	R	1	Power factor phase 2 – 60 second average – *10000
0x2157	R	1	Power factor phase 3 – 60 second average – *10000

0x2158	R	2	RMS Voltage in mV – phase 1 – last line cycle
0x215A	R	2	RMS Voltage in mV – phase 2 – last line cycle
0x215C	R	2	RMS Voltage in mV – phase 3 – last line cycle
0x215E	R	2	RMS Current in mA – phase 1 – last line cycle
0x2160	R	2	RMS Current in mA – phase 2 – last line cycle
0x2162	R	2	RMS Current in mA – phase 3 – last line cycle
0x2164	R	1	Power factor phase 1 – latest value – *10000
0x2165	R	1	Power factor phase 2 – latest value – *10000
0x2166	R	1	Power factor phase 3 – latest value – *10000
0x2167	R	1	Sign of power factor phase 1
0x2168	R	1	Sign of power factor phase 2
0x2169	R	1	Sign of power factor phase 3

Detailed Register Descriptions

Register 0x2006 – Command

The command register is a 16 bit value. You use this register to permanently store new configuration settings in EEPROM or reload factory default. The meaning of each bit and bit combination is described in the table below.

Bit #	Description
15..9	Reserved
8	Led blinks in "identification" pattern
7..4	Reserved
3	LED Control State (1 == LED on, 0 == LED off)
2	LED Control State Valid (1 == bit 3 is valid, 0 == bit 3 is ignored)
1	Enable this bit to reload default device configuration to RAM (use in conjunction with bit 0 to reset EEPROM to factory defaults)
0	Enable this bit to store current configuration to EEPROM and restart device

Register 0x2008 - Serial communication settings

The serial communication register is a 16 bit value. It allows you to edit the protocol settings for the RS485 serial interface. Take care modifying these settings. It can be tedious to rediscover what the settings are, for an unknown device.

Default settings for all devices is: 19200 - 8 - Even - 1

Firmware Limitation
The editable feature is only available for devices with firmware version 3.2 or above.

After writing a value to this register, you must write to the **Command** register (0x2006) to store settings in EEPROM and reinitialize the device. This will make the new settings take effect.

Bit #	Description
15..12	Stop Bits (normally 1, 2 is also allowed) 0001 (0x01) : 1 stop bit 0010 (0x02) : 2 stop bits
11..8	Parity (0: None, 1: Odd, 2: Even) 0000 (0x00) : Parity none 0001 (0x01) : Parity odd 0010 (0x02) : Parity even
7..0	Baud rate value (See table below)

Baud Rate Table

Values to write as the lowest byte in this register that represent pre-defined baud rates.

Lowest Byte of 0x2008	**Baud Rate**
0000 (0x00)	default (19200 at present)
0001 (0x01)	600
0010 (0x02)	1200
0011 (0x03)	2400
0100 (0x04)	4800
0101 (0x05)	9600
0110 (0x06)	19200
0111 (0x07)	38400
1000 (0x08)	57600
1001 (0x09)	115200

Examples

Contents of register 0x2008	**Description**
0x1200	Factory Default, 1 Stop bit, Even Parity, Default Baud Rate (19200)
0x1005	1 Stop Bit, No Parity, 9600 Baud
0x1209	1 Stop Bit, Even Parity, 115200 Baud
0x1101	1 Stop bit, Odd Parity, 600 Baud
0x0044	Don't do this! (Unexpected values will be converted to 1 Stop bit, No Parity, 115200)

Register 0x2009 - Modbus slave ID

The Modbus slave ID register, is a 16 bit value. It is a configurable register where you can modify the default slave ID for your device. Only the lower byte for this 16 bit value is valid for the slave ID. Take care to preserve the upper byte as is.

According to the Modbus protocol, it is only allowed to use addresses from 1 - 247.

Bit #	Description
15..8	Reserved, do not modify contents
7..0	Modbus slave ID (values from 1 to 247)

Register 0x2028 - Cumulative milli Watt hours

The cumulative milli Watt hours is a 64 bit signed integer value and is stored in 4 registers from 0x2028 to 0x202B. These registers are writable, but you should take care doing so. Common uses are for setting an initial value when installing at a new location. If you do write to these registers, make sure to use the **Command** register (0x2006) to save those values to EEPROM. Cumulative Watt Hours are defined as import *less* export, which matches how a utility meter operates, giving a representation of the actual amount of delivered active energy.

Register 0x202F - Cumulative milli VAr hours

The cumulative milli VAr hours is a 64 bit signed integer value and is stored in 4 registers from 0x202F to 0x2032. Unlike Watt Hours, this is the sum of import (lagging) *plus* export (leading) providing a representation of the total amount of reactive energy.

Active/Reactive IN/OUT micro watt hour / micro VAr hour counters

Just like the cumulative registers, these are 64bit up counters, stored per phase. They are writable, but as they are purely up counters in normal use, you should treat these as 63bit unsigned values, and not try and write negative numbers to these registers.

Register 0x2021 - Status

In register 0x2021 you find the status bitmap for the EM device family. The bitmap indicates possible error states, as explained below.
When the following bit position of the bitmap is set, or '1', this indicates an error accordingly. For no errors, the readings are all zeroes.
Any of these bits being set results in the status LED blinking faster than normal (every 0,2 sec instead of every 0,5 sec).

Bit #	Bit map	Description
	0x0000	No error
0	0x0001	SPI Connection to baseboard not working
1	0x0002	Voltage sag on phase 1
2	0x0004	Voltage sag on phase 2
3	0x0008	Voltage sag on phase 3
4	0x0010	Power Factor phase 1 < 0.4
5	0x0020	Power Factor phase 2 < 0.4
6	0x0040	Power Factor phase 3 < 0.4
7	0x0080	Zero crossing timeout on phase 1 (phase is missing)
8	0x0100	Zero crossing timeout on phase 2 (phase is missing)
9	0x0200	Zero crossing timeout on phase 3 (phase is missing)