

ETC2002

Modbus Communications Protocol
Reference Guide

Every effort has been made to ensure that the material herein is complete and accurate. However, the manufacturer is not responsible for any mistakes in printing or faulty instructions contained in this book. Notification of any errors or misprints will be received with appreciation.

For further information regarding a particular installation, operation or maintenance of equipment, contact the manufacturer or your local representative or distributor.

Modbus is a trademark of Modicon, Inc.

Table of Contents

| | |
|---|-----------|
| 1 GENERAL | 5 |
| 2 MODBUS PROTOCOL IMPLEMENTATION..... | 6 |
| 2.1 Transmission Modes..... | 6 |
| 2.2 Address Field..... | 6 |
| 2.3 Function Field | 6 |
| 2.4 Exception Responses..... | 6 |
| 2.5 Modbus Register Addresses | 6 |
| 2.6 Data Formats | 7 |
| 2.7 Password Protection..... | 7 |
| 3 DATA SERVER | 8 |
| 3.1 Data Server Features | 8 |
| 3.2 Using Real-time Data Exchanges..... | 9 |
| 3.2.1 Configuring Real-time Data Exchanges..... | 9 |
| 3.2.2 Accessing Real-time Data Exchanges..... | 9 |
| 3.2.3 Accessing Data Log Files | 10 |
| 3.3 Using Status Event Exchanges and Event Logs | 11 |
| 3.3.1 Configuring Event Exchanges..... | 11 |
| 3.3.2 Accessing the Event Log File..... | 11 |
| 3.4 Using Data Write Exchanges..... | 12 |
| 4 MODBUS REGISTER MAP..... | 13 |
| 4.1 Data Server Registers | 13 |
| Real-time Data Exchange Block..... | 13 |
| Data Write Exchange Block..... | 13 |
| Data Log Transfer Blocks | 13 |
| Event Log Transfer Blocks..... | 14 |
| 4.2 Digital Inputs | 14 |
| 4.3 Device Control and Status Registers..... | 15 |
| Device Mode Control Registers (factory set)..... | 15 |
| Device Diagnostics Register (bit map) | 15 |
| Device Authorization Register | 15 |
| Factory Diagnostic Registers..... | 15 |
| 4.4 Device Setup Registers | 15 |
| Factory Device Settings and Identification | 15 |
| Device Options Setup | 16 |
| Time/Date Setup..... | 16 |
| Clock Setup and Status | 16 |
| Modem Setup | 17 |
| Communication Ports Setup | 17 |
| Network Setup | 17 |
| RF Modem Setup..... | 17 |
| Password Setup..... | 18 |
| Expert Power Service Setup..... | 18 |
| Internet Service Provider (ISP) Accounts | 18 |
| Data Server Setup | 18 |
| Gateway Setup | 19 |
| Device Routing Table | 20 |
| Routing IP Table | 20 |

| | |
|-----------------------------------|-----------|
| 5 DATA FORMATS | 21 |
| Timestamp | 21 |
| Event Type ID | 21 |
| Event Data | 21 |
| Device Diagnostics (bitmap) | 21 |
| Shared Access Rules | 22 |

1 General

This document specifies a subset of the Modbus serial communications protocol used to transfer data between a master computer station and the ETC2002. The document provides the complete information necessary to develop third-party communications software capable of communication with the ETC2002. Additional information concerning communications operation, configuring the communications parameters, and communications connections is found in "ETC2002 Installation and Operation Manual".

IMPORTANT

Most of the advanced features are configured using multiple setup parameters that can be accessed in a number of contiguous registers. When writing the setup registers, it is recommended to write all the registers at once using a single request.

2 Modbus Protocol Implementation

For detailed information about Modbus protocol, Modbus message framing and error checking, refer to the "Modicon Modbus Protocol Reference Guide". It can be downloaded from the Modicon's Web site at www.modicon.com. The following paragraphs outline some issues concerning the implementation of the Modbus protocol in the ETC2002.

2.1 Transmission Modes

The ETC2002 can be set up to communicate on a Modbus network using ASCII or RTU transmission mode. Refer to the "ETC2002 Installation and Operation Manual " on how to select the transmission mode in your device.

2.2 Address Field

The address field contains a user assigned address of the instrument (1-247) on a Modbus network. Broadcast mode using address 0 is not supported.

2.3 Function Field

The Modbus functions implemented in the ETC2002 are shown in Table 2-1. Function 04 can be used in the same context as function 03.

Table 2-1 Modbus Function Codes

| Code (decimal) | Meaning in Modbus | Action |
|-----------------|---------------------------|--------------------------|
| 03 | Read holding registers | Read multiple registers |
| 04 | Read input registers | Read multiple registers |
| 06 | Preset single register | Write single register |
| 16 | Preset multiple registers | Write multiple registers |
| 08 ¹ | Loop-back test | Communications test |

¹The ETC2002 supports only diagnostic code 0 - return query data.

2.4 Exception Responses

The instrument sends an exception response when an error is detected in the received message. To indicate that the response is notification of an error, the high order bit of the function code is set to 1. Implemented exception response codes:

- 1 - Illegal function
- 2 - Illegal data address
- 3 - Illegal data value
- 10 - Gateway is busy. Remote access is temporarily unavailable.

When the character framing, parity, or redundancy check detects a communication error, processing of the master's request stops. The instrument will not act on or respond to the message.

2.5 Modbus Register Addresses

The ETC2002 Modbus registers are numbered in the range of 0 to 65535. From the Modbus applications, the ETC2002 Modbus registers can be accessed by simulating holding registers of the Modicon 584, 884 or 984 Programmable Controller, using a 5-digit "4XXXX" or 6-digit "4XXXXX" addressing scheme. To map the ETC2002 register address to the range of the Modbus holding registers, add a value of 40001 to the ETC2002 register address. When a register address exceeds 9999, use a 6-digit addressing scheme by adding 400001 to the ETC2002 register address.

2.6 Data Formats

The ETC2002 uses two data formats to pass data between a master application and the device: 16-bit short integer and 32-bit long integer formats. Data logged by the ETC2002 Data Server is collected from instruments in 16-bit short integer format.

2.7 Password Protection

The ETC2002 has a password protection option allowing you to protect your setups and logs from being changed or cleared through communications. You can disable or enable password protection through communications or via the Terminal. For details, refer to the ETC2002 “Installation and Operation Manual”. When password protection is enabled, the user password you set in the ETC2002 should be written into the device authorization register (44378-44379) before another write request is issued. If the correct password is not supplied while password protection is enabled, the ETC2002 will respond to all write requests with the exception code 01 (illegal operation). It is recommended to clear the password register after you have completed your changes in order to activate password protection.

3 Data Server

3.1 Data Server Features

The Data server allows the user to save the time required for polling multiple devices, especially via slow serial interfaces, by utilizing the ETC2002 non-volatile memory for automatic acquiring data. The user application can then read blocks of ready data directly from the ETC2002 memory without the need to wait until slow devices respond to master requests. The Data server can also perform a delayed write of data so the application needs not to wait until the device would be accessible, but rather can write data to the ETC2002 memory exchanges, and then the ETC2002 takes responsibility for delivering data to the target devices. The Data server can also periodically poll device status registers and record events to a log file or give alerts on specific events.

The following table lists the configurable Data server options.

Table 3-1 Data Server Options

| Description | Range |
|------------------------------|----------------------------|
| Data server enabled | 1=enabled, 0=disabled |
| Continuous polling | 1=enabled, 0=disabled |
| Data polling interval | 1-1000 min |
| Data logging interval, polls | 0-1000, 0=logging disabled |

When the Data server is enabled, it acquires data from the connected devices either continuously, or on a periodic basis. If continuous polling is disabled, the devices are requested periodically at predefined poll intervals. If continuous polling is enabled, the ETC2002 retrieves data from the devices in turn without pauses.

Acquired real-time data is normally stored in the ETC2002 exchanges (memory blocks) and is periodically refreshed at programmed intervals. The ETC2002 provides up to 250 real-time data exchanges of 1 to 114 registers long. These exchanges can be directly accessed by the master application.

If the user wishes, acquired data can be periodically moved to historical circular files where it will be kept for a long time until it is overwritten by newer data. Data is stored to historical files at regular intervals. The logging interval is always represented as a whole number of data polling intervals even if the polling itself is not synchronized with polling intervals like in the event of continuous polling. Since the ETC2002 tries to synchronize polling devices with the beginning of an hour, it is recommended to program a logging interval in such a manner that it would be a whole divisor of 60 minutes. Historical records can be accessed one record at a time either in an arbitrary order, or in a sequential manner.

The ETC2002 also provides up to 250 write data exchanges, of 1 to 12 registers long, and up to 250 status event data exchanges of 1 register long. Table 3-2 gives a summary of the available data exchanges.

Table 3-2 Data Server Exchanges

| Exchange Type | Number of Exchanges | Exchange Block Size, registers |
|-------------------------|---------------------|--------------------------------|
| Real-time data exchange | 250 | 1-114 |
| Status event exchange | 250 | 1 |
| Data write exchange | 250 | 1-12 |

NOTE:

The real-time, status event and write data exchanges with the same exchange number are internally linked in the ETC2002 to the same device ID. If you change the device ID for one of the exchanges, the same ID will be taken for other configured exchanges that have the same exchange number.

3.2 Using Real-time Data Exchanges

3.2.1 Configuring Real-time Data Exchanges

Real-time data exchanges are used for periodic retrieving data from the connected devices. The following table shows the parameters that should be configured before enabling a data exchange.

Table 3-3 Real-time Data Exchange Setup

| Description | Range |
|--------------------------------|-----------------------|
| Exchange number | 0-249 |
| Start device register address | 0-65535 |
| Exchange block size, registers | 1-114 |
| Device ID | 1-247 |
| Polling enabled | 1=enabled, 0=disabled |
| RTC update enabled (optional) | 1=enabled, 0=disabled |

A data exchange can store from 1 to 114 contiguous 16-bit registers retrieved from a single device. A block of registers for an exchange is defined by the start register address and the exchange block size. Refer to Modbus guides for specific devices for available address ranges.

You can individually enable or disable polling data for a specific exchange. If the device has an onboard real-time clock, you can enable periodic RTC updates from the ETC2002 clock, which may be synchronizes with the precise satellite GPS clock using an optional IRIG-B input.

Each data exchange utilizes 6 extra registers for time stamping and identification as shown in Table 3-4, so the actual memory required for one data exchange is 6 + exchange block size. The total size of all exchanges should not exceed 9000 registers.

Table 3-4 Data Exchange Record

| Description | Range |
|-------------------------------------|--|
| Record heading | |
| Record status | 0=no data, -1=corrupted record, 1-65534 = device response code |
| Record time, seconds since 1/1/1970 | |
| Record sequence number | 0- 65535 |
| Data block (variable size) | |
| Data registers (up to 114) | |

If data logging is globally enabled in the Data server, the retrieved data is periodically moved from the data exchange to a log file in the same format, as it is stored in the exchange (see Table 3-4). Each exchange is provided with a separate data log file.

A data record is always preceded by a status register that indicates whether the data is healthy. Each record may be identified in a file by a unique 16-bit sequence number that is incremented modulo 65536 with each new record.

3.2.2 Accessing Real-time Data Exchanges

Real-time data exchanges occupy a single continuous block of registers without gaps and are located in the ETC2002 memory in the address range of 0 to 9000. Since exchanges may have different size, the location of each exchange in the memory depends on the size of all preceding exchanges. Use the following formula to define the location of the data exchange:

$$\text{Exchange}_N \text{ Address} = \sum_{n=0}^{N-1} (\text{Exchange}_n \text{ Block Size} + 6)$$

Note that a location of the exchange in the memory is changed when you change the data exchange setup, so the previously acquired data would be lost.

3.2.3 Accessing Data Log Files

Each data exchange is provided with a separate circular data log file. The Data server allocates a total of 1.5 Mbytes of non-volatile memory for all data log files, which will be equally divided between all active exchanges.

File transfers are performed through two blocks of registers: a master request block and a read-only file transfer block, which are shared by all files. After a master application has written the request with a file exchange number (and possibly a file position) into the file request block, the requested data is available for a read through the file transfer block registers. File transfer functions allow changing the file position in order to point to the desired record in a selected log file. The sequence number of the first (oldest) record available in a file may be inspected via the file request block after the file exchange number is written into the block.

The file transfer block occupies 120 registers in the address range of 9000 to 9119. The following table shows a layout of the data log request block. Refer to Table 3-3 for the file transfer block layout.

Table 3-5 Data Log Request Block

| Description | Range |
|---|--|
| Function | 3 = set file position 5 = reset file position 11 = read record |
| Record sequence number (with functions 3, 11) | 0-65535 |
| First (oldest) record sequence number | 0-65535 |
| Exchange number | 0-249 |

Data log files can be read either in a sequence record-by-record, or in a random order. Every write to the file request block fills the file transfer block with the data of the record pointed to by the file read pointer. If you want to begin reading a file from a particular record, which sequence number is known, you can change the pointer position by issuing the Set-File-Position request with the desired sequence number. If you want to read a file from the beginning, write the Reset-File-Position request that moves the pointer to the oldest file record. If you use the Read-Record request that does not change the file position, then you will continue reading the file from the record following the one you have read the last time you accessed the file.

You need not explicitly move the file position to the following record if you want to continue reading a file in sequence after you have uploaded the current record. Instead, continue reading the record data through the file transfer block: each access to the file transfer block automatically advances the file pointer to the next record. That's why you should read all data in the file transfer block by a single request: the next read will give you the following file record.

The file transfer is completed after you have read the last record of the file. Before storing a file record to your database, always check the record status word. This word set to 0 indicates that the file read pointer does not point to any record within the file, and you should not store any record that has this word set to 0 or to -1.

The following gives a summary of steps you should perform to read a data log file:

- 1) If you want to begin reading a file from a particular record or from the first record, use either the Set-File-Position request with the desired record sequence number, or the Reset-File-Position request; otherwise use the Read-Record request.
- 2) Read the record data from the file transfer block. The file pointer will be moved to the next file record.
- 3) Repeat step 2 until all the file records are read, i.e., until the record status word indicates the end-of-file.

3.3 Using Status Event Exchanges and Event Logs

3.3.1 Configuring Event Exchanges

The Data server can provide recording binary events asserted by the connected devices and give a notification to the eXpertPower server.

The Data server can periodically poll device status registers and detect a change in the status bits to recognize asserted device events. A binary event mask specifies the critical event bits. Whenever the Data server detects a change in the status bits selected by the event mask, it records an event record to the circular Event log file. If the eXpertPower service is enabled in the ETC2002, this event will initiate a connection to the eXpertPower server.

Polling status events is performed via the status event exchanges. The Data server provides up to 250 event exchanges, each one of 1 register long, with a separate event mask. Table 3-6 shows the setup parameters that are used for configuring status event exchanges.

Table 3-6 Status Event Exchange Setup

| Description | Range |
|---|-----------------------|
| Exchange number | 0-249 |
| Device status register address | 0-65535 |
| Event mask | 0x00000000-0x0000FFFF |
| Device ID | 1-247 |
| Polling enabled | 1=enabled, 0=disabled |
| Options flags: Bit 0 – Auto reset enabled (1=enabled, 0=disabled) Bit 1 – Register type: 0=coils (Modbus request type 1) 1=holding register (Modbus request type 3) | |

The Data server supports the auto reset option you can use to clear a latched device status register after it has been read. In most devices, critical events are latched into a status register, so the new events are not generated until the register is explicitly cleared by a master application.

Since status registers may be implemented in the device either as coils, or as holding registers, you should specify a register type for each event exchange via the option flags.

3.3.2 Accessing the Event Log File

All event records are stored in a single circular event log file. Along with device status events, the file may contain the ETC2002 self-diagnostic events.

File transfers are performed through two blocks of registers: a master request block and a read-only file transfer block. The file transfer block occupies 12 registers in the address range of 10000-10011. Tables 3-7 and 3-8 show a layout of the event log request block and a layout of the event log record.

Table 3-7 Event Log Request Block

| Description | Range |
|---|--|
| Function | 3 = set file position 5 = reset file position 11 = read record |
| Record sequence number (with functions 3, 11) | 0-65535 |

Table 3-8 Event Log Record Layout

| Description | Range |
|------------------------------------|--|
| Event Log status | 0 = no new events, 1 = have new events, -1 = record corrupted |
| Event sequence number | 0-65535 |
| Event type ID | F2 (See Section 5) |
| Device ID | 0-247 |
| Event time, seconds since 1/1/1970 | F1 (See Section 5) |
| Event data | F3 (See Section 5) |

The event log file is accessed in the same manner as data log files. Refer to Section 3.2.3 for details.

3.4 Using Data Write Exchanges

The Data server can provide remote writing data to serviced devices upon user requests. Up to 250 write exchanges are supported with a size of 1 to 12 registers. Table 3-9 shows configuration parameters that can be defined for each write exchange.

Table 3-9 Write Exchange Setup

| Description | Range |
|--------------------------------|-----------------------|
| Exchange number | 0-249 |
| Start device register address | 0-65535 |
| Exchange block size, registers | 1-12 |
| Device ID | 1-247 |
| Write enabled | 1=enabled, 0=disabled |

Configured data exchanges are accessed via the data write exchange block. It occupies 14 registers in the address range of 9200 to 9213. Table 3-10 shows the write exchange layout.

Table 3-10 Write Exchange Block

| Description | Range |
|------------------|--|
| Exchange number | 0-249 |
| Write flag | 0 – Data has been written 1 – Data is waiting to be written |
| Data register 1 | |
| ... | |
| Data register 12 | |

To send data to a device via a write exchange, write data to the write exchange block with the write flag being set to 1. The write flag is automatically cleared after the data has been successfully transferred to the device. Always check this flag before writing new data to the exchange to ensure it is empty, otherwise previously written data may be lost.

4 Modbus Register Map

4.1 Data Server Registers

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|--------------------------------------|----------|---|--|-------|--------|-----|--|
| Real-time Data Exchange Block | | | | | | | |
| 0-9000 | | | | | | | |
| | | Data Exchange #1 | | | | | |
| ... | | ... | | | | | |
| | | Data Exchange #N | | | | | |
| | | Exchange Layout (120 registers maximum): | | | | | |
| +0,1 | | Record status | 0=No data, -1=corrupted record, 1-65534 = device response code | | UINT32 | R | Device response: function code or exception code (See 2.3-2.4) |
| +2,3 | | Record time, seconds since 1/1/1970 | F1 | sec | UINT32 | R | |
| +4,5 | | Record sequence number | 0-65535 | | UINT32 | R | |
| +6 | | Data register | 0-65535 | | UINT16 | R | |
| ... | | ... | | | | | |
| +119 | | Data register | 0-65535 | | UINT16 | R | |
| Data Write Exchange Block | | | | | | | |
| 9200-9213 | | | | | | | |
| +0 | | Exchange number | 0-249 | | UINT16 | R/W | Defines accessed exchange number |
| +1 | | Write flag | 0 – Data has been written 1 – Data is waiting to be written | | UINT16 | R/W | |
| +2 | | Data register | 0-65535 | | UINT16 | R/W | |
| ... | | ... | | | | | |
| +13 | | Data register | 0-65535 | | UINT16 | R/W | |
| Data Log Transfer Blocks | | | | | | | |
| 9120-9135 | | File Request Block | | | | | |
| +0 | | Function | 3 = set file position 5 = reset file position 11 = read file | | UINT16 | R/W | 3 - changes the file position 5 - sets the file position at the first (oldest) record 11 - opens the file for reading from the present file position |
| +1 | | Record sequence number (with functions 3, 11) | 0-65535 | | UINT16 | R/W | The record sequence number with function 11 does not change the file position |
| +2-12 | | Reserved | | | | | |
| +13 | | First (oldest) record sequence number | | | UINT16 | R | |
| +14 | | Next sequence number to be used | | | UINT16 | R | |
| +15 | | Data exchange number | 0-249 | | UINT16 | R/W | Write – sets new exchange number Read – returns exchange number |
| 9000-9119 | | File Transfer Block | | | | | |
| +0,1 | | Record status | 0= No new data, -1=corrupted | | UINT32 | R | Device response: function code or |

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|----------------------------------|----------|---|--|-------|--------|-----|--|
| | | | record, 1-65534 – device response code | | | | exception code (See 2.3-2.4) |
| +2,3 | | Record time, seconds since 1/1/1970 | F1 | sec | UINT32 | R | |
| +4,5 | | Record sequence number | 0-65535 | | UINT32 | R | |
| +6 | | Data register | 0-65535 | | UINT16 | R | |
| | | ... | | | | | |
| +119 | | Data register | 0-65535 | | UINT16 | R | |
| Event Log Transfer Blocks | | | | | | | |
| 10016-10031 | | File Request Block | | | | | |
| +0 | | Function | 3 = set file position 5 = reset file position 11 = read file | | UINT16 | R/W | 3 - changes the file position 5 - sets the file position at the first (oldest) record 11 - opens the file for reading from the present file position |
| +1 | | Record sequence number (with functions 3, 11) | 0-65535 | | UINT16 | R/W | The record sequence number with function 11 does not change the file position |
| +2-12 | | Reserved | | | | | |
| +13 | | First (oldest) record sequence number | | | UINT16 | R | |
| +14 | | Next sequence number to be used | | | UINT16 | R | |
| +15 | | Reserved | | | | | |
| 10000-10009 | | File Transfer Block | | | | | |
| +0,1 | | Event log status | 0 - no new events, 1 – data is valid, -1 – current record corrupted | | UINT32 | R | |
| +2,3 | | Event sequence number | 0-65535 | | UINT32 | R | |
| +4,5 | | Event type ID | F2 | | UINT32 | R | |
| +6,7 | | Device ID | 0-247 | | UINT32 | R | |
| +8,9 | | Event time, seconds since 1/1/1970 | F1 | sec | UINT32 | R | |
| +10,11 | | Event data | F3 | | UINT32 | R | |

4.2 Digital Inputs

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|---------|----------|------------------------|-----------------|-------|--------|-----|-------|
| | | Digital Inputs | | | | | |
| 12544 | | Digital inputs DI1-DI4 | 0x0000 - 0x000F | | UINT16 | R | |

The digital input register returns the state of the level-sensitive digital inputs in a 16-bit packed format.

4.3 Device Control and Status Registers

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|--|----------|--|--|-------|--------|-----|--|
| Device Mode Control Registers (factory set) | | | | | | | |
| 44134 | | ETC options | Bit0 = 1 - COM5 enabled Bit1 = 1 - IP-LINK enabled Bit2 = 1 - IRIG-B enabled | | UINT16 | R | The secondary master port may be either COM5, or IP-LINK enabled |
| 44135 | | Modem | 0 = disabled, 1 = enabled | | UINT16 | R | |
| 44136-44165 | | Reserved | | | UINT16 | | |
| 44166-44326 | | Reserved | | | UINT16 | | |
| Device Diagnostics Register (bit map) | | | | | | | |
| 44326-44327 | | Device self-diagnostics flags. When read: 0 = no faults logged, 1 = a fault bit has been set at least once since the last reset. When written: 0 = clear a fault bit, 1 = no effect. | F4 | | UINT32 | R/W | |
| 44328-44341 | | Reserved | | | | | |
| Device Authorization Register | | | | | | | |
| 44378-44379 | | When write: 8-digit password. When read: 0 = access permitted, -1 = authorization required. | 0 - 99999999 (write) 0/-1 (read) | | INT32 | R/W | |
| 44380-44385 | | Reserved | | | | | |
| 44386-45951 | | Reserved | | | | | |
| Factory Diagnostic Registers | | | | | | | |
| 45952-46079 | | Factory diagnostic registers | | | UINT32 | R | |

4.4 Device Setup Registers

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|---|----------|--------------------------------|----------------------|-------|--------|-----|------------------------|
| Factory Device Settings and Identification | | | | | | | |
| 46080-46207 | | | | | | | |
| +0, 1 | | Device serial number | | | UINT32 | R | |
| +2, 3 | | Device model ID | 2000, 2002 | | UINT32 | R | |
| +4-11 | | Device model name | "ETC2000", "ETC2002" | | CHAR16 | R | Null-terminated string |
| +12-19 | | Reserved | | | UINT16 | R | |
| +20 | | Device firmware version number | | | UINT16 | R | |
| +21 | | Device firmware build number | | | UINT16 | R | |
| +22,23 | | Reserved | | | | | |
| +24 | | Boot loader version number | | | UINT16 | R | |
| +25 | | Boot loader build number | | | UINT16 | R | |
| +26 | | RF firmware version number | | | UINT16 | R | |
| +27 | | RF firmware build number | | | UINT16 | R | |
| +28-95 | | Reserved | | | | | |
| +96 | | Ethernet MAC address 0-1 | 0x0500 | | UINT16 | R | |
| +97 | | Ethernet MAC address 2-3 | 0x00F0 | | UINT16 | R | |
| +98 | | Ethernet MAC address 4-5 | 0x0000-0xFFFF | | UINT16 | R | |

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|-------------------------------|----------|---|---|-------|--------|-----|---|
| +99-128 | | Reserved | | | | | |
| Device Options Setup | | | | | | | |
| 46256-46271 | | | | | | | |
| +0 | | Log output | Bit 0 = Dumping to console, Bit 2 = Server Log Bit 3 = XP log Bit 4 = IPLINK log Bit 5 = Data Server log Bit 6 = Sharing log | | UINT16 | R/W | |
| +1 | | Number of XP faults for reset | 0-200 | | | | 0 – never (50 - default) |
| +2 | | Global reset timer (in hours) | 0-24 | | | | 0 – never (24 - default) |
| +3-15 | | Reserved | | | UINT16 | R/W | |
| Time/Date Setup | | | | | | | |
| 46400-46415 | | | | | | | |
| +0 | | Local time offset, min | 0-+/-720 | | UINT16 | R/W | Offset in minutes from UTC (Universal Coordinated or Greenwich Mean time) |
| +1 | | Daylight savings time (DST) option | 0 = DST disabled (standard time only), 1 = DST enabled | µsec | UINT16 | R/W | |
| +2 | | DST start month | 1 - 12 | | UINT16 | R/W | |
| +3 | | DST start week of the month | 1 - 4 = 1st, 2nd, 3rd and 4th week, 5 = the last week of the month | | UINT16 | R/W | |
| +4 | | DST start weekday | 1-7 (1= Sun, 7 = Sat) | | UINT16 | R/W | |
| +5 | | DST end month | 1 - 12 | | UINT16 | R/W | |
| +6 | | DST end week of the month | 1 - 4 = 1st, 2nd, 3 rd and 4th week, 5 = the last week of the month | | UINT16 | R/W | |
| +7 | | DST end weekday | 1-7 (1= Sun, 7 = Sat) | | UINT16 | R/W | |
| +8 | | Clock synchronization source | 0 = GPS IRIG-B master clock | | UINT16 | R/W | |
| +9-15 | | Reserved | | | UINT16 | | |
| Clock Setup and Status | | | | | | | |
| 46416-46447 | | | | | | | |
| +0, 1 | | Local time, in seconds, since Jan 1, 1970 | F1 | sec | UINT32 | R | |
| +2, 3 | | Fractional seconds, µsec | | µsec | UINT32 | R | |
| +4 | | Fractional seconds, milliseconds | 0-999 | | UINT16 | R/W | |
| +5 | | Seconds | 0-59 | | UINT16 | R/W | |
| +6 | | Minutes | 0-59 | | UINT16 | R/W | |
| +7 | | Hour | 0-23 | | UINT16 | R/W | |
| +8 | | Day of month | 1-31 | | UINT16 | R/W | |
| +9 | | Month | 1-12 | | UINT16 | R/W | |
| +10 | | Year (calendar year minus 2000) | 0-99 | | UINT16 | R/W | |
| +11 | | Weekday | 1-7 (Sun = 1, Sat = 7) | | UINT16 | R | |
| +12 | | Daylight savings time status | 0 = standard time, 1 = daylight savings time is active | | UINT16 | R | |
| +13 | | IRIG-B status | 0, 1 = no signal, 2 = time unlocked, | | UINT16 | R | |

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|----------------------------------|----------|---|--|-------|--------|-----|------------------------|
| +14-31 | | Reserved | 3 = time locked | | UINT16 | | |
| Modem Setup | | | | | | | |
| 46448-46479 | | | | | | | |
| +0,1 | | Device IP Address | 0x01000000-0xFFFFFFFF | | UINT32 | R/W | |
| +2,3 | | Network subnet mask | 0x00000001-0xFFFFFFFF | | UINT32 | R/W | |
| +4,5 | | Network default gateway | 0x00000000-0xFFFFFFFF | | UINT32 | R/W | |
| +6,7 | | Number of dial attempts | 0-1000, 0-dial forever | | UINT32 | R/W | |
| +8,9 | | Connection timeout, sec | 0-9999 | | UINT32 | R/W | |
| +10,11 | | Delay between redials, sec | 0-9999 | | UINT32 | R/W | |
| +12,13 | | Idle connection timeout, sec | 0-9999 | | UINT32 | R/W | |
| +14,15 | | Number of rings before answer | 0-2 | | UINT32 | R/W | |
| +16-23 | | Modem init string | "ATZ0&F" | | CHAR16 | R/W | Null-terminated string |
| +24-31 | | Reserved | | | | | |
| Communication Ports Setup | | | | | | | |
| 46480-46575 | | | | | | | |
| +0-2 | | Reserved | | | | | |
| +3 | | Baud rate | 1=300, 2=600, 3=1200, 4=2400, 5=4800, 6=9600, 7=19200, 8=38400, 9=57600, 10=115200 | bps | UINT16 | R/W | |
| +4 | | Data format | 0 = 7 bits/even parity, 1 = 8 bits/no parity, 2 = 8 bits/even parity | | UINT16 | R/W | |
| +5 | | Port direction | 0 = master, 1 = slave (COM4 only) | | UINT16 | R/W | 0 - default |
| +6 | | Receive timeout (COM4 master and COM5 only) | 500-30000 | ms | UINT16 | R/W | 500 - default |
| +7-15 | | Reserved | | | | | |
| 46480-46495 | | COM3 Setup | | | | | |
| 46496-46511 | | COM4 Setup | | | | | |
| 46512-46527 | | COM5 Setup | | | | | |
| 46528-46575 | | Reserved | | | | | |
| Network Setup | | | | | | | |
| 46576-46703 | | | | | | | |
| +0, 1 | | Device IP Address | 0x01000000-0xFFFFFFFF | | UINT32 | R/W | |
| +2, 3 | | Network subnet mask | 0x00000001-0xFFFFFFFF | | UINT32 | R/W | |
| +4, 5 | | Network default gateway | 0x00000000-0xFFFFFFFF | | UINT32 | R/W | |
| +6, 7 | | Use DHCP | 0 = NO, 1 = YES | | UINT32 | R/W | |
| +8, 9 | | TCP service port | 502 = Modbus/TCP, 20000 = DNP3.0/TCP | | UINT32 | R/W | |
| +10,11 | | Primary DNS IP address | 0x00000000-0xFFFFFFFF | | UINT32 | R/W | |
| +12,13 | | Secondary DNS IP address | 0x00000000-0xFFFFFFFF | | UINT32 | R/W | |
| +14-31 | | Reserved | | | UINT16 | R/W | |
| RF Modem Setup | | | | | | | |
| 46608-46639 | | | | | | | |

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|---|----------|---|--|-------|--------|-----|------------------------|
| +0 | | MAC net ID | 0-255 | | UINT16 | R/W | |
| +1 | | MAC node ID | 1-255 | | UINT16 | R/W | |
| +2 | | Net cluster ID | 0-255 | | UINT16 | R/W | |
| +3 | | Net node ID | 1-255 | | UINT16 | R/W | |
| +4 | | Node type | 0=Master, 1=RN+, 2=RN-, 3=RFD | | UINT16 | R/W | |
| +5 | | RF channel | 0=907MHz, 1=910MHz, 2=912MHz, 3=915MHz, 4=917MHz, 5=920MHz, 6=922MHz | | UINT16 | R/W | |
| +6 | | RF send power | 0-255 (2=-20dBm, 9=-10dBm, 160=0dBm, 255=4dBm) | | UINT16 | R/W | |
| +7 | | Number of cluster child nodes (master only) | 0-255 | | UINT16 | R/W | |
| +8 | | Network cluster tree depth (master only) | 0-255 | | UINT16 | R/W | |
| +9 | | Network routing algorithm (master only) | 0=AODV (Ad-hoc on-demand vector), 1=cluster tree, 2=hybrid (cluster tree + AODV) | | UINT16 | R/W | |
| +10-31 | | Reserved | | | | | |
| 46640-46703 | | Reserved | | | | | |
| Password Setup | | | | | | | |
| 46704-46715 | | | | | | | |
| +0, 1 | | Communications password (8 digits) | 0 - 99999999 | | UINT32 | R/W | |
| +2 | | Password protection enabled | 0 = disabled, 1 = enabled | | UINT16 | R/W | |
| +3 | | Reserved | | | UINT16 | R/W | |
| +4-11 | | ETC2002 local Login name | "" | | CHAR16 | R/W | Null-terminated string |
| Expert Power Service Setup | | | | | | | |
| 46768-46783 | | | | | | | |
| +0,1 | | Expert Power server IP Address | 0x01000000-0xFFFFFFFF | | UINT32 | R/W | |
| +2,3 | | Expert Power server TCP service port | 0-65535 (5001 – default) | | UINT32 | R/W | |
| +4,5 | | Expert Power client enabled | 0=client disabled, 1=client enabled | | UINT32 | R/W | |
| +6,7 | | Expert Power client interval | 1- 99999 (0 = close client) | min | UINT32 | R/W | |
| +8,9 | | Expert Power client time to next session | 1- 99999 | min | UINT32 | R | |
| +10 | | Security Key | 0-7 (0 = disabled) | | UINT16 | W | |
| +11-15 | | Reserved | | | | | |
| Internet Service Provider (ISP) Accounts | | | | | | | |
| 46784-46879 | | | | | | | |
| +0-15 | | ISP telephone number | | | CHAR32 | R/W | |
| +16-31 | | Login name | | | CHAR32 | R/W | |
| +32-47 | | Login password | | | CHAR32 | R/W | |
| 46784-46831 | | Account No1 | | | | | |
| 46832-46879 | | Account No2 | | | | | |
| 46880-46927 | | Reserved | | | | | |
| Data Server Setup | | | | | | | |
| 46928-46943 | | Real-time Data Exchange Setup | | | | | |

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|----------------------|----------|------------------------------------|---|-------|--------|-----|---|
| +0 | | Exchange number | 0-249 | | UINT16 | R/W | Write an exchange number before reading the setup registers |
| +1 | | Reserved | | | | | |
| +2,3 | | Start device register address | 0-65535 | | UINT32 | R/W | |
| +4,5 | | Exchange block size, registers | 1-114 | | UINT32 | R/W | |
| +6 | | Device ID | 1-247 | | UINT16 | R/W | |
| +7 | | Polling enabled | 1=enabled, 0=disabled | | UINT16 | R/W | |
| +8 | | RTC update enabled (optional) | 1=enabled, 0=disabled | | UINT16 | R/W | |
| +9-15 | | Reserved | | | | | |
| 46944-46959 | | Status Event Exchange Setup | | | | | |
| +0 | | Exchange number | 0-249 | | UINT16 | R/W | Write an exchange number before reading the setup registers |
| +1 | | Reserved | | | UINT16 | | |
| +2,3 | | Device status register address | 0-65535 | | UINT32 | R/W | |
| +4,5 | | Event mask | 0x00000000-0x0000FFFF | | UINT32 | R/W | Lower 16 bits are used |
| +6 | | Device ID | 1-247 | | UINT16 | R/W | |
| +7 | | Polling enabled | 1=enabled, 0=disabled | | UINT16 | R/W | |
| +8 | | Options flags (bitmap) | Bit 0 – auto reset: 0=disabled 1=enabled Bit 1 – register type: 0=holding register (request 3) 1=coils (request 1) | | UINT16 | R/W | |
| +9-15 | | Reserved | | | | | |
| 46960-46975 | | Write Data Exchange Setup | | | | | |
| +0 | | Exchange number | 0-249 | | UINT16 | R/W | Write an exchange number before reading the setup registers |
| +1 | | Reserved | | | UINT16 | | |
| +2,3 | | Start device register address | 0-65535 | | UINT32 | R/W | |
| +4,5 | | Exchange block size, registers | 1-12 | | UINT32 | R/W | |
| +6 | | Device ID | 1-247 | | UINT16 | R/W | |
| +7 | | Write enabled | 1=enabled, 0=disabled | | UINT16 | R/W | |
| +8-15 | | Reserved | | | | | |
| 46976-46991 | | Data Server Options | | | | | |
| +0 | | Polling interval | 1-1000 | min | UINT16 | R/W | 1 min default |
| +1 | | Logging interval, polls | 0-1000, 0=logging disabled | | UINT16 | R/W | 1 - default |
| +2 | | Data Server Enabled | 1=enabled, 0=disabled | | UINT16 | R/W | 0- default |
| +3 | | Continuous polling | 1=enabled, 0=disabled | | UINT16 | R/W | 0- default |
| +4-15 | | Reserved | | | | | |
| Gateway Setup | | | | | | | |
| 46992-46999 | | | | | | | |
| +0 | | ETC2002 device address | 1-247 | | UINT16 | R/W | 99 default |
| +1 | | Gateway device address range start | 1-247 | | UINT16 | R/W | 1 default |
| +2 | | Gateway device address range end | 1-247 | | UINT16 | R/W | 247- default |

| Address | Point ID | Description | Options/Range | Units | Type | R/W | Notes |
|-----------------------------|----------|--|---|-------|--------|-----|------------------------------------|
| +3 | | Shared device access enabled ^{1, 2} | 0=enabled, 1=disabled | | UINT16 | R/W | 0 - default |
| +4 | | Temporarily override device access restrictions ¹ | 0=disabled, 1=enabled | | UINT16 | R/W | Active only within current session |
| +5 | | Routing table enabled ² | 0=disabled, 1=enabled | | UINT16 | R/W | 0 - default |
| +6-7 | | Reserved | | | | | |
| Device Routing Table | | | | | | | |
| 47000-47249 | | | | | | | |
| +0 | | Device 1 rout path | Bits 0-3 - destination port: 0=none, 1=Ethernet, 4=COM4, 5=COM5 Bits 4-7 - shared device access rule (see F5) ² Bits 8-15 - IP entry number in the Routing IP Table for the Ethernet port | | UINT16 | R/W | |
| +1 | | Device 2 rout path | | | UINT16 | R/W | |
| | | ... | | | | | |
| +246 | | Device 247 rout path | | | UINT16 | R/W | |
| +247-249 | | Reserved | | | | | |
| Routing IP Table | | | | | | | |
| 47250-47299 | | | | | | | |
| +0,1 | | IP Address 1 | 0x00000000-0xFFFFFFFF | | UINT32 | R/W | |
| +2,3 | | IP Address 2 | 0x00000000-0xFFFFFFFF | | UINT32 | R/W | |
| | | ... | | | | | |
| +30,31 | | IP Address 16 | 0x00000000-0xFFFFFFFF | | UINT32 | R/W | |
| +32-49 | | Reserved | | | | | |
| 47300-52149 | | Reserved | | | | | |

- 1 Shared device access allows several clients to simultaneously access device registers either for a read or a write. It is normally enabled, so different clients can poll data registers in the same device without collisions. However, simultaneous file accesses to the same device could destroy device file buffers so both clients may receive corrupted data. To prevent such collisions, the user can disable shared device accesses so the ETC2002 will reject any additional requests addressed to a device that is being accessed by another client. In this event, the ETC2002 will return an exception response with the error code 10 = "Gateway is busy" until the first client closes the session or points to a different device.
- 2 Starting with V21.5.4. If the routing table is enabled in the ETC2002 and shared device access is also enabled, then the ETC2002 manages shared device accesses according to the access rules defined in the routing table individually for each device.

5 Data Formats

| Format Code | Value | Description | Notes |
|----------------------|---|--|-------|
| Timestamp | | | |
| F1 | | Local time in a UNIX-style format. Represents the number of seconds since midnight (00:00:00), January 1, 1970. The time is valid after January 1, 2000. | |
| Event Type ID | | | |
| F2 | | IRIG-B Sync Event | |
| | 0x0800 | XP client event (See F3 for Event data contents) | |
| | | DI Event | |
| | 0x0A00 | Generic DI event (Event data contains a DI status value) | |
| | | Data Server Events | |
| | 0x1400-0x1463 | Status polling event (masked), low byte = exchange number. Event data contains the event status value | |
| | | Self-Check Diagnostics Event | |
| | 0x5D00 | Action/Cause (See F3 for Event data contents) | |
| | | Self-Update Event | |
| | 0x5E00 | RTC DST/Standard time update | |
| | | Run-time Error | |
| | 0x6000 | Run-time exception | |
| | Hardware Diagnostics Event | | |
| 0x6200 | Action/Cause (See F3 for Event data contents) | | |
| | External Event | | |
| 0x6300 | Action/Cause (See F3 for Event data contents) | | |
| Event Data | | | |
| F3 | 0 | Power down | |
| | 1 | Reserved | |
| | 2 | RAM/Data error | |
| | 3 | CPU Watchdog reset | |
| | 4 | Reserved | |
| | 5 | CPU exception | |
| | 6 | Reserved | |
| | 7 | Software Watchdog | |
| | 8 | Power up | |
| | 9 | Device reset | |
| | 10 | Configuration reset | |
| | 11 | RTC fault | |
| | 12 | Configuration fault | |
| | 13 | XP fault counter | |
| | 14 | Reset Timer expired | |
| | 15 | MASTER EEPROM fault | |
| | 16 | Event Log corrupted | |
| | 17 | Data Log corrupted | |
| | 18 | Data Server setup | |
| | 19 | Reserved | |
| | 20 | C Library error | |
| | 21 | RTOS Kernel fatal error | |
| | 22 | Task error | |
| | 24 | IRIG-B signal lost | |
| | 25 | IRIG-B time failed | |
| | 26 | IRIG-B time locked | |
| | 27 | XP session started | |
| | 28 | XP session completed successfully | |
| | 29 | XP session failed | |
| | Device Diagnostics (bitmap) | | |
| F4 | Bit 0 = 1 | Reserved | |
| | Bit 1 = 1 | Permanent fault | |
| | Bit 2 = 1 | RAM/Data error | |
| | Bit 3 = 1 | CPU Watchdog reset | |
| | Bit 4 | Reserved | |
| | Bit 5 = 1 | CPU exception | |
| | Bit 6 | Reserved | |
| | Bit 7 = 1 | Software watchdog | |
| | Bit 8 = 1 | Power down | |
| | Bit 9 = 1 | Device reset | |
| | Bit 10 = 1 | Configuration reset | |

| Format Code | Value | Description | Notes | |
|----------------------------|------------|--|--|--|
| | Bit 11 = 1 | RTC fault | | |
| | Bit 12 = 1 | Configuration fault | | |
| | Bit 13 | Reserved | | |
| | Bit 14 | Reserved | | |
| | Bit 15 = 1 | Master EEPROM fault | | |
| | Bit 16 | Reserved | | |
| | Bit 17 | Reserved | | |
| | Bit 18 | Reserved | | |
| | Bit 19 | Reserved | | |
| | Bit 20 = 1 | C Library error | | |
| | Bit 21 = 1 | RTOS Kernel error | | |
| | Bit 22 = 1 | Task error | | |
| | Bit 23 | Reserved | | |
| Shared Access Rules | | | | |
| F5 | 0 | No access limitation | | |
| | 1 | PM296 Family (PM171, PM172, PM295, PM296/RPM096) locked registers: 1120-1791 3916-4035 4608-5930 6144-6655 28160-28927 29440-32896 35456-35743 | | |
| | | 2 | SA300 Family (SA300, EM610, BFM136, PM174, PM175) locked registers: 3072-3259 63120-65151 | |
| | | | | |