7500, 7600 and 8000 Series Meter

Advanced Digital Power / Instrumentation Package

Modicon Modbus

Serial Communications Protocol

Version 0.8

POWER MEASUREMENT

Limitation of Liability

For further assistance
please contact us at:

POWER MEASUREMENT

Worldwide Headquarters
2195 Keating Cross Road
Saanichton, BC
Canada V8M 2A5
Tel: 1-250-652-7100
Fax: 1-250-652-0411
Email:
support@pwrm.com

www.pwrm.com

Revision Date: June 5, 2003
70020-0114-08

Power Measurement Ltd. ("Power Measurement") reserves the right to make changes in the device or its specifications identified in this document without notice. Power Measurement advises customers to obtain the latest version of the device specifications before placing orders to verify that the information being relied upon by the customer is current.

Regardless of whether any remedy set forth herein fails of its essential purpose, except to the extent the following limitation is prohibited by applicable law, Power Measurement shall not, in any event or under any legal claim or theory (whether based on contract, indemnity, warranty, tort (including negligence and strict liability) or otherwise), be liable to the original purchaser or any other person or entity for special, indirect, incidental, punitive, liquidated, special or consequential damages whatsoever with respect to any purchased product, including, without limitation, business interruption, loss of use, profit or revenue, even if Power Measurement has been advised of the possibility of such damages. To the extent that a limitation or exclusion of consequential damages are prohibited by applicable law, then Power Measurement's liability shall be limited to twice the amount of the relevant purchased product. Not to limit the foregoing, a) Power Measurement shall not be liable for any claim (other than a claim solely for the breach of one of the warranties as outlined in the Terms and Conditions of Sale ("Warranties")) made by the original purchaser, its employees, agents, or contractors for any loss, damage, or expense incurred due to, caused by, or related to any purchased product; and b) the Warranties are the original purchaser's exclusive remedy and Power Measurement hereby expressly disclaims all other warranties, express or implied, including, without limitation, warranties of non-infringement and the implied warranties of merchantability and fitness for a particular purpose.

The Warranties shall not apply to any product that has been subject to alteration, accident, misuse, abuse, neglect or failure to exactly follow Power Measurement's instructions for operation and maintenance. Any technical assistance provided by Power Measurement's personnel or representatives in system design shall be deemed to be a proposal and not a recommendation. The responsibility for determining the feasibility of such proposals rests with the original purchaser and should be tested by the original purchaser. It is the original purchaser's responsibility to determine the suitability of any product and associated documentation for its purposes. The original purchaser acknowledges that 100% "up" time is not realizable because of possible hardware or software defects. The original purchaser recognizes that such defects and failures may cause inaccuracies or malfunctions. Only the terms expressed in the Warranties shall apply and no distributor, corporation or other entity, individual or employee of Power Measurement or any other entity is authorized to amend, modify or extend the Warranties in any way.

For more information, refer to the Terms and Conditions of Sale document on our website at: http://www.pwrm.com/Buy/

The information contained in this document is believed to be accurate at the time of publication, however, Power Measurement assumes no responsibility for any errors which may appear here and reserves the right to make changes without notice.

Covered by one or more of the following patents:
U.S. Patent No's 6397155, 6186842, 6185508, 6000034, 5995911, 5828576, 5736847, 5650936, 60493644, D459259, D458863, D435471, D432934, D429655, D429533.
Canadian Patent No's 2148076, 2148075.
Other patents pending.

ISO 9002-94
Registration
CERT # 002188

Contents

Revision History

Revision #	Revision Date	Description of Changes
Revision 0.1	March 2, 1999	Initial draft
Revision 0.2	March 16, 1999	Updated Modbus module default frameworks
Revision 0.3	August 20, 1999	Added 8400 ION references
Revision 0.4	May 30, 2001	Added Integrator setup registers 31-35 to Enumerated Setup Registers and Numeric Bounded Registers. Checked start/end addresses in the map for ESRs and NBRs. Added 8300 columns to External Booleon Register table. Double checked number of registers for each table in Appendix B. Added Modbus Slave Read registers. Added DRE registers that were missing for 2-6. Added MSR registers 5-10 to map. Added Analog inputs 1-4. Added Alert module 1. Checked Arithmetic module. Added Calibration Pulsers (5). Added Sag Swell module (1). Added Waveform recordeds. Combined 8000 Series ION, 75/ 7600 ION meter into one document. Added Modbus meter time set (Appendix D).
Revision 0.5	June 27, 2001	minor typos
Revision 0.6	May 7, 2002	Changed ION Reference to ION Programmer's Reference. Changed the wording in the note on page 26, "Modbus Slave Module Factory Default".
Revision 0.7	October 3, 2002	Inserted reference to ION 7300 Series Meter Modbus Protocol document register map on 14. Changed ION Programmer's Reference to online ION Programmer's Reference.
Revision 0.8	June 3, 2003	Added Modbus Slave Modules 16 to 19. At time of publishing, these modules pertain to the ION 8000 series meter, v240 and beyond.

Introduction

This document explains the Modbus protocol for ION meters. The ION meter performs Modbus communications by emulating the Modicon 984 Programmable Controller. 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.

Purpose of the Communications Protocol

The Modbus protocol allows data and setup information to be efficiently transferred between a Modbus Master Station and a Modbus Slave. This includes:

- interrogation of all meter data which are exported via the Modbus Slave ION Module.
- configuration and interrogation of all meter Module Numeric Bounded and Enumerated set-up registers.
- interrogation and control of the meter External Control ION Modules.

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 following rules define the protocol for information transfer between a Modbus Master device and the meter.

- All communications on the network conform to a MASTER/SLAVE scheme. In this scheme, information and data is transferred between a Modbus MASTER device and up to 32 SLAVE devices.
- 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 2: Modbus Functions Supported by the Meters" on page 8 lists the Modbus functions supported by the meter.

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 is contained in 16-bit or 32-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 (see example below).

Example (Big Endian):

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: CRC-16 Calculation" on page 24.

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. The table below describes the exception codes supported by the meter and the possible causes.

Table 1: 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 ION 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 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.

Table 2: 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.

Read Holding Registers Packet Structure

Read Registers Request Packet (Master station to meter)	Read Registers Response Packet (meter to Master station)
Unit ID/Slave Address (1 byte)	Unit ID/Slave Address (1 byte)
03 (Function code) (1byte)	03 (Function code) (1 byte)
Start Register (sr) (2 bytes)	Byte Count (2 x nr) (1 byte)
# of Registers to Read (nr) (2 bytes)	First Register in range (2 bytes)
CRC Checksum	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 all three voltage phases (A, B, C). These three parameters are exported via a Modbus Slave Module to Modbus registers 40011, 40012 and 40013, with a scaling factor of 10. In accordance with the Modbus protocol, register 40011 is numbered as 10 when transmitted. The request must read 3 registers starting at 10.

Slave address: 100 = 64 (hex)   Start register 10 = 000A (hex)

Request Packtet: white background denotes the DATA field of the packet.

Slave	Function	Start Register (40011)		# of Registers (3)		CRC Checksum	
64*	03	00	0A	00	03	2C	3C

Response Packet:

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

The Master station retrieves the data from the response:

Register 40011: 2ECE(hex) = 11982 (scaled: 1198.2)

Register 40012: 2EE8(hex) = 12008 (scaled: 1200.8)

Register 40013: 2F13(hex) = 12051 (scaled: 1205.1)

*The values shown in illustrated packets are 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 arewritten 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 Preset Multiple Registers request and response packet formats are shown in the following example transaction.

Preset Multiple Registers

Preset Registers Request Packet (Master station to meter)	Preset Registers Response Packet (meter to Master station)
Unit ID/Slave Address (1 byte)	Unit ID/Slave Address (1 byte)
16 (Function code) (1byte)	16 (Function code) (1 byte)
Start Register (sr) (2 bytes)	Start Register (sr) (2 bytes)
# of Registers to Write (nr) (2 bytes)	# of Registers Written (nr) (2 bytes)
Byte Count (2 x nr) (1 byte)	CRC Checksum (2 bytes)
First Register in range (2 bytes)	
Second Register in range (2 bytes)	
...	
CRC Checksum (2 bytes)	

NOTE

Except for the function field, the Preset Registers Response packet is identical in format to the Read Registers Request packet.

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 46001/2 and 46003/4. Register 46001 is numbered 6000. The request must write 4 registers starting at 6000.

Slave address: 200 = C8(hex) Start register 6000 = 1770 (hex)

Value 1: 1200 = = 0000 | 04B0 (hex) Value 2: 120 = 0000 | 0078 (hex)

Request Packtet: white background denotes the DATA field of the packet.

Slave	Function	Start Register (46001)		# of Registers (4)		Byte Count	Register 1		Register 2		Register 3		Register 4		CRC Checksum	
C8*	10	17	70	00	04	08	00	00	04	B0	00	00	00	78	8B	F8

Response Packet:

Slave	Function	Start Register (46001)		# of Registers (4)		CRC Checksum	
C8	10	17	70	00	04	D4	3C

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

Invalid Registers

In the meter Modbus register map, there are gaps between some registers. For example, the next register after *42232* is *42301*. Unmapped registers (*42233* through to *42300*) are INVALID. Invalid registers store no information.

When an invalid register is read, the data field is FFFF(hex). When an invalid register is written, the data field is not stored. The meter does not reject the request.

Meter Modbus Registers

The meter Modbus register map defines a set of parameters which are treated as HOLDING REGISTERS of the Modicon 984 PLC, having addresses ***4xxxx***. According to the Modbus protocol, in response to a request for register ***4xxxx*** of a particular slave device, the Modbus master reads register ***xxxx-1*** from the slave. For example, register ***40011*** corresponds to register 10.

There are four main classes of registers available via Modbus:

- Modbus Slave Module Output Registers
- External Control Registers
- Enumerated ION Module Setup Registers
- Numeric Bounded ION Module Setup Registers.

Modbus Slave Module Output Registers

The meter contains ION Modbus Slave Modules each capable of exporting up to sixteen ION registers into the Modbus protocol. Some modules are pre-configured with common meter values. The Slave module takes Numeric or Boolean type ION registers as input, scales and formats the input values according to a configurable setup, and makes the ION data available in a contiguous set of Modbus Holding Registers.

Modbus Slave Module output registers are located in the Modbus register map (from *40001* to *41800*). The actual location depends on the setup of the individual Modbus Slave Modules.

The Modbus Slave Module can scale and offset input values, and format the outputs in one of seven selectable formats:

- Unsigned 16-bit Integer Format
- Signed 16-bit Integer Format
- Unsigned 32-bit Integer Format
- Signed 32-bit Integer Format
- Unsigned 32-bit 'Modulus-10000' Format
- Signed 32-bit 'Modulus-10000' Format
- Packed Boolean Format

16-bit Integer Format

Unsigned and Signed 16-bit Integer Formats are the simplest formats. Each ION input register to the module corresponds to one 16-bit Modbus Holding Register output. 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.

32-bit Integer Format

To accommodate values that can reach beyond the 16-bit limitation, the Modbus Slave Module provides 32-bit integer format as an output option. In Signed and Unsigned 32-bit Integer Formats, each ION input register to the module corresponds to two 16-bit Modbus Holding Register outputs.

A 32-bit register represented in 32-bit Integer format is passed via communications as two 16-bit registers:

High-Order Register

- $register_{high}$=value/65536

Low-Order Register

- $register_{low}$= value **modulus** 65536
- value = $register_{high}$ x 65536 + $register_{low}$, or
- value = $register_{high} | register_{low}$

Example (Unsigned 32-bit):

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

- 12345678 = 00BC614E Hex
- $Register_{high}$ = 00BC Hex (unsigned) = 188
- $Register_{low}$ = 614E Hex (unsigned) = 24910
- Value = 188 x 65536 + 24910 = 12345678

In Unsigned 32-bit Integer Format, both the High-Order and Low-Order registers are unsigned 16-bit integers.

Example (Signed 32-bit):

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

- -12345678 = FF439EB2 Hex
- $Register_{high}$ = FF43 Hex (signed) = -189
- $Register_{low}$ = 9EB2 Hex (unsigned) = 40626
- value = -189 x 65536 + 40626 = -12345678

In Signed 32-bit Integer Format, the High-Order register is a signed 16-bit number, but the Low-Order register is unsigned.

32-bit 'Modulus-10000' Format

The Modulo-10000 (M10K) format breaks a 32-bit value into two 16-bit registers, according to the following relationship:

High-Order Register

- $register_{high}$=value/10000

Low-Order Register

- $register_{low}$= value **modulus**10000

The 32-bit value can be retrieved by the following calculation:

Value = $register_{high}$ x 10000 + $register_{low}$

Example (Unsigned):

Value 12345678 is passed in *unsigned* 32-bit Modulus-10000 format.

- $Register_{high}$: 1234 = 04D2 Hex
- $Register_{low}$: 5678 = 162E Hex
- Value = 1234 * 10000 + 5678 = 12345678

Example (Signed):

Value -12345678 is passed in *signed* 32-bit Modulus-10000 format. Both high and low are signed.

- $Register_{high}$: -1234 = FB2E Hex
- $Register_{low}$: -5678 = E9D2 Hex
- Value = -1234 * 10000 + -5678 = -12345678

Packed Boolean Format

Boolean ION registers can be packed into a single Modbus register via the Modbus Slave Module. When the Modbus Slave Module is configured to produce packed Boolean outputs each input register (to the module) corresponds to one bit in the single output register of the module. The relationship is left to right: the first input register corresponds to the left-most bit in the 16-bit output register, etc.

Example:

Six Boolean registers are linked to a Modbus Slave Module, which is configured for Packed Boolean output format. If the first three are valued 'False', and the remaining three are valued 'True', the output register value is:

Register: 0001110000000000 Bin = 1C00 Hex

If the first input register became 'True', the output register value changes to:

Register: 1001110000000000 Bin = 9C00 Hex

Meter Firmware Revision

All ION meters contain a firmware revision string which denotes the meter type and version (e.g. “7300V200” denotes version 200 of the 7300 meter).

The firmware revision string is available via Modbus at a fixed location in the Modbus register map. While the string may vary in length from one revision to the next, the set of Modbus registers used to represent the string spans the maximum possible firmware revision string length. On the meter, the firmware revision string appears in Modbus Holding registers *41901* to *41912*.

The format of the firmware revision string in Modbus follows a ‘C’ style string convention: a series of bytes representing ASCII characters terminated by a ‘null’ byte (value 00 Hex). In Modbus, each 16-bit holding register contains two ASCII characters.

The following table shows how the Modbus encoding of the string “7300V200” appears.

Table 3: Modbus string encoding

Register	Value (Hex)	ASCII	
41901	3733	’7’	’3’
41902	3030	’0’	’0’
41903	5632	’V’	’2’
41904	3030	’0’	’0’
41905	0000	NUL	NUL

The remainder of the firmware revision string registers (in the above case, *41906* to *41912*) contains null values (0000 Hex).

ION External Control Registers

All ION external control registers in the meter can be read and written via Modbus. This section describes how the registers appear to the Modbus protocol. There are three types of external control registers:

- External Pulse Control Registers
- External Boolean Control Registers
- External Numeric Control Registers

For a complete Modbus external control register map, see the *ION 7300 Series Meter Modbus Protocol* document (Appendix E) located on the Power Measurement web site.

External Pulse Registers

External Pulse registers interface to manually triggering events in the meter. For example, they can reset counters or timers, or pulse external equipment. All of the meter external pulse registers are available via Modbus.

Pulse registers are meaningful mainly for writing. Writing a nonzero value to a pulse register causes a pulse. Writing a zero value has no effect, but is acknowledged as a successful write operation. This feature provides the capability to 'skip' triggers when pulsing multiple registers in one request.

The meter's External Pulse registers are located in the Modbus register map from *42001* to *42032*.

Example:

A meter is pre-configured with external pulse modules. See "External Pulse Registers" on page 15.

The Modbus master requests to reset Min/Max, SWD, TD, and Integrators. The outgoing write request is to write 7 registers, starting at *42001*, with values 1, 0, 1, 1, 0, 0, and 1.

External Boolean Registers

ION External Boolean registers provide an interface to manually turn a signal ON or OFF. For example, these registers can enable or disable ION modules. The functionality depends on the meter configuration.

A value of one (1) for a Boolean register represents 'ON' or 'TRUE'. A value of zero (0) represents 'OFF' or 'FALSE'. Writing a value other than zero or one result in the value of one.

The meter's External Boolean registers are located in the Modbus register map from *42201* to *42212*.

External Numeric Registers

External Numeric registers can be set to a certain value. Consult the ION meter *User's Guide* and the online *ION Programmer's Reference* for an example of how and where these registers might be used.

The External Numeric registers are 32-bit values are represented in 32-bit Signed Integer Format (see section 32-bit Integer Format of this document). Each External Numeric register spans two 16-bit Modbus registers. The first Modbus register of the pair represents the high order word of the 32-bit value. The second Modbus register represents the low order word. The 32-bit value read from or written to an External Numeric register via Modbus is represented as a 32-bit signed integer value, therefore the range of possible values is -2,147,483,648 to +2,147,483,647.

The meter's External Numeric registers are located in the Modbus register map from *42301* to *42308*.

Enumerated ION Module Setup Registers

The Enumerator setup register is a major class of setup registers in ION modules. Enumerated registers are used where there is a list of options to choose from. For example, the Power Meter Module has the following options for Volts Mode: 4W-WYE, DELTA, SINGLE, DEMO, DIRECT-DELTA, and 3W-WYE.

In Modbus protocol, Enumeration register lists are represented by a numeric relationship. For example, with the Power Meter Module Volts Mode register, the following relationship is defined:

0 = 4W-WYE
1 = DELTA
2 = SINGLE
3 = DEMO
4 = 3W-WYE
5 = DIRECT-DELTA

All Enumerated ION Module setup registers on the meter are included in the Modbus register map. The register map details how enumerations are represented numerically in Modbus for each register.

Enumerated ION Module setup registers are located in the Modbus register map in order of ION handles. The following formula shows the relationship:

EnumAddr = 44001 + dec(EnumHandle-7800 hex)

Example:

A meter has a Modbus Slave Module #1 that is configured to export data in Unsigned 32-bit Integer Format. The ION handle for the Modbus Slave Module #1 Format register is 7A53 hex. The enumeration for 'Unsigned 32B' is 2.

Register Address	= 44001 + dec(7A53 hex - 7800 hex)
	= 44001 + dec(0253 hex)
	= 44001 + 595
	= 44596

A write request of value 2 to register *44596* makes this configuration change.

Numeric Bounded ION Module Setup Registers

The Numeric Bounded setup register is another major class of setup registers in ION modules. Examples of numeric bounded setup registers include Power Meter Module PT/CT Ratios, Communications Module Unit ID, etc.

Numeric Bounded registers are represented in Modbus in Signed 32-bit Integer Format (see section 32-bit Integer Format), where each ION Numeric Bounded register spans two 16-bit Modbus registers. Because of the Modbus register format, an absolute boundary of -2,147,483,648 to +2,147,483,647 is imposed on Numeric Bounded ION Module setup registers. Even if the ION register bounds are beyond the 32-bit signed integer boundary, the bounds are effectifely limited by Modbus capabilities.

All Numeric Bounded ION Module setup registers on the meter are included in the Modbus register map. The register map details the numeric bounds in Modbus for each register.

Like Enumerated ION Module setup registers, Numeric Bounded setup registers are located in the Modbus register map in order of ION handles. The following formula shows the relationship:

NBAddr = 46001 + 2 x dec(NBHandle-7000 hex)

Example:

A meter has a Modbus Slave Module #2 to be configured to export data to Modbus register base address 40027. Modbus Slave Module #1, with 16 ION inputs, is changed from 16 to 32 bit format, thus increasing the output register range of that module. Modbus Slave Module #2 must be configured to make room for the additional Modbus registers generated by Modbus Slave #1. The ION handle for the Modbus Slave Module #2 BaseAddr register is 7238 hex. To accomodate the 16 new output registers from Modbus Slave Module #1, the new BaseAddr for Modbus Slave Module #2 should be changed to 40043.

Register Address = 46001 + dec(7238 hex - 7000 hex) = 46001 + dec(238 hex)

= 46001 + 1136

= 47137

A write request of values 0 and 40043 to two registers starting at register *47137* make this configuration change.

Note in this example, if Modbus Slave Modules #3 and #4 were configured to export registers to an address range following Modbus Slave Module #2, they also must be reconfigured by a similar process.

Modbus Configuration

Modbus on the meter is configurable in two components:

- Protocol Configuration (Communications Module)
- Register Configuration (Modbus Slave Module)

Consult the online *ION Programmer's Reference* for a full functional description of the Communications and Modbus Slave Modules.

Modbus Protocol Configuration (Communications Module)

The meter Communications Module stores all setup information that applies to a serial protocol on a communications port. Setup registers in this module store both the protocol selected and all setup parameters for that protocol.

The setup registers for the Communications Modules on the meter are accessible via Modbus as fixed-location readable and writeable registers: See sections Enumerated ION Module Setup Registers and Numeric Bounded ION Module Setup Registers of this document for format details of these Modbus registers.

Table 4: Modbus Configuration Parameters

SETUP REGISTER	MODBUS REGISTER(S)
CM1 Baud Rate	44392
CM1 Protocol	44592
CM1 RTS Delay	46977 to 46978
CM1 Unit ID	46979 to 46980
CM2 Baud Rate	44590
CM2 Protocol	44593
CM2 RTS Delay	47125 to 47126
CM2 Unit ID	47129 to 47130
CM3 Baud Rate	44591
COM3 Protocol	44594
COM3 Unit ID	47131 to 47132
COM4 Protocol	45461

These registers are explained in the following sections.

Baud Rate

Each Communications Module on the meter has a Baud Rate register, which specifies the speed of serial communications.

The following values apply to all Communications Modules:

1 = 1200 Baud

2 = 2400 Baud

3 = 4800 Baud

4 = 9600 Baud

5 = 19200 Baud

Protocol

This register defines the serial protocol to be used on the communications port..

Protocol	Value
ION	0
Modbus RTU	1
Factory	3
DNP	4
GPS: Truetime/Datum	6
GPS: Arbiter	7
GPS: Arbiter-Vorne	8
Modbus Master	9
Ethergate	100
Modemgate	101

RTS Delay

The RTS Delay parameter defines a delay between when the ION meter is ready to transmit data on the serial port and when it starts transmitting data.

The RTS Delay parameter applies to all Communications Modules, and is expressed in milliseconds. The valid value range is from 0 to 1000 ms.

Unit ID

The Unit ID register defines the slave address for the protocol being used on the communications port.

In Modbus protocol, the Unit ID parameter defines the slave address used in Modbus packets for the device in question.

Since this parameter applies to both ION and Modbus protocols, the valid range for the parameter is defined to fit both protocols. Thus the range is specified as 1 to 9999. However, since the slave address range specified for Modbus is smaller than that of the Unit ID setup register, ***the valid range of this parameter is limited to 1 to 247***.

Modbus Register Configuration (Modbus Slave Module)

The meter Modbus Slave Module provides a configurable interface to export ION data to the Modbus protocol. Consult the online *ION Programmer's Reference* for a full description of this module.

The Modbus Slave Module is configurable in two ways:

- ION Registers are 'linked' to the module
- the Modbus Slave Module setup is altered

The first type of configuration is beyond the scope of the Modbus protocol. The meter comes with a set of default linkages for Modbus Slave Modules that suit a wide range of applications.

The second type of Modbus Slave Module configuration is accomplished via the meter display, the ION protocol, or the Modbus protocol.

The setup registers for the Modbus Slave Modules on the meter are available via Modbus for control and interrogation. See sections Enumerated ION Module Setup Registers and Numeric Bounded ION Module Setup Registers of this document for format details of these Modbus registers.

Table 5: Setup registers for the Modbus Slave Module

SETUP REGISTER	MODBUS REGISTER(S)
MSR1 Format	44596
MSR1 BaseAddr	47135 to 47136
MSR1 Scaling	44600
MSR1 InFull	47151 to 47152
MSR1 InZero	47143 to 47144
MSR1 OutFull	47167 to 47168
MSR1 OutZero	47159 to 47160
MSR2 Format	44597
MSR2 BaseAddr	47137 to 47138
MSR2 Scaling	44601
MSR2 InFull	47153 to 47154
MSR2 InZero	47145 to 47146
MSR2 OutFull	47169 to 47170
MSR2 OutZero	47161 to 47162
MSR3 Format	44598
MSR3 BaseAddr	47139 to 47140
MSR3 Scaling	44602
MSR3 InFull	47155 to 47156
MSR3 InZero	47147 to 47148
MSR3 OutFull	47171 to 47172
MSR3 OutZero	47163 to 47164
MSR4 Format	44599

SETUP REGISTER	MODBUS REGISTER(S)
MSR4 BaseAddr	47141 to 47142
MSR4 Scaling	44603
MSR4 InFull	47157 to 47158
MSR4 InZero	47149 to 47150
MSR4 OutFull	47173 to 47174
MSR4 OutZero	47165 to 47166
MSR5 Format	45196
MSR5 BaseAddr	49915 to 49916
MSR5 Scaling	45202
MSR5 InFull	49939 to 49940
MSR5 InZero	49927 to 49928
MSR5 OutFull	49963 to 49964
MSR5 OutZero	49951 to 49952
MSR6 Format	45197
MSR6 BaseAddr	49917 to 49918
MSR6 Scaling	45203
MSR6 InFull	49941 to 49942
MSR6 InZero	49929 to 49930
MSR6 OutFull	49965 to 49966
MSR6 OutZero	49953 to 49954
MSR7 Format	45198
MSR7 BaseAddr	49919 to 49920
MSR7 Scaling	45204
MSR7 InFull	49943 to 49944
MSR7 InZero	49931 to 49932
MSR7 OutFull	49967 to 49968
MSR7 OutZero	49955 to 49956
MSR8 Format	45199
MSR8 BaseAddr	49921 to 49922
MSR8 Scaling	45205
MSR8 InFull	49945 to 49946
MSR8 InZero	49933 to 49934
MSR8 OutFull	49969 to 49970
MSR8 OutZero	49957 to 49958

SETUP REGISTER	MODBUS REGISTER(S)
MSR9 Format	45200
MSR9 BaseAddr	49923 to 49924
MSR9 Scaling	45206
MSR9 InFull	49947 to 49948
MSR9 InZero	49935 to 49936
MSR9 OutFull	49971 to 49972
MSR9 OutZero	49959 to 49960
MSR10 Format	45201
MSR10 BaseAddr	49925 to 49926
MSR10 Scaling	45207
MSR10 InFull	49949 to 49950
MSR10 InZero	49937 to 49938
MSR10 OutFull	49973 to 49974
MSR10 OutZero	49961 to 49962

Format

As described in section Modbus Slave Module Output Registers of this document, the Modbus Slave Modules can export ION data to Modbus Holding registers in a variety of formats. These formats are selectable via the Format setup register of the Modbus Slave Module. The following values are valid Format selections:

0 = Unsigned 16B

1 = Signed 16B

2 = Unsigned 32B

3 = Signed 32B

4 = Unsigned 32B-M10K

5 = Signed 32B-M10K

6 = Packed Boolean

BaseAddr

The BaseAddr setup register defines the starting Modbus register address to which the Modbus Slave Module exports ION data. The valid range for this setup register is 40001 to 41800.

Scaling

The Modbus Slave Module can scale and offset input values to fit within the output range for the selected format. The Scaling setup register selects if scaling (as defined by InZero, InFull, OutZero, and OutFull) is applied to the inputs. The following values are valid for the Scaling setup register:

0 = No

1 = Yes

InZero, InFull

If Scaling is set to YES for a Modbus Slave Module, the input values are scaled according to a formula derived partly from the InZero, InFull setup registers. Input values falling at or below InZero are represented as OutZero. Input values falling at or above InFull are represented as OutFull. Input values between InZero and InFull are represented as a proportionate value between OutZero and OutFull.

InZero and InFull are defined to range from $-1x10^{38}$ to $+1x10^{38}$, but via Modbus, these registers are represented in Signed 32-bit Integer format, so the integer bounds of -2,147,483,648 to +2,147,483,647 are imposed upon these registers.

OutZero, OutFull

If Scaling is set to YES, the input values to the Modbus Slave Module are scaled by a formula derived partly from OutZero, OutFull. The absolute range of these registers is -2, 147, 483, 647 to +2, 147, 483, 647, but the valid range varies depending on the selected Format for the Modbus Slave Module. The following chart shows the OutZero, OutFull ranges for the various Formats:

Table 6: Out Zero and Out Full ranges for Modbus formats

Format	Low Bound	High Bound
Unsigned 16B	0	+65535
Signed 16B	-32767	+32767
Unsigned 32B	0	+2,147,478,647
Signed 32B	-2,147,478,647	+2,147,478,647
Unsigned 32B-M10K	0	+65,535,999
Signed 32B-M10K	-32,767,999	+32,767,999
Packed Boolean	N/A	N/A

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.

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: Modbus Slave Module Factory Default

NOTE

If your ION meter's form factor does not support the parameter, it will be undefined. For example, line-to-neutral values from a Form 35S ION 8500 will be undefined.

Modbus Slave Module #1

Format: ***Unsigned 16 bit***

Base Address: ***40011***

Scaling: ***Yes***

In Zero, In Full: ***0, +6553***

Out Zero, Out Full: ***0, +65530***

Input	Modbus Register	Parameter
Source #1	40011	Vln a
Source #2	40012	Vln b
Source #3	40013	Vln c
Source #4	40014	Vln avg
Source #5	40015	Vll ab
Source #6	40016	Vll bc
Source #7	40017	Vll ca
Source #8	40018	Vll avg
Source #9	40019	I a
Source #10	40020	I b
Source #11	40021	I c
Source #12	40022	I avg
Source #13	40023	V unbal
Source #14	40024	I unbal
Source #15	40025	Freq
Source #16	40026	I 4

Modbus Slave Module #2

Format: ***Signed 32 bit***

Base Address: ***40027***

Scaling: ***Yes***

In Zero, In Full: ***-214748364, +214748364***

Out Zero, Out Full: ***-2147483640, +2147483640***

Input	Modbus Registers	Parameter
Source #1	40027 to 40028	kW a
Source #2	40029 to 40030	kW b
Source #3	40031 to 40032	kW c
Source #4	40033 to 40034	kW tot
Source #5	40035 to 40036	kVAR a
Source #6	40037 to 40038	kVAR b
Source #7	40039 to 40040	kVAR c
Source #8	40041 to 40042	kVAR tot
Source #9	40043 to 40044	kVA a
Source #10	40045 to 40046	kVA b
Source #11	40047 to 40048	kVA c
Source #12	40049 to 40050	kVA tot
Source #13	40051 to 40052	pf sign a
Source #14	40053 to 40054	pf sign b
Source #15	40055 to 40056	pf sign c
Source #16	40057 to 40058	pf sign tot

Modbus Slave Module #3

Format: ***Signed 32 bit***

Base Address: ***40059***

Scaling: ***Yes***

In Zero, In Full: ***-214748364, +214748364***

Out Zero, Out Full: ***-2147483640, +2147483640***

Input	Modbus Registers	Parameter
Source #1	40059 to 40060	Vll avg max
Source #2	40061 to 40062	I avg max
Source #3	40063 to 40064	Kw total max
Source #4	40065 to 40066	kVAR total max
Source #5	40067 to 40068	kVA total max
Source #6	40069 to 40070	Freq max
Source #7	40071 to 40072	Vll avg min
Source #8	40073 to 40074	Iavg min
Source #9	40075 to 40076	Freq min
Source #10	40077 to 40078	kW sd* del-rec
Source #11	40079 to 40080	kVA sd* del+rec
Source #12	40081 to 40082	kVAR sd* del-rec
Source #13	40083 to 40084	kW sd* max del-rec
Source #14	40085 to 40086	kVA sd* max del+rec
Source #15	40087 to 40088	kVAR sd* max del-rec
Source #16	40089 to 40090	Phase Rev(ersal)

*sd = sliding window

Modbus Slave Module #4

Format: ***Signed 32 bit-M10K***

Base Address: ***40091***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	40091 to 40092	kWh del
Source #2	40093 to 40094	kWh rec
Source #3	40095 to 40096	kWh del+rec
Source #4	40097 to 40098	kWh del-rec
Source #5	40099 to 40100	kVARh del
Source #6	40101 to 40102	kVARh rec
Source #7	40103 to 40104	kVARh del+rec
Source #8	40105 to 40106	kVARh del-rec
Source #9	40107 to 40108	kVAh del+rec
Source #10	40109 to 40110	V1 THD mx
Source #11	40111 to 40112	V2 THD mx
Source #12	40113 to 40114	V3 THD mx
Source #13	40115 to 40116	I1 THD mx
Source #14	40117 to 40118	I2 THD mx
Source #15	40119 to 40120	I3 THD mx

Modbus Slave Module #5

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

NOTE

Source Inputs #1 through #6 also apply to the ION 8500 meter.

Format: ***Unsigned 16 bit***

Base Address: ***41000***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1*	41000	PO V1-Flicker N
Source #2*	41001	PO V1-Flicker N1
Source #3*	41002	PO V2-Flicker N
Source #4*	41003	PO V2-Flicker N1
Source #5*	41004	PO V3-Flicker N
Source #6*	41005	PO V3-Flicker N1
Source #7	41006	PO Freq N
Source #8	41007	PO Freq N1
Source #9	41008	PO Freq N2
Source #10	41009	PO V1-Mag N
Source #11	41010	PO V1-Mag N1
Source #12	41011	PO V2-Mag N
Source #13	41012	PO V2-Mag N1
Source #14	41013	PO V3-Mag N
Source #15	41014	PO V3-Mag N1
Source#16	41015	PO Vunbal N

PO = Observation Period, * applies to ION 8500 meter

Modbus Slave Module #6

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41016***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41016	PO Vunbal N1
Source #2	41017	PO V1-MSignal N
Source #3	41018	PO V1-MSignal N1
Source #4	41019	PO V2-MSignal N
Source #5	41020	PO V2-MSignal N1
Source #6	41021	PO V3-MSignal N
Source #7	41022	PO V3-MSignal N1
Source #8	41023	PO V1-Harmonic N
Source #9	41024	PO V1-Harmonic N1
Source #10	41025	PO V1-Harmonic N2
Source #11	41026	PO V2-Harmonic N
Source #12	41027	PO V2-Harmonic N1
Source #13	41028	PO V2-Harmonic N2
Source #14	41029	PO V3-Harmonic N
Source #15	41030	PO V3-Harmonic N1
Source#16	41031	PO V3-Harmonic N2

PO = Observation Period, M = Mains

Modbus Slave Module #7

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41032***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41032	PO V1-Inthrm N
Source #2	41033	PO V1-Inthrm N1
Source #3	41034	PO V2-Inthrm N
Source #4	41035	PO V2-Inthrm N1
Source #5	41036	PO V3-Inthrm N
Source #6	41037	PO V3-Inthrm N1
Source #7	41038	PO V1-Dip N11
Source #8	41039	PO V1-Dip N12
Source #9	41040	PO V1-Dip N13
Source #10	41041	PO V1-Dip N14
Source #11	41042	PO V1-Dip N21
Source #12	41043	PO V1-Dip N22
Source #13	41044	PO V1-Dip N23
Source #14	41045	PO V1-Dip N24
Source #15	41046	PO V1-Dip N31
Source#16	41047	PO V1-Dip N32

PO = Observation Period

Modbus Slave Module #8

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41048***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41048	PO V1-Dip N33
Source #2	41049	PO V1-Dip N34
Source #3	41050	PO V1-Dip N41
Source #4	41051	PO V1-Dip N42
Source #5	41052	PO V1-Dip N43
Source #6	41053	PO V1-Dip N44
Source #7	41054	PO V1-Dip N51
Source #8	41055	PO V1-Dip N52
Source #9	41056	PO V1-Dip N53
Source #10	41057	PO V1-Dip N54
Source #11	41058	PO V1-Dip N61
Source #12	41059	PO V1-Dip N62
Source #13	41060	PO V1-Dip N63
Source #14	41061	PO V1-Dip N64
Source #15	41062	PO V2-Dip N11
Source#16	41063	PO V2-Dip N12

PO = Observation Period

Modbus Slave Module #9

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41064***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41064	PO V2-Dip N13
Source #2	41065	PO V2-Dip N14
Source #3	41066	PO V2-Dip N21
Source #4	41067	PO V2-Dip N22
Source #5	41068	PO V2-Dip N23
Source #6	41069	PO V2-Dip N24
Source #7	41070	PO V2-Dip N31
Source #8	41071	PO V2-Dip N32
Source #9	41072	PO V2-Dip N33
Source #10	41073	PO V2-Dip N34
Source #11	41074	PO V2-Dip N41
Source #12	41075	PO V2-Dip N42
Source #13	41076	PO V2-Dip N43
Source #14	41077	PO V2-Dip N44
Source #15	41078	PO V2-Dip N51
Source#16	41079	PO V2-Dip N52

PO = Observation Period

Modbus Slave Module #10

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41080***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41080	PO V2-Dip N53
Source #2	41081	PO V2-Dip N54
Source #3	41082	PO V2-Dip N61
Source #4	41083	PO V2-Dip N62
Source #5	41084	PO V2-Dip N63
Source #6	41085	PO V2-Dip N64
Source #7	41086	PO V3-Dip N11
Source #8	41087	PO V3-Dip N12
Source #9	41088	PO V3-Dip N13
Source #10	41089	PO V3-Dip N14
Source #11	41090	PO V3-Dip N21
Source #12	41091	PO V3-Dip N22
Source #13	41092	PO V3-Dip N23
Source #14	41093	PO V3-Dip N24
Source #15	41094	PO V3-Dip N31
Source#16	41095	PO V3-Dip N32

PO = Observation Period

Modbus Slave Module #11

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41096***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41096	PO V3-Dip N33
Source #2	41097	PO V3-Dip N34
Source #3	41098	PO V3-Dip N41
Source #4	41099	PO V3-Dip N42
Source #5	41100	PO V3-Dip N43
Source #6	41101	PO V3-Dip N44
Source #7	41102	PO V3-Dip N51
Source #8	41103	PO V3-Dip N52
Source #9	41104	PO V3-Dip N53
Source #10	41105	PO V3-Dip N54
Source #11	41106	PO V3-Dip N61
Source #12	41107	PO V3-Dip N62
Source #13	41108	PO V3-Dip N63
Source #14	41109	PO V3-Dip N64
Source #15	41110	PO V1-Intrpt N1
Source#16	41111	PO V1-Intrpt N2

PO = Observation Period, Intrpt = Interruptions

Modbus Slave Module #12

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41112***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41112	PO V1-Intrpt N3
Source #2	41113	PO V2-Intrpt N1
Source #3	41114	PO V2-Intrpt N2
Source #4	41115	PO V2-Intrpt N3
Source #5	41116	PO V3-Intrpt N1
Source #6	41117	PO V3-Intrpt N2
Source #7	41118	PO V3-Intrpt N3
Source #8	41119	PO V1-Ovlt N11
Source #9	41120	PO V1-Ovlt N12
Source #10	41121	PO V1-Ovlt N13
Source #11	41122	PO V1-Ovlt N14
Source #12	41123	PO V1-Ovlt N15
Source #13	41124	PO V1-Ovlt N21
Source #14	41125	PO V1-Ovlt N22
Source #15	41126	PO V1-Ovlt N23
Source#16	41127	PO V1-Ovlt N24

PO = Observation Period, Ovlt = Over Voltage

Modbus Slave Module #13

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41128***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41128	PO V1-Ovlt N25
Source #2	41129	PO V1-Ovlt N31
Source #3	41130	PO V1-Ovlt N32
Source #4	41131	PO V1-Ovlt N33
Source #5	41132	PO V1-Ovlt N34
Source #6	41133	PO V1-Ovlt N35
Source #7	41134	PO V2-Ovlt N11
Source #8	41135	PO V2-Ovlt N12
Source #9	41136	PO V2-Ovlt N13
Source #10	41137	PO V2-Ovlt N14
Source #11	41138	PO V2-Ovlt N15
Source #12	41139	PO V2-Ovlt N21
Source #13	41140	PO V2-Ovlt N22
Source #14	41141	PO V2-Ovlt N23
Source #15	41142	PO V2-Ovlt N24
Source#16	41143	PO V2-Ovlt N25

PO = Observation Period, Ovlt = Over Voltage

Modbus Slave Module #14

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41144***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41144	PO V2-Ovlt N31
Source #2	41145	PO V2-Ovlt N32
Source #3	41146	PO V2-Ovlt N33
Source #4	41147	PO V2-Ovlt N34
Source #5	41148	PO V2-Ovlt N35
Source #6	41149	PO V3-Ovlt N11
Source #7	41150	PO V3-Ovlt N12
Source #8	41151	PO V3-Ovlt N13
Source #9	41152	PO V3-Ovlt N14
Source #10	41153	PO V3-Ovlt N15
Source #11	41154	PO V3-Ovlt N21
Source #12	41155	PO V3-Ovlt N22
Source #13	41156	PO V3-Ovlt N23
Source #14	41157	PO V3-Ovlt N24
Source #15	41158	PO V3-Ovlt N25
Source#16	41159	PO V3-Ovlt N31

PO = Observation Period, Ovlt = Over Voltage

Modbus Slave Module #15

Modules #5 through #15 apply to ION 7600 meter only and pertain to the EN50160 Power Quality Standard; refer to the *User's Guide* for more information.

Format: ***Unsigned 16 bit***

Base Address: ***41160***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	41160	PO V3-Ovlt N32
Source #2	41161	PO V3-Ovlt N33
Source #3	41162	PO V3-Ovlt N34
Source #4	41163	PO V3-Ovlt N35

PO = Observation Period, Ovlt = Over Voltage

Modbus Slave Module: Amp/Freq/Unbal

The Amp/Freq/Unbal Module applies to the ION 8000 series meter, v240 and beyond.

Format: ***Unsigned 16 bit***

Base Address: ***40150***

Scaling: ***Yes***

Input	Modbus Registers	Parameter
Source #1	40150	Ia
Source #2	40151	Ib
Source #3	40152	Ic
Source #4	40153	I4
Source #5	40154	not used
Source #6	40155	I avg
Source #7	40156	I avg min
Source #8	40157	I avg max
Source #9	40158	I avg mean
Source #10	40159	Freq
Source #11	40160	Freq min
Source #12	40161	Freq max
Source #13	40162	Freq mean
Source #14	40163	V unbal
Source #15	40164	I unbal
Source#16	40165	Phase Reversal

Modbus Slave Module: Volts

The Volts Module applies to the ION 8000 series meter, v240 and beyond.

Format: ***Unsigned 32 bit***

Base Address: ***40166***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	40166	Vln a
Source #2	40168	Vln b
Source #3	40170	Vln c
Source #4	40172	Vln avg
Source #5	40174	Vln avg max
Source #6	40176	Vln avg mean
Source #7	40178	Vll ab
Source #8	40180	Vll bc
Source #9	40182	Bll ca
Source #10	40184	Vll avg
Source #11	40186	Vll avg max
Source #12	40188	Vll avg mean
Source #13	40190	
Source #14	40192	
Source #15	40194	
Source#16	40196	

Modbus Slave Module: kW/kVAr/kVA

The kW/kVAr/kVA Module applies to the ION 8000 series meter, v240 and beyond.

Format: ***Signed 32 bit***

Base Address: ***40198***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	40198	kW a
Source #2	40200	kW b
Source #3	40202	kW c
Source #4	40204	kW tot
Source #5	40206	kW tot max
Source #6	40208	kVAR a
Source #7	40210	kVAR b
Source #8	40212	kVAR c
Source #9	40214	kVAR tot
Source #10	40216	kVAR tot max
Source #11	40218	kVA a
Source #12	40220	kVA b
Source #13	40222	kVA c
Source #14	40224	kVA tot
Source #15	40226	kVA tot max
Source#16	40228	

Modbus Slave Module: kWh/kVArh

The kWh/kVArh Module applies to the ION 8000 series meter, v240 and beyond.

Format: ***Signed 32 bit***

Base Address: ***40230***

Scaling: ***No***

Input	Modbus Registers	Parameter
Source #1	40230	kWh del
Source #2	40232	kWh rec
Source #3	40234	kVARh del
Source #4	40236	kVARh rec
Source #5	40238	kVAh
Source #6	40240	
Source #7	40242	
Source #8	40244	
Source #9	40246	
Source #10	40248	
Source #11	40250	
Source #12	40252	
Source #13	40254	
Source #14	40256	
Source #15	40258	
Source#16	40260	

Modbus Slave Module: PF/THD/Kfactor

The PF/THD/Kfactor Module applies to the ION 8000 series meter, v240 and beyond.

Format: ***Signed 16 bit***

Base Address: ***40262***

Scaling: ***Yes***

Input	Modbus Registers	Parameter
Source #1	40262	Power Factor a
Source #2	40263	Power Factor b
Source #3	40264	Power Factor c
Source #4	40265	Power Factor total
Source #5	40266	THD V1 max
Source #6	40267	THD V2 max
Source #7	40268	THD V3 max
Source #8	40269	THD I1 max
Source #9	40270	THD I2 max
Source #10	40271	THD I3 max
Source #11	40272	I1 K Factor
Source #12	40273	I2 K Factor
Source #13	40274	I3 K Factor
Source #14	40275	I1 Crest Factor
Source #15	40276	I2 Crest Factor
Source#16	40277	I3 Crest Factor

Appendix C: Data Record / Modbus Map

This appendix contains the Data Record/Modbus register map for ION meters.

Modbus Data Recorder Registers

ION meters provide data from Data Recorder Modules to be exported into Modbus Registers. The Register Map is a dynamic map and dependent on the configuration of Data Recorder Source inputs. Consult the online *ION Programmer's Reference* for a description of Data Recorder Modules.

Modbus Data Recorder Map

Modbus Register	Contents
43001 to 43011	Record Availability and Selection Block
43012 to 43125	Data Record Block
43126 to 43137	Reserved Registers
43138 to 43153	Source Input Handle ID

Modbus Data Recorder Retrieval

To retrieve Data Record via Modbus communications the following steps must be followed:

1. Ensure the Data Recorder is on line. See the online *ION Programmer's Reference* for Data Recorder Module descriptions.
2. Write the Data Recorder Module Number to Modbus Register 43001. If an invalid Data Recorder Module Number is written, a Modbus Exception is returned.
3. Determing a valid Starting Record with a Read of Modbus Registers 43001 through 43011. This returns the Modbus Record Availability and Selection. All valid Record Numbers lie in the range of the Oldest Record Number (Modbus Registers 43008 and 43009) and the Newest Record Number (Modbus Registers 43010 and 43011).
4. After a valid Record Number is determined write it to Modbus Registers 43002 and 43003 (Master's Request for Starting Record) so a valid data is cached and read back.
5. A Read returns the data for each available record starting at record number written to Modbus Registers 43002 and 43003. The number of records returned depends on the number of Source Inputs connected to the Data Recorder and the number of records available with respect to the Start Record.
6. Repeat steps 3 through 6 for new records.

NOTE

All data is cached and can be read back at any time until a new write is requested. Any setup changes in the Data Recorder Module clears all cached Data Records.

Modbus Record Availability and Selection Block Registers

Modbus Register	# of Modbus Registers	Description	Format	Properties
43001	1	Data Recorder Module Number - write to this register with the data recorder module number you want to access.	UINT16	Read / Write
43002, 43003	2	Master's Request for Starting Record - write to these registers with the starting record number. Write the high order word to register 43002 and the low order word to register 43003.	UINT32	Read / Write
43004	1	Number of Source Inputs - read this register to return the number of source input connected to the data recorder module (register 43001).	UINT16	Read
43005	1	Module Setup Count - read this register to return the module setup count. A change in the module setup count reflects a change in the data recorder module setup.	UINT16	Read
43006	1	Maximum Number of Records / Request - read this register to return the maximum number of records per request.	UINT16	Read
43007	1	Number of Available Records / Request - read this register to return the number of available record per request.	UINT16	Read
43008, 43009	2	Oldest Record Number - read these registers to return the oldest available record number. Register 43008 returns the high order word and register 43009 returns the low order word.	UINT32	Read
43010, 43011	2	Newest Record Number - read these registers to return the newest available record number. Register 43010 returns the high order word and register 43011 return the low order word.	UINT32	Read

Modbus registers 43001 through 43011 contain the Data Recorder Record information necessary to retrieve valid records. A valid Data Recorder Module Number must be written to Modbus Register 43001 prior to reading any Modbus Data Recorder Registers otherwise a Modbus exception will be returned.

Modbus Data Record Block Registers

Modbus registers 43012 through 43125 contain the Record Number, Time Stamp, and Source Input Data for each record retrieved. This Modbus mapping is dynamic dependant on the number of source inputs connected to the Data Recorder Module.

The Record Number is returned as an unsigned 32-bit value stored in two Modbus registers. The first register is the high order followed by the low order second register.

The Time Stamp Seconds is returned as an unsigned 32-bit value stored in two Modbus registers. The first register is the high order followed by the low order second register. The format is UNIX time (UTC). Consult the online *ION Programmer's Reference* for a description of the Clock Module time format.

The Time Stamp MicroSeconds is returned as an unsigned 32-bit value stored in two Modbus registers. The first register is the high order followed by the low order second register. The format is absolute time in micro seconds.

The Source Input Data is returned as a Float value stored in two Modbus registers. The first register is the high order followed by the low order second register. The format is IEEE-754.

The following is an example of a Data Recorder Module with one source input connected (14 records maximum):

Modbus Register	# of Modbus Registers	Description	Format	Properties
43012	2	Record Number (x)	UINT32	Read
43014	2	UTC Seconds	UINT32	Read
43016	2	UTC MicroSeconds	UINT32	Read
43018	2	Source 1 Input Data	FLOAT	Read
43020	2	Record Number (x+1)	UINT32	Read
43022	2	UTC Seconds	UINT32	Read
43024	2	UTC MicroSeconds	UINT32	Read
43026	2	Source 1 Input Data	FLOAT	Read
43116	2	Record Number (x+13)	UINT32	Read
43118	2	UTC Seconds	UINT32	Read
43120	2	UTC MicroSeconds	UINT32	Read
43122	2	Source 1 Input Data	FLOAT	Read

The following is an example of a Data Recorder Module with 16 source inputs connected (3 records maximum):

Modbus Register	# of Modbus Registers	Description	Format	Properties
43012	2	Record Number (x)	UINT32	Read
43014	2	UTC Seconds	UINT32	Read
43016	2	UTC MicroSeconds	UINT32	Read
43018	2	Source 1 Input Data	FLOAT	Read
43020	2	Source 2 Input Data	FLOAT	Read
43022	2	Source 3 Input Data	FLOAT	Read
43024	2	Source 4 Input Data	FLOAT	Read
43026	2	Source 5 Input Data	FLOAT	Read
43028	2	Source 6 Input Data	FLOAT	Read
43030	2	Source 7 Input Data	FLOAT	Read
43032	2	Source 8 Input Data	FLOAT	Read
43034	2	Source 9 Input Data	FLOAT	Read
43036	2	Source 10 Input Data	FLOAT	Read
43038	2	Source 11 Input Data	FLOAT	Read
43040	2	Source 12 Input Data	FLOAT	Read
43042	2	Source 13 Input Data	FLOAT	Read
43044	2	Source 14 Input Data	FLOAT	Read
43046	2	Source 15 Input Data	FLOAT	Read
43048	2	Source 16 Input Data	FLOAT	Read
43088	2	Record Number (x+2)	UINT32	Read
43090	2	UTC Seconds	UINT32	Read
43092	2	UTC MicroSeconds	UINT32	Read
43094	2	Source 1 Input Data	FLOAT	Read
43096	2	Source 2 Input Data	FLOAT	Read
43098	2	Source 3 Input Data	FLOAT	Read
43100	2	Source 4 Input Data	FLOAT	Read
43102	2	Source 5 Input Data	FLOAT	Read
43104	2	Source 6 Input Data	FLOAT	Read
43106	2	Source 7 Input Data	FLOAT	Read
43108	2	Source 8 Input Data	FLOAT	Read
43110	2	Source 9 Input Data	FLOAT	Read

Modbus Register	# of Modbus Registers	Description	Format	Properties
43112	2	Source 10 Input Data	FLOAT	Read
43114	2	Source 11 Input Data	FLOAT	Read
43116	2	Source 12 Input Data	FLOAT	Read
43118	2	Source 13 Input Data	FLOAT	Read
43120	2	Source 14 Input Data	FLOAT	Read
43122	2	Source 15 Input Data	FLOAT	Read
43124	2	Source 16 Input Data	FLOAT	Read

Modbus Handle ID Registers

Modbus registers 43138 through 43153 contain the Handle ID's for the Source Inputs.

Modbus Register	# of Modbus Registers	Description	Format	Properties
43138	1	Source 1 Handle ID	UINT16	Read
43139	1	Source 2 Handle ID	UINT16	Read
43140	1	Source 3 Handle ID	UINT16	Read
43141	1	Source 4 Handle ID	UINT16	Read
43142	1	Source 5 Handle ID	UINT16	Read
43143	1	Source 6 Handle ID	UINT16	Read
43144	1	Source 7 Handle ID	UINT16	Read
43145	1	Source 8 Handle ID	UINT16	Read
43146	1	Source 9 Handle ID	UINT16	Read
43147	1	Source 10 Handle ID	UINT16	Read
43148	1	Source 11 Handle ID	UINT16	Read
43149	1	Source 12 Handle ID	UINT16	Read
43150	1	Source 13 Handle ID	UINT16	Read
43151	1	Source 14 Handle ID	UINT16	Read
43152	1	Source 15 Handle ID	UINT16	Read
43153	1	Source 16 Handle ID	UINT16	Read

Appendix D: Modbus Meter Time Set

This appendix contains the Modbus Meter UNIX Time Set function of ION meters.

Modbus Meter Time Set

Unix Time (UTC) Seconds is an unsigned 32-bit value stored in two Modbus registers. The first register is the high order followed by the low order second register. Consult the online *ION Programmer's Reference* for a description of the Clock Module time format.

UTC microseconds is an unsigned 32-bit value stored in two Modbus registers. The first register is the high order followed by the low order second register. The format is absolute time in MicroSeconds.

Only resolution by seconds is supported when setting Meter Time via Modbus.

Modbus Register	# of Modbus Registers	Description	Format	Properties
41926	2	UTC Seconds	UINT32	Read / Write
41928	2	UTC microseconds	UINT32	Read

Modbus Time Set

To set the Meter time via Modbus communications do the following:

1. Set the ION Clock Module Time Sync Source Register to the Modbus communications port.
2. Write the UNIX time in seconds as an unsigned 32-bit value to Modbus Registers 41926 (high order) and 41927 (low order).