
The Information contained in this guide is for use only by installers trained to make electrical power & communications installations and is intended to describe the communications function of the products covered. However, Tyco Electronics has no control over the field conditions which influence product installation. It is the user's responsibility to determine the suitability of the installation method in the user's field conditions. Tyco Electronics' only obligations are those in Tyco Electronics' standard Conditions of Sale for this product and in no case will Tyco Electronics be liable for any other incidental, indirect or consequential damages arising from the use or misuse of the products. Crompton is a trade mark.



Tyco Electronics UK Limited
Crompton Instruments

Freebournes Road, Witham, Essex, CM8 3AH, UK
Phone: +44 1376 509 509 Fax: +44 1376 509 511

<http://energy.tycoelectronics.com>

Introducing the MODBUS Protocol

This document is intended to provide an introduction to the MODBUS implementation of Integra and SPR products. The document was generated in response to repeated questions from the field. The section at the end of the document, "When things go wrong", identifies a number of potential problems some of which have been experienced in the field.

The MODBUS protocol "defines a message structure that controllers will recognise and use, regardless of the type of networks over which they communicate. It describes the process a controller uses to request access to another device, how it will respond to requests from the other devices, and how errors will be detected and reported. It establishes a common format for the layout and contents of message fields." - Modbus Protocol Reference Guide, PI-MBUS-300 Rev. J. This document may be purchased from "Groupe Schneider, Basingstoke, England 44 (0) 1256 843184.

There are a number of hardware configurations used for MODBUS networks, this guide will consider only the two wire RS485 network as that is the configuration currently supported by the SPR and Integra family of products.

MODBUS is a trademark of AEG Schneider Automation Inc

Contents

INTRODUCING THE MODBUS PROTOCOL.....	3
CONTENTS	4
WHAT IS RS485?.....	5
WHAT IS HALF DUPLEX?.....	5
CONNECTING THE INSTRUMENTS.....	5
WHICH IS A AND WHICH IS B?	6
MODBUS MESSAGES.....	7
MODBUS MESSAGE FORMAT	7
THE SERIAL TRANSMISSION MODES	9
ASCII MODE	9
RTU MODE	10
MODBUS MESSAGE TIMING (RTU MODE).....	10
HOW CHARACTERS ARE TRANSMITTED SERIALY	11
ERROR CHECKING METHODS	11
PARITY CHECKING.....	12
FUNCTION CODES.....	13
PRODUCT SPECIFIC CONSIDERATIONS.	13
MODBUS COMMANDS SUPPORTED	15
READ INPUT REGISTERS	15
READ HOLDING REGISTERS.....	16
WRITE HOLDING REGISTERS	17
DIAGNOSTICS.....	18
WHEN THINGS GO WRONG.....	19
PRODUCT INFORMATION	20
EXCEPTION CODES.....	20
START UP CONSIDERATIONS.	20
IDENTIFYING IMPORTING, EXPORTING, LEADING AND LAGGING.....	21
LINE TO LINE AND LINE TO NEUTRAL VOLTAGES:	21
INSTRUMENT MODBUS REGISTERS.....	21
INPUT REGISTERS	21
HOLDING REGISTERS.....	22
HOLDING REGISTERS SPECIFIC TO THE 1560 /1580.....	22
HOLDING REGISTER SPECIFIC TO THE SPR	25
APPENDIX A: INSTRUMENT INPUT REGISTERS	29
APPENDIX B: INSTRUMENT HOLDING REGISTERS	34

What is RS485?

RS485 or EIA (Electronic Industries Association) RS485 is a balanced line, half-duplex transmission system allowing transmission distances of up to 1.2 km. The following table summarises the RS-485 Standard:

PARAMETER	
Mode of Operation	Differential
Number of Drivers and Receivers	32 Drivers, 32 Receivers
Maximum cable length (metres)	1200
Maximum data rate (baud)	10 M
Maximum common mode voltage (Volts)	12 to -7
Minimum Driver Output Levels (Loaded)	+/- 1.5
Minimum Driver Output Levels (Unloaded)	+/- 6
Drive Load (Ohms)	60 (min)
Driver Output short circuit current Limit (mA)	150 to Gnd, 250 to -7 or 12 V
Minimum receiver input Resistance (kohms)	12
Receiver sensitivity	+/- 200mv

Further information relating to RS485 may be obtained from either the EIA or the various RS485 device manufacturers:

- Texas Instruments
- Maxim Semiconductors

What is half duplex?

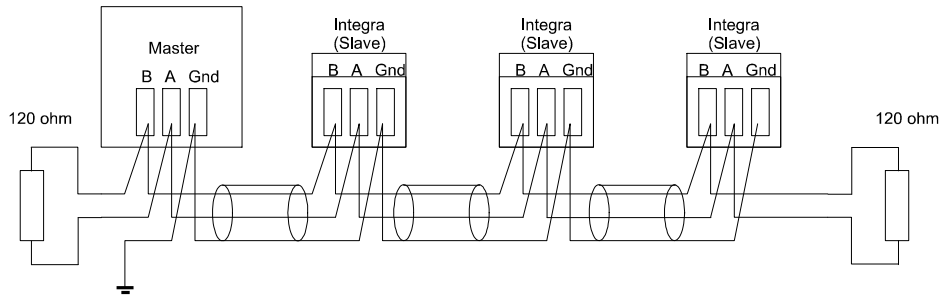
Half duplex is a system in which one or more transmitters (talkers) can communicate with one or more receivers (listeners) with only one transmitter being active at any one time. For example, a "conversation" is started by asking a question, the person who has asked the question will then listen until he gets an answer or until he decides that the individual who was asked the question is not going to reply.

In a 485 network the "master" will start the "conversation" with a "Query" addressed to a specific "slave", the "master" will then listen for the "slave's" response. If the "slave" does not respond within a pre-defined period, (set by control software in the "master"), the "master" will abandon the "conversation".

Connecting the Instruments

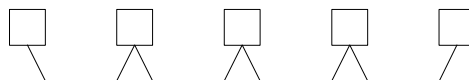
Screened twisted pair cable should be used. All "A" connections should be connected together using one conductor of the twisted pair cable, all "B" connections should be connected together using the other conductor in the pair. The cable screen should be connected to the "Gnd" terminal.

A Belden 9841 (Single pair) or 9842 (Two pair) cable with a characteristic impedance of 120 ohms is recommended, the cable should be terminated at each end with a 120 ohm, quarter watt (or greater) resistor.

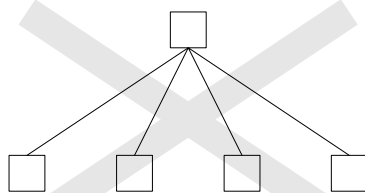


There must be no more than two wires connected to each terminal, this ensures that a "Daisy Chain or "straight line" configuration is used. A "Star" or a network with "Stubs (Tees)" is not recommended as reflections within the cable may result in data corruption.

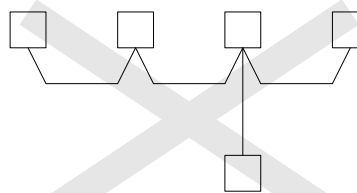
Daisy Chained Connection (Correct)



Star Connection (WRONG)

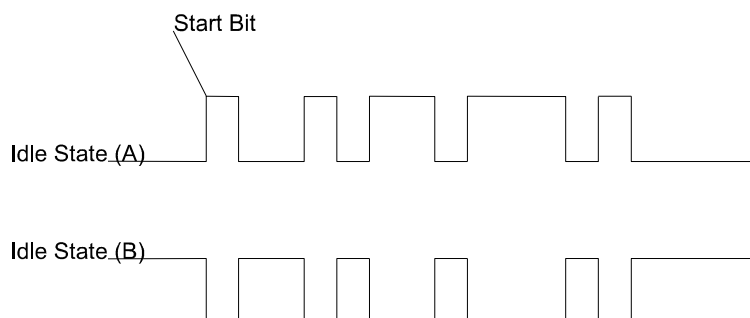


Stub Connection (WRONG)



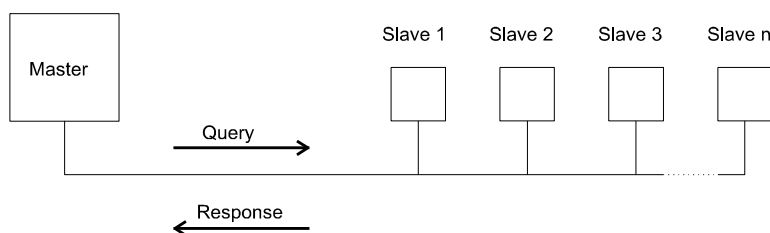
Which is A and Which is B?

The A and B connections to the SPR and Integra Products can be identified by the signals present on them whilst there is activity on the RS485 bus:



MODBUS Messages

Communication on a MODBUS Network is initiated (started) by a “Master” with a “query” to a “Slave”. The “Slave” which is constantly monitoring the network for “Queries” will recognise only the “Queries” addressed to it and will respond either by performing an action (setting a value for example) or by returning a “response”. Only the Master can initiate a query.



In the MODBUS protocol the master can address individual slaves, or, using a special “Broadcast” address, can initiate a broadcast message to all slaves. The SPR and Integra products do not support the broadcast address.

MODBUS Message Format

The MODBUS protocol defines the format for the master’s query and the slave’s response.

The query contains the device (or broadcast) address, a function code defining the requested action, any data to be sent, and an error-checking field.

The response contains fields confirming the action taken, any data to be returned, and an error-checking field. If an error occurred in receipt of the message, or if the slave is unable to perform the requested action, the slave will construct an error message and send it as its response.

Query

The example illustrates a request for a single 16-bit Modbus Register.

Slave Address	Function Code	Start Address (Hi)	Start Address (Lo)	Number of Points (Hi)	Number of Points (Lo)	Error Check (Lo)	Error Check (Hi)
---------------	---------------	--------------------	--------------------	-----------------------	-----------------------	------------------	------------------

Slave Address: 8-bit value representing the slave being addressed (1 to 247), 0 is reserved for the broadcast address. The SPR and Integra products do not support the broadcast address.

Function Code: 8-bit value telling the addressed slave what action is to be performed. (3, 4, or 16 are valid for Integra)

Start Address (Hi): The top (most significant) eight bits of a 16-bit number specifying the start address of the data being requested.

Start Address (Lo): The bottom (least significant) eight bits of a 16-bit number specifying the start address of the data being requested.

Number of Points (Hi): The top (most significant) eight bits of a 16-bit number specifying the number of registers being requested.

Number of Points (Lo): The bottom (least significant) eight bits of a 16-bit number specifying the number of registers being requested.

Error Check (Lo): The bottom (least significant) eight bits of a 16-bit number representing the error check value.

Error Check (Hi): The top (most significant) eight bits of a 16-bit number representing the error check value.

Response

The example illustrates the normal response to a request for a single 16-bit Register.

Slave Address	Function Code	Byte Count	Data (Hi)	Data (Lo)	Error Check (Lo)	Error Check (Hi)
---------------	---------------	------------	-----------	-----------	------------------	------------------

Slave Address: 8-bit value representing the address of slave, which has just responded.

Function Code: 8-bit value which, when a copy of the function code in the query, indicates that the slave recognised the query and has responded. (See also Exception Response).

Byte Count: 8-bit value indicating the number of data bytes contained within this response

Data (Hi): The top (most significant) eight bits of a 16-bit number representing the register(s) requested in the query.

Data (Lo): The bottom (least significant) eight bits of a 16-bit number representing the register(s) requested in the query.

Error Check (Lo): The bottom (least significant) eight bits of a 16-bit number representing the error check value.

Error Check (Hi): The top (most significant) eight bits of a 16-bit number representing the error check value.

Exception Response

If an error is detected in the content of the query (excluding parity errors and Error Check mismatch), the function code will be modified to indicate that the response is an error response (called an exception response), and the data bytes will contain a code that describes the error. The exception response is identified by the function code being a copy of the query function code but with the most-significant bit set to logic '1'.

Slave Address	Function Code'	Error Code	Error Check (Lo)	Error Check (Hi)
---------------	----------------	------------	------------------	------------------

Slave Address: 8-bit value representing the address of slave, which has just responded.

Function Code:	8 bit value which is the function code in the query OR'ed with Hex (80), indicating the slave either does not recognise the query or could not carry out the action requested.
Error Code:	8-bit value indicating the nature of the exception detected. (See "Exception Codes" in the section "Product Information for a list of SPR and Integra supported codes).
Error Check (Lo):	The bottom (least significant) eight bits of a 16-bit number representing the error check value.
Error Check (Hi):	The top (most significant) eight bits of a 16-bit number representing the error check value.

The Serial Transmission Modes

There are two MODBUS serial transmission modes, ASCII and RTU.

ASCII Mode

In ASCII (American Standard Code for Information Interchange) mode, each 8-bit byte in a message is sent as two ASCII characters. The main advantages of this mode are that it allows time intervals of up to one second to occur between characters without causing a timeout error and that messages may be monitored more easily on a simple ASCII terminal.

In ASCII mode messages begin with the ':' character (Hex (3A)) and end with a Carriage Return – Line Feed pair (Hex (0D) and Hex (0A)).

The format for each byte in ASCII mode is:

Coding System:	Hexadecimal, ASCII characters 0-9, A-F One hexadecimal character contained in each ASCII character of the message, each 8-bit byte to be transmitted requires two characters.
Bits per Byte:	1 start bit 7 data bits, least significant bit sent first 1 parity bit for even/odd parity; no parity bit for no parity, 1 stop bit if parity is used; 2 stop bits if no parity
Error Check Field:	Longitudinal Redundancy Check (LRC)

The SPR and Integra products do not support the ASCII mode.

RTU Mode

In RTU (Remote Terminal Unit) mode, each 8-bit byte in a message contains two 4-bit hexadecimal characters. The main advantage of this mode is that its greater character density allows better data throughput than ASCII for the same baud rate, however each message must be transmitted in a continuous stream.

The format for each byte in RTU mode is:

Coding System:	8-bit binary, hexadecimal 0-9, A-F Two hexadecimal characters contained in each 8-bit field of the message
Bits per Byte:	1 start bit, 8 data bits, least significant bit sent first 1 parity bit for even/odd parity; no parity bit for no parity 1 stop bit if parity is used; 2 stop bits if no parity
Error Check Field:	Cyclical Redundancy Check (CRC)

The SPR and Integra products support MODBUS RTU mode. Integra additionally supports a common alternative format with 1 stop bit and no parity.

MODBUS Message Timing (RTU Mode)

A MODBUS message has defined beginning and ending points. The receiving devices recognise the start of the message, read the "Slave Address" to determine if they are being addressed and know when the message is completed so that they can use the Error Check bytes to confirm the integrity of the query. Partial messages can be detected and discarded:

In RTU mode, messages start with a silent interval of at least 3.5 character times.

The first field then transmitted is the device address.

The allowable characters transmitted for all fields are hexadecimal 0-9, A-F. Devices monitor the network bus continuously, including during the 'silent' intervals. When the first field (the address field) is received, each device decodes it to find out if it is the addressed device. If the device determines that it is the one being addressed it decodes the whole message and acts accordingly, if it is not being addressed it continues monitoring for the next message.

Following the last transmitted character, a silent interval of at least 3.5 character times marks the end of the message. A new message can begin after this interval.

In the Integra 1000 and 2000, a silent interval of 60msec minimum is required in order to guarantee successful reception of the next request.

The entire message frame must be transmitted as a continuous stream. If a silent interval of more than 1.5 character times occurs before completion of the frame, the receiving device flushes the incomplete message and assumes that the next byte will be the address field of a new message.

Similarly, if a new message begins earlier than 3.5 character times following a previous message, the receiving device will consider it a continuation of the previous message. This will result in an error, as the value in the final CRC field will not be valid for the combined messages.

How Characters are Transmitted Serially

When messages are transmitted on standard MODBUS serial networks each character or byte is sent in this order (left to right):

Least Significant Bit (LSB)

Most Significant Bit (MSB)

With Parity Checking (11 bit characters):

Start	1	2	3	4	5	6	7	8	Parity	Stop
-------	---	---	---	---	---	---	---	---	--------	------

Without Parity Checking, 2 Stop Bits (11 bit characters):

Start	1	2	3	4	5	6	7	8	Stop	Stop
-------	---	---	---	---	---	---	---	---	------	------

The Integra products additionally support No parity, One stop bit.

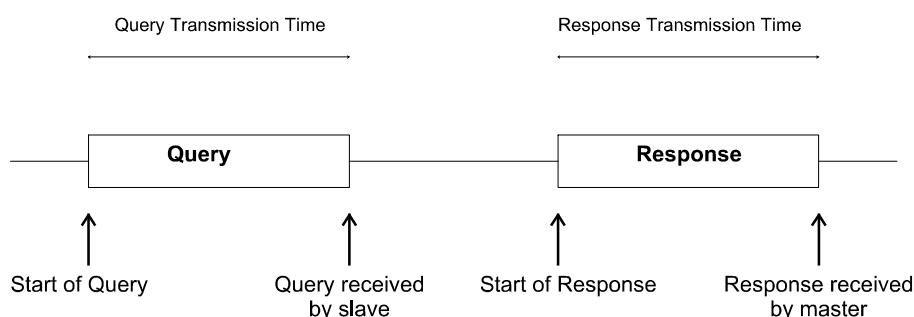
Without Parity Checking, 1 Stop Bit (10 bit characters):

Start	1	2	3	4	5	6	7	8	Stop
-------	---	---	---	---	---	---	---	---	------

Error Checking Methods

Standard MODBUS serial networks use two error checking processes, the error check bytes mentioned above check message integrity whilst Parity checking (even or odd) can be applied to each character in the message.

The master is configured by the user to wait for a predetermined timeout interval. The master will wait for this period of time before deciding that the slave is not going to respond and that the transaction should be aborted. Care must be taken when determining the timeout period from both the master and the slaves' specifications. The slave may define the 'response time' as being the period from the receipt of the last bit of the query to the transmission of the first bit of the response. The master may define the 'response time' as period between transmitting the first bit of the query to the receipt of the last bit of the response. It can be seen that message transmission time, which is a function of the baud rate, must be included in the timeout calculation.



Parity Checking

If parity checking is enabled - either Even or Odd Parity is specified - the quantity of "1's" will be counted in the data portion of each of the eight bits in the character. The parity bit will then be set to a 0 or 1 to result in an Even or Odd total of "1's".

Note that parity checking can only detect an error if an odd number of bits are picked up or dropped in a character frame during transmission, if for example two 1's are corrupted to 0's the parity check will not find the error.

If No Parity checking is specified, no parity bit is transmitted and no parity check can be made. An additional stop bit is transmitted to fill out the character frame when 2 stop bits are selected.

If No Parity checking is specified and one stop bit is selected the character is effectively shortened by one bit.

CRC Checking

The error check bytes of the MODBUS messages contain a Cyclical Redundancy Check (CRC) value that is used to check the content of the entire message. The error check bytes must always be present to comply with the MODBUS protocol; there is no option to disable it.

The error check bytes represent a 16-bit binary value, calculated by the transmitting device. The receiving device must recalculate the CRC during receipt of the message and compare the calculated value to the value received in the error check bytes. If the two values are not equal, the message should be discarded.

The error check calculation is started by first pre-loading a 16-bit register to all 1's (i.e. Hex (FFFF)) each successive 8-bit byte of the message is applied to the current contents of the register. Note: only the eight bits of data in each character are used for generating the CRC, start bits, stop bits and the parity bit, if one is used, are not included in the error check bytes.

During generation of the error check bytes, each 8-bit character is exclusive OR'ed with the register contents. The result is shifted in the direction of the least significant bit (LSB), with a zero filled into the most significant bit (MSB) position. The LSB prior to the shift is extracted and examined. If the LSB was a 1, the register is then exclusive OR'ed with a pre-set, fixed value. If the LSB was a 0, no exclusive OR takes place.

This process is repeated until eight shifts have been performed. After the last (eighth) shift, the next 8-bit byte is exclusive OR'ed with the register's current value, and the process repeated. The final contents of the register, after all the bytes of the message have been applied, is the error check value.

In the following pseudo code "ErrorWord" is a 16-bit value representing the error check values.

```
BEGIN
  ErrorWord = Hex (FFFF)
  FOR Each byte in message
    ErrorWord = ErrorWord XOR byte in message
    FOR Each bit in byte
      LSB = ErrorWord AND Hex (0001)
      IF LSB = 1 THEN ErrorWord = ErrorWord - 1
      ErrorWord = ErrorWord / 2
    IF LSB = 1 THEN ErrorWord = ErrorWord XOR Hex (A001)
  NEXT bit in byte
NEXT Byte in message
END
```

Function Codes

The function code part of a MODBUS message defines the action to be taken by the slave. The SPR and Integra products support the following function codes:

Code	MODBUS name	Description
03	Read Holding Registers	Read the contents of read/write location (4X references)
04	Read Input Registers	Read the contents of read only location (3X references)
08	Diagnostics	Only sub-function zero is supported. This returns the data element of the query unchanged.
16	Pre-set Multiple Registers	Set the contents of read/write location (4X references)

Product Specific Considerations.

The MODBUS protocol defines “Registers” for the variables; a register is 16 bits long. As a 16-bit number would prove restrictive for the power parameters measured (maximum of 65535) there are a number of approaches, which have been adopted to overcome this restriction. The Crompton products use two consecutive registers to represent a floating-point number, effectively expanding the range to $\pm 1 \times 10^{37}$.

The values produced by the Crompton products can be used directly without any requirement to “scale” the values, for example, the units for the voltage parameters are volts, the units for the power parameters are watts etc.

What is a floating point Number?

A floating-point number is a number with two parts, a mantissa and an exponent and is written in the form 1.234×10^5 . The mantissa (1.234 in this example) must have the decimal point moved to the right with the number of places determined by the exponent (5 places in this example) i.e. $1.234 \times 10^5 = 123400$. If the exponent is negative the decimal point is moved to the left.

What is an IEEE 754 format floating-point number?

An IEEE 754 floating point number is the binary equivalent of the floating-point number shown above. The major difference being that the most significant bit of the mantissa is always arranged to be 1 and is thus not included in the representation of the number. The process by which the most significant bit is arranged to be 1 is called normalisation; the mantissa is thus referred to as a “normal mantissa”. During normalisation the bits in the mantissa are shifted to the left whilst the exponent is decremented until the most significant bit is one. In the special case where the number is zero both mantissa and exponent are zero.

The bits in an IEEE 754 format have the following significance:

Data Hi Word, Hi Byte	Data Hi Word, Lo Byte	Data Lo Word, Hi Byte	Data Lo Word, Lo Byte
EEEE EEEE	EMMM MMMM	MMMM MMMM	MMMM MMMM

Where:

- S represents the sign bit where 1 is negative and 0 is positive
- E is the two's complement exponent with an offset of 127 i.e. an exponent of zero is represented by 127, an exponent of 1 by 128 etc.
- M is the 23-bit normal mantissa. The highest bit is always 1 and, therefore, is not stored.

Using the above format the floating point number 240.5 is represented as <43><70><80><00>:

Data Hi Word, Hi Byte	Data Hi Word, Lo Byte	Data Lo Word, Hi Byte	Data Lo Word, Lo Byte
43	70	80	00

The following example demonstrates how to convert floating-point numbers from their hexadecimal storage equivalents. For this example, we will use the value for 240.5 shown above

Note that the floating-point storage representation is not an intuitive format. To convert this value to a floating-point number, the bits should be separated as specified in the floating-point number storage format table shown above. For example:

Data Hi Word, Hi Byte	Data Hi Word, Lo Byte	Data Lo Word, Hi Byte	Data Lo Word, Lo Byte
0100 0011	0111 0000	1000 0000	0000 0000

From this you can determine the following information.

- The sign bit is 0, indicating a positive number.
- The exponent value is 10000110 binary or 134 decimal. Subtracting 127 from 134 leaves 7, which is the actual exponent.
- The mantissa appears as the binary number 11100001000000000000000

There is an implied decimal point at the left of the mantissa that is always preceded by a 1. This digit is not stored in the hexadecimal representation of the floating-point number. Adding 1 and the decimal point to the beginning of the mantissa gives the following:

1.111000010000000000000000

Now, we adjust the mantissa for the exponent. A negative exponent moves the decimal point to the left. A positive exponent moves the decimal point to the right. Because the exponent is 7, the mantissa is adjusted as follows:

11110000.1000000000000000

Finally, we have a binary floating-point number. Binary digits that are to the left of the decimal point represent the power of two corresponding to their position. For example, 11110000 represents $(1 \times 2^7) + (1 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (0 \times 2^0) = 240$.

Binary digits that are to the right of the decimal point also represent a power of 2 corresponding to their position. As the digits are to the right of the decimal point the powers are negative. For example: 100 represents $(1 \times 2^{-1}) + (0 \times 2^{-2}) + (0 \times 2^{-3}) + \dots$ which equals 0.5.

Adding these two numbers together and making reference to the sign bit produces the number +240.5.

For each floating point value requested two MODBUS registers or points (four bytes) must be requested. The received order and significance of these four bytes for the SPR and Integra products¹ is shown below:

Data Hi Word, Hi Byte	Data Hi Word, Lo Byte	Data Lo Word, Hi Byte	Data Lo Word, Lo Byte
--------------------------	--------------------------	--------------------------	--------------------------

MODBUS Commands supported

All Crompton Integra Instruments support the “Read Input Register” (3X registers), the “Read Holding Register” (4X registers) and the “Pre-set Multiple Registers” (write 4X registers) commands of the MODBUS RTU protocol. All values stored and returned are in floating point format to IEEE 754 with the most significant byte first².

Read Input Registers

MODBUS code 04 reads the contents of the 3X registers.

Example

The following query will request ‘Volts 1’ from an instrument with node address 1:

Field Name	Example (Hex)
Slave Address	01
Function	04
Starting Address High	00
Starting Address Low	00
Number of Points High	00
Number of Points Low	02
Error Check Low	71
Error Check High	CB

Note: Data must be requested in register pairs i.e. the “Number of Points” must be even to request a floating point variable. If the “Number of points” is odd or the “Starting Address” falls in the middle of a floating point variable the product will return an error message.

¹ The Integra 1500 Series supports the ability to reverse the word order. In reversed mode the words are transmitted and received Data Lo Word followed by Data Hi Word.

² Except where the instrument is an Integra 1500 Series configured for reversed mode.

The following response returns the contents of Volts 1 as 230.2

Field Name	Example (Hex)
Slave Address	01
Function	04
Byte Count	04
Data, High Word, High Byte	43
Data, High Word, Low Byte	66
Data, Low Word, High Byte	33
Data, Low Word, Low Byte	34
Error Check Low	1B
Error Check High	38

Read Holding Registers

MODBUS code 03 reads the contents of the 4X registers.

Example

The following query will request the prevailing 'Demand Time':

Field Name	Example (Hex)
Slave Address	01
Function	03
Starting Address High	00
Starting Address Low	00
Number of Points High	00
Number of Points Low	02
Error Check Low	C4
Error Check High	0B

Note: Data must be requested in register pairs i.e. the "Number of Points" must be even to request a floating point variable. If the "Number of points" is odd or the "Starting Address" falls in the middle of a floating point variable the Integra products will return an error message.

The following response returns the contents of Demand Time as 1

Field Name	Example (Hex)
Slave Address	01
Function	03
Byte Count	04
Data, High Word, High Byte	3F
Data, High Word, Low Byte	80
Data, Low Word, High Byte	00
Data, Low Word, Low Byte	00
Error Check Low	F7
Error Check High	CF

Write Holding Registers

MODBUS code 16 decimal (10 Hex) writes the contents of the 4X registers.

Example

The following query will set the Demand Time to zero, which effectively resets the Demand Period:

Field Name	Example (Hex)
Slave Address	01
Function	10
Starting Address High	00
Starting Address Low	00
Number of Registers High	00
Number of Registers Low	02
Byte Count	04
Data, High Word, High Byte	00
Data, High Word, Low Byte	00
Data, Low Word, High Byte	00
Data, Low Word, Low Byte	00
Error Check Low	3D
Error Check High	23

Note: Data must be written in register pairs i.e. the “Number of Points” must be even to set a floating point variable. If the “Number of points” is odd or the “Starting Address” falls in the middle of a floating point variable the Integra products will return an error message.

The following response indicates that the write has been successful.

Field Name	Example (Hex)
Slave Address	01
Function	10
Starting Address High	00
Starting Address Low	00
Number of Registers High	00
Number of Registers Low	02
Error Check Low	41
Error Check High	C8

Diagnostics

MODBUS code 08 provides a number of diagnostic sub-functions. Only the "Return Query Data" sub-function (sub-function 0) is supported on the Integra 1540, 1560, 1580.

Example

The following query will send a diagnostic "return query data" query with the data elements set to Hex(AA) and Hex(55) and will expect these to be returned in the response:

Field Name	Example (Hex)
Slave Address	01
Function	08
Sub-Function High	00
Sub-Function Low	00
Data Byte 1	AA
Data Byte 2	55
Error Check Low	5E
Error Check High	94

Note: Data must be written in register pairs i.e. the "Number of Points" must be even. If the "Number of points" is odd or the "Starting Address" falls in the middle of a floating point variable product will return an error message.

The following response indicates the correct reply to the query.

Field Name	Example (Hex)
Slave Address	01
Function	08
Sub-Function High	00
Sub-Function Low	00
Data Byte 1	AA
Data Byte 2	55
Error Check Low	5E
Error Check High	94

When things go wrong

- Start with a simple network, one master and one slave, with SPR and Integra products this is easily achieved as the network can be left intact whilst individual instruments are disconnected by removing the RS485 connector from the rear of the instrument.
- Check that all the “A’s” are connected together, check that all the “B’s” are connected together, check that all the “Gnd’s” are connected together.
- If using SpecView or PCView (PC software) with a RS232 to RS485 converter confirm that the data “transmitted” onto the RS485 is not echoed back to the PC on the RS232 lines. (This facility is sometimes a link option within the converter). Many PC based packages seem to be “confused” when they receive an echo of the message they are transmitting.
- Confirm that the Address of the instrument is the same as the “master” is expecting.
- If the “network” operates with one instrument but not more than one check that each instrument has a unique address.
- Each request for data must be restricted to 20 parameters (40 in the case of the 1500 Series) or less. Violating this requirement will impact the performance of the instrument and may result in a response time in excess of the specification.
- Check that the MODBUS mode (RTU or ASCII) and serial parameters (baud rate, number of data bits, number of stop bits and parity) are the same for all devices on the network.
- Check that the “master” is requesting floating-point variables (pairs of registers placed on floating point boundaries) and is not “splitting” floating point variables.
- Check that the floating-point byte order expected by the “master” is the same as that used by the SPR and Integra products. (PCView and Citect packages can use a number of formats including that supported by SPR and Integra).
- If possible obtain a second RS232 to RS485 converter and connect it between the RS485 bus and an additional PC equipped with a software package, which can display the data on the bus. Check for the existence of valid requests.

Product Information

The following section outline the general information about the use of Modbus with the products covered by this guide. It also outlines the information available over the Modbus interface of the products.

Appendix A and B, provides a cross-reference as to which features are available over the Modbus interface for individual models.

On the instrument, each electrical parameter measured or calculated and each configuration parameter is given a number that maps onto a Modbus address. To convert a Crompton Parameter number into a Modbus address, subtract one from the parameter number and double it.

Exception Codes

Whenever an SPR or Integra product receives a MODBUS message with valid parity and error check but which contains some other error (e.g. a request to set a register to an illegal value or a request for part of a floating point variable), an Exception code will be generated. (The message format is shown in the message Formats section)

Exceptions are indicated by a value in the function code field of the response greater than Hex (80), obtained by OR'ing the original function code in the query with Hex (80). For example, if a function code of Hex (84) and exception code 2 were present in an exception response this indicates that a function 4 query (Read Holding Registers) has resulted in an illegal data address error.

The error codes and the corresponding types of error returned by the models covered in this guide are given in the following table:

Exception Code	MODBUS name	Description	Reported by
01	Illegal Function	The function code is not supported by the product	1000, 2000, 1540, 1560, 1580
02	Illegal Data Address	Attempt to access an invalid address or an attempt to read or write part of a floating point value	1000, 2000, 1540, 1560, 1580
03	Illegal Data Value	Attempt to set a floating point variable to an invalid value	1000, 2000, 1540, 1560, 1580
05	Slave Device Failure	An error occurred when the instrument attempted to store an update to it's configuration	1540, 1560, 1580

Start Up Considerations.

The products covered by this guide do not require any special start up messages; the "master" simply requests the data it requires.

On power up the Integra 1560 and 1580 have a Display Auto Detect facility. For about five seconds the instrument will monitor but not reply to Modbus queries. If it detects a Crompton display unit is powered up and attached, the communication settings on that port will be fixed to 9600 baud, 1 start, 2 stop and no parity until the instrument is powered down.

If the display auto detect period expires without a display unit being detected, the Integra 1560 and 1580 will set-up the port(s) to use the communication parameters previously configured by the user.

Identifying Importing, Exporting, Leading and Lagging.

The products generate signed values for Watts and VAR that may be used to indicate the direction of power flow (Import or Export) and whether the load is inductive (current lagging voltage) or capacitive (current leading voltage). The following table provides a key to determining these characteristics used with the respective Phase Angle and Power Factor displays.

Sign of Watts	Sign of Vacs	Direction of Power Flow	Current Lead/Lag
+ve	+ve	Import	Lagging
+ve	-ve	Import	Leading
-ve	+ve	Export	Leading
-ve	-ve	Export	Lagging

Line to Line and Line to Neutral Voltages:

On a 3 phase 4 wire system parameters 1-3 will contain the measured parameters, that is the Line to Neutral voltages, the values of parameters 22, 23, 55-60, 67-70 will also refer to line to neutral values. The values contained in parameters 101-112 will be calculated from the line to neutral voltages and represent the line to line values.

On a 3 phase 3 wire system parameters 1-3 will contain the measured parameters, that is the Line to Line voltages, the values of parameters 22, 23, 55-60, 67-70 will also refer to line to line values.

Instrument Modbus Registers

The Integra 1000 provides a subset of the Integra 2000 Input Registers, note that the addresses of the parameters for the Integra 1000 have been kept the same as those for the Integra 2000. This will allow an Integra 2000 to replace an Integra 1000 without the need to change the addressing.

The Integra 1540 input and holding registers extend the set of registers used on the Integra 2000. This will allow an Integra 1540 to replace an Integra 1000 or an Integra 2000 without the need to change the addressing.

The Integra 1560/80 input and holding registers extend the set of registers used on the Integra 1540. This will allow an Integra 1540/1580 with a DIS-1540-xxx display unit to replace an Integra 1540, Integra 1000 or an Integra 2000 without the need to change the addressing.

The SPR input and holding registers are a sub-set of registers used on the Integra family with some additions to make some of the more specialised features of the SPR available over the Modbus interface.

Input Registers

Input registers are used to indicate the prevailing values of measured and derived (calculated) electrical quantities and parameters. The table in 'Appendix A' lists the information available via input registers for each of the instruments covered by this document.

All input registers not listed in the table should be considered as reserved for Crompton Instruments use and the installation should not rely on any value(s) read from an undocumented register.

Holding Registers

Holding registers are generally used to store and display instrument configuration settings. All Holding registers not listed in the table in Appendix B should be considered as reserved for Crompton Instruments use and no attempt should be made to modify their values.

Each of the parameters available via the holding registers are briefly described below:

Demand Time (parameter 1) is used to reset the demand period. A value of zero must be written to this register to accomplish this. Writing any other value will cause an error to be returned.

Demand Period (Parameter 2) is used to view or set the time over which the demand value should be calculated. Valid values are 8, 15, 20 and 30 minutes.

Demand Interval (parameter 3) provides frequency at which demand figures are accumulated. This is a read only register and is usually fixed at 1-minute intervals. So for a demand period of 8 minutes for calculation of the demands will be based on 8 accumulated readings, So for a demand period of 15 minutes for calculation of the demands will be based on 15 accumulated readings, etc. An error will occur if an attempt is made to write an unsupported value into this parameter location.

System Type (parameter 6) will display 1 for 'Single phase', 2 for '3 Phase 3 Wire' or 3 for '3 Phase 4 Wire'.

Reset Energy Variables (parameter 8) is used to clear down the energy readings. A value of zero must be written to this register to accomplish this. Writing any other value will cause an error to be returned.

System Power (Parameter 19), the maximum system power based on the values of system type, system volts and system current.

Register Order (Parameter 21), the instrument can receive or send floating-point numbers in normal or reversed register order. In normal mode, the two registers that make up a floating-point number are send lower register first. In reversed register mode, the two registers that make up a floating-point number are send higher register first. To set the mode, write the value '2141.0' into this register - the instrument will detect the order used to send this value and set that order for all Modbus transactions involving floating point numbers.

Secondary Volts (Parameter 143) indicates the voltage on the VT secondary when the voltage on the Primary is equal to the value of System Volts (Parameter 4). The value of this register can be set to between the minimum and maximum instrument input voltage.

Holding Registers Specific to the 1560 /1580

Analogue Hardware Maximum (Parameter148), the maximum analogue output level (expressed in mA) supported by the installed analogue output module.

Analogue Hardware Minimum (Parameter149) the minimum analogue output level (expressed in mA) supported by the installed analogue output module.

Analogue Output 1 Output Parameter (Parameter150), the number of the input parameter that is to be output on analogue output 1. A value of zero signifies the analogue output is unused.

Analogue Output 1 Parameter Maximum (Parameter151), the maximum value that the selected input parameter can reach.

Analogue Output 1 Parameter Minimum (Parameter152), the minimum value that the selected input parameter can reach.

Analogue Output 1 Reading Top (Parameter153), the upper limit of the parameter value that will be output. This value can range between Parameter Minimum and Parameter Maximum.

Analogue Output 1 Reading Bottom (Parameter154), the lower limit of the parameter value that will be output. This value can range between Parameter Minimum and Parameter Maximum.

Analogue Output 1 Output Top (Parameter155), the analogue output level that will be achieved when the parameter reading reaches Reading Top. The value of output top will be greater than Output Bottom and Less than or equal to Analogue Hardware Maximum.

Analogue Output 1 Output Bottom (Parameter156), the analogue output level that will be achieved when the parameter reading reaches Reading Bottom. The value of Output Bottom will be greater than or equal to Analogue Hardware Minimum and Less than Output Top.

Analogue Output 2 Output Parameter (Parameter160), the number of the input parameter that is to be output on analogue output 2. A value of zero signifies the analogue output is unused.

Analogue Output 2 Parameter Maximum (Parameter161), the maximum value that the selected input parameter can reach.

Analogue Output 2 Parameter Minimum (Parameter162), the minimum value that the selected input parameter can reach.

Analogue Output 2 Reading Top (Parameter163), the upper limit of the parameter value that will be output. This value can range between Parameter Minimum and Parameter Maximum.

Analogue Output 2 Reading Bottom (Parameter164), the lower limit of the parameter value that will be output. This value can range between Parameter Minimum and Parameter Maximum.

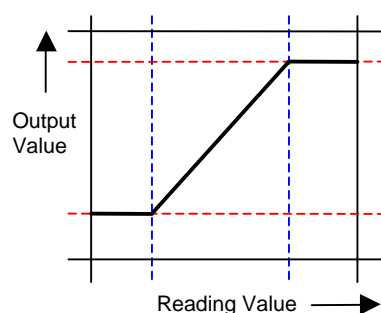
Analogue Output 2 Output Top (Parameter165), the analogue output level that will be achieved when the parameter reading reaches Reading Top. The value of output top will be greater than Output Bottom and Less than or equal to Analogue Hardware Maximum.

Analogue Output 2 Output Bottom (Parameter166), the analogue output level that will be achieved when the parameter reading reaches Reading Bottom. The value of Output Bottom will be greater than or equal to Analogue Hardware Minimum and Less than Output Top.

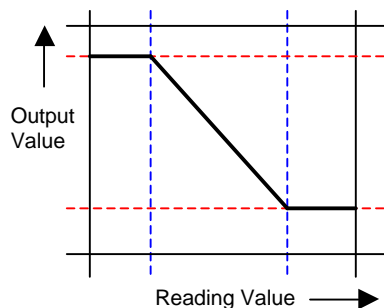
Analogue Output Operation

Depending on the settings for output top, output bottom, reading top and reading bottom, the analogue output can be configured to operate in a number of modes.

Normal Mode – As the reading value increases between the user set values for reading top and reading bottom the output also increases in magnitude between output top and bottom. As the reading value decreases between reading top and reading bottom, the output also decreases.

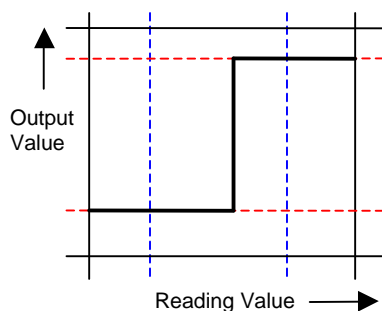


Reverse Mode – as the reading value increases between the user set values for reading top and reading bottom the output decreases in magnitude between output top and bottom. As the reading value decreases between reading top and reading bottom, the output increases between output top and bottom.



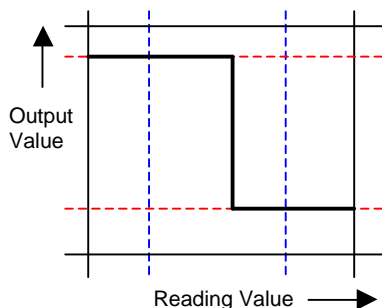
Normal Threshold Mode – The values for reading top and bottom are set the same, this is called the threshold value. The value set for output bottom is more negative than the value set for output top.

If the reading value is between reading min and the threshold value, the output value will be that of output bottom (the more negative of the output limits). If the reading increases beyond the threshold then the output value switches to output top (the more positive of the output limits).

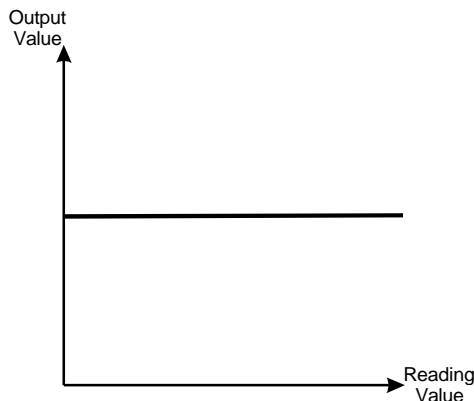


Reverse Threshold Mode – The values for reading top and bottom are set to the magnitude (the threshold value) and output bottom is set more positive than output top.

If the reading value is between reading min and the threshold value, the output value will be that of output bottom (the more positive of the two limits). If the reading increases beyond the threshold then the output value switches to output top (the more negative of the two limits).



Constant Current Mode - the values for output top and bottom are set to the same magnitude (the desired output level). The setting of reading top and bottom has no effect.



Regardless of the prevailing reading the analogue output will remain at the same level

Setting the Analogue Operating Mode

The following table illustrates how the various operating modes can be configured by setting the relationship between the values of output top & output bottom and reading top & reading bottom.

Output Limits	Reading Limits	Mode
Bottom < Top	Bottom < Top	Normal
Bottom < Top	Top < Bottom	Reverse
Top < Bottom	Bottom < Top	Reverse
Top < Bottom	Top < Bottom	Normal
Bottom < Top	Bottom = Top	Normal Threshold
Top < Bottom	Bottom = Top	Reverse Threshold

In each case, the values of the output limits must be between Output Min. and Output Max. and the values of reading limits must be between Reading Min. and Reading Max.

Holding Register specific to the SPR

Relay Number (Parameter 30) The relay number associated with the relay configuration parameters that follow. Writing a relay number in the range 1 to 16 to this register updates the relay configuration parameters to reflect the settings for that relay.

Trip Function (Parameter 31) The trip function the relay given by parameter 30 is currently performing. This is an enumerated type that determines the trip function.

Number	Function
0	Undefined
1	Over Voltage (ANSI 59)
2	Under Voltage (ANSI 27)
3	Over Frequency (ANSI 81O)
4	Under Frequency (ANSI 81U)
5	Reverse Power (ANSI 32R)
6	Phase Unbalance Voltage (ANSI 47)
7	Time Over current with Voltage Restraint (ANSI 51V)
8	Time Over current (ANSI 51)
9	Ground Fault (ANSI 51G)
10	Synchronising Check (ANSI 25)
11	Synch Check with Dead Bus (ANSI 25)
12	Phase Sequence (ANSI 47)
13	Logical AND
14	Logical OR
15	Logical NAND
16	Logical NOR
17	Logical XOR
18	Logical Vote
19	Logical Disc
20	Alarm
21	Unacknowledged Alarm

Trip Mode (Parameter 32) This is a bit set, returned as a floating-point number it has the following values, which may be added for composites functions. The setting is binary weighted, e.g. A register setting of 3 indicates inverted and alarm functions are enabled ($1 + 2 = 3$).

Function	Value
Inverted	1
Alarm	2
Latched	4
Logged	8

Trip Status (Parameter 33) This is a bit set, returned as a floating point number and has the functions shown below.

Name	Description	Value	Access
CurrentState	The current state of the trip condition	1	R
Active	The current state of the latch (if set)	2	R
Acked	Acknowledged alarm	4	R/W
InAlarm	The trip is in alarm	8	R
ClearLatch	Set to clear the latch	16	W

Trip Parameter 1-6 (Parameter 34 to 39) These are the other parameters for the trip. The function of these is dependent on the trip type, and relates directly to the menu layout through the front panel.

Trip Name	P 1	P 2	P 3	P 4	P 5	P 6
Over Volts 59	Trip Point	Differential	Time Delay			
Under Volts 27	Trip Point	Differential	Time Delay			
Over Freq. 81O	Trip Point	Differential	Time Delay			
Under Freq. 81U	Trip Point	Differential	Time Delay			
Reverse Pwr 32R	Trip Point	Differential	Time Delay			
UnBal Volts 47	Trip Point	Differential	Time Delay			
I/T+V.R. 51V	Trip Point	Nominal Volts	Time Curve	Time Dial		
I/T 51	Trip Point	Time Curve	Time Dial			
Ground Fault 51G	Trip Point	Time Dial				
Sync 25	Phase Angle	Voltage Delta	Voltage Min.	Time Delay	Slip Freq.	
Sync+DB 25	Phase Angle	Voltage Delta	Voltage Min.	Deadbus	Time Delay	Slip Freq.
Phase Order 47	Time Delay					
Logical AND	Input 1	Input 2	Input 3	Delay On	Delay Off	
Logical OR	Input 1	Input 2	Input 3	Delay On	Delay Off	
Logical NAND	Input 1	Input 2	Input 3	Delay On	Delay Off	
Logical NOR	Input 1	Input 2	Input 3	Delay On	Delay Off	
Logical XOR	Input 1	Input 2	Input 3	Delay On	Delay Off	
Logical Vote	Input 1	Input 2	Input 3	Delay On	Delay Off	
Logical Disc	Input 1	Input 2	Input 3	Delay On	Delay Off	
Alarm	Time Delay					
Unack Alarm	Time Delay					

Appendix A: Instrument Input Registers

In the table the “System Configuration” is defined by the number of wires, 4 represents a 3 phase 4 wire system, 3 represents a 3 phase 3 wire system and 2 represents a single phase system. A dot (●) in the system configuration column indicates that the parameter is valid for that wiring system, a dash (-) will return the value zero.

Parameter No	Modbus Start Address		Parameter	Integra 2000 Configuration (No of Wires)			Integra 1000 Configuration (No of Wires)			Integra 1540 Configuration (No of Wires)			Integra 1560/80 Configuration (No of Wires)				SPR Configuration (No of Wires)		
	High Byte	Low Byte		4	3	2	4	3	2	4	3	2	4	3	1p 3w	1p 2w	4	3	2
1	00	00	Volts 1	●	●	●	●	●	●	●	●	-	●	●	●	●	●	●	-
2	00	02	Volts 2	●	●	-	●	●	-	●	●	-	●	●	●	-	●	●	-
3	00	04	Volts 3	●	●	-	●	●	-	●	●	-	●	●	-	-	●	●	-
4	00	06	Current 1	●	●	●	●	●	●	●	●	-	●	●	●	●	●	●	-
5	00	08	Current 2	●	●	-	●	●	-	●	●	-	●	●	●	-	●	●	-
6	00	0A	Current 3	●	●	-	●	●	-	●	●	-	●	●	-	-	●	●	-
7	00	0C	Watts 1	●	-	●	-	-	-	●	-	-	●	-	●	●	●	-	-
8	00	0E	Watts 2	●	-	-	-	-	-	●	-	-	●	-	●	-	●	-	-
9	00	10	Watts 3	●	-	-	-	-	-	●	-	-	●	-	-	-	●	-	-
10	00	12	VA 1	●	-	●	-	-	-	●	-	-	●	-	●	●	●	-	-
11	00	14	VA 2	●	-	-	-	-	-	●	-	-	●	-	●	-	●	-	-
12	00	16	VA 3	●	-	-	-	-	-	●	-	-	●	-	-	-	●	-	-
13	00	18	VAr 1	●	-	●	-	-	-	●	-	-	●	-	●	●	●	-	-
14	00	1A	VAr 2	●	-	-	-	-	-	●	-	-	●	-	●	-	●	-	-
15	00	1C	VAr 3	●	-	-	-	-	-	●	-	-	●	-	-	-	●	-	-
16	00	1E	Power Factor 1	●	-	●	-	-	-	●	-	-	●	-	●	●	●	-	-
17	00	20	Power Factor 2	●	-	-	-	-	-	●	-	-	●	-	●	-	●	-	-
18	00	22	Power Factor 3	●	-	-	-	-	-	●	-	-	●	-	-	-	●	-	-
19	00	24	Phase Angle 1	●	-	●	-	-	-	●	-	-	●	-	●	●	●	-	-
20	00	26	Phase Angle 2	●	-	-	-	-	-	●	-	-	●	-	●	-	●	-	-
21	00	28	Phase Angle 3	●	-	-	-	-	-	●	-	-	●	-	-	-	●	-	-

Parameter No	Modbus Start Address		Parameter	Integra 2000 Configuration (No of Wires)			Integra 1000 Configuration (No of Wires)			Integra 1540 Configuration (No of Wires)			Integra 1560/80 Configuration (No of Wires)				SPR Configuration (No of Wires)		
	High Byte	Low Byte		4	3	2	4	3	2	4	3	2	4	3	1p 3w	1p 2w	4	3	2
22	00	2A	Volts Ave	●	●	●	●	●	●	●	-	-	●	●	●	●	●	-	-
23	00	2C	Voltage Sum	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
24	00	2E	Current Ave	●	●	●	●	●	●	●	-	-	●	●	●	●	●	●	-
25	00	30	Current Sum	●	●	●	-	-	-	●	-	-	●	●	●	●	●	●	-
26	00	32	Watts Ave	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
27	00	34	Watts Sum	●	●	●	●	●	●	●	●	-	●	●	●	●	●	●	-
28	00	36	VA Ave	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
29	00	38	VA Sum	●	●	●	●	●	●	●	●	-	●	●	●	●	●	●	-
30	00	3A	VAr Ave	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
31	00	3C	VAr Sum	●	●	●	●	●	●	●	●	-	●	●	●	●	●	●	-
32	00	3E	Power Factor Ave	●	●	●	●	●	●	●	●	-	●	●	●	●	-	-	-
33	00	40	Power Factor Sum	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
34	00	42	Phase Angle Ave	●	●	●	-	-	-	●	●	-	●	●	●	●	-	-	-
35	00	44	Phase Angle Sum	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
36	00	46	Frequency	●	●	●	●	●	●	●	●	-	●	●	●	●	●	●	-
37	00	48	W.hr Import	●	●	●	●	●	●	●	●	-	●	●	●	●	-	-	-
38	00	4A	W.hr Export	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
39	00	4C	VAr.h Import	●	●	●	●	●	●	●	●	-	●	●	-	●	-	-	-
40	00	4E	VAr.h Export	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
41	00	50	VA.h	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
42	00	52	A.h	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
43	00	54	W Demand Import	●	●	●	●	●	●	●	●	-	●	●	●	●	-	-	-
44	00	56	W Max. Demand Import	●	●	●	●	●	●	●	●	-	●	●	●	●	-	-	-
45	00	58	W Demand Export	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
46	00	5A	W Max. Demand Export	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
47	00	5C	VAr Demand Import	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
48	00	5E	VAr Max Demand Import	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
49	00	60	VAr Demand Export	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
50	00	62	VAr Max Demand Export	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
51	00	64	VA Demand	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
52	00	66	VA Max. Demand	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-

Parameter No	Modbus Start Address		Parameter	Integra 2000 Configuration (No of Wires)			Integra 1000 Configuration (No of Wires)			Integra 1540 Configuration (No of Wires)			Integra 1560/80 Configuration (No of Wires)				SPR Configuration (No of Wires)		
	High Byte	Low Byte		4	3	2	4	3	2	4	3	2	4	3	1p 3w	1p 2w	4	3	2
53	00	68	A Demand	●	●	●	●	●	●	●	●	-	●	●	●	●	-	-	-
54	00	6A	A Max. Demand	●	●	●	●	●	●	●	●	-	●	●	●	●	-	-	-
55	00	6C	Volts 1 Max.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
56	00	6E	Volts 1 Min.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
57	00	70	Volts 2 Max.	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-
58	00	72	Volts 2 Min.	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-
59	00	74	Volts 3 Max.	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-
60	00	76	Volts 3 Min.	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-
61	00	78	Current 1 Max.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
62	00	7A	Current 1 Min.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
63	00	7C	Current 2 Max.	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-
64	00	7E	Current 2 Min.	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-
65	00	80	Current 3 Max.	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-
66	00	82	Current 3 Min.	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-
67	00	84	Volts Average Max.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
68	00	86	Volts Average Min.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
69	00	88	Volts Sum Max.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
70	00	8A	Volts Sum Min.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
71	00	8C	Current Ave Max.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
72	00	8E	Current Ave Min.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
73	00	90	Current Sum Max.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
74	00	92	Current Sum Min.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
75	00	94	Watt 1 Max.	●	-	●	-	-	-	-	-	-	-	-	-	-	-	-	-
76	00	96	Watt 1 Min.	●	-	●	-	-	-	-	-	-	-	-	-	-	-	-	-
77	00	98	Watt 2 Max.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
78	00	9A	Watt 2 Min.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
79	00	9C	Watt 3 Max.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
80	00	9E	Watt 3 Min.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
81	00	A0	Watt Sum Max.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
82	00	A2	Watt Sum Min.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
83	00	A4	VAr 1 Max.	●	-	●	-	-	-	-	-	-	-	-	-	-	-	-	-

Parameter No	Modbus Start Address		Parameter	Integra 2000 Configuration (No of Wires)			Integra 1000 Configuration (No of Wires)			Integra 1540 Configuration (No of Wires)			Integra 1560/80 Configuration (No of Wires)				SPR Configuration (No of Wires)		
	High Byte	Low Byte		4	3	2	4	3	2	4	3	2	4	3	1p 3w	1p 2w	4	3	2
84	00	A6	VAr 1 Min.	●	-	●	-	-	-	-	-	-	-	-	-	-	-	-	-
85	00	A8	VAr 2 Max.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
86	00	AA	VAr 2 Min.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
87	00	AC	VAr 3 Max.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
88	00	AE	VAr 3 Min.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
89	00	B0	VAr Sum Max.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
90	00	B2	VAr Sum Min.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
91	00	B4	VA1 Max.	●	-	●	-	-	-	-	-	-	-	-	-	-	-	-	-
92	00	B6	VA 1 Min.	●	-	●	-	-	-	-	-	-	-	-	-	-	-	-	-
93	00	B8	VA 2 Max.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
94	00	BA	VA 2 Min.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
95	00	BC	VA3 Max.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
96	00	BE	VA3 Min.	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
97	00	C0	VA Sum Max.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
98	00	C2	VA Sum Min.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
99	00	C4	Frequency Max.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
100	00	C6	Frequency Min.	●	●	●	-	-	-	-	-	-	-	-	-	-	-	-	-
101	00	C8	V L1-L2	●	-	-	●	-	-	●	-	-	●	-	●	-	●	-	-
102	00	CA	V L2-L3	●	-	-	●	-	-	●	-	-	●	-	-	-	●	-	-
103	00	CC	V L3-L1	●	-	-	●	-	-	●	-	-	●	-	-	-	●	-	-
104	00	CE	V L-L Ave	●	-	-	-	-	-	●	-	-	●	-	●	-	●	-	-
105	00	D0	V L1-L2 Max	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
106	00	D2	V L1-L2 Min	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
107	00	D4	V L2-L3 Max	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
108	00	D6	V L2-L3 Min	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
109	00	D8	V L3-L1 Max	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
110	00	DA	V L3-L1Min	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
111	00	DC	V L-L Ave Max	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
112	00	DE	V L-L Ave Min	●	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
113	00	E0	I Neutral	●	-	-	●	-	-	●	-	-	●	-	●	-	●	-	-
114	00	E2	Bus Frequency	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-

Parameter No	Modbus Start Address		Parameter	Integra 2000 Configuration (No of Wires)			Integra 1000 Configuration (No of Wires)			Integra 1540 Configuration (No of Wires)			Integra 1560/80 Configuration (No of Wires)				SPR Configuration (No of Wires)		
	High Byte	Low Byte		4	3	2	4	3	2	4	3	2	4	3	1p 3w	1p 2w	4	3	2
115	00	E4	Bus-Gen Phase Angle	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
116	00	E6	Bus Volts	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
117	00	E8	I Ground	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
118	00	EA	THD Volts 1	-	-	-	-	-	-	●	●	-	●	●	●	●	-	-	-
119	00	EC	THD Volts 2	-	-	-	-	-	-	●	●	-	●	●	●	-	-	-	-
120	00	EE	THD volts 3	-	-	-	-	-	-	●	●	-	●	●	-	-	-	-	-
121	00	F0	THD Current 1	-	-	-	-	-	-	●	●	-	●	●	●	●	-	-	-
122	00	F2	THD Current 2	-	-	-	-	-	-	●	●	-	●	●	●	-	-	-	-
123	00	F4	THD Current 3	-	-	-	-	-	-	●	●	-	●	●	-	-	-	-	-
125	00	F8	THD Voltage Mean	-	-	-	-	-	-	●	●	-	●	●	●	●	-	-	-
126	00	FA	THD Current Mean	-	-	-	-	-	-	●	●	-	●	●	●	●	-	-	-
128	00	FE	Power Factor (+Ind/ - Cap)	-	-	-	-	-	-	-	-	-	●	●	●	●	-	-	-
400	03	1E	Event: Log Number	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
401	03	20	Event: Relay Number	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
402	03	22	Event: Year	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
403	03	24	Event: Month	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
404	03	26	Event: Day	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
405	03	28	Event: Hour	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
406	03	2A	Event: Minute	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
407	03	2C	Event: Second	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-
408	03	2E	Event Log Increment	-	-	-	-	-	-	-	-	-	-	-	-	-	●	●	-

Appendix B: Instrument Holding Registers

In the table the “System Configuration” is defined by the number of wires, 4 represents a 3 phase 4 wire system, 3 represents a 3 phase 3 wire system and 2 represents a single phase system. A dot (●) in the system configuration column indicates that the parameter is valid for that wiring system, a dash (-) will return the value zero.

Parameter No	Modbus Start Address		Parameter	Integra 2000	Integra 1000	Integra 1540	Integra 1560/80	SPR
	High Byte	Low Byte						
1	00	00	Demand Time	●	●	●	●	-
2	00	02	Demand Period	●	●	●	●	-
3	00	04	Demand Interval	●	●	●	●	-
4	00	06	System Volts	●	●	●	●	●
5	00	08	System Current	●	●	●	●	●
6	00	0A	System Type	●	●	●	●	●
7	00	0C	Relay Pulse Width	●	●	●	●	-
8	00	0E	Energy Reset	●	●	●	●	-
12	00	16	Pulse Width Divisor	●	●	●	●	-
19	00	18	System Power	-	-	●	●	-
21	00	1A	Modbus Word Order	-	-	●	●	-
30	00	3A	Relay Number	-	-	-	-	●
31	00	3C	Trip Function	-	-	-	-	●
32	00	3E	Trip Mode	-	-	-	-	●
33	00	40	Trip Status	-	-	-	-	●
34	00	42	Trip 1 st Parameter	-	-	-	-	●
35	00	44	Trip 2 nd Parameter	-	-	-	-	●
36	00	46	Trip 3 rd Parameter	-	-	-	-	●
37	00	48	Trip 4 th Parameter	-	-	-	-	●
38	00	4A	Trip 5 th Parameter	-	-	-	-	●
39	00	4C	Trip 6 th Parameter	-	-	-	-	●
50	00	62	Real-time clock: Year	-	-	-	-	●
51	00	64	Real-time clock: Month	-	-	-	-	●
52	00	66	Real-time clock: Day	-	-	-	-	●

Parameter No	Modbus Start Address		Parameter	Integra 2000	Integra 1000	Integra 1540	Integra 1560/80	SPR
	High Byte	Low Byte						
53	00	68	Real-time clock: Hour	-	-	-	-	●
54	00	6A	Real-time clock: Minute	-	-	-	-	●
55	00	6C	Real-time clock: Second	-	-	-	-	●
150	01	2A	Secondary Volts	-	-	●	●	-
152	01	2B	Nominal Volts	-	-	●	●	-
155	01	34	*Analogue Hardware Max	-	-	-	●	-
156	01	36	*Analogue Hardware Min	-	-	-	●	-
157	01	38	*Analogue Output 1: Output Parameter	-	-	-	●	-
158	01	3A	*Analogue Output 1: Parameter Max	-	-	-	●	-
159	01	3C	*Analogue Output 1: Parameter Min	-	-	-	●	-
160	01	3E	*Analogue Output 1: Reading Top	-	-	-	●	-
161	01	40	*Analogue Output 1: Reading Bottom	-	-	-	●	-
162	01	42	*Analogue Output 1: Output Top	-	-	-	●	-
165	01	48	*Analogue Output 1: Output Bottom	-	-	-	●	-
166	01	4A	*Analogue Output 2: Output Parameter	-	-	-	●	-
167	01	4C	*Analogue Output 2: Parameter Max	-	-	-	●	-
168	01	4E	*Analogue Output 2: Parameter Min	-	-	-	●	-
169	01	50	*Analogue Output 2: Reading Top	-	-	-	●	-
170	01	52	*Analogue Output 2: Reading Bottom	-	-	-	●	-
171	01	54	*Analogue Output 2: Output Top	-	-	-	●	-
172	01	56	*Analogue Output 2: Output Bottom	-	-	-	●	-

* Requires Analogue Output option to be fitted.