FANUC Robot series

R-30*i*A/R-30*i*A Mate CONTROLLER

Modbus TCP Interface

OPERATOR'S MANUAL

B-82844EN/01

Before using the Robot, be sure to read the “FANUC Robot Safety Manual (B-80687EN)” and understand the content.

The products in this manual are controlled based on Japan’s “Foreign Exchange and Foreign Trade Law”. The export from Japan may be subject to an export license by the government of Japan.
Further, re-export to another country may be subject to the license of the government of the country from where the product is re-exported. Furthermore, the product may also be controlled by re-export regulations of the United States government.
Should you wish to export or re-export these products, please contact FANUC for advice.

In this manual we have tried as much as possible to describe all the various matters.
However, we cannot describe all the matters which must not be done, or which cannot be done, because there are so many possibilities.
Therefore, matters which are not especially described as possible in this manual should be regarded as “impossible”.

SAFETY

1 SAFETY PRECAUTIONS

For the safety of the operator and the system, follow all safety precautions when operating a robot and its peripheral devices installed in a work cell.

1.1 OPERATOR SAFETY

Operator safety is the primary safety consideration. Because it is very dangerous to enter the operating space of the robot during automatic operation, adequate safety precautions must be observed.

The following lists the general safety precautions. Careful consideration must be made to ensure operator safety.

(1) Have the robot system operators attend the training courses held by FANUC.

FANUC provides various training courses. Contact our sales office for details.

(2) Even when the robot is stationary, it is possible that the robot is still ready to move state and is waiting for a signal. In this state, the robot is regarded as still in motion. To ensure operator safety, provide the system with an alarm to indicate visually or aurally that the robot is in motion.

(3) Install a safety fence with a gate so that no operator can enter the work area without passing through the gate. Equip the gate with an interlock that stops the robot when the gate is opened.

The controller is designed to receive this interlock signal. When the gate is opened and this signal received, the controller stops the robot in an emergency. For connection, see Fig.1.1.

(4) Provide the peripheral devices with appropriate grounding (Class 1, Class 2, or Class 3).

(5) Try to install the peripheral devices outside the work area.

(6) Draw an outline on the floor, clearly indicating the range of the robot motion, including the tools such as a hand.

(7) Install a mat switch or photoelectric switch on the floor with an interlock to a visual or aural alarm that stops the robot when an operator enters the work area.

(8) If necessary, install a safety lock so that no one except the operator in charge can turn on the power of the robot.

The circuit breaker installed in the controller is designed to disable anyone from turning it on when it is locked with a padlock.

(9) When adjusting each peripheral device independently, be sure to turn off the power of the robot.

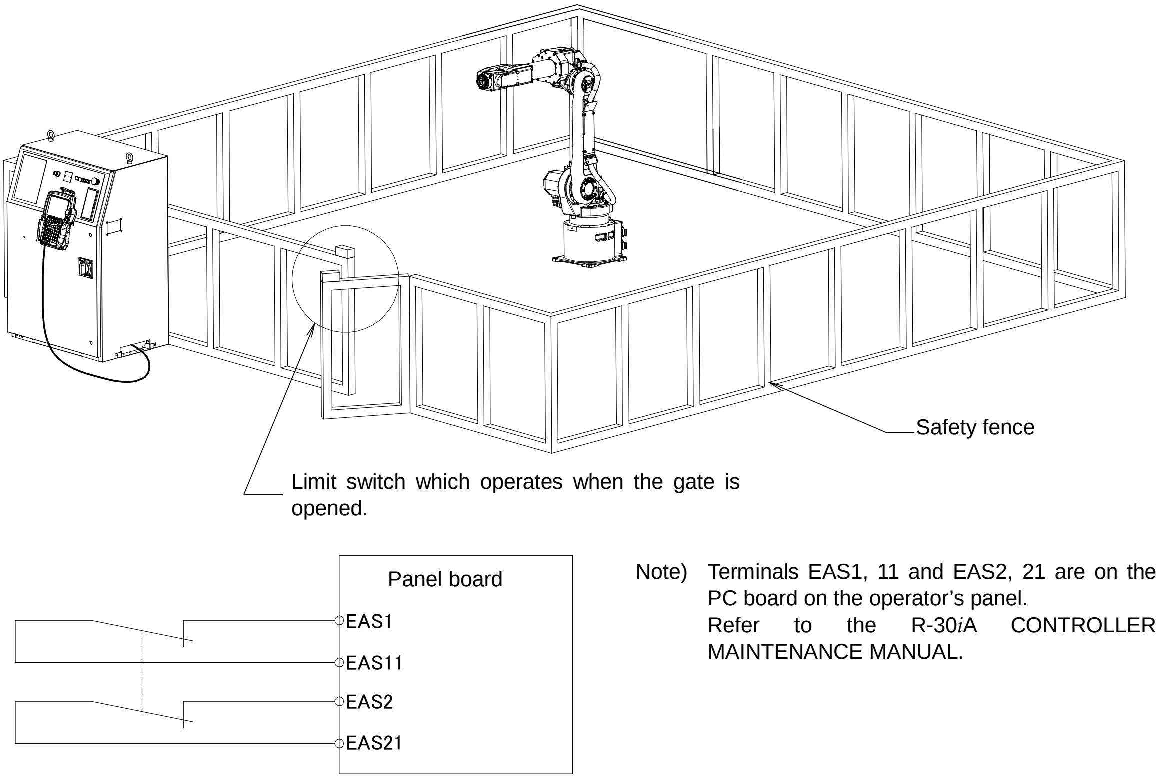


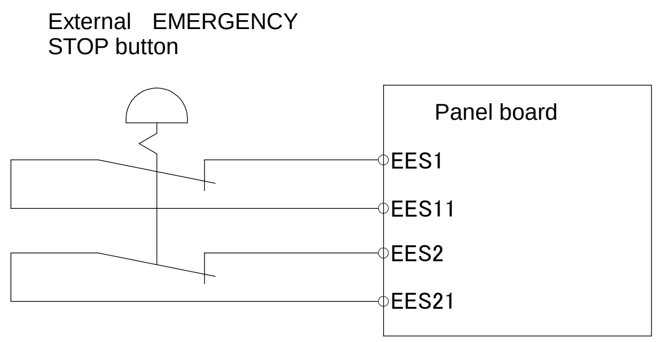
Fig.1.1 Safety fence and safety (for R-30*i*A controller)

1.1.1 Operator Safety

The operator is a person who operates the robot system. In this sense, a worker who operates the teach pendant is also an operator. However, this section does not apply to teach pendant operators.

(1) If it is not necessary for the robot to operate, turn off the power of the robot controller or press the EMERGENCY STOP button, and then proceed with necessary work
(2) Operate the robot system at a location outside the work area.
(3) Install a safety fence with a safety gate to prevent any worker other than the operator from entering the work area unexpectedly and also to prevent the worker from entering a dangerous area.
(4) Install an EMERGENCY STOP button within the operator's reach.

The robot controller is designed to be connected to an external EMERGENCY STOP button. With this connection, the controller stops the robot operation when the external EMERGENCY STOP button is pressed. See the diagram below for connection.



Note) Connect between EES1 and EES11 and between EES2 and EES21. Terminals EES1, EES11, EES2 and EES21 are on the Panel board. Refer to R-30*i*A CONTROLLER MAINTENANCE MANUAL.

Fig.1.1.1 Connection diagram for external emergency stop switch (for R-30*i*A controller)

1.1.2 Safety of the Teach Pendant Operator

While teaching the robot, it is necessary for the operator to enter the work area of the robot. It is particularly necessary to ensure the safety of the teach pendant operator.

(1) Unless it is specifically necessary to enter the robot work area, carry out all tasks outside the area.
(2) Before teaching the robot, check that the robot and its peripheral devices are all in the normal operating condition.
(3) When entering the robot work area and teaching the robot, be sure to check the location and condition of the safety devices

(such as the EMERGENCY STOP button and the deadman's switch on the teach pendant).

The teach pendant supplied by FANUC is provided with a teach pendant enable switch and a deadman's switch in addition to the EMERGENCY STOP button. The functions of each switch are as follows.

EMERGENCY STOP button	:	Pressing this button stops the robot in an emergency, irrespective to the condition of the teach pendant enable switch.
Deadman's switch	:	The function depends on the state of the teach pendant enable switch.
When the enable switch is on	-	Releasing the finger from the dead man's switch stops the robot in an emergency.
When the enable switch is off	-	The deadman's switch is ineffective

NOTE

The deadman's switch is provided so that the robot operation can be stopped simply by releasing finger from the teach pendant in case of emergency.

(4) The teach pendant operator should pay careful attention so that no other workers enter the robot work area.

NOTE

In addition to the above, the teach pendant enable switch and the deadman's switch also have the following function. By pressing the deadman's switch while the enable switch is on, the emergency stop factor (normally the safety gate) connected to the controller is invalidated. In this case, it is possible for an operator to enter the fence during teach operation without pressing the EMERGENCY STOP button. In other words, the system understands that the combined operations of pressing the teach pendant enable switch and pressing the deadman's switch indicates the start of teaching.

The teach pendant operator should be well aware that the safety gate is not functional under this condition and bear full responsibility to ensure that no one enters the fence during teaching.

(5) When entering the robot work area, the teach pendant operator should enable the teach pendant whenever he or she enters the robot work area. In particular, while the teach pendant enable switch is off, make certain that no start command is sent to the robot from any operator's panel other than the teach pendant.

The teach pendant, operator panel, and peripheral device interface send each robot start signal. However, the validity of each signal changes as follows depending on the ON/OFF switch on the Teach pendant and the three modes switch on the Operator's panel and Remote condition on the software.

Operator 's panel Three modes switch	Teach pendant ON/OFF switch	Software remote condition	Teach pendant	Operator's panel	Peripheral devices
T1/T2 AUTO (Except RIA)	On	Independent	Allowed to start	Not allowed	Not allowed
AUTO	Off	Remote OFF	Not allowed	Allowed to start	Not allowed
AUTO	Off	Remote ON	Not allowed	Not allowed	Allowed to start

NOTE) When starting the system using the teach pendant in the RIA specification, the three modes switch should be T1/T2.

(6) To start the system using the operator's box, make certain that nobody is in the robot work area and that there are no abnormal conditions in the robot work area.

(7) When a program is completed, be sure to carry out a test run according to the procedure below.

(a) Run the program for at least one operation cycle in the single step mode at low speed.
(b) Run the program for at least one operation cycle in the continuous operation mode at low speed.
(c) Run the program for one operation cycle in the continuous operation mode at the intermediate speed and check that no abnormalities occur due to a delay in timing.
(d) Run the program for one operation cycle in the continuous operation mode at the normal operating speed and check that the system operates automatically without trouble.
(e) After checking the completeness of the program through the test run above, execute it in the automatic operation mode.

(8) While operating the system in the automatic operation mode, the teach pendant operator should leave the robot work area.

1.1.3 Safety During Maintenance

For the safety of maintenance personnel, pay utmost attention to the following.

(1) Except when specifically necessary, turn off the power of the controller while carrying out maintenance. Lock the power switch, if necessary, so that no other person can turn it on.
(2) When disconnecting the pneumatic system, be sure to reduce the supply pressure.
(3) Before the start of teaching, check that the robot and its peripheral devices are all in the normal operating condition.
(4) If it is necessary to enter the robot work area for maintenance when the power is turned on, the worker should indicate that the machine is being serviced and make certain that no one starts the robot unexpectedly.
(5) Do not operate the robot in the automatic mode while anybody is in the robot work area.
(6) When it is necessary to maintain the robot alongside a wall or instrument, or when multiple workers are working nearby, make certain that their escape path is not obstructed.
(7) When a tool is mounted on the robot, or when any moving device other than the robot is installed, such as belt conveyor, pay careful attention to its motion.
(8) If necessary, have a worker who is familiar with the robot system stand beside the operator's panel and observe the work being performed. If any danger arises, the worker should be ready to press the EMERGENCY STOP button at any time.
(9) When replacing or reinstalling components, take care to prevent foreign matter from entering the system.
(10) When handling each unit or printed circuit board in the controller during inspection, turn off the power of the controller and also turn off the circuit breaker to protect against electric shock.
(11) When replacing parts, be sure to use those specified by FANUC. In particular, never use fuses or other parts of non-specified ratings. They may cause a fire or result in damage to the components in the controller.

1.2 SAFETY OF THE TOOLS AND PERIPHERAL DEVICES

1.2.1 Precautions in Programming

(1) Use a limit switch or other sensor to detect a dangerous condition and, if necessary, design the program to stop the robot when the sensor signal is received.

(2) Design the program to stop the robot when an abnormal condition occurs in any other robots or peripheral devices, even though the robot itself is normal.

(3) For a system in which the robot and its peripheral devices are in synchronous motion, particular care must be taken in programming so that they do not interfere with each other.

(4) Provide a suitable interface between the robot and its peripheral devices so that the robot can detect the states of all devices in the system and can be stopped according to the states.

1.2.2 Precautions for Mechanism

(1) Keep the component cells of the robot system clean, and operate the robot in an environment free of grease, water, and dust.

(2) Employ a limit switch or mechanical stopper to limit the robot motion so that the robot does not encounter its peripheral devices or tools.

1.3 SAFETY OF THE ROBOT MECHANISM

1.3.1 Precautions in Operation

(1) When operating the robot in the jog mode, set it at an appropriate speed so that the operator can manage the robot in any eventuality.
(2) Before pressing the jog key, be sure you know in advance what motion the robot will perform in the jog mode.

1.3.2 Precautions in Programming

(1) When the work areas of robots overlap, make certain that the motions of the robots do not interfere with each other.
(2) Be sure to specify the predetermined work origin in a motion program for the robot and program the motion so that it starts from the origin and terminates at the origin. Make it possible for the operator to easily distinguish at a glance that the robot motion has terminated.

1.3.3 Precautions for Mechanisms

(1) Keep the work area of the robot clean, and operate the robot in an environment free of grease, water, and dust.

1.4 SAFETY OF THE END EFFECTOR

1.4.1 Precautions in Programming

(1) To control the pneumatic, hydraulic and electric actuators, carefully consider the necessary time delay after issuing each control command up to actual motion and ensure safe control.
(2) Provide the end effector with a limit switch, and control the robot system by monitoring the state of the end effector.

1.5 SAFETY IN MAINTENANCE

(1) Never enter the robot work area while the robot is operating. Turn off the power before entering the robot work area for inspection and maintenance.
(2) If it is necessary to enter the robot work area with the power turned on, first press the EMERGENCY STOP button on the operator’s box.
(3) When replacing or reinstalling components, take care to prevent foreign matter from entering the system. When replacing the parts in the pneumatic system, be sure to reduce the pressure in the piping to zero by turning the pressure control on the air regulator.
(4) When handling each unit or printed circuit board in the controller during inspection, turn off the power of the controller and turn off the circuit breaker to protect against electric shock.
(5) When replacing parts, be sure to use those specified by FANUC. In particular, never use fuses or other parts of non-specified ratings. They may cause a fire or result in damage to the components in the controller.
(6) Before restarting the robot, be sure to check that no one is in the robot work area and that the robot and its peripheral devices are all in the normal operating state.

1.6 WARNING LABEL

(1) Greasing and degreasing label

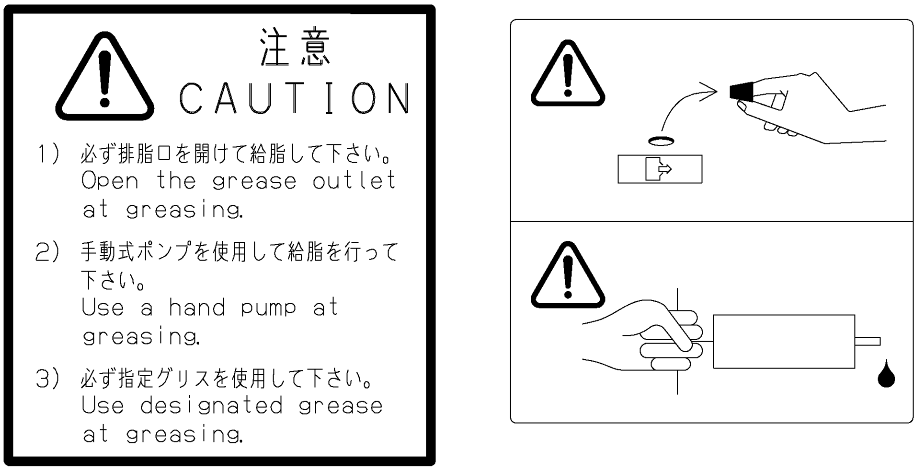


Fig. 1.6 (a) Greasing and degreasing label

Description

When greasing and degreasing, observe the instructions indicated on this label.

1) When greasing, be sure to keep the grease outlet open.
2) Use a manual pump to grease.
3) Be sure to use specified grease.

(2) Step-on prohibitive label

Fig. 1.6 (b) Step-on prohibitive label

Description

Do not step on or climb the robot or controller as it may adversely affect the robot or controller and you may get hurt if you lose your footing as well.

(3) High-temperature warning label

Fig. 1.6 (c) High-temperature warning label

Description

Be cautious about a section where this label is affixed, as the section generates heat. If you have to inevitably touch such a section when it is hot, use a protective provision such as heat-resistant gloves.

TABLE OF CONTENTS

1 INTRODUCTION

The Modbus TCP interface supports an I/O exchange with other Modbus TCP enabled devices over an Ethernet network. The Modbus TCP option on the robot only acts as a server (slave), and therefore will only exchange I/O with client (master) devices. Modbus TCP uses TCP/IP and is based on a Client-Server model. Modbus TCP can be configured to exchange up to the full amount of I/O supported on the robot. The Modbus TCP specification is managed by the Modbus-IDA.

Ethernet is nondeterministic and Modbus TCP does not guarantee any timing. The robot Modbus TCP server will time-out and post an error if an I/O exchange is not received within the configured time limit, unless time-outs are disabled. Good network design and topology is very critical for successful deployment of Modbus TCP for fast and reliable communications between any two devices.

2 SYSTEM OVERVIEW

2.1 MODBUS OVERVIEW

The robot Modbus TCP server supports up to 4 simultaneous Modbus TCP connections. These connections are classified into two connection pools: the Priority connection pool, and the Non-Priority connection pool. By default, all connections are Non-Priority.

Up to two connections might be marked as Priority Connections by specifying the Modbus TCP clients' IP address. This means no other Modbus TCP client device can connect to the robot and use a Priority connection except the device with the corresponding IP address.

Any Modbus TCP client device can connect to a Non-Priority connection, unless the robot is configured with a Non-Priority connection pool of size 0. If all available non-priority connections are used, and another Modbus TCP client attempts to make a connection to the Non-Priority pool, the oldest existing Non-Priority connection will be closed, and the new connection request will be honored.

The Modbus TCP interface corresponds to Rack 96 Slot 1 in the robot for I/O mapping. Any amount of I/O can be mapped with Modbus TCP, up to the maximum supported on the robot.

Good network design is critical to having reliable communications. Excessive traffic and collisions must be avoided or managed.

2.2 I/O TABLES

2.2.1 Overview

Modbus bases its data model on a series of tables:

Table 2.1 Modbus data model

Table	Object Type	Type of	Robot Mapping
Discrete inputs	Single bit	Read-Only	Digital Output (DO)
Coils	Single bit	Read-Write	Digital Input (DI)
Input Registers	16–bit word	Read-Only	Digital Output (DO)
Holding Registers	16–bit word	Read-Write*	Digital Input (DI) and Digital Output (DO)

* Even though Robot Digital Outputs (DO) can be accessed as Modbus Holding Registers, Robot Digital Outputs are still Read-Only. A Modbus Illegal Address error code will be returned if Robot Digital Outputs are written to as Holding Registers.

The Modbus Discrete Inputs and Input Register tables are overlaid and mapped to robot Digital Outputs (DO) on the robot. Likewise, the Coils and Holding Register tables are overlaid and mapped to Digital Inputs (DI).
Table 2.2 is a quick reference for the Modbus to Robot address map.

NOTE
In all examples throughout this documentation, DO [1] refers to the first Digital Output point allocated to Modbus Rack 96, slot 1 on the robot. Likewise, DI [1] refers to the first Digital Input point allocated to Modbus Rack 96, slot 1 on the robot

NOTE
All registers are returned by the robot in big-endian format

Table 2.2 Modbus to Robot address mapping (quick reference)

Modbus Address	Robot I/O
Discrete input 0 Discrete input 1 Discrete input 2 ...	Digital Output: DO [1] Digital Output: DO [2] Digital Output: DO [3] ...
Coils 0 Coils 1 Coils 2 ...	Digital Input: DI [1] Digital Input: DI [2] Digital Input: DI [3] ...
Input Register 0 Input Register 1 Input Register 2 ...	Digital Outputs (DO [1] — DO [16]) Digital Outputs (DO [17] — DO [32]) Digital Outputs (DO [33] — DO [48]) ...
Holding Registers 0 Holding Registers 1 Holding Registers 2 ...	Digital Inputs (DI [1] — DI [16]) Digital Inputs (DI [17] — DI [32]) Digital Inputs (DI [33] — DI [48]) ...
Holding Registers 10000 (read-only) Holding Registers 10001 (read-only) Holding Registers 10002 (read-only) ...	Digital Outputs (DI [1] — DI [16]) Digital Outputs (DI [17] — DI [32]) Digital Outputs (DI [33] — DI [48]) ...

2.2.2 Discrete Inputs and Input Registers

Discrete Inputs are accessible starting at address 0. For example, “read input discrete 4” would access the fifth Robot Digital Output point (DO [5]), as seen by the diagonal-striped area shown in Fig. 2.1.
Input Registers are also accessible starting at address 0. For example, “read input register 2” would access the third set of 16 Digital Output points (DO [33]-DO [48]) as a register, as seen in the horizontally-striped area Fig. 2.1 Note that when reading Modbus Registers, robot data is returned in big-endian format with the output points numbered from the least significant bit (right hand side) to the most significant bit (left hand side), also shown in Fig. 2.1.
Having these two tables overlaid means that reading discrete inputs 0 through 15 would effectively be the same as reading input register 0. In addition, reading discrete inputs 16 through 31 would effectively be the same as reading input register 1, and so on.

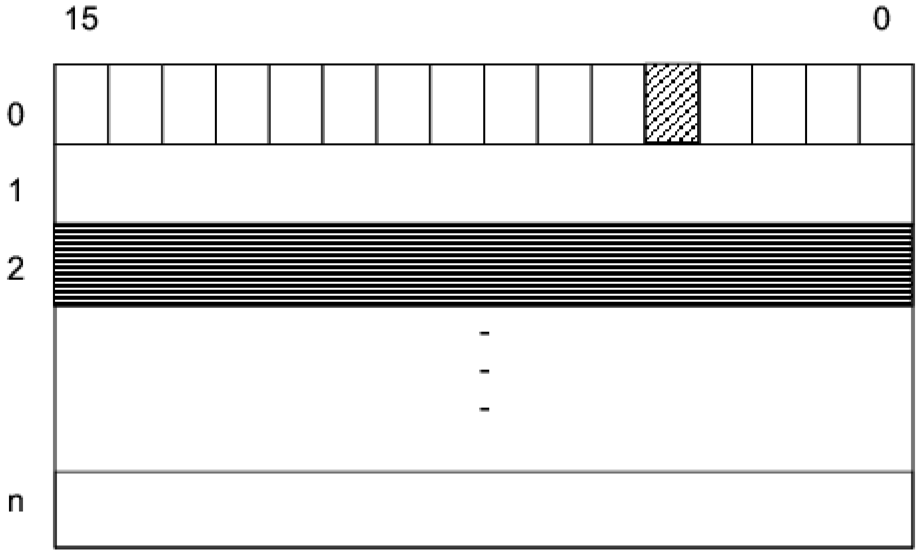


Fig. 2.1 Robot digital outputs (Rack 96 Slot 1)

In Fig. 2.1, the diagonally-striped output point represents the output returns from a Read Input Discrete 4 request. The horizontal-striped portion of the data map represents the 16-point register that would be returned on a Read Input Register 2 request.

2.2.3 Coils and Holding Registers

Coils are accessible starting at address 0. For example, "write coil 4" would access the fifth Robot Digital Input point, as seen in the diagonally-striped area shown in Fig. 2.2.

Holding Registers are also accessible starting at address 0. For example, "write register 2" would access the third set of 16 Digital Input points (DI [33]-DI [48]), as seen in the horizontally-striped area shown in Fig. 2.2. When writing Holding Registers, Coils are written in big-endian format with the Input points numbered from the least significant bit (right hand side) to the most significant bit (left hand side), also shown in Fig. 2.2.

Having these two tables overlaid means that writing coils 0 through 15 would effectively be the same as writing holding register 0. In addition, writing coils 16 through 31 would effectively be the same as writing input register 1, and so on.

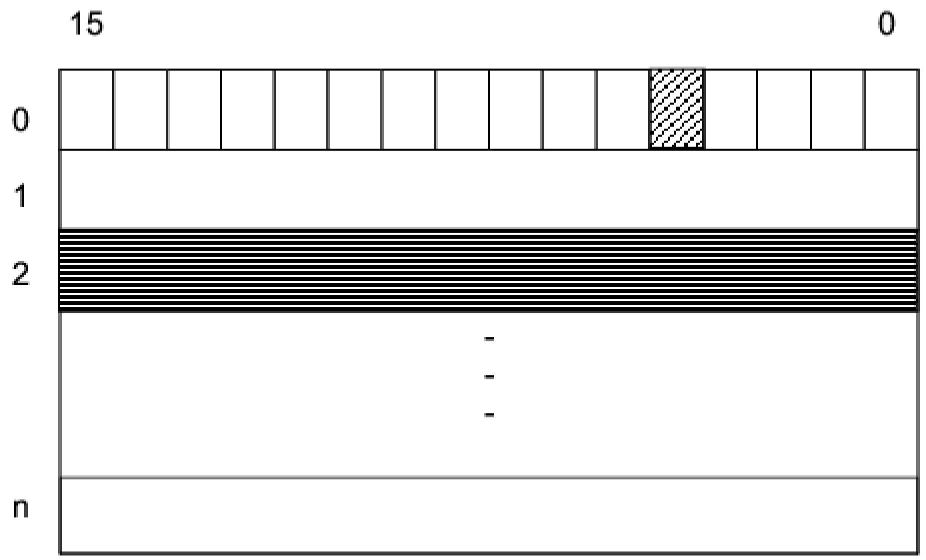


Fig. 2.2 Robot digital inputs (Rack 96 Slot 1)

In Fig. 2.2, the diagonal-striped input point represents the input written in a Write Coil 4 request. The horizontal-striped portion of the data map represents the 16-point register that would be written on a Write Register 2 request.

Robot Digital Outputs are also accessible as read-only Modbus Holding Registers, starting at address 10000. For example, “read register 10002” would access the third set of 16 Digital Output points (DO [33]-DO [48]), as seen in the horizontally-striped area shown in Fig. 2.2. When reading Holding Registers, Coils are written in big-endian format with the Output points numbered from the least significant bit (right hand side) to the most significant bit (left hand side), also shown in Fig. 2.3.

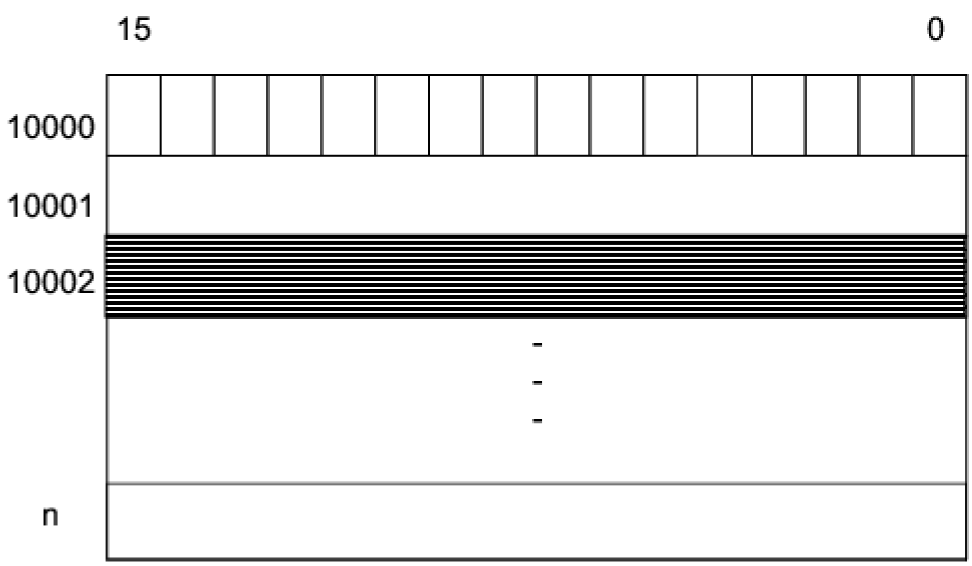


Fig. 2.3 Robot digital outputs (Rack 96 Slot 1) as holding registers

In Fig. 2.3, the horizontally-striped portion of the data map represents the 16-point register that would be returned on a Read Input Register 2 request.

Accessing both Robot Digital Inputs and Digital Output through Holding Registers allows you to take advantage of Modbus function 23: ReadWrite Registers. Through a single Modbus request, a Modbus client can read and write robot digital I/O.

2.2.4 Zero-based Versus One-based Addressing

Modbus addressing is zero-based, and robot addressing is one-based. This can be confusing to some users. For example, when accessing Robot Digital Input 1 (DI [1]), the Modbus client should be configured to access Coil 0.

However, the robot Modbus software also offers a feature to support one-based addressing at the Modbus level. When the system variable $MODBUSTCP.$BASEZERO is set to FALSE, one-based addressing is enforced, in which case the Modbus client should write to Coil 1 to set DI [1] on the robot. Writing to Coil 0 would result in a Modbus Illegal Address error.

All examples in this documentation assume zero-based addressing is being used.

2.3 SUPPORTED MODBUS FUNCTIONS

The robot Modbus TCP server supports the following Modbus functions as shown in Table 2.3.

Table 2.3 Modbus functions

Decimal Code	Hexadecimal Code	Function
01	0x01	Read Coils
02	0x02	Read Discrete Inputs
03	0x03	Read Holding Registers
04	0x04	Read Input Registers
05	0x05	Write Single Coil
06	0x06	Write Single Register
16	0x10	Write Multiple Registers
23	0x17	Read/Write Multiple Registers

2.4 ETHERNET CONNECTION AND IP ADDRESS ASSIGNMENT

The robot must have a valid IP (Internet protocol) address and subnet mask to operate as a Modbus TCP node. Details on the Ethernet interface and TCP/IP configuration can be found in the Internet Options Setup and Operations Manual.

The Ethernet interface supports 10Mbps and 100Mbps baud rates, along with half and full duplex communication. By default, both interfaces will auto-negotiate and should be connected to a switch that supports 100Mbps full duplex connections. If auto-negotiation is configured only on one side (the robot or the switch) and not on the other side, a duplex mismatch is likely to occur causing serious problems.

The LEDs located near the RJ45 connectors on the main CPU board are useful in confirming link establishment (for details on the LEDs, refer to appendix "Diagnostic Information" in the Internet Options Setup and Operations Manual).

The IP address (es) can be configured in the following ways:

- Manually configured on the robot teach pendant – Refer to the "Setting Up TCP/IP" chapter in the Internet Options Setup and Operations Manual.
- DHCP (Dynamic Host Configuration Protocol) – Refer to the "Dynamic Host Configuration Protocol" chapter in the Internet Options Setup and Operations Manual.

NOTE

DHCP is an optional software component. It is important to utilize static or infinite lease IP addresses when using Modbus TCP.

Either one or both Ethernet ports can be configured for use with Modbus TCP. Note that in order to use both ports at the same time they must be properly configured on separate subnets. Refer to the "Setting Up TCP/IP" chapter in the Internet Options Setup and Operations Manual.

NOTE

Be sure that all Modbus TCP node IP addresses are configured properly before you perform the functions in this manual. The PING utility can be used to verify basic communications. Refer to Section 6.3 for more information.

3 MODBUS TCP SERVER CONFIGURATION

3.1 OVERVIEW

The robot supports up to four Modbus TCP connections. These connections are normally to a cell controller or PLC to exchange cell interface I/O data. The Modbus TCP Option must be loaded to support this functionality.

3.2 CONFIGURATION

The Modbus TCP screen is split into two sections: Status and configuration. The Status section includes one field: Slave Status, which is read-only. The Configuration section has six user configurable fields. A power-cycle is required for any changes made to the configuration fields to take effect. Table 3.1 describes all the Modbus TCP screen items. Use Procedure 3-1 to set up a Modbus TCP slave.

Table 3.1 Modbus Screen Items

Field Name	Description
Slave Status	The Slave Status field will be either RUNNING or IDLE. RUNNING indicates that I/O is being exchanged with a Modbus TCP Master, whereas IDLE indicates I/O is currently not being exchanged.
Number of Connections	The Number of Connections field allows you to specify how many Modbus TCP connections the slave can have at one time. This can be set to 0 to completely disable the robot Modbus slave, up to a maximum of 4 connections. Per the Modbus TCP protocol specification, when a new connection request arrives and no unused connections remain, the oldest connection will be closed, and the new connection request will be accepted. The first time this happens after a power-cycle, a PRIO-494 warning alarm will be posted.
Timeout	The Timeout field defines the time for an idle Modbus TCP connection to expire (in milliseconds). If no message is received from any master for this timeout value, the slave device will assume that the network connection has been lost or terminated, will close all connections, and will post a Timeout alarm. Setting this field to 0 disables timeouts.
Error Severity	The Error Severity field defines the severity of Modbus TCP alarms. You can set this to STOP, WARN, or PAUSE by using the F4, [CHOICE] key.
Keep Input on Timeout	The Keep Input on Timeout field specifies how to handle Inputs in the case of a timeout. When set to FALSE, all Modbus Inputs will be set to zero when a timeout occurs. Otherwise, inputs will be left in their last state.
Input Words	The Input Words field specifies how much Digital Input to allocate. In this context a word would be 16-bits. So, if 4 words were configured, 64 points of Digital Input would be allocated to Rack 96, Slot 1. All master devices connected to the robot Modbus slave would have read-write access to this Digital Input data.
Output Words	The Output Words field specifies how much Digital Output to allocate. In this context a word would be 16-bits. So, if 4 words were configured, 64 points of Digital Output would be allocated to Rack 96, Slot 1. All master devices connected to the robot Modbus slave would have read access to this Digital Output data.

Procedure 3-1 Setting Up a Modbus TCP Slave

1. Press MENUS.
2. Select I/O.
3. Press F1, [TYPE], and select Modbus TCP. You will see a screen similar to the following.

```
Modbus TCP Slave Rack 96
  Status
   Slave Status:       IDLE
  Configuration
   Number of Connections:    1
   Timeout (0 = None):     300
   Error Severity:       STOP
   Keep Input on Timeout: FALSE
   Input Words:           4
   Output Words:          4
```

4. Select each of the following and set them as desired (refer to Table 3.1.
   - Number of Connections
   - Timeout
   - Error Severity
   - Keep Input on Timeout
   - Input Words
   - Output Words

5. Turn off the controller, then turn it on again.

3.3 PRIORITY CONNECTIONS

The Modbus TCP server supports Priority connections. Up to two connections can be marked as Priority Connections by specifying the Modbus TCP clients' IP address. This means no other Modbus TCP client device can connect to the robot and use a Priority connection except the device with the corresponding IP address. Moreover, the robot Modbus TCP server will never close a Priority connection unless the connection is first closed by the Modbus TCP client, or in the case of a time-out.

Any Modbus TCP client device can connect to a Non-Priority connection--a connection not marked as a priority connection. If all available non-priority connections are used, and another Modbus TCP client attempts to make a connection to the Non-Priority pool, the oldest existing Non-Priority connection will be closed, and the new connection attempt will be honored. The first time this happens after a power-cycle, a PRIO-494 warning alarm will be posted.

By default, all connections are Non-Priority. Priority connections can be configured by setting one or both of the following system variables with either a valid IP address or hostname:

- $MODBUSTCP.$PRIO1
- $MODBUSTCP.$PRIO2

For example, suppose we had a PLC with an IP address of 192.168.1.11, and a HMI device at 192.168.1.22. We might want the PLC connection to be marked as a priority connection, while the HMI device can connect using a non-priority connection. We would set the "Number of Connections" field to be 2. We would also set $MODBUSTCP.$PRIO1 = "192.168.1.11". Lastly, we would set the "Timeout" field as appropriate based on the faster scan rate of either the PLC or HMI device.

In the example above, two Modbus TCP connections were configured. One of the two was marked as a Priority connection for the PLC. Since only one Priority connection was configured, the second connection defaulted to a Non-Priority connection and can be used by any Modbus TCP client, including the HMI device.

3.4 CONFIGURING THE REMOTE MODBUS TCP CLIENT

The robot will be ready to accept Modbus TCP client connections when the Number of Connections field on the Modbus TCP screen is set to a value of 1 or greater. Use Procedure 3-2 to configure the Schneider Premium PLC. For other Modbus TCP clients, refer to their configuration software documentation.

Procedure 3-2 Configuring the Modbus TCP Client Using PL7 PRO

1. Double-click on the Ethernet module in the Hardware Configuration and display the Messaging tab. See Fig. 3.1.

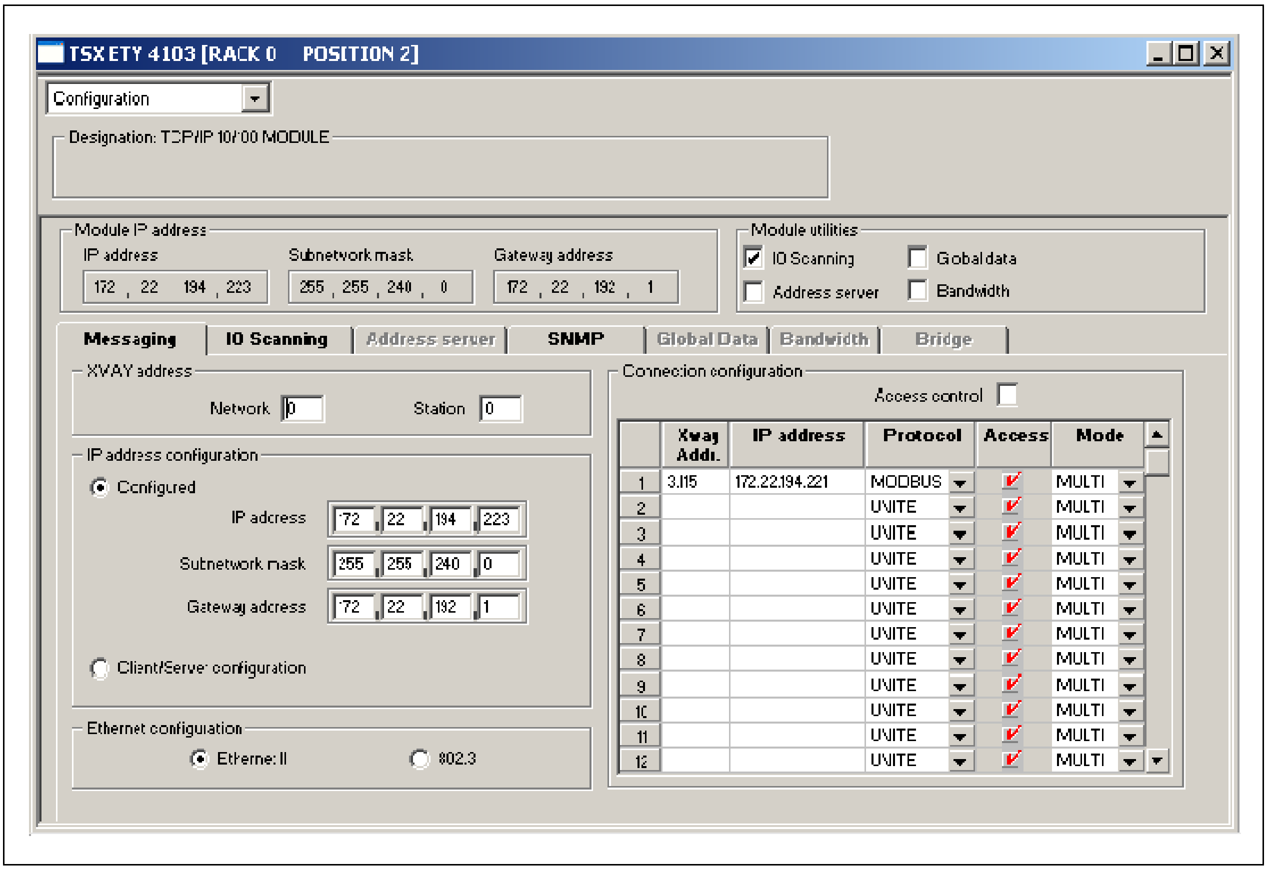


Fig. 3.1 Ethernet module

2. Configure the Xway address for your connection in the Connection Configuration section. See Fig. 3.2.

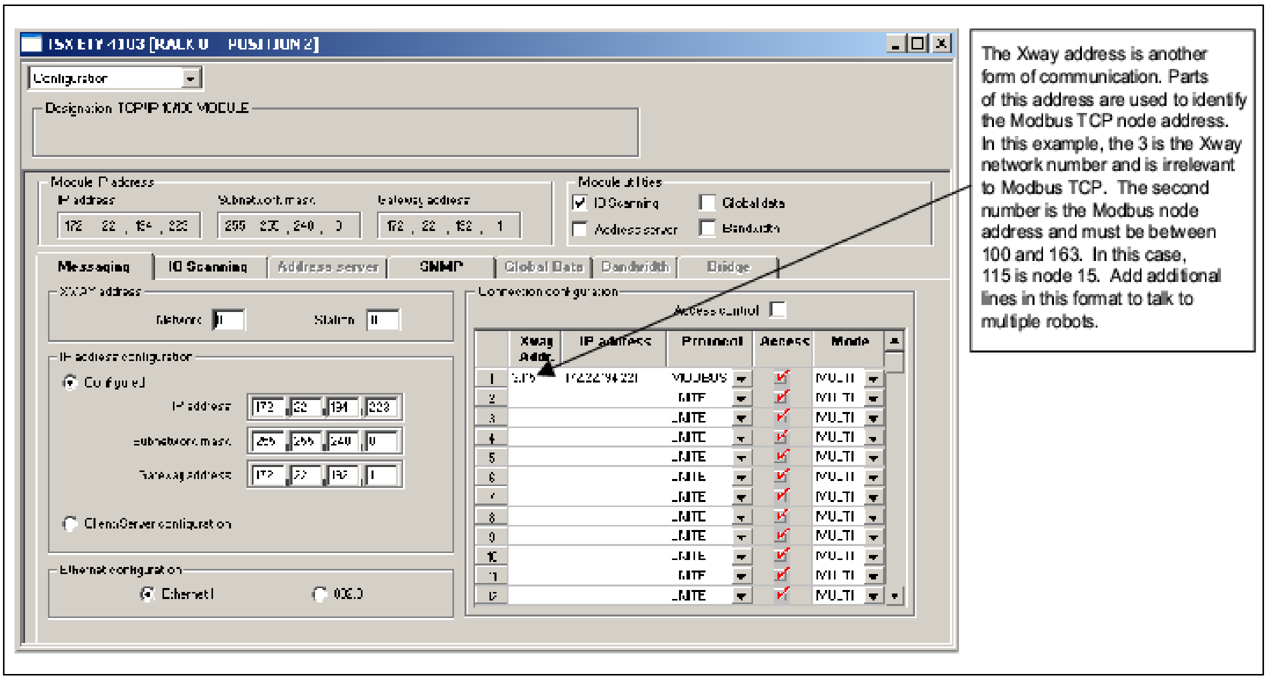


Fig. 3.2 Xway address

3. Enter the IP address of the robot in the IP address field in the Connection Configuration section. See Fig. 3.3.

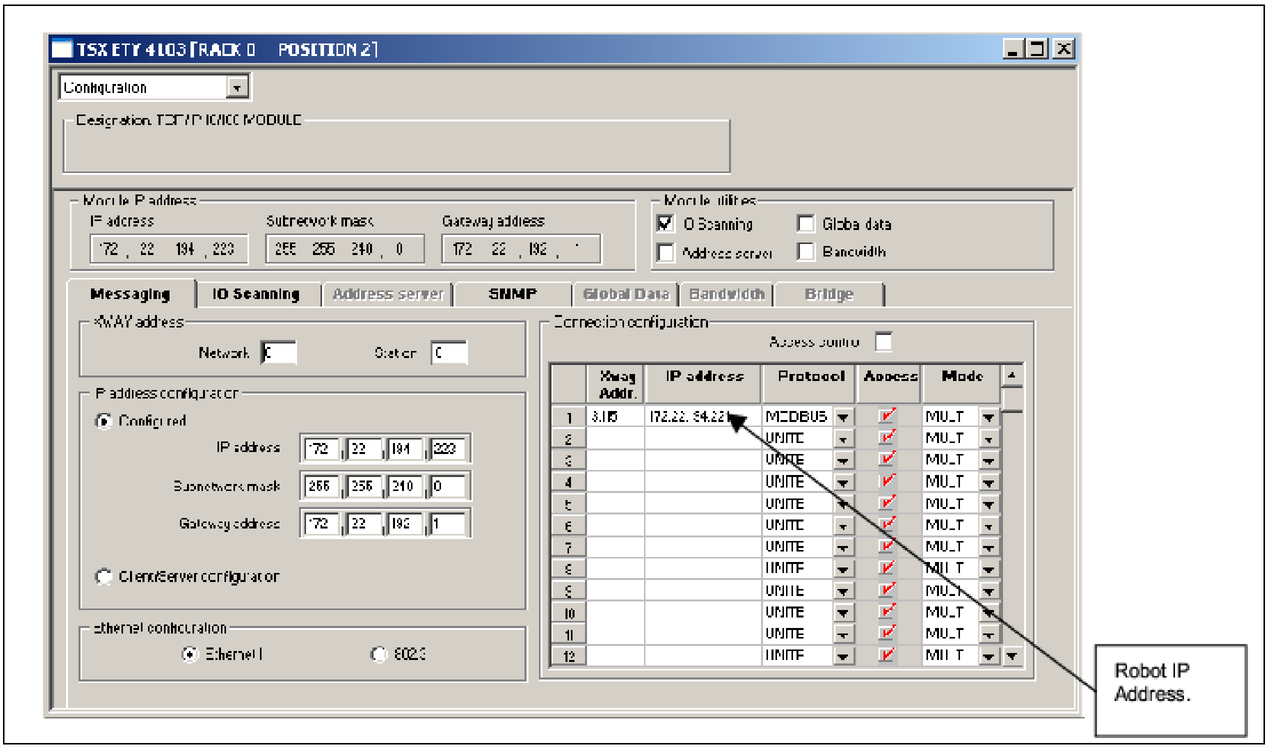


Fig. 3.3 I/P address

4. Set the protocol to MODBUS in the Protocol field in the Connection Configuration section. See Fig. 3.4.

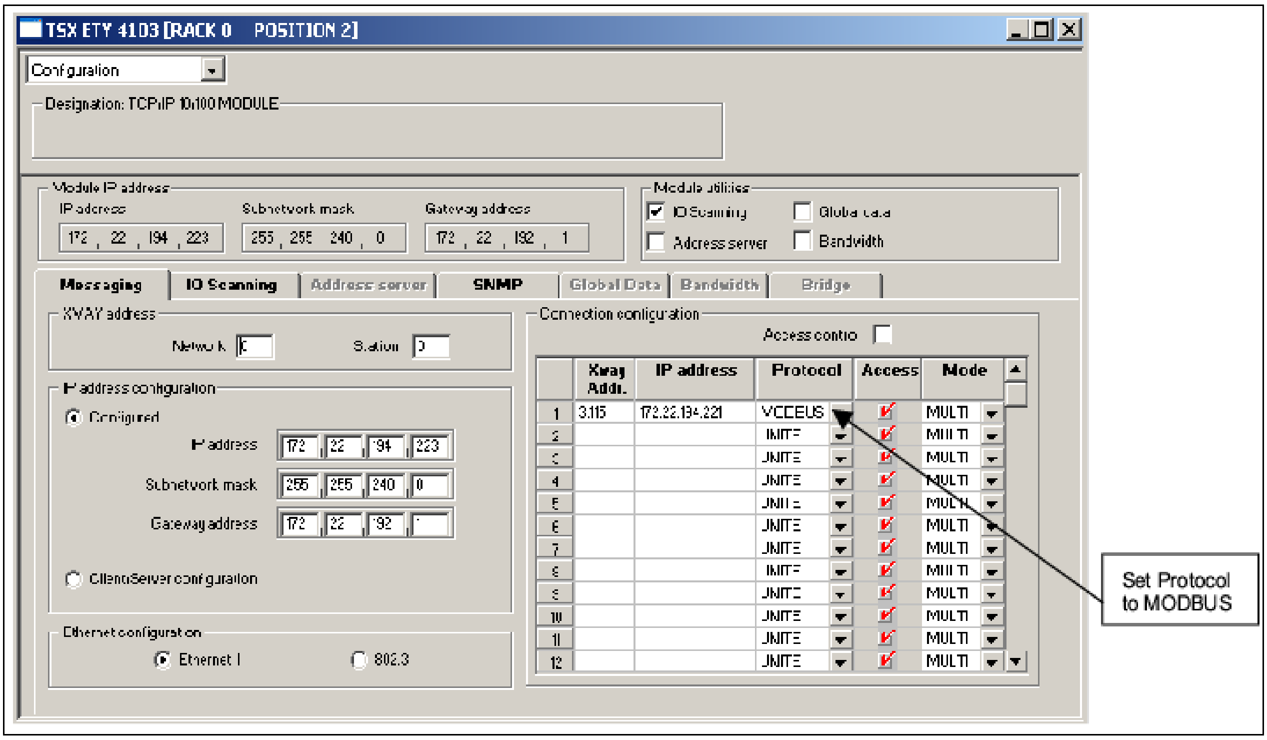


Fig. 3.4 Protocol

5. Go to the IO Scanning Tab. See Fig. 3.5.

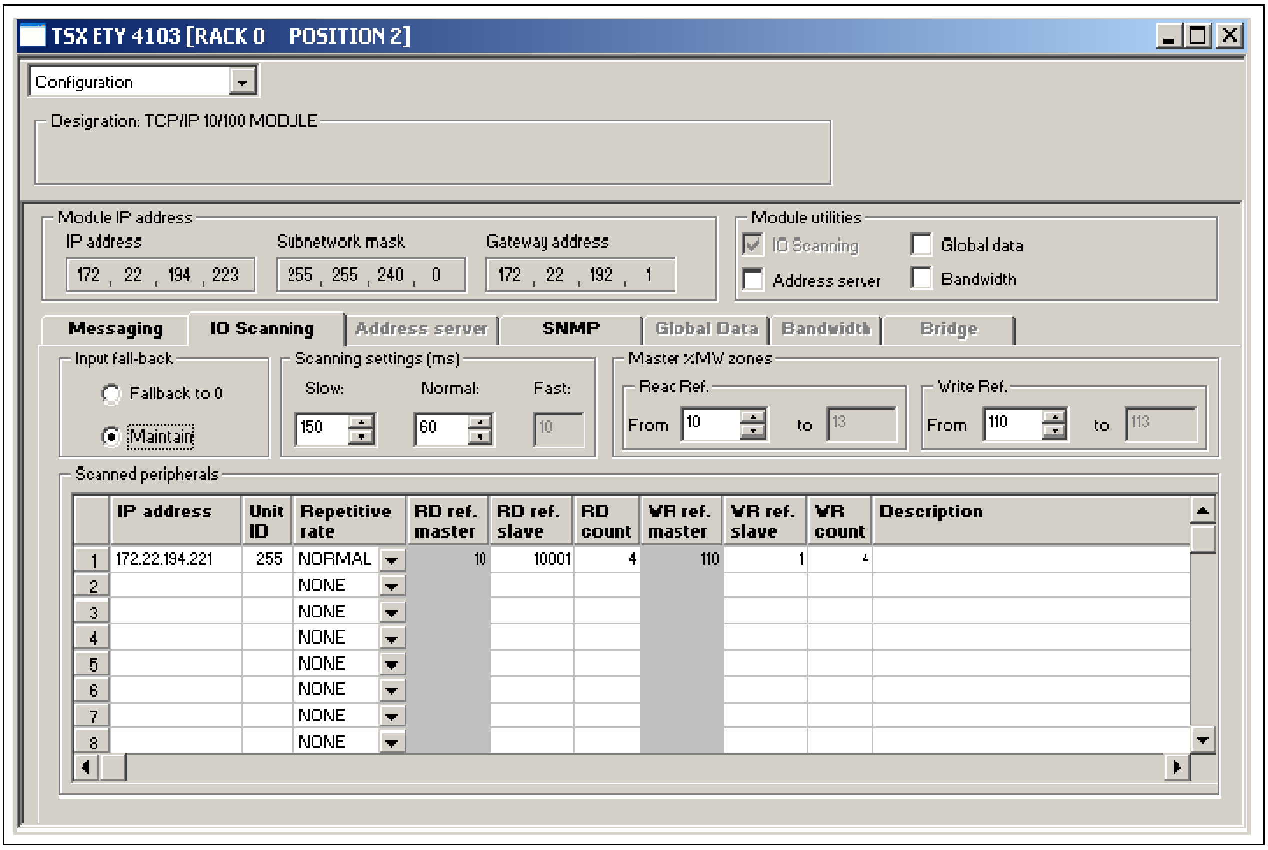


Fig. 3.5 I/O scanning

6. Set the RD ref slave to the first Holding Registers address of robot Digital Output (DO [1]). This will be 10000 if zero-based addressing is used, or 10001 for one-based addressing. This number should correspond with the configuration of Output Words on the robot Modbus TCP screen. See Fig. 3.6.

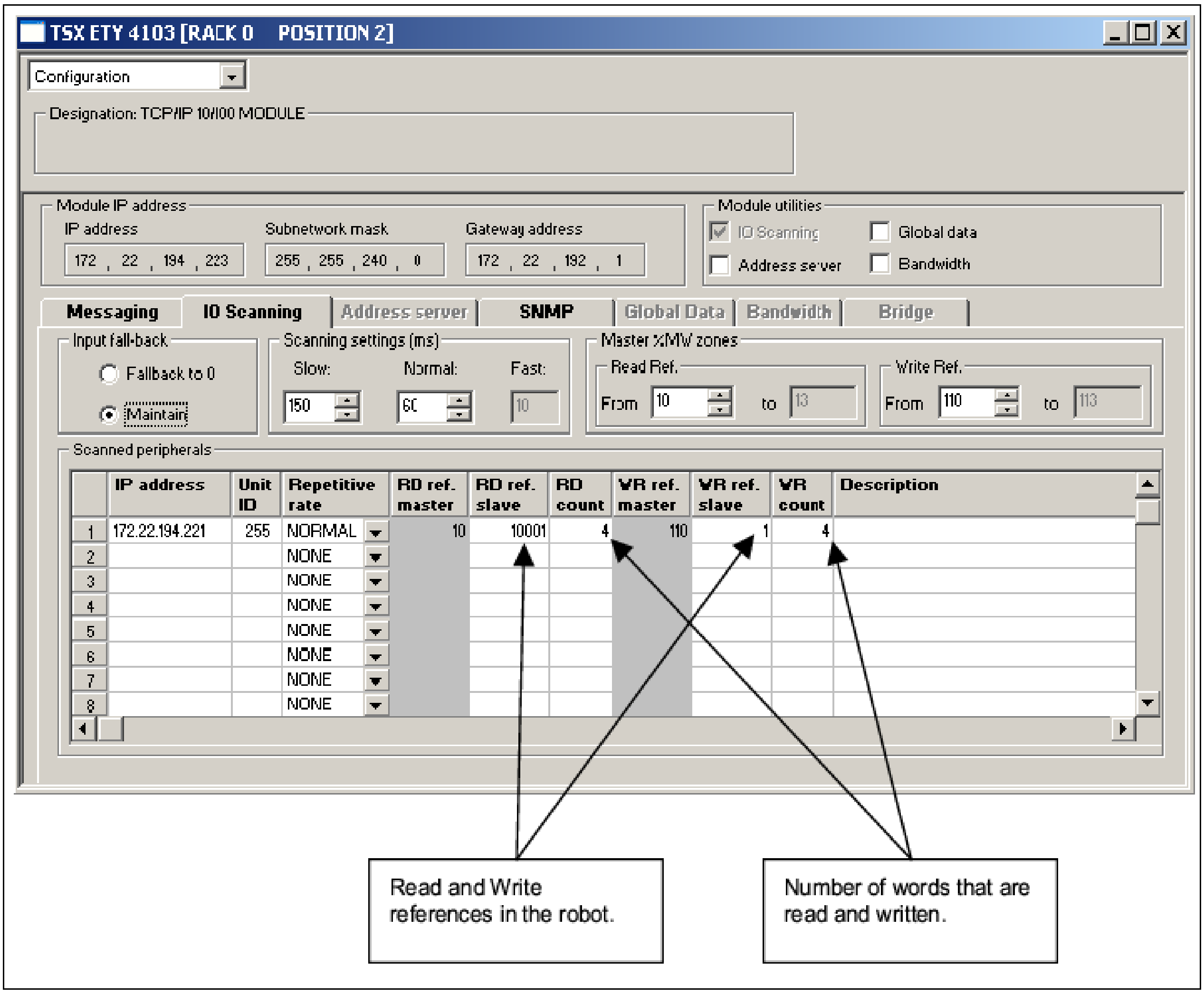


Fig. 3.6 RD slave

7. Set the RD count to the number Holding Registers to be read off the robot.

8. Set the WR ref slave to the first Holding Registers address of robot Digital Input (DI [1]). This will be 0 if zero-based addressing is used, or 1 for one-based addressing.

9. Set the RD count to the number Holding Registers to be written to the robot. This number should correspond with the configuration of Input Words on the robot Modbus TCP screen.

10. Set the Master %MW zones as appropriate for you PLC configuration. See Fig. 3.7.

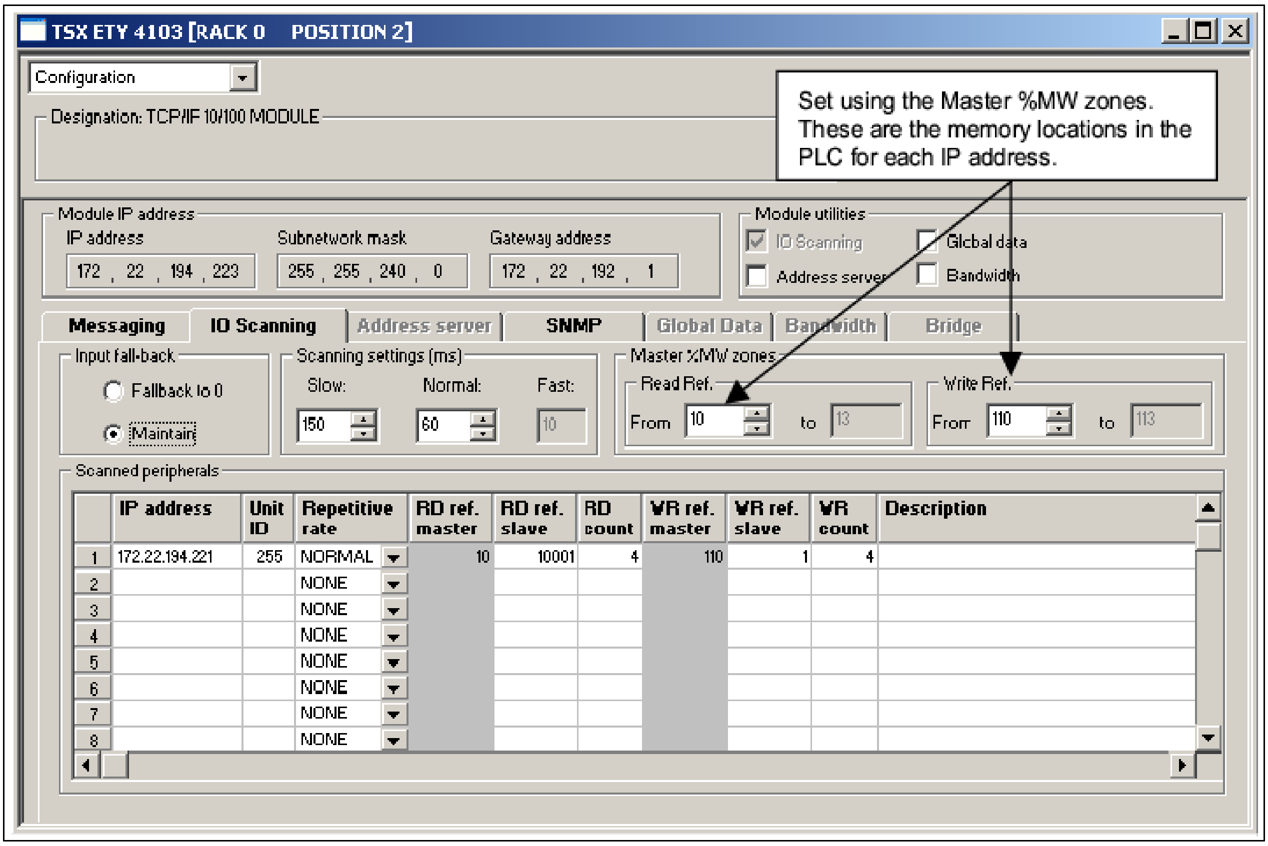


Fig. 3.7 Master %MW zones

11. Set the Repetitive rate to NORMAL to enable communication. See Fig. 3.8.

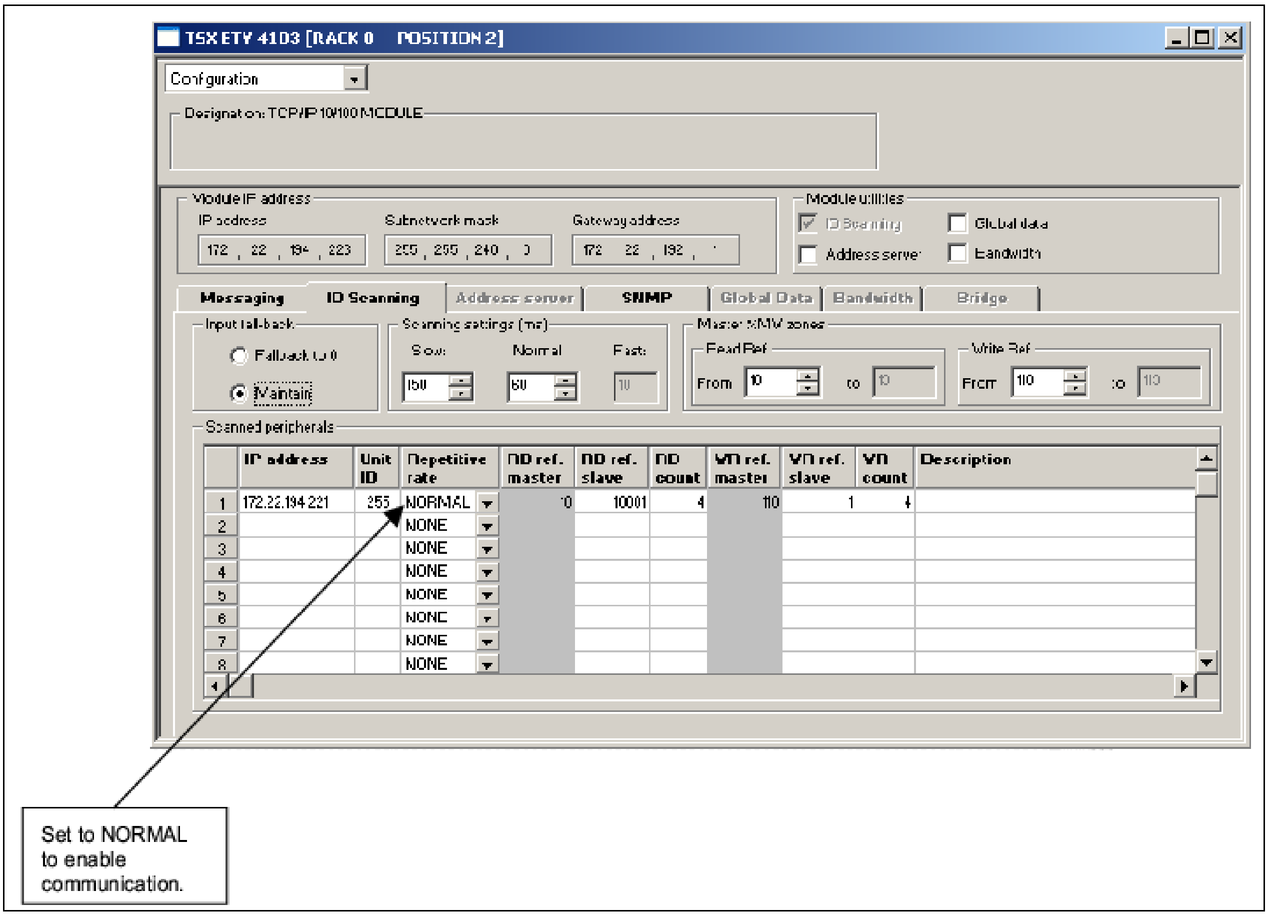


Fig. 3.8 Repetitive rate

4 I/O CONFIGURATION

4.1 OVERVIEW

This chapter describes how to make Modbus TCP I/O available within the robot by mapping it to digital, and group. It also describes procedures for backing up and restoring the Modbus TCP and I/O configurations.

4.2 MAPPING I／O ON THE ROBOT

The Modbus TCP I/O can be mapped to digital or group I/O. This is similar to mapping other I/O points on the robot where the rack, slot, and starting point number are used to map physical I/O to logical I/O within the I/O map. Modbus TCP I/O uses rack 96, Slot 1.
Use Procedure 4-1 to Map I/O on the Robot.

Procedure 4-1 Mapping I/O on the Robot

1. Press MENUS.
2. Select I/O.
3. Press F1, [TYPE], and select Digital, or Group.
4. Press F2, CONFIG.
5. Set the Range to the appropriate value.
6. Set the Rack to 96, set the slot number to 1, and set the starting point as required.

NOTE
Refer to the Input/Output (I/O) Setup chapter in the application-specific Setup and Operations Manual for additional information on I/O configuration.

In some application software, the I/O is automatically configured when you turn on the controller. The system variable $IO_AUTO_CFG (for digital I/O) controls this behavior. If the system has already automatically configured the I/O, and sizes are changed, the I/O assignments can be cleared to force the system to re-map all the I/O. This is done by clearing assignments (CLR_ASG). Use Procedure 4-2 to clear I/O assignments.

Procedure 4-2 Clearing I/O Assignments

1. Press MENUS.
2. Select I/O.
3. Press F1, [TYPE], and select Link Device.
4. Press F5, CLR_ASG.
5. To remap all I/O, turn the controller off and then back on.

NOTE
This clears ALL I/O assignments. The I/O will be remapped when you turn off then turn on the controller based on the settings of $IO_AUTO_CFG (for digital I/O) and
$IO_AUTO_UOP (for UOP I/O).

4.3 BACKING UP AND RESTORING MODBUS TCP

There are two files, which contain information on the configuration of Modbus TCP and I/O mappings:
DIOCFGSV.IO contains general I/O configuration and all I/O mappings (for example, mappings between Modbus TCP and digital, and group).
SYSMBUS.SV contains Modbus TCP specific configuration.
Use Procedure 4-3 to back up files manually.
Use Procedure 4-4 to do a full application backup, which includes DIOCFGSV.IO and SYSMBUS.SV.

Procedure 4-3 Backing Up Files Manually

1. Select the default file device where files will be saved:
   - **a.** Press MENUS.
   - **b.** Select File.
   - **c.** Press F5, [UTIL], and choose SET DEVICE.
   - **d.** Select the device to which you want to save the files.

2. Save DIOCFGSV.IO.
   - **a.** Press MENUS.
   - **b.** Select I/O.
   - **c.** Press F1, [TYPE], and choose DIGITAL.
   - **d.** Press FCTN.
   - **e.** Select Save to save DIOCFGSV.IO to the default device.

3. Save SYSMBUS.SV:
   - **a.** Press MENUS.
   - **b.** Select I/O.
   - **c.** Press F1, [TYPE], and choose Modbus TCP.
   - **d.** Press FCTN.
   - **e.** Select Save to save SYSMBUS.SV to the default device.

Procedure 4-4 Performing a Full Application Backup

1. Select the default file device (where files will be saved):
   - **a.** Press MENUS.
   - **b.** Select File.
   - **c.** Press F5, [UTIL], and choose SET DEVICE.
   - **d.** Select the device to which you want to save the files.

2. Press F4, [BACKUP], and choose “All of above”.

5 MODBUS TCP GUIDELINES

Good network design is critical for reliable operation. It is important to pay special attention to wiring guidelines and environmental conditions affecting the cable system and equipment. It is also necessary to control network traffic to avoid wasted network bandwidth and device resources.
Keep in mind the following wiring guidelines and environmental considerations:

- Use category 5 twisted pair (or better) rated for 100-BaseTX Ethernet applications and the application environment. Consider shielded versus unshielded twisted pair cabling.
- Pay careful attention to wiring guidelines such as maximum length from the switch to the device (100 meters).
- Do not exceed recommended bending radius of specific cabling being used.
- Use connectors appropriate to the environment. There are various industrial Ethernet connectors in addition to the standard open RJ45 that should be used where applicable. For example, connectors are available with IP65 or IP67 ratings.
- Route the wire runs away from electrical or magnetic interference or cross at ninety degrees to minimize induced noise on the Ethernet network.

Keep the following in mind as you manage network traffic:

- Control or eliminate collisions by limiting the collision domain. ·
- Control broadcast traffic by limiting the broadcast domain.
- Control multicast traffic with intelligent routing.
- Use QOS (Quality of Service) techniques in very demanding applications.

Collisions are a traditional concern on an Ethernet network but can be completely avoided by using switches—rather than hubs—and full duplex connections. It is critical to use switches and full duplex connections for any Ethernet I/O network, because it reduces the collision domain to only one device so that no collisions will occur. The robot interface will auto-negotiate by default and use the fastest connection possible. Normally this is 100Mbps and full duplex. The robot can be set for a specific connection speed and duplex (refer to the chapter titled “Setting Up TCP/IP” in the Internet Options Setup and Operations Manual).
However, be very careful that both ends of the connection use the same speed and duplex mode, or be sure both ends are set to auto-negotiate. If auto-negotiation is configured on only one end, a duplex mismatch is likely to occur causing serious problems.

The LEDs near the RJ45 connector on the robot will confirm a connection link (refer to the appendix titled “Diagnostic Information” in the Internet Options Setup and Operations Manual for details on the LEDs). Link State can be confirmed using the TCP/IP status Host Comm screen by following Procedure 5-1.

Procedure 5-1 Verifying Link State

1. Press MENUS.
2. Select Setup.
3. Press [F1] TYPE and select Host Comm.
4. Select TCP/IP.
5. Toggle to the correct port (port #1 or port #2) by pressing [F3] PORT.
6. Press NEXT, then [F2] STATUS.

Broadcast traffic is traffic that all nodes on the subnet must listen for and in some cases respond to. Excessive broadcast traffic wastes network bandwidth and wastes resources in all affected nodes. The broadcast domain is the range of devices (typically the entire subnet) that must listen to all broadcasts. FANUC Robotics recommends limiting the broadcast domain to only the control devices (for example, Modbus TCP nodes) by using a separate subnet for the control equipment or by using VLANs (virtual LANs) supported by some higher end switches. If the Modbus TCP network is completely isolated as a separate control network, this is not a concern. However, when connecting into larger networks, this becomes important.

Quality of Service (QOS) techniques provides mechanisms to prioritize network traffic. Generally on an Ethernet network, all packets are equal. Packets can be dropped or delayed within network infrastructure equipment (for example, switches) in the presence of excessive traffic. Which packets are dropped or delayed is random. QOS is a term covering several different approaches to prioritizing packets including:

- MAC layer (layer 2) prioritization (IEEE 802.1p).
- IP layer (layer 3) prioritization using source/destination IP addresses.
- Transport layer (layer 4) prioritization using source/destination ports.

These QOS mechanisms are generally implemented within the network infrastructure equipment and are beyond the scope of this manual. Some form of QOS should be considered on complex networks requiring the highest possible level of determinism in I/O exchanges within the control network.

- 100 Mbps baud rate
- Full duplex connections
- Port auto-negotiation · Environmental specifications appropriate for the application (for example, temperature)
- Power supply requirements and redundancy (for example, support for 24vdc or 120vac and support for a second redundant power supply if warranted)

NOTE
If there is a significant amount of multicast traffic on the network, the switch should support IGMP snooping (multicast aware).

NOTE
If the control network will be part of a larger network, the control network should be on a separate VLAN or subnet. This can be done within the control switch or possibly based on how the larger network connects to the control switch.

Some examples of switch products are:

- RJLinx ENHSAURR8, 8 port unmanaged industrial switch
- Telesyn AT-FS716, 16 port unmanaged switch
- Cisco 2950-12, 12 port managed switch
- Cisco 2955 (industrialized version of 2950)
- Hischmann MICE (modular industrial switch)
- Phoenix Contact (managed/unmanaged industrial switch)
- Interlink-BT SE-84x-E524 (5 port managed/unmanaged industrial switch)

6 DIAGNOSTICS AND TROUBLESHOOTING

6.1 OVERVIEW

There are two basic tools for verifying network connections:

- Ethernet status LEDs
- PING

The LEDs and PING utility are basic tools but they give a good indication of whether or not devices are able to communicate on the network. If the LINK LED is off, or if PING times out, then no other network functionality will work for that device.
Refer to Section 6.2 for more information about Ethernet status LEDs.
Refer to Section 6.3 for more information about the PING utility.

6.2 ETHERNET STATUS LEDS

The Ethernet status LEDs at the Ethernet RJ45 connector on the robot will indicate if the robot is connected. Most Ethernet switches and other equipment will have similar LEDs indicating a physical connection. If the LINK LED is off then there is no Ethernet connectivity at all. This generally implies a disconnected or bad cable or bad connections. The speed and duplex must match between the robot and the switch. For more information about the Ethernet status LEDs, refer to the appendix titled “Diagnostic Information” in the Internet Options Setup and Operations Manual. Details on auto-negotiating and manually setting speed and duplex level can be found in the chapter titled “Setting Up TCP/IP” in the of the Internet Options Setup and Operations Manual. The robot will auto-negotiate by default and should not be changed in most cases.

6.3 PING UTILITY

PING is a network utility that sends a request to a specific IP address and expects a response. The request is essentially “Can you hear me?” The destination node will send a response that it received the request. The requesting node will receive either the response or timeout. PING is a basic network utility that is included with most operating systems, such as Linux, Unix, and Windows and is also supported on the robot. Even devices that do not support generating PING requests (for example, a Modbus TCP block with no user interface) will respond to the PING request.

The PING utility is also available on the robot to PING any name or IP address. Use Procedure 6-1.

The PING utility is also available from any Windows PC. Use Procedure 6-2.

Procedure 6-1 Using PING on the Robot

1. Press MENUS.
2. Select Setup
3. Press F1, [TYPE].
4. Select Host Comm.
5. Move the cursor to select PING in the Protocol List and press ENTER.
6. Enter the name or IP address of the node to PING.
7. Press F2, PING.

NOTE

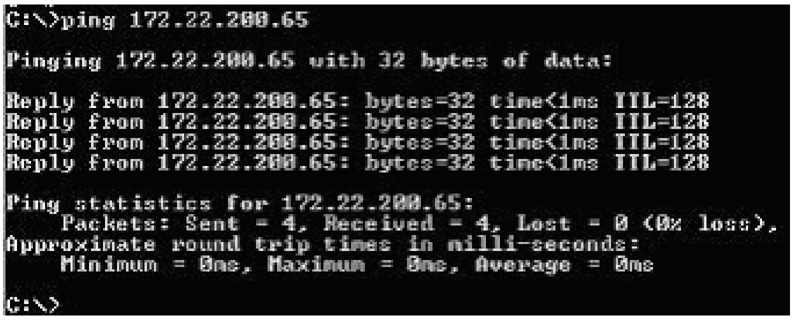
The prompt line on the teach pendant will indicate if the PING was successful or if the PING request timed out.

Procedure 6-2 Using PING on a Windows PC

1. Open a DOS command prompt.
2. Type the following command, replacing the IP address with the IP address you want to PING, and press ENTER. See the following screen for an example.

```
PING 192.168.0.10
```

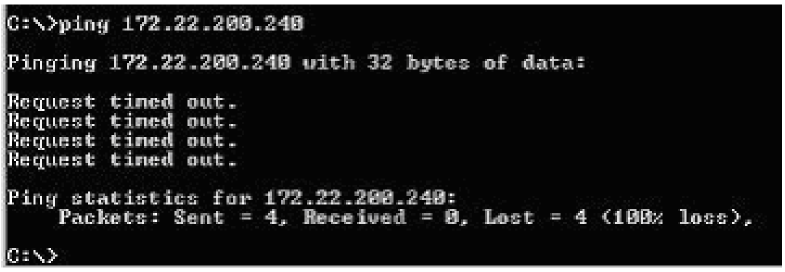
The following image shows a successful PING.



```
C:\>ping 172.22.200.65

Pinging 172.22.200.65 with 32 bytes of data:

Reply from 172.22.200.65: bytes=32 time<1ms TTL=128
Reply from 172.22.200.65: bytes=32 time<1ms TTL=128
Reply from 172.22.200.65: bytes=32 time<1ms TTL=128
Reply from 172.22.200.65: bytes=32 time<1ms TTL=128

Ping statistics for 172.22.200.65:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

Fig. 6.1 Successful PING

The following image shows an unsuccessful PING.



```
C:\>ping 172.22.200.240

Pinging 172.22.200.240 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 172.22.200.240:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

Fig. 6.2 Unsuccessful PING

INDEX

Revision Record

FANUC Robot R-30*i*A/R-30*i*A Mate CONTROLLER Modbus TCP Interface OPERATOR'S MANUAL (B-82844EN)

Edition	Date	Contents	Edition	Date	Contents
01	Feb., 2008	——			

B-82844EN/01

B-82844EN/01