MSOC AUTOMATION TECHNICAL GUIDE

TG 7.2.1.7-16

Version 1.6

02/02

CONTENTS

SECTION 5

TABLES

SECTION 3

SECTION 4

FIGURES

SECTION 2

SECTION 4

Section 1 - Introduction

With the introduction of a microprocessor based protective relay, today's relay protection engineer must be familiar with topics outside of traditional relaying schemes. It is intended that the production of this guide will enable the relay engineer to understand the principles of a microprocessor-based relay's inclusion in a substation automation project.

Substation automation is heavily dependent upon integration of the appropriate components to allow reporting of metering and event data. The foundation of a successful automation solution is thorough engineering of a communication system. The MicroShield Overcurrent Relay (MSOC) is the culmination of intensive design efforts, which combine protective relaying and communication capabilities at an economical price. Through the evolution of protective relays, it was decided that a special guide was needed to serve today's power automation specialist.

This guide is intended to give the reader an in-depth explanation of the communication interfaces available with the MicroShield Overcurrent Relay. Successful integration of microprocessor based relays like the MSOC depends on not just understanding the bits and bytes of a particular protocol. It is the inherent understanding and application of such esoteric topics as physical interfaces, real time control, manufacturer independent device integration, throughput vs. speed of communication, … which influences the success of an automation project.

In many cases the individual performing the SCADA integration is not a relay protection engineer. This guide departs from the standard type of relay manual in that each data type is explained and each bit and word is explained. Several application examples are given within each section. A description of each protocol command is illustrated for the benefit of the user. Appendices are included detailing application notes, which augment the text. An explanation of the product's physical interfaces and the connectivity required is explored in depth. Explanations of register's uses to increase overall throughput are also explored. Throughput is always an issue when the system is commissioned. Understanding ways to improve the system data update is explained.

Three steps are required to permit successful communication between devices:

1. Correct physical cable connection between devices (Section 2).

2. Correct configuration of port protocol and device parameters (Section 3).

3. Generation and interpretation of the protocol command strings (Section 4).

The following sections shall explore the following procedures in depth when establishing a communication automation system, utilizing the MSOC.

Throughout this guide, implementation tips are included to aid the user in quick setup of the unit and advice to aid in troubleshooting if a problem is encountered. Implementation tips are denoted by the symbol ÿ preceeding the Implementation Tip Text. This makes the tip easier to find in the guide's text.

Section 2 - MSOC Physical Layer Connectivity

The MSOC allows connectivity to a variety of devices through one of two physical interfaces. The physical interfaces available are RS 232 and RS 485. Only one of the two available interfaces may be enabled at one time. RS 232 allows direct point to point connectivity to a host device. RS 485 allows connectivity to a single or multiple devices. A DB 9 connector is available at the front of the unit. The DB 9 connector is the MSOC's RS 232 physical interface. Depending upon the protocol selected for the RS 232 port; it may be a port to which an address is assigned. One protocol, Modbus allows for assignment of an address if the protocol is mapped to the front RS 232 or rear RS 485 port. MSOC Native ASCII protocol does not allow for assignment of an address if the protocol is enabled for front port RS 232 operation. The MSOC Native ASCII protocol requires address vectoring if assigned to the rear RS 485 interface.

The RS 485 physical interface allows for attachment of multiple devices on a single communication cable. The RS 485 interface allows for connection of up to 31 MicroShield Overcurrent Relays on a single cable. Connection to the RS 485 interface is available the through three of the screw terminals located at the rear of the unit. If Modbus or Native MSOC ASCII protocol is selected, each command received through the port must have a device address imbedded within the communication request.

The MSOC has two protocols within the unit, Modbus ASCII and MSOC Native ASCII protocol. In both cases, the MSOC acts as a slave device in that it waits for a command to be received from a host device and transmits a response to the command. Each device on a multi-dropped network has a unique address. This is required so that a command may be interpreted by a single device when transmitted. If two devices have the same address, a communications collision will occur when both devices try to respond to the received command.

Physical Attachment of an RS 232 Device to the MSOC

The RS 232 port is an electronically isolated port. The port allows for up to 30 V isolation from its internal circuitry. It is recommended that the length of the RS 232 cable is no more than 10 ft and that it be shielded and that the shield be grounded at one end. This insures for reliable communications in a substation environment. RS 232 is a point to point network, that is, only two devices may be connected on an RS 232 Link. Appendix C of this guide describes the RS 232 interface in further detail. Establishment of communication is dependent on three steps, physical connection between the Host and slave device(s), configuration of the MSOC to accept the correct protocol through the correct port (RS 232 front port or RS 485 rear port), and host transmission of the protocol string to the receiving device.

The appropriate null modem cable RS 232 must be used for test. Figure 2-1 illustrates the correct RS 232 pinout of a null modem cable. Later sections describe the procedure to enable testing of the protocol selected for the front RS 232 port.

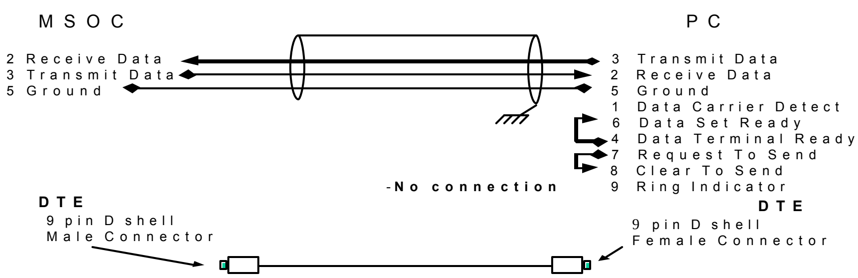


Figure 2-1. MSOC to PC Cable 9 to 9 Pin-Out

The RS 232 port does not offer handshaking. RS 232 handshaking is further explained in the application note in Appendix C. Figure 2-1 cable diagram does not require hardware jumpers to enable the RTS/CTS control. DSR and DTR does not require jumping on the MSOC to enable communications. The cable pin-out above does include jumpers for the host device in case the software program executing in the PC host requires hardware handshaking.

Figure 2-2 illustrates the physical connection between a PC executing Modbus Host Software and an MSOC. This is a point to point connection. It is a typical point to point communication architecture common to an RS 232 interface connection. For a more in-depth look at an RS 232 implementation, please refer to Appendix B of this document.

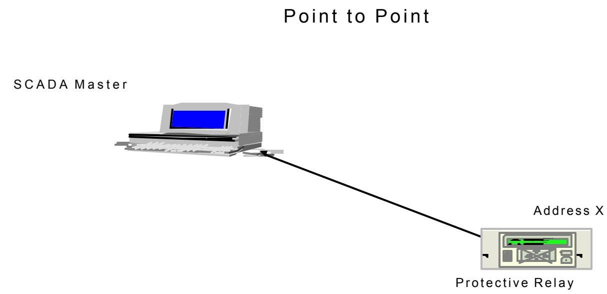


Figure 2-2. RS 232 Point to Point Configuration

Physical Attachment of an RS 485 Device to the MSOC

RS 485 is one of the more popular physical interfaces in use today. It was developed as an enhancement of the RS 422 physical interface. Its inherent strength is its ability to transmit a message over a twisted pair copper medium of 4000 feet in length. An RS 485 interface is able to transmit and receive a message over such a distance because it is a balanced interface. That is, it does not reference the signal to the system's electrical ground, as is the case in an RS 232 interface. RS 485 references the communication voltage levels to a pair of wires isolated from system ground. Depending on the manufacturer's implementation, isolation may be optical or electronic. RS 485 has two variants, two-wire and four-wire. In the two-wire format, communication occurs over one single wire pair. In four-wire format, communication occurs over two-wire pairs, transmit and receive. The two-wire format is the most common in use. The MSOC supports half duplex two-wire format only. The RS 485 port is also optically isolated to provide for 3000 V of isolation.

The RS 485 network supported and recommended by ABB requires the use of three conductor shielded cable. Suggested RS 485 cable and the respective manufacturer's wire numbers are:

- ALPHA 58902
- Belden 9729
- Belden 9829
- Carol 58902

ABB does not support deviations from the specified cables. The selected cable types listed are of the type, which have the appropriate physical and electrical characteristics for installation in substation environments.

A point to point connection is illustrated in Figure 2-4. Three wires, Positive (Terminal 9), Negative (Terminal 8) and Ground (Terminal 10). RS 485 requires a termination resistor at each end of the communication cable. The resistance shall be from 90 to 120 ohms. Additionally, depending upon the RS 485 physical interface converter used, a pull-up and pull-down resistor may be added to bias the line to decrease the amount of induced noise coupled onto the line when no communications are occurring. Internal to the MSOC are jumpers which when inserted in the proper position (as referenced in Figure 2-3), bias the line by inserting the proper pull-up, pull-down, and termination resistors.

To configure the Jumpers J6, J7, and J8, execute the following procedure:

- Face the front of the MSOC and loosen the two knurled screws at the front of the unit.
- Grasp the two handles at the front of the unit and pull it towards you. The MSOC has make before break contacts in the CT connectors. Powering down the unit need not be done when performing this step.
- Refer to Figure 2-3 illustrating the placement of J6, J7 and J8. J6 inserts a 120 ohm resistor between transmit and receive lines. J7 and J8 inserts a pull-up and pull-down resistor. The

value of J7 and J8 is 470 ohms. The IN position inserts the associated resistor in to the circuit. The OUT position removes the resistor from the circuit.

- Insert the MSOC unit into the chassis.
- Tighten the knurled screws at the front of the unit.
- It is advisable to place a sticker on the front of MSOC indicating that it is a terminated end of line unit.

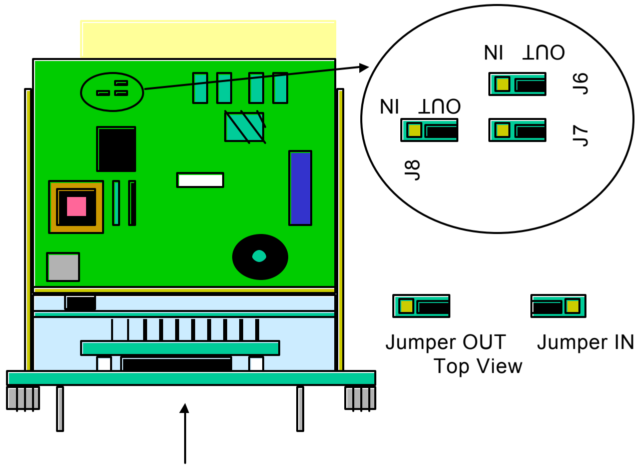


Figure 2-3. Location of RS 485 Resistor Configuration Jumpers in the MSOC

The following example illustrates an interconnection of the MSOC with a host device through a UNICOM physical interface connection using a 3-wire connection method. It should be noted that the RS 485 design on ABB relay products incorporates isolation. That is, the RS 485 ground is electrically isolated from the internal circuitry thereby assuring minimal interference from the extreme noise environments found in a substation. Care should be used when installing an RS 485 communication network. The recommended configuration must be followed as shown in Figure 2-3. Jumpers J6, J7, and J8 should be inserted to provide termination and pull-up at the MSOC end. Although not shown in Figure 2-4, a 120 ohm resistor should be inserted between the TX/RX + and TX/RX- pairs to provide for termination at the transmission end.

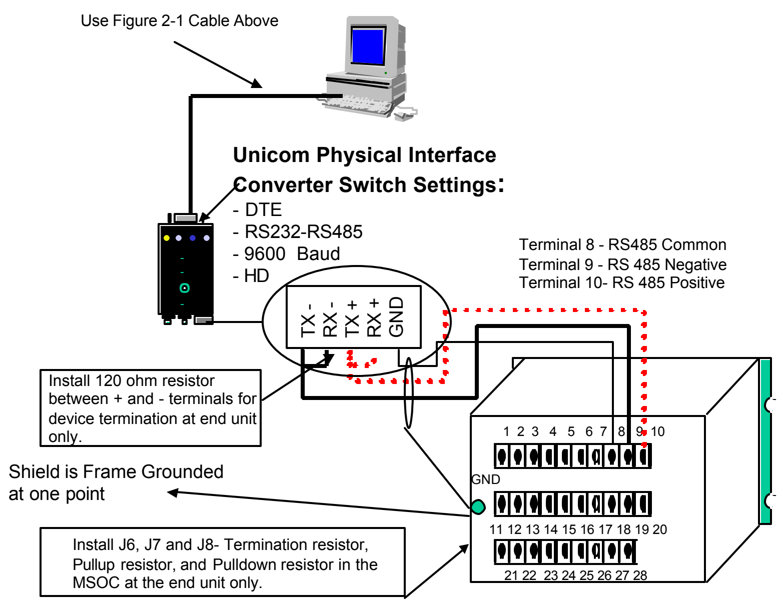


Figure 2-4. RS 485 Communication Device Connection

Figure 2-5 illustrates the connection of multiple devices on an RS 485 network using an ABB RS 232 to RS 485 physical interface converter. It should be stressed that the ABB converter works only at baud rates less than 9600. Additionally, the ABB converter does not offer handshaking and is a DCE RS 232 configuration node. A straight through cable can be used if the host does not require handshaking. It should be pointed out that the RS 485 portion of the diagram illustrates the three-wire connection of Positive, Negative and Ground. The shield is isolated at each point and tied to ground at one end only.

The physical configurations contained within the section for RS 485 must be followed implicitly. The diagrams are protocol independent. The ABB implementation of RS 485 uses an optically isolated port, which allows 3000 V isolation. Some vendors protocol converters will not work with isolated ports. For a more complete example of interconnecting multiple devices on an RS 485 network, please refer to the application note contained in Appendix C of this guide.

> ÿ **IMPLEMENTATION TIP–**Some manufacturer's RS 485 physical interface converters require that a physical switch or hardware jumper be inserted to allow the device to operate correctly with devices such as the MSOC which do not incorporate hardware port handshaking. The jumper allows RS 485 receiver transmitter control via sensing of the transmit data pin of the RS 232 interface. Consult the physical interface vendor's documentation for additional information.

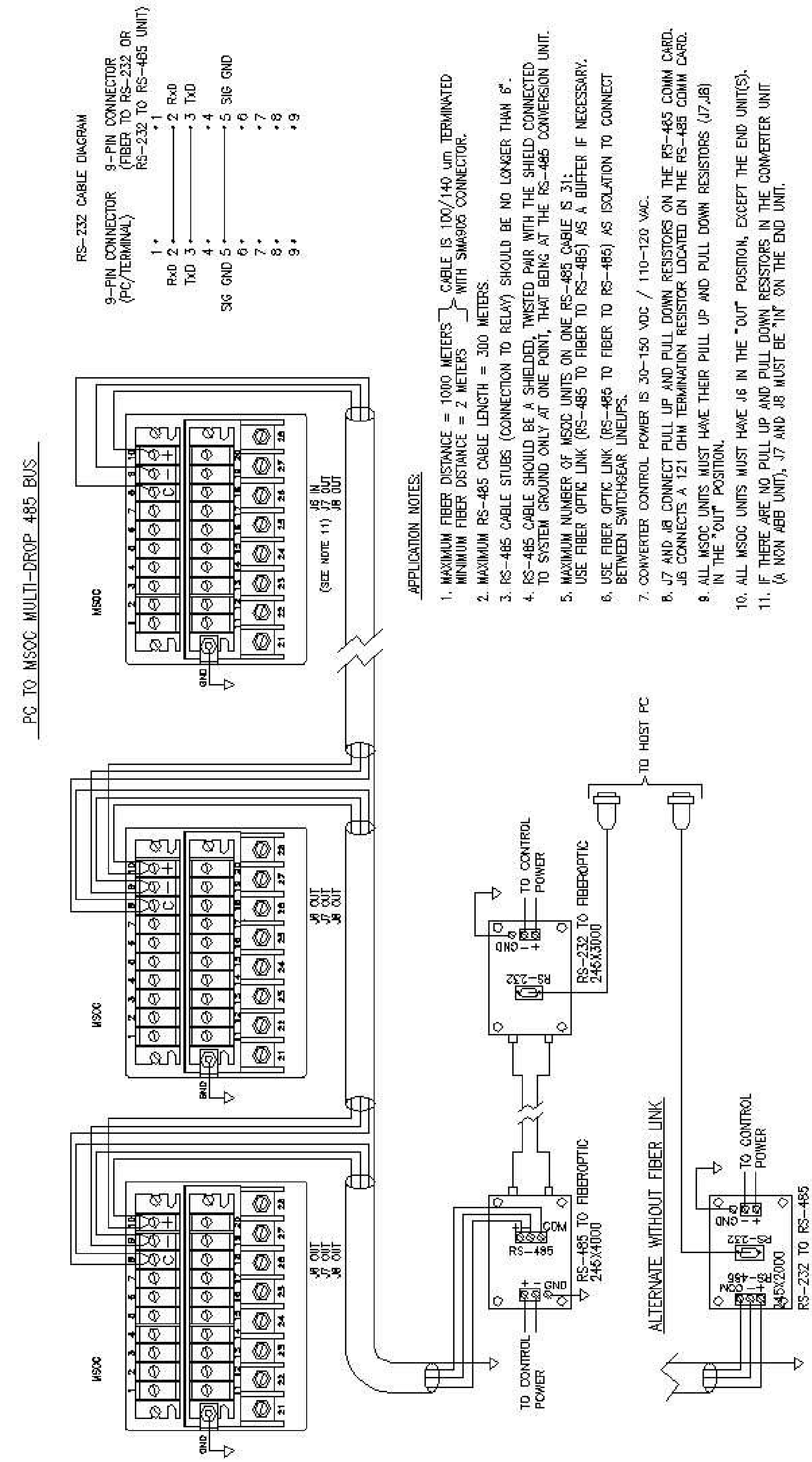


Figure 2-5. ABB Physical Interface Converter to MSOC

Section 3 - MSOC Protocol Assignment

The MSOC has a front panel interface, which allows the operator to assign a protocol and port parameters. The UP Arrow, DOWN Arrow, LEFT Arrow, RIGHT Arrow, <C> (Change) and <E> (Enter) keys permit selection and change of the selected parameters. The following subsections explain the configuration process and selection sequence for protocol assignment to each physical interface.

The MSOC is preloaded with Communication Port Factory Default Settings. Table 3-1 lists the default settings upon initial power-up.

Table 3-1. MSOC Communication Port Default Settings

Parameter	Rear RS 485 Port	Front RS 232 Port
Port Address	1	N/A
Port Protocol	ASCII	ASCII
Port Speed	9600	9600
Port Framing	8, None, 1	8, None, 1
Port Echo	N/A	OFF
Tx Delay Time	0 {disabled}	0 {disabled}

A description of MSOC port parameter selections are outlined in Table 3-2. Two parameters in Table 3-2 should be explained in further detail.

Local Echo is a parameter which when the MSOC receives the command, it automatically echo's the received command string back to the host. This is useful in using terminal emulators in ASCII mode, which require the echo to place the characters on the display screen for viewing by the operator. Although a misnomer, some may refer to this mode as "Half Duplex". If Local Echo is selected as a "YES" the command string will echo back to the originating host device. If Local Echo is selected as a "NO", the command string will not be echoed back to the host. Local Echo is important for MSOC Native ASCII Mode since the unit may be configured via a dumb terminal emulator. Modbus ASCII protocol must have this parameter selected with the "NO" option. Modbus is inherently an "echo less- full duplex" based protocol.

TX_DLY is found in the MSOC's Communication Menu. The MSOC is able to respond to a communication quickly (in some cases in less than 3 mS). Some older host units and terminal emulators do not have the ability to recognize or "turn around" their UART receivers that quickly. The host may miss the command response from the MSOC. To solve that problem, a configurable delay is present within the MSOC Communication Menu. The delay may be configured from 0 – 200 mS. If a delay is selected, the MSOC will insert a delay time of the selected duration before transmitting a response to a received command.

ÿ **IMPLEMENTATION TIP–** If the MSOC does not communicate to the Host after the wiring and communication parameters have been correctly configured and verified, increase the transmit delay time (DLY TX) in 10 mS increments. If the unit then communicates, the issue has been resolved in that the MSOC has responded much quicker than the Host. The Host in this case may have missed the MSOC response.

ÿ **IMPLEMENTATION TIP–** Some manufacturers RS 232 to RS 485 Converters may employ "active line biasing". If any device on an RS 485 network has this capability "do not install" J7 or J8 (pull up or pull down) resistors within an MSOC. The end unit(s) must still have a termination resistor installed (J6). Installation of pull-ups/downs with a device having active termination causes ground loops and can cause excessive communication errors on a network.

Table 3-2. Valid Parameter Selections Per Protocol

	Protocol Selected	Modbus ASCII	MSOC Native ASCII
	Port Address	1-247 (1- F7 HEX) RS 232 or RS 485 Ports	1-255 (01-FF HEX) RS 485 Port ONLY
	Baud Rate	1200, 2400, 4800, 9600, 19200	1200, 2400, 4800, 9600, 19200
	Frame Selection	Odd Parity, 7 Data Bits, One Stop Bit Odd Parity , 7 Data Bits, Two Stop Bits Even Parity, 7 Data Bits, One Stop Bit Even Parity, 7 Data Bits, Two Stop Bits	Odd Parity, 7 Data Bits, One Stop Bit Odd Parity , 7 Data Bits, Two Stop Bits Even Parity, 7 Data Bits, One Stop Bit Even Parity, 7 Data Bits, Two Stop Bits No Parity, 8 Data Bits, One Stop Bit
	Local Echo	NO	NO = Fast ASCII Mode YES = Terminal ASCII Mode
$$	TX_DLY	Transmit Delay = 0 – 200 in 5 mS increments.	Transmit Delay = 0 – 200 in 5 mS increments.
		$$ = Added in Version 1.6 Software	

Mapping Modbus Protocol to the RS 232 and RS 485 Ports

Modbus is a protocol invented by Modicon in the early 1970's. It has become the device standard protocol by many utilities. Modbus has many advantages that make it suitable for substation automation. The specifics of Modbus protocol emulation shall be explained in depth in later sections of this guide.

If the MSOC is attached to a PC using the cable illustrated in Figure 2-1 in the architecture shown in Figure 2-2, the next step is to configure the unit for the protocol used. The front and rear ports may be configured with different parameters, but only one port may be active at one time.

The following configuration sequence shall enable the RS 232 port and map the MSOC to communicate Modbus ASCII emulation with the following interface parameters: 9600 baud, 7 data bits, odd parity, two stop bits. The following setup sequence is as follows:

Configure the relay front port through the operator keypad as follows:

A. Apply power to begin the process to enable the RS 232 port.
B. Depress E
C. Main Menu heading should be visible.
D. Depress down arrow three times.
E. Operations Menu heading should now be visible.
F. Depress E.
G. Depress the down arrow eight times. Depress E.
H. Enter the Password of the Unit.
I. Depress E.
J. If Enable FP is shown depress the right arrow for YES. If Disable FP is shown then the front panel is already enabled. If this is the case, depress the left arrow button then depress C.
K. Depress E.
L. The Operations menu will be visible. Depress the C pushbutton to go to the Main Menu.
M. Depress the up arrow button twice to attain the Settings Menu selection.
N. Depress E. The Settings Menu shall be visible.
O. Depress the down arrow once. The Change Settings will be visible.
P. Depress E.
Q. Depress the down arrow 6 times. The Comm selection shall be visible.
R. Depress the E button. The password selection shall be visible. Enter the correct password for the unit. Depress E.

S. Select the appropriate address for the unit. For this instance it shall be 1. To change–depress E, right arrow to the correct value, depress E and then depress C.
T. Depress the down arrow once. FP Baud rate shall be visible. Select 9600. To change–depress E, right arrow to the correct value, depress E and then depress C.
U. Depress the down arrow once. FP Frm shall be visible. Select O,7,2. This sets the port to 7 data bits, Odd Parity, and two stop bits. To change–depress E, right arrow to the correct value, depress E and then depress C.
V. Depress the down arrow once. FP Prot shall be visible. Select Modbus. This enables the Modbus protocol resident in the unit. To change–depress E, right arrow to the correct value, depress E and then depress C.
W. Depress the down arrow once. RP Baud rate shall be visible. Select 9600. To change–depress E, right arrow to the correct value, depress E and then depress C.
X. Depress the down arrow once. RP Frm shall be visible. Select O,7,2. This sets the port to 7 data bits, Odd Parity, and two stop bits. To change–depress E, right arrow to the correct value, depress E and then depress C.
Y. Depress the down arrow once. RP Prot shall be visible. Select Modbus. This enables the Modbus protocol resident in the unit. To change–depress E, right arrow to the correct value, depress E and then depress C
Z. Depress the down arrow once. Local Echo shall be visible. Select OFF. These selections shall place the unit in the native Modbus protocol mode. To change–depress E, right arrow to the correct value, depress E and then depress C.
AA. After each of the settings, perform a unit save to maintain the unit settings entered in this session.

The above process has configured the front (RS 232) and the rear (RS 485) ports for Modbus communication. The RS 232 port has been enabled for communication. Sequence steps A through L enable the front port for communication. It should be noted that both ports need not be configured for the same protocol.

Mapping MSOC Native ASCII Protocol

Modbus Protocol is an industry standard in the industrial sector. It has also been widely accepted in the Utility sector. Prior to the acceptance of a de-facto protocol, each manufacturer created its own protocol. A universal method of communication was to send a command string to a unit in an ASCII format. An ASCII formatted protocol was developed by ABB for the MSOC. It can be used for network connectivity as is explained in future sections of this document. The protocol developed by ABB to be interpreted by the MSOC shall be referred to as the MSOC Native ASCII Protocol. This section shall describe in detail the method to enable the RS 232 port and map the MSOC Native Protocol to the both the RS 232 and RS 485 ports.

To enable the MSOC Native ASCII Protocol for 8 data bits, 1 stop bit, no parity, the configuration sequence to be executed by the user is as follows:

A. Depress E
B. Main Menu heading should be visible.
C. Depress down arrow three times.
D. Operations Menu heading should now be visible.
E. Depress E.
F. Depress the down arrow six times.
G. Enter the Password of the Unit. It is assumed that the factory default is the password–depress E.
H. Depress E.
I. If Enable FP is shown Depress the right arrow for YES and Depress E. If Disable FP is shown then the front panel is already enabled. If this is the case, depress the left arrow button and depress C.
J. Depress E.
K. The Operations menu will be visible. Depress the C button to go to the Main Menu.
L. Depress the up arrow button twice to attain the Settings Menu selection.

M. Depress E. The Settings Menu shall be visible.
N. Depress the down arrow once. The Change Settings will be visible.
O. Depress E.
P. Depress the down arrow 6 times. The Comm selection shall be visible.
Q. Depress the E button. The password selection shall be visible. Enter the correct password for the unit. Then Depress E to accept the password.
R. Select the appropriate address for the unit. For this instance it shall be 1. To change the address–press E, right arrow to the proper value, depress E then depress C.
S. Depress the down arrow once. FP Baud rate shall be visible. Select 9600. To change the Baud Rate– press E, right arrow to the proper value, depress E then depress C.
T. Depress the down arrow once. FP Frm shall be visible. Select 8N1. This sets the port to 8 data bits, No Parity, and one stop bit. To change the Parity–press E, right arrow to the proper value, depress E then depress C.
U. Depress the down arrow once. FP Prot shall be visible. Select ASCII. This enables the ASCII protocol resident in the unit. To change the protocol– press E, right arrow to the proper value, depress E then depress C.
V. Depress the down arrow once. Local Echo shall be visible. Select OFF. To change the Local Echo–press E, right arrow to the proper value, depress E then depress C.These selections shall place the unit in the native ASCII protocol mode.
W. After each of the settings, perform a unit save to maintain the unit settings entered in this session. Depressing the C button at the main menu shall allow a setting query to be sent to the operator. Depress the right arrow key and depress the E button to save the settings.

The functions to test connectivity shall be described in future sections within this guide.

Directing Communications Via the Front (RS 232) or Rear (RS 485) Port

As stated previously, only one port may be enabled on the MSOC at one time. Communications may be directed through the front RS 232 port or the rear RS 485 port. To direct the communications from the front RS 232 port to the rear RS 485 port the following procedure must be followed in addition to the aforementioned procedure for the native ASCII and the Modbus ASCII protocols.

A. Depress E.
B. Depress the down arrow 4 times to display the Operations Menu.
C. Depress E.
D. Depress the down arrow 8 times to display the selection Enable FP COM.
E. Depress E.
F. Enter the Password as described in the previous illustrations.
G. The user shall be prompted, with the selection Enable FP < >. If the selection is YES, then the Front Port (RS 232) shall have the communications options configured via the above process. If the Selection is NO then the Rear Port (RS 485) shall be enabled with the protocol configured via the above process. Enter the correct password for the unit. Then depress E to accept the password.

The above procedure has enabled the RS 485 port and disabled the RS 232 port. Selection of the Enable FP<> selection allows for the direction of the protocols selected.

NOTE: If the MSOC has its supply power removed (unit powered down) and then re-attached (unit powered up), the MSOC will Disable the front (RS 232) ports and Enable the rear (RS 485) port. If the MSOC is unit reset via the front MMI display panel (by simultaneously depressing the “C” “E” and “Up Arrow” keys), the MSOC will also Disable the front (RS 232) port and Enable the rear (RS 485) port. If the operator requires communication through the front panel port, the above procedure must be followed to re-initialize, the RS 232 port).

Section 4 - MSOC Protocol Description

If the cable connections and front panel configuration procedures are followed implicitly, the MSOC will communicate with a host computer capable of transmitting and receiving the selected protocol strings. The following sections will describe the commands for Modbus ASCII and MSOC Native ASCII protocols.

Modbus Command Set

Modbus is available in two emulations, Modbus RTU and Modbus ASCII. Modbus RTU is a bit oriented protocol (normally referred to as Synchronous), and Modbus ASCII is a byte-oriented protocol (normally referred to as Asynchronous). Both emulations support the same command set. **Networked nodes cannot communicate unless the same emulation of the Modbus protocol is interpreted.** This is an extremely important issue. The MSOC only supports the Modbus ASCII protocol. Modbus RTU emulation hosts cannot directly communicate to an MSOC.

Modbus ASCII Protocol

Modbus operates in the following fashion. A host device transmits a command, and one of the attached device(s) responds. Each device has a unique address assigned to it. Each device is configured for the same protocol emulation of Modbus. Figures 4-1 and 4-2 illustrate the polling sequence.

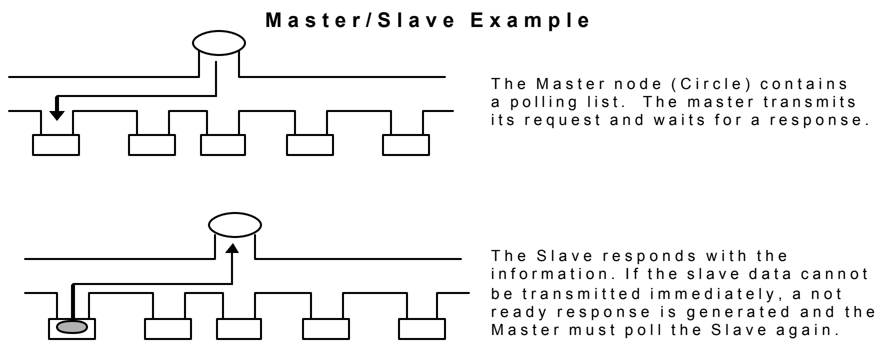
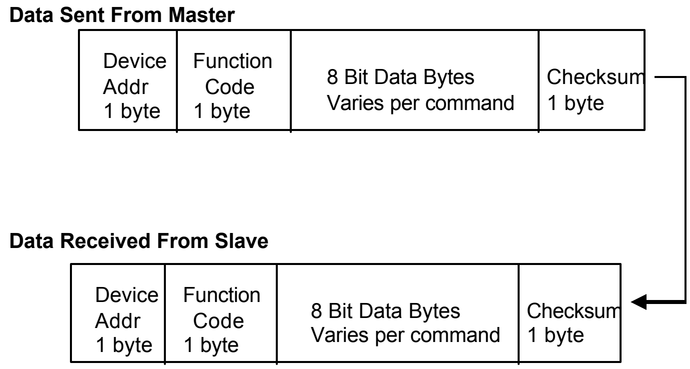


Figure 4-1. Modbus Polling Sequence

An ASCII character is defined as 7 data bits. A character is represented as a number from 00 HEX to 7F HEX. Appendix A contains a HEX to ASCII character conversion chart. If a 0 is transmitted, it must be encoded into a HEX representation to be interpreted by the receiving device. 0 ASCII is 30 HEX. The frame format for Modbus is represented in Figure 4-2. The device address, function code and checksum is part of the transmitted frame. The Checksum is a Longitudinal Redundancy Check (LRC). Its calculation shall be described later in this guide.

The generic Modbus Frame is analyzed in Figure 4-3. The start of an ASCII frame is always a colon (: = 3A HEX) and a termination of the command is a line feed and carriage return (lf cr = 0D 0A). The format is the same for the host transmitting the frame and the slave node responding to the host's transmission. The device address is imbedded within the frame along with the Modbus command function code. A checksum is appended to the entire command.



(Device Address = 0 (Null Command), 1 - 247, 255 (Broadcast)

Figure 4-2. Modbus ASCII Transmitted and Received Frame Formats

START	**ADDRESS**	**FUNCTION**	**DATA**	**LRC**	**END**
1 Char :	2 Chars	2 Chars	N Chars	2 Chars	2 Chars CR LF

Figure 4-3. Modbus ASCII Frame Format

The Modbus characters are encoded with a variety of frame sizes. An analysis of each frame is illustrated in Figure 4-4. When selecting a common frame size, (as explained in the configuration setup examples), parity, word length, and stop bits are selected to form a 10 bit data frame (1 start bit + 7 data bits + 1 stop bit + 1 Parity bit "OR" 1 start bit + 2 stop bits + 7 data bits + NO Parity = 10 bits per frame). It is important to note this distinction since if MSOC device attachment is to occur through a device, the device must support 10-bit asynchronous data framing.

Least Significant Bit ……………………….….Most Significant Bit

START	1	2	3	4	5	6	7	PARITY	STOP

With Parity Checking

START	1	2	3	4	5	6	7	STOP	STOP

Without Parity Checking

Figure 4-4. Modbus ASCII Frame Analysis

The MSOC offers a variety of frame sizes. If the frame size, 8N1 is selected (8 Data Bits, No Parity, 1 Stop Bit), then an additional stop bit is inserted. The frame format follows that of Figure 4-4 " Without Parity Checking".

ÿ **IMPLEMENTATION TIP-**When Commissioning a Modbus system, it is always advisable to connect a communication analyzer in-line with the Host. It is always uncertain whether the Host is sending the command correctly. Within the MSOC, an incorrect address request will always generate an exception response from the relay. If an exception response is generated, many Host devices will not display the Modbus exception response generated by the unit. A communication analyzer allows for rapid troubleshooting of a malfunctioning network connection.

The MSOC emulates a slave device. The following Modbus Commands are supported within the unit:

- 01 – Read 0X Coil Status
- 02 – Read 1X Contact Status
- 03 – Read 4X Holding Registers
- 16 – Write 4X Holding Registers
- 23 – Write 4X and Read 4X Holding Registers
- 20 – Read 6X Extended Registers
- 21 – Write 6X Extended Registers

Any other Modbus command sent to the MSOC shall result in a Modbus exception code being sent to the transmitting device. The following sections will further describe the Modbus functionality within the MSOC.

ÿ **IMPLEMENTATION TIP-**Although the MSOC allows configuration of Modbus for a frame of N-8-1, some implementations will interpret this emulation of Modbus to be RTU mode. The MSOC does not support this mode. It is advisable to contact the manufacturer of the Host and Host software to determine the interpretation of the command string. For example, the Modicon XMIT and COMM block allowing the PLC to emulate a Host device only allows block frame size designation of 7 data bits.

0X Discrete Coils

Discrete Modbus Coil status is available via a Function 01 request via Modbus. Figure 4-5 illustrates a typical command sequence. The Host polls the MSOC for the Data. The MSOC receives the request and responds with the expected data. The Host then interprets the command response, checks the LRC checksum in ASCII mode and then displays the interpreted data.

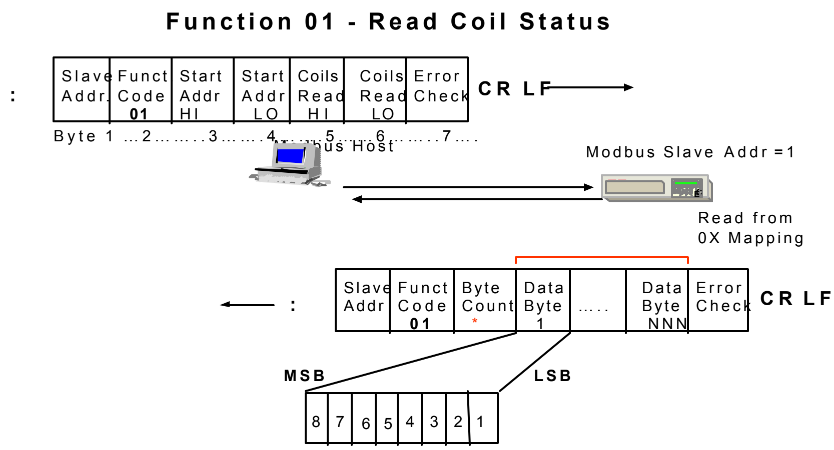


Figure 4-5. Modbus Protocol Function 01 Frame Format

Function Code 1 (Read Coil Status) – Read Only Data

The 0X read command allows for access of Logical and Physical Input data. The information listed in Tables 3-1, 3-2, and 4-1 is that which is reported in real time. In other words, if the bits are polled as per the table, the status of each data bit is reported at the time the data is requested. If the data is momentary in nature, then access of status is dependent upon reading the information at the time the function or signal is present.

Table 4-2 lists the Latched Data reported by the MSOC. **This data is only available in 0X coil access. It is not available in 1X or 4X data format types.**

Momentary data reporting is not available at the present time. The data within the 0X discretes are also available in 4X memory for those hosts, which are not able to access this data type.

If data is requested from memory addresses not defined within this document, a Modbus Exception Code shall be generated.

Figures 4-6 and 4-7 illustrate a simple example of a host requesting data from an MSOC relay where Physical Relay Coil Status is requested of the MSOC. The data is requested in the Modbus ASCII frame format illustrated in Figure 4-6. Raw Data received by the host is decoded from ASCII to HEX. The individual status bits are parsed by the host as illustrated in Figure 4-7.

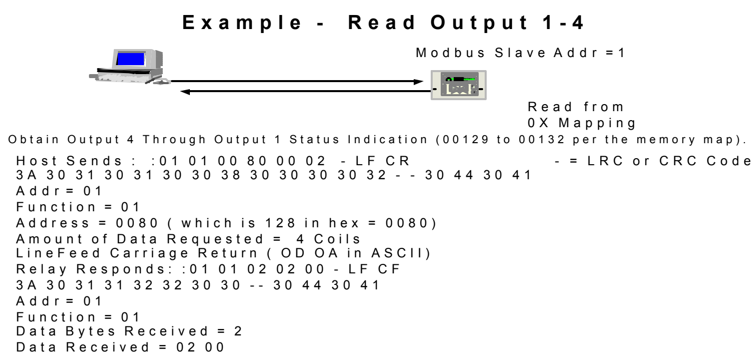


Figure 4-6. Example Transaction Request for Four Coils

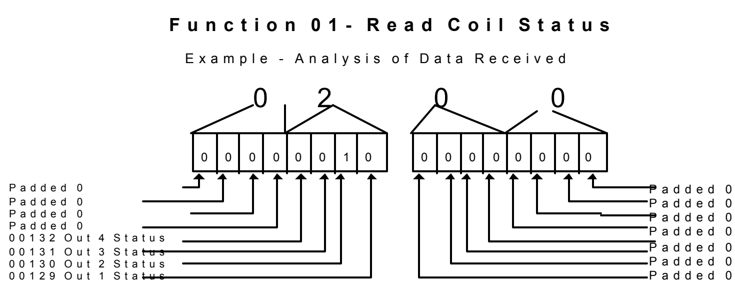


Figure 4-7. Example of Raw Data Decode

Logical Outputs – 14 Discrete Coils (14 Elements Defined)

Relay Element Status as described in Table 4-1. Additional coil status has been added in the latest version of MSOC executive firmware. Consult the symbol keys in the table for revision level feature inclusion.

The status information reported in Table 4-1 is reported as real time status bits. For example, if the breaker is the process of tripping the status of 0009 and 00010 shall report the breaker action. If the status of 00009 or 00010 is polled after the trip or close has been completed, the status of these bits shall report a status of 0.

ÿ **IMPLEMENTATION TIP-**If breaker status is to be required after the event occurs, it is advisable to read the seal in bits described in Table 4-3 (Address 00255 and 00256). Seal in bit capability is available in MSOC firmware Version 1.70. If one has an earlier version of MSOC, Address 10006 and 10007 could be polled to monitor the actual state of 52a and 52b. The polled state may then be saved and compared to the earlier values to determine whether a trip or close has occurred between device polls.

The PUA (Pick Up Alarm, address 00006), status follows the status of the PUA LED located on the front panel of the MSOC. PUA status reports the actual state of the LED.

The FAIL bit indicates a status of 1 when the relay has verified that its diagnostics have passed. If the relay's internal diagnostics fail, bit 0007 status will be reported as a 0. Modbus Registers 40130 and 40131 may be polled to indicate the precise cause of the diagnostic failure.

All other bits in Table 4-1 shall report a status of 1 when its associated function is active.

Table 4-1. Logical Output Modbus Address Map Definition

	Register Address	Item	Description
	00001	51LT	Phase Time Overcurrent
	00002	51P	Phase Time Overcurrent
	00003	50P	Phase Instantaneous Overcurrent
	00004	51N	Ground Time Overcurrent
	00005	50N	Ground Instantaneous Overcurrent
	00006	PUA	Timeout or Trip Pickup Alarm
	00007	FAIL	MSOC Diagnostic Failure
	00008	CBFAIL	Circuit Breaker Fail Alarm
	00009	TRIP	Breaker is Tripping
	00010	CLOSE	Breaker is Closing
$$	00011	RCLINPROG	Recloser In Progress
$$	00012	RCLALARM	Recloser Lockout Alarm
$$	00013	RDA	Recloser Disable Alarm
$$	00014	RCLRMA	Recloser Max Recl Alarm
			$$ = Added in Version 1.6 Software

Physical Outputs (4 Elements Defined)

Output status is described in Table 4-2. If the MSOC does not have voltage inputs, additional output coil status is available to the user. The state of the addresses 00129 through 00132 follow the state of the physical output hardware contacts located at the rear screw terminals of the relay.

Table 4-2. Physical Output Modbus Address Map Definition

Register Address	Item	Description
00129	OUT1	Physical Relay Output Contact Status (Terminal 17 & 18)
00130	OUT2	Physical Relay Output Contact Status (Terminal 15 & 16)
00131	OUT3	Available if no Voltage Inputs – Physical Relay Output Contact Status (Terminal 13 & 14)
00132	OUT4	Available if no Voltage Inputs – Physical Relay Output Contact Status (Terminal 11 & 12)

Seal-In Outputs (14 Elements Defined)

This block of elements is the same as block 00001 through 00014. Elements within this block are latched, whereas elements in 00001 through 00014 are unlatched real time elements. If one of the elements 00257 through 00270 is set, a transition has occurred and is latched in as status. These bits are reset via Mask Control Block manipulation as listed in Table 4-25 of this document. Bit 9 of Word 41669 controls the reset of discrete Seal in Outputs 00257 through 00270.

Table 4-3. Seal In Output Modbus Address Map Definition

	Register Address	Item	Description
**	00257	51LT*	Phase Time Overcurrent
**	00258	51P*	Phase Time Overcurrent
**	00259	50P*	Phase Instantaneous Overcurrent
**	00260	51N*	Ground Time Overcurrent
**	00261	50N*	Ground Instantaneous Overcurrent
**	00262	PUA*	Timeout or Trip Pickup Alarm
**	00263	FAIL*	MSOC Diagnostic Failure
**	00264	CBFAIL*	Circuit Breaker Fail Alarm
**	00265	TRIP*	Breaker is Tripping
**	00266	CLOSE*	Breaker is Closing
**	00267	RCLINPROG* (Recloser In Progress)	Recloser In Progress
**	00268	RCLALARM* (Recloser Lockout Alarm)	Recloser Lockout Alarm
**	00269	RDA* (Recloser Disable Alarm)	Recloser Disable Alarm
**	00270	RCLRMA* (Recloser Max Recl Alarm)	Recloser Max Recl Alarm
		** = Added in Version 1.7 Software	* = Latched Signal State

1X Discrete Contact Input

Discrete physical input and relay element status are available via a Function 02 request through Modbus. Figure 4-8 illustrates a typical command sequence. The Host polls the MSOC for the Data. The MSOC receives the request and responds with the expected data. The Host then interprets the command response, checks the LRC checksum in ASCII mode and then displays the interpreted data. The same information is available through a 4X Register read command, which allows a host without 1X data access capabilities to obtain physical input and relay element information. Tables 4-4 and 4-5 list the 1X Discrete contact memory map as defined in the Modbus ASCII protocol memory map.

Function Code 2 - Read Input Status (Read Only Data)

Figure 4-8 illustrates the command format required for execution of Function Code 2.

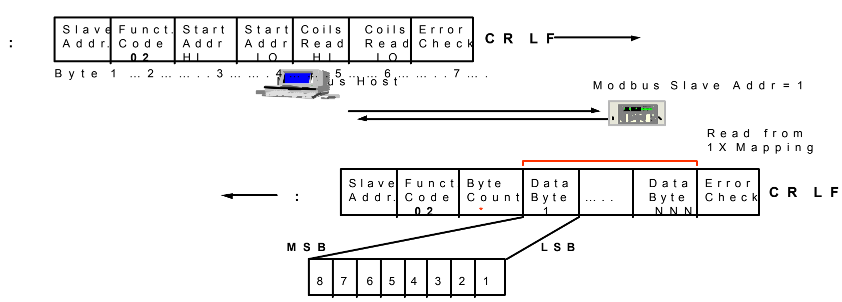


Figure 4-8. 1X Input Request Using Modbus Command 02

It should be noted that every MSOC allows for real time status reporting when the unit is polled. If a status is momentary and is missed during the host poll, then the data is lost. Momentary data reporting is not available. It should also be noted that data requested from 1X data address ranges not defined within this document should generate Modbus exception codes.

Logical Inputs (14 Elements Defined)

This section of relay information allows access of relay element data. Some of the status bit information reported in 1X discrete is available as 0X Register definition table. All of the individual information is available in the 4X Register definition table (Modbus Function Code 03). Table 4-4 lists the discrete point address assignment for physical inputs and control elements within the MSOC.

52a and 52b status may be mapped and wired to the Physical Inputs provided at the rear of the MSOC. However, MSOC does not have these contacts wired to the MSOC physical input, status of 52a/52b, shall be reported as a function of sensed current flow. If 52a and 52b are unmapped, the MSOC shall report breaker status as such:

A closed breaker is determined when the input current is above 5% of the neutral nominal current rating (refer to Table 4-15) of the relay. An open breaker (cleared breaker) is determined when the sensed current is above 5% of the neutral nominal current rating.

Table 4-4. Logical Input Modbus Address Map Definition

	Register Address	Item	Description
	10001	51LT	Phase Time Overcurrent
	10002	51P	Phase Time Overcurrent
	10003	50P	Phase Instantaneous Overcurrent
	10004	51N	Ground Time Overcurrent
	10005	50N	Ground Instantaneous Overcurrent
	10006	52a	Relay Status (52a = 1, 52b = 0 CB Open)
	10007	52b	Relay Status (52a = 0, 52b = 1 CB Closed)
	10008	TRIP	Relay Present State is Tripping
	10009	CLOSE	Relay Present State is Closing
	10010	43a	Enable Reclose Function Asserted
$$	10011	EXTRI	External Reclose Initiate Asserted
$$	10012	ZSC	Zone Sequence Closing Function State (79)
$$	10013	DTL	Drive To Lockout Asserted
**	10014	RSI	Reset Seal Ins
			$$ = Added in Version 1.6 Software ** = Added in Version 1.7 Software

Physical Inputs (2 Elements Defined)

The two physical inputs are mappable for various functional inputs. Their status is available at the following addresses as illustrated in Table 4-5.

Table 4-5. Physical Input Modbus Address Map Definition

Register Address	Item	Description
10129	IN1	Input Status (Terminal 5 & 4 [common])
10130	IN2	Input Status (Terminal 6 & 4 [common])

Application Example: Obtain the Relay Status from MSOC Address 1. The relay status is available from input 10006 and 10007. Figures 4-9 and 4-10 illustrate the polling sequence and raw data returned over the network utilizing Function Code 02.

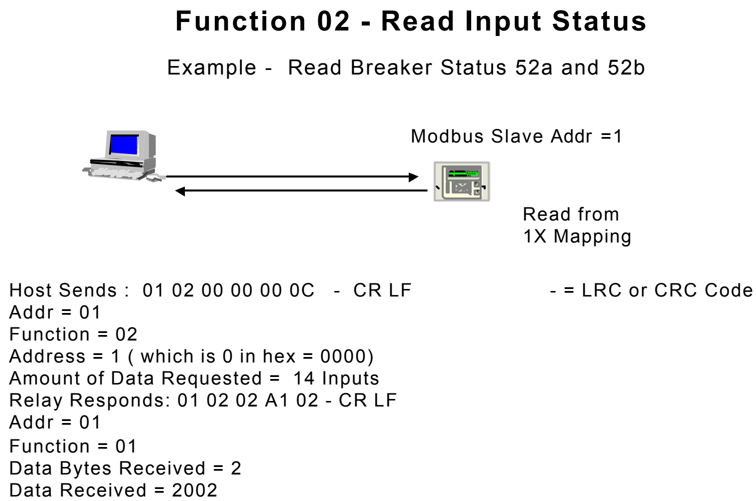


Figure 4-9. Read Input Breaker Status Example

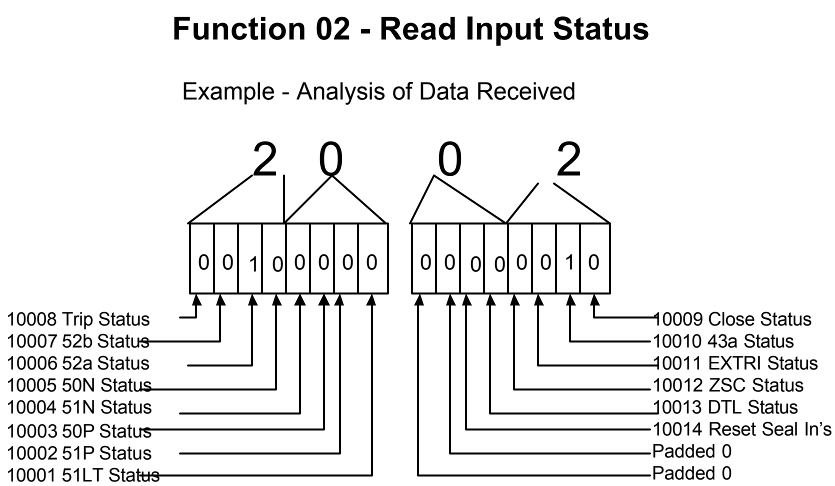


Figure 4-10. Decode of Raw Data Bits Seen on Data Scope Analyzer

The above example illustrates the breaker is closed and reclosing is enabled.

4X Register Read Capabilities

The MSOC implementation of 4X Registers allows for both statuses reads and in limited cases for control register writes. Many host devices do not allow the access of data from discrete data types (such as 0X and 1X discrete output and input function codes). The Modbus implementation within the MSOC overcurrent relay allows for Modbus commands 03, 16 (10 HEX) and 23 (17 HEX) register commands. Real time relay status is available for the following relay data types and functionality:

- Relay Status
- Diagnostic Status
- Unit Information
- CT and PT Information
- Physical Input Status
- Logical Input Status

- Physical Output Status
- Logical Output Status
- Load Metering Data
- Demand Metering Data
- Master Trip Functionality
- Fault Record Buffering (1-32)
- Event Record Buffering (1-128)
- Breaker Counter Operation Retrieval
- Force of Physical Outputs
- Breaker Control Functions over the network
- Reset of Counter, Event Buffer, Operational Buffer, Seal In and Target information

Each function code and data type shall be explained in detail, within the following sections.

Modbus protocol allows a variety of information to be placed within the 4X Register types. The interpretation of the returned data is key to data received in the request. Modbus protocol is predicated upon register information being returned. A register is 2 bytes, or 16 bits which translates into one word. Multiple words may be combined to form a longer word, which allows a larger read to obtain from the MSOC. The MSOC supports the following data return types for 4X formats:

- Unsigned – 16 bits – 2 bytes – Range 0 to + 65,535
- Signed – 16 bits – 2 bytes – Range –32,768 to 32,767
- Unsigned Long – 32 bits – 4 bytes – Range 0 to +4,294,967,295
- Signed Long – 32 bits – 4 bytes – Range –2,147,483,648 to +2,147,483,647
- ASCII – 16 bits – 2 bytes – 2 characters per register (Reference Appendix A)

The tables contained within this document reference the above definitions and give the cadence of bytes or words as:

- MSB Most Significant Byte
- LSB Least Significant Byte
- MSW Most Significant Word
- LSW Least Significant Word
- Msb Most Significant Bit
- Lsb Least Significant bit

Function Code 03 – Read Holding Registers (Read Only)

The 4X frame sequence is illustrated in Figure 4-11 for Function 03 (Read Holding Registers). The Host sends the protocol request and the MSOC responds. The host decodes the data requested dependent upon the definition of the register data. Tables 4-4 through 4-14 list the register mapping for Modbus reads. Access of Latched and Momentary data access is not available through 4X data access.

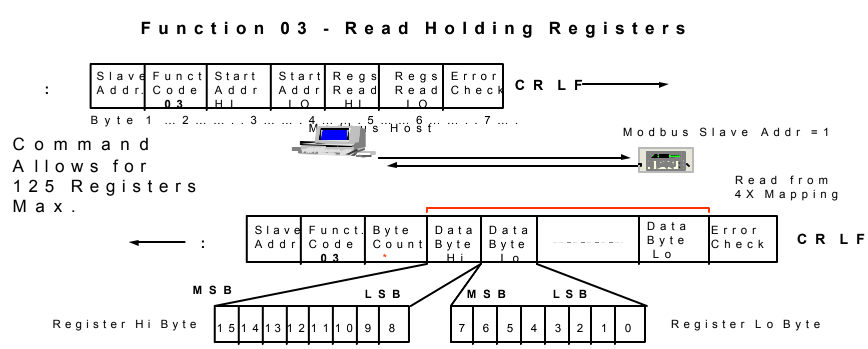


Figure 4-11. 4X Data Read Frame Format

Relay Status (1 Register Defined)

Bit 0 shall update if the unit has failed Self Test. The front panel "STATUS" LED shall illuminate in a red color. The bit shall be a value of 1 until the unit is power cycled or a front panel reset procedure occurs. The self tests take approximately 20 minutes to execute. The factory default setup of the MSOC shall have the FAIL logical output status mapped to OUT 2 physical output at the rear of the relay. If factory defaults are not changed, the FAIL bit will indicate a state of 1 and OUT 2 shall de-energize, indicating a SELF TEST failure.

Bits 1 (Lsb) through Bit 4, Bit 9 and 10, shall update to a 1 if any of the corresponding data to the bit group changes. The Bits shall reset when the register is polled by the host.

Bits 5, 6, 7, 8, and 11, update the status real time indicates the state of the bit defined MSOC relay feature.

If Bit 12 is enabled, then 6X Register parameterization data has been changed through the front panel of the MSOC. In an automation application, a read of this register allows for quick determination of which data to access for immediate display. The Bits are reset when read by the host. Table 4-6 lists the bit mapping for the relay status register.

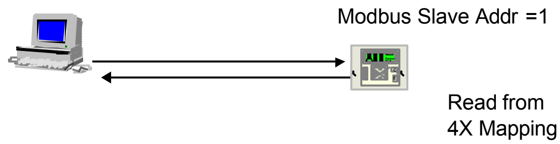
The Relay Status register is especially valuable if an MSOC value has changed, it may be determined via a read of Register 40129. Once the register has been accessed by the host, all bits in Register 40129 will be reset by the relay. The status shall then be refreshed by the MSOC until the next host read of Register 40129. Once the host has detected the status change, the host may access specific registers further detailing the status change.

Table 4-6. Relay Status Modbus Address Map Definition

Register Address	Item	Description
40129	Relay Status	Unsigned 16 bit
	Bit 0 Self Test (Lsb)	Self Test In Progress
	Bit 1 Input Changed	Input Transitioned
	Bit 2 New Event Record	New Record in Buffer
	Bit 3 New Fault Record	New Fault Record In Buffer
	Bit 4 Unit Power Cycle	Unit Power Cycled Since Last Read
	Bit 5 Recloser Disabled	43A Disabled
	Bit 6 Ground Disabled	Disable 51N and 50N Elements
	Bit 7 Supervisor Disabled	Control Via Network Disabled
	Bit 8 Battery Failure	Battery Failed
	Bit 9 – 10 Reserved	
	Bit 11 FP Com Enabled	RS 232 Port Enabled RS 485 Disabled
	Bit 12 Local Settings Changed	Settings Changed via Front Panel
	Bit 13 – 14 Reserved	
	Bit 15 Reserved (Msb)	

Application Example: A Modbus Host is able to parse data in a bit format which it access through the network. The host is required to monitor an MSOC for new fault and event records. What command should be sent to an MSOC to gather the information.

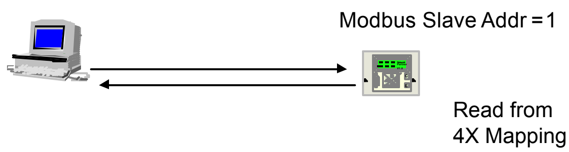
Figures 4-12 and 4-13 illustrate data strings sent to the MSOC to determine if a new event or operation record has been stored.



Obtain the Relay Flag from the MSOC (Register 40129)

Host Sends : 01 03 00 80 00 01 - - = LRC or CRC Code
Addr = 01
Function = 03
Address = 40129 (which is 128 in hex = 0080)
Amount of Data Requested = 1 Register
Relay Responds: 01 03 02 08 2C -
Addr = 01
Function = 03
Data Bytes Received = 2
Data Received = 08 2C

Figure 4-12. Application Example: Fetch Relay Status From MSOC



Obtain the Relay Diagnostic Status Flag from the DPU 2000R (Register 40129)

Data Received = 08 2C

MSB = 08	LSB = 2C
Bits 13 - 15 = 000	Supervisor Disabled = 0
Local Settings Changed = 0	Ground Disabled = 0
Front Panel Com Enabled = 1	Recloser Disabled = 1
New Peak Demand Value = 0	Unit Power Cycle = 0
New Minimum Demand Value = 0	New Fault Record = 1
Battery Failure = 0	New Event Record= 1
	Input Changed = 0
	Self Test = 0

Figure 4-13. Application Example: Returned Relay Response

Since the last read of the status register, a new fault record and event record has been input within the MSOC buffers. The recloser function (43A) is disabled. Communication is enabled through the RS 232 front panel port. The host may then access additional status such as Fault or Event Records contained within the relay.

Diagnostic Status (2 Registers Defined)

Bits 0, 1, or 2 are updated continuously. The MSOC performs diagnostics:

- Upon power-up of the unit.
- Continuously thereafter on a periodic basis. A variety of MSOC diagnostics are performed and completed in 20 minute intervals.

If a “SELF TEST” failure is reported in Register 40129 Bit 0 or discrete output 0007, access of Register 40129 shall enable the user to access the cause. Diagnostic Status is reported via MMI front panel or Network port access.

Bit 3 Reflects the OR’ing of all EEPROM Settings stored. (i.e. if one fails [bit 0, 1, or 2 is set to a 1] this is set.) Within the MSOC are three relay parameter copies. Upon power-up, the copies are compared to each other. If there is a miscompute, an MSOC PROM Failure is logged. Bit 3 is set when the unit failures to successfully read from all three copies of the Stored Parameters.

Bits 0 through 3 are cleared only at a unit Power On Reset, or a unit MSOC reset through the front panel.

ÿ **IMPLEMENTATION TIP-** Front panel reset is accomplished by pressing the "C", "E", and "UP arrow keys" simultaneously on the front MMI panel of the MSOC.

The bit shall indicate 1 for diagnostic failure indication. These bits show the status.

Table 4-7. Diagnostic Status Modbus Address Map Definition

Register Address	Item	Description
40130	Main CPU Diag. Status Bit 0 MramFailure (Lsb) Bit 1 MepromFailure Bit 2 Clock Failure Bit 3 MEEPromFailure Bit 4 – 14 Reserved Bit 15 Reserved (Msb)	Unsigned 16 Bit Main CPU RAM FAILURE Checksum Failure on EPROM If Clock Enabled, Clock update error. MEEP Checksum Failure on refresh or read.
40131	Reserved	Reserved

Unit Information (15 Registers Defined)

Unit information status allows retrieval of MSOC Executive firmware revision numbers; MSOC Catalog numbers as well as MSOC Unit Serial numbers. The MSOC has the use of only one communication port; access of Register 40143 allows a remote host to determine which port is designated for use. Two of the registers within the unit information block are scaled, 40140 and 40141. The returned unsigned 16 bit data values when divided by 100 will mirror the revision numbers as seen on the front LCD panel within the Unit Information menu of the MSOC. These are the only scaled registers within this block of 4X Registers available for read. Table 4-8 further defines the Unit Information status block.

Table 4-8. Unit Information Status Modbus Address Map Definition

Register Address	Item	Description
40132	Catalog Number (MSW)	ASCII – 2 Characters (Leftmost Digits)
40133	Catalog Number	ASCII – 2 Characters
40134	Catalog Number	ASCII – 2 Characters
40135	Catalog Number	ASCII – 2 Characters
40136	Catalog Number	ASCII – 2 Characters
40137	Catalog Number	ASCII – 2 Characters
40138	Catalog Number	ASCII – 2 Characters
40139	Catalog Number (LSW)	ASCII – 2 Characters (Rightmost Digits)
40140	MainCPUSwVersionNumber	Unsigned 16 Bit – (Scale Factor 100)
40141	AuxCPUSwVersionNumber	Unsigned 16 Bit – (Scale Factor 100)
40142	Protocol	Unsigned 16 Bit 0 = Modbus ASCII, 1 = ASCII
40143	Active Port	Unsigned 16 Bit 0 = Rear RS485 Port, 1 = Front RS232 Port
40144	Serial Number	ASCII – 2 Characters (Leftmost Digits)
40145	Serial Number	ASCII – 2 Characters
40146	Serial Number	ASCII – 2 Characters (Rightmost Digits)

Read Quick Status (3 Registers Defined)

CT and PT ratios configuration data is available. As standard, The CT ratio is to 1 as is the Neutral and PT ratios are to 1.

Table 4-9. Quick Status Modbus Address Map Definition

Register Address	Item	Description
40147	Phase CT Ratio	Unsigned 16 Bit
40148	Neutral Ratio	Unsigned 16 Bit
40149	PT Ratio	Unsigned 16 Bit

Fast Status (1 Register Defined)

Fast Status is available for an operator interface to determine the device queried. The Division Code for the MSOC is 1A HEX. The product ID for the MSOC is 40 HEX.

Table 4-10. Fast Status Modbus Address Map Definition

Register Address	Item		Description
40150	Fast Status		Unsigned 16 Bit
	Bit 0 – 5	Division Code (Lsb)	01 1010 = 1A HEX
	Bit 6 – 9	Reserved	
	Bit 10 – 15	Product ID (Msb)	0100 00 = 40 HEX left justified

Logical Inputs (1 Register Defined)

The logical input data is also available through Modbus Command Code 02 (Read Inputs 1X). The information is available in register format for host devices, which cannot emulate the 02 Modbus protocol command. Table 4-11 further defines the bits in word 40257. As with the discrete Logical Input command status described earlier in this guide, each bit follows the real time state of the element when it is accessed and processed within the MSOC. 52a and 52b status may be mapped and wired to the Physical Inputs provided at the rear of the MSOC. However, MSOC does not have these contacts wired to the MSOC physical input, status of 52a/52b, shall be reported as a function of sensed current flow. If 52a and 52b are unmapped, the MSOC shall report breaker status as such:

A closed breaker is determined when the input current is above 5% of the neutral nominal current rating (refer to Table 4-15) of the relay. An open breaker (cleared breaker) is determined when the sensed current is above 5% of the neutral nominal current rating.

Table 4-11. Logical Input Modbus Address Map Definition

Register Address	Item	Description
40257	Logical Inputs	Unsigned 16 Bit
	Bit 0 51LT (Lsb)	Phase Time Overcurrent
	Bit 1 51P	Phase Time Overcurrent
	Bit 2 50P	Phase Instantaneous Overcurrent
	Bit 3 51N	Ground Time Overcurrent
	Bit 4 50N	Ground Instantaneous Overcurrent
	Bit 5 52a	Relay Status (52a = 1, 52b = 0 CB Open)
	Bit 6 52b	Relay Status (52a = 0, 52b = 1 CB Closed)
	Bit 7 TRIP	Relay Present State is Tripping
	Bit 8 CLOSE	Relay Present State is Closing
$$	Bit 9 43a	Enable Reclose Function Asserted
$$	Bit 10 EXTRI	External Reclose Initiate Asserted
$$	Bit 11 ZSC	Zone Sequence Closing Function State (79)
$$	Bit 12 DTL	Drive To Lockout
**	Bit 13 RSI	Reset Seal Ins
	Bit 14 Reserved	
	Bit 15 Reserved (Msb)	
		$$ = Added in Version 1.6 Software ** = Added in Version 1.7 Software

Physical Inputs (1 Register Defined)

The physical input data defined for 40258 is also available through Modbus 02 (Read Inputs 1X) command. The information is available in register format for host devices, which cannot emulate the 01 or 02 Modbus protocol commands. Table 4-12 further defines the bit positions for access of input status via the 4X data access.

Table 4-12. Physical Input Modbus Address Map Definition

Register Address	Item	Description
40258	Physical Inputs	Unsigned 16 Bit
	Bit 0 IN1 (Lsb)	Input Status (Terminal 5 & 4 [common])
	Bit 1 IN2	Input Status (Terminal 6 & 4 [common])
	Bit 2 – 14 Reserved	
	Bit 15 Reserved (Msb)	

Logical Outputs (1 Register Defined)

The Logical Output data defined for 40385 is also available through Modbus Code 01 (Read Coil 0X). The information is available in register format for host devices, which cannot emulate the 01 Modbus protocol command.

ÿ **IMPLEMENTATION TIP-** If breaker status is to be required after the event occurs, it is advisable to read the seal in bits described in Table 4-3 (Address 00255 and 00256). Seal in bit capability is available in MSOC firmware Version 1.70. If one has an earlier version of MSOC, Address 40257 (Bits 5 and 6) could be polled to monitor the actual state of 52a and 52b. The polled state may then be saved and compared to the earlier values to determine whether a trip or close has occurred between device polls.

Table 4-13. Logical Output Modbus Address Map Definition

Register Address	Item	Description
40385	Logical Outputs	Unsigned 16 Bit
	Bit 0 51LT (Lsb)	Phase Time Overcurrent
	Bit 1 51P	Phase Time Overcurrent
	Bit 2 50P	Phase Instantaneous Overcurrent
		Ground Time Overcurrent
	Bit 3 51N	Ground Instantaneous Overcurrent
	Bit 4 50N	Timeout or Trip Pickup Alarm
		MSOC Diagnostic Failure
	Bit 5 PUA	Circuit Breaker Fail Alarm
	Bit 6 FAIL	Breaker is Tripping
	Bit 7 CBFAIL	Breaker is Closing
	Bit 8 TRIP	Reclosing In Progress
	Bit 9 CLOSE	Recloser Lockout Alarm
$$	Bit 10 RCLINPROG	Recloser Disable Alarm
$$	Bit 11 RCLALARM	Recloser Max Recl Alarm
$$	Bit 12 RDA	
$$	Bit 13 RCLRMA	
	Bit 14 Reserved	
	Bit 15 Reserved (Msb)	
		$$ = Added in Version 1.6 Software

Physical Outputs (1 Register Defined)

The Logical Output data defined for 40386 is also available through Modbus Code 01 (Read Coil 0X). The information is available in register format for host devices, which cannot emulate the 01 Modbus protocol command. The Physical Output data block only reports the real time status of the OUTPUT coils when the unit is polled. Status bits within the register is defined in Table 4-14.

Table 4-14. Physical Output Modbus Address Map Definition

Register Address	Item	Description
40386	Physical Outputs	Unsigned 16 Bits
	Bit 0 OUT1 (Lsb)	Physical Relay Output Contact Status (Terminal 17 & 18)
	Bit 1 OUT2	Physical Relay Output Contact Status (Terminal 15 & 16)
	Bit 2 OUT3	Available if no Voltage Inputs – Physical Relay Output Contact Status (Terminal 13 & 14)
	Bit 3 OUT4	Available if no Voltage Inputs- Physical Relay Output Contact Status (Terminal 11 & 12)
	Bit 4 – 14 Reserved	
	Bit 15 Reserved (Msb)	

Load Metering (27 Registers Defined)

Metering Values are defined Table 4-16. Various data types are associated with each element. Some values, such as Kwatts (32 Bit, 4 byte, 2 word) are signed to denote power flow. Other numbers are scaled, such as frequency to denote a decimal point when read. Frequency 40539 should be divided by 100 to obtain the decimal point, which is visible when viewing the value from the front panel. The metering values obtained through the front panel of the MMI are the same values reported through the network port of the MSOC.

It should be noted that metering values should be calculated if the measured values are above 5% of nominal calculated values for MSOC executive Versions of 1.63 or greater. Metering values shall be calculated if the measured values are above 1.25% of the nominal calculated values for MSOC executive software versions of 1.70

or greater. If the values are less than those calculated, then the returned value for the underloaded quantity shall be reported as 0. The values used for I nominal and V nominal are:

Table 4-15. Nominal Ratings for Current and Voltage

Catalog Number	Phase Range	Phase "I" Nominal	Neutral Range	Neutral "I" Nominal	"V" Nominal
474 M 0XXX-XXX	1.5 – 12 A	5 A	1.5 – 12 A	5 A	120 V
474 M 1XXX-XXX	1.5 – 12 A	5 A	0.5 – 4.0 A	1.67 A	120 V
474 M 2XXX-XXX	1.5 – 12 A	5 A	0.1 – 0.8 A	0.33 A	120 V
474 M 4XXX-XXX	0.3 – 2.4 A	1 A	0.3 – 2.4 A	1 A	120 V
474 M 5XXX-XXX	0.3 – 2.4 A	1 A	0.1 – 0.8 A	0.33 A	120 V
474 M 6XXX-XXX	0.1 – 0.8 A	0.33 A	0.1 – 0.8 A	0.33 A	120 V
474 M 7XXX-XXX	1.5 – 12 A	5 A	0.3 - 2.4 A	1 A	120 V
X = Don't Care					

Application Example: The following metering values are measured using discrete meters. Is the feeder loaded enough to allow the MSOC to register metering data through the network or through the front MMI panel?

Ia = 93 A kVab = 13.85 kV
Ib = 91 A kVbc = 13.72 kV
Ic = 95 A kVca = 13.88 kV
In = 2 A

Solution: Calculate minimum values required to provide metering values for I (phase), I (neutral) V, and KW from the front panel MMI or network. The relay configuration parameters are as such:

VT Conn = Delta
PH CT Ratio = 200:1 (CT Ratio for a relay is different than one for metering which is Turns: 5)
PH N Ratio = 10:1 (CT Ratio for a relay is different than one for metering which is Turns: 5)
VT Ratio = 125:1
Relay Type 474M 1421 – 6010

After consulting Table 4-15, Phase I nom = 5 A, Neutral I nom = 1.67 A, V nom = 120 V.
The formulas to calculate the desired values are as such:

Formula 1 V min. measurement for Version 1.63 or earlier:
*V min measurement = 5% * V nom * VT Ratio*
Formula 2 V min. measurement for Version 1.70 or later:
*V min measurement = 1% * V nom * VT Ratio*
Formula 3 I phase min. measurement for Version 1.63 or earlier:
*I min measurement = 5 % * I Phase nom * CT Phase Ratio*
Formula 4 I phase min. measurement for Version 1.70 or later:
*I min measurement = 1 % * I Phase nom * CT Phase Ratio*
Formula 5 I neutral min. measurement for Version 1.63 or earlier:
*I min measurement = 5% * I Neutral nom * CT Neutral Ratio*
Formula 6 I neutral min. measurement for Version 1.70 or later:
*I min measurement = 1% * I Neutral nom * CT Neutral Ratio*
Formula 7 KW minimum measurement for Version 1.63 or earlier:
*KW min measurement = (CT ratio * VT Ratio * I nom * V nom * 3 phases * 5%)/ 1000*
Formula 8 KW minimum measurement for Version 1.63 or earlier:
*KW min measurement = (CT ratio * VT Ratio * I nom * V nom * 3 phases * 1%)/ 1000*
Using Formula 1 and 2 V minimum to measure should be:
0.750 kV = V min measurement = 0.05 * 120 V * 125 for Version 1.63 or earlier.
0.150 kV = V min measurement = 0.01 * 120 V * 125 for Version 1.70 or later.

Using Formula 3 and 4 I phase minimum to measure should be:
50 A = I min measurement = 0.05 * 5A * 200 for Version 1.63 or earlier.
10 A = I min measurement = 0.01 * 5A * 200 for Version 1.70 or later.
Using Formula 5 and 6 I neutral minimum to measure should be:
0.835 A = I min measurement = 0.05 * 1.67 * 10 for Version 1.63 or earlier.
0.167 A = I min measurement = 0.01 * 1.67 * 10 for Version 1.70 or later.
Using Formula 7 and 8 KW minimum to measure should be:
2,250KW nom KW min measurement (3 * 200 * 125 * 5 * 120 * 0.05)/1000 for Version 1.63 or earlier.
450KW nom KW min measurement (200 * 125 * 5 * 120 * 0.01 * 3)/1000 for Version 1.70 or earlier.

So far V, I phase, I neutral are sufficiently loaded to allow the MSOC to calculate metering values. Since a Wattmeter was not inserted into the circuit to record KW, a value may be calculated.

The circuit is wired in a Delta configuration, the V (line to neutral) shall be calculated from V phase to phase per equation 9:

EQUATION 9 V (phase to neutral) = V (phase to phase) * I Phase * square root of 3
KW = I * KV * square root of 3 (which computes voltage line to neutral) * PF
2222.91 KW = 93 * 13.8KV * 1.732

CONCLUSION: If the MSOC Version is 1.63 or earlier, the relay will report a value of 0 for kW. KVar will probably not be metered also. All other values will meter correctly. If the MSOC Version is 1.70 or greater, the relay will report all values correctly. If the metering value is not registering, it would be advisable to check the power factor of the feeder being measured.

Table 4-16. Load Metering Modbus Address Map Definition

Register Address	Item	Description
40513	Ia Magnitude	Unsigned 16 Bit
40514	Ib Magnitude	Unsigned 16 Bit
40515	Ic Magnitude	Unsigned 16 Bit
40516	In Magnitude	Unsigned 16 Bit
40517	Vab Magnitude	Unsigned 32 Bit High Order Word MSW
40518	Vab Magnitude	Unsigned 32 Bit Low Order Word LSW
40519	Vbc Magnitude	Unsigned 32 Bit High Order Word MSW
40520	Vbc Magnitude	Unsigned 32 Bit Low Order Word LSW
40521	Vca Magnitude	Unsigned 32 Bit High Order Word MSW
40522	Vca Magnitude	Unsigned 32 Bit Low Order Word LSW
40523	Van Magnitude	Unsigned 32 Bit High Order Word MSW
40524	Van Magnitude	Unsigned 32 Bit Low Order Word LSW
40525	Vbn Magnitude	Unsigned 32 Bit High Order Word MSW
40526	Vbn Magnitude	Unsigned 32 Bit Low Order Word LSW
40527	Vcn Magnitude	Unsigned 32 Bit High Order Word MSW
40528	Vcn Magnitude	Unsigned 32 Bit Low Order Word LSW
40529	Kwatts (Three Phase)	Signed 32 Bit High Order Word MSW
40530	Kwatts (Three Phase)	Signed 32 Bit Low Order Word LSW
40531	KVArs (Three Phase)	Signed 32 Bit High Order Word MSW
40532	KVArs (Three Phase)	Signed 32 Bit Low Order Word LSW
40533	KWHrs	Unsigned 32 Bit High Order Word MSW
40534	KWHrs	Unsigned 32 Bit Low Order Word LSW
40535	KVArHrs	Unsigned 32 Bit High Order Word MSW
40536	KVArHrs	Unsigned 32 Bit Low Order Word LSW
40537	Power Factor Value	Signed 16 Bit (Multiplier = 100)
40538	Power Factor Direction	Unsigned 16 Bit – 1= Lagging 0 = Leading
40539	Frequency	Unsigned 16 Bit (Multiplier = 100)

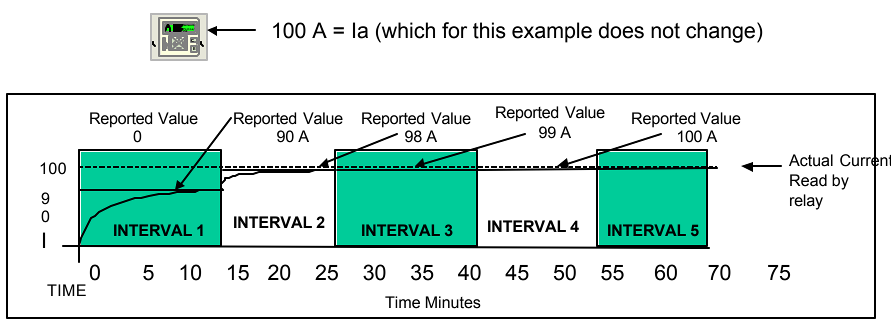
Demand Metering (8 Registers Defined)

Demand Metering is reported within Table 4-17. The accumulated magnitudes are reported in 16 bit unsigned and 32 bit unsigned numerical values as indicated in the next table. The demands are reset by writing a reset command to the 4X Register, 41668 Bit 5. Please reference Table 4-19 of this document for the control register group and bit designation to reset this group of registers. Refer to Table 4-4 Register 40129; bit 10, which will indicate that a new Peak Demand Value has been accumulated within this table.
Demand metering is calculated on a fixed demand window accumulation. The demands are based upon a time window of 15, 30, or 60-minute calculation intervals. Refer to Table 4-21 within this document to reference the procedure for setting the sliding demand window time base.

Demand Metering initiates at time = 0 which may be a unit power up, system reset via the front panel or through a demand metering reset via the network as described in Table 4-19 of this document. It is not dependent upon the time-of-day clock (TOD) within the unit. The MSOC has an internal timer that is monitored to determine the end of the selected interval (15, 30, or 60 minutes) and the start of the new interval.

Current (Ia, Ib, Ic, and In) and power (KW and KVAR) are calculated and integrated within the demand calculation for that interval on a 32 cycle time period interval within the demand time window selected. The following figures illustrate the method of calculating and reporting the Demand Values depending upon reporting of current or energy.

Current Demand Metering



Interval sample is taken every 32 cycles (533mS at 60 Hz). The logaritmic function shall report 90% of the integrated value calculated beneath the demand curve.

At Time = 15 min.
Current is integrated logarithmically for the entire window time selected (in this case 15 min) such that the reported value is 90 % of the present value at the end of demand time.
The Demand value register is updated every demand cycle which in this case is every 15 minutes.

Figure 4-14. Demand Current Calculation

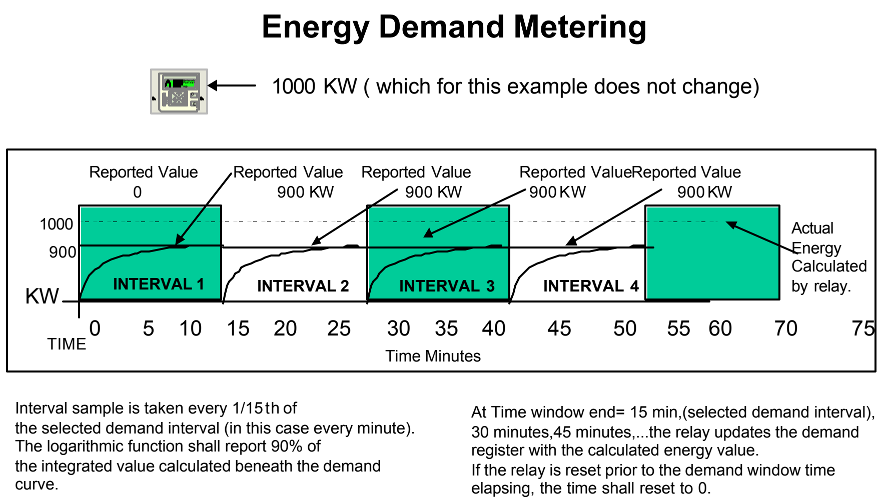


Figure 4-15. Energy Demand Calculation

Figures 4-13 and 4-14 illustrate the energy and current calculation methods and data reported when accessed via the network. To simplify the explanation, the current and energy has been kept constant. This example illustrates a calculation based upon a window size of 15 minute demand intervals.

Table 4-17. Demand Metering Modbus Address Map Definition

Register Address	Item	Description
40769	Ia Magnitude	Unsigned 16 Bit
40770	Ib Magnitude	Unsigned 16 Bit
40771	Ic Magnitude	Unsigned 16 Bit
40772	In Magnitude	Unsigned 16 Bit
40773	Kwatts	Unsigned 32 Bit High Order Word MSW
40774	Kwatts	Unsigned 32 Bit Low Order Word LSW
40775	Kvars	Unsigned 32 Bit High Order Word MSW
40776	Kvars	Unsigned 32 Bit Low Order Word LSW

Master Trip (5 Registers Defined)

The Master Trip function allows internal relay elements to energize the logical MSOC Master Trip. If the following registers are read (as listed in Table 4-18) with a value of 1, the element defined within the register is mapped to the Master Trip logical function. The Master Trip MSOC internal logical element will energize if the corresponding mapped bit is set to ON (1). The Master Trip logical elements energizes the Physical Output mapped to the Master Trip function. To configure the elements to the Master Trip function, refer to Registers 61665–61674 in Table 4-38 for a more complete explanation.

It should be noted that a fault record is generated if the protective element mapped to the Master Trip function (reference Table 4-19 of this document) is energized. If an element is not mapped to the Master Trip function, a Fault Record shall not be generated upon the energizing of the unmapped protective element.

An Event Record shall still be generated regardless of Master Trip mapping.

Table 4-18. Master Trip Modbus Address Map Definition

Register Address	Item	Description
41537	Master Trip 51LT	Phase Time Overcurrent (Long Time)
41538	Master Trip 51P	Phase Time Overcurrent
41539	Master Trip 50P	Phase Instantaneous Overcurrent
41540	Master Trip 51N	Ground Time Overcurrent
41541	Master Trip 50N	Ground Instantaneous Overcurrent
	0 = OFF 1 = ON	Note: Default is all bits on (= 1)

4X Register Write Capabilities

All of the Modbus Write commands discussed so far have involved reads only. Modbus allows for two types of commands involving control. One Modbus command allows register writes. Another Modbus command allows for register writes and reads with one command. The type of functionality performed with relay writes is as such:

- Access of Fault Records
- Access of Event Records
- Trip/Close Initiation
- Enable/Disable of Protective Functions
- Clearing of Event Counters
- Enable/Disable of Supervisory Functions
- Reset of Targets
- Clear of Seal In's

Function Code 16 - Preset 4X Registers (Write Only)

Figure 4-16 illustrates the Modbus command structure writing multiple registers.

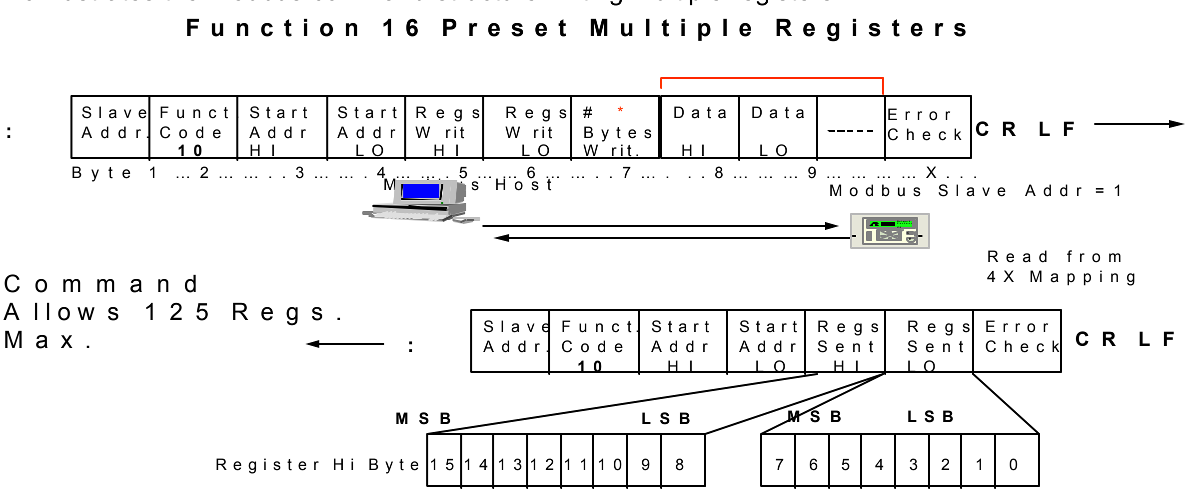


Figure 4-16. Modbus Write Command 16 (10 HEX) Allowing Writes to the MSOC

The write multiple register command is convenient for writing the following control blocks:

- Control Block 1 – 41664 through 41670
- Control Block 2 – 41671 through 41676
- Control Block 3 – 41677 through 41682
- Control Block 4 – 41683 through 41688

Control Block 1 allows for:

- Initiation of Relay Trip
- Initiation of Relay Close
- Resetting of Targets
- Enabling or Disabling of Selective Protective Functions
- Disabling or Enabling of Communication Port Functions
- Enabling or Disabling of Supervisory Control Functions via the Communication Ports

Control Block 2 allows for:

- Forcing of Outputs 1 through 4 for a limited pulse duration time

Control Block 3 allows for:

- Reset of Latch Bit Functions

Control Block 4 allows for:

- Reserved Functions

Whenever a write occurs to the MSOC:

- The MSOC receives the command
- Command Interpreted in 1 quarter cycle
- Relay Protection Occurs
- Command acts on the device
- The command response is generated to the Host from the MSOC after the action is completed

The defined control blocks 1 and 2 are write capable and are well suited for access control via the Modbus command 16 (10 HEX).

Function 23 Read/Write Register (Read/Write Concurrently)

Another format command which allows for a simultaneous read/write is command 23 (17 Hex). Figure 4-17 illustrates the Read/Write 4X Register command format. The 23 command is used when the user wishes to write a register for control buffer access and read a group of registers which was accessed via the read.

Control Blocks 1 and 2 allows for access of protective device function state. If a user wished to read the status of each function within the relay, a Function Read/Write Register command would be the most desirable command to be issued. Read/Write Register data commands are also useful in accessing the Operation and Fault record blocks.

Review of the Modbus 23 command allows for write and read of data if the total amount of read and write registers do not exceed over 125 words. An advantage of using a combined read/write command is that of speed. If conventional commands were to be used, a 16 Write 4X Register command would be issued and thereafter, within 10 seconds, a 03 Modbus (Read 4X Register command) would then be issued to extract the data from the relay. Using Modbus command 23 allows for decreasing of the overhead associated with multiple register reads and writes.

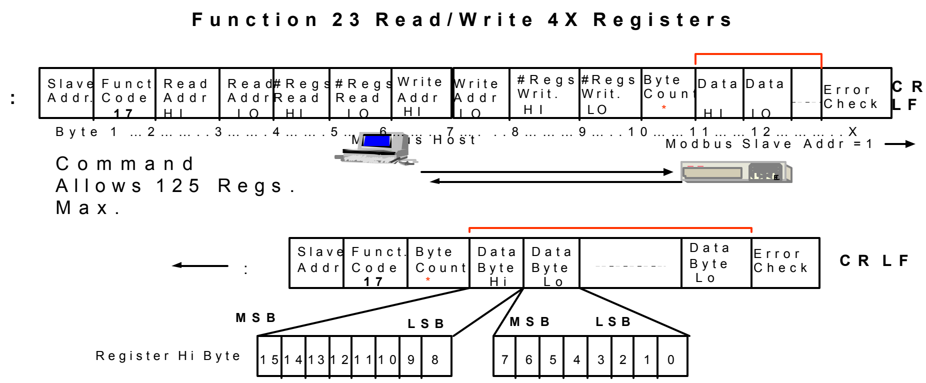


Figure 4-17. Function 23 Read/Write Command Format

Fault Records (27 Registers Defined)

Fault records are stored in the MSOC according to the following format. Figure 4-18 illustrates the method of accessing the Fault Record Data via the MSOC. The MSOC has an internal circular buffer, which stores a maximum of 32 faults. These faults are stored internally to the MSOC's fault stack as indicated in the figure. Each fault is defined as a block of 26 registers as shown in Tables 4-19 and 4-20. The first defined register in the table is the fault record control register. Fault records are viewed by writing a data word to 41025 as defined in the table below and reading the block of consecutive Registers from 41026 through 41051.

If the number of faults exceed 32, then the buffer overwrites the oldest record contained within its internal stack. Access and control can be accomplished over Modbus in one of two methods.

If 41025 has a value of 1 written to it, Registers 41026 through 41051 will fill with the FIRST fault within the 32 records stored in the unit. 41025 will then reset to a value of 0 when Registers 41046 through 41051 are read.

If 41025 has a value of 2 written to it, Registers 41026 through 41051 will contain the NEXT record of fault data which was pointed after the write command executed. 41025 will reset to a value of 0 after the record has entered the buffer and is read by the host.

If 41025 has a value of 3 written to it, Registers 41026 through 41051 will fill with the LAST record of fault data in the 32 records of fault data stored in the unit. For example, if two records of data accumulated between reads, a read last record command would point to the last record of data accumulated in the buffer. The user would have to re-index the pointer to access the previous data fault record in the buffer. 41025 shall reset to a value of 0 after the record data is read.

If no data accumulated within the fault record, (such as after a system reset), values of 0 shall be returned in the buffer. A new fault record entry is indicated via Bit 3 of Register 40129 being set to a 1. Reference Table 4-1 of this document for a more detailed explanation of the registers bit map.

The Fault Record number can be a number from 1 to 999. Only the previous 32 records are kept in the fault record buffer. Fault Records are sequentially numbered from 1 to 999. If the fault number is presently at 999, and an additional fault is recorded, the fault number shall rollover to 1. The Record number and fault buffer can be cleared and reset through a keypad reset procedure or a reset via the network as explained in Section 3 as a note. If the Fault number retrieved is 0 then, no faults have been recorded by the relay. Front panel LCD access of the faults shall indicate that no fault records are available.

METHOD 1:

The host writes a Modbus 23 command (Modbus 4X Register Read/Write) in which a control code (1,2, or 3) is written to 41025 and the buffer is filled with fault data in Registers 40126 through 41051 to be returned as a response to the command. A command of 1 = Points to the First Record in the Fault Table.

A command of 2 Points to the next fault in the fault table. A command of 3 points to the last fault in the fault table. Figure 4-18 graphically illustrates the write/read process for access of fault or operation records.

METHOD 2:

The host writes a Modbus command 16 (Modbus 4X Register Write) in which a control code (1,2, or 3) is written to 41025 and the buffer is filled with fault data in Registers 41026 through 41051. Within 10 seconds after the 16 command is issued, the host issues a Modbus 03 command (Modbus 4X Register Read) in which the fault data is retrieved from the buffer in Registers 41126 through 41051.

ÿ **IMPLEMENTATION TIP-** Only elements mapped to the master trip output shall generate fault records. If the elements are not mapped to the master trip output, only an event record shall be recorded.

Table 4-19. Fault Status Modbus Address Map Definition

Register Address	Item	Description
41025	FltRecCtlReg 1 = First Record 2 = Next Record 3 = Last Record	Fault Record Control Register Unsigned 16 Bit 1 = Fill 41026 – 41051 with First Record Data. 2 = Fill 41026 – 41051 with Next Record Data pointed to in buffer. 3 = Fill 41026 – 41051 with the last record of data.
41026	Fault Record Number	Unsigned 16 Bit (1 – 999, only last 32 kept)
41027	Fault Trip Type (see below)	Unsigned 16 Bit (See Reference at end of table)
**41028	Reclose Sequence	Event Recloser Sequence which generated record 0 – 5.
41029	Month 1 – 12	Unsigned 16 Bit Month of Fault
41030	Day 1 – 31	Unsigned 16 Bit Day of Fault
41031	Year 2 digit (00 – 99)	Unsigned 16 Bit Year of Fault
41032	Hour 00 – 23	Unsigned 16 Bit Hour of Fault
41033	Minute 00 – 59	Unsigned 16 Bit Minute of Fault
41034	Second 00 – 59	Unsigned 16 Bit Second of Fault
41035	Hundred Seconds 0 – 99	Unsigned 16 Bit Hundredth Second of Fault Time
41036	Thousand Seconds 0 – 999	Unsigned 16 Bit Thousandth Second of Fault Time
41037	Fault Ia Magnitude	Unsigned 16 Bit
41038	Fault Ib Magnitude	Unsigned 16 Bit
41039	Fault Ic Magnitude	Unsigned 16 Bit
41040	Fault In Magnitude	Unsigned 16 Bit
41041	Fault Van Magnitude	Unsigned 32 Bit High Order Word MSW
41042	Fault Van Magnitude	Unsigned 32 Bit Low Order Word LSW
41043	Fault Vbn Magnitude	Unsigned 32 Bit High Order Word MSW
41044	Fault Vbn Magnitude	Unsigned 32 Bit Low Order Word LSW
41045	Fault Vcn Magnitude	Unsigned 32 Bit High Order Word MSW
41046	Fault Vcn Magnitude	Unsigned 32 Bit Low Order Word LSW
41047	Relay TripTime In MS	Unsigned 32 Bit High Order Word MSW. Time for Element to Trip.
41048	Relay TripTime In MS	Unsigned 32 Bit Low Order Word LSW
41049	Clearing Time in MS	Unsigned 32 Bit High Order Word MSW-
41050	Clearing Time in MS	Unsigned 32 Bit Low Order Word LSW-
**41051	Year (Year 2000 Compliant)	Unsigned 16 Bit 4 digit year
** = Version 1.70 or later		

Table 4-20. Fault Codes

Fault Trip Type (Reported in Register 41027)
0 = 51LT Fault
1 = 51P Fault
2 = 50P Fault
3 = 51N Fault
4 = 50N Fault
% * 5 = External Trip Fault
% * 6 = Input Trip Fault
% * 7 = Max Fault = Error
Note % = Version 1.90 or later

Fault and Event Record Layout (MSOC)

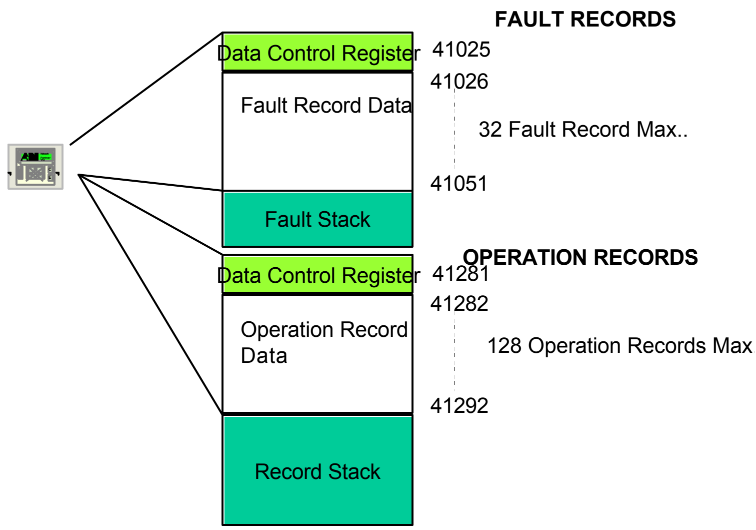


Figure 4-18. Event and Operation Memory Map for MSOC

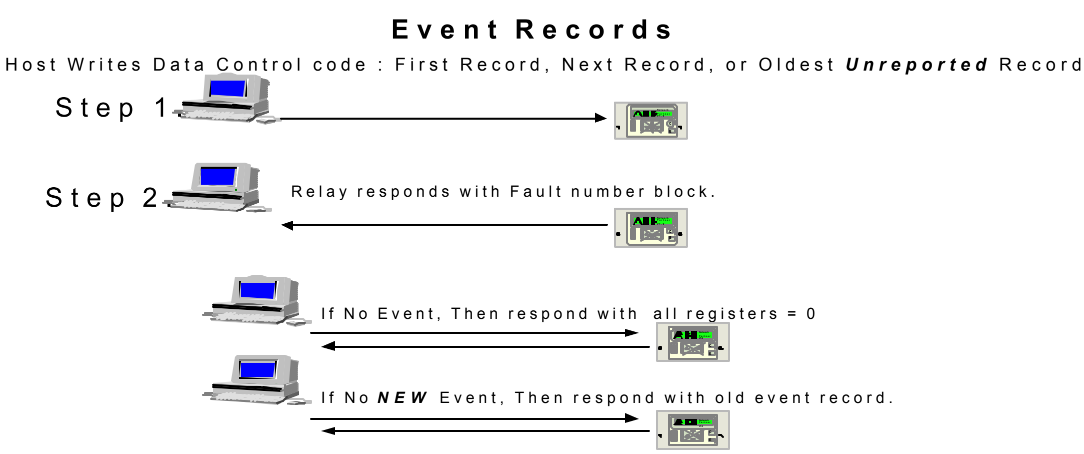


Figure 4-19. Event Record Access Illustration If Function 23 Issued to MSOC Device

Event Records (12 Registers Defined)

Event Record data is stored in the same manner as the Fault Record Data. Figure 4-20 illustrates the method of storage of the Event Record Data. As illustrated, 128 Groups of fault data is stored internal to the MSOC. Each group is comprised of 11 registers of data as defined in Tables 4-21 and 4-22 below. The register for pointing to a group is defined in Register 41281. Fault records are viewed by writing a data word to 41281 as defined in the table below and reading the block of consecutive Registers from 41282 through 41292.

If the number of Operation Records exceed 128, then the buffer overwrites the oldest record contained within its internal stack. Access and control can be accomplished over Modbus in one of two methods.

If 41281 has a value of 1 written to it, Registers 41282 through 41292 will fill with the FIRST Operation record within the 128 records stored in the unit. 41281 will then reset to a value of 0 when Registers 41282 through 41292 are read.

If 41281 has a value of 2 written to it, Registers 41282 through 41292 will contain the NEXT record of Operation Record data which was pointed after the write command executed. 41281 will reset to a value of 0 after the record has entered the buffer and is read by the host.

If 41281 has a value of 3 written to it, Registers 41282 through 41292 will fill with the LAST record of Operation Record data in the 128 records of fault data stored in the unit. For example, if two records of data accumulated between reads, a read last record command would point to the last record of data accumulated in the buffer. The user would have to re-index the pointer to access the previous data fault record in the buffer. 41281 shall reset to a value of 0 after the record data is read.

If no data accumulated within the fault record, (such as after a system reset), values of 0 shall be returned in the buffer. A new fault record entry is indicated via Bit 4 of Register 40129 being set to a 1. Reference Table 4-1 of this document for a more detailed explanation of the registers bit map.

As with fault records, there are two methods of obtaining the information via the Modbus 23 (Write/Read) command or a combination of the Modbus 16 (Write Register) and 03 (Read Register) commands.

METHOD 1:

The host writes a Modbus 23 command (Modbus 4X Register Read/Write) in which a control code (1,2, or 3) is written to 41281 and the buffer is filled with fault data in Registers 41282 through 41292 to be returned in response to the command. A command of 1 = Points to the First Record in the Fault Table. A command of 2 Points to the next fault in the fault table. A command of 3 points to the last fault in the fault table.

METHOD 2:

The host writes a Modbus Command 16 (Modbus 4X Register Write) in which a control code (1,2, or 3) is written to 41281 and the buffer is filled with fault data in Registers 41282 through 41292. Within 10 seconds after the 16 command is issued, the host issues a Modbus 03 command (Modbus 4X Register Read) in which the fault data is retrieved from the buffer in Register 41282 through 41292.

One should note the operation record event codes are arranged in groups to easily indicate the type of error dependent on the value of the operation record. Table 4-22 lists the Operation Record Event Codes.

Table 4-21. Operation Record Address Definition

	Register Address	Item	Description
	41281	EvtRecCtlReg 1 = First Record 2 = Next Record 3 = Last Record	Fault Record Control Register Unsigned 16 Bit 1 = Fill 41281 – 41291 with First Record Data. 2 = Fill 41281 – 41291 with Next Record Data pointed to in buffer. 3 = Fill 41281 – 41291 with the last record of data.
	41282	Event Record Number	16 Bit Unsigned (1 – 999, only last 128 kept)
	41283	Event Type	16 Bit Unsigned (See Description Below)
	41284	Month	Unsigned 16 Bit Month of Event
	41285	Day	Unsigned 16 Bit Day of Event
	41286	Year (0 – 99)	Unsigned 16 Bit Year of Event
	41287	Hour	Unsigned 16 Bit Hour of Event
	41288	Minute	Unsigned 16 Bit Minute of Event
	41289	Second	Unsigned 16 Bit Second of Event
	41290	Hundredths of a Second	Unsigned 16 Bit Hundredth Second of Event Date
	41291	Thousandth of a seconds	Unsigned 16 Bit Thousands of Second of Event Date
$$	41292	Year (Year 2000 Compliant)	Unsigned 16 Bit 4 digit year
			$$ = Added in Version 1.6 Software

Table 4-22. Event Record Type (Register 41283 Code Definition)

	Circuit Breaker Events 1 - 9
	0 = CB Closed
	1 = CB Open
	2 = Unknown CB
	3 = CB Fail to Trip
	4 = CB Fail to Close
	5 = CB Popped Open
	6 = CB Popped Closed
	7 = External Trip
	8 = External Close
	9 = SPARE_CB_10
	Diagnostic Status 10 - 39
	10 = RAM1 Failure
	11 = RAM 2 Failure
	12 = RTC Failure
	13 = EE Prim Failure
	14 = EE Config Failure
	15 = EE Output Failure
	16 = EE Input Failure
	17 = EE Comm Failure
$$	18 = EE Recl Failure
	19 = EE Master Trip Failure
	20 = EEP PROM Checksum Failure
	21 = Analog Failure
	22 = Editor Access
	23 = Battery Failure
	24 = Watchdog Reset
	25 = Power On Reset
**	26 = EE Seal-in Failure
**	27 = SPARE_DIAG_18
**	28 = SPARE_DIAG_19
**	29 = SPARE_DIAG_20
**	30 = SPARE_DIAG_21
**	31 = SPARE_DIAG_22
**	32 = SPARE_DIAG_23
**	33 = SPARE_DIAG_24
**	34 = SPARE_DIAG_25
	35 = SPARE_DIAG_26
	36 = SPARE_DIAG_27
	37 = SPARE_DIAG_28
	38 = SPARE_DIAG_29
	39 = SPARE_DIAG_30

	Relay Element Status 40 - 49
	40 = 51LT Disabled
	41 = 51LT Enabled
	42 = 51P Disabled
	43 = 51P Enabled
	44 = 50P Disabled
	45 = 50P Enabled
	46 = 51N Disabled
	47 = 51N Enabled
	48 = 50N Disabled
	49 = 50N Enabled
	Logical Input Status 50 - 69
	50 = 52a Open
	51 = 52a Closed
	52 = 52b Open
	53 = 52b Closed
	54 = Direct Trip
	55 = Direct Close
&&	56 = RCL 43a Enabled
&&	57 = RCL 43a Disable
&&	58 = RCL Ext Init
&&	59 = Manual Reset
	60 = SPARE_INPUT_21
	61 = SPARE_INPUT_22
	62 = SPARE_INPUT_23
	63 = SPARE_INPUT_24
	64 = SPARE_INPUT_25
	65 = SPARE_INPUT_26
	66 = SPARE_INPUT_27
	67 = SPARE_INPUT_28
	68 = SPARE_INPUT_29
	69 = SPARE_INPUT_30

	Logical Output Events 70 - 99
	70 = 51LT Trip
	71 = 51LT Pickup
	72 = 51LT Dropout
	73 = 51P Trip
	74 = 51P Pickup
	75 = 51P Dropout
	76 = 50P Trip
	77 = 50P Pickup
	78 = 50P Dropout
	79 = 51N Trip
	80 = 51N Pickup
	81 = 51N Dropout
	82 = 50N Trip
	83 = 50N Pickup
	84 = 50N Dropout
	85 = Clear 51LT Event
	86 = Clear 51P Event
	87 = Clear 50P Event
	88 = Clear 51N Event
	89 = Clear 50N Event
	90 = Reset 51LT Event
	91 = Reset 51P Event
	92 = Reset 50P Event
	93 = Reset 51N Event
	94 = Reset 50N Event
&&	95 = Recloser Enabled
&&	96 = Recloser Disabled
	97 = SPARE_OUTPUT_28
	98 = SPARE_OUTPUT_29
	99 = SPARE_OUTPUT_30
	Internal MSOC Events 100 - 109
	100 = QSI Initialization failure
	101= Com Init MSG Failure
	102 = APC Param Update
	103 = Int Tx Failure
	104 = Wrong (Modbus) Block Rec'd
	105 = NW Param update
	106= Int Rx Failure
	107 = SPARE_COMM_08
	108 = SPARE_COMM_09
	109 = SPARE_COMM_10

	Operations Events 110 - 119
	110 = Reset Energy Meters
	111 = Clear Record
**	112 = Reset Seal-Ins
	113 = SPARE_OPS_4
	114 = SPARE_OPS_5
	115 = SPARE_OPS_6
	116 = SPARE_OPS_7
	117 = SPARE_OPS_8
	118 = SPARE_OPS_9
	119 = SPARE_OPS_10
	Recloser Events 120 - 139
**	120 = Recloser 43a Enabled
**	121 = Recloser 43a Disabled
**	122 = Recloser Enabled
**	123 = Recloser Disabled
**	124 = Recloser in Progress
**	125 = External Recloser Initiated
**	126 = Recloser Lockout
**	127 = Recloser Reset
**	128 = Recloser Out Of Sequence
**	129 = Max Recover Exceeded
**	130 = Recloser Recover
**	131 = Max Recloses Exceeded
**	132 = Max Recloser Resets
**	133 = Zone Step
**	134 = Recloser Error
**	135 = Recloser Initiated
**	136 = SPARE_RECL_17
**	137 = SPARE_RECL_18
**	138 = SPARE_RECL_19
**	139 = SPARE_RECL_20
	Internal Errors Events 140 - 141
	140 = Illegal Event Code
	141 = Event Code Error
	$$ = Added in Version 1.6 Software ** = Added in Version 1.7 Software && = Added in Version 1.8 Software

Mask Control Block (27 Registers Defined)

As described in the beginning of this section, four groups of control blocks are resident in the MSOC. Each group is comprised of 6 registers. The four control blocks defined in Table 25 require a password to control the functionality within the unit.

The method to perform control through the Mask Control Block is as follows:

- Write all registers other than the register associated with the "Execute Register"
- Write a "1" or "2" to execute the command or refresh the registers with the control command.

Groups I, II, III and IV share commonality in that an operation type must be written to the Execute Register (Register 41665 in Group I, 41671 in Group II, 41677 in Group III, and 41683 in Group IV). Also a Password must be written to the block for the desired function to execute. The registers allow for refresh and update. Figure 24 illustrates graphically the difference between the two terms. In the MSOC, all write terms are transferred to a buffer and written to the relay. In an update buffer scenario, the contents are transferred from the buffer and burned into the MSOC's internal flash RAM for retention. The memory is also updated automatically. In a refresh, the contents of memory are taken from the MSOC's internal memory and transferred to the unit's visible memory area. Thus the memory seen by the operator is " refreshed" with the latest data.

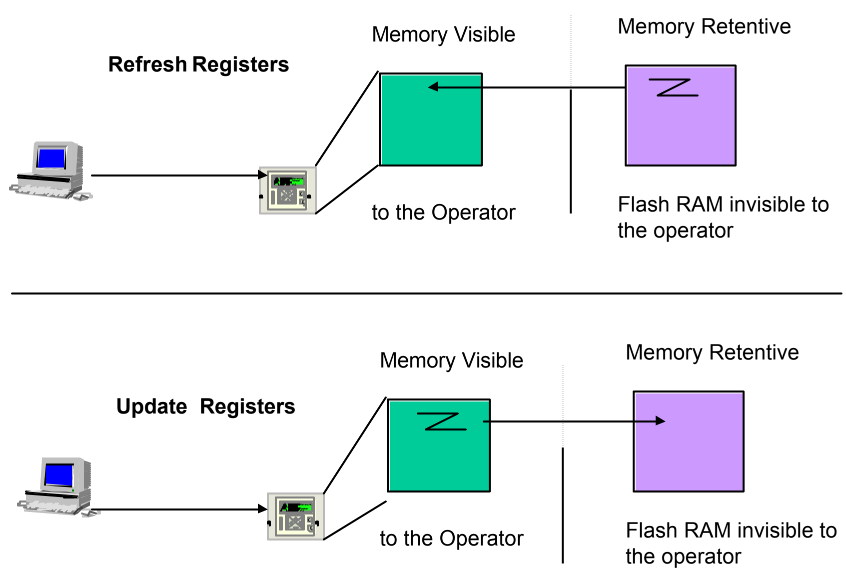


Figure 4-20. Update and Refresh Functionality

Group 1 allows for control and access of Trip and Close functionality (through bits 0 and 1 of register 41669 and 41670) within the relay. The length of trip is three cycles and length of the close is six cycles. Bit 2 resets all active targets within the relay. The function's execution can be visually seen as the LED's on the front panel of the unit extinguish upon a fault. Bits 3, 4 and 5, enable or disable the functionality of the internal protective functions. Bit 6 clears all records, Fault and Operational within the relay. Bit 7 is important in that all SCADA type control functions can be disabled via the network. If the functions are disabled, then all control is via the front panel interface or via "Torque Control" performed via the input terminals on the relay. The front RS 232 communication interface may be enabled or disabled from operation via a write. Remember, only one port on the MSOC may be active at any one time.

The unit password is set via the front panel interface of the MSOC. The default password for the unit is all spaces. The HEX code for all spaces is 20 20. Thus if the MSOC has the default password within the unit, registers 41666 and 41667 must send the HEX equivalent of the ASCII code via Modbus. The register data would be 41666 = 2020 41667 = 2020 (for Group I access for example).

If a user wished to perform a check before operate control sequence for Group I, the control could be performed in three separate write operations, Registers 41666 through 41669 written first, Register 41670 written second. Registers 41666 through 40670 checked for validity via a Modbus Read (Code 03). Finally, a Modbus Write to 41665 to initiate control would initiate control. A Check Before Operate was a common method of control before protocol checksums were the norm. Such sequences, although through the use of modern protocols are unnecessary, can still be duplicated over Modbus. The registers will retain their values for an indefinite time period. Only the execute register resets to a value of 0 for a control sequence.

The method for performing Group I functions is shown in Figures 4-21 through 4-23. Each control bit defined in Registers 41669 and 41670 has a specific task for function enable, function disable and function performance. Table 4-23 Illustrates the control bit combinations for each task.

Table 4-23. Group 1 Control Function Bit Combinations

Register 41669	Register 41670	Description
Bit 0 = 1	Bit 0 = 1	Trip Breaker (Rightmost Bit)
Bit 1 = 1	Bit 1 = 1	Close Breaker
Bit 2 = 1	Bit 2 = 1	Reset Targets
Bit 3 = 1 Bit 3 = 1	Bit 3 = 1 Bit 3 = 0	Disable 79 (Recloser Trip Function Disabled) Enable 79 (Recloser Trip Function Enabled)
Bit 4 = 1 Bit 4 = 1	Bit 4 = 1 Bit 4 = 0	Disable Ground (Ground Fault Function Disabled) Enable Ground (Ground Fault Function Enabled)
Bit 5 = 1	Bit 5 = 1	Reset Energy Meters
Bit 6 = 1	Bit 6 = 1	Clear Fault and Event Buffers
Bit 7 = 1 Bit 7 = 1	Bit 7 = 1 Bit 7 = 0	Disable Supervisory (Supervisory Network Control Disabled) Enable Supervisory (Supervisory Network Control Enabled)
Bit 8 = 1 Bit 8 = 1	Bit 8 = 1 Bit 8 = 0	Disable Front RS 232 Port (Rear RS 485 Port Enabled) Enable Front RS 232 Port (Rear RS 485 Port Disabled)
Bit 9 = 1	Bit 9 = 1	Reset Seal-In Points

Group II, allows for control of the Physical Outputs at the back of the relay. Control initiation occurs using an execute command, password and two registers (41675 and 41676). Figure 4-22 illustrates the control of the output using a specific sequence. Table 4-24 illustrates the configuration of the Change Register State register (41675) and the Force Register State register (41676).

Table 4-24. Configuration of Change Register State and Force Register State

Register 41675	Register 41676	Description
Bit X = 1	Bit X = 1	Pulse the selected output to a state of 1 for ½ second.

Group III allows for reset of each individual latched bits within the MSOC.

Group IV is reserved.

Table 4-25. Write Group Functionality

	Register Address	Item	Description
		GROUP I	
	41664	RESERVED	
	41665	Execute Register 0 = No Action 1 = Update Registers 2 = Refresh Registers	Unsigned 16 Bit
	41666	Password	ASCII – 2 Characters (Leftmost Digits)
	41667	Password	ASCII – 2 Characters (Rightmost Digits)
	41668	Spare	

	41669	Change Register Bit 0 Initiate CB Trip (Lsb) Bit 1 Initiate CB Close Bit 2 Reset Targets Bit 3 Disable 79 Bit 4 Disable Ground Bit 5 RESET Energy Meters Bit 6 Clear Records Bit 7 Disable Supervisor Bit 8 Enable Front Port Bit 9 Reset Seal Ins Bit 10 – Bit14 Reserved Bit15 Reserved (Msb)	16 Bit Unsigned Register Force Circuit Breaker Trip Force Circuit Breaker Close Reset Targets in Relay and Front Panel Recloser Trip Function Disabled Disable Ground Fault Function Reset Energy Meter. Clear Fault And Event Record Buffers Disable Network Supervisory Control Enable RS 232 Port Disable RS 485 Port 1= Enable Network Reset of Seal Ins
** (Bit 9)			
	41670	Confirmation Register	Contents of this register are ANDED with Register 41669 to perform control functionality. If Registers 41670 and 41669 are not equal control will not occur.
		GROUP II	
	41671	Execute Register 0 = No Action 1 = Update Registers 2 = Refresh Registers	16 Bit Unsigned
	41672	Password	ASCII – 2 Characters (Leftmost Digits)
	41673	Password	ASCII – 2 Characters (Rightmost Digits)
	41674	Spare	
	41675	Change Register Bit 0 Output 1 Bit 1 Output 2 Bit 2 Output 3 Bit 3 Output 4 Bit 4 – 14 Reserved Bit15 Reserved	Unsigned 16 Bit Physical Relay Output Contact Status (Terminal 17 & 18) Physical Relay Output Contact Status (Terminal 15 & 16) Available if no Voltage Inputs – Physical Relay Output Contact Status (Terminal 13 & 14) Available if no Voltage Inputs – Physical Relay Output Contact Status (Terminal 11 & 12)
	41676	Confirmation Register	Contents of this register are ANDED with Register 41675 to perform control functionality. If Registers 41675 and 41676 are not equal control will not occur.
		GROUP III	
**	41677	Execute Register 0 = No Action 1 = Update Registers 2 = Refresh Registers	16 Bit Unsigned
**	41678	Password	ASCII – 2 Characters (Leftmost Digits)
**	41679	Password	ASCII – 2 Characters (Rightmost Digits)
**	41680	Spare	
**	41681	Reset Register#1 (for Seal-Ins) Bit 0 51LT* Bit 1 51P* Bit 2 50P* Bit 3 51N* Bit 4 50N* Bit 5 PUA* Bit 6 FAIL* Bit 7 CBFAIL* Bit 8 TRIP*	Unsigned 16 Bits- Latched Data Control Phase Time Overcurrent Phase Time Overcurrent Phase Instantaneous Overcurrent Ground Time Overcurrent Ground Instantaneous Overcurrent Timeout or Trip Pickup Alarm MSOC Diagnostic Failure Circuit Breaker Fail Alarm Breaker is Tripping

		Bit 9 CLOSE* Bit 10 RCLINPROG* Bit 11 RCLALARM* Bit 12 RDA* Bit 13 RCLRMA* All other bits reserved	Breaker is Closing Recloser In Progress Recloser Lockout Alarm Recloser Disable Alarm Recloser Max Recl Alarm
**	41682	Reset Register #2 (for Seal-Ins) Reserved	Reserved (MUST BE 0)
**	41683	Confirmation Register #1	Contents of this register are ANDED with Register 41681 to perform control functionality. If Registers 41681 and 41681 are not equal control will not occur.
**	41684	Confirmation Register #2	Reserved (MUST BE 0)
		GROUP IV	
**	41685	Execute Register 0 = No Action 1 = Update Registers 2 = Refresh Registers	16 Bit Unsigned
**	41686	Password	ASCII – 2 Characters (Leftmost Digits)
**	41687	Password	ASCII – 2 Characters (Rightmost Digits)
**	41688	Spare	
**	41689	Reserved	
**	41690	Reserved	
			** = Added in Version 1.7 Software

EXAMPLE 1 - Trip the breaker via a Modbus Command Sequence.

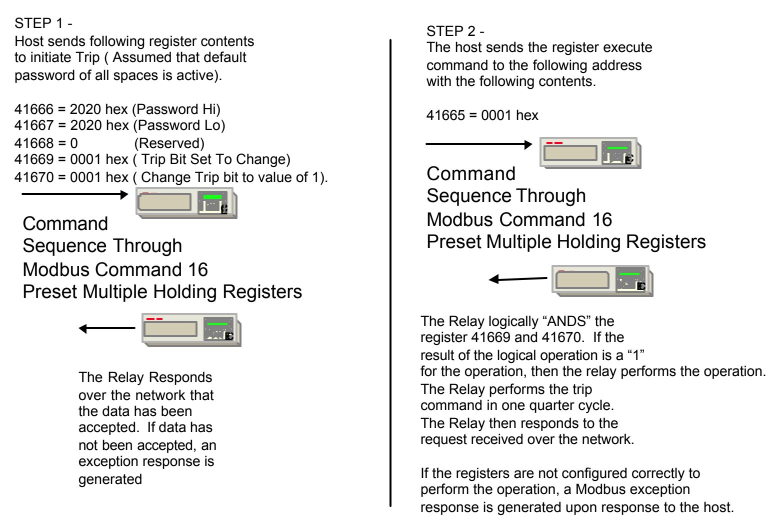


Figure 4-21. SCADA Trip Sequence Example

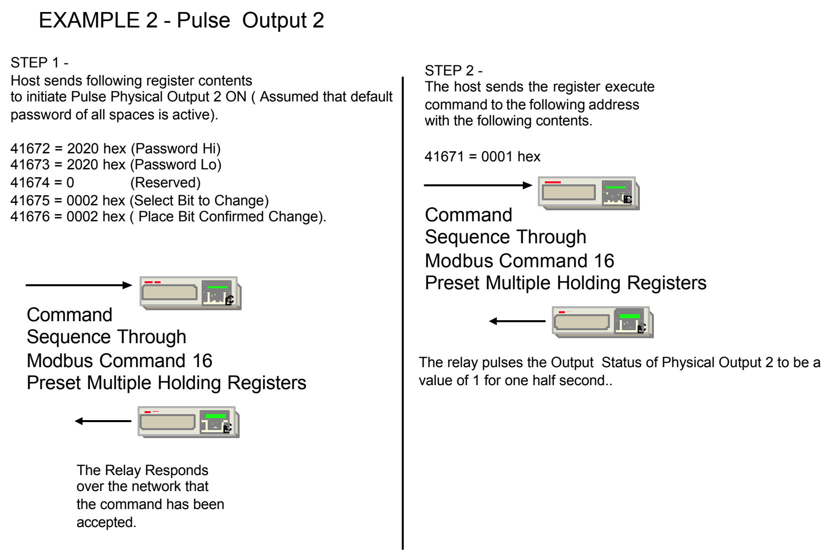


Figure 4-22. Pulse Physical Output Sequence Example

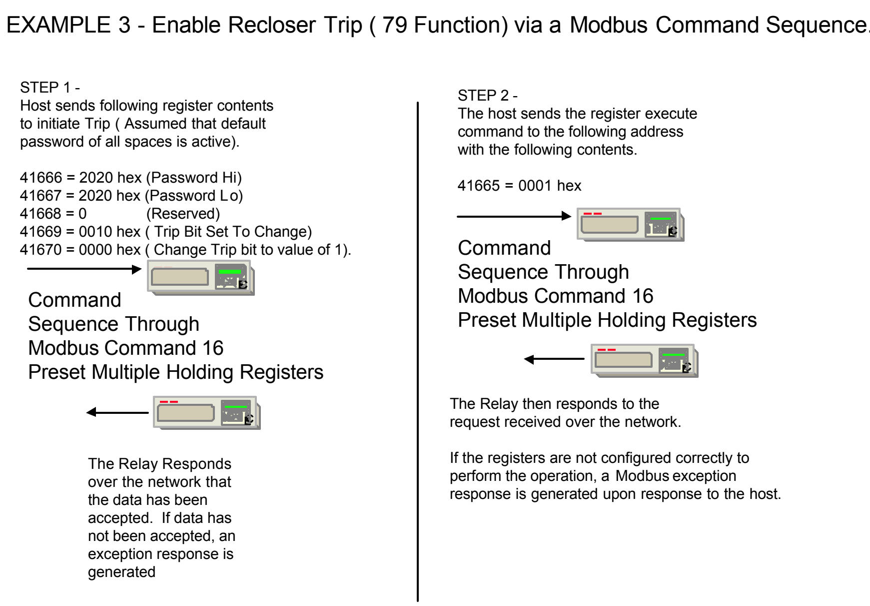


Figure 4-23. Enable Recloser Trip Example

Breaker Counters (11 Registers Defined) Modbus Function 03 Read Only

Breaker Counters allow diagnostic evaluation of operations for maintenance purposes. With Reclosing enabled (79), the counters 41796 through 41802 are updated. The MSOC allows selection of reclosure for up to 4 shots with the fifth event initiating lockout. The counter registers are reset via a write to Registers 61921 through 61936 as defined in Table 4-38. Table 4-26 defines the register map for the Breaker Counter capabilities within the unit. This feature has only been implemented in MSOC units with Version 1.6 or later firmware.

Table 4-26. Breaker Counter Definition Table

	Register Address	Item	Description
$$	41792	Total Breaker Operations	Unsigned 16 Bit (0 – 9999)
$$	41793	KSIA	Unsigned 16 Bit (0 – 9999) Kiloamps Symmetrical Ia – Current existing when breaker opened on Phase A.
$$	41794	KSIB	Unsigned 16 Bit (0 – 9999) Kiloamps Symmetrical Ib – Current existing when breaker opened on Phase B.
$$	41795	KSIC	Unsigned 16 Bit (0 – 9999) Kiloamps Symmetrical Ic – Current existing when breaker opened on Phase C.
$$	41796	First Reclose Counter	Unsigned 16 Bit (0 – 9999)
$$	41797	Second Reclose Counter	Unsigned 16 Bit (0 – 9999)
$$	41798	Third Reclose Counter	Unsigned 16 Bit (0 – 9999)
$$	41799	Fourth Reclose Counter	Unsigned 16 Bit (0 – 9999)
$$	41800	Total Recloser Counter	Unsigned 16 Bit (0 – 9999)
$$	41801	Overcurrent Trip Counter	Unsigned 16 Bit (0 – 9999) Does this update if Reclose is not initiated?
$$	41802	Successful Reclose Counter	Unsigned 16 Bit (0 – 9999)
			$$ = Added in Version 1.6 Software

6X Registers

General Relay settings are available for viewing via the MSOC front panel interface. All parameters accessible through the front panel are accessible through the Modbus 6X Registers.

A protective relay may have thousands of parameters stored in its configuration. The Modbus capable 984 programmable logic controllers have historically only defined up to 1890 or 1920 registers for access within its products. Later definitions of the Modbus Protocol and programmable logic controllers defined up to 10000 4X Registers. Even with this improvement, this amount of registers was still too limited to store the vast amount of information available for retrieval and storage within a Modbus node or protective relay.

Modbus protocol included a standard 6X Register type. The protocol defines this memory as extended memory. Modbus 6X memory is available in groups of 10,000 registers. Up to 10 groups may be defined within a node.

It is a standard ABB practice to store any configuration settings in 6X Register memory. The MSOC has all its parameters stored in Block 0 of the 6X memory definition (Blocks being defined from 0 through 9).

Function Code 20 (Read General Reference) and 21 (Write General Reference)

Modbus Protocol defines two commands 20 and 21 to read and write the registers within the 6X Register groups (or blocks). Figure 4-24 illustrates the frame sequence of Function 20 and Figure 4-25 illustrates the frame sequence of Function 21.

Function 20 Read Gen. Ref.

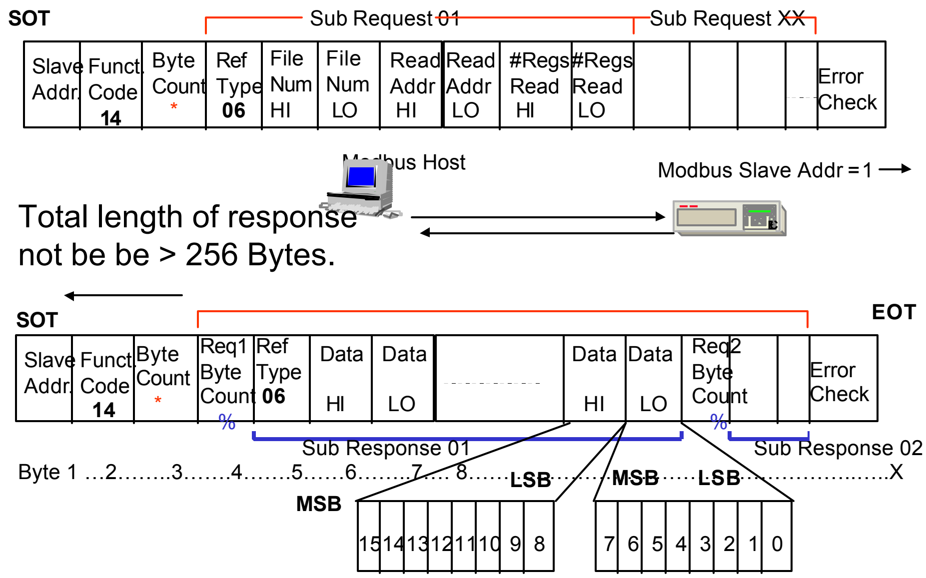


Figure 4-24. Function 20 Read 6X Register Frame Definition

Function 21 Read Gen. Ref.

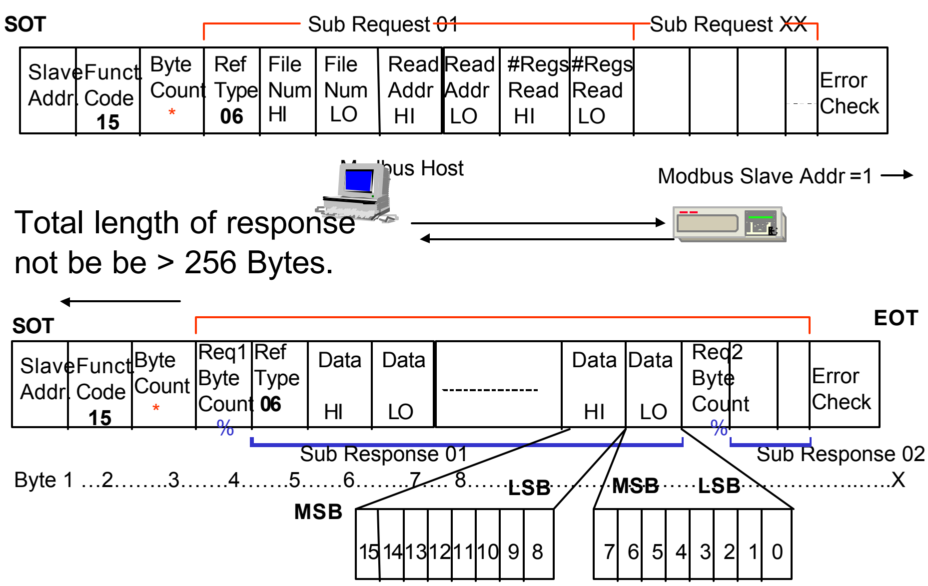


Figure 4-25. Modbus Command 21- Write General Reference Format

ÿ **IMPLEMENTATION TIP-** When 6X Registers are written, a 10 minute execute timer is initiated upon the first write to a 6X block. If the execute register is not written with a value of 1 within the 10 minutes of the initial write, the 6X viewable register segment will be restored to the original values from the MSOC's internal flash ram memory.

Relay Configuration Settings (20 Registers Defined)

The MSOC allows for query or changing of Relay Configuration Data via the Modbus ASCII protocol. Table 4-27 further describes the register assignment for viewing or changing the MSOC parameters.

Table 4-27. Relay Configuration Setting Definition

Register Address	Item	DESCRIPTION
60001	SPARE_1	
60002	Execute Register	Unsigned 16 Bit –Range 0-2 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
60003	Access Password	ASCII – 2 Characters Leftmost Digits
60004	Access Password	ASCII – 2 Characters Rightmost Digits
60005	SPARE_2	
60006	Phase CT Ratio	Unsigned 16 Bit – Range 1 – 2000:1
60007	Neutral CT Ratio	Unsigned 16 Bit – Range 1 – 2000:1
60008	Voltage PT Ratio	Unsigned 16 Bit – Range 1 – 1000:1
60009	VT Connection	Unsigned 16 Bit – Range 0 –1 0 = Wye, 1 = Delta
60010	Trip Fail Timer 1 - 60, in cycles	Unsigned 16 Bit – Range 5 – 60 Cycles Time if breaker not tripped, CB FAIL status is updated.
60011	Demand Time Constant	Unsigned 16 Bit – Range 0 – 2 Demand Average Period for Demand Calculations 0 = 15 Min, 1 = 30 Min, 2 = 60 Min
60012	Reset Mode	Unsigned 16 Bit – Range 0 – 1 Reset on Overcurrent 0 = Instantaneous, 1 = Delayed
60013	Voltage Mode	Unsigned 16 Bit – Range 0 – 1 Front Panel Display Config 0 = Line to Neutral, 1 = Line to Line
60014	Cold Load Timer Mode	Unsigned 16 Bit – Range 0 – 2 0 = Disable, 1 = Seconds, 2 = Minutes
60015	Cold Load Timer	Unsigned 16 Bit – Range 1 – 255 1 – 250 seconds
60016	Close Failure Timer	Unsigned 16 Bit – Range 3 – 999 3 – 999 cycles
60017	Spare_Config_4	
60018	Spare_Config_5	
60019	Password	ASCII – 2 Characters (Leftmost Digits)
60020	Password	ASCII – 2 Characters (Rightmost Digits)

Primary Settings (39 Registers Defined)

Primary settings are dependent upon which type of MSOC is used, IEC or ANSI. The IEC relay has a part number in the format of:

474 M XXXX- X 0 XX (X digit positions are don’t care).

The ANSI MSOC types have a part number in the format of:

474 M XXXX – X 1 XX (X digit positions are don’t care).

Version 1.69 software includes the IEC curves. ANSI nomenclature is used in the setup screens for the IEC MSOC.

The MSOC has only one settings group "PRIMARY". Each setting within the relay may be changed via the Primary Settings set of defined registers. These registers are described in Tables 4-28 through 4-30.

The settings preserved in this set of registers are curve and timing selections for the following protective elements:

- 51LT Phase Time Overcurrent Element (Long Time – mimics electromechanical relays)
- 51P Phase Time Overcurrent Element (Phase Current element)
- 51N Phase Time Overcurrent Element (Neutral Current Element)
- 50P Phase Instantaneous Overcurrent Element (Phase Current Element)
- 50N Ground Instantaneous Overcurrent Element (Ground Current Element)

Tables 4-29 through 4-31 define the protection curve codes, which are assigned to each protective element type.

NOTES FOR ASSIGNMENT OF PHASE TIME OVERCURRENT ELEMENT 51LT: If Register 60268 is assigned code 1 as defined in Table 4-29, Register 60264 defines the definite pickup time for 51 LT from 1 to 10 seconds in 0.1 second increments. For example, if

- 60262 = 1 then 60264 must be in a range from 10 through 100 (which equals 1.0 sec to 10.0 seconds). This is the definition of Time Delay Dial as indicated in Table 4-28. Resolution is 0.1 second increments.
- 60262= 0 or 2<= 60264 <=8, then 60624 must be in a range from 10 through 120 (which equals 10 through 120 seconds). This is the definition of Time Delay as indicated in Table 4-28. Resolution is in 10 second increments**.**

NOTE – Tdelay is in the range of 0.05 to 1.00 in increments of 0.05 for IEC firmware Version 1.69 only.

NOTES FOR ASSIGNMENT OF PHASE TIME OVERCURRENT ELEMENT 51P: If Register 60268 is assigned a code of 1 or 13 as defined in Table 4-30, Register 60270 defines the definite pickup time for 51P from 1 to 10 seconds in 0.1 second increments. For example:

- 60268 = 1 or 13 then 60270 must be in a range from 10 through 100 (which equals 1.0 sec to 10.0 seconds). This is the definition of Time Delay Dial as indicated in Table 4-28. Resolution is 0.1 second increments.
- 60268= 0 or 2<= 60268 <=12, then 60270 must be in a range from 0 through 300 (which equals 0.0 through 30.0 seconds). This is the definition of Time Delay as indicated in Table 4-28. Resolution is in 0.1 second increments.

NOTE – Tdelay is in the range of 0.05 to 1.00 in increments of 0.05 for IEC firmware Version 1.69 only.

NOTES FOR ASSIGNMENT OF GROUND TIME OVERCURRENT ELEMENT 51N: If Register 60280 is assigned a code of 1 or 13 as defined in Table 4-30, Register 60282 defines the definite pickup time for 51N from 1 to 10 seconds in 0.1 second increments. For example:

- 60280 = 1 or 13 then 60282 must be in a range from 10 through 100 (which equals 1.0 sec to 10.0 seconds). This is the definition of Time Delay Dial as indicated in Table 4-28. Resolution is 0.1 second increments.
- 60280= 0 or 2<= 60280 <=12, then 60282 must be in a range from 0 through 300 (which equals 0.0 through 30.0 seconds). This is the definition of Time Delay as indicated in Table 4-28. Resolution is in 0.1 second increments.

NOTE – Tdelay is in the range of 0.05 to 1.00 in increments of 0.05 for IEC firmware Version 1.69 only.

NOTES FOR ASSIGNMENT OF GROUND INSTANTANEOUS OVERCURRENT ELEMENT 50N: If Register 60286 is assigned a code of 1 as defined in Table 4-31, Register 60288 defines the definite pickup time for 50N from 1 to 10 seconds in 0.1 second increments. For example:

- 60286 = 1 or 4 then 60288 must be in a range from 10 through 100 (which equals 1.0 sec to 10.0 seconds). This is the definition of Time Delay Dial as indicated in Table 4-31. Resolution is 0.1 second increments.
- 60286= 0 or 2<= 60286 <=3, then 60288 must be in a range from 0 through 300 (which equals 0.0 through 30.0 seconds). This is the definition of Time Delay as indicated in Table 4-31. Resolution is in 0.1 second increments.

NOTE – Tdelay is in the range of 0.00 to 3.00 in increments of 0.01 for IEC firmware Version 1.69 only.

NOTES FOR ASSIGNMENT OF PHASE INSTANTANEOUS OVERCURRENT ELEMENT 50P: If Register 60274 is assigned a code of 1 as defined in Table 4-31, Register 60276 defines the definite pickup time for 51P from 1 to 10 seconds in 0.1 second increments. For example:

- 60274 = 1 or 4 then 60276 must be in a range from 10 through 100 (which equals 1.0 sec to 10.0 seconds). This is the definition of Time Delay Dial as indicated in Table 4-31. Resolution is 0.1 second increments.
- 60274= 0 or 2<= 60276 <=3, then 60282 must be in a range from 0 through 300 (which equals 0.0 through 30.0 seconds). This is the definition of Time Delay as indicated in Table 4-31. Resolution is in 0.1 second increments.

NOTE – Tdelay is in the range of 0.00 to 3.00 in increments of 0.01 for IEC firmware Version 1.69 only.

Table 4-28. Primary Settings Register Definition

Register Address	Item	Description
60257	SPARE_1	
60258	Execute Register	Unsigned 16 Bits 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
60259	Access Password	ASCII – 2 Characters (Leftmost Digits)
60260	Access Password	ASCII – 2 Characters (Rightmost Digits)
60261	SPARE_2	
60262	51LTCurve (Type I)	Unsigned 16 Bits – Range 0 – 8 (See Table 4-23)
60263	51LTPickupAmps	Unsigned 16 Bits – Range 30 – 240 Divide by 100 for front panel value.
60264	51LTTimeDelayDial	Unsigned 16 Bits 10 – 100 / 10 – 120. SEE NOTES PRECEEDING TABLE
60265	SPARE1_51LT	Spare
60266	SPARE2_51LT	Spare
60267	SPARE3_51LT	
60268	51PCurve (Type II)	Unsigned 16 Bits – Range 0 – 8 (See Table 4-23)
60269	51PPickupAmps	Unsigned 16 Bits – Range 30 – 240. 30 – 240. Divide by 100 for front panel value.
60270	51PTimeDelayDial	Unsigned 16 Bits 10 – 100 / 0 – 300. SEE NOTES PRECEEDING TABLE
60271	SPARE1_51P	
60272	SPARE2_51P	
60273	SPARE3_51P	
60274	50PCurve (Type III)	Unsigned 16 Bits – Range 0 – 4 (See Table 4-24)
60275	50PPickupXAmps	Unsigned 16 Bits – Range 1– 200 10 – 200. Divide by 10 for front panel value.
60276	50PTimeDelayDial	Unsigned 16 Bits 10 – 100 / 0 – 300. SEE NOTES PRECEEDING TABLE
60277	SPARE1_50P	
60278	SPARE2_50P	
60279	SPARE3_50P	
60280	51NCurve (Type II)	Unsigned 16 Bits – Range 0 –13 (See Table 4-23)
60281	51NPickupAmps	Unsigned 16 Bits – Range 30 – 240 30 – 240. Divide by 100 for value.
60282	51NTimeDelayDial	Unsigned 16 Bits 10 – 100 / 0 – 300. SEE NOTES PRECEEDING TABLE
60283	SPARE1_51N	
60284	SPARE2_51N	
60285	SPARE3_51N	
60286	50NCurve (Type III)	Unsigned 16 Bits – Range 0 – 4 (See Table 4-25)
60287	50NPickupXAmps	Unsigned 16 Bits – Range 0 – 200 0 – 200. Divide by 10 for value. Multiply * 51 N for pickup setting.
60288	50NTimeDelayDial	Unsigned 16 Bits 10 – 100 / 0 – 300. SEE NOTES PRECEEDING TABLE
60289	SPARE1_50N	
60290	SPARE2_50N	
60291	SPARE3_50N	
60292	SPARE_PRIM1	
60293	SPARE_PRIM2	
60294	SPARE_PRIM3	
60295	SPARE_PRIM4	

Table 4-29. Curve Selection Definition for 51LT (Register 60262)

	Curve Selection Type I (51LT)
	0 = Disable
	1 = Definite
	2 = Long Time Extremely Inverse
	3 = Long Time Very Inverse
	4 = Long Time Inverse
	5 = Very Long Time Ext. Inverse
	6 = Very Long Time Very Inverse
	7 = Very Long Time Inverse
**	8 = IEC – Long Time Inverse
	** = Added in Version 1.7 Software

Table 4-30. Curve Selection Definition for 51N and 51P (Registers 60274 and 60287)

	Curve Selection Type II (51 Curves)
	0 = Disable
	1 = Definite
	2 = Long Time Extremely Inverse
	3 = Long Time Very Inverse
	4 = Long Time Inverse
	5 = Extremely Inverse
	6 = Very Inverse
	7 = Inverse
	8 = Short Time Inverse
**	9 = IEC – Extremely Inverse
**	10 = IEC – Very Inverse
**	11 = IEC – Inverse
**	12 = IEC – Long Time Inverse
**	13 = Definite Time #2
	** = Added in Version 1.7 Software

Table 4-31. Curve Selection Definition for 50N and 50P (Registers 60274 and 60287)

	Curve Selection Type III (50 Curves)
	0 = Disable
	1 = Definite
	2 = Standard
	3 = Short Time Inverse
**	4 = Definite Time #2
	** = Added in Version 1.7 Software

Aux Communication Settings (22 Registers Defined)

This set of defined registers allows for monitoring or changing the communication port parameters assigned to a specific port. An operator should be aware that changing these ports will immediately effect the internal MSOC communication hardware and cause a communication error. The host's communication port should be reconfigured to reflect the changes made to the unit's port parameters. Only one port may be enabled at any time. Table 4-32 lists and defines the memory mapping for Modbus.

Register 60786 is the TX_DLY time. This is a unique register in that the MSOC can respond to a Modbus request rapidly (within 3 mS). Some hosts may not see such a rapid response in that the hardware is unable to process communication requests at that speed. TX_DLY shall pause the command response by the amount of milliseconds contained within that register.

Table 4-32. Communication Setting Register Definition

	Register Address	Item	Description
	60769	SPARE_1	
	60770	Execute Register	Unsigned 16 Bit 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
	60771	Access Password	ASCII – 2 Characters (Leftmost Digits)
	60772	Access Password	ASCII – 2 Characters (Rightmost Digits)
	60773	SPARE_2	
	60774	MSOC Network Comm Address	Unsigned 16 Bits – Range 1 – 247
	60775	Font Panel Baud Rate (RS 232)	Unsigned 16 Bits – Range 0 – 4 0 = 1200 Baud, 1 = 2400 Baud, 2 = 4800 Baud, 3 = 9600 Baud, 4 = 19200 Baud
	60776	Front Panel Frame (RS 232)	Unsigned 16 Bits – Range 0 – 4 0 = O71, 1 = E71, 2 = N81, 3 = O72, 4 = E72
	60777	Front Panel Protocol (RS 232)	Unsigned 16 Bits – Range 0 – 1 0 = Native MSOC ASCII, 1 = Modbus ASCII
	60778	Front Panel Local Echo (RS 232)	Unsigned 16 Bits – Range 0 – 1 0 = No Echo, 1 = Local Echo
	60779	FP_SPARE1	
	60781	FP_SPARE2	
	60781	FP_SPARE2	
	60782	FP_SPARE4	
	60783	Rear Panel Baud Rate (RS 485)	Unsigned 16 Bits – Range 0 – 4 0 = 1200 Baud, 1 = 2400 Baud, 2 = 4800 Baud, 3 = 9600 Baud, 4 = 19200 Baud
	60784	Rear Panel Frame	Unsigned 16 Bits – Range 0 – 4 0 = O71, 1 = E71, 2 = N81, 3 = O72, 4 = E72
	60785	Rear Panel Protocol	Unsigned 16 Bits – Range 0 – 2 0 = Native MSOC ASCII, 1 = Modbus ASCII
$$	60786	Tx Delay Timer	Unsigned 16 Bits – Range 0 – 200 0. 200 ms in 5 ms increments
	60787	RP_SPARE1	
	60788	RP_SPARE2	
	60789	RP_SPARE2	
	60790	RP_SPARE4	
			$$ = Added in Version 1.6 Software

Real Time Clock (13 Registers Defined)

The real time clock data can be set via the network. This clock is the master which is used to time stamp operational records and event records in Registers 41029 through 41036 (as defined in Table 4-14) and Registers 41284 through 41291 (as defined in Table 4-18). It should be noted that the clock registers have been updated to reflect the four digit year required for Y2K compliance in time reporting.

If the month is set to 0, the real time clock is disabled. **The real time clock cannot be enabled or disabled via Modbus. The real time clock may only be disabled via MSOC Native ASCII or MSOC Fast ASCII**.

Table 4-33 lists the register definition for Real Time Clock configuration.

Table 4-33. Real Time Clock Register Definition Assignment

	Register Address	Item	Description
	61025	SPARE_1	
	61026	Execute Register	Unsigned 16 Bits 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
	61027	Access Password	ASCII – 2 Characters (Leftmost Digits)
	61028	Access Password	ASCII – 2 Characters (Rightmost Digits)
	61029	SPARE_2	
	61030	Month	Unsigned 16 Bit Month Range 1 – 12
	61031	Day	Unsigned 16 Bit Day Range 1 – 31
$$	61032	Year (Accepts 2 or 4 digits)	Unsigned 16 Bit Year Range 00 – 3000
	61033	Hour	Unsigned 16 Bit Hour Range 0 – 23
	61034	Minute	Unsigned 16 Bit Minute Range 0 – 59
	61035	Second	Unsigned 16 Bit Second Range 0 – 59
	61036	Hundredths of a Second	Unsigned 16 Bit Hundredth Second Range 0 – 999
	61037	Thousandths of a Second	Unsigned 16 Bit Thousands of Second Range 0 – 9999
			$$ = Added in Version 1.6 Software

Logical Input Map (19 Registers Defined)

The MSOC has physical input terminals (IN1 and IN2). These terminals can be mapped to protection elements. The inputs may be logically “AND ed” or “OR ed” or assigned to the input or output. Each Register 61286 through 61299 may be assigned a value of 0 through 4 indicating a mapped state as indicated in Table 4-34. Inputs 52a and 52b are mapped as factory defaults to input 1 and input 2 respectively. Table 4-34 indicates the elements, which may be mapped to each physical input.

The Physical Input Terminals are:

IN 1 = Terminal 5
IN 2 = Terminal 6
INPUT COMMON = Terminal 4

Table 4-34. Logical Input Mapping Definition

	Register Address	Item	Description
	61281	SPARE_1	
	61282	Execute Register	Unsigned 16 Bits 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
	61283	Access Password	ASCII – 2 Characters (Leftmost Digits)
	61284	Access Password	ASCII – 2 Characters (Rightmost Digits)
	61285	SPARE_2	
	61286	51LT	Phase Time Overcurrent
	61287	51P	Phase Time Overcurrent
	61288	50P	Phase Instantaneous Overcurrent
	61289	51N	Ground Time Overcurrent
	61290	50N	Ground Instantaneous Overcurrent
	61291	52a	Relay Status (52a = 1, 52b = 0 CB Open)
	61292	52b	Relay Status (52a = 0, 52b = 1 CB Closed)
	61293	TRIP	Relay Present State is Tripping
	61294	CLOSE	Relay Present State is Closing
$$	61295	43a	Enable Reclose Function Asserted
$$	61296	EXTRI	External Reclose Initiate Asserted
$$	61297	ZSC	Zone Sequence Closing Function State (79)
$$	61298	DTL	Drive To Lockout Asserted
**	61299	RSI	Reset Seal Ins
			$$ = Added in Version 1.6 Software ** = Added in Version 1.7 Software

Table 4-35. Logical Input Codes for Mapping of Registers 61286-61299

	Logical Mappings
	0 = Unmapped
	1 = IN1
	2 = IN2
**	3 = IN1 OR IN2
**	4 = IN1 & IN2
** = Version 1.90	

Logical Outputs (19 Registers Defined)

The MSOC contains 3 or 5 physical outputs. This number varies depending upon the unit's inclusion of VT inputs. If no VT inputs are within the MSOC, then 5 physical outputs are on the unit. If a VT input is included on the unit, then 3 physical outputs are on the MSOC. One of the outputs is permanently assigned to TRIP, 2 or 4 of the physical outputs are then mappable to internal protective element outputs. Table 4-36 lists the elements, which may be assigned to the MSOC physical outputs. Table 4-34 lists the assignment codes for each output. A list of the MSOC model numbers follows:

474 MXX1X–XXX0:X = Don't Care: Include 1 Trip and 4 Programmable Output Contacts
474 M XX2X–XXX0:X = Don't Care: Includes 1 Trip and 2 Programmable Output Contacts–Voltage Inputs

The MicroShield Over Current Relay terminal definition is as such:

Master Trip – Terminals 19 and 20
OUT 1 – Terminals 17 and 18
OUT 2 – Terminals 15 and 16
OUT 3 – Terminals 13 and 14
OUT 4 – Terminals 11 and 12

Table 4-36. Logical Output Map Register Definition

	Register Address	Item	Description
	61537	SPARE_1	
	61538	Execute Register	Unsigned 16 Bits 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
	61539	Access Password	ASCII – 2 Characters (Leftmost Digits)
	61540	Access Password	ASCII – 2 Characters (Rightmost Digits)
	61541	SPARE_2	
	61542	51LT	Phase Time Overcurrent
	61543	51P	Phase Time Overcurrent
	61544	50P	Phase Instantaneous Overcurrent
	61545	51N	Ground Time Overcurrent
	61546	50N	Ground Instantaneous Overcurrent
	61547	PUA	Timeout or Trip Pickup Alarm
	61548	FAIL	MSOC Diagnostic Failure
	61549	CBFAIL	Circuit Breaker Fail Alarm
	61551	TRIP	Breaker is Tripping
	61552	CLOSE	Breaker is Closing
$$	61553	RCLINPROG	Recloser In Progress
$$	61554	RCLALARM	Recloser Lockout Alarm
$$	61555	RDA	Recloser Disable Alarm
$$	61556	RCLRMA	Recloser Max Recl Alarm
			$$ = Added in Version 1.6 Software

Table 4-37. Logical Output Map Logical Mapping Definition for Registers 61542 Through 61556

Logical Mappings
0 = Unmapped
1 = OUT1
2 = OUT2
3 = OUT3
4 = OUT4

Master Trip (10 Registers Defined)

The Master Trip function allows internal relay elements to energize the logical MSOC Master Trip. If the following registers are set to a value of 1, the element written is mapped to the Master Trip logical function. The Master Trip is hard wired to a physical trip output at the rear terminals of the MSOC. Status of the master trip assignment may be accomplished via reading the 6X Registers defined in Table 4-38 through a refresh. Status may also be accomplished via a Modbus 03 function (Read 4X Registers), as described in Table 4-13.

Table 4-38. Master Trip Assignment Definition Table

Register Address	Item	Description
61665	SPARE_1	
61666	Execute Register	Unsigned 16 Bits 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
61667	Access Password	ASCII – 2 Characters (Leftmost Digits)
61668	Access Password	ASCII – 2 Characters (Rightmost Digits)
61669	SPARE_2	
61670	Master 51LT	Phase Time Overcurrent (Long Time)
61671	Master 51P	Phase Time Overcurrent
61672	Master 50P	Phase Instantaneous Overcurrent
61673	Master 51N	Ground Time Overcurrent
61674	Master 50N	Ground Instantaneous Overcurrent
		0 = OFF 1 = ON

Recloser Settings (69 Registers Defined)

MSOC recloser capabilities were introduced in Version 1.60 software. Table 4-39 lists the register definitions for each of the Recloser parameters. The ABB implementation allows for multiple shots of reclosing. EXTRI input must be mapped to a Physical Input.

Also the 52a and or 52b contact must be mapped to an input contact for reclosing to function. Time overcurrent and instantaneous overcurrent (50 and 51) functions may be enabled, disabled or assigned "Lock Out" status as illustrated in the table.

Table 4-39. Recloser Assignment Definition Table

	Register Address	Item	Description
$$	61793	SPARE_1	
$$	61794	Execute Register	Unsigned 16 Bits 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
$$	61795	Access Password	ASCII – 2 Characters (Leftmost Digits)
$$	61796	Access Password	ASCII – 2 Characters (Rightmost Digits)
$$	61797	SPARE_2	
$$	61798	Recloser R1 Open Time	Unsigned 16 Bits – Range 1 – 2000 0.1 – 200, Lockout. Divide by 10 to obtain front panel value in seconds. (79-1)
$$	61799	Recloser R1 50P Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Phase 0 = Enable, 1 = Disable 2 = Lockout
$$	61800	Recloser R1 50N Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61801	Recloser R1 51N Select	Unsigned 16 Bit Phase Time Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61802	Recloser R1 51LT Select	Unsigned 16 Bit Phase Time Overcurrent Long Time 0 = Enable, 1 = Disable, 2 = Lockout
$$	61803	Recloser R1 51P Lockout	Unsigned 16 Bit Phase Time Overcurrent Phase 0 = Enable, 1 = Disable, 2 = Lockout

$$	61804	Recloser R1 SPARE1	
$$	61805	Recloser R1 SPARE2	
$$	61806	Recloser R1 SPARE3	
$$	61807	Recloser R1 SPARE4	
$$	61808	Recloser R1 SPARE5	
$$	61809	Recloser R2 Open Time	Unsigned 16 Bits – Range 1 – 2000 0.1 – 200, Lockout. Divide by 10 to obtain front panel value in seconds. (79-2)
$$	61810	Recloser R2 50P Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Phase 0 = Enable, 1 = Disable, 2 = Lockout
$$	61811	Recloser R2 50N Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61812	Recloser R2 51N Select	Unsigned 16 Bit Phase Time Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61813	Recloser R2 51LT Select	Unsigned 16 Bit Phase Time Overcurrent Long Time 0 = Enable, 1 = Disable, 2 = Lockout
$$	61814	Recloser R2 51P Lockout	Unsigned 16 Bit Phase Time Overcurrent Phase 0 = Enable, 1 = Disable, 2 = Lockout
$$	61815	Recloser R2 SPARE1	
$$	61816	Recloser R2 SPARE2	
$$	61817	Recloser R2 SPARE3	
$$	61818	Recloser R2 SPARE4	
$$	61819	Recloser R2 SPARE5	
$$	61820	Recloser R3 Open Time	Unsigned 16 Bits – Range 1 – 2000 0.1 – 200, Lockout. Divide by 10 to obtain front panel value in seconds. (79-3)
$$	61821	Recloser R3 50P Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Phase 0 = Enable, 1 = Disable, 2 = Lockout
$$	61822	Recloser R3 50N Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61823	Recloser R3 51N Select	Unsigned 16 Bit Phase Time Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61824	Recloser R3 51LT Select	Unsigned 16 Bit Phase Time Overcurrent Long Time 0 = Enable, 1 = Disable, 2 = Lockout
$$	61825	Recloser R3 51P Lockout	Unsigned 16 Bit Phase Time Overcurrent Phase 0 = Enable, 1 = Disable, 2 = Lockout
$$	61826	Recloser R3 SPARE1	
$$	61827	Recloser R3 SPARE2	
$$	61828	Recloser R3 SPARE3	
$$	61829	Recloser R3 SPARE4	
$$	61830	Recloser R3 SPARE5	
$$	61831	Recloser R4 Open Time	Unsigned 16 Bits – Range 1 – 2000 0.1 – 200, Lockout. Divide by 10 to obtain front panel value in seconds. (79-4)

$$	61832	Recloser R4 50P Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Phase 0 = Enable, 1 = Disable, 2 = Lockout
$$	61833	Recloser R4 50N Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61834	Recloser R4 51N Select	Unsigned 16 Bit Phase Time Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61835	Recloser R4 51LT Select	Unsigned 16 Bit Phase Time Overcurrent Long Time 0 = Enable, 1 = Disable, 2 = Lockout
$$	61836	Recloser R4 51P Lockout	Unsigned 16 Bit Phase Time Overcurrent Phase 0 = Enable, 1 = Disable, 2 = Lockout
$$	61837	Recloser R4 SPARE1	
$$	61838	Recloser R4 SPARE2	
$$	61839	Recloser R4 SPARE3	
$$	61840	Recloser R4 SPARE4	
$$	61841	Recloser R4 SPARE5	
$$	61842	Recloser R5 50P Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Phase 0 = Enable, 1 = Disable, 2 = Lockout
$$	61843	Recloser R5 50N Select	Unsigned 16 Bit Phase Instantaneous Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61844	Recloser R5 51N Select	Unsigned 16 Bit Phase Time Overcurrent Neutral 0 = Enable, 1 = Disable, 2 = Lockout
$$	61845	Recloser R5 51LT Select	Unsigned 16 Bit Phase Time Overcurrent Long Time 0 = Enable, 1 = Disable, 2 = Lockout
$$	61846	Recloser R5 SPARE1	
$$	61847	Recloser R5 SPARE2	
$$	61848	Recloser R5 SPARE3	
$$	61849	Recloser R5 SPARE4	
$$	61850	Recloser R5 SPARE5	
$$	61851	Recloser Reset Time	Unsigned 16 Bits – Range 3-200 Seconds
$$	61852	Recloser Cutout Time	Unsigned 16 Bits – Range 1 – 200 1 – 200, 0 = Disable
$$	61853	Recloser Max Recovery Timer	Unsigned 16 Bits – Range 1 – 9999 0 = Disable
$$	61854	Recloser Limit Counter	Unsigned 16 Bits – Range 1 – 99
$$	61855	Recloser Limit Timer	Unsigned 16 Bits – Range 1 – 9999
$$	61856	Maximum Reclosures To Lockout	Unsigned 16 Bits – Range 1 – 9999 0 = Disable
$$	61857	SPARE_RECL1	
$$	61858	SPARE_RECL2	
$$	61859	SPARE_RECL3	
$$	61860	SPARE_RECL4	
$$	61861	SPARE_RECL5	
		$$ = Added in Version 1.6 Software	

Breaker Counters (16 Registers Defined)

MSOC has the ability to count breaker operations in a variety of modes. The same registers can be accessed via a Modbus Code 03 (Read Holding Registers). For 4X read access, refer to Table 4-20. The same information can be read via the refresh register capability through Register 61922. To reset the Breaker Counters, write the value of 0 to Registers 61926 through 61936.

Table 4-40. Breaker Counter Register Assignment

Register Address	Item	Description
61921	SPARE_1	
61922	Execute Register	Unsigned 16 Bit 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
61923	Access Password	ASCII – 2 Characters (Leftmost Digits)
61924	Access Password	ASCII – 2 Characters (Rightmost Digits)
61925	SPARE_2	
61926	Total Breaker Operations	Unsigned 16 Bit (0 – 9999)
61927	KSIA	Unsigned 16 Bit (0 – 9999) Kiloamps Symmetrical Ia – Current existing when breaker opened on Phase A.
61928	KSIB	Unsigned 16 Bit (0 – 9999) Kiloamps Symmetrical Ib – Current existing when breaker opened on Phase B.
61929	KSIC	Unsigned 16 Bit (0 – 9999) Kiloamps Symmetrical Ic – Current existing when breaker opened on Phase C.
61930	First Reclose Counter	Unsigned 16 Bit (0 – 9999)
61931	Second Reclose Counter	Unsigned 16 Bit (0 – 9999)
61932	Third Reclose Counter	Unsigned 16 Bit (0 – 9999)
61933	Fourth Reclose Counter	Unsigned 16 Bit (0 – 9999)
61934	Total Recloser Counter	Unsigned 16 Bit (0 – 9999)
61935	Overcurrent Trip Counter	Unsigned 16 Bit (0 – 9999)
61936	Successful Reclose Counter	Unsigned 16 Bit (0 – 9999)

Seal-In Mapping Table (20 Registers Defined)

Seal-In Mapping is available in Version 1.70 or greater. Each seal in element may be mapped to a physical output. Table 4-41 lists the register definition for each of the elements. The code written to each of the elements to assign them to the Physical Output is shown in Table 4-42.

Table 4-41. Seal In Output Definition Table

Register Address	Item	Description
**62049	SPARE_1	
**62050	Execute Register	Unsigned 16 Bits 0 = No Action, 1 = Update Registers, 2 = Refresh Registers
**62051	Access Password	ASCII – 2 Characters (Leftmost Digits)
**62052	Access Password	ASCII – 2 Characters (Rightmost Digits)
**62053	SPARE_2	
**62054	51LT*	Phase Time Overcurrent
**62055	51P*	Phase Time Overcurrent
**62056	50P*	Phase Instantaneous Overcurrent
**62057	51N*	Ground Time Overcurrent
**62058	50N*	Ground Instantaneous Overcurrent
**62059	PUA*	Timeout or Trip Pickup Alarm
**62060	FAIL*	MSOC Diagnostic Failure
**62061	CBFAIL*	Circuit Breaker Fail Alarm

**62062	TRIP*	Breaker is Tripping
**62063	CLOSE*	Breaker is Closing
**62064	RCLINPROG*	Recloser In Progress
**62065	RCLALARM*	Recloser Lockout Alarm
**62066	RDA*	Recloser Disable Alarm
**62067	RCLRMA*	Recloser Max Recl Alarm
		** = Added in Version 1.7 Software

Table 4-42. Seal In Output Mapping Code Definition Table

Logical Mappings
0 = Unmapped
1 = OUT1
2 = OUT2
3 = OUT3

The easiest method to initiate communications in the Modbus protocol is to read known discrete and register data. As per the MSOC Modbus Register documentation, the unit catalog number is resident at Register 40132. A list of the register definitions of the MSOC is presented and explained in the next section. A Read Holding Register Modbus command is explained. Documentation is available from Groupe Schneider further describing the Modbus ASCII emulation characteristics. The explanation contained within this document is intended to be a quick start guide to communication initiation.

The length of the catalog number is 12 characters or 6 registers. The following command string format, when sent will retrieve the catalog number from the unit.

: 01 03 00 83 00 06 73 lf cr

The above string in Modbus ASCII format should be sent:

3A 30 31 30 33 30 30 38 33 30 30 30 36 37 33 0D 0A

The string is translated as such:

Colon (in HEX) , unit address = 01 (in HEX), Read Holding Registers (Code 3 in HEX), data memory address –1 = 131 decimal (0083 in HEX), number of registers read = 6 (0006 in HEX), message calculated LRC code 73 (37 33), and line feed (0D) and (0A).

A typical response shall include the following:

: Address number (01), Read Holding Registers command (Code 3 in HEX), Byte Count Returned 12 in decimal (OC in HEX), Data Register 40132 = 47, Data Register 40133= 4M, Data Register 40134 = 14 Data Register 40135 = 11, Data Register 40136 = 60 Data Register 40137 = 10, and calculated LRC =77 (HEX) and line feed with carriage return (0D 0A).

The aforementioned response would be returned as such:

3A 30 31 30 33 30 43 34 37 34 4D 31 34 31 31 36 30 31 30 37 37 0A 0D.

Calculation of the LRC (Longitudinal Redundancy Code)

Modbus ASCII protocol uses a Longitudinal Redundancy Code to verify correct reception of the command. This error check is used in addition to the parity option (used by the UART in the PC) and other data such as the byte count which verifies data returned. The process for calculation of the checksum is described as such:

1. Add all bytes in the message except for the colon, line feed, and carriage return. Exclude the LRC checksum, which in included in the message structure.
2. Invert all bits in the word after the addition.
3. Add 1 to the inverted result. This is the checksum.

An example is as follows:
Command sent:

3A 30 31 30 33 30 30 38 33 30 30 30 36 37 33 0D 0A

Decode of the data from ASCII to HEX yields.

: 01 03 00 83 00 06 73 lf cr

The decoded LRC checksum is 73. The calculation of the checksum is as such:

1. Neglect the colon (3A) and the lf (Line Feed 0A) and cr (Carriage Return OD). This decreases the string to 01 03 00 83 06 73.
2. The LRC checksum 73 (37 33 in ASCII) should also be saved for comparison to the original data string. The string for LRC calculation is 01 03 00 83 00 06.
3. The byte data should be added thus 01 + 03 + 00 +83 + 00 + 06 = 8D in HEX. Notice that the bytes have been decoded from ASCII before performing the addition.
4. A Two's compliment must be performed on the number to determine the LRC Checksum. Inversion of the number 8D hex yields 72 HEX.
5. To complete the Two's compliment addition for accurate compilation of the checksum 1 HEX must be added to the inverted bits to yield 72 + 1 = 73 HEX. Thus the two calculated values agree.

Please reference the Modicon Modbus documentation for additional command configuration on each data type (0X, 1X, 4X and 6X read write capabilities).

MSOC Modbus Exception Response Analysis

If the MSOC does not understand the command sent to the device or if the command is sent in the wrong format, the MSOC shall generate an exception response. A Modbus exception response is in the format of that shown in Figure 4-26. As illustrated, the function code is "ANDed" with 80 HEX. Following the modified function code, an exception code byte will follow. The customary LRC and terminator of a Carriage Return and Line feed will terminate the communication string.

Table 4-43 shall list the standard Modbus Exception Codes and Table 4-45 lists the exception codes as the MSOC reports them. Notice that the MSOC does not report its exception codes as per the Modbus standard defined codes.

Table 4-43. Modbus Standard Exception Codes

Code	Name
01	Illegal Function
02	Illegal Data Address
03	Illegal Data Value
05	Acknowledge
06	Slave Device Busy
07	Negative Acknowledge
08	Memory Parity Error

Table 4-44. MSOC Defined Exception Codes

Code	Description
01	Invalid Password
04	Invalid Register Address
05	Invalid Range Accessed
06	Invalid Data
34	Invalid Function Code
36	Supervisory Control Disabled

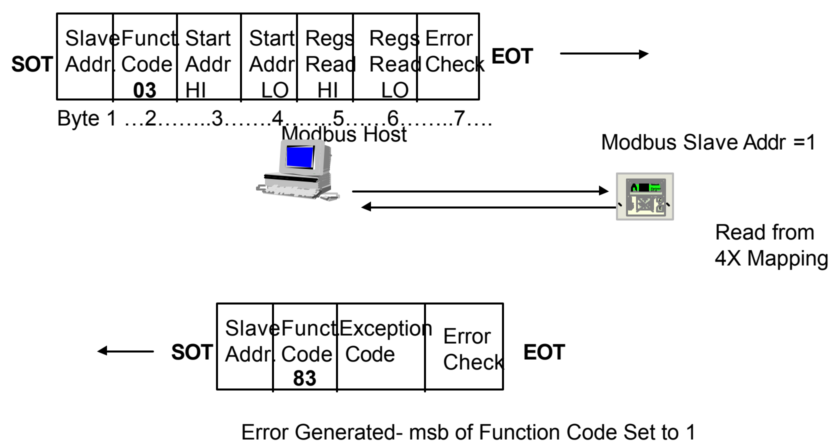


Figure 4-26. Exception Code Example for Holding Register Read

MSOC Modbus Communication Timing Analysis

Perhaps the most common error in implementing a Modbus network is timing setup for communication. Modbus protocol operates according to the following timing rules:

- If the MSOC **receives** a command without a communication error (LRC, PARITY, FRAMING, OVERRUN… errors), a normal response occurs.
- If the MSOC **does not receive** a command without a communication error (LRC, PARITY, FRAMING, and OVERRUN… Errors), no response is returned. The host (master) device will sense a timeout according to its timeout parameter. The host could then send a new command or retry sending the original command.

- Internals up to 1 second between characters are acceptable gaps. The MSOC will not timeout. Character send gaps in excess of 1 second will result in MSOC Modbus port timeouts.

MSOC Modbus network implementers will usually notice communication errors in the form of excessive communication retries, errors, or non-responses. Understanding communication timing is a subject rarely covered in protocol manuals, but an important topic in network implementation.

Network timing is predicated upon the following factors:

1. Host Latency (How long does it take a host device to generate a command, receive the response and interpret the data).
2. Intermediate Device Latency (If a Modem, data concentrator or other device is between the end device required for data retrieval, how long does it take for each device to receive the command and process it downline to the next device).
3. MSOC Device Latency (How long does it take for an MSOC to receive a command, and return a response to the network).
4. Baud Rate (How fast is each data bit propagated on the medium).
5. Protocol Efficiency [Network Bandwidth Utilization] (Does the protocol utilized allow for the issuance of another command before a response is received from an outstanding communication request).

The common question to a network system engineer is usually “How fast can I get my relay alarm data to appear on the screen?” An analysis of the amount of data requested and the above 5 areas is required.

Host latency varies widely by manufacturer or the PC or host computer. Software speed and port access varies widely. Most manufacturers of these hardware and software platforms have general benchmarks to supply to the users for processing time once the device acquires the data from the communication port.

Intermediate Device Latency also varies from the type of device used. Some modems have a device turnaround of 5 mS per transactions whereas; a radio modem may require hundreds of mS to obtain an open frequency from which to transmit.

This section shall illustrate and explain a simple network transaction based upon a simple point to point communication from a single MSOC to a host device as illustrated in Figure 4-27. This example shall exclude SCADA Master host latency.

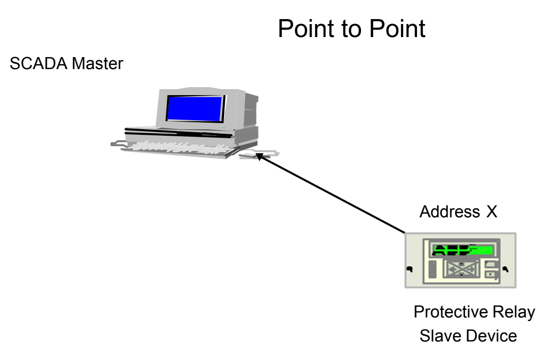


Figure 4-27. Example Communication Timing Topology

Modbus Baud Rate Analysis

If Section 4 Modbus ASCII Protocol is re-examined, the Modbus frame is illustrated in Figure 4-4. The frame is a standard 10 bit frame. One character (7 data bits) is transmitted as 10 bits per frame.

The rate of the data transfer is determined by the selected baud rate. The faster the baud rate, the faster the communication. The MSOC supports baud rates of 1200, 2400, 4800, 9600 and 19200. The effect of transfer time is shown in Table 4-45. Each bit has specific transfer time which correlates to a specific character transfer time.

Table 4-45. Character Transfer Time vs Baud Rate

Baud Rate	Transfer Time Per Bit	Transfer Time Per Character
1200	0.833 mS	8.333 mS
2400	0.4167 mS	4.167 mS
4800	0.2083 mS	2.083 mS
9600	0.1041 mS	1.041 mS
19200	0.0521 mS	0.521 mS

These are fixed times are standard for asynchronous bit stream transfers.

Each Modbus transfer varies in the amount of bytes transmitted and requested. Table 4-46 lists the amount of fixed data per some of the common Modbus commands. For example, each data transmission contains the following characters as per Figure 4-6:

- Colon (:)[3A Hex]
- Slave Address (Two Characters)
- Function (Two Characters)
- Error Check (Two Characters)
- Line Feed (One Character)
- Carriage Return (One Character)

Each base transmitted and received command has at least 9 characters for transmission. The transmission time, depending upon baud rate can range from 74.97 mS (at 1200 baud) to 4.689 mS (at 19200 baud). For example Figure 4-5 illustrates the Function 01 Read Coil Status format. Figure 4-6 illustrates the transaction request for four coils. Analysis of the data transmitted and received yields the following:

Transmission Request: Common characters 9 + 4 address characters + 4 data request characters
Total characters for transmission request = 17 characters

Returned Response: Common characters 9 + 2 data byte characters + 4 data returned characters
Total characters returned by MSOC = 15 characters

Depending upon the baud rate the total time for the communication characters to propagate along the network could range from:

Transmission Request: 17 characters at 141.61. mS (1200 Baud) to 8.857 mS (19200 Baud)

Returned Response: 15 characters at 124.95 mS (1200 Baud) to 7.815 (19200 Baud)

Total network transfer time via the physical medium can range from 265.56 mS at 1200 baud to 16.672 at 19,200 baud. Baud rate is a major influence at 1200 baud and a lesser influence at 19200 baud. It must be realized that this is only one of three components analyzed for a complete throughput analysis. In this case the Host time to generate the command.

MSOC Throughput Analysis

Communication implementation within a protective relay is a demanding task. In other devices, communications may take first priority. Within an ABB protective relay PROTECTION IS THE FIRST PRIORITY**.** Communication shall not compromise protection capabilities. Thus communication throughput may vary depending upon the demands of the protection. Table 4-46 illustrates the MSOC average benchmark times for recognition of a Modbus command at the physical port and the time it takes to generate a reply to the respective port. The times listed in the table are average times and do not include the calculated values generated in Section 4.

Table 4-46. MSOC Modbus Command Throughput (Average Time in mS)

			Reading from MSOC		Write to MSOC	
Modbus Command	**Register Start**	**Num. Refs.**	**Min (ms)**	**Max (ms)**	**Min (ms)**	**Max (ms)**
Read Logical Outs	00001	14	5.023	14.417		
Read Physical Outs	00129	4	1.497	10.688		
Read Physical Inputs	10129	2	1.381	13.726		
Load Metering	40513	27	21.270	30.184		
Configuration & Status	40129	22	18.848	26.324		
Event Records	41281	12	9.927	23.237		
Config Settings	60001	21	18.657	23.477	39.224	289.634
Primary Settings	60257	39	27.557	37.834	67.129	497.97
Master Trip Settings	61665	10	9.728	17.660	22.935	110.169
Test Setup:						
MSOC COMM Port Settings: 9600, E,7,2, through the FRONT RS-232 port						
MODLINK Setup: 500 ms Poll though COM1 on a 486DX100 Notebook Serial Port						
MSOC is "idle", No Current/Voltage applied.						
Write Min - Writing to update the Write Link side in ModLink						
Write Max - Time to Write 3 Sets of parameters to EEPROM and Return Response. Write Max times ARE						
proportional to the size of the block being written, the larger the block, the longer the write time.						

For the example, the MSOC generation time for the sample example can range from 1.497 mS to 10.688 mS.

Final Throughput Calculation and Analysis

A final calculation of our example throughput is warranted. For this example, the host update time shall now be assumed to be 250 mS. This 250 mS shall be an example estimation or time to generate a command, interpret the received command and update the screen. This is just for this example and varies according to:

- Speed of the host processor (hardware bus structure, # of processors, video card update, RAM memory, microprocessor speed…)
- Operating system selected (LINUX, UNIX, OS2, WIN NT, WIN 3.1, WIN 98, WIN 95)
- MMI Port Driver Efficiency (PRICOM, Power RICH, USDATA)

Two results will be calculated, operation at 1200 and 19200 baud. The example is described as per Figure 4-6 within this document. The formula used to produce the typical response is:

System Throughput = Host Processing Time + String Transmit Time + MSOC Processing Time + String Reception Time

At 1200 Baud: 527.248 mS = 250 mS + 141.61 mS + 10.688mS (using worst case)+ 124.95 mS

At 19200 Baud: 277.36 = 250 mS + 8.857 mS + 10.688 (using worst case) mS + 7.815 mS

Figures 4-28 and 4-29 illustrate the individual contributions from each of the components as a percentage of total transaction time.

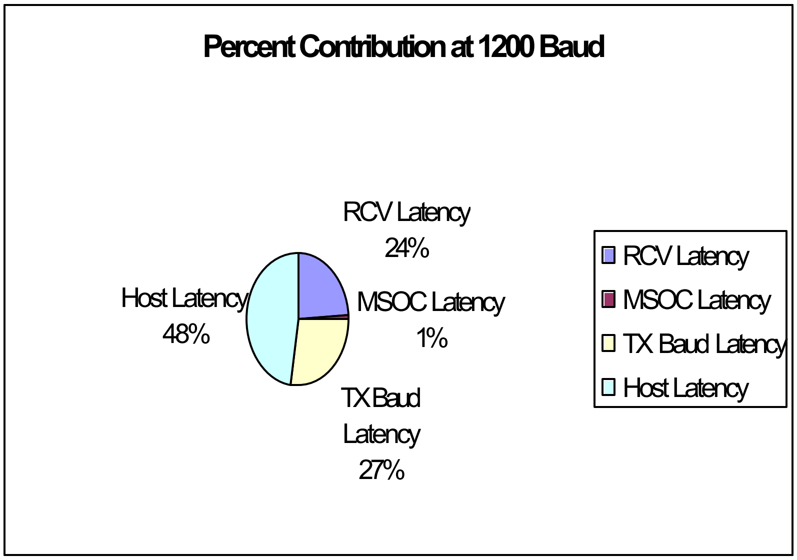


Figure 4-27. Network Throughput Analysis at 1200 Baud

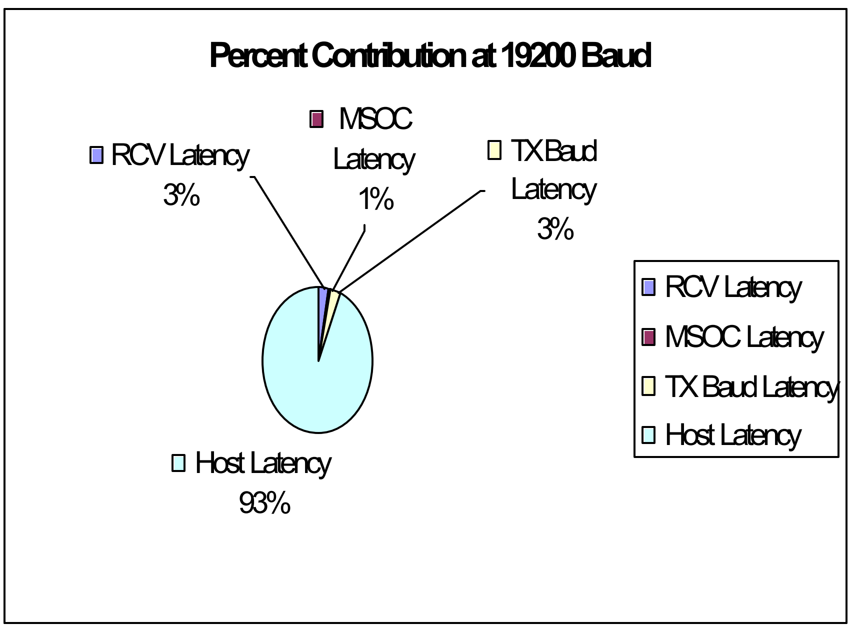


Figure 4-28. Network Throughput Analysis at 19200 Baud

Analysis of the simple example yields a few points to be considered when analyzing system throughput. Each element involved in communication timing contributes significantly to overall throughput. If the host executed and updated faster, overall throughput could be improved. If intermediate devices were inserted within the network, transaction time would increase proportionately. Baud rate is only one of many contributing factors in calculating system throughput. If one network access was required for retrieval of system data, overall network efficiency would be improved. If in a networked system, the protocol utilized would allow for additional data request commands while the slave device is processing a response, Network throughput time would be improved.

Virtual treatises have been written on improving system throughput and data updates. This simple example illustrates and allows the user to calculate system throughput times. This is especially critical so that the system user will not be surprised with overall system response.

ABB has implemented features within the protective relay to maintain system data integrity. Latched bit status, momentary change detect are a few features implemented within the various implementations of the Modbus protocol.

MSOC ASCII Protocol

Prior to the establishment of industry standard protocols, each vendor of utility equipment designed and included as part of the product a vendor specific protocol. These protocols were primarily based upon ASCII protocol. The MSOC has, in addition to the Modbus ASCII protocol, a Native ASCII protocol included on the standard product.

There are two different versions of Native the ASCII Communications for the MicroShield. The first is called “Fast ASCII”. This provides a fixed command set that is defined to allow the user to develop a simple program to communicate with the MicroShield. These commands are documented in this specification. This allows a host to communicate to the MSOC in a semi-automated method using proprietary device specific protocols.

The other ASCII Mode, “Menu ASCII”, is a user interactive based mode of communications. The User will be presented with screens of data and be required to respond in order to change parameters in the MicroShield. Usually implementers will attach a dumb terminal emulator to the unit allowing the modification and access of MSOC via a network viewing the specific parameters via a computer screen.

“Menu ASCII” was developed for users to configure the MSOC via a computer terminal instead of by the LCD MMI screen incorporated in the MSOC unit. As with the Modbus Protocol, Native MSOC ASCII is a Master-Slave Protocol. MSOC Native ASCII protocol has two sets of emulation “Menu Mode” and “Fast ASCII Mode”. Each of these sets has two subsets.

Subset 1 is via the RS 232 front port. No address is sent with the command. Communications through this port (as dictated by the protocol and the physical interface) is point to point only. This makes it useful for use with a dumb terminal emulator.

Subset 2 is via the RS 485 rear port. An address command is sent. Each device determines that it is an address command (since the command is in capital letter format only). Once the device establishes that its address command has been sent, it will listen and interpret all commands sent along the network. The device will stop interpreting commands when another address command is sent with differing parameters.

MSOC Native ASCII “Menu Mode”

Menu mode is the state in which the MSOC may communicate via a “dumb terminal emulator”. The MSOC present a menu to the operator via the terminal to which it is connected. The default state for the MSOC is MSOC Native ASCII “Fast Mode” described in Section 4. As explained in Section 3, the local echo setup parameter should be set to “YES” when configuring the port. This will set the MSOC to echo whatever is typed by the operator, thus allowing visualization on the terminal screen. Most terminals require duplex operation as such.

The MSOC Native ASCII “Menu Mode” operates via the front RS 232 port or the rear RS 485 port. The front port requires no address be sent for communication initiation strings, whereas communication through the rear port requires the string ADDR <xxx> <CR> <LF> (NOTE: The “ADDR” **MUST BE** upper case for proper reception) be sent to enable the unit to listen to communication ASCII strings sent along the communication network. The front port requires only a line feed carriage return <LF> <CR> to be sent to the unit. The front port is described as a point to point connection port (not suitable for multiple device connection). The rear port is described as a network port.

One the MSOC receives the initialization string; the following banner will be issued to the unit:

MicroShield O/C,0010,474M04116000, 974677,1.00,1.00
Type ‘menu’ to enter Menu Mode

Where:
MicroShield O/C is the Product’s Name

0010	is the Unit's Address, for verification of the connection
474M. . .	is the Unit's Catalog Number
9746. . .	is the Unit's Serial Number
1.00	is the Software Version for the Main CPU
1.00	is the Software Version for the Comm CPU

All commands in MSOC Native ASCII Menu Mode are in lower case. To enter "menu mode" the user must type "menu" (omit quotation marks).

MSOC Native ASCII "Menu Mode" Terminal Configuration

ASCII communication through a "dumb terminal" requires configuration of the terminal. Popular terminal emulators are listed below according to the operating system:

- DOS – Procomm
- Windows 3.1 – Terminal (Standard Accessory with Windows 3.1)
- Windows 95, NT or 98 – Hyperterminal

Application Example Configuration: Example Hyperterminal settings for Windows 95, NT or 98 are listed below for a personal computer connected to an MSOC on Com port 1 through the RS 232 port on the front of the MSOC:

Connect Using PC port Com 1
Com Port Settings:
    9600 Baud
    8 data Bits
    no Parity
    1 Stop Bit
    no Flow Control
Settings:
    Function Arrow and Control keys as Terminal Keys
    VT 100 Terminal Emulation
    Backscroll buffer lines 100
    Beep three times when connecting or disconnecting
Terminal Modes:
    Cursor Block Blink
    Keypad Application Mode
    Cursor Keypad Mode
    Character Set ASCII
ASCII Send Setup:
    Send line ends with line feed
    Echo Typed Characters Locally (Set Local Echo Off in the MSOC)
    Line delay: 5 milliseconds
    Character delay 5 milliseconds
ASCII Receiving
    Enable line wrap function

NOTE: The MSOC should be set to the same Com Port Settings as listed above. Consult Section 2 and 3 for MSOC cabling and port configuration procedures. Consult the Microsoft operating system documentation for Hyperterminal configuration instructions.

It should be noted that other settings may be used to accomplish the same connectivity. The above settings are intended to serve as an example only.

MSOC Native ASCII "Menu Mode" Command Format

The menu mode requires the user to type a single command letter to the MSOC. The letter may be in upper or lower case. Since the terminal setup (Section 4) appends a line feed (<lf> HEX code OD) to the enter key (carriage return <cr>) HEX code OA), the command is interpreted by the MSOC device. Figure 4-30 lists the menu structure viewed by the operator on the terminal screen when unit communications are established.

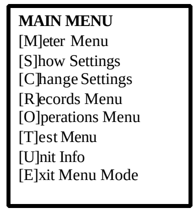


Figure 4-30. Main Menu Selection Screen

The operator must select one of the letters enclosed within the brackets. In the Menu Mode UPPERCASE or LOWERCASE letters may be used. If [M]eter Menu is selected (M or m) the screen depicted in Figure 4-31 shall be displayed to the operator.

Figure 4-31 also includes submenus and example displays for each selection. Additionally, the following keystrokes are valid:

- Return (configured to give a <cr> <lf>) shall advance to the next submenu selection.
- Any menu selection as depicted by the arrows.

If the MSOC has no voltage inputs, then voltage and power values are not available from the relay as depicted in Figure 4-31. If voltage is included within the MSOC, Volt per phase values (as indicated on the front panel) will be displayed. Power values will also be calculated and reported to the operator.

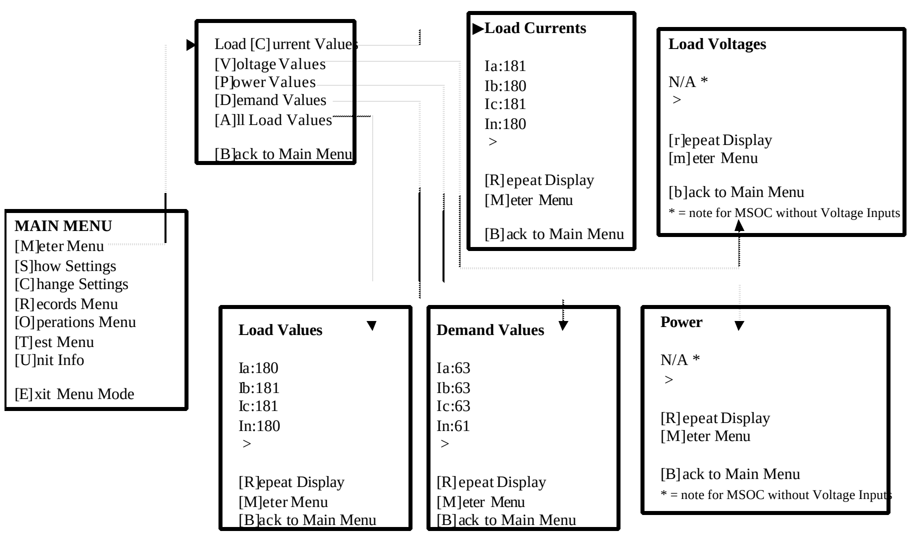


Figure 4-31. Meter Menu Submenu Selections

Meter display values are not updated continuously on the screen. The values displayed are those provided by the unit as a "snapshot" when the values are requested by the host.

If the operator selects the [S]how Settings screen, from the Main Menu, various selections shall be displayed upon the operator terminal. The screens displayed are illustrated in Figures 4-31 and 4-32. The settings can only be viewed by the operator in this screen. Figures 4-30 and 4-33 also display example values visible when the submenus are selected. These menus are analogous to the “Fast ASCII” command types found in Table 4-47 of this guide.

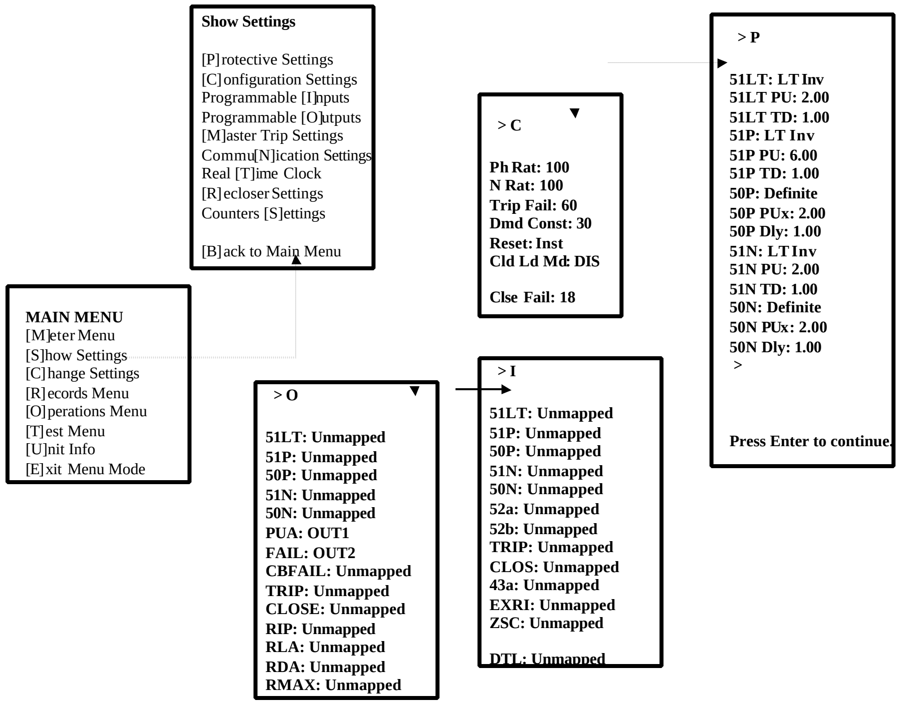


Figure 4-32. Show Settings Menu Selections

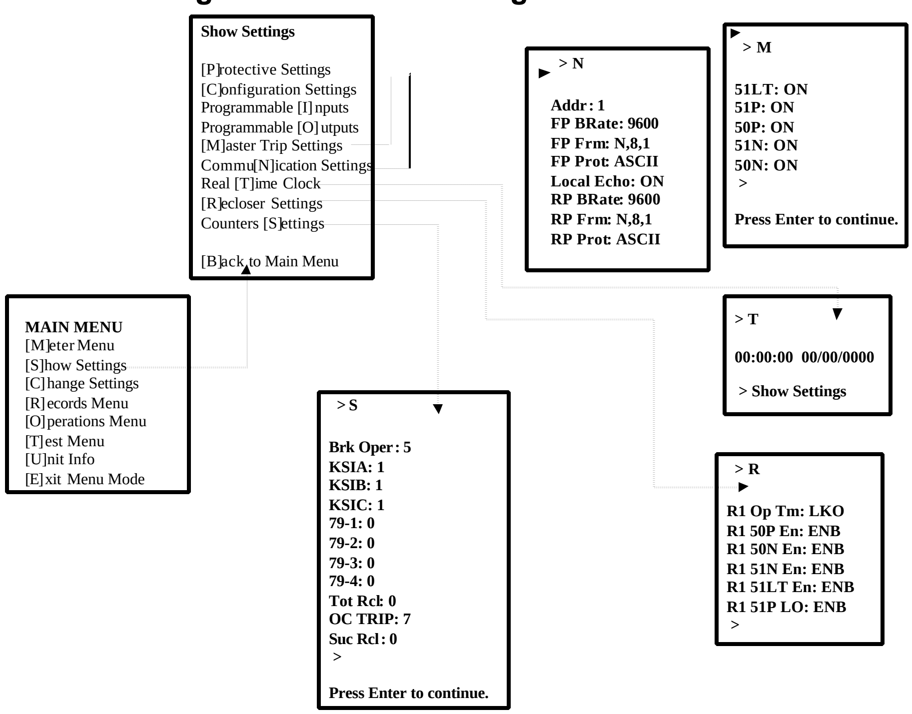


Figure 4-33. Show Settings Menu Selections (continued)

If the show settings command are selected, the command “Press Enter to Continue” shall be displayed. If enter is depressed on the keyboard, the Show Settings screen shall be displayed. The Show Settings display allows operator visualization of the parameters programmed in the MSOC.

Selection of [C]hange Settings shall display options for changing the relay settings through the communication interface. The same selections may be changed via the MSOC front panel interface also. The following tips should be remembered when operating in the [C]hange Settings menu:

- Password is required (the default is four spaces). Three tries are allowed. If the password is rejected, control is transferred to the previous level of menu selection.
- A <cr> <lf> (configured as depressing the enter key on the terminal keyboard), shall advance the display to the next selection without changing the value in the presented and presently displayed item.
- If the user types an “=” sign, the menu shall advance to the “**SAVE SETTINGS**” prompt and wait for the user to answer the query with a “Y” or “N” response.
- If the user types a value outside of the valid range of responses as indicated in the menu selection, a response of “**INVALID SELECTION**” will be displayed to the operator.

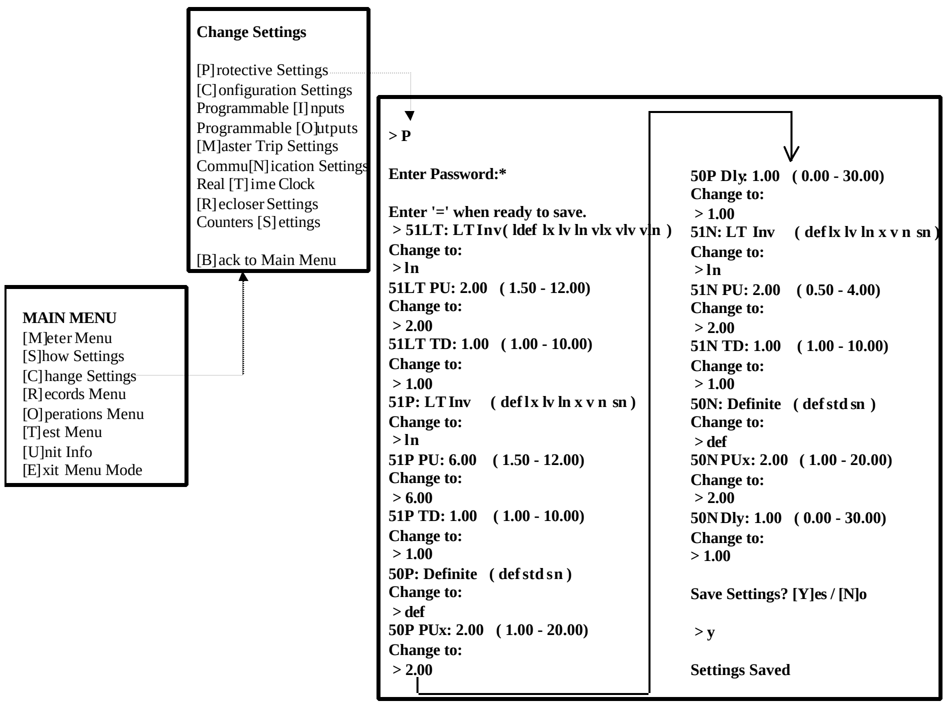


Figure 4-34. Example Change Setting Responses for Submenu [P]rotective Settings

The example shown in Figure 4-34 is an example of the operator selecting each response within the range presented within the menu item selections. The MSOC indicates that the settings have been selected and saved.

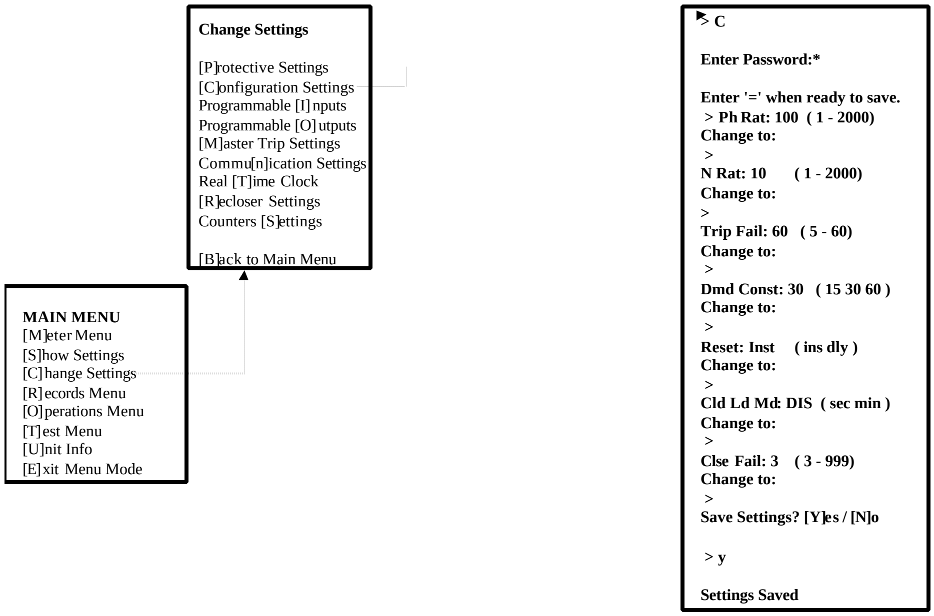


Figure 4-35. Example Change Setting Responses for Submenu [C]onfiguration Settings

Figure 4-35 illustrates an example menu screen for default settings within the MSOC. The operator in this case has entered a <cr> <lf> (in this case hyperterminal is configured to give this string when the keyboard's enter key is depressed).

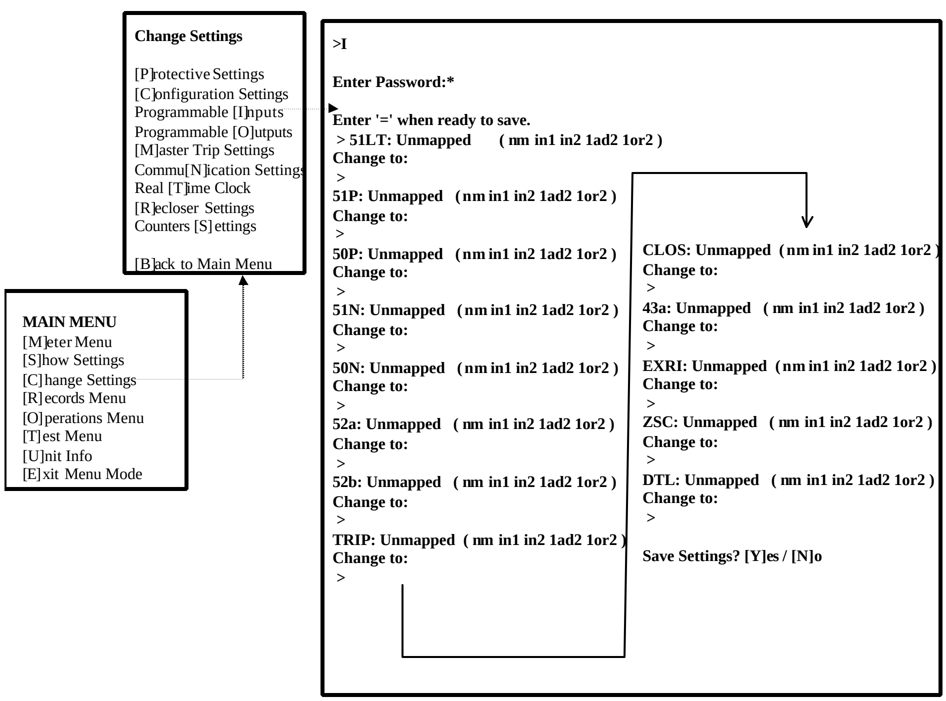


Figure 4-36. Example Change Setting Responses for Submenu Programmable [I]nputs

Figure 4-36 illustrates the sample queries viewed over a terminal emulator when the submenu Programmable [I]nputs is entered. The operator has in this case depressed the enter key on the terminal keyboard. The relay settings for each parameter remains the same.

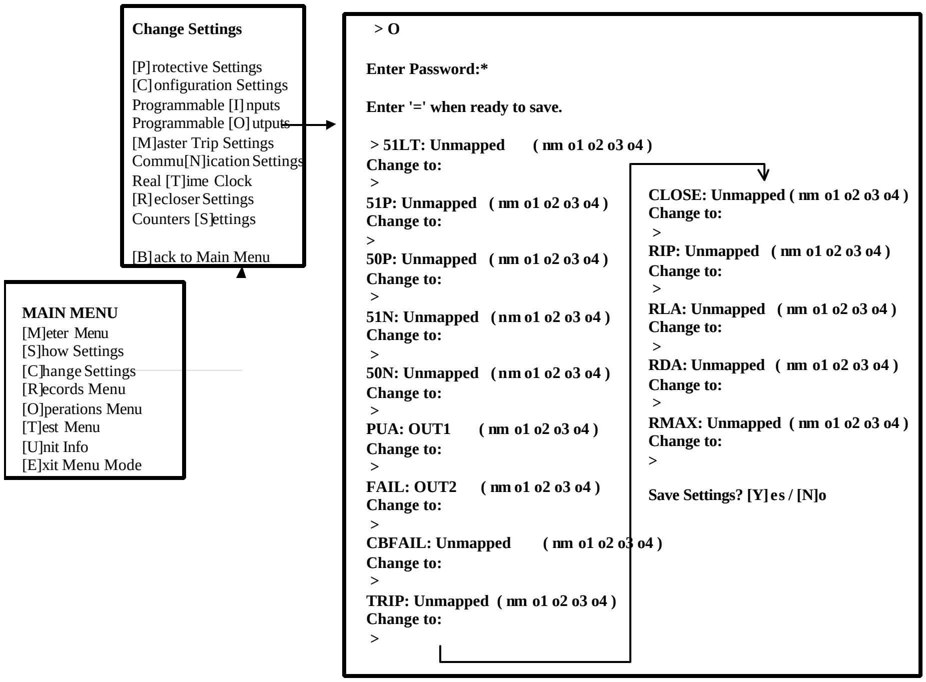


Figure 4-37. Example Change Setting Responses for Submenu Programmable [O]utputs

Programmable outputs may also be mapped via the "Menu ASCII" communication connection. Figure 4-37 illustrates a sample screen and the operator options for each item.

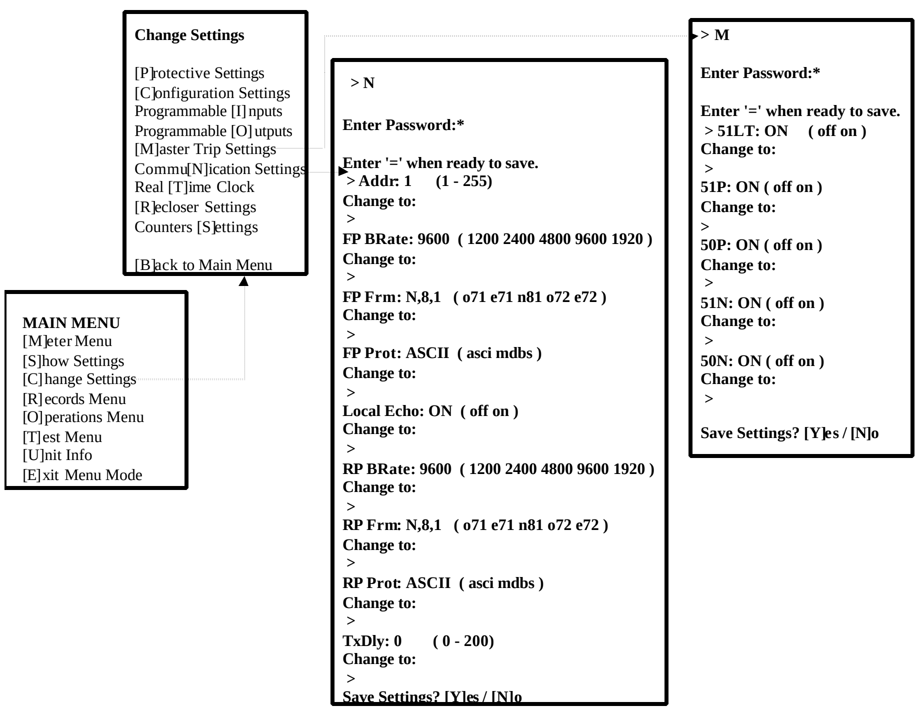


Figure 4-38. Change Setting Screen Example Illustrating the [M]aster Trip and Commu[N]ication Settings Submenus

The [M]aster Trip and Commu[N]ication Settings submenus are illustrated in Figure 4-38. As in all the other menu's the password must be supplied by the operator and the "Save Settings" query must be answered.

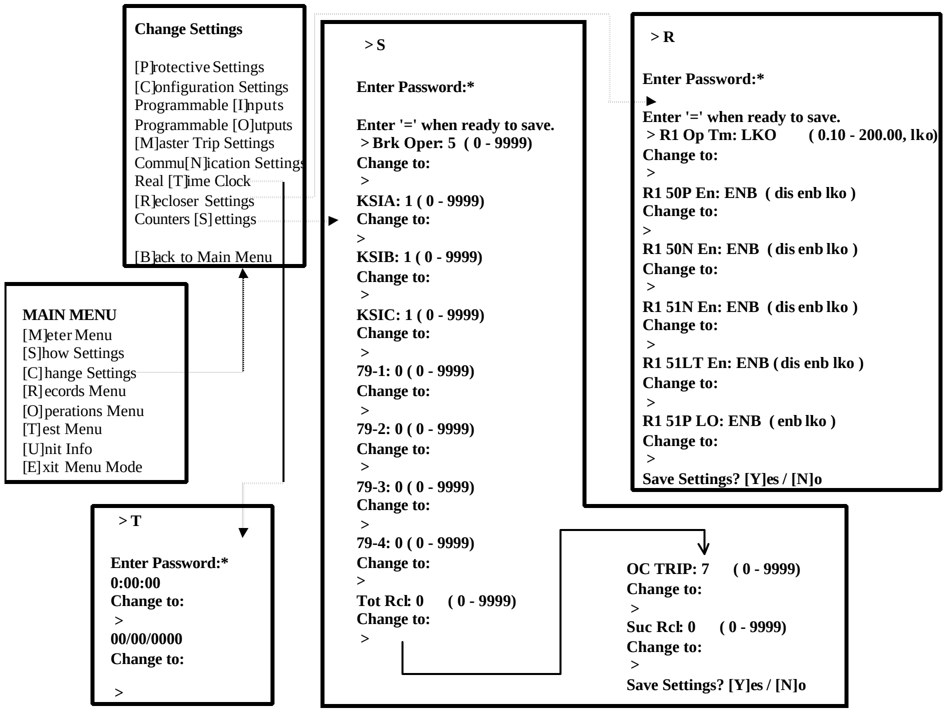


Figure 4-39. Change Setting Screen for Real [T]ime Clock, [R]ecloser Settings and Counter [S]ettings Submenus

The real time clock, recloser and counter settings may also be changed via "Menu ASCII" mode within the MSOC.

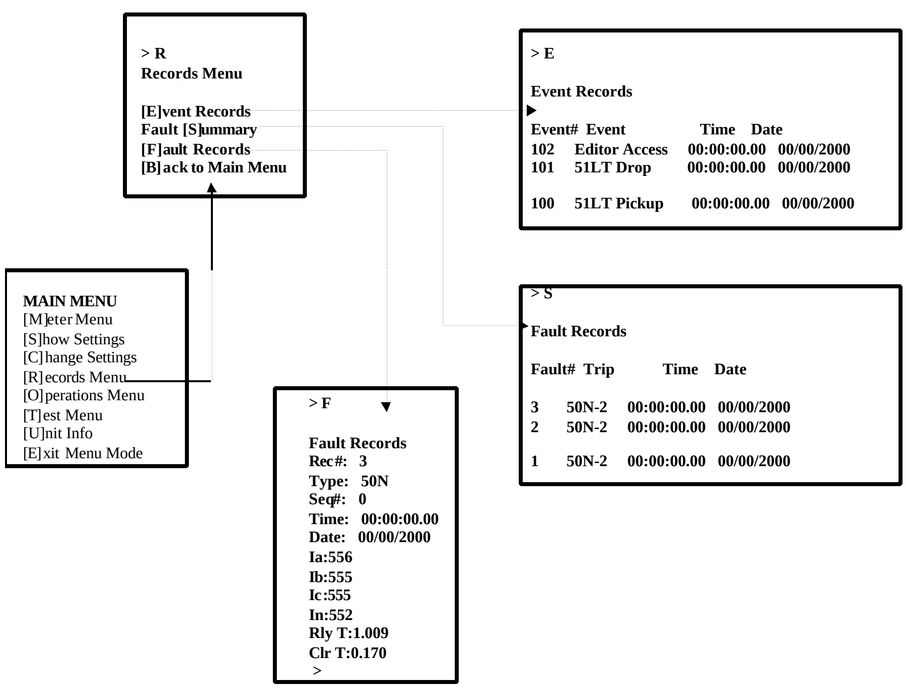


Figure 4-40. Record Menu Screen and Submenu Illustration

Fault records, event records and fault record summaries may be viewed from a terminal operating from "Menu ASCII" mode. Figure 4-40 illustrates an example record access. The example illustrates the MSOC from which the records were obtained has the clock disabled (set to 00:00:00 for the time and the date disabled as 00/00). The year defaults to 2000. The year is Y2K compliant for Version 1.60 and greater firmware revision relays.

Y2K compliant units determine the year as such. If the year is less than 80 then it is assumed 2000 is the decade. If the year is 80 or greater, the decade is assumed to be 1900.

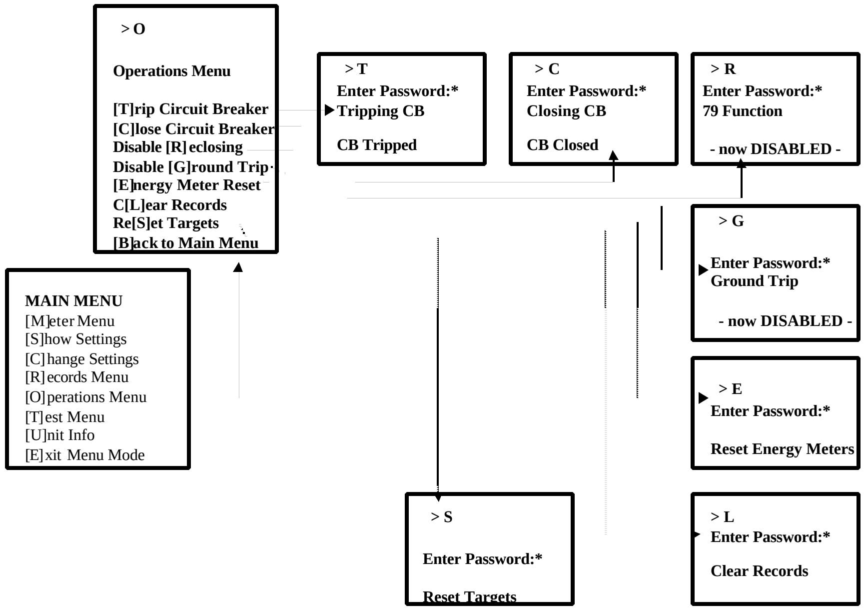


Figure 4-31. Operations Menu Selection with Submenu Illustrations

The operation menu performs test functions within the relay. Certain functions may be enabled or disabled. A password is required for enabling these test control functions as illustrated in Figure 4-41.

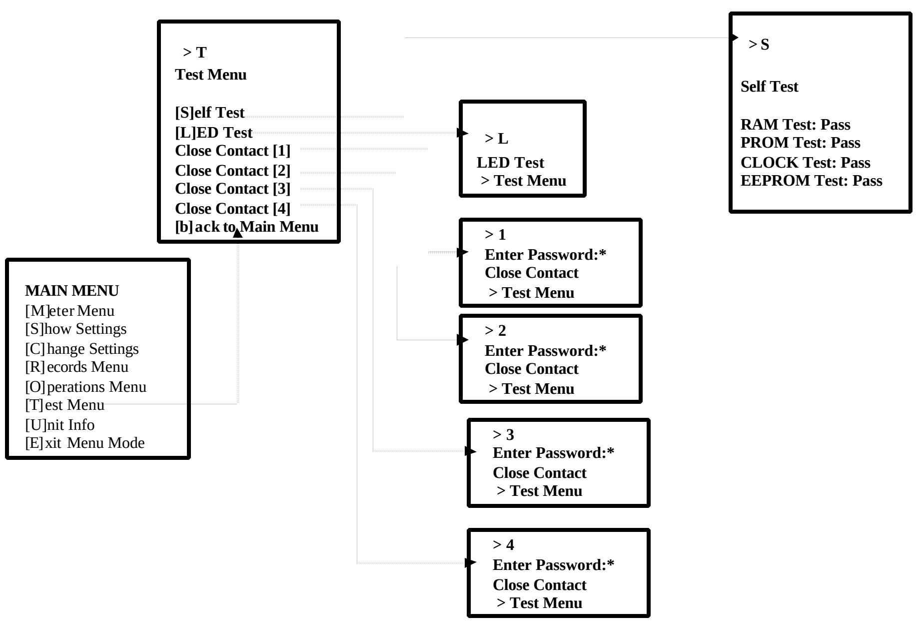


Figure 4-42. Test Menu Selection Example with Submenu Illustrations

The test menu allows for an operator to pulse the physical outputs on the relay. If the physical outputs are not available on the MSOC, an indication is given to the operator that the function cannot be performed. If a unit self-test is performed, the response from the relay to the terminal screen will not appear until the diagnostic tests are complete. A response will be received from the MSOC before the LED test is completed.

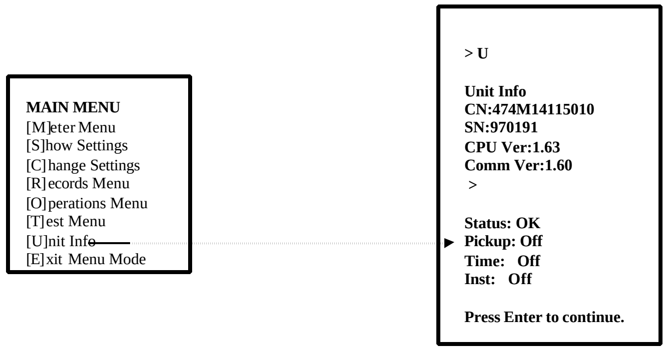


Figure 4-43. Unit Information Example Screen

The unit information screen displays information in excess of that available from the MSOC MMI screen. MSOC Native ASCII " Menu Mode" is an easy and intuitive method of accessing and changing MSOC parameters. Its most convenient feature is each command is listed on the screen in a prompted format. Another convenient feature is that there is no special software configuration package to purchase or learn.

MSOC Native ASCII "Fast ASCII Mode"

Fast ASCII mode allows for computers, PLC's and other devices to send commands to the MSOC and receives responses. Fast ASCII does not include terminal drivers (which is more efficient since less characters are be sent across the network).

Fast ASCII has the following constraints:

- The commands from the host to the MSOC are limited to 8 characters each (excluding the <CR><LF>) and **MUST BE** lowercase! Each response line is limited to an 80 character text string.
- Each command must be terminated with a <CR><LF> combination. A plain <CR> will NOT work. See your terminal emulator documentation on enabling CR-LF Outbound conversions.

Upon first connecting to the Unit, through either port, the following banner will be issued to the User:

The MSOC Native ASCII "Fast ASCII Mode" operates via the front RS 232 port or the rear RS 485 port. The front port requires no address to be sent for communication initiation strings, whereas communication through the rear port requires the string ADDR <xxx> <CR> <LF> (**NOTE**: The "ADDR" **MUST BE** upper case for proper reception) be sent to enable the unit to listen to communication ASCII strings sent along the communication network. The front port requires only a line feed carriage return <LF> <CR> to be sent to the unit. The front port is described as a point to point connection port (not suitable for multiple device connection). The rear port is described as a network port.

One the MSOC receives the initialization string; the following banner will be issued to the unit:

MicroShield O/C,0010,474M04116000,974677,1.00,1.00
Type 'menu' to enter Menu Mode

Where:
MicroShield O/C Is the Product's Name
0010 is the Unit's Address, for verification of the connection
474M. . . is the Unit's Catalog Number
9746. . . is the Unit's Serial Number
1.00 is the Software Version for the Main CPU

1.00 is the Software Version for the Comm CPU, if Present, else blank

The second line is information on how the user can enter the ASCII Menu Mode of operation. Since the mode of operation is Fast ASCII, the string ‘menu’ is not to be sent to the unit, as it was to initiate “Menu ASCII Mode”.

Fast ASCII Command Format

If the following tables are reviewed, a commonality will be readily evident. Comparison of the “MSOC Native Menu ASCII” choices and the “MSOC Native Fast ASCII” commands illustrate the commonality between the protocol derivatives. Although MSOC Native Fast ASCII is intended to be used with a PC, it may be driven by a “dumb terminal emulator” as is done with “MSOC Native Menu ASCII” mode. The commands may be typed as per the following tables. There is no timeout for the port.

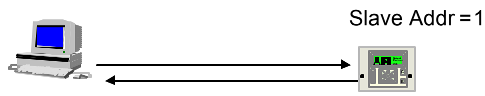
Fast ASCII commands cannot be grouped together, only 1 command per line will be recognized. To simplify the decoding of received commands; each command type will begin with a unique character heading. These command types are organized logically so that commands can be remembered. For example, **m** is for all Load Meter Values; d is for all Demand Meter Values, and so on. In some cases, the first two characters are significant, like to for Physical Outputs (or Terminal Outputs).

Table 4-47 lists the currently defined command types for the MicroShield.

Table 4-47. Read and Read/Write Native MSOC ASCII Command Types

	Command Type	1st Characters
	Read Commands	
	Load Values	m
	Demand Values	d
	Event Records	e
	Fault Records	f
	Logical Input States	li
	Logical Output States	lo
**	Seal-In States	si
	Physical Inputs	ti
	Physical Outputs	to
	Misc. Information and Commands	x
	Read/Write Commands	
	Protective Settings	p
$$	Recloser Settings	r
	Configuration Settings	c
	Prog. Input Settings	i
	Prog. Output Settings	o
**	Seal-In Output Settings	so
	Comm. Settings	n
	Clock Settings	k
	Breaker Counters	b
	Master Trip Output Settings	q
	$$ = Added in Version 1.6 Software ** = Added in Version 1.7 Software	

A command that does not begin with any of the prefixes above will return a **02: Invld Cmnd Recvd** error message. If a space (HEX code 20) is seen as the first character in the message, a 02**: Invld Cmnd Recvd message** will be generated. All error messages are listed at the end of this document in Section 4.



Read Command Structure [(SENT FROM HOST) for Read and Read/Write Commands

mia <CR><LF>
(access Phase A Load Current)

MSOC Fast ASCII Representation

6D 69 61 0D 0A

Write Command Structure [(SENT FROM HOST) for Read/Write Commands]

sirda <CR> <LF>
(access Recloser Disable Alarm Status)

MSOC Fast ASCII Representation

73 69 72 64 61 0D 0A

Figure 4-44. Fast ASCII Command in Hex Format

As illustrated in Figure 4-44 the Fast ASCII format is of the following form for a read command:

{Command}<CR><LF>

This command has not been preceded by the ADDR command. Thus, the access is for front port only. The read commands are listed in Tables 4-48 through 4-80. If the User desires to write a setting to the Unit, the following format is required:

{Command}, value, password<CR><LF>

Where: The value parameter entered must have the same format as described in the Response column of the Tables contained in Section 4.

If the received string is not understood, a "**03: Invld Cmnd Syntx"** response will be returned.
When a setting is changed successfully, a "**01: OK"** response will be returned.

Fast ASCII Argument Definition

When the Unit receives a valid command, it is processed and an ASCII Text String is returned to the User. When numeric values are to be returned, they may or may not be formatted with a decimal point. The following notation is used in the tables defined below:

DP: Decimal Places, e.g. 2DP is 2 Decimal Places
LDP: Decimal Point all the way to the Left
RDP: Decimal Point all the way to the Right

For Example:

min	12345	5 digits, RDP	Load IN

defines the Load Meter Neutral Current, IN to be a 5 digit number, with the decimal point to the right. In another example:

mvan	657.80	5 digits, 2DP	VA Magnitude

defines the Load Meter Voltage of VA to N Magnitude to be a 5 digit number with 2 decimal places. These decimal points are returned in the response. The number of digits shown in each response will be the number of digits returned (In text form). If a field underflows, there will be preceding zero's to make the field size fixed.

When writing to the Unit, as noted above, the following rules apply:

- If a decimal point is required, and none is provided, then the decimal point is assumed to be at the end and the number ends with ".00".
- If too many decimal places are provided, the number is truncated.
- Numbers are rounded down to their next lowest step value. For example, if the step size on a parameter is .25 and the User enters a number .45, then the number stored will be .25.
- On reads, if the number only has 1 decimal place of significance, it is returned as 2 decimal places, with the last place being 0.
- If the accessed value is a signed value, and is reported as a negative value, the negative sign is returned (HEX 2D) to indicate sign.

The following are tables, reflect the currently defined Fast mode commands, and their responses.

Load Values (Read Only Commands)

Table 4-42 lists Fast ASCII Load Value metering commands present within the MSOC. Each value has certain restrictions as explained in the notes section following Table 4-48. If the MSOC does not have Voltage input capabilities, refer to the notes for sample "exception responses" indicating the unit's inability to interpret or respond to the command.

Application Example: Figure 4-42 illustrates the base MSOC Native Fast ASCII commands sent to the MSOC to access Ia phase current.

Example : Read Load Current Phase a:

TEXT FORMAT mia <CR><LF>
(access Phase A Load Current)

ASCII FORMAT 6D 69 61 0D 0A

Host to MSOC Transmission

MSOC to Host Response

TEXT FORMAT 00070<CR><LF>

ASCII FORMAT 30 30 30 37 30 0D 0A
(Phase A Load Current is 70 A)

Figure 4-45. Read Load Current Phase A Example in HEX Format

Table 4-48. Load Value Command Listing

Command	Typical Response	Format	Description
Mia	12345	5 digits, RDP	Load IA – Unsigned – 00000<=Range<=99999
Mib	12345	5 digits, RDP	Load IB – Unsigned – 00000<=Range<=99999
Mic	12345	5 digits, RDP	Load IC – Unsigned – 00000<=Range<=99999
Min	12345	5 digits, RDP	Load IN – Unsigned – 00000<=Range<=99999
Milst	{Note 1}		All Current Load Meter Values
Mvan	657.80 {Note 3}	5 digits, 2DP	VA Magnitude (line-neutral) 000.00<=Range<=999.99
Mvbn	658.70 {Note 3}	5 digits, 2DP	VB Magnitude (line-neutral) 000.00<=Range<=999.99
Mvcn	659.80 {Note 3}	5 digits, 2DP	VC Magnitude (line-neutral) 000.00<=Range<=999.99
Mvab	657.80 {Note 3}	5 digits, 2DP	VAB Magnitude (line-line) 000.00<=Range<=999.99
Mvbc	658.70 {Note 3}	5 digits, 2DP	VBC Magnitude (line-line) 000.00<=Range<=999.99
Mvca	659.80 {Note 3}	5 digits, 2DP	VCA Magnitude (line-line) 000.00<=Range<=999.99
Mkw	4332180 {Note 3}	7 digits, RDP	3 Phase Kwatts – Signed – -999.99<=Range<=999.99
Mkvar	3432190 {Note 3}	7 digits, RDP	3 Phase KVAR – Signed – -999.99<=Range<=999.99
Mkwh	6321000 {Note 3}	7 digits, RDP	3 Phase KWatt Hours – Unsigned 00000<=Range<=99999
Mkvarh	5468910 {Note 3}	7 digits, RDP	3 Phase KVAR Hours – Unsigned 00000<=Range<=99999
Mfq	60.00 {Note 3}	4 digits, 2DP	Line Frequency 00.00<=Range<=99.99
Mpfv	-0.54 {Note 3}	4 digits, 2DP	Power Factor Value 01.00<=Range<=0.00
Mpfd	1,0 {Note 3}	1 digit, RDP	Power Factor Direction 1 = Lead, 0 = Lag
Mvlst	{Notes 2,3}		All Voltage Load Meter Values

NOTES:

1. Returns a comma separated list of each of the Load Current Values in the following order:
   mia, mib, mic, min
   If no VT's are present, any command below **milst** in the table will return a "**03: Invld Cmnd Syntx"** response.

2. Returns a comma separated list of each of the Load Voltage Values, if the VT's are present, in the following order:
   mva, mvb, mvc, mkw, mkvar, mkwh, mkvarh, mfq, mpf
   The voltages will be displayed once either line-neutral (i.e., mvan) or line-line (i.e., mvab) depending on the Voltage Display Mode setting, cvdmd, in the Configuration Settings menu.

3. Power Factor is now an UNSIGNED number between 0 and 1. The Leading and Lagging indication determines the appropriate quadrant.

Demand Values (Read Only Commands)

Table 4-49 lists MSOC Native ASCII "Fast ASCII" Demand Value Access commands. The commands access the real time status of each defined element. At the end of the table, a note section is listed for the assigned elements. An example hexadecimal command string follows and is illustrated in Figure 4-46.

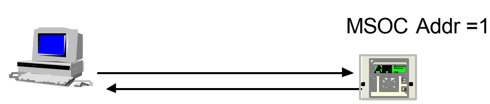
Table 4-49. Demand Value Access Command Codes

Command	Response	Format	Description
Dia	12345	5 digits, RDP	Demand IA IB – Unsigned – 00000<=Range<=99999
Dib	12345	5 digits, RDP	Demand IB IB – Unsigned – 00000<=Range<=99999
Dic	12345	5 digits, RDP	Demand IC IB – Unsigned – 00000<=Range<=99999
Din	12345	5 digits, RDP	Demand IN IB – Unsigned – 00000<=Range<=99999
Dilst	{Note 1}		All Current Demand Status
Dkw	4332180 {Note 3}	7 digits, RDP	3 Phase Kwatts – Unsigned 00000<=Range<=99999
Dkvar	3432190 {Note 3}	7 digits, RDP	3 Phase KVAR –-Unsigned 00000<=Range<=99999
Dplst	{Notes 2,3}		All Power Demand Status

NOTES:

1. Returns a comma separated list of each of the Current Demand Values in the following order:
   dia, dib, dic, din
   If no VT's are present, any command below **dilst** in the table will return a "**03: Invld Cmnd Syntx"** response.
2. Returns a comma separated list of each of the Power Demand Values in the following order: dkw, dkvar
3. If no VT's are present, the MSOC will return a "**03: Invld Cmnd Syntx"** response.

Application Example: Figure 4-46 illustrates the process of reading 3 Phase KVars from the MSOC.



Example : Read 3 phase KiloVAR Demand Value:

TEXT FORMAT dkw <CR><LF>
(3 Phase KVAR Demand Value)

HEX ASCII FORMAT 64 6B 77 0D 0A

Host to MSOC Transmission

MSOC to Host Response

TEXT FORMAT 03: Invld Cmnd Syntx<CR><LF>

ASCII FORMAT 30 33 3A 20 49 6E 76 6C 64 20 43 6D 6E 64 20 53 79 6E 74 78 0D 0A
(The unit does not have voltage inputs)

Figure 4-46. Read 3 Phase K Vars Example

Event and Fault Records (Read Only)

This section details the Reading of the Event Records and Fault Records. In this mode, the MSOC will output the entire record requested, as stored in the Unit. No Summary information (all records) is provided. The Y2K Compliant versions of these commands are available as of V1.50. All previous versions will return a **03: Invalid Cmnd Syntax** response.

Table 4-50. Event and Fault Record Access Codes

	Command	Response	Format	Description
	ecurevt	{Note 1}		Current Event (newest)
	enxtevt	{Note 1}		Next Event
	fcurflt	{Note 2}		Current Fault (newest)
	fnxtflt	{Note 2}		Next Fault
	fclr	{Note 3}	{Note 3}	Clear all Records
$$	ecurev2k	{Note 4}		Current Event (newest) Year 2000 Compliant
$$	enxtev2k	{Note 4}		Next event Year 2000 Compliant
$$	fnxtft2k	{Note 5}		Next fault Year 2000 Compliant
$$	fcurft2k	{Note 5}		Current Fault (newest) Year 2000 Compliant
	fdetail	{Note 6}		Read Fault Record Details
$$ = Added in Version 1.6 Software				

NOTES:

1. Returns a comma separated list of the Event Record values in the following order:
   event number, event code, time, date.

Parameter	Format
Event Number	3 digits, RDP
Event Code	3 digits, RDP
Time	01:23:45.64
Date	01/23/97

A list of Event Codes is provided at the end of this document.

The date will be formatted as 97/01/23 when unit is in IEC mode.

If there are no event records, a **"07:No Records Avail**" response is returned.

2. Returns a comma separated list of the Fault Record values in the following order:
   Fault number, fault code, time, date
   For V1.90, this command now returns Fault Summary information in the format shown below. The 2 new commands provide the details (below)

Parameter	Format
Fault Number	3 digits, RDP
Fault Code	3 digits, RDP {See Fault Code Table for Encoding of Recloser Seq.}
Time	01:23:45.64
Date	01/23/97

The date will be formatted as 97/01/23 when unit is in IEC mode.

If there are no fault records, a **"07:No Records Avail**" response is returned.

3. Fclr requires a password as is done when changing settings, e.g.:
   Fclr, password<CR><LF>
4. Returns a comma separated list of the Event Record values in the following order:
   event number, event code, time, date.

Parameter	Format
Event Number	3 digits, RDP
Event Code	3 digits, RDP
Time	01:23:45.64
Date	01/23/97

A list of Event Codes is provided at the end of this document.

The date will be formatted as 97/01/23 when unit is in IEC mode.

If there are no event records, a **"07:No Records Avail**" response is returned.

5. Returns a comma separated list of the Fault Record values in the following order:
   Fault number, fault code, time, date
   For V1.90, this command now returns Fault Summary information in the format shown below. The 2 new commands provide the details (below)

Parameter	Format
Fault Number	3 digits, RDP
Fault Code	3 digits, RDP {See Fault Code Table for Encoding of Recloser Seq.}
Time	01:23:45.64
Date	01/23/97

The date will be formatted as 97/01/23 when unit is in IEC mode.

6. In V1.90 it was discovered that the returned string for the Fault information was larger than allowed (exceeded 80 characters). Therefore, in order to read ALL fault information, one now needs to execute 2 commands, one to read the Summary information (the existing commands) and another to read the details of the fault record. This new command returns the following comma separated list of the Fault Record Details:
   Fault number, mia, mib, mic, min, mvan, mvbn, mvcn,
   Relay Time, Clearing Time (in milliseconds)
   The fdetail command relies on the other 4 (the existing 2 current and 2 next) to position the "pointer" of the record details that it is going to read. For example: fcurflt reads current fault record #4. The fdetail command will read the details of fault record #4. The fxntflt command moves to Record #3, and fdetail will read Record #3. And so on through the records stored in the MSOC.

Parameter	Format
Fault Number	3 digits, RDP {to synchronize with the Summary information}
Current Load Values	See table above for encoding
Voltage Load Values	See table above for encoding, if present (based on Unit Catalog Number)
Relay Time	5 digits, RDP
Clearing Time	5 digits, RDP

If there are no event records, a **"07:No Records Avail**" response is returned.

Table 4-51. Event Codes (E-prefix based commands)

	Circuit Breaker Events 000 - 009	
	000	CB Closed
	001	CB Open
	002	CB State Unknown
	003	CB Failed to Trip
	004	CB Failed to Close
	005	CB Pop Open
	006	CB Pop Closed
	007	Ext. Trip
	008	Ext. Close
	009	Event #10 (Spare)
	Diagnostic Events 010 - 039	
	010	RAM 1 Failure
	011	RAM 2 Failure
	012	RTC Failure
	013	EEP Primary Failure
	014	EEP Config Failure
	015	EEP Output Failure
	016	EEP Input Failure
	017	EEP Comm Failure
$$	018	EEP Recloser Failure
	019	EEP Master Trip Failure
	020	EEP PROM Checksum Failure
	021	Analog Failure
	022	Editor Access
	023	Battery Low
	024	Watchdog Reset
	025	Power On Reset
**	026	EEP Seal-In Failure
**	027	Self Diag # 18 (Spare)
**	028	Self Diag # 19 (Spare)
**	029	Self Diag # 20 (Spare)
**	030	Self Diag # 21 (Spare)
**	031	Self Diag # 22 (Spare)
**	032	Self Diag # 23 (Spare)
**	033	Self Diag # 24 (Spare)
**	034	Self Diag # 25 (Spare)
	035	Self Diag # 26 (Spare)
	036	Self Diag # 27 (Spare)
	037	Self Diag # 28 (Spare)
	038	Self Diag # 29 (Spare)
	039	Self Diag # 30 (Spare)
	Input Event Codes 040 - 069	
	040	51LT Disabled
	041	51LT Enabled
	042	51P Disabled
	043	51P Enabled
	044	50P Disabled
	045	50P Enabled

	047	51N Enabled
	048	50N Disabled
	049	50N Enabled
	050	52a Opened
	051	52a Closed
	052	52b Opened
	053	52b Closed
	054	Direct Trip
	055	Direct Close
$$	056	RCL 43a Enabled
$$	057	RCL 43a Disabled
$$	058	Ext RCL Init
$$	059	Manual Reset
	060	Input # 21 (Spare)
	061	Input # 22 (Spare)
	062	Input # 23 (Spare)
	063	Input # 24 (Spare)
	064	Input # 25 (Spare)
	065	Input # 26 (Spare)
	066	Input # 27 (Spare)
	067	Input # 28 (Spare)
	068	Input # 29 (Spare)
	069	Input # 30 (Spare)
	Output Event Codes 070 - 099	
	070	51LT Trip
	071	51LT Pickup
	072	51LT Drop
	073	51P Trip
	074	51P Pickup
	075	51P Drop
	076	50PT Trip
	077	50PT Pickup
	078	50PT Drop
	079	51N Trip
	080	51N Pickup
	081	51N Drop
	082	50N Trip
	083	50N Pickup
	084	50N Drop
	085	51LT Clear
	086	51P Clear
	087	50P Clear
	088	51N Clear
	089	50N Clear
	090	51LT Reset
	091	51P Reset
	092	50P Reset
	093	51N Reset
	094	50N Reset
&&	095	Recloser Enabled

	046	51N Disabled
	097	Output #28 (Spare)
	098	Output #29 (Spare)
	099	Output #30 (Spare)
	Communication Event Codes 100 - 109	
	100	QSI Init Fail
	101	Com Init Fail
	102	Param Update
	103	Int Tx Failure
	104	Wrong (Modbus) Block Rec'd
	105	NW Param Upd
	106	Int Rx Failure
	107	Spare Comm #8
	108	Spare Comm #9
	109	Spare Comm #10
	Operation Event Codes 110 - 119	
	110	Rst Enrgy Mtrs
	111	Records Cleared
**	112	Reset Seal-Ins
	113	Spare Ops #4
	114	Spare Ops #5
	115	Spare Ops #6
	116	Spare Ops #7
	117	Spare Ops #8
	118	Spare Ops #9
	119	Spare Ops #10

&&	096	Recloser Disabled
	Recloser Event Codes 120 - 139	
**	120	Recloser 43a Enabled
**	121	Recloser 43a Disabled
**	122	Recloser Enabled
**	123	Recloser Disabled
**	124	Recloser in Progress
**	125	External Recloser Initiated
**	126	Recloser Lockout
**	127	Recloser Reset
**	128	Recloser Out Of Sequence
**	129	Max Recover Exceeded
**	130	Recloser Recover
**	131	Max Recloses Exceeded
**	132	Max Recloser Resets
**	133	Zone Step
**	134	Recloser Error
**	135	Recloser Init
**	136	Spare Recloser #17
**	137	Spare Recloser #18
**	138	Spare Recloser #19
**	139	Spare Recloser #20
	Miscellaneous Codes 140 - 141	
	140	Illegal Event Code
	141	Event Code Error
$$ = Added in Version 1.6 Software ** = Added in Version 1.7 Software && = Added in Version 1.8 Software		

The date will be formatted as 97/01/23 when unit is in IEC mode (year/month/day).

If there are no event records, a "**07: No Records Avail"** response is returned.

2. Returns a comma separated list of the Fault Record values in the following order:

   fault number, fault code, recloser sequence number,
   time, date, mia, mib, mic, min, mvan, mvbn, mvcn

Table 4-52. Fault Record Parameters and Formats (F-based commands)

Parameter	Format
Fault Number	3 digits, RDP
Fault Code	Text String, 4 chrs.
Time	01:23:45.64
Date	01/23/97

The date will be formatted as 97/01/23 (year/month/day) when unit is in IEC mode. The load values will have the same format described previously.

Table 4-53. MSOC Fault Codes

	Code	Text
	000	51LT – LO
	001	51P – LO
	002	50P – LO
	003	51N – LO
	004	50N – LO
&&	005	External Trip
&&	006	Input Trip
&&	007	Fault Code Error
	100	51LT – Reclose Step 1
$$	101	51P – Reclose Step 1
$$	102	50P – Reclose Step 1
$$	103	51N – Reclose Step 1
$$	104	50N – Reclose Step 1
&&	105	External Trip Recloser Step 1
&&	106	Input Trip Recloser Step 1
$$	200	51LT – Reclose Step 2
$$	201	51P – Reclose Step 2
$$	202	50P – Reclose Step 2
$$	203	51N – Reclose Step 2
$$	204	50N – Reclose Step 2
&&	205	External Trip Recloser Step 2
&&	206	Input Trip Recloser Step 2
$$	300	51LT – Reclose Step 3
$$	301	51P – Reclose Step 3
$$	302	50P – Reclose Step 3
$$	303	51N – Reclose Step 3
$$	304	50N – Reclose Step 3
&&	305	External Trip Recloser Step 3
&&	306	Input Trip Recloser Step 3
$$	400	51LT – Reclose Step 4
$$	401	51P – Reclose Step 4
$$	402	50P – Reclose Step 4
$$	403	51N – Reclose Step 4
$$	404	50N – Reclose Step 4
&&	405	External Trip Recloser Step 4
&&	406	Input Trip Recloser Step 4
$$ = Added in Version 1.6 Software		
&& = Added in Version 1.90 Software		

For Fast ASCII commands, the Recloser Sequence is embedded in the Fault Number. Refer to the table above. These codes are only available in Version 1.60 or greater MSOC's.

If there are no fault records, a "**07: No Records Avail"** response is returned.

3. fclr requires a password as is done when changing settings,
   e.g.: fclr, password<CR><LF>

4. Returns a comma separated list of the Event Record values in the following order:
   event number, event code, time, date.

Table 4-54. Event Record Value Formats

	Parameter	Format
$$	Event Number	3 digits, RDP
$$	Event Code	3 digits, RDP
$$	Date	01/23/1997
$$	Time	01:23:45.64
$$ = Added in Version 1.6 Software		

The date will be formatted as 1997/01/23 (year/month/day) when unit is in IEC mode.
If there are no event records, a "**07: No Records Avail"** response is returned.

5. Returns a comma separated list of the Fault Record values in the following order:
fault number, fault code, time, date, mia, mib, mic, min, mvan, mvbn, mvcn

Table 4-55. Event Record Parameter Formats

	Parameter	Format
$$	Fault Number	3 digits, RDP
$$	Fault Code	Text String, 4 chrs.
$$	Date	01/23/1997
$$	Time	01:23:45.64
$$ = Added in Version 1.6 Software		

The date will be formatted as 1997/01/23 (year/month/day) when unit is in IEC mode. The load values will have the same format described previously. (Version 1.60 or greater)

If there are no fault records, a "**07: No Records Avail"** response is returned.

Logical I/O States

This section details the Reading of Inputs and Output status mappings from the MicroShield. All Logical Input State commands begin with the 'li' characters. All Logical Output State commands begin with the 'lo' characters. The li43a, liextrii, lizsc, lidtl, lorip, lolkoa, lorda, lorma commands are available as of V1.50. All previous versions will return a **03: Invalid Cmnd Syntax** response.

Logical Input State (Read Only)

This selection of Native MSOC "Fast ASCII" commands indicate whether the physical inputs have been mapped to a protective function indicator bit. As indicated, if the feature is not implemented in the queried MSOC version. All previous versions will return a **03: Invalid Cmnd Syntax** response.

Table 4-56. Logical Input Command Syntax

	Command	Response	Format	Description
	li51lt	actv, inac	Text String, 4 chrs.	51LT Logical In
	li51p	actv, inac	Text String, 4 chrs.	51P Logical In
	li50p	actv, inac	Text String, 4 chrs.	50P Logical In
	li51n	actv, inac	Text String, 4 chrs.	51N Logical In
	li50n	actv, inac	Text String, 4 chrs.	50N Logical In
	li52a	actv, inac	Text String, 4 chrs.	52a Logical In
	li52b	actv, inac	Text String, 4 chrs.	52b Logical In
	litrip	actv, inac	Text String, 4 chrs.	Ext. Trip Logical In
	licls	actv, inac	Text String, 4 chrs.	Ext. Close Logical In

$$	li43a	actv, inac	Text String, 4 chrs.	43a Logical In
$$	liextrii	actv, inac	Text String, 4 chrs.	External Reclose Initiate
$$	lizsc	actv, inac	Text String, 4 chrs.	Zone Sequence Enable
**	lirsi	actv, inac	Text String, 4 chrs.	Reset Seal-Ins
$$	lidtl	actv, inac	Text String, 4 chrs.	Drive to Lockout
$$ = Added in Version 1.6 Software ** = Added in Version 1.7 Software				

Logical Output State (Read Only)

This selection of Native MSOC “Fast ASCII” commands indicate whether the physical outputs have been mapped to a protective function indicator bit. As indicated, if the feature is not implemented in the queried MSOC version. All previous versions will return a **03: Invalid Cmnd Syntax** response.

Table 4-57. Logical Output Command Syntax

	Command	Response	Format	Description
	lo51lt	actv, inac	Text String, 4 chrs.	51LT Logical Out
	lo51p	actv, inac	Text String, 4 chrs.	51P Logical Out
	lo50p	actv, inac	Text String, 4 chrs.	50P Logical Out
	lo51n	actv, inac	Text String, 4 chrs.	51N Logical Out
	lo50n	actv, inac	Text String, 4 chrs.	50N Logical Out
	lopua	actv, inac	Text String, 4 chrs.	Pickup Alarm Logical Out
	lofail	actv, inac	Text String, 4 chrs.	Relay Fail Alarm Logical Out
	locbfl	actv, inac	Text String, 4 chrs.	CB Fail Alarm Logical Out
	lotrp	actv, inac	Text String, 4 chrs.	Trip Logical Out
	locls	actv, inac	Text String, 4 chrs.	Close Logical Out
$$	lorip	actv, inac	Text String, 4 chrs.	Reclose In Process LO
$$	lolkoa	actv, inac	Text String, 4 chrs.	Recloser LockOut Alarm LO
$$	lormax	actv, inac	Text String, 4 chrs.	Recloser Max Reclose Alarm
$$	lorda	actv, inac	Text String, 4 chrs.	Recloser Disable Alarm
$$ = Added in Version 1.6 Software				

Seal-in Output State (Read Only)

This selection of Native MSOC “Fast ASCII” commands indicate whether the physical outputs have been mapped to a latched protective function indicator bit. The latched bit feature is added in Version 1.7 of the MSOC firmware. As indicated, if the feature is not implemented in the queried MSOC version. All previous versions will return a **03: Invalid Cmnd Syntax** response

Table 4-58. Latched Element Physical Output Syntax

	Command	Response	Format	Description
**	si51lt	actv, inac	Text String, 4 chrs.	51LT Logical Out
**	si51p	actv, inac	Text String, 4 chrs.	51P Logical Out
**	si50p	actv, inac	Text String, 4 chrs.	50P Logical Out
**	si51n	actv, inac	Text String, 4 chrs.	51N Logical Out
**	si50n	actv, inac	Text String, 4 chrs.	50N Logical Out
**	sipua	actv, inac	Text String, 4 chrs.	Pickup Alarm Logical Out
**	sifail	actv, inac	Text String, 4 chrs.	Relay Fail Alarm Logical Out
**	sicbfl	actv, inac	Text String, 4 chrs.	CB Fail Alarm Logical Out
**	sitrp	actv, inac	Text String, 4 chrs.	Trip Logical Out
**	sicls	actv, inac	Text String, 4 chrs.	Close Logical Out

**	sirip	actv, inac	Text String, 4 chrs.	Reclose In Process Logical Out
**	silkoa	actv, inac	Text String, 4 chrs.	Recloser Lockout Alarm Logical Out
**	sirma	actv, inac	Text String, 4 chrs.	Recloser Max Reclose Alarm
**	sirda	actv, inac	Text String, 4 chrs.	Recloser Disable Alarm
** = Added in Version 1.7 Software				

Physical I/O States

This section details the Reading of the Physical Inputs and Output status from the MicroShield. All Physical Input State commands begin with the 'ti' characters. All Physical Output State commands begin with the 'to' characters. These functions allow for determining the state of the I/O without having to know what logicals are assigned to them.

Physical Input State (Read Only)

The Physical input command indicates whether the queried input is at a state of 1 [(energized) or actv (active)] or 0 [(de-energized) or inac (inactive)]. The command formats are listed in Table 4-59.

Table 4-59. Physical Input Command Formats

Command	Response	Format	Description
ti1	actv, inac	Text String, 4 chrs.	IN 1 State
ti2	actv, inac	Text String, 4 chrs.	IN 2 State

Physical Output State (Read Only)

The Physical output command indicates whether the queried output is at a state of 1 [de-energized (clse) closed] or 0 [(de-energized) or open]. The command formats are listed in Table 4-60.

Table 4-60. Physical Output Command Formats

Command	Response	Format	Description
to1	open, clse	Text String, 4 chrs.	OUT 1 State
to2	open, clse	Text String, 4 chrs.	OUT 2 State
to3	open, clse	Text String, 4 chrs.	OUT 3 State {Note 1}
to4	open, clse	Text String, 4 chrs.	OUT 4 State {Note 1}

NOTES:

1. Commands to3 and to4 will return a 02**: Invld Cmnd Recvd** if the unit has voltage inputs.

Clock Commands (Read/Write)

This section details the reading and writing of the Real Time Clock. All Real Time Clock Settings begin with the 'k' character. The Y2K Compliant versions of these commands are available as of V1.50. All previous versions will return a **03: Invalid Cmnd Syntax** response. Table 4-61 illustrates the command formats for each of the commands.

Table 4-61. Clock Command Syntax

	Command	Response	Format	Description
	kdate	01/23/97	Text String, 8 chrs.	Date (read/write)
	ktime	01:23:45	Text String,12 chrs.	Time (read/write)
	kts	01/23/97, 01:23:45	Text String, 21 chrs.	Time Stamp, both together
$$	kts2k	01/23/1997, 01:23:45	Test String, 23 chrs.	Time Stamp, both together Year 2000 compliant
$$	kdate2k	01/23/1997	Text String, 10 chrs	Date - Year 2000 compliant (read/write)
$$ = Added in Version 1.6 Software				

NOTES:

1. Fractional seconds are not displayed and cannot be set by the User. When setting the time, they are automatically set to 00. t is assumed that this level of time accuracy is not relevant in ASCII communications. Hundredth of seconds are returned in the time stamp of Fault and Event records.
2. Date and time can be formatted differently for IEC enabled Units.
3. When the clock is disabled, all digits will be '0.' The clock is disabled by entering '00' for the month value.
4. The commands kts and kts2k are read-only commands, used to read the unit's timestamp.

Miscellaneous Commands

This section details the reading of other miscellaneous information from the Unit and the sending of other miscellaneous commands to the Unit. All of the Miscellaneous commands begin with the 'x' character.

Table 4-62. Relay Status Commands

Command	Response	Format	Description
xrdtrg	{Note 1}		Read Targets (LED's)
xinfo	Banner, w/o name	see page 1	Unit Information
xustat	{Note 2}		Relay Status

Table 4-63 describes the correspondence between each bit in the hexadecimal converted value of the Status number returned by the **xustat** command and its' meaning. A '1' means the status is active a '0' means inactive.

Table 4-63. Status Bit Code Designations for xustat Command

Bit	Text
0	Self Test (1 = fail)
1	Physical Input Setting Change
2	New Event Record
3	New Fault Record
4	Unit Power Cycled
5	Processor Exception (failure)
6	Recloser Disabled
7	Ground Trip Disabled
8	Supervisory Control Disabled
9	Battery Failed
10	New Minimum Demand Value
11	New Peak Demand
12 - 15	Future

Physical Input Setting Change, New Event Record, New Fault Record, Unit Power Cycle, Processor Exception, New Minimum Demand Value, and New Peak Demand are cleared after they are read. Xustat is similar to Modbus Register 40129 (Relay Status).

Miscellaneous Commands (Read Only)

This section details the reading of other miscellaneous information from the Unit and the sending of other miscellaneous commands to the Unit. All of the Miscellaneous commands begin with the ‘x’ character.

Miscellaneous Information

Command	Response	Format	Description
Xrdtrg	{Note 1}		Read Targets (LED’s)
Xinfo	Banner, w/o name	See page 1	Unit Information
xustat	{Note 2}		Relay Status

Miscellaneous Commands

Table 4-64. Miscellaneous Command Definition in Fast ASCII Mode

	Command	Response	Format	Description
	xtrip	{Note 3}	{Note 6}	Operate Trip Output
	xcls	{Note 3}	{Note 6}	Operate Close Output
	x79	{Note 4}	{Note 6}	Enable/Disable 79 Function on a temporary basis
	xgnd	{Note 4}	{Note 6}	Enable/Disable Gnd
	xrsrec	{Note 5}	{Note 6}	Clear All Records
	xrsemtr	{Note 5}	{Note 6}	Reset Energy Meters
	xoc1	{Note 5}	{Note 6}	Pulse OUT1 for 1 Second
	xoc2	{Note 5}	{Note 6}	Pulse OUT2 for 1 Second
	xoc3	{Note 5}	{Note 6}	Pulse OUT3 for 1 Second
	xoc4	{Note 5}	{Note 6}	Pulse OUT4 for 1 Second
	xrstrg	{Note 5}		Reset Targets
	xslftst	pass, fail	{Note 8}	Status of Self Test Commands
	xledtst	{Note 5}		Visual test of LEDs
	xkeytst	{Note 7}		Visual test of Front Panel keys
**	xrstsi	{Note 5}	{Note 6}	Reset All Seal-Ins
	xexit	impossible		Disables Comm Port
** = Added in Version 1.7 Software				

NOTES:

1. Targets will return a comma separated list of each target name followed by its’ status. The Status target can have values of “ok” or “fail.” The others can have values of “on” or “off.”
   Example: stat: ok, pu: off, time: on, inst: off
2. xustat will return a 5 digit number with the relay status encoded. The number must be converted to hexadecimal and each individual status bit examined. t is up to the user to extract each individual status bit. See the end of this document for the encoding format.
3. xtrip and xcls will return a **09: Operation Failed** response if the action was not performed and a **01: OK** response if it was performed.
4. x79 and xgnd, are toggled on and off by this command. The response is as follows:
   10: Disabled if the command leaves the function disabled.
   11: Enabled if the command leaves the function enabled.
5. xrstrg, xledtst, xrsrec, xrsemtr, xrstsi (Rev 1.7) and xoc1 through xoc4 will return a **01: OK** response when completed.

6. xtrip, xcls, x79, xgnd, xrsrec, xrsemtr, xrstsi (Rev 1.7) and xoc1 through xoc4, require a password as is done when changing settings, e.g.:
   xtrip, password<CR><LF>
7. The User will determine the response to the xkeytst command by pressing the E or C key when prompted. If the C key is pressed, the Unit will return a **01: OK** message. If the E key is pressed, the Unit will return a **06: Test Failed** message.
8. Self Test will return a comma separated list of each test name followed by its' status. Each can have a value of "pass" or "fail."
   Example: ram: pass, prom: pass, clck: fail, eprm: pass

Protective Settings (Read/Write)

This section details the Protective Settings in the MicroShield. All of the Protective Setting commands begin with the 'p' character. In specifying these commands, the following abbreviations are used as described in Table 4-65.

Table 4-65. Protective Curve Abbreviations

Abbrev	Description
cv:	Curve
pu:	Pickup
td:	Time Dial or Delay

The Table 4-66 defines the Protective Settings. The NOTES section provides details on the different curve types defined for the different functions.

Table 4-66. Protective Setting Fast ASCII Commands

Command	Example Response	Format	Description
pcv51lt	{Note 1}	Text String, 4 chrs.	Curve for 51LT function
ppu51lt	11.75	4 digits, 2DP	pickup Amps for 51LT {See Note 1A}
ptd51lt	05.40	4 digits, 2DP	Time Dial/Delay for 51LT {See Note 1A}
pcv51p	{Note 2}	Text String, 4 chrs.	Curve for 51P function
ppu51p	11.75	4 digits, 2DP	pickup Amps for 51P {See Note 2A}
ptd51p	05.40	4 digits, 2DP	Time Dial/Delay for 51P {See Note 2A}
pcv50p	{Note 3}	Text String, 4 chrs.	Curve for 50P function
ppu50p	19.90	4 digits, 2DP	pickup X for 50P {See Note 3A}
ptd50p	09.90	4 digits, 2DP	Time Dial/Delay for 50P {See Note 3A}
pcv51n	{Note 2}	Text String, 4 chrs.	Curve for 51N function
ppu51n	11.75	4 digits, 2DP	pickup Amps for 51N {See Note 2B}
ptd51n	09.90	4 digits, 2DP	Time Dial/Delay for 51N {See Note 2B}
pcv50n	{Note 3}	Text String, 4 chrs.	Curve for 50N function
ppu50n	19.90	4 digits, 2DP	pickup X for 50N {See Note 3B}
ptd50n	09.90	4 digits, 2DP	Time Dial/Delay for 50N {See Note 3B}

NOTES:

1. Choices for 51LT curve types are as follows:

Table 4-67. 51LT Curve Type Descriptions

	51LT Curve	Description
	dis	Disabled
	ldef	Long Definite Time

	lx	Long Time Extremely Inverse
	lv	Long Time Very Inverse
	ln	Long Time Inverse
	vlx	Very Long Time Extremely Inverse
	vlv	Very Long Time Very Inverse
**	iln	IEC – Long Time Inverse
	vln	Very Long Time Inverse
** = Added in Version 1.7 Software		

1A.

- If the curve type is “def” then “ppu51lt” must be in a range from 1.0 through 10.0. This is the definition of Time Delay Dial as indicated in Table 4-64. Resolution is 0.1 second increments.
- If the curve type is “dis”, or “lx”, “lv”, “ln”, “vlx” “vlv”, “iln”, or “vln”, then “ptd51lt” must be in a range from 10 through 120 (which equals 10 through 120 seconds). This is the definition of Time Delay as indicated in Table 4-64. Resolution is in 10 second increments.
- If the curve type is “iln”, the range must be 0.01 through 1.00 with a step allocation of 0.01 (which equals a value entered from 10 through 100). This is the definition of Time Delay as indicated in Table 4-64.

2. Choices for 51P and 51N curve are as follows:

Table 4-68. 51N and 51P Curve Type Definition

	51P or 51N Curve Type	Description
	dis	Disabled (not for 51P)
	def	Definite Time #1
	x	Extremely Inverse
	v	Very Inverse
	n	Inverse
	sn	Short Time Inverse
	lx	Long Time Extremely Inverse
	lv	Long Time Very Inverse
	ln	Long Time Inverse
**	ix	IEC – Extremely Inverse
**	iv	IEC – Very Inverse
**	in	IEC – Inverse
**	def2	Definite Time #2
**	iln	IEC – Long Time Inverse
** = Added in Version 1.7 Software		

2A.

- If the curve type is “def” or “def2” then “ppu50p” must be in a range from 1.0 through 10.0 (which equals 1.0 sec to 10.0 seconds). This is the definition of Time Delay Dial as indicated in Table 4-65. Resolution is 0.1 second increments.
- If the curve type is “dis”,”x”, “v”, “n”, “sn”, “lx”,”lv”,”ln”,”ix”, “iv”, “in”,or “iln”, then “ptd51p” must be in a range from 0 through 30.0 (which equals 0.0 through 30.0 seconds). This is the definition of Time Delay as indicated in Table 4-63. Resolution is in 0.1 second increments.
- If the curve type is,”ix”, “iv”, or “in”, then “ptd51p” must be in a range from 0.05 through 1.00. This is the definition of Time Delay as indicated in Table 4-65. Resolution is in 0.05 second increments.

2B.

- If the curve type is "def" or "def2" then "ppu50n" must be in a range from 1.0 through 10.0 (which equals 1.0 sec to 10.0 seconds).This is the definition of Time Delay Dial as indicated in Table 4-65. Resolution is 0.1 second increments.
- If the curve type is "dis","x", "v", "n", "sn", "lx","lv","ln","ix", "iv", "in", or "lln", then "ptd51n" must be in a range from 0 through 30.0 (which equals 0.0 through 30.0 seconds). This is the definition of Time Delay as indicated in Table 4-65. Resolution is in 0.1 second increments.

3. Choices for 50P and 50N curves are as follows:

Table 4-69. 50P and 50N Curve Selection Definitions

	50P and 50N Curve Types	Description
	dis	Disabled
	def	Definite Time #1
	sn	Short Time Inverse
**	def2	Definite Time #2
	std	Standard (no delay)
** = Added in Version 1.7 Software		

3A.

- If the curve type is "def" or "def2" then "pcv50p"must be in a range from 1.0 through 10.0 (which equals 1.0 sec to 10.0 seconds). This is the definition of Time Delay Dial as indicated in Table 4-65. Resolution is 0.1 second increments.
- If curve type is "dis", "sn", "795", or"std" then "ppu50p" must be in a range from 0 through 30.0 (which equals 0.0 through 30.0 seconds). This is the definition of Time Delay as indicated in Table 4-65. Resolution is in 0.1 second increments.

3B.

- If the curve type is "def" or "def2" then "pcv50n"must be in a range from 1.0 through 10.0 (which equals 1.0 sec to 10.0 seconds). This is the definition of Time Delay Dial as indicated in Table 4-65. Resolution is 0.1 second increments.
- If curve type is "dis", "sn", "795", or"std" then "ppu50n" must be in a range from 0 through 30.0 (which equals 0.0 through 30.0 seconds). This is the definition of Time Delay as indicated in Table 4-65. Resolution is in 0.1 second increments.

Recloser Settings (Read/Write)

This section details the reading and writing of Recloser Settings to/from the MicroShield. All of the Recloser Settings commands begin with the 'r' character. The following abbreviations listed in Table 4-70 are used: These commands are available as of V1.60. All previous versions will return a **08: Setting Not Avail** response.

Table 4-70. Recloser Command Codes

	Recloser Command Code	Description
$$	of:	Open Time
$$	s:	Selected Functions
$$	791:	1st Reclose
$$	792:	2nd Reclose
$$	793:	3rd Reclose
$$	794:	4th Reclose
$$	795:	5th Reclose
$$ = Added in Version 1.6 Software		

Table 4-71 defines the Recloser Settings. The NOTES section provides details on the different response types defined for the different functions.

Table 4-71. Recloser Setting Command Codes

	Command	Response	Format	Description
$$	rot791	199.90	5 digits, 2DP	Open Timer 79-1 0.1< = Range <= 200.0
$$	rs79150p	{Note 1}	Text String, 4 chrs.	Selection for 50P Trip
$$	rs79150n	{Note 1}	Text String, 4 chrs.	Selection for 50N Trip
$$	rs79151n	{Note 1}	Text String, 4 chrs.	Selection for 51N Trip
$$	rs79151l	{Note 1}	Text String, 4 chrs.	Selection for 50LT Trip
$$	rs79151p	{Note 2}	Text String, 4 chrs.	Selection for 50P Lockout
$$	rot792	199.90	5 digits, 2DP	Open Timer 79-2 0.1< = Range <= 200.0
$$	rs79250p	{Note 1}	Text String, 4 chrs.	Selection for 50P Trip
$$	rs79250n	{Note 1}	Text String, 4 chrs.	Selection for 50N Trip
$$	rs79251n	{Note 1}	Text String, 4 chrs.	Selection for 51N Trip
$$	rs79251l	{Note 1}	Text String, 4 chrs.	Selection for 50LT Trip
$$	rs79251p	{Note 2}	Text String, 4 chrs.	Selection for 50P Lockout
$$	rot793	199.90	5 digits, 2DP	Open Timer 79-3 0.1< = Range <= 200.0
$$	rs79350p	{Note 1}	Text String, 4 chrs.	Selection for 50P Trip
$$	rs79350n	{Note 1}	Text String, 4 chrs.	Selection for 50N Trip
$$	rs79351n	{Note 1}	Text String, 4 chrs.	Selection for 51N Trip
$$	rs79351l	{Note 1}	Text String, 4 chrs.	Selection for 50LT Trip
$$	rs79351p	{Note 2}	Text String, 4 chrs.	Selection for 50P Lockout
$$	rot794	199.90	5 digits, 2DP	Open Timer 79-4 0.1< = Range <= 200.0
$$	rs79450p	{Note 1}	Text String, 4 chrs.	Selection for 50P Trip
$$	rs79450n	{Note 1}	Text String, 4 chrs.	Selection for 50N Trip
$$	rs79451n	{Note 1}	Text String, 4 chrs.	Selection for 51N Trip
$$	rs79451l	{Note 1}	Text String, 4 chrs.	Selection for 50LT Trip
$$	rs79451p	{Note 2}	Text String, 4 chrs.	Selection for 50P Lockout
$$	rs79550p	{Note 1}	Text String, 4 chrs.	Selection for 50P Trip
$$	rs79550n	{Note 1}	Text String, 4 chrs.	Selection for 50N Trip
$$	rs79551n	{Note 1}	Text String, 4 chrs.	Selection for 51N Trip
$$	rs79551l	{Note 1}	Text String, 4 chrs.	Selection for 50LT Trip
$$	rrt79	200	3 digits, RDP	Reset Time 0.1< = Range <= 200.0
$$	rcot79	200, dis {Note 3}	3 digits, RDP	Cutout Time 0.1< = Range <= 200.0
$$	rrect79	9999, dis {Note 3}	4 digits, RDP	Recovery Time
$$	rlim79	9999	4 digits, RDP	Limit Time 0< = Range <= 9999
$$	rrtl79	9999, dis {Note 3}	4 digits, RDP	Reclosures to Lockout 0< = Range <= 9999
$$	rlimc79	99	2 digits, RDP	Recovery Count 0< = Range <= 99
$$ = Added in Version 1.6 Software				

NOTES:

1. Choices for Selection for Trip are as follows:

Table 4-72. Trip Setting Codes for Reclosure Function

	Trip Setting Codes	Description
$$	enb	Function Enabled
$$	lko	Sends to Lockout (same as enb for 795)
$$	dis	Function Disabled
$$ = Added in Version 1.6 Software		

2. Choices for 51P Lockout are as follows:

Table 4-73. 51P Lockout Setting Codes for Reclosure Function

	Close Setting Codes	Description
$$	lko	Sends to Lockout
$$	enb	Enabled Function
$$ = Added in Version 1.6 Software		

3. The rcot79, rrect79, and rrtl79 parameters have disable options.

Configuration Settings (Read/Write)

This section details the Reading of Configuration Settings from the MicroShield. All of the Configuration Settings commands begin with the ‘c’ character.

Table 4-74. Configuration Command Settings

Command	Response	Format	Description
cpctr	2000	4 digits, RDP	Phase CT Ratio 0001<=Range<=2000
cnctr	2000	4 digits, RDP	Neutral CT Ratio 00001<=Range<=2000
cvtr	1000	4 digits, RDP	VT Ratio 0001<=Range<=1000
cvtcon	wye, delt	Text String, 4 chrs.	VT Connection type
ctft	60	2 digits, RDP	Trip Fail Time 01<=Range<=60 Cycles
cdtc	15	2 digits, RDP	Demand Time Constant 0<=Range<=60 in 15 minute intervals
crstmd	dly, ins	Text String, 3 chrs.	51 Reset mode
cvdmd	ln, ll	Text String, 2 chrs.	V Display Mode
cclmd	sec, min, dis	Text String, 3 chrs.	Cold Load Time Mode
cclt	254	3 digits, RDP	Cold Load Time 000<=Range<=255
cclft	123	3 digits, RDP	Close Fail Time 000<=Range<=999

NOTES:

1. There is no method supported for changing the password
2. If cclmd is set to dis, then cclt will respond with **08: Setting Not Avail**

Programmable Input/Output Settings

This section details the Programmable Input and Output settings from the MicroShield. The MSOC allows Boolean logic of the inputs and outputs tied to protective functions as described in Tables 4-75 through 4-78. All Programmable Input Settings begin with the 'i' character. All Programmable Output Settings begin with the 'o' character. Seal In control commands (included in Version 1.70) begin with the 's' character. The i43a, iextri, izsc, idtl, orip, olkoa, orda, and orma commands are available as of V1.50. All previous versions will return a **08: Setting Not Avail** response.

Programmable Input Settings (Read/Write)

Table 4-75 lists the mapping of the physical inputs to the protective function. The inputs may be discretely mapped or mapped as a function of the "AND" or "OR" boolean function assignment. Table 4-78 lists the assignment codes for mapping the Input Settings.

Table 4-75. Programmable Input Setting Codes

	Command	Response	Format	Description
	i51lt	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	51LT Logical In Mapping
	i51p	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	51P Logical In Mapping
	i50p	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	50P Logical In Mapping
	i51n	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	51N Logical In Mapping
	i50n	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	50N Logical In Mapping
	i52a	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	52a Logical In Mapping
	i52b	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	52b Logical In Mapping
	itrip	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	Ext. Trip Logical In Mapping
	icls	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	Ext. Close Logical In Mapping
	i43a	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	43a Logical In Mapping
	iextrii	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	EXTRI Logical In Mapping
	izsc	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	ZSC Logical In Mapping
	idtl	in1,in2,1or2,1ad2,nm	Text String, 4 chrs.	DTL Logical In Mapping
**	irsi	in1,in2,1or2,1ad2,nm	Text String, 4 chrs	RSI Logical In Mapping
** = Added in Version 1.7 Software (See Table 4-71)				

Programmable Output Settings (Read/Write)

Table 4-76 lists the mapping of the physical outputs to the protective function. The outputs are discretely mapped. Table 4-78 lists the assignment codes for mapping the Output Settings.

Table 4-76. Programmable Output Setting Codes

Command	Response	Format	Description
o51lt	o1,o2,o3,o4,nm	Text String, 2 chrs.	51LT Logical Out Mapping
o51p	o1,o2,o3,o4,nm	Text String, 2 chrs.	51P Logical Out Mapping
o50p	o1,o2,o3,o4,nm	Text String, 2 chrs.	50P Logical Out Mapping
o51n	o1,o2,o3,o4,nm	Text String, 2 chrs.	51N Logical Out Mapping
o50n	o1,o2,o3,o4,nm	Text String, 2 chrs.	50N Logical Out Mapping
opua	o1,o2,o3,o4,nm	Text String, 2 chrs.	Pickup Alarm Log. Out Mapping
ofail	o1,o2,o3,o4,nm	Text String, 2 chrs.	Relay Fail Alarm Log. Out Mapping
ocbfl	o1,o2,o3,o4,nm	Text String, 2 chrs.	CB Fail Alarm Log. Out Mapping
otrp	o1,o2,o3,o4,nm	Text String, 2 chrs.	Trip Logical Out Mapping
ocls	o1,o2,o3,o4,nm	Text String, 2 chrs.	Close Logical Out Mapping
orip	o1,o2,o3,o4,nm	Text String, 2 chrs.	Reclose in Progress

olkoa	o1,o2,o3,o4,nm	Text String, 2 chrs.	Recloser Lockout Alarm
orda	o1,o2,o3,o4,nm	Text String, 2 chrs.	Recloser Disabled Alarm Mapping
ormax	o1,o2,o3,o4,nm	Text String, 2 chrs	Recloser Max Reclose Alarm

Programmable Seal-In Output Settings

Table 4-77 lists the mapping of the “Seal In” physical outputs to the protective function. The outputs may be discretely mapped or mapped as a function of the “AND” or “OR” boolean function assignment. Table 4-78 lists the assignment codes for mapping the Output Settings.

Table 4-77. Programmable Seal In Output Setting Codes

	Command	Response	Format	Description
**	so51lt	o1,o2,o3,o4,nm	Text String, 2 chrs.	51LT Logical Out Mapping
**	so51p	o1,o2,o3,o4,nm	Text String, 2 chrs.	51P Logical Out Mapping
**	so50p	o1,o2,o3,o4,nm	Text String, 2 chrs.	50P Logical Out Mapping
**	so51n	o1,o2,o3,o4,nm	Text String, 2 chrs.	51N Logical Out Mapping
**	so50n	o1,o2,o3,o4,nm	Text String, 2 chrs.	50N Logical Out Mapping
**	sopua	o1,o2,o3,o4,nm	Text String, 2 chrs.	Pickup Alarm Log. Out Mapping
**	sofail	o1,o2,o3,o4,nm	Text String, 2 chrs.	Relay Fail Alarm Log. Out Mapping
**	socbfl	o1,o2,o3,o4,nm	Text String, 2 chrs.	CB Fail Alarm Log. Out Mapping
**	sotrp	o1,o2,o3,o4,nm	Text String, 2 chrs.	Trip Logical Out Mapping
**	socls	o1,o2,o3,o4,nm	Text String, 2 chrs.	Close Logical Out Mapping
**	sorip	o1,o2,o3,o4,nm	Text String, 2 chrs.	Reclose in Progress
**	solkoa	o1,o2,o3,o4,nm	Text String, 2 chrs.	Recloser Lockout Alarm
**	sorda	o1,o2,o3,o4,nm	Text String, 2 chrs.	Recloser Disabled Alarm Mapping
**	sorm	o1,o2,o3,o4,nm	Text String, 2 chrs	Recloser Max Reclose Alarm
** = Added in Version 1.7 Software				

Table 4-78. Programmable Input, Output, and Seal–In nm Code Listing

Code	Description
nm	not mapped
1or2	input 1 OR input 2
1ad2	input 1 AND input 2
o1	output 1
o2	output 2
o3	output 3
o4	output 4

Master Trip Output Settings

This section details the Master Trip Output settings from the MicroShield. All Master Trip Output settings begin with the ‘q’ character. The assignment codes in the “Fast ASCII” Mode are detailed below in Table 4-79.

Table 4-79. Master Trip Output Setting Assignment Codes

Command	Response	Format	Description
qmt51lt	off, on	Text String, 2 chrs.	51LT to Trip Output
qmt51p	off, on	Text String, 2 chrs.	51P to Trip Output
qmt50p	off, on	Text String, 2 chrs.	50P to Trip Output
qmt51n	off, on	Text String, 2 chrs.	51N to Trip Output
qmt50n	off, on	Text String, 2 chrs.	50N to Trip Output

Communications Settings & Information (Read/Write)

This section details the Reading of Communications Settings and Information. All Communications commands begin with the 'n' character. The ntxdly command is available as of V1.60. All previous versions will return a **08: Setting Not Avail** response. Table 4-80 lists the commands available for Communication Setting Parameters.

Table 4-80. Communication Setting Parameter Assignment Codes

	Command	Response	Format	Description
	naddr	1234	4 digits, RDP	Network Address 0<= Range <= 255
	nfprot	asci, mdbs	Text String, 4 chrs.	Front Port Protocol
	nfbaud	{Note 1}	4 digits, RDP	Baud Rate, Front Port
	nffrm	{Note 2}	Text String, 3 chrs.	Framing, Front Port
	nfech	on, off	Text String, 1 chrs.	Front Port Echo
	nrprot	asci, mdbs	Text String, 4 chrs.	Rear Port Protocol
	nrbaud	{Note 1}	4 digits, RDP	Baud Rate, Rear Port
	nrfrm	{Note 2}	Text String, 3 chrs.	Framing, Rear Port
$$	ntxdly	123 {Note 3}	4 digits, RDP	Transmit Delay Time
$$ = Added in Version 1.6 Software				

NOTES:

1. Possible choices for baud rate are: 1200, 2400, 4800, 9600, 19200
2. Possible choices for framing are: o71, e71, n81, o72, e72
3. Range is 0 to 200 in increments of 5. Any value not divisible by 5 will be rounded down.

Breaker Counter Settings

This section details the breaker counter for the MicroShield. All breaker counters begin with a 'b' prefix. These commands are available as of V1.60. Versions prior to Version 1.60 will return a **08: Setting Not Avail** response. Version 1.60 software allows selection of reclosure for up to 4 shots with the fifth event initiating lockout. Breaker counters allow diagnostic evaluation of operation aiding in system maintenance evaluation. Table 4-81 lists the breaker counter setting parameter lists.

Table 4-81. Breaker Counter Setting Command List

	Command	Response	Format	Description
$$	bcops	12345	5 digits, RDP	Total Breaker Operations 0<= Range <= 9999
$$	bcksia	12345	5 digits, RDP	KSIA Kiloamperes Phase A Current 0<= Range <= 9999
$$	bcksib	12345	5 digits, RDP	KSIB Kiloamperes Phase B Current 0<= Range <= 9999
$$	bcksic	12345	5 digits, RDP	KSIC Kiloamperes Phase C Current 0<= Range <= 9999
$$	bc791	12345	5 digits, RDP	First Reclose Counter 0<= Range <= 9999
$$	bc792	12345	5 digits, RDP	Second Reclose Counter 0<= Range <= 9999
$$	bc793	12345	5 digits, RDP	Third Reclose Counter 0<= Range <= 9999
$$	bc794	12345	5 digits, RDP	Fourth Reclose Counter 0<= Range <= 9999
$$	bctrcl	12345	5 digits, RDP	Total Recloser Counter

				0<= Range <= 9999
$$	bcoctrp	12345	5 digits, RDP	Overcurrent Trip Counter 0<= Range <= 9999
$$	bcrclok	12345	5 digits, RDP	Successful Reclose Counter 0<= Range <= 9999
$$ = Added in Version 1.6 Software				

MSOC Native ASCII Error Codes

This section describes error codes returned from the Unit. Each code is preceded by a number so the code can be quickly determined by an automated system. An error code will be returned in “Fast ASCII mode” if a <lf><cr> is received by the MSOC. If no <lf><cr> is received, the host will experience a timeout since the MSOC will not respond to a host request.

The current list of codes is:

00: Invld Err Code This error is used by us as a check on the error code transmission software. The user should never see this once we debug the code.
01: OK Returned when a setting has been changed or action has been performed successfully.
02: Invld Cmnd Recvd Returned when the prefix of the command received does not match any in the table.
03: Invld Cmnd Syntx Returned if the received command does not match any in the list for that category.
04: Invld Pswrd Returned when the password included with the command does not match the password programmed in the relay.
05: Invld Param Returned when the parameter sent with the command: contains too many characters, contains invalid characters, or is out of range.
06: Test Failed Returned when the User indicates from the front panel that the key test has failed.
07: No Records Avail Returned when a fault or event record is requested and none are currently logged.
08: Setting Not Avail This setting is current unavailable to the User.
09: Operation Failed Returned when trip and fail commands are not performed, probably because breaker state is unknown.
10: Disabled Returned when the function was left disabled.
11: Enabled Returned when the function was left enabled.

MSOC Native ASCII “Fast Mode” Timing Analysis

Network timing is predicated upon the following factors:

1. Host Latency (How long does it take a host device to generate a command, receive the response and interpret the data).
2. Intermediate Device Latency (If a Modem, data concentrator or other device is between the end device required for data retrieval, how long does it take for each device to receive the command and process it downline to the next device).
3. MSOC Device Latency (How long does it take for an MSOC to receive a command, and return a response to the network).
4. Baud Rate (How fast is each data bit propagated on the medium).
5. Protocol Efficiency [Network Bandwidth Utilization] (Does the protocol utilized allow for the issuance of another command before a response is received from an outstanding communication request).

The common question to a network system engineer is usually “How fast can I get my relay alarm data to appear on the screen?” An analysis of the amount of data and the above 5 areas is required.

Host latency varies widely by manufacturer or the PC or host computer. Software speed and port access varies widely. Most manufacturers of these hardware and software platforms have general benchmarks to supply to the users for processing time once the device acquires the data from the communication port.

Intermediate Device Latency also varies from the type of device used. Some modems have a device turnaround of 5 mS per transactions whereas; a radio modem may require hundreds of mS to obtain an open frequency from which to transmit.
This section shall illustrate and explain a simple network transaction based upon a simple point to point communication from a single MSOC to a host device as illustrated in Figure 4-43 of this document. This example shall exclude SCADA Master host latency.

Fast ASCII Baud Rate Analysis

The rate of the data transfer is determined by the selected baud rate. The faster the baud rate, the faster the communication. The MSOC supports baud rates of 1200, 2400, 4800, 9600 and 19200. The effect of transfer time is shown in Table 4-82. Each bit has specific transfer time which correlates to a specific character transfer time.

Table 4-82. Character Transfer Time vs Baud Rate

Baud Rate	Transfer Time Per Bit	Transfer Time Per Character
1200	0.833 mS	8.333 mS
2400	0.4167 mS	4.167 mS
4800	0.2083 mS	2.083 mS
9600	0.1041 mS	1.041 mS
19200	0.0521 mS	0.521 mS

Each Fast ASCII transfer varies in the amount of bytes transmitted and requested. Table 4-77 lists the amount of fixed data per some of the common Fast ASCII commands. For example, each data transmission contains the following characters as per Figure 4-27:

- Line Feed (One Character)
- Carriage Return (One Character)

Each base transmitted and received command has at least 2 characters for transmission. The transmission time, depending upon baud rate can range from 16.666 mS (at 1200 baud) to 1.041 mS (at 19200 baud). For example Figure 4-45 illustrates the mia Read metering phase Ia current. Figure 4-45 illustrates the transaction request for Ia. Analysis of the data transmitted and received yields the following:

Transmission Request:
Common characters 2 + 3 command characters (mia)
Total characters for transmission request = 5 characters.

Returned Response:
Common characters 2 + 5 data byte characters
Total characters returned by MSOC = 7 characters.

Depending upon the baud rate the total time for the communication characters to propagate along the network could range from:

Transmission Time Request at 1200 Baud:
41.65 mS=5 characters * 10 bits per character * 0.833 mS per bit/character
Transmission Response at 1200 Baud
51.31 mS= 7 characters * 10 bits per character * 0.833 mS per bit/character
Transmission Time Request at 19200 Baud:
2.065 mS=5 characters * 10 bits per character * 0.0521 mS per bit/character
Transmission Response at 19200 Baud

3.672 mS= 7 characters * 10 bits per character * 0.0521 mS per bit/character

Total network transfer time via the physical medium can range from 92.96 mS at 1200 baud to 7.737 mS at 19200 Baud.

Baud rate is a major influence at 1200 baud and a lesser influence at 19200 baud. It must be realized that this is only one of three components analyzed for a complete throughput analysis. In this case the Host time to generate the command.

MSOC Throughput Analysis

Communication implementation within a protective relay is a demanding task. In other devices, communications may take first priority. Within an ABB protective relay PROTECTION IS THE FIRST PRIORITY**.** Communication shall not compromise protection capabilities. Thus communication throughput may vary depending upon the demands of the protection. Table 4-83 illustrates the MSOC minimum and maximum benchmark times for recognition of a Fast ASCII command at the physical port and the time it takes to generate a reply to the respective port. The times listed in the table are average times and do not include the calculated data string transmission and receptions values.

Table 4-83. MSOC Fast ASCII Command Throughput (Average Time in mS)

Command	Overall Time(mS) min	Overall Time(mS) max	Comments
Blank cmd (CR)	18.828	18.728	
xspy2	48.0396	47.9396	
mia	8.2056	8.1056	
mvlst	58.662	58.562	10 parameters
dia	10.8612	10.7612	5 dig #
dplst	21.4836	21.3836	2 X 7 dig #'s
ecurevt	24.1392	24.0392	dumps a record
fcurflt	16.1724	16.0724	No fault records
li51lt	10.8612	10.7612	resp: inac
lidtl	13.5168	13.4168	resp: inac
lo51lt	8.2056	8.1056	resp: inac
lormax	10.8612	10.7612	resp: inac
ti1	8.2056	8.1056	resp: inac
to1	10.8612	10.7612	resp: open
kts	26.7948	26.6948	
kts2k	29.4504	29.3504	
xinfo	26.7948	26.6948	
pcv51lt,ldef	1458.1632	1458.0632	
ppu51lt,1145	1402.3956	1402.2956	
ptd51lt,6.40	1450.1964	1450.0964	
rot791,155.32	2809.8636	2809.7636	
rs79150p,lko	2812.5192	2812.4192	
ptd50n,2.45	1455.5076	1455.4076	
rrtl79,2000	2817.8304	2817.7304	
cpctr,1000	823.4748	823.3748	
cclft,100	823.4748	823.3748	
i51lt,in1	674.7612	674.6612	
i51p,1or2	674.7612	674.6612	

i51p,1ad2	674.7612	674.6612	
i50p,in2	672.1056	672.0056	read in 6639
i51n,1or2	669.45	669.35	
i50n,in1	672.1056	672.0056	
i52a,in1	674.7612	674.6612	
i52a,in2	674.7612	674.6612	
itrip,1or2	674.7612	674.6612	
icls,in2	674.7612	674.6612	
i43a,1ad2	674.7612	674.6612	
iextri,in1	672.1056	672.0056	
izsc,1or2	674.7612	674.6612	
idtl,1ad2	674.7612	674.6612	
o51lt,o1	722.562	722.462	
ormax,o2	722.562	722.462	
qmt51lt,off	295.0104	294.9104	
naddr,100	844.7196	844.6196	Port Reset!
drprot,mdbs	5.55	5.45	Port Reset!
nrfrm,e72	855.342	855.242	Port Reset!
bcops,1000	24.1392	24.0392	RAM Based
bcrclok,2000	26.7948	26.6948	RAM Based
bc791,500	26.7948	26.6948	RAM Based

For the example, the MSOC generation time for mia (Phase a Current) ranges from 8.1056 to 8.2056 mS.

Final Throughput Calculation and Analysis

A final calculation of our example throughput is warranted. For this example, the host update time shall now be assumed to be 250 mS. This 250 mS shall be an example estimation or time to generate a command, interpret the received command and update the screen. This is just for this example and varies according to:

- Speed of the host processor (hardware bus structure, # of processors, video card update, RAM memory, microprocessor speed…)
- Operating system selected (LINUX, UNIX, OS2, WIN NT, WIN 3.1, WIN 98, WIN 95)
- MMI Port Driver Efficiency (PRICOM, Power RICH, Wonderware, USDATA)

Two results will be calculated, operation at 1200 and 19200 baud. The example is described as per Figure 4-39 within this document. The formula used to produce the typical response is:

System Throughput = Host Processing Time + String Transmit Time + MSOC Processing Time + String Reception Time

At 1200 Baud:
351.17 mS = 250 mS + 41.65 mS + 8.2056 mS +51.31 mS

At 19200 Baud:
263.93 mS = 250 mS + 2.07 mS + 8.2056 mS + 3.67 mS

Figures 4-47 and 4-48 illustrate the individual contributions from each of the components as a percentage of total transaction time.

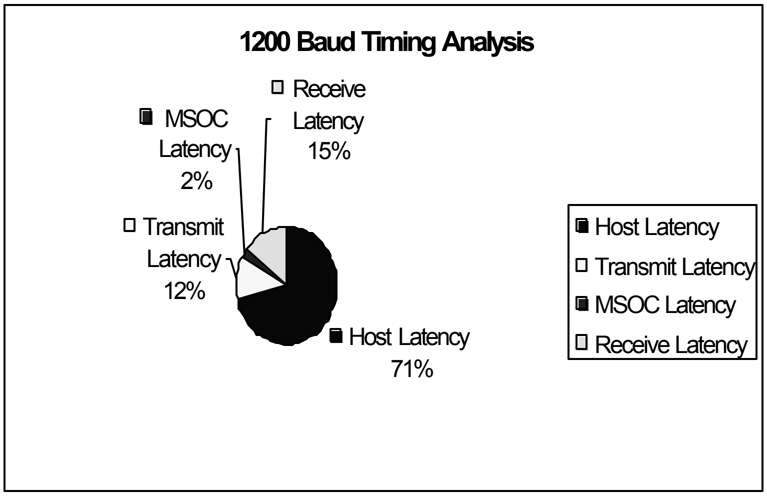


Figure 4-47. Network Throughput Analysis at 1200 Baud

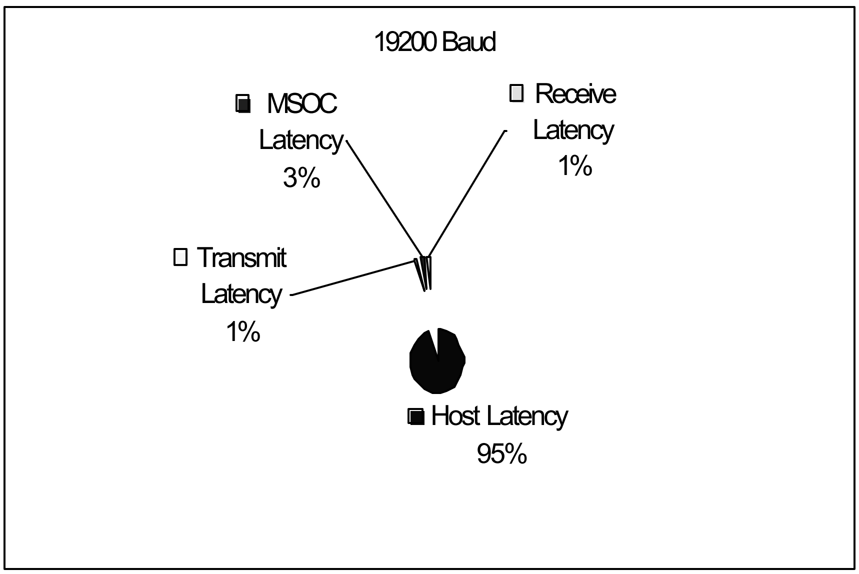


Figure 4-48. Network Throughput Analysis at 19200 Baud

Analysis of the simple example yields a few points to be considered when analyzing system throughput. Each element involved in communication timing contributes significantly to overall throughput. If the host executed and updated faster, overall throughput could be improved. If intermediate devices were inserted within the network, transaction time would increase proportionately. Baud rate is only one of many contributing factors in calculating system throughput. If one network access was required for retrieval of system data, overall network efficiency would be improved. If in a networked system, the protocol utilized would allow for additional data request commands while the slave device is processing a response, network throughput time would be improved.

Virtual treatises have been written on improving system throughput and data updates. The simple example illustrates and allows the user to calculate system throughput times. This is especially critical so that the system user will not be surprised with overall system response.

APPENDIX A
ASCII Coding Chart

Decimal Value	Hexidecimal Value	Control Character	Character	Decimal Value	Hexidecimal Value	Control Character	Character
0	00	NUL (CTRL @)	Null	50	32		2
1	01	SOH (CTRL A)		51	33		3
2	02	STX (CTRL B)		52	34		4
3	03	ETX (CTRL C)		53	35		5
4	04	EOT (CTRL D)		54	36		6
5	05	ENQ (CTRL E)		55	37		7
6	06	ACK (CTRL F)		56	38		8
7	07	BEL (CTRL G)	Beep	57	39		9
8	08	BS (CTRL H)	Backspace	58	3A		
9	09	HT (CTRL I)	Tab	59	3B		
10	0A	LF (CTRL J)	Line-feed	60	3C		<
11	0B	VT (CTRL K)	Cursor home	61	3D		
12	0C	FF (CTRL M)	Form-feed	62	3E		>
13	0D	CR (CTRL N)	Carriage Return	63	3F		?
14	0E	SO (CTRL O)	Shift Out	64	40		@
15	0F	SO	Shift In	65	41		A
16	10	DLE	Data Link Escape	66	42		B
17	11	DCI		67	43		C
18	12	DC2		68	44		D
19	13	DC3		69	45		E
20	14	DC4		70	46		F
21	15	NAK		71	47		G
22	16	SYN		72	48		H
23	17	ETB		73	49		I
24	18	CAN		74	4A		J
25	19	EM		75	4B		K
26	1A	SUB		76	4C		L
27	1B	ESC		77	4D		M
28	1C		Cursor right	78	4E		N
29	1D		Cursor left	79	4F		O
30	1E		Cursor up	80	50		P
31	1F		Cursor down	81	51		Q
32	20		Space	82	52		R
33	21		!	83	53		S
34	22		“	84	54		T
35	23		#	85	55		U
36	24		$	86	56		V
37	25		%	87	57		W
38	26		&	88	58		X
39	27		‘	89	59		Y
40	28		(	90	5A		Z
41	29		)	91	5B		[
42	2A		*	92	5C		\
43	2B		+	93	5D		]
44	2C		,	94	5E		^
45	2D		-	95	5F		_
46	2E		.	96	60		`
47	2F		/	97	61		a
48	30		0	98	62		b
49	31		1	99	63		c

Decimal Value	Hexidecimal Value	Control Character	Character	Decimal Value	Hexidecimal Value	Control Character	Character
100	64		d	114	72		r
101	65		e	115	73		s
102	66		f	116	74		t
103	67		g	117	75		u
104	68		h	118	76		v
105	69		i	119	77		w
106	6A		j	120	78		x
107	6B		k	121	79		y
108	6C		l	122	7A		z
109	6D		m	123	7B		{
110	6E		n	124	7C		|
111	6F		o	125	7D		}
112	70		p	126	7E		~
113	71		q	127	7F		DEL

APPENDIX B

RS 232 Device Connectivity

RS 232 Point to Point Connectivity for the DPU/TPU/GPU 2000(R) Protection Units

Today's requirements for rapid data acquisition necessitate understanding the physical interface requirements for connection of a DPU2000(R), TPU2000(R), or GPU2000R. One of the many available options within the ABB DPU2000(R), TPU2000(R), or GPU2000R units is RS 232 connectivity. What follows is an explanation of RS 232 serial port connectivity and physical connection considerations in a point to point communication architecture.

What is RS 232?

RS 232 is perhaps the most utilized and least understood communication interface in use. RS 232 is sometimes misinterpreted to be a protocol; it is in fact a physical interface. A physical interface is the hardware and network physical media used to propagate a signal between devices. Examples of physical interfaces are RS 232 serial link, printer parallel port, current loop, V. 24, IEEE Bus… Examples of network media are, twisted copper pair, coaxial cable, free air…

A protocol is a specific set of rules and regulations regulating data transmission and reception. Examples of protocols are: DNP 3.0, Modbus, INCOM, etc., which can communicate over a variety of physical interfaces and media. Every DPU2000(R), TPU2000(R), or GPU2000R protective device includes an RS 232 connection as part of the base product.

RS 232 gained widespread acceptance due to its ability to connect to another RS 232 device or to a modem. A modem is a device, which takes a communication signal and modulates it into another form. Common forms of modems include telephone, fiber optic, microwave, and radio frequency. Modem connectivity allows attachment of multiple devices on a communication network or allows extension of communication distances in a network with two nodes. Physical connection of two devices or more than two devices require differing approaches. This note shall explore point to point serial communication in depth and illustrate its inclusion into substation applications. The first RS 232 specification issued by the Electronic Industries Association defined the interface as consisting of 25 pins within a D shell connector. One of four categories of functionality is assigned to each pin (ground, data, control and timing). With the introduction of the IBM PC AT, the RS 232 port was reduced in form factor and functionality to consist of 9 pins within a D shell connector. That has been the de-facto standard for RS 232 port implementation included on most electronic equipment. The DPU/TPU/GPU 2000(R) family of protective devices includes at least one RS 232 port emulating a common utility industry protocol.

RS 232 Connectivity Issues

Common issues arise when two devices communicate. Typical communication architectures are illustrated in Figures B1 and B2. Configuration of an ABB DPU2000(R), TPU2000(R), or GPU2000R family of protection relays is accomplished via a software program called ECP (External Communication Program). Figure B1 illustrates a typical connection in a point to point topology. DPU2000R configuration through a personal computer requires two steps, 1.) Correct physical interconnection between the PC and DPU2000, 2.) Configuration of the DPU and the PC with a common protocol. The DPU2000(R), TPU2000(R), or GPU2000R protocol may be assigned to each port through a configuration sequence via the front keypad of the device. Please refer to the respective product manuals for communication port configuration instructions.

Figure B2 illustrates another typical communications topology, multi-drop. A typical implementation of multi-drop architecture is one in which a host computer (for example, a personal computer with Pricom SE software) collects metering information from a distributed set of protective relays. In this case a great distance separates the personal computer from each device. However, an RS 232 port is available on each node. Device interconnection is possible through the addition of a modem or line driver. Interconnection of these devices requires a different setup and implementation process. The topic of multi-drop topology is covered in other application notes.

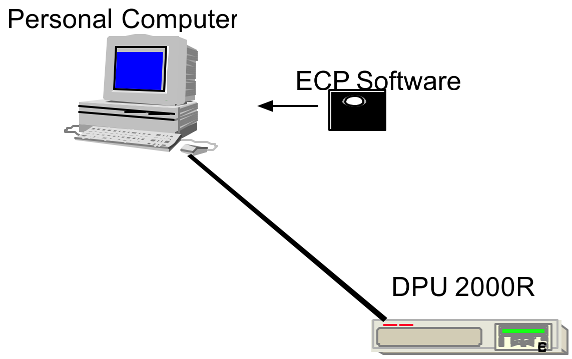


Figure B1. Point to Point Topology

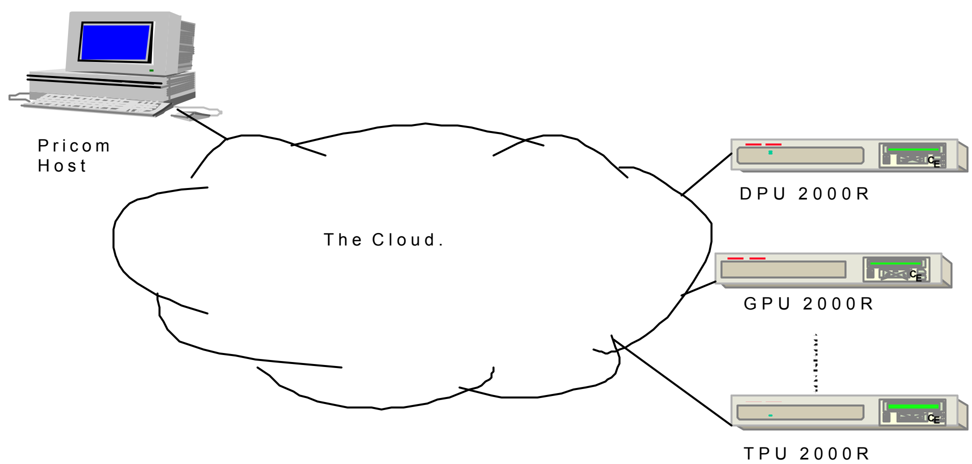


Figure B2. Multi-Drop Topology

Point to point connectivity can be accomplished via a copper cable connection. The physical copper cable can be referred to as the physical medium. Other types of physical medium can be air (used in radio or microwave modems), as well as plastic or glass fiber (used in fiber optics).

EIA specifications of an RS 232 port provide port and line characteristics. The applied communication signal transmission voltage, can range from 5 to 25 Volts (Space or 0) or –5 to –25 Volts (Mark or 1) through a load impedance with a resistance of 3,000 ohms to 7,000 ohms and a shunt capacitance of no more than 2500 picofarads. If these electrical characteristics are met, a signal can propagate up to 50 feet. If the distance between nodes is greater than 50 feet, additional devices may be utilized to extend communication distances. These devices are known as short haul line drivers, telephone modems, radio frequency modems or microwave modems. Some device manufacturers lower voltage on the RS 232 port to a level of 3 volts. While a bipolar 3-volt level is within the electrical range of the RS 232 specification, transmission length is severely decreased.

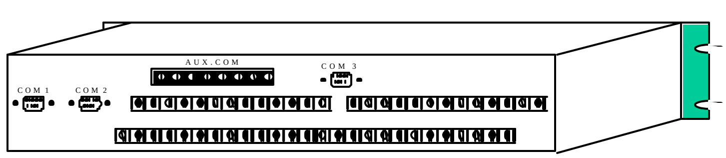


Figure B3. Rear Panel Com Port Location

The RS 232 ports on the protective relay are designated as Com 0, Com 1, Com 2, and Com 3. Com 0 is located on the front of the unit. As illustrated on FigureB3, Com ports 1 through 3 are located on the back of the unit. The inherent design of the RS 232 electrical interface is designed for point to point communication over copper medium. The RS 232 pin designations on the DPU2000(R), TPU2000(R), or GPU2000R is illustrated in Figures B4 and B5.

Port Isolation

Network installation within a substation requires special considerations. A substation environment is harsh in that high levels of electromagnetic interference are present. Additional ground currents are present in such installations. RS 232 is an unbalanced network in that all signals are referenced to a common ground. On longer cable runs, the potential of the signals at the sending device can be significantly lower than at the receiving end due to electrical interference and induced ground current. This increases with long runs of cable and use of unshielded cable. ABB's Substation Automation and Protection Division recommends the length of RS 232 cable be less than 10 feet (3 meters) and that the cable be shielded. Internal to a typical device, the RS 232 transceivers are referenced to the electronic components internal ground. Any electrical interference could be coupled through the chip set and fed back to the device. Typical isolation ratings of a non-isolated port could be as low as 1 volt. Such a port could allow electrical feedback of noise to the electronics for any signal interference over 1 volt. Coms 0 through 2 on DPU/TPU/GPU units are non–isolated. However an RS 232 implementation on Com 3 uses opto-isolation technology which increases electrical isolation from the port to the devices internal circuitry to 2.3 kV. It is highly desirable to utilize this port in connection to devices in longer cable runs and dedicated communication networks.

Point to Point Cable

A cable diagram is illustrated in Figure B4. Figure B4 shows the direction of communication signal transmission and the gender of the connectors used in constructing a communication cable.

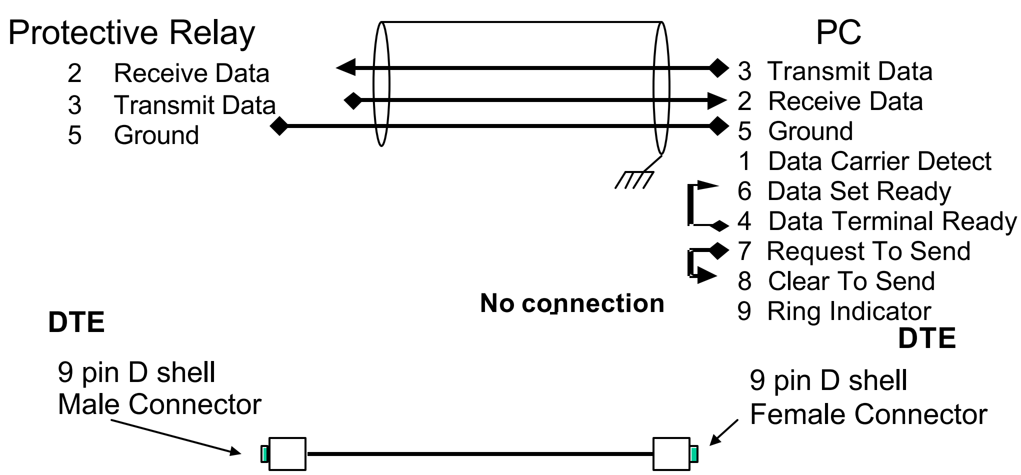


Figure B4 - DPU/TPU/GPU 2000(R) to
PC Cable 9 to 9 Pin-out

An RS 232 interface was designed to simplify the interconnection of devices. Definition of terms may demystify issues concerning RS 232 interconnection. Two types of RS 232 devices are available, DTE and DCE. DTE stands for Data Terminal Equipment whereas DCE stands for Data Communication Equipment. These definitions categorize whether the device originates/receives the data (DTE) or electrically modifies and transfers data from location to location (DCE) Personal Computers are generally DTE devices while line drivers/modems/converters are DCE devices. DPU/TPU/GPU devices have RS 232 DTE implementation. Generally, with a few exceptions, a "straight through cable" (a cable with each pin being passed through the cable without jumping or modification) will allow a DTE device to communicate to a DCE device.

Connection of a PC to a DPU/TPU/GPU requires cable modification since the interconnected devices are both DTE. The same cabling would be utilized if one would connect two DCE devices. The classifications of DTE/DCE devices allow the implementers to determine which device generates the signal and which device receives the signal. Studying Figure B4, pins 2 and 3 are data signals, pin 5 is ground whereas pins 1, 6, 7, 8, 9 are control signals. The arrows illustrate signal direction in a DTE device. The family of DPU/TPU/GPU protective devices do not incorporate hardware or software 'handshaking'.

Handshaking is the ability of the device to control the flow of data between devices. There are two types of 'handshaking', hardware and software. Hardware handshaking involves the manipulation of the RTS (Request to Send) and CTS (Clear to Send) card control signal lines allowing data communication direction and data flow rates to be controlled by the DTE device. Also the flow is controlled by the DTR (Data Terminal Ready) signal which allows the DCE operation.

Software handshaking involves the data flow control by sending specific characters in the data streams. To enable transmission, the XON character is transmitted. To disable reception of data, the transmitting device sends an XOFF character. If the XOFF character is imbedded within the data stream as information, the receiving node automatically turns off. This is the main weakness of software handshaking; inadvertent operation due to control characters being imbedded within data streams. Software handshaking is usually used in printer control. The DPU/TPU/GPU devices do not incorporate handshaking; therefore, the control lines may be ignored as illustrated in Figure B4. However, some PC software utilizes handshaking, thus the port on the personal computer may require a special hardware configuration of the cable to the port. Consult with the software vendor to determine RS 232 control and buffering requirements and the need for signal jumpers required in RS 232 cabling.

The ports on the DPU/TPU/GPU have been tested for operation up to a speed of 19,200 baud. 19,200 baud is the typical data rate applicable for the operation of an asynchronous communication connection over RS 232 without the use of additional timing lines.

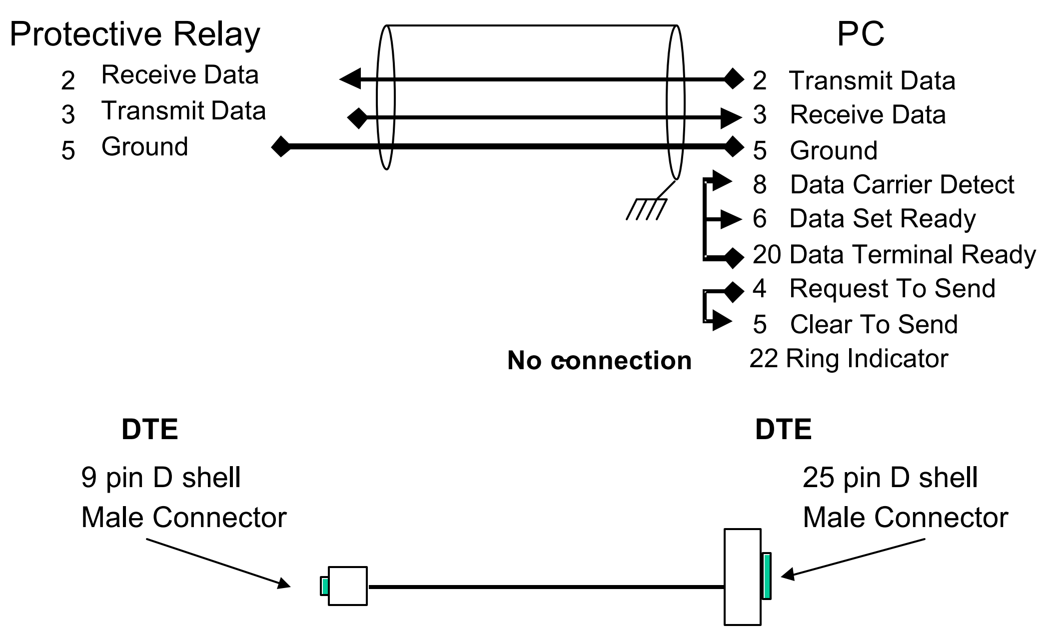


Figure B5 - DPU/TPU/GPU 2000(R) to PC Cable 9 to 25 Pin-out

Some installations require connection to a traditional 25-pin RS 232 interface. The pin designations differ from the 9-pin implementation. Figure B5 illustrates the connection cable for implementation of a 9 to 25 pin cable.

Conclusion

RS 232 communication through a copper connection in a point to point configuration is easily achieved with the correct cable connection. Understanding common terms as DTE, DCE and handshaking are important in designing the correct cable to connect similar devices on an RS 232 physical interface.

APPENDIX C

Connectivity of the MSOC with a DPU/TPU/GPU 2000(R)

MSOC Communication Commonality with DPU/TPU/GPU 2000(R) Protective Relays

Introduction

There seems to be some confusion with ABB's implementation of Modbus on the 2000R and MSOC product lines. This Application Note is intended to dispel the ambiguities. The DPU/TPU/GPU 2000R and MSOC devices allow connectivity as a slave device (receiving commands and responding to a host request) on the Modbus Network. The DPU/TPU/GPU protective devices allow for both emulations of Modbus, namely RTU and ASCII. The MSOC allows for one single emulation of Modbus, namely ASCII. The DPU 2000R and the MSOC can communicate on the same network if both devices are configured as Modbus ASCII slave nodes. This application note illustrates the step by step method of configuring the MSOC and DPU/TPU/GPU 2000R units attached to the same network.

RS 485 Device Connectivity

An RS 232 physical interface offers point to point connectivity. If multiple devices must be interconnected (multi-drop architecture), the physical interface must be changed from RS 232 to a physical interface configuration which allows for a multi-drop architecture.

Both the MSOC and DPU/TPU/GPU 2000R products have isolated RS 485 ports (MSOC through terminals 8,9, and 10. The DPU/TPU/GPU products through the AUX com port as a standard via terminals 55,56 and 57). Communications in a multi-dropped architecture can occur via an RS 485 network with DPU/TPU/GPU 2000R's and MSOC's attached within the wiring constraints of the RS 485 ANSI standards. Figure C1 illustrates the RS 485 connectivity to a host device connected via an RS 485 network. Figure C2 illustrates the RS 232 cable connecting the PC with the RS 485 physical interface converter.

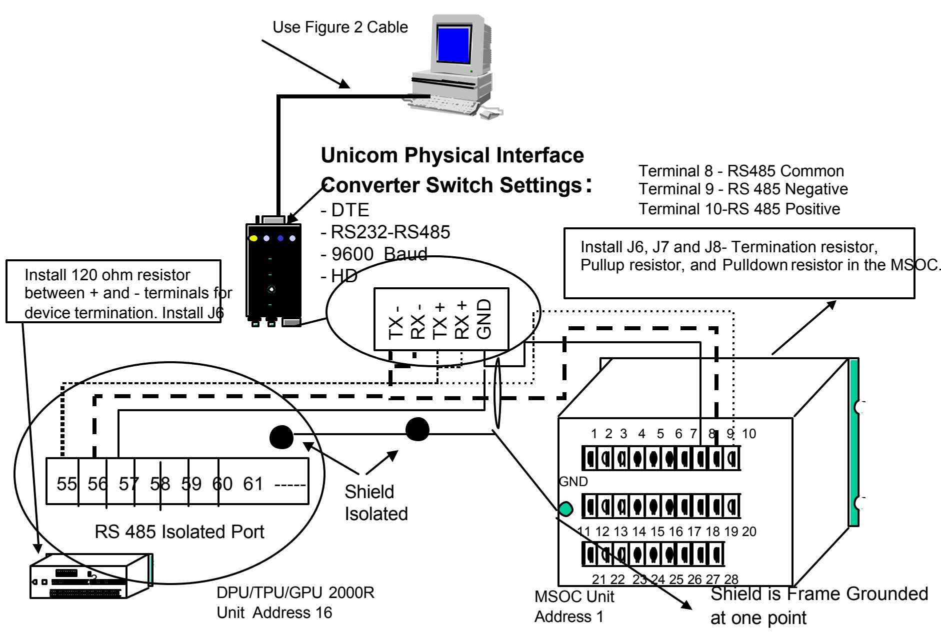


Figure C1: RS 485 Architecture for example system

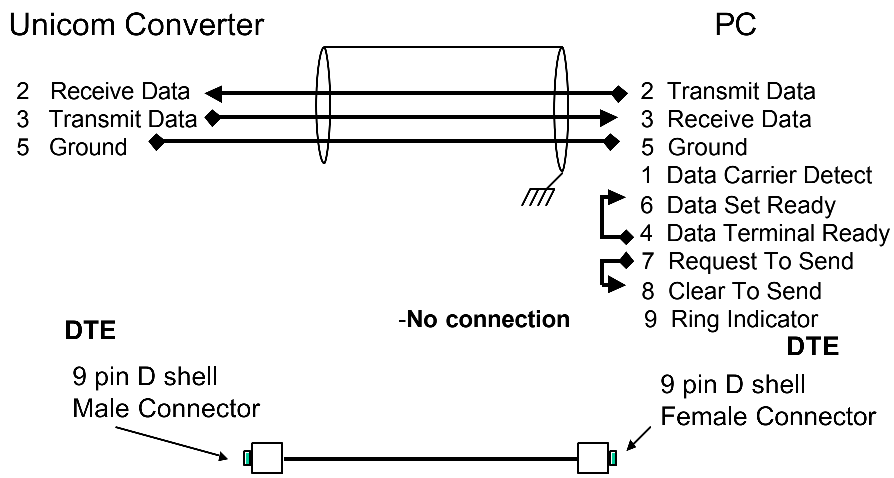


Figure C2 - Unicom RS 485 Converter to PC Cable
9 to 9 Pin-out

Device Configuration Via The Front Panel Interface

To further allow communications between the devices, a common port set-up must be configured, namely, baud rate, stop bits, data bits, and parity. The MSOC and the DPU/TPU/GPU 2000R share the following configurations:

Shared baud rates are:

- 1200
- 2400
- 4800
- 9600
- 19200

Shared recommended frame configurations are:

- Odd parity, 7 Data Bits, One stop Bit
- Even parity, 7 Data Bits, One stop Bit

With each device on the network configured in one shared and common configuration (as referenced above), communications between a Host device and MSOC's and DPU/TPU/GPU 2000R's shall occur. The MSOC and DPU/TPU/GPU product families are designed to communicate on a common Modbus network. This example shall illustrate a network configuration with two devices attached communicating at 9600 baud 7 data bits, one stop bit, and odd parity. This is only one of several combinations, which may be used in a communication configuration.

MSOC Configuration of Modbus ASCII

The MSOC allows communication through the rear port; however, the front port (RS 232) must be disabled. To disable the front port the following procedure must be followed:

To direct the communications from the front RS 232 port to the rear RS 485 port the following procedure must be followed in addition to the aforementioned procedure for the Native ASCII and the Modbus ASCII protocols.

A. Depress E.
B. Depress the down arrow 4 times to display the OPERATIONS MENU.
C. Depress E.
D. Depress the down arrow 7 times to display the selection ENABLE FP COM.

E. Depress E.
F. Enter the Password as described in the previous illustrations.
G. The user shall be prompted, with the selection ENABLE FP < >. If the selection is YES, then the Front Port (RS 232) shall have the communications options configured via the above process. If the Selection is NO then the Rear Port (RS 485) shall be enabled with the protocol configured via the above process. Enter the correct password for the unit. Then Depress E to accept the password. Select "NO" and the RS 485 port shall be enabled for communication.

The MSOC process for Baud Rate and Frame Size configuration is as follows:

A. Depress E.
B. Main Menu heading should be visible.
C. Depress E.
D. The Operations menu will be visible. Depress the C pushbutton to go to the Main Menu.
E. Depress the up arrow pushbutton twice to attain the Settings menu selection.
F. Depress E. The Settings menu shall be visible.
G. Depress the down arrow once. The Change Settings will be visible.
H. Depress E.
I. Depress the down arrow 5 times. The Com selection shall be visible.
J. Depress the E pushbutton. The password selection shall be visible. Enter the correct password for the unit. Depress E.
K. Select the appropriate address for the unit. For this instance it shall be 1. To change–Depress E, right arrow to the correct value, depress E and then depress C.
L. Depress the down arrow once. FP Baud rate shall be visible. Select 9600. To change–Depress E, right arrow to the correct value, depress E and then depress C.
M. Depress the down arrow once. FP Frm shall be visible. Select O,7,1. This sets the port to 7 data bits, Odd Parity, and one stop bit. To change–Depress E, right arrow to the correct value, depress E and then depress C.
N. Depress the down arrow once. FP Prot shall be visible. Select Modbus. This enables the Modbus protocol resident in the unit. To change–Depress E, right arrow to the correct value, depress E and then depress C.
O. Depress the down arrow once. RP Baud rate shall be visible. Select 9600. To change–Depress E, right arrow to the correct value, depress E and then depress C.
P. Depress the down arrow once. RP Frm shall be visible. Select O,7,1. This sets the port to 7 data bits, Odd Parity, and one stop bit. To change–Depress E, right arrow to the correct value, depress E and then depress C.
Q. Depress the down arrow once. RP Prot shall be visible. Select Modbus. This enables the Modbus protocol resident in the unit. To change–Depress E, right arrow to the correct value, depress E and then depress C
R. Depress the down arrow once. Local Echo shall be visible. Select OFF. These selections shall place the unit in the native Modbus protocol mode. To change–Depress E, right arrow to the correct value, depress E and then depress C.
S. After each of the settings, perform a unit save to maintain the unit settings entered in this session.

The above process has configured the front and the rear port for Modbus communication as 9600 baud, 7 data bits, one stop bit, odd parity. As per the previous section, the Rear Port has been enabled.

DPU/TPU/GPU 2000R Configuration of Modbus ASCII

The DPU/TPU/GPU 2000R allows for Modbus communication over Com 3 or the AUX COM port. If the Type 2 Option board is within the unit, Com 3 contains the RS 232 Physical interface and the AUXCOM board communicates via RS 485. If the Type 8 Option board is contained within the unit, both Com 3 and the AUX COM port allow for RS 485 Communications. The following configuration shall enable Com 3 and AUX COM for

Modbus ASCII communications. It should be noted that the configuration sequence is outlined for units having a Front Panel interface. Similar communication configuration steps could be followed when using ECP software. The communication procedure is the same for DPU/TPU/GPU 2000R units.
With the front panel showing metering values:

A. Depress E to display the configuration menu.
B. Depress the down arrow to display the Settings Selection Depress E.
C. Depress the down arrow to display the Change Setting. Depress E.
D. Depress the down arrow seven times to display the Communication Display. Depress E.
E. The query Enter Password shall be displayed. Enter the correct password for the unit and depress E.
F. The first selection in the Communication section shall be unit address. This address is to be entered in HEX. This example shall address the unit as Decimal 16. Enter the HEX value 10 for this unit. Right arrow (→) and left arrow (←) keys select the digit to change. The up arrow and down arrow keys increment and decrement the digit. Depress C to exit.
G. Depress the down arrow three times to select the RP RS 232 Baud menu item. Depress E.
H. A menu item for RP RS 232 baud rate selection shall appear. The present baud rate NOW < > and NEW < > is displayed. Depress the Right Arrow (→) or Left Arrow (←) to select the baud rate of 9600 baud. Depress C to exit the menu.
I. Depress the down arrow once and select the RP RS 232 frame size menu item. Depress E.
J. The Right Arrow (→) and Left Arrow (←) buttons must be depressed to select the frame size: <Odd, 7,1>, which is odd parity, 7 data bits, one stop bit. Depress C.
K. Depress the down arrow to select RP 485 Baud. Depress E to display selection.
L. A menu item for RP RS 485 baud rate selection shall appear. The present baud rate NOW < > and NEW < > is displayed. Depress the Right Arrow (→) or Left Arrow (←) to select the baud rate of 9600 baud. Depress C to exit the menu.
M. Depress the down arrow to display the RP 485 Frame menu selection. Depress E.
N. The Right Arrow (→) and Left Arrow (←) buttons must be depressed to select the frame size: <Odd, 7,1>, which is odd parity, 7 data bits, one stop bit. Depress C.
O. Depress the down arrow twice to display the Parameter 1 option. Depress E to enter the selection menu.
P. Enter the number 3 to enable the both ports to communicate in Modbus. Depressing the Left Arrow (←) or Right Arrow (→) decrements or increments the selection. Depress C to enter.
Q. Depress the down arrow ten times to display the Mode Parameter 1 selection. Depress E to enter the submenu.
R. This selection must be enabled to allow Modbus ASCII communication through Com 3. Depress the Left Arrow (←) or Right Arrow (→) to select the Enable Option. Depress C to exit the menu.
S. Depress the down arrow once to select the Mode 2 Parameter selection. This selection must be enabled to allow Modbus ASCII communications through the AUX COM Port. Depress E.
T. Enable the Mode 2 Parameter by depressing the Left Arrow (←) or Right Arrow (→) to select the Enable parameter. Depress C to exit the menu.
U. Depress C to save the changes to the communication port changes made.
V. The user shall be prompted with the question "Enter <YES> to save settings?" Depress the right arrow button (→) and depress E. Which will save the unit communication settings.
W. Depress the black system reset pushbutton which is recessed and located at the lower left-hand side of the unit.

The DPU/TPU/GPU unit is now configured at 9600 baud, odd parity, 7 data bits, 1 stop bit for Modbus ASCII communications through Com 3 or the AUX COM port.

Network Communication Test

A host PC shall be connected to the RS 485-converter unit as illustrated in Figure 2-5 of this document. This section shall further illustrate the method of test for the unit connection verification. The units to be polled are Unit 1 (MSOC) and Unit 16 (DPU/TPU/GPU 2000R).

The easiest method to initiate communications in the Modbus protocol is to read known discrete and register data. As per the MSOC Modbus Register documentation, the unit catalog number is resident at Register 40132. Appendix A of this document lists the register definitions of the MSOC. A Read Holding Register Modbus Command is explained. Appendix A of this document includes the Modbus Register Definition for the MSOC. Additionally, documentation is available from Groupe Schneider further describing the Modbus ASCII emulation characteristics. The explanation contained within this document is intended to be a quick start guide to communication initiation.

The length of the catalog number is 12 characters or 6 registers. The following command string format, when sent will retrieve the catalog number from the unit.

: 01 03 00 83 00 06 73 lf cr

The above string in Modbus ASCII format should be sent:

3A 30 31 30 33 30 30 38 33 30 30 30 36 37 33 0D 0A

The string is translated as such:

Colon (in HEX), unit address = 01 (in HEX), Read Holding Registers (Code 3 in HEX), data memory address –1 = 131 decimal (0083 in HEX), number of registers read = 6 (0006 in HEX), message calculated LRC code 73 (37 33), and line feed (0D) and (0A).

A typical response shall include the following:

: Address number (01), Read Holding Registers command (Code 3 in HEX), Byte Count Returned 12 in decimal (OC in HEX), Data Register 40132 = 47, Data Register 40133= 4M, Data Register 40134 = 14 Data Register 40135 = 11, Data Register 40136 = 60 Data Register 40137 = 10, and calculated LRC =77 (HEX) and line feed with carriage return (0D 0A).

The aforementioned response would be returned as such:

3A 30 31 30 33 30 43 34 37 34 4D 31 34 31 31 36 30 31 30 37 37 0A 0D.

As per the DPU/TPU/GPU 2000(R) Modbus Register documentation (each protective relay has a different register map, however, in this example the catalog number address is common between all units), the unit catalog number is resident at Register 40133.

The length of the catalog number is 20 characters or 10 registers. The following command string format, when sent will retrieve the catalog number from the unit.

: 10 03 00 84 00 0A 73 lf cr

The above string in Modbus ASCII format should be sent:

3A 31 30 30 33 30 30 38 34 30 30 30 41 37 33 0D 0A

The string is translated as such:

Colon (in HEX), unit address = 01 (in HEX), Read Holding Registers (Code 3 in HEX), data memory address –1 = 132 decimal (0084 in HEX), number of registers read = 10 (000A in HEX), message calculated LRC code 73 (37 33), and line feed (0D) and (0A).
A typical response shall include the following:

: Address number (10 in HEX 16 in decimal), Read Holding Registers command (Code 3 in HEX), Byte Count Returned 12 in decimal (OC in HEX), Data Register 40132 = 47, Data Register 40133= 4M, Data Register 40134 = 14 Data Register 40135 = 11, Data Register 40136 = 60 Data Register 40137 = 10, and calculated LRC =77 (HEX) and line feed with carriage return (0D 0A).

The aforementioned response would be returned as such:

3A 31 30 30 33 30 43 34 37 34 4D 31 34 31 31 36 30 31 30 37 37 0A 0D.

The Unicom converter used for this test has 3 LED's which show the communication state of the Transmit and Receive RS 232 lines. Visual indication allows for determination of successful communication transmission and reception independently of the PC program monitoring.

A variety of commercially available packages are capable of sending Modbus ASCII commands. The intent of this example is not to review or recommend a specific vendor's Modbus software platform for sending commands, but to allow analysis of data received and viewed via a communication analyzer.

Conclusion

Connectivity of the MSOC and DPU/TPU/GPU product families is possible with the knowledge of RS 485 device connectivity, memory map register addressing, Modbus ASCII protocol and the host package sending the command strings. Following a set procedure as outlined in this presentation shall enable the reader to easily and effectively initiate communications between host and relay devices.

APPENDIX D

MSOC Native ASCII Revision History

This section details changes made to the Native MSOC ASCII Protocol for the different Release Versions.

Additions for Release 1.50 and 1.60:

Add (r) commands for Recloser
Add (b) commands for Breaker Counter
Add the TxDelayTimer to the COMM Block
Add Logical Inputs used by Recloser
Add Logical Outputs used by Recloser
Year 2000 Compliance for the Real Time Clock and Records Storage

Modifications to Release 1.69 Only:

Modify the curves and parameters to IEC standards
Keep nomenclature in IEEE.ANSI format for each parameter
Relay for Export Only

Additions for Release 1.70:

Add Seal-Ins (si) to the Read Command Types
Add Seal-Ins (so) to the Read/Write for Configuration of these Outputs
Add RSI to the Logical Inputs
Modify Event Records, Remove EEP xxxx Restore, add EEP Seal-In Failure
Add Options for IEC curves for existing Protective Functions

INDEX

This page intentionally left blank