Modbus Plus Interface Module

(M/N 57C441)

Industrial

CONTROLS

Instruction Manual J2-3051-1

RELIANCE ELECTRIC

The information in this user's manual is subject to change without notice.

DANGER

ONLY QUALIFIED ELECTRICAL PERSONNEL FAMILIAR WITH THE CONSTRUCTION AND OPERATION OF THIS EQUIPMENT AND THE HAZARDS INVOLVED SHOULD INSTALL, ADJUST, OPERATE, OR SERVICE THIS EQUIPMENT. READ AND UNDERSTAND THIS MANUAL AND OTHER APPLICABLE MANUALS IN THEIR ENTIRETY BEFORE PROCEEDING. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN SEVERE BODILY INJURY OR LOSS OF LIFE.

DANGER

THE USER IS RESPONSIBLE FOR CONFORMING WITH ALL APPLICABLE LOCAL, NATIONAL, AND INTERNATIONAL CODES. WIRING PRACTICES, GROUNDING, DISCONNECTS, AND OVER-CURRENT PROTECTION ARE OF PARTICULAR IMPORTANCE. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN SEVERE BODILY INJURY OR LOSS OF LIFE.

WARNING

THE USER MUST PROVIDE AN EXTERNAL, HARDWIRED EMERGENCY STOP CIRCUIT OUTSIDE THE CONTROLLER CIRCUITRY. THIS CIRCUIT MUST DISABLE THE SYSTEM IN CASE OF IMPROPER OPERATION. UNCONTROLLED MACHINE OPERATION MAY RESULT IF THIS PROCEDURE IS NOT FOLLOWED. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY.

WARNING

INSERTING OR REMOVING A MODULE MAY RESULT IN UNEXPECTED MACHINE MOTION. POWER TO THE MACHINE SHOULD BE TURNED OFF BEFORE INSERTING OR REMOVING THE MODULE. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

CAUTION

THIS MODULE CONTAINS STATIC-SENSITIVE COMPONENTS. CARELESS HANDLING CAN CAUSE SEVERE DAMAGE. DO NOT TOUCH THE CONNECTORS ON THE BACK OF THE MODULE. WHEN NOT IN USE, THE MODULE SHOULD BE STORED IN AN ANTI-STATIC BAG. THE PLASTIC COVER SHOULD NOT BE REMOVED. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN DAMAGE TO OR DESTRUCTION OF THE EQUIPMENT.

Table of Contents

Appendices

List of Figures

1.0 INTRODUCTION

The products described in this manual are manufactured or distributed by Reliance Electric Industrial Company.

The AutoMax Modbus Plus™ Interface module (M/N 57C441) provides a direct connection for the AutoMax® system to the Modicon™ Modbus Plus network. The Modbus Plus Interface module can be placed in any slot in an AutoMax rack that contains at least one AutoMax Processor, and can communicate with up to four AutoMax Processors in that rack. The AutoMax rack containing the Modbus Plus Interface module appears as a single station on the Modbus Plus network. Figure 1.1 illustrates an AutoMax system connected to a Modbus Plus network.

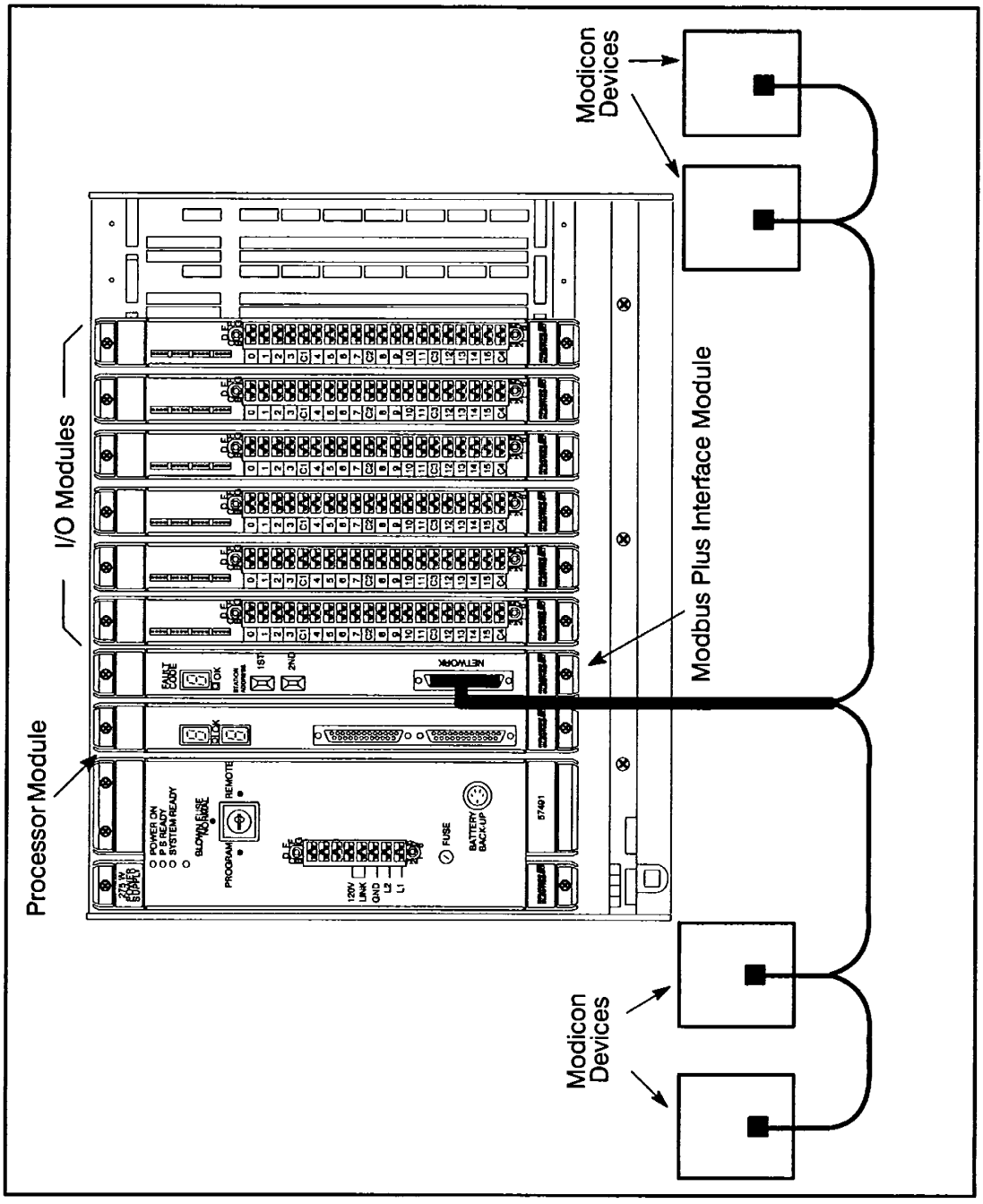


Figure 1.1 - Connecting the AutoMax System to the Modbus Plus Network

The remainder of this manual describes the functions and specifications of the AutoMax Modbus Plus Interface module. It also includes a detailed description of module installation and troubleshooting procedures, as well as programming methods.

1.1 Additional Information

You must be familiar with all the instruction manuals that describe your system configuration. This may include, but is not limited to, the following:

- J-3616 KERMIT™ COMMUNICATIONS SOFTWARE INSTRUCTION MANUAL
- J-3618 NORTON™ EDITOR INSTRUCTION MANUAL
- J-3636 COMMON MEMORY MODULE INSTRUCTION MANUAL
- J-3649 AutoMax CONFIGURATION TASK INSTRUCTION MANUAL
- J-3650 AutoMax PROCESSOR MODULE INSTRUCTION MANUAL
- J-3669 AutoMax POCKET REFERENCE
- J-3675 AutoMax ENHANCED BASIC LANGUAGE NSTRUCTION MANUAL
- J-3676 AutoMax CONTROL BLOCK LANGUAGE INSTRUCTION MANUAL
- J-3677 AutoMax LADDER LOGIC LANGUAGE INSTRUCTION MANUAL
- J-3684 ReSource™ AutoMax PROGRAMMING EXECUTIVE INSTRUCTION MANUAL VERSION 2.0
- J-3750 ReSource AutoMax PROGRAMMING EXECUTIVE INSTRUCTION MANUAL VERSION 3.0
- IEEE 518 GUIDE FOR THE INSTALLATION OF ELECTRICAL EQUIPMENT TO MINIMIZE ELECTRICAL NOISE INPUTS TO CONTROLLERS
- Your personal computer and DOS operating system manuals
- GM-MBPL-001 rev C Modicon Modbus Plus Network Planning and Installation Guide
- Other instruction manuals applicable to your hardware configuration

1.2 Related Hardware and Software

M/N 57C441 contains one Modbus Plus Interface module. The module can be used with the following hardware and software. See Appendix F for network cabling information.

1. M/N 57C430A, 57C431, 57C435 Automax Processor
2. IBM™-AT compatible computer running DOS version 3.1 or later
3. M/N 61C127 RS-232C ReSource Interface cable. This cable is used to connect the personal computer to the AutoMax Processor module.
4. M/N 57C413 or 57C423 Common Memory module. This module is used when there is more than one AutoMax Processor in a rack.
5. (various model numbers) ReSource AutoMax Programming Executive software

2.0 MECHANICAL/ELECTRICAL DESCRIPTION

This section describes the mechanical and electrical characteristics of the AutoMax Modbus Plus Interface module.

2.1 Mechanical Description

The Modbus Plus Interface module is a printed circuit assembly that plugs into the backplane of an AutoMax rack. The module consists of a printed circuit board, faceplate, and protective enclosure. The faceplate contains ejector tabs at the top and bottom to simplify removing the module from the rack. See figure 2.1 for an illustration of the module faceplate.

The 9-pin D-shell connector on the faceplate is used to make the connection to the Modbus Plus network by means of a standard Modbus Plus cable. Refer to Appendix F or to Modicon documentation for the cable connections.

The two thumbwheel switches on the faceplate are used to set the module's station address (01-64) on the Modbus Plus network. Note that the station address is also called the "node number" in Modicon Modbus Plus literature. The upper thumbwheel switch represents the most significant digit; the lower thumbwheel switch represents the least significant digit.

For diagnostic purposes, the faceplate contains a seven-segment LED which displays error codes. The error codes are defined in Appendix C. A green status LED (labeled "OK") on the faceplate indicates when the module is operational (ON) or when it should be replaced (OFF).

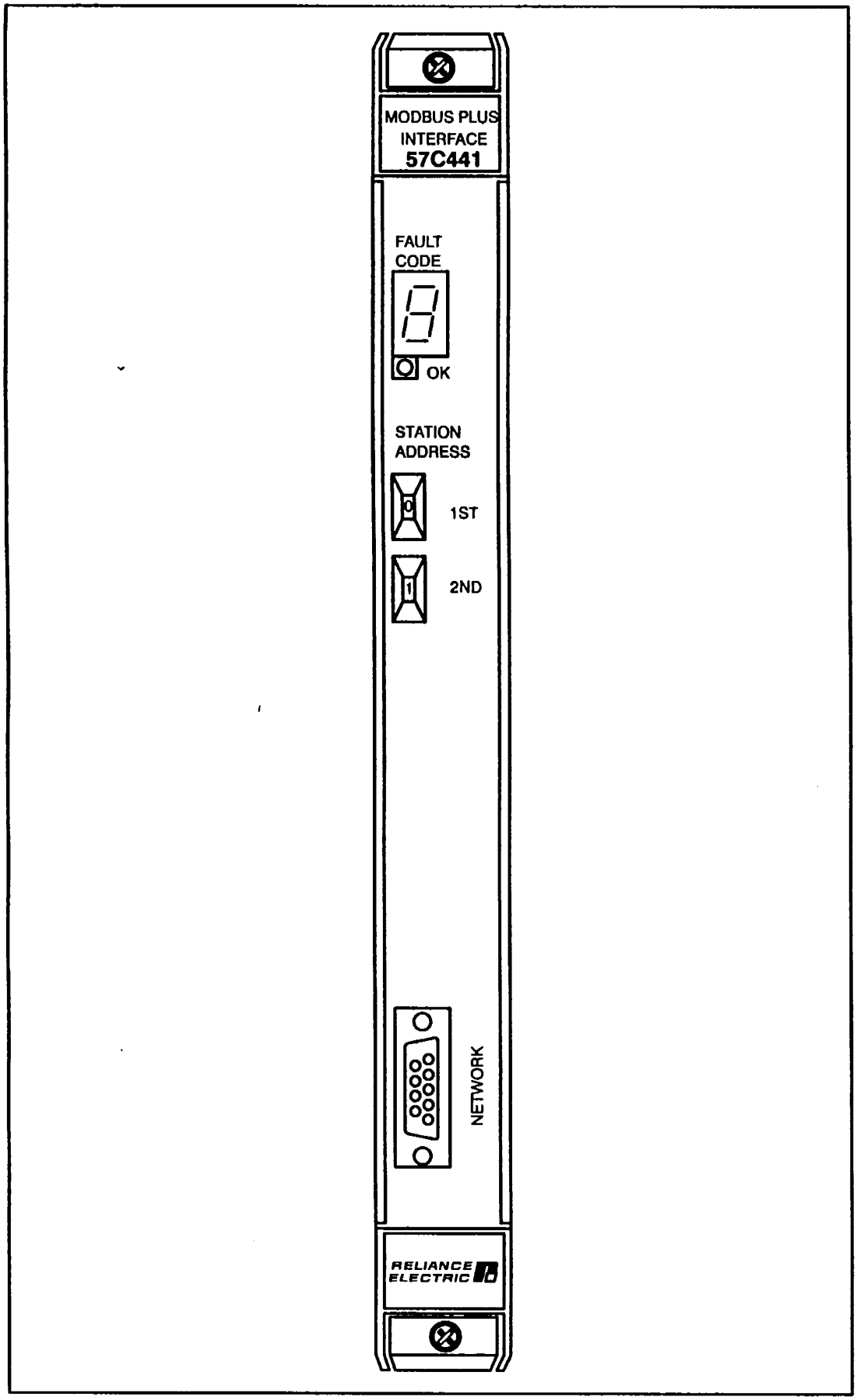


Figure 2.1 - Modbus Plus Interface Module Faceplate

2.2 Electrical Description

The Modbus Plus Interface module contains an 8-MHz microprocessor that shares 32K of dual port memory with the AutoMax rack backplane. This memory is accessible to both the on-board microprocessor and AutoMax Processors in the rack. A block diagram is shown in Appendix B. The module contains Modicon Modbus Plus circuitry which performs the actual communications tasks on the Modbus Plus network. The module contains a watchdog timer which is enabled when power is turned on to the module. The microprocessor must reset the watchdog timer within a specified time or the microprocessor will shut down and the status LED labeled "OK" on the faceplate will turn off.

At power-up, the on-board microprocessor runs diagnostics on the microprocessor, EPROM, RAM and dual port memory, as well as performing system-level diagnostics. As each test is run, a number is written out to the seven-segment display. If there is a fault during these tests, the microprocessor halts, the watchdog times out, and the seven-segment LED displays the code of the failed diagnostic. See Appendix C for a list of error codes.

3.0 INSTALLATION

This section provides instructions on how to install the Modbus Plus Interface module.

3.1 Wiring

DANGER

THE USER IS RESPONSIBLE FOR CONFORMING WITH ALL APPLICABLE LOCAL, NATIONAL, AND INTERNATIONAL CODES. WIRING PRACTICES, GROUNDING, DISCONNECTS, AND OVER-CURRENT PROTECTION ARE OF PARTICULAR IMPORTANCE. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN SEVERE BODILY INJURY OR LOSS OF LIFE.

To reduce the possibility of electrical noise interfering with the operation of the control system, exercise care when installing the wiring from the control system to the external devices. For detailed recommendations refer to IEEE 518.

3.2 Initial Installation

Use the following procedure to install the module:

Step 1. Stop any application tasks that may be running.

DANGER

THIS EQUIPMENT IS AT LINE VOLTAGE WHEN A-C POWER IS CONNECTED. DISCONNECT AND LOCK OUT ALL UNGROUNDED CONDUCTORS OF THE A-C POWER LINE. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN SEVERE BODILY INJURY OR LOSS OF LIFE.

WARNING

INSERTING OR REMOVING A MODULE MAY RESULT IN UNEXPECTED MACHINE MOTION. POWER TO THE MACHINE SHOULD BE TURNED OFF BEFORE INSERTING OR REMOVING THE MODULE. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

Step 2. Turn off power to the rack. All power to the rack as well as all power leading to the rack should be off.

CAUTION

THIS MODULE CONTAINS STATIC-SENSITIVE COMPONENTS. CARELESS HANDLING CAN CAUSE SEVERE DAMAGE. DO NOT TOUCH THE CONNECTORS ON THE BACK OF THE MODULE. WHEN NOT IN USE, THE MODULE SHOULD BE STORED IN AN ANTI-STATIC BAG. THE PLASTIC COVER SHOULD NOT BE REMOVED. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN DAMAGE TO OR DESTRUCTION OF THE EQUIPMENT.

Step 3. Take the module out of its shipping container. Take it out of the anti-static bag, being careful not to touch the connectors on the back of the module.

Step 4. Insert the module in the desired slot in the rack, making sure it is well-seated in the rack. The module may reside in any slot in the rack. Use a screwdriver to secure the module in the rack.

Step 5. Set the station address (node number) of the Modbus Plus Interface module (in decimal) using the two thumbwheel switches on the module faceplate. The upper switch is the most significant digit and the lower switch is the least significant digit. The station address is read from the thumbwheel switches only at power-up. Therefore, make sure each device on the network has a unique station address (01-64) before power is applied. If more than one device on the network is given the same station address, the module will not be able to communicate and an error will be displayed. Node number 00 is an invalid number.

Step 6. Connect the Modbus Plus cable to the 9-pin connector. Refer to Appendix F or to the Modicon Modbus Plus Network Planning and Information Guide for information on cabling and termination.

Step 7. Turn on power to the rack. An internal diagnostic routine is automatically executed by the module. If an error is detected, an error code is displayed on the seven-segment LED. If the green status LED is OFF and no seven-segment error code is displayed, a local watchdog failure has occurred. If a diagnostic fault code other than ".5" is displayed, the Modbus Plus Interface module must be replaced.

If the thumbwheel switches are set to an invalid device number, fault code ".5" will be displayed on the seven-segment LED on the module faceplate after power-up. To clear the invalid device number fault code, refer to section 5.2.

Step 8. Verify the installation by monitoring registers on the module. Refer to Appendix G. After power-up diagnostics are complete, the green status LED will go on.

3.3 Module Replacement

Use the following procedure to replace the Modbus Plus Interface module:

Step 1. Stop any application tasks that may be running.

DANGER

THIS EQUIPMENT IS AT LINE VOLTAGE WHEN A-C POWER IS CONNECTED. DISCONNECT AND LOCK OUT ALL UNGROUNDED CONDUCTORS OF THE A-C POWER LINE. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN SEVERE BODILY INJURY OR LOSS OF LIFE.

WARNING

INSERTING OR REMOVING A MODULE MAY RESULT IN UNEXPECTED MACHINE MOTION. POWER TO THE MACHINE SHOULD BE TURNED OFF BEFORE INSERTING OR REMOVING THE MODULE. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

Step 2. Turn off power to the rack. All power to the rack as well as all power leading to the rack should be off.

Step 3. Disconnect the Modbus Plus cable from the module faceplate.

Step 4. Use a screwdriver to loosen the screws that hold the module in the rack. Take the module out of the slot in the rack.

CAUTION

THIS MODULE CONTAINS STATIC-SENSITIVE COMPONENTS. CARELESS HANDLING CAN CAUSE SEVERE DAMAGE. DO NOT TOUCH THE CONNECTORS ON THE BACK OF THE MODULE. WHEN NOT IN USE, THE MODULE SHOULD BE STORED IN AN ANTI-STATIC BAG. THE PLASTIC COVER SHOULD NOT BE REMOVED. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN DAMAGE TO OR DESTRUCTION OF THE EQUIPMENT.

Step 5. Take the new module out of its shipping container. Take it out of the anti-static bag, being careful not to touch the connectors on the back of the module.

Step 6. Insert the module in the rack, making sure that it is well-seated in the rack.

Step 7. Set the device number of the new module to the same number as the old module by using the two thumbwheel switches on the module faceplate. The upper switch is the most significant digit and the lower switch is the least significant digit.

Step 8. Attach the Modbus Plus cable to the module faceplate.

Step 9. Turn on power to the rack.

Step 10. Verify the installation by monitoring registers on the module. Refer to Appendix G. After power-up diagnostics are complete, the green status LED will go on.

4.0 PROGRAMMING

The Modbus Plus network is a token-passing network, which means that any device on the network can initiate communication with any other device. Each node can also pass up to 32 words of global data with the token. This global data is available to all other nodes on the network.

Up to 32 devices can be connected directly to the network, with station addresses ranging from 1 to 64. Duplicate addresses are not allowed. Repeater devices can be used to extend the cable distance and node count. Modicon bridge devices can be used to connect Modbus Plus networks and to connect Modbus devices. The Modbus Plus network data rate is 1 million bits per second.

This section describes how the data is organized in the module and provides examples of how the module is accessed by the application software. For more detailed information on programming, refer to the AutoMax Programming Reference Binder (J-3686 or J2-3049).

4.1 Register Organization

The Modbus Plus Interface module contains dual-port memory that can be accessed through the AutoMax rack backplane by application tasks running on the AutoMax Processor as well as by the microprocessor on the module. The memory organization is as shown in the following diagram. The following sections describe the different register areas in more detail.

Registers	Description
0 - 3	Interrupt control registers
4 - 63	Status and control registers
64 - 319	Discrete output registers
320 - 575	Discrete input registers
576 - 1599	Input registers
1600 - 3583	Holding registers
3584 - 4095	Message buffer registers
4096 - 6143	Global data registers
6144 - 6197	Network statistics
6198 - 7167	Reserved for future use

Figure 4.1 - Dual Port Memory Map

4.1.1 Interrupt Control Registers

Registers 0-3 are reserved for system use only. They must not be used by AutoMax application tasks.

4.1.2 Status and Control Registers

The status and control registers are described in figure 4.2. Registers 4-31 are read-write registers from the host (AutoMax Processor); registers 32-63 are read-only.

R/W	Reg #	Description
R/W	4	Not used.
R/W	5	Bit 0 is used to enable the module. The default is 1 (ON). Note that even if bit 0 is not set, the module still passes the token and replies to diagnostic messages from remote nodes; however, no messages can be sent and ordinary data messages from remote nodes are not replied to until bit 0 is set. This bit is set on a hardware or software reset. Bits 1-15 are not used.
R/W	6	Message response timeout in 100 ms. increments. The default is 100 (10 seconds). The value can range from 1 to 65535. If a reply to a message is not returned within this time period, the module returns an error on its 7-segment LED.
R/W	7	Number of words of global data for this node. Default is 0. The value can range from 0 to 32.
R/W	8	Global data mask register. Low byte nodes 1-8, high byte nodes 9-16.
R/W	9	Global data mask register. Low byte nodes 17-24, high byte nodes 25-32.
R/W	10	Global data mask register. Low byte nodes 33-40, high byte nodes 41-48.
R/W	11	Global data mask register. Low byte nodes 49-56, high byte nodes 57-64.
R/W	12-21	Not used. Reserved for future use.
R/W 22	22	Response granularity. After the GATEWAY_CMD_OK@ function writes the specified command to the Modbus Plus Interface module, it checks for completion and, if not complete, delays for the time specified in this register and then repeats the test for completion. The time used for the delay is the value in this register divided by 8. For example, if this register contains 5, the GATEWAY_CMD_OK@ function tests for completion every 625 msec.
R/W	23-31	Not used. Reserved for future use.
R/O	32	Modbus Plus node address in decimal (1-64). The value is read from the thumbwheel switches on the faceplate at power-up. After power-up, the module will begin passing the token, transmitting and receiving data messages, and responding to diagnostic commands.
R/O	33	Software revision for the software on the module. A value of 100 indicates version 1.00, and so on.
R/O	34-61	Not used. Reserved for future use.
R/O	62-63	Interface module ID (ASCII 'GTWY').

Figure 4.2 - Status and Control Register Assignments

4.1.3 Data Registers

The Modicon data image in the dual port memory appears as a subset of the registers normally supported in Modicon programmable controllers. The equivalent AutoMax and Modicon registers on the module are shown in figure 4.3.

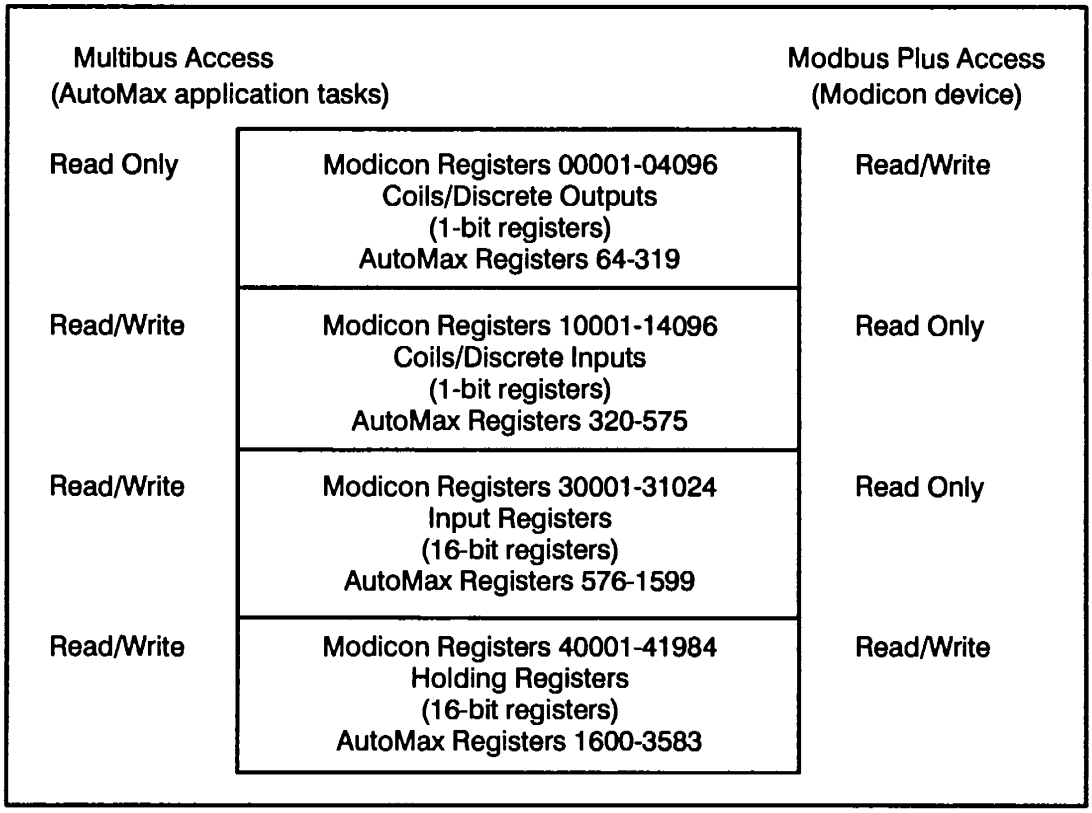


Figure 4.3 - Modicon Data Image

Mapping is handled transparently by the Modbus Plus Interface module software. For example, if another node writes to this node at address 40001, the data is written starting at AutoMax register 1600, and so on.

4.1.4 Message Buffer Registers

The message buffer registers (3584-4095) are used by the module software to send and receive messages. These registers are reserved for system use and must not be used in application tasks.

4.1.5 Modicon Global Data Registers

Registers 4096-6143 contain all of the global data that can exist on a Modbus Plus network. Each node is assigned 32 registers. Particular locations are meaningful only if the corresponding node is writing global data. All registers are read only from the point of view of the AutoMax Processor except for the registers on the Modbus Plus Interface module, which are read-write.

Register	Description	Register	Description
4096-4127	Global data node 1	5120-5151	Global data node 33
4128-4159	Global data node 2	5152-5183	Global data node 34
4160-4191	Global data node 3	5184-5215	Global data node 35
4192-4223	Global data node 4	5216-5247	Global data node 36
4224-4255	Global data node 5	5248-5279	Global data node 37
4256-4287	Global data node 6	5280-5311	Global data node 38
4288-4319	Global data node 7	5312-5343	Global data node 39
4320-4351	Global data node 8	5344-5375	Global data node 40
4352-4383	Global data node 9	5376-5407	Global data node 41
4384-4415	Global data node 10	5408-5439	Global data node 42
4416-4447	Global data node 11	5440-5471	Global data node 43
4448-4479	Global data node 12	5472-5503	Global data node 44
4480-4511	Global data node 13	5504-5535	Global data node 45
4512-4543	Global data node 14	5536-5567	Global data node 46
4544-4575	Global data node 15	5568-5599	Global data node 47
4576-4607	Global data node 16	5600-5631	Global data node 48
4608-4639	Global data node 17	5632-5663	Global data node 49
4640-4671	Global data node 18	5664-5695	Global data node 50
4672-4703	Global data node 19	5696-5727	Global data node 51
4704-4735	Global data node 20	5728-5759	Global data node 52
4736-4767	Global data node 21	5760-5791	Global data node 53
4768-4799	Global data node 22	5792-5823	Global data node 54
4800-4831	Global data node 23	5824-5855	Global data node 55
4832-4863	Global data node 24	5856-5887	Global data node 56
4864-4895	Global data node 25	5888-5919	Global data node 57
4896-4927	Global data node 26	5920-5951	Global data node 58
4928-4959	Global data node 27	5952-5983	Global data node 59
4960-4991	Global data node 28	5984-6015	Global data node 60
4992-5023	Global data node 29	6016-6047	Global data node 61
5024-5055	Global data node 30	6048-6079	Global data node 62
5056-5087	Global data node 31	6080-6111	Global data node 63
5088-5119	Global data node 32	6112-6143	Global data node 64

Figure 4.4 - Global Data Register Assignments

The initial register number in the range assigned for a particular node can be calculated with the following formula:

[32 * (node # - 1)] + 4096

For example, the initial global data register for node 10 is

[32 * (10-1)] + 4096 = 4384

4.1.6 Modbus Plus Network Statistics Registers

The network statistics registers (6144-6197) contain status and error statistics for the Modbus Plus network. Refer to Appendix E for a description of what each register contains. These registers are read only. If you require further information, refer to Modicon Modbus Plus documentation.

Registers 6167-6170 within this register range contain the Active Station list. Each bit corresponds to one node. By reading this table, an application task can determine what other nodes are active on the network.

4.2 AutoMax Application Programming

The sections that follow describe how to configure registers in the AutoMax Modbus Plus Interface module's dual port memory and how to initiate commands from the module. Link configuration and application programming examples are provided.

4.2.1 Variable Configuration

AutoMax application tasks communicate with the Modbus Plus Interface module by referencing registers in the dual port memory. These registers must first be configured using the AutoMax Programming Executive software. In AutoMax Programming Executive Version 3.0 and later, you define these registers using the Variable Configurator. If you are using AutoMax Programming Executive Version 2.1 or earlier, you define these registers using MODDEF and IODEF statements in the rack configuration task. The MODDEF statement is used to define all data registers (64-3583). The IODEF statement is used to define registers 4-63, 4096-6143, and 6144-6197. IODEFs reference the actual dual port memory registers.

The format for the MODDEF statement is as follows:

nnnn MODDEF var_name [SLOT=slot number, &
REGISTER=register number]

where:

nnnn = configuration task line number

var_name = integer or boolean variable. Double integer variables can be used but should be avoided because of the possibility that all 32 bits will not transfer in one operation.

WARNING

IF YOU USE DOUBLE INTEGER VARIABLES IN THIS INSTANCE, YOU MUST IMPLEMENT A SOFTWARE HANDSHAKE BETWEEN THE TRANSMITTER AND RECEIVER TO ENSURE THAT BOTH THE LEAST SIGNIFICANT AND MOST SIGNIFICANT 16 BITS HAVE BEEN TRANSMITTED BEFORE THEY ARE READ BY THE RECEIVING APPLICATION PROGRAM. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY.

slot number = slot number of the Modbus Plus Interface module in the rack. This number may range from 0 to 15.

register number = Modicon register number on the module, for example, 10001, 40004, etc.

The format for the IODEF statement is as follows:

nnnn IODEF variable_name [SLOT=slot number, &
REGISTER=register number, BIT=bit number]

where:

nnnn = configuration task line number

variable_name = integer or boolean variable. Double integer variables can be used but should be avoided because of the possibility that all 32 bits will not transfer in one operation.

WARNING

IF YOU USE DOUBLE INTEGER VARIABLES IN THIS INSTANCE, YOU MUST IMPLEMENT A SOFTWARE HANDSHAKE BETWEEN THE TRANSMITTER AND RECEIVER TO ENSURE THAT BOTH THE LEAST SIGNIFICANT AND MOST SIGNIFICANT 16 BITS HAVE BEEN TRANSMITTED BEFORE THEY ARE READ BY THE RECEIVING APPLICATION PROGRAM. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY.

slot number = slot number of the Modbus Plus Interface module in the rack. This number may range from 0 to 15.

register number = Multibus register number on the Modbus Plus Interface module. Register numbers correspond to 16-bit words on the module and range from 4 to 63 and 4096 to 6197.

bit number = bit number of the I/O point in the register. Bit numbers range from 0 to 15. Bit numbers are specified for boolean variables only.

Any variables contained in the rack configuration are accessible by any task on any AutoMax Processor in the rack. When AutoMax variables are referenced in AutoMax tasks, the data is directly obtained from or written to the register image on the module. The data storage for an AutoMax variable mapped to a Modicon register will always exist in the Modbus Plus module's dual port memory.

None of the coils (discrete outputs) or registers are retentive, that is, data is lost if the power is cycled. Recall that all of the registers in the Modicon register image are readable from the Modbus Plus link. However, it is not possible to write to the discrete inputs (10001-14096) and input registers (30001-31024) from the Modbus Plus link or to write to the discrete outputs (coils) from the AutoMax side. See section 4.1.3 for more information.

The registers in the Modbus Plus Interface module's Modicon register image may be displayed using the AutoMax Programming Executive Monitor I/O function. Refer to Appendix G for the method used to convert Modicon register numbers so that they can be viewed using the monitor.

4.2.2 Application Programming

Commands from the Modbus Plus Interface module are initiated from a BASIC application task by executing the GATEWAY_CMD_OK@ function:

```
GATEWAY_CMD_OK@(status%, cmd_code%, slave_drop%,          &
                slave_reg$, master_var!, num_regs%)
```

where:

status is an integer variable representing the location where the status resulting from the operation is stored. Refer to Appendix D for a list of status values.

cmd_code is a variable name or expression of type integer representing the Modbus Plus command sent to the module. The commands are described in section 4.2.4.

slave_drop is a variable name or expression of type integer containing the destination node address. This is the destination node on your Modbus Plus network. If slave_reg contains routing information, slave_drop is the first element in the routing path, i.e., the address of the bridge node on this node's network.

slave_reg is a variable name or expression of type string that specifies the register number on the target device. Register numbers must exist at the target node.

This field can also contain Modbus Plus routing information. The first member of the routing path is placed in the slave_drop field (above). Any additional routing path elements (up to 4 entries) are entered at the beginning of this field. Elements are delimited by periods. There must be a space between the routing path and the target register, for example, "2.6.33 40001". If the target device is another Modbus Plus Interface module, the last element in the routing path must be the slave channel (1-8) to which the message is to be routed. See section 4.2.5 for more information.

master_var is a variable name or expression (usually via the BASIC language VARPTR! function) of type double integer, representing the physical address of first register on Modbus Plus Interface module to be read from/written to.

num_regs is a variable name or expression of type integer that defines the number of bits or registers (16 bits each) to be transferred; cmd_code determines whether the variable represents bits or registers.

The GATEWAY_CMD_OK@ function will be true if the command was successfully completed. If the function is false, the returned status will be an error code. See Appendix D for the error codes returned by the GATEWAY_CMD_OK@ function.

Multiple tasks can access the module. Note, however, that if more than four tasks try to initiate messages at the same time, error code "22" will be returned to the additional tasks attempting accesses.

4.2.3 Enabling the Modbus Plus Interface Module

The connection between the Modbus Plus Interface module and the Modbus Plus network is configured through an AutoMax BASIC application task. Before turning power on to the rack, set the thumbwheel switches to a valid station address (1-64). Once the module is powered up, it will start passing the token, sending and receiving data messages, and responding to diagnostic commands from other nodes.

If you are using AutoMax Programming Executive software version 2.1 or earlier, any required registers must be defined using IODEFs or MODDEFs in the configuration task. If you are using AutoMax Programming Executive software version 3.0 or later, these registers are defined using the Variable Configurator within the Programming Executive. The following example illustrates one method of enabling the module.

Example:

In the configuration, define the following registers:

```
TIMEOUT% [SLOT=4, REGISTER=6]
GLOBAL_LEN% [SLOT=4, REGISTER=7]
RESP_TIME% [SLOT=4, REGISTER=22]
```

The application could enable the module as follows:

```
20  TIMEOUT% = 50      \!   message timeout 5 seconds
30  GLOBAL_LEN% = 7    \!   global data length is 7 words
40  RESP_TIME% = 2     \!   GATEWAY_CMD_OK@ should poll   &
                            for response every 250 ms.
```

4.2.4 Commands Initiated by the Modbus Plus Interface Module

The commands described in figure 4.5 can be used by the AutoMax Processor in the GATEWAY_CMD_OK@ function to initiate a command on the Modbus Plus Interface module. See section 4.2.2 for more information on the GATEWAY_CMD_OK@ function.

Command	Description	Modbus Plus Functions
01	Read Discrete Data	01, 02
02	Write Discrete Data	15
03	Read Register Data	03, 04
04	Write Register Data	16
07	Get Remote Statistics	08
08	Clear Remote Statistics	08

Figure 4.5 - Commands Used with GATEWAY_CMD_OK@

Command 01, Read Discrete Data, transfers discrete data from a remote node's coil (0xxxx) or input status (1xxxx) registers to this node's coil data or holding register data areas. The remote register number must be 0xxxx or 1xxxx while this node's variable must be assigned to the 0xxxx or 4xxxx registers. The transfer size is the number of bits to transfer from the remote node and must not exceed 2000.

Command 02, Write Discrete Data, writes discrete data from any data area on this node to the coil (0xxxx) registers on a remote node. The destination register number must be 0xxxx while this node's variable may be assigned to any register (0xxxx, 1xxxx, 3xxxx or 4xxxx). The transfer size is the number of bits to transfer to the remote node and must not exceed 800.

Command 03, Read Register Data, transfers 16-bit data from a remote node's input (3xxxx) or holding (4xxxx) registers to this node's holding register or coil data areas. The destination register number must be 3xxxx or 4xxxx while this node's variable must be assigned to the 4xxxx or 0xxxx registers. The transfer size is the number of 16-bit values to transfer from the remote node and must not exceed 125. Note that if data is being written into this node's 0xxxx area, the variable name must specify a coil register starting at bit "0" in a 16-bit word, such as 00001, 00017, 00033, 00049, etc.

Command 04, Write Register Data, transfers 16-bit data from any of this node's data areas to a remote node's holding (4xxxx) registers. The destination register number must be 4xxxx while this node's variable may be assigned to any register (0xxxx, 1xxxx, 3xxxx or 4xxxx). The transfer size is the number of 16-bit values to transfer to the remote node and must not exceed 100. Note that if data is being written into this node's 0xxxx area, the variable name must specify a coil register starting at bit

"0" in a 16-bit word, such as 00001, 00017, 00033, 00049, etc.

Command 07, Get Remote Statistics, reads the statistics from the specified remote node into the designated 4xxxx registers on the module. The slave_reg parameter is ignored; the num_regs parameter applies. This command does not support routing paths. The destination (slave_drop) cannot be the local node. The AutoMax application task can read the network statistics for the local node from dual port memory on the Modbus Plus module. The network statistics are located in registers 6144-6197. Refer to Appendix E for a description of these registers.

Command 08, Clear Remote Statistics, is used to clear the remote or local statistics on the specified destination node. The slave_reg, master_var, and num_regs parameters are ignored. However, master_var must be a valid pointer; you cannot enter a zero in this location. This is the only command that can be issued to the local node to clear its network statistics. This command does not support routing paths.

4.2.5 Commands from Remote Nodes

The Modbus Plus Interface module will respond to the following incoming commands from remote nodes:

Function	Description
01	Read output coil
02	Read input coil
03	Read holding registers
04	Read input registers
05	Force single coil
06	Preset single register
08	subfunction 21. Read or clear network statistics
0F	Force multiple coils
10	Preset multiple registers

Figure 4.6 - Commands from Remote Nodes

These commands write directly to or read directly from the register image area on the module. If the address does not exist, an error is returned to the remote device.

When messages from remote nodes are sent to the module, the last element in the routing path must be a number from 1 to 8 which specifies the slave channel to which the message is to be routed. The slave channel is simply a number designating a reserved routing path. Note that although the slave channel range is 1-8, only four paths at a time can be used. Because the slave channel must be included in the path, the maximum number of bridges through which the message can be routed is three. Only one message can be sent to any slave path at a time. As soon as the reply has been sent by the module, the channel is free to receive new messages. This means that without some handshaking, no more than four remote nodes can send messages to this node at one time without generating errors.

Example:

To send a message to a module with a node address of 22, using slave channel 1, the routing path is 22.1. Note that no other node may send a message to slave channel 1 until the reply to the first message has been sent by node 22.

4.2.6 Global Data

Each node can pass up to 32 words of global data with the token. The global data area on the module is read only except for the specific area for this node's global data.

The amount of global data this node sends is set by writing to register 7. The default at startup is zero words of global data. An application task can write to this register at any time to change the amount of global data this node sends.

Each time the Modbus Plus circuitry sees a token pass with global data attached, it notifies the software running on the module. The module software transfers global data from up to eight nodes into its global data registers on each scan (complete token rotation). If all 32 nodes on a network have global data, it takes longer for all global data on the module to be updated.

The global data mask registers (8-11) can be used to improve the efficiency of global data updates. If your application does not care about the global data from specific nodes, set the corresponding bit in the global data mask registers. The software on the module first checks these registers before it moves any data. If the bit is set, it goes on to the next node to check for global data. The global data mask registers can be changed at any time.

4.2.7 Programming Examples

This section contains examples of initiating commands by using the GATEWAY_CMD_OK@ function within an AutoMax task. Recall that commands to a Modbus Plus Interface module node must contain the number of the slave channel as the last element in the routing path in the GATEWAY_CMD_OK@ function. In the following examples, each register is defined by means of MODDEFs in the configuration for the rack. For example,

```
REG_0XXXX@[SLOT=1, REGISTER=00001]
REG_1XXXX@[SLOT=1, REGISTER=10001]
REG_3XXXX%[SLOT=1, REGISTER=30001]
REG_4XXXX%[SLOT=1, REGISTER=40001]
```

The statements above are used in the following examples. The VARPTR! function is used to return a double integer value for the address of the specified argument.

Example 1: Read Discrete Data from a Modicon Node

```
10  COMMON REG_0XXXX@
20  LOCAL STAT%
32  IF NOT GATEWAY_CMD_OK@(STAT%, 01, 7, "00017",          &
        VARPTR!(REG_0XXXX@), 16) THEN 20000 \! process errors
```

Command 01 transfers 16 coils starting at address 00017 on node 7 to a block starting at address 00001 (register 64, bit 0) on this node. The status is returned in variable STAT%.

Example 2: Read Discrete Data from Another Modbus Plus Interface Module Mode

```
10 COMMON REG_0XXXX@
20 LOCAL STAT%
32 IF NOT GATEWAY_CMD_OK@(STAT%, 01, 7, "1 00017",        &
     VARPTR!(REG_0XXXX@), 16) THEN 20000 \! process errors
```

Command 01 transfers 16 coils starting at address 00017 on node 7 using slave channel 1 on the remote node to a block starting at address 00001 (register 64, bit 0) on this node. The status is returned in variable STAT%.

Example 3: Write Discrete Data

```
10 COMMON REG_4XXXX%
20 LOCAL STAT%
21 LOCAL CMD%
22 LOCAL NODE%
30 CMD% = 02
31 NODE% = 3
40 IF NOT GATEWAY_CMD_OK@(STAT%, CMD%, NODE%,             &
     "00033", VARPTR!(REG_4XXXX%), 32)                    &
     THEN 20000 \! process errors
```

Command 02 transfers 32 coils starting at address 40001 on this node to a block starting at address 00033 on remote node 3. The status is returned in variable STAT%. Note that the 32 coils correspond to two holding registers, Modicon 40001 and 40002 (AutoMax registers 1600 and 1601).

Example 4: Read Registers

```
10 COMMON REG_4XXXX%
20 LOCAL STAT%
21 LOCAL CMD%
22 LOCAL NODE%
30 CMD% = 03
31 NODE% = 32
40 IF NOT GATEWAY_CMD_OK@(STAT%, CMD%, NODE%              &
     "30031", VARPTR!(REG_4XXXX%), 125)                   &
     THEN 20000 \! process errors
```

Command 03 transfers the contents of 125 input registers starting at 30031 on remote node 32 to the registers starting at 40001 on this node. The status is returned in variable STAT%.

Example 5: Write Registers

```
10 COMMON REG_3XXXX%
20 LOCAL STAT%
21 LOCAL CMD%
22 LOCAL NODE%
30 CMD% = 04
31 NODE% = 15
40 IF NOT GATEWAY_CMD_OK@(STAT%, CMD%, NODE%,             &
     "40017", VARPTR!(REG_3XXXX%), 100)                   &
     THEN 20000 \! process errors
```

Command 04 transfers the contents of 100 input registers starting at 30001 on this node to the holding registers starting at 40017 on remote node 15. The status is returned in variable STAT%.

Example 6: Get Remote Statistics

```
10 COMMON REG_4XXXX%
20 LOCAL STAT%
21 LOCAL CMD%
22 LOCAL NODE%
30 CMD% = 07
31 NODE% = 55
40 IF NOT GATEWAY_CMD_OK@(STAT%, CMD%, NODE%, "",    &
      VARPTR!(REG_4XXXX%), 12) THEN 20000 \! process errors
```

Command 07 transfers 12 registers of network statistics on remote node 55 to the holding registers starting at address 40001 on this node. The status is returned in variable STAT%. Note that the slave_reg argument is not used.

Note that this command cannot be sent to a Modbus Plus Interface module node.

Example 7: Clear Remote Statistics

```
10 COMMON REG_4XXXX%
20 LOCAL STAT%
21 LOCAL CMD%
22 LOCAL NODE%
30 CMD% = 08
31 NODE% = 19
40 IF NOT GATEWAY_CMD_OK@(STAT%, CMD%, NODE%, "",    &
      VARPTR!(REG_4XXXX%), 12) THEN 20000 \! process errors
```

Command 08 clears the network statistics on remote node 19. The slave_reg, master_var, and num_regs parameters will be ignored. The status is returned in variable STAT%.

Note that this command cannot be sent to a Modbus Plus Interface module node.

Example 8: Clear Local Statistics

```
10 COMMON REG_4XXXX%
11 COMMON MYNODE%
20 LOCAL STAT%
21 LOCAL CMD%
30 CMD% = 08
40 IF NOT GATEWAY_CMD_OK@(STAT%, CMD%, MYNODE%, "",  &
      VARPTR!(REG_4XXXX%), 12) THEN 20000 \! process errors
```

Command 08 clears the network statistics on the local node. The variable MYNODE% is defined in the rack configuration to be register 32. The status is returned in variable STAT%.

Example 9: Write Registers using Routing Path

```
10 COMMON REG_3XXXX%
20 LOCAL STAT%
21 LOCAL CMD%
22 LOCAL NODE%
30 CMD% = 04
31 NODE% = 15
40 IF NOT GATEWAY_CMD_OK@(STAT%, CMD%, NODE%,        &
      "1.2 40017", VARPTR!(REG_3XXX%), 1)            &
      THEN 20000 \! process errors
```

Command 04 transfers the contents of input register 30001 on this node to the holding register 40017 on remote node 2 which is accessed through a bridge node (15) on the local network and a second bridge node (1). The status is returned in variable STAT%.

Example 10: Read Registers from a Node on Another Network

```
10 COMMON REG_4XXXX%
20 LOCAL STAT%
21 LOCAL CMD%
22 LOCAL NODE%
30 CMD% = 03
31 NODE% = 32
20 IF NOT GATEWAY_CMD_OK@(STAT%, CMD%, NODE%,        &
     "1.2.3.4 30031", VARPTR!(REG_4XXXX%), 1)        &
     THEN 20000 \! process errors
```

Command 03 transfers the contents of input register 30031 on remote node 4 to the module register 40001 on this node. The first bridge (on the local network) is node 32. The routing is then through nodes 1, 2, and 3 on intermediate networks. The status is returned in variable STAT%.

5.0 DIAGNOSTICS AND TROUBLESHOOTING

This section describes how to troubleshoot the AutoMax Modbus Plus Interface module. See Appendix C for a list of error codes that can be displayed by the module. If the problem cannot be corrected using the procedures below, the unit is not user-serviceable.

DANGER

ONLY QUALIFIED ELECTRICAL PERSONNEL FAMILIAR WITH THE CONSTRUCTION AND OPERATION OF THIS EQUIPMENT AND THE HAZARDS INVOLVED SHOULD INSTALL, ADJUST, OPERATE OR SERVICE THIS EQUIPMENT. READ AND UNDERSTAND THIS MANUAL AND OTHER APPLICABLE MANUALS IN THEIR ENTIRETY BEFORE PROCEEDING. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN SEVERE BODILY INJURY OR LOSS OF LIFE.

5.1 The OK LED is OFF

Problem: The green status LED (labeled "OK") on the Modbus Plus Interface module faceplate is off. The LED should be on when the module has passed its internal diagnostics after power-up. If the green OK status LED is off, and no error code is displayed, a local watchdog failure has occurred. Try cycling power to the rack. If the OK status LED remains off, replace the module.

5.2 Invalid Station Address

Problem: Error code ".5" appears on the Modbus Plus Interface module's LED display at power-up. This error code indicates an invalid station address. To clear the error code, change the thumbwheel switch settings and cycle power on the module.

5.3 Transmission Link Failures

Problem: Error code "41" is returned by the GATEWAY_CMD_OK@ function. This error code indicates a Response Timeout.

Verify that the cable connections to each device are secure. Check the cable connection at the module faceplate and at the remote device. Check the network cabling and termination.

5.4 Bus Error

Problem: Error code "31" appears on a Processor module's LED display. This error indicates the system has a problem accessing a module in the rack though the backplane bus. A bus error may be caused by removal of a module, a module failure, or a rack backplane failure.

Use the following procedure to isolate a bus error:

Step 1. Verify that all modules are in the correct slot.

Verify that the slot number being referenced in the application tasks agrees with the slot number defined during configuration.

Step 2. Verify the station address is correct.

The thumbwheel switches used to set the station address can be changed while the module is on-line without having any effect on the system. On the next power up, if the thumbwheel switches define a node that is valid, the Processor will accept it as a valid node.

Step 3. Verify that the application software is correct.

Verify that the application software is not attempting to write to READ ONLY registers on the module.

Step 4. Verify that the hardware is working correctly.

DANGER

THIS EQUIPMENT IS AT LINE VOLTAGE WHEN AC POWER IS CONNECTED. DISCONNECT AND LOCKOUT ALL UNGROUNDED CONDUCTORS OF THE AC POWER LINE. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN SEVERE BODILY INJURY OR LOSS OF LIFE.

Make certain that power is off before removing any module from the rack. Systematically replace the Modbus Plus Interface module, the Processor module(s), and the rack/backplane, one at a time, with a corresponding module or assembly known to be operating correctly. After replacing each module or assembly, if the problem is not corrected, replace the original item before going on to the next item.

Appendix A

Technical Specifications

Ambient Conditions

- Storage Temperature: −40°C to 85°C
- Operating Temperature: 0°C to 60°C
- Humidity: 5% to 95%, non-condensing
- Altitude: 1000 meter (3300 feet) without derating

Dimensions

Height: 29.85 cm (11.75 inches)

Width: 3.18 cm (1.25 inches)

Depth: 18.73 cm (7.375 inches)

Weight: 0.9 kg (2 pounds)

Maximum Power Dissipation

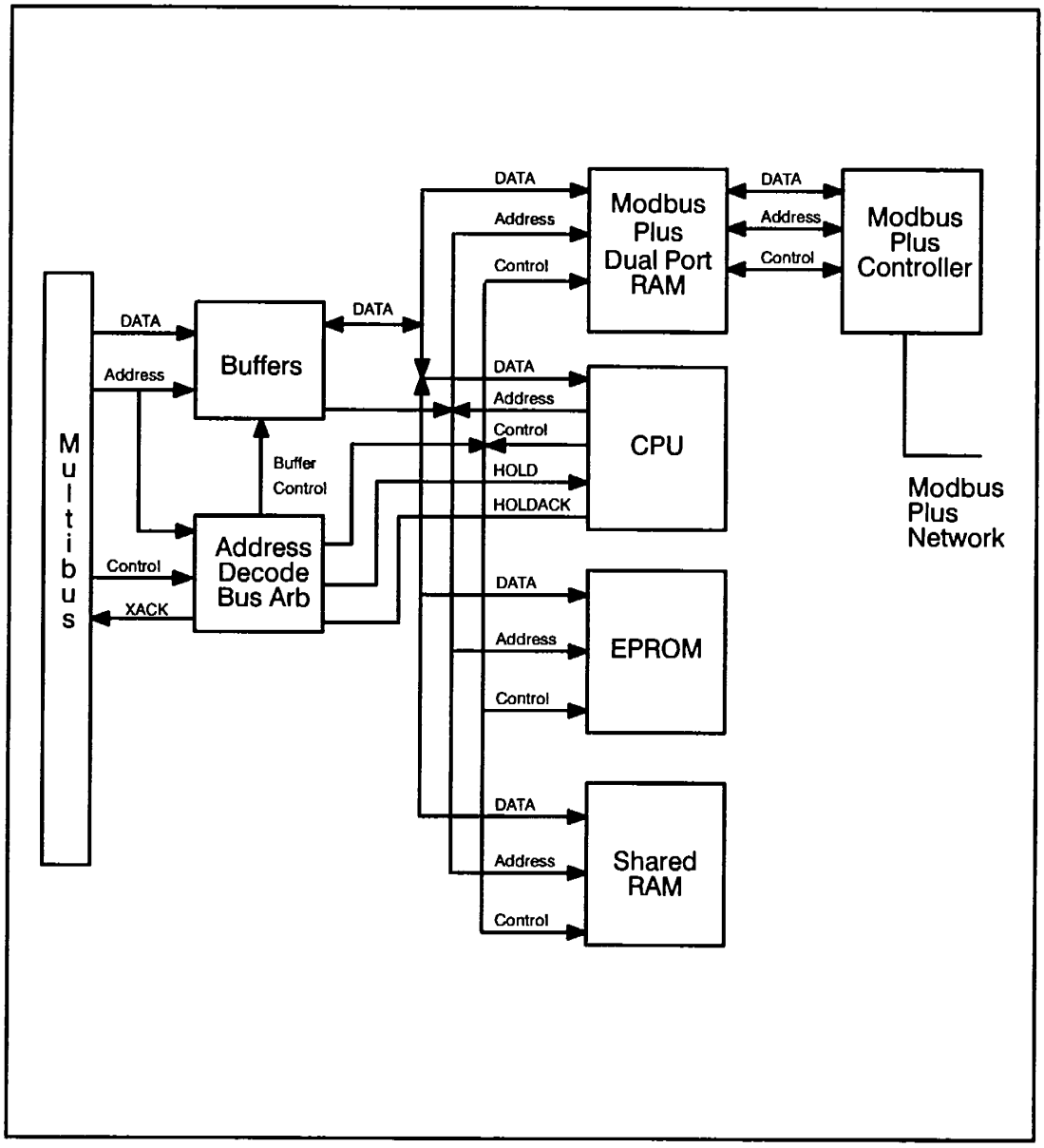
- 3 W

System Power Requirements:

- +5 VDC 600 mA
- +12 VDC (not used)
- −12 VDC (not used)

Appendix B

Module Block Diagram



Appendix C

Modbus Plus Module 7-Segment LED Error Codes

The following error codes may be displayed on the 7-segment display. There are two classes of error codes:

- **non-flash** error codes always have the "." part of the 7-segment display on and consist of just one character
- **flash** error codes sequentially display three digits. Any flash error codes below 200 are Modbus Plus status values and are described in Modicon documentation.

Non-Flash Error Codes

.0	EPROM checksum failure
.1	RAM failure
.2	Modbus Plus DPRAM failure
.3	Memory initialization failure
.5	Bad Modbus Plus station address
.6	Interrupt initialization error
.d	Module re-start failure
.E	Local watchdog timeout
.F	Power failed

Flash Error Codes

000	Monitor link operation
064	Never getting token
096	Sole station
128	Duplicate node detected
0FF	Module is off-line (register 5 bit 0 is 0)
2xx	**Module usage errors**
200	Invalid global data length (> 32)
3xx	**Modbus Plus chipset initialization errors**
300	Interrupt execution error
301	Waiting for Modbus Plus hardware startup
302	Modbus Plus diagnostic failure
303	Modbus Plus diagnostic data mismatch
304	Initial configuration status timeout
305	Initial data master path abort error
306	Initial data slave path abort error
307	Initial program master path abort error
308	Initial program slave path abort error
309	Main loop start failure
4xx	**Modbus Plus interrupt command timeouts**
400	Get service request command timeout
401	Get master reply command timeout
402	Master reply abort command timeout
403	Get slave message command timeout
404	Read hold registers put slave reply command timeout
405	Write hold registers put slave reply command timeout
406	Unrecognized command slave path abort command timeout
407	Program master abort command timeout
408	Program slave abort command timeout

5xx	**Modbus Plus main loop command timeouts**
500	Configuration command status timeout
501	Get service request command timeout
502	Get global data command timeout
503	Put global data command timeout
504	Put master message command timeout
505	Master path timeout abort command timeout
6xx	**Modbus Plus interrupt state error, xx = crash code**
7xx	**Modbus Plus reset error, xx = crash code**

Appendix D

GATEWAY_CMD_OK@ Status Codes

0 Operation successful
1 Invalid operation
3 Invalid length
4 Invalid offset
5 Invalid length + offset
6 Invalid destination node
7 Invalid node in routing path
8 Invalid master_var data address
9 Invalid route (equal to own address)
A Global read request more than available
B Inconsistent Modbus slave response
C Inconsistent network response
1x Modbus slave exception response
x can have the following values:

- 1 Illegal function for the addressed slave
- 2 Illegal data address within the information field for the addressed slave
- 3 Illegal data value in the information field for the addressed slave
- 6 Busy. The function just requested cannot be performed at this time because a long duration program command is being processed; re-issue the command later.
- 7-F unassigned

20 Parameter specifying data address on Modbus Plus module is invalid. The slave_reg or destination path is invalid.
21 Modbus Plus module not found or inaccessible
22 No available master data path
4x Routing failure x can have the following values:

- 1 No response received
- 2 Program access denied
- 3 Node is offline and unable to communicate
- 4 exception response received
- 5 router node data paths busy
- 6 slave device down
- 7 bad destination node address
- 8 invalid station type in data path
- 9 slave rejected the Modbus command
- A initiated transaction forgotten by slave
- B unexpected master output path received
- C unexpected response received

Appendix E

Register Map for Network Statistics

This area contains statistics for the network as a whole and also for this node. These values are returned to any remote node which requests them from this node. This is also the format of network statistics returned from remote nodes to this node. These registers are all read only.

Word	Byte	Meaning
6144		Station type ID 0 unknown node type 1 standard PLC node 2 Modbus bridge node 3 PC-Plus node 4 Router node
6145		Communications microprocessor version. First release is version 1.00 and displays as 0100 (hex)
6146		Network address for this node
6147		MAC state variable as follows: 0 power-up state 1 Monitor offline state 2 Duplicate offline state 3 Idle state 4 Use token state 5 Work response state 6 Pass token state 7 Solicit response state 8 Check pass state 9 Claim token state 10 Claim response state
6148		Peer status of this unit relative to the network, as follows: 0 Monitor link operation 32 Normal link operation 64 Never getting token 96 Sole station 128 Duplicate station
6149		Token pass counter. Increments each time this station gets the token.
6150		Token rotation time in milliseconds
6151	LO	Data master failed during token ownership bit map
	HI	Program master failed during token ownership bit map
6152	LO	Data master token owner work bit map
	HI	Program master token owner work bit map
6153	LO	Data slave token owner work bit map
	HI	Program slave token owner work bit map
6154	LO	Data master/get master response transfer request bit map
	HI	Data slave/get slave command transfer request bit map
6155	LO	Program-master-get-master-response transfer request bit map
	HI	Program slave/get slave command transfer request bit map
6156	LO	Program-master connect status bit map
	HI	Program-slave automatic logout request bit map

6157	LO	Pre-transmit deferral error counter
	HI	Receive buffer DMA overrun error counter
6158	LO	Repeated command received counter
	HI	Frame size error counter
6159	LO	Receiver collision - abort error counter
	HI	Receiver alignment error counter
6160	LO	Receiver CRC error counter
	HI	Bad packet length error counter
6161	LO	Bad link-address error counter
	HI	Transmit buffer DMA underrun error counter
6162	LO	Bad internal packet length error counter
	HI	Bad MAC function code error counter
6163	LO	Communication retry counter
	HI	Communication failed error counter
6164	LO	Good receive packet success counter
	HI	No response received error counter
6165	LO	Exception response received error counter
	HI	Unexpected path error counter
6166	LO	Unexpected response error counter
	HI	Forgotten transaction error counter
6167	LO	Active station table bit map, nodes 1-8
	HI	Active station table bit map, nodes 9-16
6168	LO	Active station table bit map, nodes 17-24
	HI	Active station table bit map, nodes 25-32
6169	LO	Active station table bit map, nodes 33-40
	HI	Active station table bit map, nodes 41-48
6170	LO	Active station table bit map, nodes 49-56
	HI	Active station table bit map, nodes 57-64
6171	LO	Token station table bit map, nodes 1-8
	HI	Token station table bit map, nodes 9-16
6172	LO	Token station table bit map, nodes 17-24
	HI	Token station table bit map, nodes 25-32
6173	LO	Token station table bit map, nodes 33-40
	HI	Token station table bit map, nodes 41-48
6174	LO	Token station table bit map, nodes 49-56
	HI	Token station table bit map, nodes 57-64
6175	LO	Global data present table bit map, nodes 1-8
	HI	Global data present table bit map, nodes 9-16
6176	LO	Global data present table bit map, nodes 17-24
	HI	Global data present table bit map, nodes 25-32
6177	LO	Global data present table bit map, nodes 33-40
	HI	Global data present table bit map, nodes 41-48
6178	LO	Global data present table bit map, nodes 49-56
	HI	Global data present table bit map, nodes 57-64
6179	LO	Receive buffer in use bit map, buffers 1-8
	HI	Receive buffer in use bit map, buffers 9-16
6180	LO	Receive buffer in use bit map, buffers 17-24
	HI	Receive buffer in use bit map, buffers 25-32
6181	LO	Receive buffer in use bit map, buffers 33-40
	HI	Station management command processed initiation counter

6182	LO	Data master output path 1 command initiation counter
	HI	Data master output path 2 command initiation counter
6183	LO	Data master output path 3 command initiation counter
	HI	Data master output path 4 command initiation counter
6184	LO	Data master output path 5 command initiation counter
	HI	Data master output path 6 command initiation counter
6185	LO	Data master output path 7 command initiation counter
	HI	Data master output path 8 command initiation counter
6186	LO	Data slave input path 41 command processed counter
	HI	Data slave input path 42 command processed counter
6187	LO	Data slave input path 43 command processed counter
	HI	Data slave input path 44 command processed counter
6188	LO	Data slave input path 45 command processed counter
	HI	Data slave input path 46 command processed counter
6189	LO	Data slave input path 47 command processed counter
	HI	Data slave input path 48 command processed counter
6190	LO	Program master output path 81 command initiation counter
	HI	Program master output path 82 command initiation counter
6191	LO	Program master output path 83 command initiation counter
	HI	Program master output path 84 command initiation counter
6192	LO	Program master output path 85 command initiation counter
	HI	Program master output path 86 command initiation counter
6193	LO	Program master output path 87 command initiation counter
	HI	Program master output path 88 command initiation counter
6194	LO	Program slave input path C1 command processed counter
	HI	Program slave input path C2 command processed counter
6195	LO	Program slave input path C3 command processed counter
	HI	Program slave input path C4 command processed counter
6196	LO	Program slave input path C5 command processed counter
	HI	Program slave input path C6 command processed counter
6197	LO	Program slave input path C7 command processed counter
	HI	Program slave input path C8 command processed counter
6198 - 7167		Reserved for future use

Appendix F

Modbus Plus Network Cable Components

The Modbus Plus network is connected using Belden 9841 shielded twisted pair cable. The cable should be run directly between the network device locations. Each cable segment must be a continuous run between the device connectors at two locations. Splices, taps, splitters, or other configurations such as "star" or "tree" configurations should not be used. The only media components allowed are the network cable and network device connectors.

Two types of 9-pin D-shell connectors are available from Modicon for connecting devices to the network:

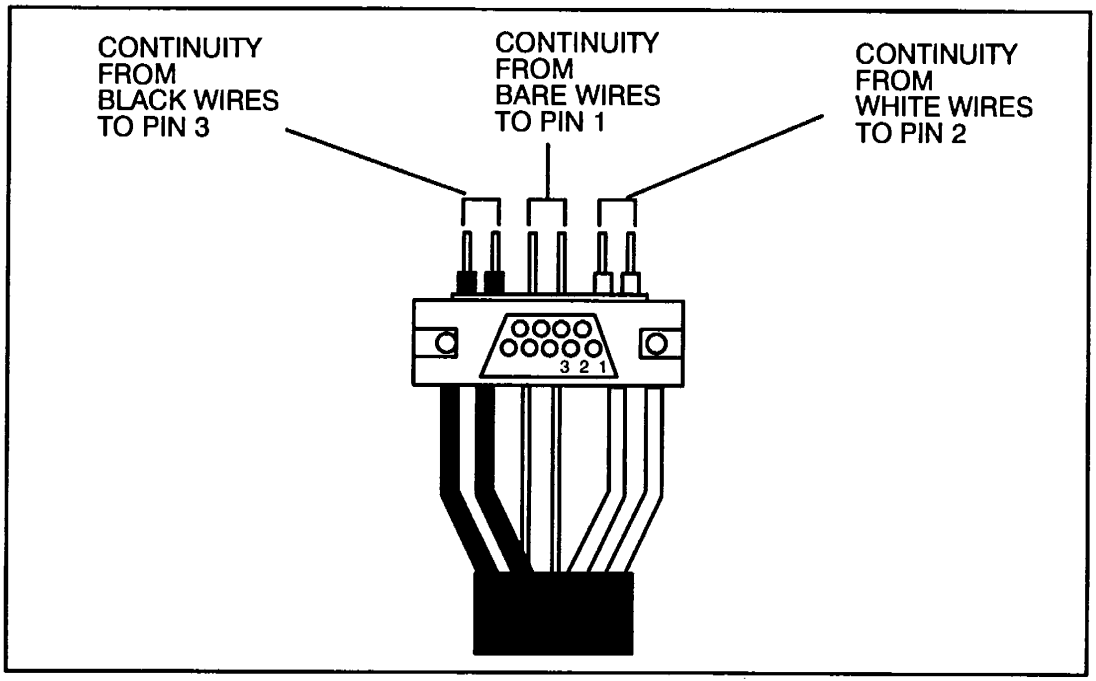
- Each in-line node (station) requires a line connector, Modicon part number AS-MBKT-085, or equivalent.
- The nodes at the two ends of the network each require a terminating connector, Modicon part number AS-MBKT-185. You can also use a standard connector equipped with a 120 ohm terminating resistor at each end of the network.

When the terminating connectors (or connectors with 120 ohm resistors) are installed at the two extreme ends of the cable, they furnish the proper terminating impedance for the network. The cable termination is maintained regardless of whether a node is connected to the terminating connector at either end of the cable. No other termination is required.

The 9-pin D-shell connector labeled "NETWORK" has the following pinout:

Modbus Plus Interface Module			**Other Nodes**	
Line	**Pin**		**Pin**	**Line**
shield	1	--------	1	shield
white	2	--------	2	white
black	3	--------	3	black

Note that connectors to in-ine nodes require two cables (six wires); connectors to terminating nodes require one cable (three wires).



Appendix G

Converting Modicon Register Numbers to AutoMax Register and Bit Numbers for Monitoring

The data registers in the dual port memory of the Modbus Plus Interface module may be displayed using the Monitor I/O function in the AutoMax Programming Executive. This appendix provides a method for converting the Modicon register number to a register number recognized by the Monitor I/O application.

The following table shows the correspondence between Modicon register numbers and AutoMax register numbers.

Modicon Register	AutoMax Register
00001-00017	64
.	.
.	.
.	.
04081-04096	319
10001-10017	320
.	.
.	.
.	.
14081-14096	575
30001 576	
.	.
.	.
.	.
31024	1599
40001	1600
.	.
.	.
.	.
41984	3583

Single Bit Registers

Modicon registers 0xxxx and 1xxxx are single-bit registers. Therefore, each AutoMax register contains 16 Modicon "registers." The lowest number Modicon "register" is stored in the least significant bit (bit 0) of the AutoMax register. Use the following formula to calculate the AutoMax register number for Modicon registers 0xxxx or 1xxxx.

Step 1:

$$I.R = \frac{(xxxx - 1)}{16}$$

where:

xxxx = last four digits of the Modicon register number

I = integer portion of the quotient
R = fractional part of the quotient

For 0xxxx registers, the AutoMax register = I + 64
For 1xxxx registers, the AutoMax register = I + 320

Step 2:

The formula to calculate the corresponding bit number within the AutoMax register is:

Bit# = 0.R x 16

Example 1: Modicon register 03547

$$I.R = \frac{3547 - 1}{16} = 221.625$$

Monitor I/O register = 221 + 64 = 285
Bit# = 0.625 x 16 = 10

Example 2: Modicon register 11233

$$I.R = \frac{1233 - 1}{16} = 77.0$$

Monitor I/O register = 77 + 320 = 397
Bit# = 0.0 x 16 = 0

16-Bit Registers

The conversion process is more straightforward for Modicon integer registers (3xxxx and 4xxxx) since both the Modicon registers and the AutoMax registers are 16 bits wide.

For 3xxxx registers, the AutoMax register = xxxx + 575. For 4xxxx registers, the AutoMax register = xxxx + 1599.

Example 3: Modicon register 30962

Monitor I/O register = 0962 + 575 = 1537

Example 4: Modicon register 40251

Monitor I/O register = 0251 + 1599 = 1850

Appendix H

Compatibility with Earlier Versions of the Module

The register description for the module (B/M 57441-A) has been revised such that the default value for the module enable bit (register 5, bit 0) now defaults to ON/TRUE. The module will be enabled upon power-up. The enable bit in the original version of the module (B/M 57441) defaulted to OFF/FALSE.

For additional information

1 Allen-Bradley Drive
Mayfield Heights, Ohio 44124 USA
Tel: (800) 241-2886 or (440) 646-3599
http://www.reliance.com/automax

www.rockwellautomation.com

Corporate Headquarters
Rockwell Automation, 777 East Wisconsin Avenue, Suite 1400, Milwaukee, WI, 53202-5302 USA, Tel: (1) 414.212.5200, Fax: (1) 414.212.5201

Headquarters for Allen-Bradley Products, Rockwell Software Products and Global Manufacturing Solutions
Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation SA/NV, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, 27/F Citicorp Centre, 18 Whitfield Road, Causeway Bay, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Headquarters for Dodge and Reliance Electric Products
Americas: Rockwell Automation, 6040 Ponders Court, Greenville, SC 29615-4617 USA, Tel: (1) 864.297.4800, Fax: (1) 864.281.2433
Europe/Middle East/Africa: Rockwell Automation, Brühlstraße 22, D-74834 Elztal-Dallau, Germany, Tel: (49) 6261 9410, Fax: (49) 6261 17741
Asia Pacific: Rockwell Automation, 55 Newton Road, #11-01/02 Revenue House, Singapore 307987, Tel: (65) 6356-9077, Fax: (65) 6356-9011

Publication J2-3051-1 - December 1993