Paladin Advantage
Universal Programmable Transducer
Communications Guide

A programmable transducer for the conversion of electrical parameters in an electrical system.

Contents

1 Paladin Advantage - Modbus Protocol Implementation

1.1 Modbus Protocol Overview

This section provides basic information for interfacing the Paladin Advantage Transducers to a Modbus Protocol network. If background information or more details of the Paladin Advantage implementation is required please refer to section 0 and 3 of this document.
Paladin Advantage offers the option of an RS485 communication facility for direct connection to SCADA or other communications systems using the Modbus Protocol RTU slave protocol. The Modbus Protocol establishes the format for the master's query by placing into it the device address, a function code defining the requested action, any data to be sent, and an error checking field. The slave's response message is also constructed using Modbus Protocol. It contains fields confirming the action taken, any data to be returned, and an error-checking field. If an error occurs in receipt of the message, Paladin Advantage will make no response. If the Paladin Advantage is unable to perform the requested action, it will construct an error message and send it as the response.
The electrical interface is 2-wire RS485, via 2 screw terminals. Connection should be made using twisted pair screened cable (Typically 22 gauge Belden 8761 or equivalent). All "A" and "B" connections are daisy chained together. The screens should also be connected to the “Gnd” terminal. To avoid the possibility of loop currents, an Earth connection should be made at only one point on the network.
Line topology may or may not require terminating loads depending on the type and length of cable used. Loop (ring) topology does not require any termination load.
The impedance of the termination load should match the impedance of the cable and be at both ends of the line. The cable should be terminated at each end with a 120 ohm (0.25 Watt min.) resistor.
A total maximum length of 3900 feet (1200 metres) is allowed for the RS485 network. A maximum of 32 electrical nodes can be connected, including the controller.
The address of each Paladin Advantage can be set to any value between 1 and 247.
The product also supports the broadcast address (00h); in this case all the devices connected to the bus will be written and none of them will send a response.

The minimum interval between the end of a response and the beginning of the next query (to the same device) is 150ms.
The minimum interval between the end of a response and the beginning of the next query (to a different device): 10ms.
Minimum response time-out (to be set on the master): 500ms.
The supervisory programme must allow this period of time to elapse before assuming that the Paladin Advantage Transducer is not going to respond.

The format for each byte in RTU mode is:

Coding System:	8-bit per byte
Data Format:	4 bytes (2 registers) per parameter. Most significant register first.
Error Check Field:	2 byte Cyclical Redundancy Check (CRC)
Framing:	1 start bit 8 data bits, least significant bit sent first 1 bit for even/odd parity (or no parity) 1 stop bit if parity is used; 2 bits if no parity

Data Coding
All data values in the Paladin Advantage Transducers are transferred as 32 bit Integers, (input and output) therefore each Paladin Advantage Transducers value is transferred using two MODBUS Protocol registers. All register read requests and data write requests must specify an even number of registers. Attempts to read/write an odd number of registers prompt the Paladin Advantage Transducers to return a MODBUS Protocol exception message. However, for compatibility with some SCADA systems, Paladin Advantage Transducers will respond to any single input or holding register read with an instrument type specific value

1.2 Modbus Protocol Holding Registers

The following table contains the available variables and the addresses of the registers where they are allocated.

Registers marked with “R” are read only, those marked with “R/W” are read and write registers.

All the measurements are expressed in real (primary) values: the CT’s and VT’s ratios are already included.

Some variables use two registers; when reading or writing these values, both registers must be read or written together using the function 03 “Read Holding registers” or 16 (10 hex) “Preset multiple holding registers”; Reading or writing an odd number of registers, or an even number of registers but across a couple is not allowed and generates an exception response 02 “Illegal data address”.

In reading operations, using the function 03 “Read holding registers”, the maximum number of registers that can be requested in a single query is 124 (38 for Q15/96B4W).
Requesting more than 124 (38) registers in a single query generates an exception response 03 “illegal data value”

Writing operations must be preceded by writing the value 0000 00A5h in the Write enable registers (40513 and 40514). Writing remains enabled until this value is changed or the instrument is switched off.
Writing operations performed when the content of the Write enable registers is incorrect generate an exception response 01 “Illegal function”

Data format is:

- **long (32 bits integer), big-endian, for the variables using two registers;**
- **word (16 bits integer), for the variables using one register.**

Variables which could be negative are expressed in “two’s complement”.

The registers of the variables not available for a specific model contain a value equal to zero.

The registers available for the single-phase versions are the system ones (Sys) or, in their absence, those of the phase1 (L1).

Registers from 40307 to 40318 (containing data of total harmonic distortion THD) and from 41281 to 41792 (containing data of individual harmonics) are expressed or as % of nominal value, or as % of RMS value, or as % of the fundamental value, according to the mode set in the programming menu of the instrument.

When reading the energy values (kWh+, kVArh+, kWh-, kVArh-, Partial kWh+), the Energy multiplier registers (40287 and 40288) must also be read.
Then the content of the energy registers must be multiplied by the Energy multiplier in order to get the correct energy values (expressed in Wh or VArh).
The Energy multiplier value is automatically adjusted by the meter when the primary values of the CT's and VT's are set, and will not change anymore as long as they are not changed again.
This is a sort of auto-scale to accommodate the energy values (which can vary from few kWh or kVArh to many GWh or GVArh according to the VT's and CT's primary values) in 32 bit numbers (two registers).

1.2.1 Paladin Advantage MODBUS Holding Register Parameters

Address (Register)	Address (HEX)	Parameter	Unit	R/W	Notes	254-XZZ
40001	0000	RESERVED				
…	…					
40256	00FF					
40257	0100	V L1-N	1mV	R		✓
40258	0101					
40259	0102	V L2-N	1mV	R		✓
40260	0103					
40261	0104	V L3-N	1mV	R		✓
40262	0105					
40263	0106	V L1-L2	1mV	R		✓
40264	0107					
40265	0108	V L2-L3	1mV	R		✓
40266	0109					
40267	010A	V L3-L1	1mV	R		✓
40268	010B					
40269	010C	I L1	1mA	R		✓
40270	010D					
40271	010E	I L2	1mA	R		✓
40272	010F					
40273	0110	I L3	1mA	R		✓
40274	0111					
40275	0112	F	1mHz	R	L1	✓
40276	0113					
40277	0114	P Sys	1W	R	P L1 + P L2 + P L3	✓
40278	0115					
40279	0116	Q Sys	1VAr	R	Q L1 + Q L2 + Q L3	✓
40280	0117					
40281	0118	P.F. Sys	0.001	R	P Sys / S Sys	✓
40282	0119					
40283	011A	kWh+ Sys	1Wh	R/W		✓
40284	011B					
40285	011C	kVArh+ Sys	1VArh	R/W		✓
40286	011D					
40287	011E	Energy multiplier	1	R		✓
40288	011F					
40289	0120	V L-L Sys	1mV	R	(V L1-L2 + V L2-L3 + V L3-L1) / 3	✓
40290	0121					
40291	0122	V L-N Sys (V for C/Q 15/96	1mV	R	(V L1-N + V L2-N + V L3-N) / 3	✓
40292	0123					
40293	0124	I Sys	1mA	R	(I L1 + I L2 + I L3) / 3	✓
40294	0125					
40295	0126	Delta V L-L	%	R	(V LL max - V LL min) / V LL med	✓
40296	0127					
40297	0128	Delta V L-N	%	R	(V LN max - V LN min) / V LN med	✓
40298	0129					
40299	012A	Delta I	%	R	(I L max - I L min) / I L med	✓
40300	012B					
40301	012C	I Neutral	1mA	R	Vector sum I L1 + I L2 + I L3	✓
40302	012D					
40303	012E	Cos Phi Sys	0.001	R	P Sys / S Sys (Fundamentals)	✓
40304	012F					
40305	0130	P.F. Avg Sys	0.001	R	Atan kVArh+ / kWh+	✓
40306	0131					
40307	0132	THD V L1	0.1 %	R	% Nom. or RMS or Fundamental	✓
40308	0133					
40309	0134	THD V L2	0.1 %	R	% Nom. or RMS or Fundamental	✓
40310	0135					
40311	0136	THD V L3	0.1 %	R	% Nom. or RMS or	✓

Address (Register)	Address (HEX)	Parameter	Unit	R/W	Notes	254-XZZ
40312	0137				Fundamental	
40313	0138	THD I L1	0.1 %	R	% Nom. or RMS or Fundamental	✓
40314	0139					
40315	013A	THD I L2	0.1 %	R	% Nom. or RMS or Fundamental	✓
40316	013B					
40317	013C	THD I L3	0.1 %	R	% Nom. or RMS or Fundamental	✓
40318	013D					
40319	013E	kWh- Sys	1Wh	R/W		✓
40320	013F					
40321	0140	kVArh- Sys	1VArh (1mAh)	R/W		✓
40322	0141					
40323	0142	S Sys	1VA	R	S L1 + S L2 + S L3	✓
40324	0143					
40325	0144	P L1	1W	R		✓
40326	0145					
40327	0146	P L2	1W	R		✓
40328	0147					
40329	0148	P L3	1W	R		✓
40330	0149					
40331	014A	Q L1	1VAr	R		✓
40332	014B					
40333	014C	Q L2	1VAr	R		✓
40334	014D					
40335	014E	Q L3	1VAr	R		✓
40336	014F					
40337	0150	S L1	1VA	R	V L1 rms x I L1 rms	✓
40338	0151					
40339	0152	S L2	1VA	R	V L2 rms x I L2 rms	✓
40340	0153					
40341	0154	S L3	1VA	R	V L3 rms x I L3 rms	✓
40342	0155					
40343	0156	P.F. L1	0.001	R	P L1 / S L1	✓
40344	0157					
40345	0158	P.F. L2	0.001	R	P L2 / S L2	✓
40346	0159					
40347	015A	P.F. L3	0.001	R	P L3 / S L3	✓
40348	015B					
40349	015C	Cos Phi L1	0.001	R	P L1 / S L1 (Fundamentals)	✓
40350	015D					
40351	015E	Cos Phi L2	0.001	R	P L2 / S L2 (Fundamentals)	✓
40352	015F					
40353	0160	Cos Phi L3	0.001	R	P L3 / S L3 (Fundamentals)	✓
40354	0161					
40355	0162	P max Sys	1W	R/W		✓
40356	0163					
40357	0164	P avg Sys	1W	R/W	Moving average	✓
40358	0165					
40359	0166	I max L1	1mA	R/W		✓
40360	0167					
40361	0168	I max L2	1mA	R/W		✓
40362	0169					
40363	016A	I max L3	1mA	R/W		✓
40364	016B					
40365	016C	I avg L1 (I avg Sys for C/Q	1mA	R/W	Moving average	✓
40366	016D					
40367	016E	I avg L2	1mA	R/W	Moving average	✓
40368	016F					
40369	0170	I avg L3	1mA	R/W	Moving average	✓
40370	0171					

Address (Register)	Address (HEX)	Parameter	Unit	R/W	Notes	254-XZZ
40371	0172	RESERVED				
…	…					
40404	0193					
40405	0194	Phase Angle Sys	0.001 deg.	R		✓
40406	0195					
40407	0196	Phase Angle L1	0.001 deg.	R		✓
40408	0197					
40409	0198	Phase Angle L2	0.001 deg.	R		✓
40410	0199					
40411	019A	Phase Angle L3	0.001 deg.	R		✓
40412	019B					
40413	019C	RESERVED FOR FUTURE ADDITIONAL VARIABLES				
…	…					
40512	01FF					
40513	0200	WRITE ENABLE		R/W	0000 00A5 = Enabled	✓
40514	0201					
40515	0202	DEVICE LOGIC ADDRESS		R/W		✓
40516	0203					
40517	0204	SYSTEM DATA AND SETTINGS				
…	…					
40768	02FF					
40769	0300	RESERVED				
…	…					
41024	03FF					
41025	0400	RESERVED				
…	…					
41280	04FF					

Notes

Registers from 40307 to 40318 (containing data of total harmonic distortion THD) are expressed as

a. % of nominal value
b. % of RMS value
c. % of the fundamental value

according to the mode set in the programming menu of the instrument.

2 RS485 General Information

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	1200 m
Maximum Data Rate	10 M baud
Maximum Common Mode Voltage	12 V to –7 V
Minimum Driver Output Levels (Loaded)	+/– 1.5 V
Minimum Driver Output Levels (Unloaded)	+/– 6 V
Drive Load	Minimum 60 ohms
Driver Output Short Circuit Current Limit	150 mA to Gnd, 250 mA to 12 V 250 mA to –7 V
Minimum Receiver Input Resistance	12 kohms
Receiver Sensitivity	+/– 200 mV

Further information relating to RS485 may be obtained from either the EIA or the various RS485 device manufacturers, for example Texas Instruments or Maxim Semiconductors. This list is not exhaustive.

2.1 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”.

2.2 Technical Data

TE's Crompton Instruments are equipped, to transfer data to and from a PC, PLC and other supervision systems, with an (optional) insulated serial interface according to RS-485 standard (ANSI/TIA/EIA-485-A-98 R2003).

The data format used is the following:

Baud-rate:	9600 or 19200 (or 38400 and 57600 on some models) bps (programmable, default 9600)
Data length:	8 bits
Parity:	none or even or odd (programmable, default none)
Stop bits:	1 with even or odd parity, 2 with parity none (or programmable on some models)

2.3 Wiring of the line

It is of fundamental importance, for its proper operation, to perform a correct wiring of the RS485 line and of all devices connected to it.

Use a good quality cable, suitable for a 485 line, preferably 24 AWG, shielded, twisted pair type (Belden 9841 or similar).
Note: The RS485 interface of the Paladin 254-XZZ is completely isolated and floating, and therefore does not require the ground connection.
Connect all the devices in cascade (such as in-out) without making stubs from the main line.
Note: Ensure the continuity of the screen between a piece of wire and the next.
Terminate the line at both ends with a resistance equal to the characteristic impedance of the cable used (typically 120 ohm 1/4 W)
Note: Frequently the master (PLC or data acquisition system) has internal terminating resistor that can be inserted via dip switches or jumpers, or through a software configuration of the communication module (consult the manufacturer of the PC or the acquisition system).
Connect the shield to ground at one end only of the line.

2.3.1 Polarization of the line

When on the RS485 line is not in progress a data exchange, usually all the devices connected to it (including the Master) have their transceiver switched to reception; in this condition it is necessary that the line is polarized to ensure that the line itself is in a defined state recognizable by the receivers.
This state is verified when, in the absence of communication, it is possible to measure a potential difference between the wires A(+) and B(-) greater than +200 mV.
Frequently the master (PLC or data acquisition system) is equipped internally with the polarization of the line, which can be activated via dip switches or jumpers, or through a software configuration of the communication module (consult the manufacturer of the PC or the acquisition system).
If this were not true, it is necessary to add the polarization to one of the two ends of the line in this way:

Remove the original terminating resistor and replace it with a 133 ohm 1/4 W
Connect a resistor of 619 ohm 1/4 W between the wire A(+) and a dc power source of +5 V
Note: The power source must be isolated and floating.
Connect a resistor of 619 ohm 1/4 W between the wire B(-) and the ground of the above-mentioned power source.
Verify that, in the absence of communication, it is possible to measure a potential difference between the wires A (+) and B (-) greater than +200 mV.
Note: The over mentioned resistors values are valid for a cable with a characteristic impedance of 120 ohms and a power source of +5 V.

Note: Frequently, in environments with small electromagnetic disturbances, the communication may appear to function smoothly, even with a poor wiring, also in the absence of the proper termination and polarization of the line.
This should not lead to neglect the need of performing a correct wiring, including termination and polarization: this is the only way to get a robust and reliable communication line.

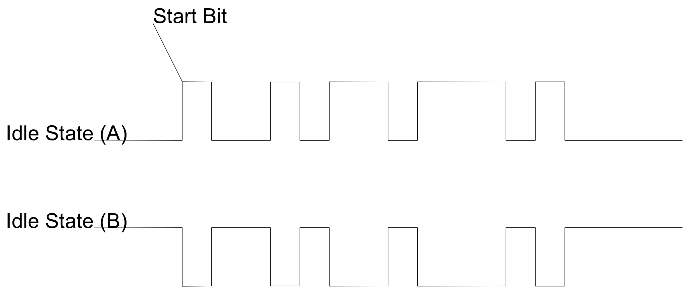
Terminals naming
At present there are two ways to name the two wires of the RS485:
According to the EIA-485 standard: A(-) and B(+).

According to many manufacturers of transceivers (including Texas Instruments, Maxim, Intersil): A(+) and B(-).

The signs (+) and (-) indicate which of the two conductors must have a greater potential than the other when the line is idle.
TE's Crompton Instruments adopted, for the designation of the terminals on its instruments, the second option: A(+) and B(-).
This is because, while not adhering to the EIA-485 standard, it is the denomination most widely used in the industrial field.
If you are having problems establishing communication, one of the first tests to do is the inversion of the two wires A and B: it does not involve any danger of damage to either the transmitters or receivers.

2.4 A and B terminals

The A and B connections to the Paladin Advantage Transducer products can be identified by the signals present on them whilst there is activity on the RS485 bus:

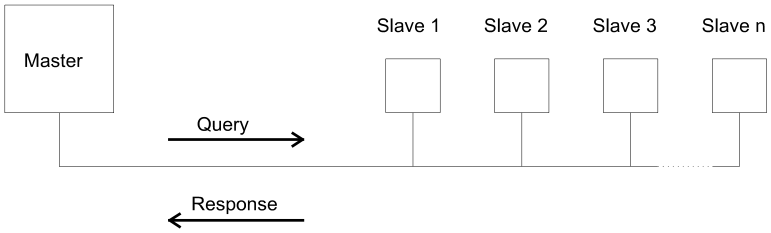


2.5 Troubleshooting

- Start with a simple network, one master and one slave. With Paladin Advantage Transducers product this is easily achieved as the network can be left intact whilst individual instruments are disconnected by removing the RS485 connection from the rear of the instrument.
- Check that the network is connected together correctly. That is all of the “A’s” are connected together, and all of the “B’s” are connected together.
- 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 not perform well when they receive an echo of the message they are transmitting. SpecView and PCView (PC software) with a RS232 to RS485 converter are believed to include this feature.
- 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.
- Any Modbus communications will impact the performance of the instrument and may result in a response time in excess of the specification.
- Check that the MODBUS Protocol 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.
- 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.

3 MODBUS Protocol General Information

Communication on a MODBUS Protocol Network is initiated (started) by a “Master” sending a query to a “Slave”. The “Slave“, which is constantly monitoring the network for queries addressed to it, will respond by performing the requested action and sending a response back to the ”Master”. 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 Paladin Advantage Transducer does not support the broadcast address.

3.1 MODBUS Protocol 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 then the message is ignored, if the slave is unable to perform the requested action, then it will construct an error message and send it as its response.
The MODBUS Protocol functions used by the Paladin Advantage Transducer copy 16 bit register values between master and slaves. However, the data used by the Paladin Advantage Transducer is in 32 bit Integers. Thus each instrument parameter is conceptually held in two adjacent MODBUS Protocol registers.
Query
The following example illustrates a request for a single floating point parameter i.e. two 16-bit Modbus Protocol Registers.

First Byte							Last Byte
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 Paladin Advantage Transducer do not support the broadcast address.
Function Code:	8-bit value telling the addressed slave what action is to be performed. (3, 4, 8 or 16 are valid for Paladin Advantage Transducer)
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. As registers are used in pairs and start at zero, then this must be an even number.
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. As registers are used in pairs, then this must be an even number.
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 floating point parameter i.e. two 16-bit Modbus Protocol Registers.

First Byte ... Last Byte

Slave Address	Function Code	Byte Count	First Register (Hi)	First Register (Lo)	Second Register (Hi)	Second Register (Lo)	Error Check (Lo)	Error Check (Hi)

Slave Address: 8-bit value representing the address of slave that is responding.
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
First Register (Hi)*: The top (most significant) eight bits of a 16-bit number representing the first register requested in the query.
First Register (Lo)*: The bottom (least significant) eight bits of a 16-bit number representing the first register requested in the query.
Second Register (Hi)*: The top (most significant) eight bits of a 16-bit number representing the second register requested in the query.
Second Register (Lo)*: The bottom (least significant) eight bits of a 16-bit number representing the second register 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.

* These four bytes together give the value of the floating point parameter requested.

Exception Response
If an error is detected in the content of the query (excluding parity errors and Error Check mismatch), then an error response (called an exception response), will be sent to the master. The exception response is identified by the function code being a copy of the query function code but with the most-significant bit set. The data contained in an exception response is a single byte error code.

First Byte ... Last Byte

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

Slave Address: 8-bit value representing the address of slave that is responding.
Function Code: 8 bit value which is the function code in the query OR'ed with 80 hex, indicating that 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 “Table Of Exception 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.

3.2 Serial Transmission Modes

There are two MODBUS Protocol serial transmission modes, ASCII and RTU. Paladin Advantage Transducer does not support the ASCII mode.

In RTU (Remote Terminal Unit) mode, each 8-bit byte is used in the full binary range and is not limited to ASCII characters as in ASCII Mode. The greater data density allows better data throughput for the same baud rate, however each message must be transmitted in a continuous stream. This is very unlikely to be a problem for modern communications equipment.

The format for each byte in RTU mode is:

Coding System:	Full 8-bit binary per byte. In this document, the value of each byte will be shown as two hexadecimal characters each in the range 0-9 or A-F.
Line Protocol:	1 start bit, followed by the 8 data bits. The 8 data bits are sent with least significant bit first.
User Option Of Parity And Stop Bits:	No Parity and 2 Stop Bits No parity and 1 stop bit is NOT supported. Even Parity and 1 Stop Bit. Odd Parity and 1 Stop Bit.
User Option of Baud Rate:	9600 ; 19200 ; 38400

The baud rate, parity and stop bits must be selected to match the master's setting**s.**

3.3 MODBUS Protocol Message Timing (RTU Mode)

A MODBUS Protocol message has defined beginning and ending points. The receiving devices recognises the start of the message, reads the "Slave Address" to determine if they are being addressed and knowing when the message is completed they can use the Error Check bytes and parity bits to confirm the integrity of the message. If the Error Check or parity fails then the message is discarded.
In RTU mode, messages starts with a silent interval of at least 3.5 character times.
The first byte of a message is then transmitted, the device address.
Master and slave devices monitor the network continuously, including during the 'silent' intervals. When the first byte (the address byte) is received, each device checks it to find out if it is the addressed device. If the device determines that it is the one being addressed it records the whole message and acts accordingly, if it is not being addressed it continues monitoring for the next message.
Following the last transmitted byte, a silent interval of at least 3.5 character times marks the end of the message. A new message can begin after this interval.
The entire message must be transmitted as a continuous stream. If a silent interval of more than 1.5 character times occurs before completion of the message, the receiving device flushes the incomplete message and assumes that the next byte will be the address byte of a new message.
Similarly, if a new message begins earlier than 3.5 character times following a previous message, the receiving device may 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.

3.4 How Characters are Transmitted Serially

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

Transmit Character = Start Bit + Data Byte + Parity Bit + 1 Stop Bit (11 bits total):

Least Significant Bit (LSB) ↓ (bit 1) — Most Significant Bit (MSB) ↓ (bit 8)

Start	1	2	3	4	5	6	7	8	Parity	Stop

Transmit Character = Start Bit + Data Byte + 2 Stop Bits (11 bits total):

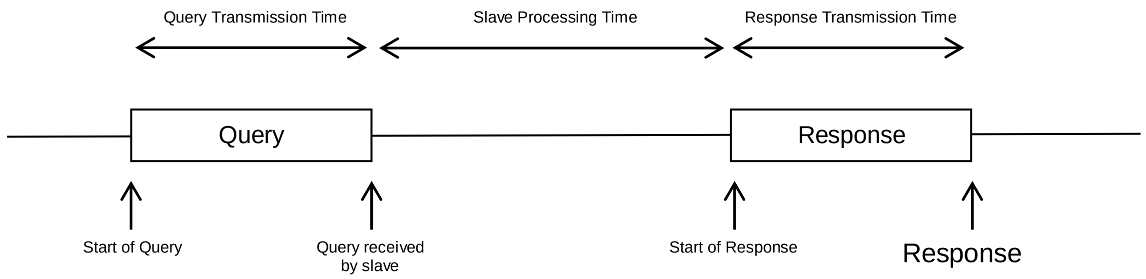
Start	1	2	3	4	5	6	7	8	Stop	Stop

Paladin Advantage Transducer additionally support No parity, One stop bit.

Transmit Character = Start Bit + Data Byte + 1 Stop Bit (10 bits total):

Start	1	2	3	4	5	6	7	8	Stop

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.



3.5 Error Checking Methods

Standard MODBUS Protocol 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 byte in the message.

3.5.1 Parity Checking

If parity checking is enabled – by selecting either Even or Odd Parity - the quantity of "1's" will be counted in the data portion of each transmit 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 transmit character 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. Also, if No Parity checking is specified and one stop bit is selected the transmit character is effectively shortened by one bit.

3.5.2 CRC Checking

The error check bytes of the MODBUS Protocol 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 transmit 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 message byte is exclusive OR'ed with the lower half of the 16 bit register. The register is then shifted eight times in the direction of the least significant bit (LSB), with a zero filled into the most significant bit (MSB) position. After each shift 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 all eight shifts have been performed. After the last shift, the next 8-bit message byte is exclusive OR'ed with the lower half of the 16 bit register, 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.

3.6 Function Codes

The function code part of a MODBUS Protocol message defines the action to be taken by the slave. Paladin Advantage Transducer supports the following function codes:

Code	MODBUS Protocol name	Description
03	Read Holding Registers	Read the contents of read/write location (4X references)
08	Diagnostics	Only sub-function zero is supported. This returns the data element of the query unchanged.
16 (10h)	Pre-set Multiple Registers	Set the contents of read/write location (4X references)

3.7 MODBUS Protocol Commands supported

All Paladin Advantage Transducers support the "Read Holding Register" (4X registers) and the "Pre-set Multiple Registers" (write 4X registers) commands of the MODBUS Protocol RTU protocol. All values stored and returned are in floating point format to IEEE 754 with the most significant register first.

3.8 Holding Registers

3.8.1 Read Holding Registers

MODBUS Protocol code 03 reads the contents of the 4X registers.
Example
The following query will request the V L1-N:

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 "Starting Address" and the "Number of Points" must be even numbers to request a floating point variable. If the "Starting Address" or the "Number of points" is odd then the query will fall in the middle of a floating point variable the product will return an error message.

The following response returns the contents of V L1-N, But see also "Exception Response" later.

Field Name	Example (Hex)
Slave Address	01
Function	03
Byte Count	04
Data, High Reg, High Byte	00
Data, High Reg, Low Byte	00
Data, Low Reg, High Byte	00
Data, Low Reg, Low Byte	E6
Error Check Low	F7
Error Check High	CF

3.8.2 Write Holding Registers

MODBUS Protocol code 16 (10h hex) writes the contents of the 4X registers.
<u>Example</u>
The following query will set the Write Enable register to 0000 00A5 (hex), which enables writing to other registers. Writing remains enabled until this value is changed or the power supply is removed.

Field Name	Example (Hex)
Slave Address	01
Function	10
Starting Address High	02
Starting Address Low	00
Number of Registers High	00
Number of Registers Low	02
Byte Count	04
Data, High Reg, High Byte	00
Data, High Reg, Low Byte	00
Data, Low Reg, High Byte	00
Data, Low Reg, Low Byte	A5
Error Check Low	2A
Error Check High	B4

Note: Data must be written in register pairs i.e. the "Starting Address" and the "Number of Points" must be even numbers to write a floating point variable. If the "Starting Address" or the "Number of points" is odd then the query will fall in the middle of a floating point variable the product will return an error message. In general only one 32-bit Integer value can be written per value.

The product does not respond to a successful write.

3.9 Exception Response

If the slave in the "Write Holding Register" example above, did not support that function then it would have replied with an Exception Response as shown below. The exception function code is the original function code from the query with the MSB set i.e. it has had 80 hex logically ORed with it. The exception code indicates the reason for the exception. The slave will not respond at all if there is an error with the parity or CRC of the query. However, if the slave can not process the query then it will respond with an exception. In this case a code 01, the requested function is not support by this slave.

Field Name	Example (Hex)
Slave Address	01
Function	10 OR 80 = 90
Exception Code	01
Error Check Low	8D
Error Check High	C0

3.10 Exception Codes

3.10.1 Table of Exception Codes

Paladin Advantage Transducers support the following exception codes:

Exception Code	MODBUS Protocol name	Description
01	Illegal Function	The function code is not supported by the product OR Writing not enabled
02	Illegal Data Address	Attempt to access an invalid address
03	Illegal Data Value	Attempt to set a floating point variable to an invalid value

TE Energy – innovative and economical solutions for the electrical power industry: cable accessories, connectors & fittings, insulators & insulation, surge arresters, switching equipment, street lighting, power measurement and control.

Tyco Electronics UK Ltd
TE Energy
Freebournes Road
Witham, Essex CM8 3AH
Phone: +44 (0)870 870 7500
Fax: +44 (0)870 240 5287

Email: Crompton.info@te.com
www.crompton-instruments.com