SIEMENS

9200 Serial Communications Protocol and ION / Modbus Register Map

This document explains the Modbus protocol on the 9200 meter.

The 9200 meter performs Modbus RTU communications according to the Modicon Standard. Refer to www.modicon.com for Modbus/TCP standard and Modicon Modbus Serial Communications Protocol documentation.

This document describes the Modbus communications protocol employed by the meter and how to pass information into and out of the meter in a Modbus network. It is assumed that the reader is familiar with the Modbus protocol and serial communications in general.

In This Document

Purpose of the Communications Protocol

The 9200 meter Modbus communications protocol allows measured data and setup information to be efficiently transferred between a Modbus master station and a meter. This includes:

- Interrogation of all data measured by the meter
- Configuration and interrogation of the meter
- Interrogation and control of the meter's digital outputs
- Clearing of accumulated demand and energy parameters

Modbus Implementation on the Meter

Ground Rules

The meter is capable of communicating via the RS-485 serial communication standard. The RS-485 medium allows for multiple devices on a multi-drop network.

The points below follow the Modicon standard:

- All communications on the communications loop conforms to a master/slave scheme. In this scheme, information and data is transferred between a Modbus master device and up to 32 slave monitoring devices (more, if repeaters are used).
- The master initiates and controls all information transfer on the communications loop.
- A slave device never initiates a communications sequence.
- All communications activity on the loop occurs in the form of "packets." A packet is a serial string of 8-bit bytes. The maximum number of bytes contained within one packet is 255.
- All packets transmitted by the master are requests. All packets transmitted by a slave device are responses.
- At most one slave can respond to a single request from a master.

Modes of Transmission

The Modbus protocol uses ASCII and RTU modes of transmission. The meter supports only the RTU mode of transmission, with 8 data bits, no parity, and one stop bit.

Description of the Modbus Packet Structure

Every Modbus packet consists of four fields:

- Slave Address Field
- Function Field
- Data Field
- Error Check Field (Checksum)

Slave Address Field

The slave address field of a Modbus packet is one byte in length and uniquely identifies the slave device involved in the transaction. Valid addresses range between 1 and 247. A slave device performs the command specified in the packet when it receives a request packet with the slave address field matching its own address. A response packet generated by the slave has the same value in the slave address field.

Function Field

The function field of a Modbus request packet is one byte in length and tells the addressed slave which function to perform. Similarly, the function field of a response packet tells the master what function the addressed slave has just performed. "Table 1: Modbus Functions Supported by the Meters" lists the Modbus functions supported by the meter.

Table 1: Modbus Functions Supported by the Meters

Function	Meaning	Action
03	Read Holding registers	Obtains the current value in one or more holding registers of the meter.
16	Preset Multiple registers	Places specific values into a series of consecutive holding registers of the meter. The holding registers that can be written to the meter are shown in the register map.

Data Field

The data field of a Modbus request is of variable length, and depends upon the function. This field contains information required by the slave device to perform the command specified in a request packet or data being passed back by the slave device in a response packet.

Data in this field are contained in 16-bit registers. Registers are transmitted in the order of high-order byte first, low-order byte second. This ordering of bytes is called "Big Endian" format.

Example:

A 16-bit register contains the value 12AB Hex. This register is transmitted:

- High order byte = 12 Hex
- Low order byte = AB Hex

This register is transmitted in the order 12 AB.

Error Check Field (Checksum)

The checksum field lets the receiving device determine if a packet is corrupted with transmission errors. In Modbus RTU mode, a 16-bit Cyclic Redundancy Check (CRC-16) is used.

The sending device calculates a 16-bit value, based on every byte in the packet, using the CRC-16 algorithm. The calculated value is inserted in the error check field.

The receiving device performs the calculation, without the error check field, on the entire packet it receives. The resulting value is compared to the error check field. Transmission errors occur when the calculated checksum is not equal to the checksum stored in the incoming packet. The receiving device ignores a bad packet.

The CRC-16 algorithm is detailed in appendix A of this document.

Exception Responses

If a Modbus master device sends an invalid command to a meter or attempts to read an invalid holding register, an exception response is generated. The exception response follows the standard packet format. The high order bit of the function code in an exception response is set to 1.

The data field of an exception response contains the exception error code. "Table 2: Exception Codes Supported by the Meter" on page 5 describes the exception codes supported by the meter and the possible causes.

Table 2: Exception Codes Supported by the Meter

Code	Name	Meaning
01	Illegal Function	An invalid command is contained in the function field of the request packet. The meter only supports Modbus functions 3 and 16.
02	Illegal Address	The address referenced in the data field is an invalid address for the specified function. This could also indicate that the registers requested are not within the valid register range of the meter.
03	Illegal Value	The value referenced in the data field is not allowed for the referenced register on the meter.

Broadcast Packets

The 9200 Modbus protocol supports broadcast request packets. The purpose of a broadcast request packet is to allow all slave devices to receive the same command from the master station.

A broadcast request packet is the same as a normal request packet, except the slave address field is set to zero (0). All Modbus slave devices receive and execute a broadcast request command, but no device will respond.

The Preset Multiple registers command (function 16) is the only command supporting broadcast packets.

Packet Communications

This section illustrates the Modbus functions supported by the meter.

Function 03: Read Holding Registers

To read meter parameter values, a master station must send the slave device a Read Holding registers request packet.

The Read Holding registers request packet specifies a start register and a number of registers to read. The start register is numbered from zero (40001 = zero, 40002 = one, etc.).

The meter responds with a packet containing the values of the registers in the range defined in the request.

Read Holding Registers

Read Registers Request Packet (master station to meter)

Unit ID/Slave Address (1 byte)

03 (Function code) (1byte)

Start Register (sr) (2 bytes)

# of Registers to Read (nr) (2 bytes)

CRC Checksum

Read Registers Response Packet (meter to master station)

Unit ID/Slave Address (1 byte)

03 (Function code) (1 byte)

Byte Count (2 x nr) (1 byte)

First Register in range (2 bytes)

Second Register in range (2 bytes)

...

CRC Checksum (2 bytes)

Example:

A meter in 4-wire WYE volts mode is configured as a Modbus slave device with slave address 100. The master station requests to read realtime volts on all three phases (A, B, C). These three parameters are available in Modbus registers 40100, 40101 and 40102, with a user programmable scaling factor (default 10). In accordance with the Modbus protocol, register 40100 is numbered as 99 when transmitted. The request must read 3 registers starting at 99.

Slave address: 100 = 64 (hex) Start register 99= 0063 (hex)

Request Packet

Slave	Function	Start Register		# of Registers (3)		CRC Checksum	
64*	03	00	63	00	03	FC	20

Response Packet

Slave	Function	Byte Count	Register 1		Register 2		Register 3		CRC Checksum	
64	03	06	2E	CE	2E	E8	2F	13	58	0D

The master station retrieves the data from the response:

Register 40100: 2ECE (hex) = 11982 (scaled: 1198.2)

Register 40101: 2EE8 (hex) = 12008 (scaled: 1200.8)

Register 40102: 2F13 (hex) = 12051 (scaled: 1205.1)

* The values shown in illustrated packets are represented in hexadecimal format.

Function 16: Preset Multiple Registers

The Preset Multiple registers command packet allows a Modbus master to configure or control the meter.

A Preset Multiple registers data-field request packet contains a definition of a range of registers to write to, and the values that write to those registers.

The meter responds with a packet indicating that a write was performed to the range of registers specified in the request.

The table below, shows the Preset Multiple registers request and response packet formats, and an example transaction.

Preset Multiple Registers

Preset Registers Request Packet (master station to meter)

Unit ID/Slave Address (1 byte)

16 (Function code) (1byte)

Start Register (sr) (2 bytes)

# of Registers to Write (nr) (2 bytes)

Byte Count (2 x nr) (1 byte)

First Register in range (2 bytes)

Second Register in range (2 bytes)

...

CRC Checksum (2 bytes)

Preset Registers Response Packet (meter to master station)

Unit ID/Slave Address (1 byte)

16 (Function code) (1 byte)

Start Register (sr) (2 bytes)

# of Registers Written (nr) (2 bytes)

CRC Checksum (2 bytes)

Example:

A meter is configured as a Modbus slave device with slave address 200. The master station requests to set the PT ratio to 1200:120. From the register map, the Power Meter PT Primary and Secondary setup registers are Modbus registers 44002 and 44003. Register 44002 is numbered 4001 when transmitted. The request must write two registers starting at 4001.

Slave address: 200 = C8 (hex)

Start register 4001 = 0FA1 (hex)

Value 1: 1200 = 04B0 (hex)

Value 2: 120 = 0078 (hex)

Request Packet

Slave	Function	Start Register		# of Registers (4)		Byte Count	Register 1		Register 2		CRC Checksum	
C8*	10	0F	A1	00	02	04	04	B0	00	78	EE	3F

Response Packet

Slave	Function	Start Register		# of Registers (4)		CRC Checksum	
C8	10	0F	A1	00	02	01	E0

* The values shown in illustrated packets are represented in hexadecimal format.

Invalid Registers

In the meter Modbus register map, there are reserved sections. For example, registers before 40500 are reserved.

When a reserved register is read, the value returned is FFFF (hex). When a reserved register is written, the value supplied is not stored. The meter does not reject the request.

Meter Modbus Registers

The 9200 meter supports Modbus Holding registers (address range 4xxxx). There are three types of parameters:

- Metered Parameters
- External Control/Reset Parameters
- Setup Parameters
  - Enumerated Setup Parameters
  - Numeric Bounded Parameters

Metered Parameters

All the values measured by the meter are available through the Modbus protocol. The majority of these parameters have fixed scaling. However, the scaling of voltage, current and power values is configurable via a numeric setup register.

There are three blocks. Factory registers such as serial number, firmware revision, etc. are located between addresses 40001 and 40099. Measured quantities such as voltage, current, power and energy are located between addresses 40100 and 40499. The register addresses for the first Feature Pack start at 40500; the register addresses for the second Feature Pack start at 40600. Consult the Feature Pack documentation for the contents of these registers.

The 9200 supports 4 data formats:

- Unsigned 16-bit Integer Format
- Signed 16-bit Integer Format
- Unsigned 32-bit Integer Format
- Signed 32-bit Integer Format

16-bit Integer Format

Unsigned and Signed 16-bit Integer Formats are the simplest formats. If the format is unsigned the value range for the output registers is 0 to 65535. If the format is signed, the value range is -32767 to +32767 (two's-complement).

32-bit Integer Format

To accommodate values that can reach beyond the 16-bit limitation, the 9200 provides 32-bit integer format. In Signed and Unsigned 32-bit Integer Formats, the 32-bit value is split into two consecutive 16-bit registers. The first register is the low-order word and the second register is the high-order word.

To interpret the value, take the second register (high-order word) and multiply by 65536. Then add the first register (low-order word). The formula is:

value = (second register x 65536) + first register

In Unsigned 32-bit Integer Format, both the high-order and low-order registers are unsigned 16-bit integers.

Example:

Value 12345678 is passed in *unsigned* 32-bit integer format:

12345678 = 00BC614E Hex

First Register = 614E Hex (unsigned) = 24910

Second Register = 00BC Hex (unsigned) = 188

value = (188 x 65536) + 24910 = 12345678

In Signed 32-bit Integer Format, the high-order register is a signed 16-bit number, but the low-order register is unsigned.

Example:

Value -12345678 is passed in *signed* 32-bit integer format:

-12345678 = FF439EB2 Hex

First Register = 9EB2 Hex (unsigned) = 40626

Second Register = FF43 Hex (signed) = -189

value = (-189 x 65536) + 40626 = -12345678

Control Parameters

There are two types of control parameters in the meter which can be accessed via Modbus. This section describes how the parameters appear to the Modbus protocol. The two control parameters types are:

- Digital Output Control registers
- Accumulation Reset registers

Digital Output Control

Registers 42004 and 42005 are available to remotely control the meter's digital outputs. A non-zero value written to these registers places the corresponding digital output in an asserted state. Conversely, a logic zero written to one of these registers de-asserts the output.

To use the Read-Write Control Map, it is recommended that your 9200 meter has firmware version 202 or later. For detailed information contact Technical Services.

Reset Accumulation

Registers 42001 to 42003 are available to remotely reset energy accumulation and maximum demand values. Writing any value to one of these registers causes the corresponding parameter to reset. If read, these registers will return an error.

Setup Parameters

Meters can be configured remotely via Modbus communications. Registers 4400 to 44029 offer enumerated or numeric parameters.

Enumerated Setup

Enumerated registers are used where a list of options are available. For example, the Volts Mode register has five options: 4W-WYE, DELTA, SINGLE, DEMO, 3W-WYE and DELTA DIRECT. These options are represented by a numeric relationship; for example, the following relationship is defined for the Volts Mode register:

0 = 4W-WYE

1 = DELTA

2 = SINGLE

3 = DEMO

4 = 3W-WYE

5 = DELTA DIRECT

For example, to set the meter to 3W-WYE mode, you write a 4 into the *Volts Mode* setup register (44001).

See the register map for details.

Numeric Setup

The numeric setup parameters include: PT/CT ratios, demand intervals, digital output pulse values, unit ID, password, and RTS delay.

All 9200 numeric parameters are represented in Unsigned 16-bit Integer Format. See the register map for details. Note that all parameters have bounds. For example, unit IDs must be in the range 1 to 247; any attempt to write a value outside this range will fail.

See the introduction to Appendix B for a discussion on scaling.

Appendix A: CRC-16 Calculation

This appendix describes the procedure for obtaining the CRC-16 error check field for a Modbus RTU frame.

Procedure

A frame can be considered as a continuous, serial stream of binary data (ones and zeros). The 16-bit checksum is obtained by multiplying the serial data stream by 2^{16} (10000000000000000) and then dividing it by the ***generator polynomial*** $x^{16}+x^{15}+x^{2}+1$, which can be expressed as the 16-bit binary number 11000000000000101. The quotient is ignored and the 16-bit remainder is the checksum, which is appended to the end of the frame.

In calculating the CRC, all arithmetic operations (additions and subtractions) are performed using MODULO TWO, or EXCLUSIVE OR operation. A step-by-step example shows how to obtain the checksum for a simple Modbus RTU frame.

Steps for generating the CRC-16 checksum:

1. Drop the MSB (Most Significant Bit) of the generator polynomial and reversing the bit sequence to form a new polynomial. This yields the binary number 1010 0000 0000 0001, or A0 01 (hex).
2. Load a 16-bit register with initial value FF FF (hex).
3. Exclusive OR the first data byte with the low-order byte of the 16-bit register. Store the result in the 16-bit register.
4. Shift the 16-bit register one bit to the right.
5. If the bit shifted out to the right is one, Exclusive OR the 16-bit register with the new generator polynomial, store the result in the 16-bit registers. Return to step 4.
6. If the bit shifted out to the right is zero, return to step 4.
7. Repeat steps 4 and 5 until 8 shifts have been performed.
8. Exclusive OR the next data byte with the 16-bit register.
9. Repeat steps 4 through 7 until all bytes of the frame are Exclusive Ored with the 16-bit register and shifted 8 times.
10. The content of the 16-bit register is the checksum and is appended to the end of the frame.

Example:

A Modbus master node requests to read register 40011 from a Modbus slave with address 100 (64 hex). As per the Modbus protocol, reading register 40011 means using the READ HOLDING REGISTERS function (03 hex) with start register 10.

Initial frame:

Slave Address	Function	Start Register		# of Registers		Error Check (CRC-16)
64	03	00	0A	00	01	To be calculated

Step	Byte	Bits Shifted	Action	16-Bit Register	Bit Shifted Out
2	1		Initial Value Load First Data Byte	1111 1111 1111 1111 0000 0000 0110 0100	
3			Exclusive OR	1111 1111 1001 1011	
4		1	Shift 1 bit to the Right Generator Polynomial	0111 1111 1100 1101 1010 0000 0000 0001	1
5a			Exclusive OR	1101 1111 1100 1100	
4		2	Shift 1 bit to the Right	0110 1111 1110 0110	0
4		3	Shift 1 bit to the Right	0011 0111 1111 0011	0
4		4	Shift 1 bit to the Right Generator Polynomial	0001 1011 1111 1001 1010 0000 0000 0001	1
5a			Exclusive OR	1011 1011 1111 1000	
4		5	Shift 1 bit to the Right	0101 1101 1111 1100	0
4		6	Shift 1 bit to the Right	0010 1110 1111 1110	0
4		7	Shift 1 bit to the Right	0001 0111 0111 1111	0
4		8	Shift 1 bit to the Right Generator Polynomial	0000 1011 1011 1111 1010 0000 0000 0001	1
5a			Exclusive OR	1010 1011 1011 1110	
7	2		Load 2nd Data Byte Exclusive OR	0000 0000 0000 0011 1010 1011 1011 1101	
4		1	Shift 1 bit to the Right Generator Polynomial	0101 0101 1101 1110 1010 0000 0000 0001	1
5a			Exclusive OR	1111 0101 1101 1111	
4		2	Shift 1 bit to the Right Generator Polynomial	0111 1010 1110 1111 1010 0000 0000 0001	1
5a			Exclusive OR	1101 1010 1110 1110	
4		3	Shift 1 bit to the Right	0110 1101 0111 0111	0
4		4	Shift 1 bit to the Right Generator Polynomial	0011 0110 1011 1011 1010 0000 0000 0001	1
5a			Exclusive OR	1001 0110 1011 1010	

Step	Byte	Bits Shifted	Action	16-Bit Register	Bit Shifted Out
4		5	Shift 1 bit to the Right	0100 1011 0101 1101	0
4		6	Shift 1 bit to the Right Generator Polynomial	0010 0101 1010 1110 1010 0000 0000 0001	1
5a			Exclusive OR	1000 0101 1010 1111	
4		7	Shift 1 bit to the Right Generator Polynomial	0100 0010 1101 0111 1010 0000 0000 0001	1
5a			Exclusive OR	1110 0010 1101 0110	
4		8	Shift 1 bit to the Right	0111 0001 0110 1011	0
7	3		Load 3rd Data Byte Exclusive OR	0000 0000 0000 0000 0111 0001 0110 1011	
4		1	Shift 1 bit to the Right Generator Polynomial	0011 1000 1011 0101 1010 0000 0000 0001	1
5a			Exclusive OR	1001 1000 1011 0100	
4		2	Shift 1 bit to the Right	0100 1100 0101 1010	0
4		3	Shift 1 bit to the Right	0010 0110 0010 1101	0
4		4	Shift 1 bit to the Right Generator Polynomial	0001 0011 0001 0110 1010 0000 0000 0001	1
5a			Exclusive OR	1011 0011 0001 0111	
4		5	Shift 1 bit to the Right Generator Polynomial	0101 1001 1000 1011 1010 0000 0000 0001	1
5a			Exclusive OR	1111 1001 1000 1010	
4		6	Shift 1 bit to the Right	0111 1100 1100 0101	0
4		7	Shift 1 bit to the Right Generator Polynomial	0011 1110 0110 0010 1010 0000 0000 0001	1
5a			Exclusive OR	1001 1110 0110 0011	
4		8	Shift 1 bit to the Right Generator Polynomial	0100 1111 0011 0001 1010 0000 0000 0001	1
5a			Exclusive OR	1110 1111 0011 0000	
7	4		Load 4th Data Byte Exclusive OR	0000 0000 0000 1010 1110 1111 0011 1010	
4		1	Shift 1 bit to the Right	0111 0111 1001 1101	0
4		2	Shift 1 bit to the Right Generator Polynomial	0011 1011 1100 1110 1010 0000 0000 0001	1
5a			Exclusive OR	1001 1011 1100 1111	
4		3	Shift 1 bit to the Right Generator Polynomial	0100 1101 1110 0111 1010 0000 0000 0001	1
5a			Exclusive OR	1110 1101 1110 0110	
4		4	Shift 1 bit to the Right	0111 0110 1111 0011	0

Step	Byte	Bits Shifted	Action	16-Bit Register	Bit Shifted Out
4		5	Shift 1 bit to the Right Generator Polynomial	0011 1011 0111 1001 1010 0000 0000 0001	1
5a			Exclusive OR	1001 1011 0111 1000	
4		6	Shift 1 bit to the Right	0100 1101 1011 1100	0
4		7	Shift 1 bit to the Right	0010 0110 1101 1110	0
4		8	Shift 1 bit to the Right	0001 0011 0110 1111	0
7	5		Load 5th Data Byte Exclusive OR	0000 0000 0000 0000 0001 0011 0110 1111	
4		1	Shift 1 bit to the Right Generator Polynomial	0000 1001 1011 0111 1010 0000 0000 0001	1
5a			Exclusive OR	1010 1001 1011 0110	
4		2	Shift 1 bit to the Right	0101 0100 1101 1011	0
4		3	Shift 1 bit to the Right Generator Polynomial	0010 1010 0110 1101 1010 0000 0000 0001	1
5a			Exclusive OR	1000 1010 0110 1100	
4		4	Shift 1 bit to the Right	0100 0101 0011 0110	0
4		5	Shift 1 bit to the Right	0010 0010 1001 1011	0
4		6	Shift 1 bit to the Right Generator Polynomial	0001 0001 0100 1101 1010 0000 0000 0001	1
5a			Exclusive OR	1011 0001 0100 1100	
4		7	Shift 1 bit to the Right	0101 1000 1010 0110	0
4		8	Shift 1 bit to the Right	0010 1100 0101 0011	0
7	6		Load 6th Data Byte Exclusive OR	0000 0000 0000 0001 0010 1100 0101 0010	
4		1	Shift 1 bit to the Right	0001 0110 0010 1001	0
4		2	Shift 1 bit to the Right Generator Polynomial	0000 1011 0001 0100 1010 0000 0000 0001	1
5a			Exclusive OR	1010 1011 0001 0101	
4		3	Shift 1 bit to the Right Generator Polynomial	0101 0101 1000 1010 1010 0000 0000 0001	1
5a			Exclusive OR	1111 0101 1000 1011	
4		4	Shift 1 bit to the Right Generator Polynomial	0111 1010 1100 0101 1010 0000 0000 0001	1
5a			Exclusive OR	1101 1010 1100 0100	
4		5	Shift 1 bit to the Right	0110 1101 0110 0010	0
4		6	Shift 1 bit to the Right	0011 0110 1011 0001	0

Step	Byte	Bits Shifted	Action	16-Bit Register	Bit Shifted Out
4		7	Shift 1 bit to the Right Generator Polynomial	0001 1011 0101 1000 1010 0000 0000 0001	1
5a			Exclusive OR	1011 1011 0101 1001	
4		8	Shift 1 bit to the Right Generator Polynomial	0101 1101 1010 1100 1010 0000 0000 0001	1
5a			Exclusive OR	1111 1101 1011 1100	
			RESULT	Hex FD Hex AD	

The frame completed with the CRC-16 checksum is as follows:

Slave Address	Function	Start Register		# of Registers		Error Check (CRC-16)	
64	03	00	0A	00	01	AD	FD

Pseudocode For CRC-16 Generation

For users familiar with computer programming, the following is the pseudocode for calculating the 16-bit Cyclic Redundancy Check.

```
Initialize a 16-bit register to FFFF Hex
      Initialize the generator polynomial to A001 Hex

FOR n=1 to # of bytes in packet
      BEGIN
        XOR nth data byte with the 16-bit register
        FOR bits_shifted = 1 to 8
        BEGIN
          SHIFT 1 bit to the right
          IF (bit shifted out EQUAL 1)
            XOR generator polynomial with the 16-bit register
      and store result in the 16-bit register
        END
      END
```

The resultant 16-bit register contains the CRC-16 checksum.

Appendix B: ION / Modbus Map

This appendix contains the ION/Modbus register map for the 9200 meter.

In the following Modbus map many numeric measurements are scaled. Scaling is either fixed or programmable.

For an example of fixed scaling, suppose that the frequency register 40115 contains the value 5987. Since this register is scaled by “x100” the actual measured value is 59.87 Hz.

For an example of programmable scaling, suppose that the meter Vln a value is 480.1. If the PVS scaling is set to “x10” (default), then register 40100 will contain a value of 4801. If the PVS scaling is set to “0.1,” then register 40100 will contain a value of 48. The meter supports programmable scaling for voltage (PVS), current (PCS), neutral current (PnS), and power (PPS).

Modbus Addr	Measurement (9200 Megawatt meters)	Measurement (all other 9200 meters)	Format	Scale	Default Scale	Description
40001	serial number	serial number	UINT32	x1		See note 1.
40003	firmware revision	firmware revision	UINT16	x1		
40004	oem identification	oem identification	UINT16	x1		See note 2.
40005	meter options	meter options	UINT32	x1		See note 3.
40007	# meter power ups	# meter power ups	UINT16	x1		
40008	# peak demand resets	# peak demand resets	UINT16	x1		
40009	meter on-time	meter on-time	UINT32	x1		See note 4.
40011	# flash erase cycles	# flash erase cycles	UINT32	x1		
40013	device type	device type	UINT16	x1		See note 5.
40014	Reserved	Reserved				
40015	demand interval down counter	demand interval down counter	UINT16	x1		
40016 - 40099	Reserved	Reserved				
40100	kVln a	Vln a	UINT16	PVS	x10	See notes 6 & 7.
40101	kVln b	Vln b	UINT16	PVS	x10	See note 6.
40102	kVln c	Vln c	UINT16	PVS	x10	See note 6.
40103	kVln avg	Vln avg	UINT16	PVS	x10	See note 6.
40104	kVll ab	Vll ab	UINT16	PVS	x10	
40105	kVll bc	Vll bc	UINT16	PVS	x10	
40106	kVll ca	Vll ca	UINT16	PVS	x10	
40107	kVll avg	Vll avg	UINT16	PVS	x10	
40108	I a	I a	UINT16	PCS	x10	See note 8.

Modbus Addr	Measurement (9200 Megawatt meters)	Measurement (all other 9200 meters)	Format	Scale	Default Scale	Description
40109	I b	I b	UINT16	PCS	x10	
40110	I c	I c	UINT16	PCS	x10	
40111	I avg	I avg	UINT16	PCS	x10	
40112	I demand	I demand	UINT16	PCS	x10	
40113	I peak demand	I peak demand	UINT16	PCS	x10	
40114	I4	I4	UINT16	PnS	x10	See note 9.
40115	Frequency	Frequency	INT16	x100		
40116	PF sign total	PF sign total	INT16	x100		
40117	PF sign a	PF sign a	INT16	x100		
40118	PF sign b	PF sign b	INT16	x100		
40119	PF sign c	PF sign c	INT16	x100		
40120	MW total	kW total	INT16	PPS	x1	See note 10.
40121	MVAR total	kVAR total	INT16	PPS	x1	
40122	MVA total	kVA total	INT16	PPS	x1	
40123	MW a	kW a	INT16	PPS	x1	
40124	MW b	kW b	INT16	PPS	x1	
40125	MW c	kW c	INT16	PPS	x1	
40126	MVAR a	kVAR a	INT16	PPS	x1	
40127	MVAR b	kVAR b	INT16	PPS	x1	
40128	MVAR c	kVAR c	INT16	PPS	x1	
40129	MVA a	kVA a	INT16	PPS	x1	
40130	MVA b	kVA b	INT16	PPS	x1	
40131	MVA c	kVA c	INT16	PPS	x1	
40132	MW demand	kW demand	INT16	PPS	x1	
40133	MW peak demand	kW peak demand	INT16	PPS	x1	
40134	MVAR demand	kVAR demand	INT16	PPS	x1	
40135	MVA demand	kVA demand	INT16	PPS	x1	
40136	MVAR peak demand	kVAR peak demand	INT16	PPS	x1	
40137	MVA peak demand	kVA peak demand	INT16	PPS	x1	
40138	MWh del	kWh del	UINT32	x1		See note 11.
40140	MWh rec	kWh rec	UINT32	x1		
40142	MVARh del	kVARh del	UINT32	x1		
40144	MVARh rec	kVARh rec	UINT32	x1		
40146	MVAh del+rec	kVAh del+rec	UINT32	x1		

Modbus Addr	Measurement (9200 Megawatt meters)	Measurement (all other 9200 meters)	Format	Scale	Default Scale	Description
40148	V1 THD	V1 THD	UINT16	x10		
40149	V2 THD	V2 THD	UINT16	x10		
40150	V3 THD	V3 THD	UINT16	x10		
40151	I1 THD	I1 THD	UINT16	x10		
40152	I2 THD	I2 THD	UINT16	x10		
40153	I3 THD	I3 THD	UINT16	x10		
40154	I a demand	I a demand	UINT16	PCS	x10	
40155	I b demand	I b demand	UINT16	PCS	x10	
40156	I c demand	I c demand	UINT16	PCS	x10	
40157	I a peak demand	I a peak demand	UINT16	PCS	x10	
40158	I b peak demand	I b peak demand	UINT16	PCS	x10	
40159	I c peak demand	I c peak demand	UINT16	PCS	x10	
40160	MWh a del	kWh a del	UINT32	x1		See note 11.
40162	MWh b del	kWh b del	UINT32	x1		
40164	MWh c del	kWh c del	UINT32	x1		
40166	MWh a rec	kWh a rec	UINT32	x1		
40168	MWh b rec	kWh b rec	UINT32	x1		
40170	MWh c rec	kWh c rec	UINT32	x1		
40172	MVARh a del	kVARh a del	UINT32	x1		
40174	MVARh b del	kVARh b del	UINT32	x1		
40176	MVARh c del	kVARh c del	UINT32	x1		
40178	MVARh a rec	kVARh a rec	UINT32	x1		
40180	MVARh b rec	kVARh b rec	UINT32	x1		
40182	MVARh c rec	kVARh c rec	UINT32	x1		
40184	MVAh a	kVAh a	UINT32	x1		
40186	MVAh b	kVAh b	UINT32	x1		
40188	MVAh c	kVAh c	UINT32	x1		
40189 - 40499	Reserved	Reserved				
40500	Expansion, SnapOn 1, 25 regs	Expansion, SnapOn 1, 25 regs	UINT16	x1		
40700	Expansion, SnapOn 2, 25 regs	Expansion, SnapOn 2, 25 regs	UINT16	x1		

Modbus Addr	Measurement (9200 Megawatt meters)	Measurement (all other 9200 meters)	Format	Scale	Default Scale	Description
41138	MWh del	kWh del	UINT32	x1		See note 12.
41140	MWh rec	kWh rec	UINT32	x1		
41142	MVARh del	kVARh del	UINT32	x1		
41144	MVARh rec	kVARh rec	UINT32	x1		
41146	MVAh del+rec	kVAh del+rec	UINT32	x1		
41160	MWh a del	kWh a del	UINT32	x1		
41162	MWh b del	kWh b del	UINT32	x1		
41164	MWh c del	kWh c del	UINT32	x1		
41166	MWh a rec	kWh a rec	UINT32	x1		
41168	MWh b rec	kWh b rec	UINT32	x1		
41170	MWh c rec	kWh c rec	UINT32	x1		
41172	MVARh a del	kVARh a del	UINT32	x1		
41174	MVARh b del	kVARh b del	UINT32	x1		
41176	MVARh c del	kVARh c del	UINT32	x1		
41178	MVARh a rec	kVARh a rec	UINT32	x1		
41180	MVARh b rec	kVARh b rec	UINT32	x1		
41182	MVARh c rec	kVARh c rec	UINT32	x1		
41184	MVAh a	kVAh a	UINT32	x1		
41186	MVAh b	kVAh b	UINT32	x1		
41188	MVAh c	kVAh c	UINT32	x1		

Read-Write Configuration Map

Modbus Addr	Configuration Parameter (all 9200 meters)	Format	Scale	Default	Description
44000	Configuration via display password	UINT16	x1	0	0 – 9999
44001	Volts Mode	Enumerated		5 = Delta direct	0 = 4W (4-Wire WYE) 1 = dELt (Delta) 2 = 2W (Single Phase) 3 = dEM (Demonstration) 4 = 3W (3-Wire WYE) 5 = dELd (Delta direct)
44002	PT Primary	UINT16	x1	480	1 – 65535
44003	PT Secondary	UINT16	x1	480	
44004	CT Primary	UINT16	x1	400	
44005	CT Secondary	UINT16	x1	5	
44006	V1 Polarity	Enumerated		0 = Normal	0 = nor (Normal) 1 = inv (Inverted)
44007	V2 Polarity				
44008	V3 Polarity				
44009	I1 Polarity				
44010	I2 Polarity				
44011	I3 Polarity				
44012	Programmable Voltage Scale (PVS)	Enumerated		4 = 10	0 = 0.001 1 = 0.01 2 = 0.1 3 = 1 4 = 10 5 = 100 6 = 1000
44013	Programmable Current Scale (PIS)				
44014	Programmable Neutral Current Scale (PnS)				
44015	Programmable Power Scale (PPS)			3 = 1	
44016	Demand Sub Interval	UINT16	x1	15	1 – 60 minutes
44017	Demand #Sub Intervals	UINT16	x1	1	1 – 5
44018	Kt, Digital Output #1	UINT16	x10	1.0	0.1 - 999.9
44019	Kt, Digital Output #2	UINT16	x10		
44021	Output Mode, Digital Output #1	Enumerated		0 = kWh	0 = kWh Del. 1 = kVAh 2 = kVARh Del. 3 = Ext1 4 = Ext2 5 = kWh Rec. 6 = kVARh Rec. See note 13.
44022	Output Mode, Digital Output #2			2 = kVARh	
44024	Baud Rate	Enumerated		3 = 9600bps	0 = 1200bps 1 = 2400bps 2 = 4800bps 3 = 9600bps 4 = 19200bps
44025	Protocol	Enumerated		1 = MODBUS	0 = PML 1 = MODBUS See note 14.

Modbus Addr	Configuration Parameter (all 9200 meters)	Format	Scale	Default	Description
44026	Unit ID	UINT16	x1	100 plus the last 2 digits of the Serial Number	1 – 247
44027	RTS Delay	UINT16	x1	20	0 – 1000 milliseconds See note 15.
44028	Display Scroll Time	UINT16	x1	0	0 – 30 seconds See note 16.
44029	Display Refresh Period	UINT16	x1	2	1 – 6 seconds See note 17.
44030	PT Scale	Enumerated		0 = x1	0 = x1 1 = x1000

Read-Write Control Map

Modbus Addr[18]	Control Parameter (all 9200 meters)	Format	Scale	Description
42001	Energy Reset	UINT16	x1	Reset MWh, MVAh, and MVARh to 0 (Megawatt meters) Reset kWh, kVAh, and kVARh to 0 (all other models)
42002	Peak Power Demand Reset	UINT16	x1	Reset MW, MVA, and MVAR peak demand to 0 (Megawatt meters) Reset kW, kVA, and kVAR peak demand to (all other models)
42003	Peak Current Demand Reset	UINT16	x1	Reset I peak demand.
42004	Digital Output #1	UINT16	x1	Refer to “Digital Output Control” on page 10.
42005	Digital Output #2	UINT16	x1	

Notes

1 Format is YYMM##### where YY and MM represent the manufacture date.

2 Identifies the Original Equipment Manufacturer (OEM).

3 Options codes that identify meter options. To see a table that cross-references Options Cards and options codes, refer to the *9200 Options Card Retrofit Instructions*.

4 Number of seconds that the meter has been powered up.

5 Device identification.

6 When Volts Mode is set to Delta or Delta Direct, these registers are invalid.

7 PVS = Programmable Voltage Scale. Ensure that scaling is compatible with the expected register value; this provides maximum resolution and prevents register overflow.

8 PCS = Programmable Current Scale. Ensure that scaling is compatible with the expected register value; this provides maximum resolution and prevents register overflow.

9 PnS = Programmable Neutral Current Scale. Ensure that scaling is compatible with the expected register value; this provides maximum resolution and prevents register overflow.

10 PPS = Programmable Power Scale. Ensure that scaling is compatible with the expected register value; this provides maximum resolution and prevents register overflow.

11 When it is read from that address the Low Order Word is fetched first.

12 These registers are duplicate of other UINT32 read-only registers differing only in that the High Order Word is fetched first.

[13] Digital output modes are: MW pulsing, MVAR pulsing, MVA pulsing, and digital control (Megawatt meters) or kW pulsing, kVAR pulsing, kVA pulsing, and digital control (all other models). For digital output control information, refer to "Digital Output Control" on page 10.

[14] The PML protocol is an ION compatible protocol used when other ACCESS devices are sharing a RS-485 network.

[15] The RTS Delay parameter defines the delay between the 9200 becoming ready to transmit data on the serial port, and the 9200 transmitting the data.

[16] Number of seconds that a display is shown before scrolling to the next, 0 = no scrolling (disabled).

[17] Number of seconds that a measurement value is held on the display before being refreshed; limited to 1-6 seconds.

[18] To use the Read-Write Control Map, it is recommended that your meter has firmware version 202 or later. For detailed information, contact Customer Service.