# FANUC Robot series

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

Ethernet Function

OPERATOR'S MANUAL

B-82974EN/02

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

- No part of this manual may be reproduced in any form.
- All specifications and designs are subject to change without notice.

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 PRECAUTIONS

Thank you for purchasing FANUC Robot.
This chapter describes the precautions which must be observed to ensure the safe use of the robot.
Before attempting to use the robot, be sure to read this chapter thoroughly.

Before using the functions related to robot operation, read the relevant operator's manual to become familiar with those functions.

If any description in this chapter differs from that in the other part of this manual, the description given in this chapter shall take precedence.

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.
In addition, refer to the “FANUC Robot SAFETY HANDBOOK (B-80687EN)”.

## 1 WORKING PERSON

The personnel can be classified as follows.

Operator:
- Turns robot controller power ON/OFF
- Starts robot program from operator’s panel

Programmer or teaching operator:
- Operates the robot
- Teaches robot inside the safety fence

Maintenance engineer:
- Operates the robot
- Teaches robot inside the safety fence
- Maintenance (adjustment, replacement)

- An operator cannot work inside the safety fence.
- A programmer, teaching operator, and maintenance engineer can work inside the safety fence. The working activities inside the safety fence include lifting, setting, teaching, adjusting, maintenance, etc.
- To work inside the fence, the person must be trained on proper robot operation.

During the operation, programming, and maintenance of your robotic system, the programmer, teaching operator, and maintenance engineer should take additional care of their safety by using the following safety precautions.

- Use adequate clothing or uniforms during system operation
- Wear safety shoes
- Use helmet

# 2 DEFINITION OF WARNING, CAUTION AND NOTE

To ensure the safety of user and prevent damage to the machine, this manual indicates each precaution on safety with "Warning" or "Caution" according to its severity. Supplementary information is indicated by "Note". Read the contents of each "Warning", "Caution" and "Note" before attempting to use the oscillator.

WARNING
Applied when there is a danger of the user being injured or when there is a danger of both the user being injured and the equipment being damaged if the approved procedure is not observed.

CAUTION
Applied when there is a danger of the equipment being damaged, if the approved procedure is not observed.

NOTE
Notes are used to indicate supplementary information other than Warnings and Cautions.

- Read this manual carefully, and store it in a sales place.

# 3 WORKING PERSON SAFETY

Working person 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 working person safety.

(1) Have the robot system working persons 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 in a ready to move state, and is waiting for a signal. In this state, the robot is regarded as still in motion. To ensure working person 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 working person can enter the work area without passing through the gate. Install an interlocking device, a safety plug, and so forth in the safety gate so that the robot is stopped as the safety gate is opened.

The controller is designed to receive this interlocking signal of the door switch. When the gate is opened and this signal received, the controller stops the robot (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type). For connection, see Fig.3 (a) and Fig.3 (b).

(4) Provide the peripheral devices with appropriate grounding (Class A, Class B, Class C, and Class D).
(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 a working person enters the work area.
(8) If necessary, install a safety lock so that no one except the working person 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
(10) Operators should be ungloved while manipulating the operator's panel or teach pendant. Operation with gloved fingers could cause an operation error.
(11) Programs, system variables, and other information can be saved on memory card or USB memories. Be sure to save the data periodically in case the data is lost in an accident.
(12) The robot should be transported and installed by accurately following the procedures recommended by FANUC. Wrong transportation or installation may cause the robot to fall, resulting in severe injury to workers.
(13) In the first operation of the robot after installation, the operation should be restricted to low speeds. Then, the speed should be gradually increased to check the operation of the robot.
(14) Before the robot is started, it should be checked that no one is in the area of the safety fence. At the same time, a check must be made to ensure that there is no risk of hazardous situations. If detected, such a situation should be eliminated before the operation.
(15) When the robot is used, the following precautions should be taken. Otherwise, the robot and peripheral equipment can be adversely affected, or workers can be severely injured.
- Avoid using the robot in a flammable environment.
- Avoid using the robot in an explosive environment.
- Avoid using the robot in an environment full of radiation.
- Avoid using the robot under water or at high humidity.
- Avoid using the robot to carry a person or animal.
- Avoid using the robot as a stepladder. (Never climb up on or hang from the robot.)

(16) When connecting the peripheral devices related to stop(safety fence etc.) and each signal (external emergency , fence etc.) of robot. be sure to confirm the stop movement and do not take the wrong connection.
(17) When preparing trestle, please consider security for installation and maintenance work in high place according to Fig.3 (c). Please consider footstep and safety bolt mounting position.

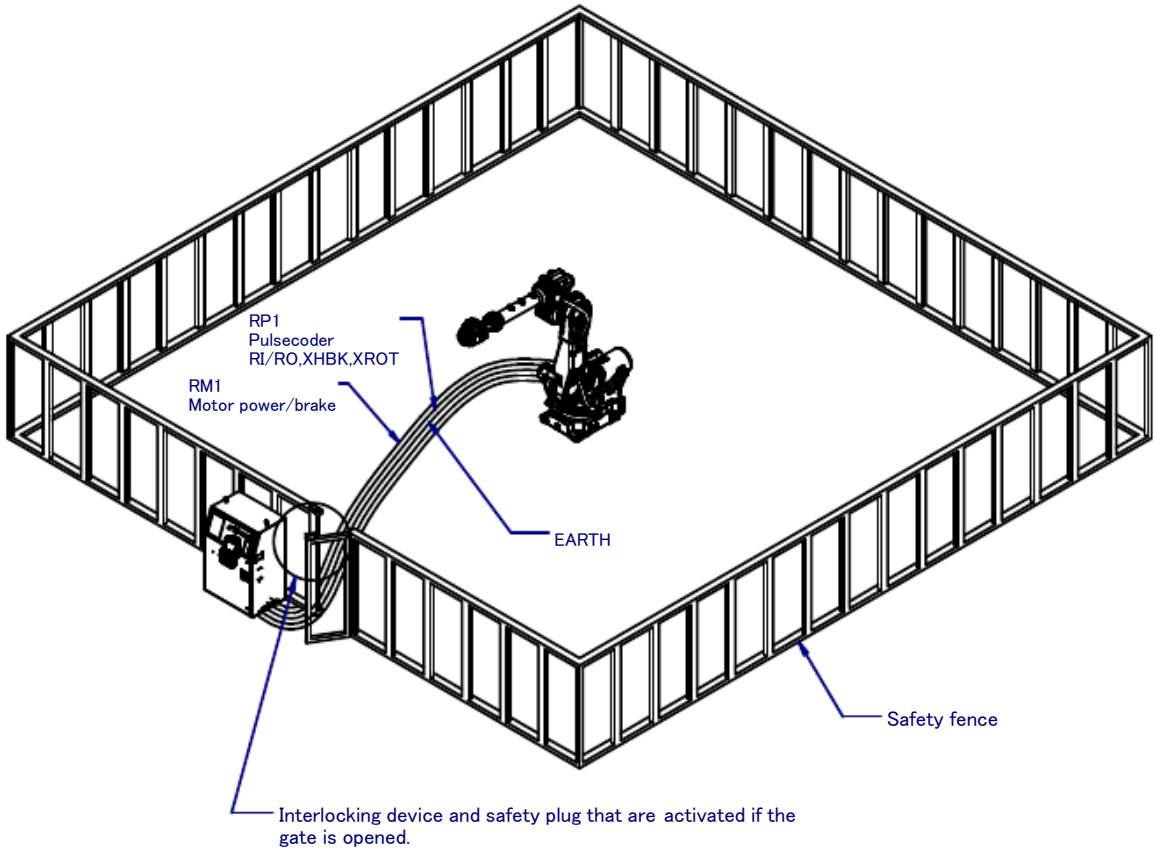



Fig. 3 (a) Safety fence and safety gate

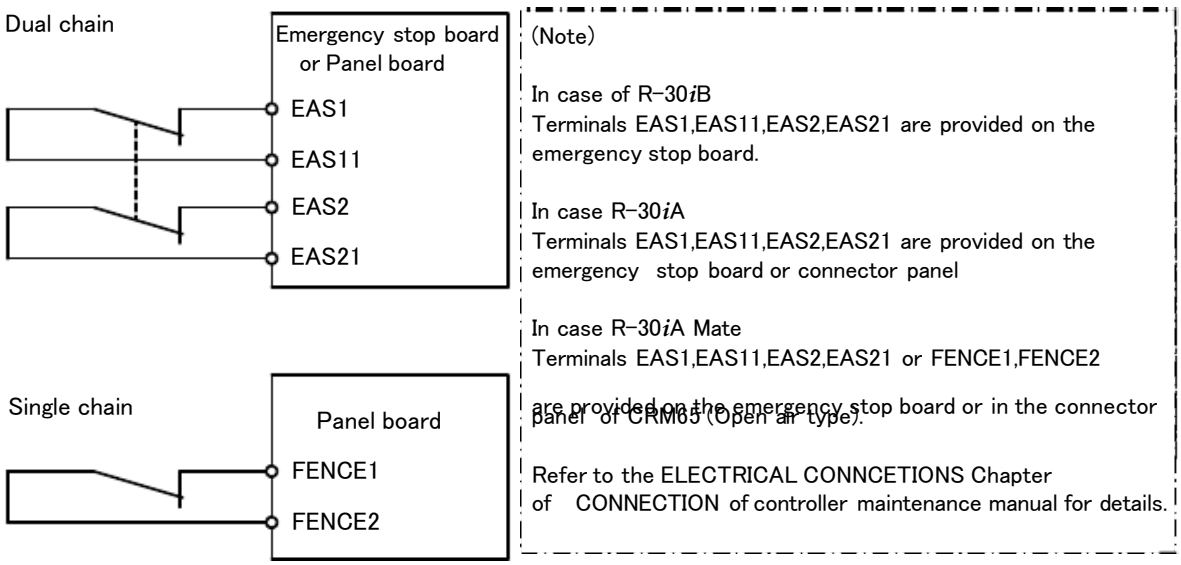



Fig. 3 (b) Limit switch circuit diagram of the safety fence

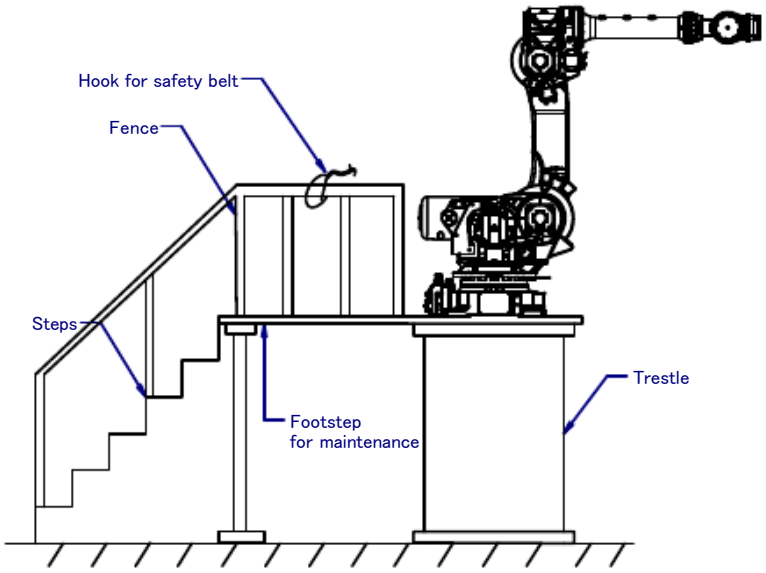



Fig.3 (c) Footstep for maintenance

# 3.1 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 you do not have to operate the robot, 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 of the safety fence.
(3) Install a safety fence with a safety gate to prevent any worker other than the operator from entering the work area unexpectedly and 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 (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type), when the external EMERGENCY STOP button is pressed. See the diagram below for connection.

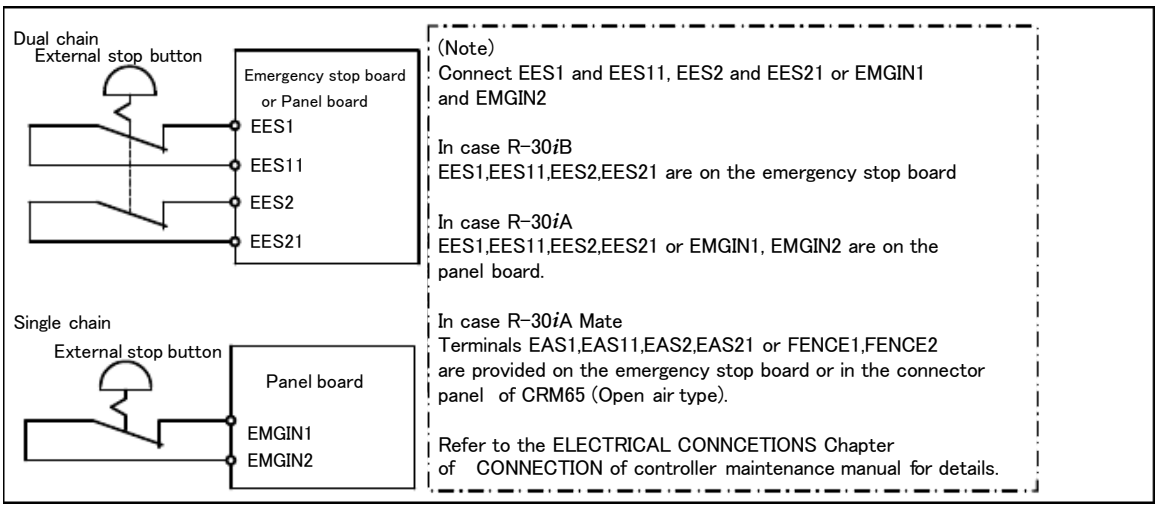



Fig.3.1 Connection diagram for external emergency stop button

# 3.2 SAFETY OF THE PROGRAMMER

While teaching the robot, the operator must enter the work area of the robot. The operator must ensure the safety of the teach pendant operator especially.

(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) If it is inevitable to enter the robot work area to teach the robot, check the locations, settings, and other conditions of the safety devices (such as the EMERGENCY STOP button, the DEADMAN switch on the teach pendant) before entering the area.
(4) The programmer must be extremely careful not to let anyone else enter the robot work area.
(5) Programming should be done outside the area of the safety fence as far as possible. If programming needs to be done in the area of the safety fence, the programmer should take the following precautions:
   - Before entering the area of the safety fence, ensure that there is no risk of dangerous situations in the area.
   - Be prepared to press the emergency stop button whenever necessary.
   - Robot motions should be made at low speeds.
   - Before starting programming, check the entire system status to ensure that no remote instruction to the peripheral equipment or motion would be dangerous to the user.

Our operator panel is provided with an emergency stop button and a key switch (mode switch) for selecting the automatic operation mode (AUTO) and the teach modes (T1 and T2). Before entering the inside of the safety fence for the purpose of teaching, set the switch to a teach mode, remove the key from the mode switch to prevent other people from changing the operation mode carelessly, then open the safety gate. If the safety gate is opened with the automatic operation mode set, the robot stops (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type). After the switch is set to a teach mode, the safety gate is disabled. The programmer should understand that the safety gate is disabled and is responsible for keeping other people from entering the inside of the safety fence. (In case of R-30*i*A Mate Controller standard specification, there is no mode switch. The automatic operation mode and the teach mode is selected by teach pendant enable switch.)

Our teach pendant is provided with a DEADMAN switch as well as an emergency stop button. These button and switch function as follows:

(1) Emergency stop button: Causes an emergency stop (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type) when pressed.
(2) DEADMAN switch: Functions differently depending on the teach pendant enable/disable switch setting status.

   (a) Disable: The DEADMAN switch is disabled.
   (b) Enable: Servo power is turned off when the operator releases the DEADMAN switch or when the operator presses the switch strongly.

   Note) The DEADMAN switch is provided to stop the robot when the operator releases the teach pendant or presses the pendant strongly in case of emergency. The R-30*i*B/R-30*i*A/ R-30*i*A Mate employs a 3-position DEADMAN switch, which allows the robot to operate when the 3-position DEADMAN switch is pressed to its intermediate point. When the operator releases the DEADMAN switch or presses the switch strongly, the robot stops immediately.

The operator's intention of starting teaching is determined by the controller through the dual operation of setting the teach pendant enable/disable switch to the enable position and pressing the DEADMAN switch. The operator should make sure that the robot could operate in such conditions and be responsible in carrying out tasks safely.

Based on the risk assessment by FANUC, number of operation of DEADMAN SW should not exceed about 10000 times per year.

maintenance personnel must indicate that maintenance work is in progress and be careful not to allow other people to operate the robot carelessly.

(4) When entering the area enclosed by the safety fence, the maintenance worker must check the entire system in order to make sure no dangerous situations exist. In case the worker needs to enter the safety area whilst a dangerous situation exists, extreme care must be taken, and entire system status must be carefully monitored.

(5) Before the maintenance of the pneumatic system is started, the supply pressure should be shut off and the pressure in the piping should be reduced to zero.

(6) Before the start of teaching, check that the robot and its peripheral devices are all in the normal operating condition.

(7) Do not operate the robot in the automatic mode while anybody is in the robot work area.

(8) When you maintain the robot alongside a wall or instrument, or when multiple workers are working nearby, make certain that their escape path is not obstructed.

(9) 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.

(10) If necessary, have a worker who is familiar with the robot system stand beside the operator 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.

(11) When replacing a part, please contact FANUC service center. If a wrong procedure is followed, an accident may occur, causing damage to the robot and injury to the worker.

(12) When replacing or reinstalling components, take care to prevent foreign material from entering the system.

(13) When handling each unit or printed circuit board in the controller during inspection, turn off the circuit breaker to protect against electric shock.
If there are two cabinets, turn off the both circuit breaker.

(14) A part should be replaced with a part recommended by FANUC. If other parts are used, malfunction or damage would occur. Especially, a fuse that is not recommended by FANUC should not be used. Such a fuse may cause a fire.

(15) When restarting the robot system after completing maintenance work, make sure in advance that there is no person in the work area and that the robot and the peripheral devices are not abnormal.

(16) When a motor or brake is removed, the robot arm should be supported with a crane or other equipment beforehand so that the arm would not fall during the removal.

(17) Whenever grease is spilled on the floor, it should be removed as quickly as possible to prevent dangerous falls.

(18) The following parts are heated. If a maintenance worker needs to touch such a part in the heated state, the worker should wear heat-resistant gloves or use other protective tools.
- Servo motor
- Inside the controller
- Reducer
- Gearbox
- Wrist unit

(19) Maintenance should be done under suitable light. Care must be taken that the light would not cause any danger.

(20) When a motor, reducer, or other heavy load is handled, a crane or other equipment should be used to protect maintenance workers from excessive load. Otherwise, the maintenance workers would be severely injured.

(21) The robot should not be stepped on or climbed up during maintenance. If it is attempted, the robot would be adversely affected. In addition, a misstep can cause injury to the worker.

(22) When performing maintenance work in high place, secure a footstep and wear safety belt.

(23) After the maintenance is completed, spilled oil or water and metal chips should be removed from the floor around the robot and within the safety fence.

(24) When a part is replaced, all bolts and other related components should put back into their original places. A careful check must be given to ensure that no components are missing or left not mounted.

(25) In case robot motion is required during maintenance, the following precautions should be taken :

- Foresee an escape route. And during the maintenance motion itself, monitor continuously the whole system so that your escape route will not become blocked by the robot, or by peripheral equipment.
- Always pay attention to potentially dangerous situations, and be prepared to press the emergency stop button whenever necessary.

(26) The robot should be periodically inspected. (Refer to the robot mechanical manual and controller maintenance manual.) A failure to do the periodical inspection can adversely affect the performance or service life of the robot and may cause an accident

(27) After a part is replaced, a test operation should be given for the robot according to a predetermined method. (See TESTING section of "Controller operator's manual".) During the test operation, the maintenance staff should work outside the safety fence.

# 4 SAFETY OF THE TOOLS AND PERIPHERAL DEVICES

## 4.1 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.

## 4.2 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) Don't use unconfirmed liquid for cutting fluid and cleaning fluid.

(3) Employ a limit switch or mechanical stopper to limit the robot motion so that the robot or cable does not strike against its peripheral devices or tools.

(4) Observe the following precautions about the mechanical unit cables. When theses attentions are not kept, unexpected troubles might occur.

- Use mechanical unit cable that have required user interface.
- Don't add user cable or hose to inside of mechanical unit.
- Please do not obstruct the movement of the mechanical unit cable when cables are added to outside of mechanical unit.
- In the case of the model that a cable is exposed, Please do not perform remodeling (Adding a protective cover and fix an outside cable more) obstructing the behavior of the outcrop of the cable.
- Please do not interfere with the other parts of mechanical unit when install equipments in the robot.

(5) The frequent power-off stop for the robot during operation causes the trouble of the robot. Please avoid the system construction that power-off stop would be operated routinely. (Refer to bad case example.) Please execute power-off stop after reducing the speed of the robot and stopping it by hold stop or cycle stop when it is not urgent. (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type.)
(Bad case example)

- Whenever poor product is generated, a line stops by emergency stop.
- When alteration was necessary, safety switch is operated by opening safety fence and power-off stop is executed for the robot during operation.
- An operator pushes the emergency stop button frequently, and a line stops.
- An area sensor or a mat switch connected to safety signal operate routinely and power-off stop is executed for the robot.

(6) Robot stops urgently when collision detection alarm (SRVO-050) etc. occurs. The frequent urgent stop by alarm causes the trouble of the robot, too. So remove the causes of the alarm.

# 5 SAFETY OF THE ROBOT MECHANISM

## 5.1 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.

## 5.2 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.

## 5.3 PRECAUTIONS FOR MECHANISMS

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

## 5.4 PROCEDURE TO MOVE ARM WITHOUT DRIVE POWER IN EMERGENCY OR ABNORMAL SITUATIONS

For emergency or abnormal situations (e.g. persons trapped in or by the robot), brake release unit can be used to move the robot axes without drive power.
Please refer to controller maintenance manual and mechanical unit operator's manual for using method of brake release unit and method of supporting robot.

# 6 SAFETY OF THE END EFFECTOR

## 6.1 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.

# 7 STOP TYPE OF ROBOT

The following three robot stop types exist:

### Power-Off Stop (Category 0 following IEC 60204-1)

Servo power is turned off and the robot stops immediately. Servo power is turned off when the robot is moving, and the motion path of the deceleration is uncontrolled.
The following processing is performed at Power-Off stop.
- An alarm is generated and servo power is turned off.
- The robot operation is stopped immediately. Execution of the program is paused.

### Controlled stop (Category 1 following IEC 60204-1)

The robot is decelerated until it stops, and servo power is turned off.
The following processing is performed at Controlled stop.
- The alarm "SRVO-199 Controlled stop" occurs along with a decelerated stop. Execution of the program is paused.
- An alarm is generated and servo power is turned off.

### Hold (Category 2 following IEC 60204-1)

The robot is decelerated until it stops, and servo power remains on.
The following processing is performed at Hold.
- The robot operation is decelerated until it stops. Execution of the program is paused.

> ⚠ WARNING
> The stopping distance and stopping time of Controlled stop are longer than the stopping distance and stopping time of Power-Off stop. A risk assessment for the whole robot system, which takes into consideration the increased stopping distance and stopping time, is necessary when Controlled stop is used.

When the emergency stop button is pressed or the FENCE is open, the stop type of robot is Power-Off stop or Controlled stop. The configuration of stop type for each situation is called *stop pattern*. The stop pattern is different according to the controller type or option configuration.

There are the following 3 Stop patterns.

| Stop pattern | Mode | Emergency stop button | External Emergency stop | FENCE open | SVOFF input | Servo disconnect |
|---|---|---|---|---|---|---|
| A | AUTO | P-Stop | P-Stop | C-Stop | C-Stop | P-Stop |
| | T1 | P-Stop | P-Stop | - | C-Stop | P-Stop |
| | T2 | P-Stop | P-Stop | - | C-Stop | P-Stop |
| B | AUTO | P-Stop | P-Stop | P-Stop | P-Stop | P-Stop |
| | T1 | P-Stop | P-Stop | - | P-Stop | P-Stop |
| | T2 | P-Stop | P-Stop | - | P-Stop | P-Stop |
| C | AUTO | C-Stop | C-Stop | C-Stop | C-Stop | C-Stop |
| | T1 | P-Stop | P-Stop | - | C-Stop | P-Stop |
| | T2 | P-Stop | P-Stop | - | C-Stop | P-Stop |

P-Stop: Power-Off stop
C-Stop: Controlled stop

-: Disable

The following table indicates the Stop pattern according to the controller type or option configuration.

| Option | R-30*i*B |
|---|---|
| Standard | A (*) |
| Controlled stop by E-Stop (A05B-2600-J570) | C (*) |

(*) R-30*i*B does not have servo disconnect.

| Option | R-30*i*A | | | | R-30*i*A Mate | | |
|---|---|---|---|---|---|---|---|
| | Standard (Single) | Standard (Dual) | RIA type | CE type | Standard | RIA type | CE type |
| Standard | B (*) | A | A | A | A (**) | A | A |
| Stop type set (Stop pattern C) (A05B-2500-J570) | N/A | N/A | C | C | N/A | C | C |

(*) R-30*i*A standard (single) does not have servo disconnect.

(**) R-30*i*A Mate Standard does not have servo disconnect, and the stop type of SVOFF input is Power-Off stop.

The stop pattern of the controller is displayed in "Stop pattern" line in software version screen. Please refer to "Software version" in operator's manual of controller for the detail of software version screen.

## "Controlled stop by E-Stop" option

When "Controlled stop by E-Stop" (A05B-2600-J570) option (In case of R-30*i*A/R-30*i*A Mate, it is Stop type set (Stop pattern C) (A05B-2500-J570)) is specified, the stop type of the following alarms becomes Controlled stop but only in AUTO mode. In T1 or T2 mode, the stop type is Power-Off stop which is the normal operation of the system.

| Alarm | Condition |
|---|---|
| SRVO-001 Operator panel E-stop | Operator panel emergency stop is pressed. |
| SRVO-002 Teach pendant E-stop | Teach pendant emergency stop is pressed. |
| SRVO-007 External emergency stops | External emergency stop input (EES1-EES11, EES2-EES21) is open. (R-30*i*A/R-30*i*B controller) |
| SRVO-194 Servo disconnect | Servo disconnect input (SD4-SD41, SD5-SD51) is open. (R-30*i*A controller) |
| SRVO-218 Ext.E-stop/Servo Disconnect | External emergency stop input (EES1-EES11, EES2-EES21) is open. (R-30*i*A Mate/R-30*i*B controller) |
| SRVO-408 DCS SSO Ext Emergency Stop | In DCS Safe I/O connect function, SSO[3] is OFF. |
| SRVO-409 DCS SSO Servo Disconnect | In DCS Safe I/O connect function, SSO[4] is OFF. |

Controlled stop is different from Power-Off stop as follows:

- In Controlled stop, the robot is stopped on the program path. This function is effective for a system where the robot can interfere with other devices if it deviates from the program path.
- In Controlled stop, physical impact is less than Power-Off stop. This function is effective for systems where the physical impact to the mechanical unit or EOAT (End Of Arm Tool) should be minimized.
- The stopping distance and stopping time of Controlled stop is longer than the stopping distance and stopping time of Power-Off stop, depending on the robot model and axis. Please refer to the operator's manual of a particular robot model for the data of stopping distance and stopping time.

In case of R-30*i*A or R-30*i*A Mate, this function is available only in CE or RIA type hardware.

When this option is loaded, this function cannot be disabled.

The stop type of DCS Position and Speed Check functions is not affected by the loading of this option.

> ⚠ WARNING
> The stopping distance and stopping time of Controlled stop are longer than the stopping distance and stopping time of Power-Off stop. A risk assessment for the whole robot system, which takes into consideration the increased stopping distance and stopping time, is necessary when this option is loaded.

120919

# TABLE OF CONTENTS

# APPENDIX

# 1 OVERVIEW

## 1.1 OVERVIEW

This manual contains information about robot networking options including FTP, Advanced Internet Connectivity and Customization, and Socket Messaging.
For information on the PC-Interface option see the PC Developer Kit documentation, and related help files.
The FTP option is loaded by default with all application software packages. The FTP option on the robot includes :

- FTP Server, which allows remote FTP clients to initiate file transfers with the robot ( Section 1.3 )
- FTP Client, which allows the robot to initiate file transfers with remote FTP servers ( Section 1.3 )
- Telnet Server, which allows remote telnet clients to access teach pendant display ( Section 1.6 )
- Web Server, which allows remote browsers to access the robot web server and accessing error logs, ascii program listings, and a wealth of diagnostic content ( Section 1.8 )
- Remote access to the robot through a serial modem using the point to point protocol (PPP). ( Chapter 8 )

The Advanced Internet Connectivity and Customization option includes :

- *i*Pendant Proxy Server, allowing the *i*Pendant to browse outside of the robot to other web servers across the robot Ethernet connections
  ( Chapter 7 )
- Enhanced Web Server, allowing access to customized web pages on the robot with dynamic content
  ( Chapter 6 )
- Domain Name Service (DNS), allowing the robot DNS client to contact a remote DNS server to resolve network names into IP addresses. This is useful for FTP client functionality on the robot when network names are used and also for browsing with the *i*Pendant
  ( Chapter 4 )
- Dynamic Host Configuration Protocol (DHCP), allowing the robot DHCP client to contact a remote DHCP server to get network identity such as IP address, name, subnet mask, and router settings.
  ( Chapter 9 )
- Simple Network Time Protocol (SNTP), allowing the robot SNTP client to get updated date/time information from a remote SNTP server
  ( Chapter 11 )
- PC Share, allowing the robot to connect to and perform file operations on remote PC network Shares.
  ( Chapter 14 )

> NOTE
> Note that the Advanced Internet Connectivity and Customization option will also load the FTP option if it is not already loaded.

The Socket Messaging option enables an application developer to write KAREL applications on the robot to communicate with unique application protocols based on TCP/IP and the sockets interface. ( Chapter 10 )
It is extremely useful to understand the various file devices available on the robot when accessing the robot remotely using FTP or Web Server. These include :

- Memory Device (MD:). Files on this device are created dynamically based on the current contents of user programs, variables, and diagnostic data in both binary and ASCII formats. This is the default device when first connecting to the robot FTP server (however you can change directory to other devices).

  - Binary Memory Device (MDB:). This subset of memory device includes only the binary versions and is roughly equivalent to a “Backup – all of the above” from the teach pendant file menu.
- FlashRom (FR:)
- RamDisk (RD:)
- Memory Card (MC:)
- USB Memory Stick Device (UD1:, UT1:)

See the “Storage Devices” section under the “Program and File Manipulation” chapter in the application tool OPETATOR’S MANUAL for additional details on these devices.
You need to set up the TCP/IP parameters for your robot’s controller before you can set up and use any of these options. Refer to Chapter 2 for information about setting up the TCP/IP parameters.

NOTE
You must supply the Ethernet cable to attach to the Ethernet port in the controller.

Figure 1.1 shows the typical components used in a communications network.

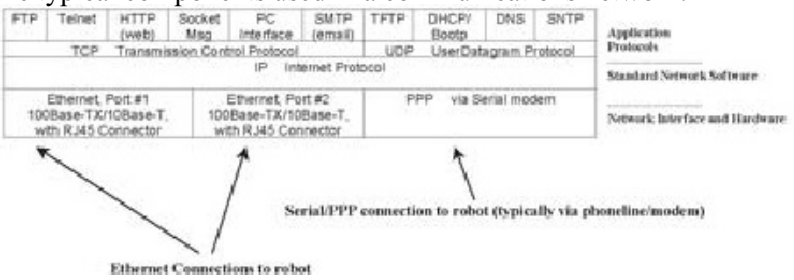



Fig. 1.1 Network components

# 1.2 BACKUP AND RESTORE

## 1.2.1 Overview

The following kinds of backup and restore methods are provided:

- Application file backup and restore
- Image backup and restore - includes complete operating system, loaded options, and application files (generally, non-volatile memory) as image

## 1.2.2 Application File Backup and Restore

File-based backup and restore operates on discrete files. The controller must be operational in the Controlled start or Cold start state and have FTP installed to perform file operations.

# 1.3 FILE TRANSFER PROTOCOL (FTP)

The File Transfer Protocol (FTP) comes from the TCP/IP Internet protocol suite. It promotes sharing of files between diverse computers. The FTP Interface uses the following commands:

- Server
  - get
  - put
  - mget
  - mput
  - dir
  - delete

  - rename
  - cd
- Client
  - get
  - put
  - mget
  - mput
  - dir
  - delete

The FTP Interface, or FTP, was designed to conform with the appropriate subset of the FTP Specification. FTP is the application layer of the "File Transfer Protocol (FTP)," RFC 959, ISI, October 1985.
The commands listed above are for use with FTP on a robot.
FTP function is a standard function. FTP function is loaded by default with all application software packages.

## 1.4 TCP/IP PROTOCOL

Transmission Control Protocol (TCP) is intended for use as a highly reliable host-to-host protocol between hosts in packet-switched computer communications networks. It fits into a layered protocol architecture just above a basic Internet Protocol (IP). The IP provides a way for TCP to send and receive variable-length segments of information enclosed in Internet datagram "envelopes."

## 1.5 BOOTP AND TFTP PROTOCOLS

1.4 BOOTP AND TFTP PROTOCOLS The BOOTP and TFTP protocols are generally used to boot diskless workstations on a TCP/IP communications network. BOOTP provides the identity (IP address) to the diskless workstation based on an Ethernet hardware address. TFTP is then used to transfer information to load the workstation. The robot uses these protocols for image backup and restore operations.

## 1.6 TELNET

The controller can support three Telnet connections. Telnet can be used to establish terminal sessions between a robot controller and a remote PC with Telnet software installed on it. This allows you to access your robot’s teach pendant display remotely, CRT/KB options, or a Diagnostic terminal depending on your system’s configuration.
Telnet function is a standard function. Telnet function is loaded by default with all application software packages.

## 1.7 DOMAIN NAME SERVICE (DNS)

Domain Name Service (DNS) allows a robot controller to establish an Ethernet connection to a remote server without having to know the IP address of the remote server.
DNS function is a option software. R558 is required for this function.

## 1.8 WEB SERVER

The robot controller supports the hypertext transfer protocol (http) and can act as a web server, which allows it to respond to a remote web browser's request for information from the robot controller. In addition, the web server option can allow you to access diagnostic information, ASCII versions of system

variables, and teach pendant programs. The FANUC Robotics web server option is compatible with most http software packages.
Web server function is standard function. Web server function is loaded by default with all application software packages.

## 1.9 PROXY SERVER

The proxy server on the robot allows you to browse web servers on the network from the *i*Pendant. For the browser on the *i*Pendant to be able to view web servers on the network, it needs a proxy server to *proxy* web requests from the *i*Pendant to the remote server.
Proxy server function is standard function. Proxy server function is loaded by default with all application software packages.

## 1.10 POINT-TO-POINT PROTOCOL CONNECTIVITY

Point-to-Point Protocol (PPP) allows devices to connect to each other across a dedicated point to point link.
Point-to-point protocol connectivity is standard function. Point-to-point protocol connectivity is loaded by default with all application software packages.

## 1.11 DYNAMIC HOST CONFIGURATION PROTOCOL

DHCP ( **Dynamic Host Configuration Protocol** ) is a service which automates robot configuration on an existing Ethernet network.
DHCP function is an option software. R558 is required for this function.

## 1.12 SOCKET MESSAGING

The User Socket Messaging Option gives you the benefit of using TCP/IP socket messaging from KAREL.
Socket messaging function is an option software. R648 is required for this function.

## 1.13 SIMPLE NETWORK TIME PROTOCOL (SNTP)

Simple Network Time Protocol (SNTP), allowing the robot SNTP client to get updated date/time information from a remote SNTP server.
Simple network time protocol function is an option software. R610 is required for this function.

## 1.14 ETHERNET PACKET SNIFFER

The Ethernet Packet Sniffer allows for packets to be captured directly on the robot controller and then saved to a file.
Ethernet packet sniffer function is an option software. R659 is required for this function.

## 1.15 ROS INTERFACE PACKETS OVER ETHERNET (RIPE)

Real Time Operating System (ROS) Interface Protocol over Ethernet feature (also called Robot Ring, RIPE or ROSIP) allows robots doing a common job to share information.

RIPE function is an option software. This function will be loaded when iRVision or Robot link function is ordered.

# 1.16 HOST COMMUNICATIONS

## 1.16.1 Overview

The FTP Interface enables the controller to communicate with external or host devices across an Ethernet network. FTP uses host communications to perform communications operations.
To use the FTP Interface, you must understand host communications. This section contains information on

- Host communication architecture
- Host communication devices

## 1.16.2 Architecture

The host communications architecture is based on a *client-server* model. In this model,

- The **client** is the device that needs a service.
- The **server** is the device that provides the service.

### Clients

Host communications clients request a service to be performed and receive service replies. You access robot clients using a client device name, called a tag. Client tags are C1: through C8:. When the controller acts as the client, all service requests will pass from the controller to the host device. After a tag is started, it becomes a device available to the controller. The host device will operate as a server, responding to requests from the controller as they are received. See Figure 1.16.2(a).

NOTE
Client operation is available from the teach pendant.

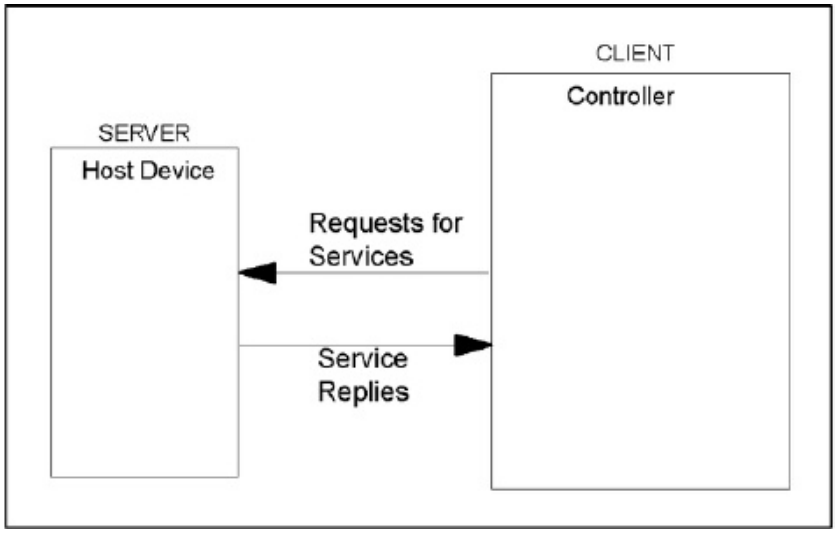



Fig. 1.16.2(a) The controller as a client

### Servers

You access robot servers using a server device name, called a *tag* . Host communications servers are started on devices with server tags S1: through S8:. These devices cannot be accessed directly. A server is normally started on a tag and runs transparently to the controller.
A host device operating as a client will make service requests to the server, which is the controller. See Figure 1.16.2(b) .

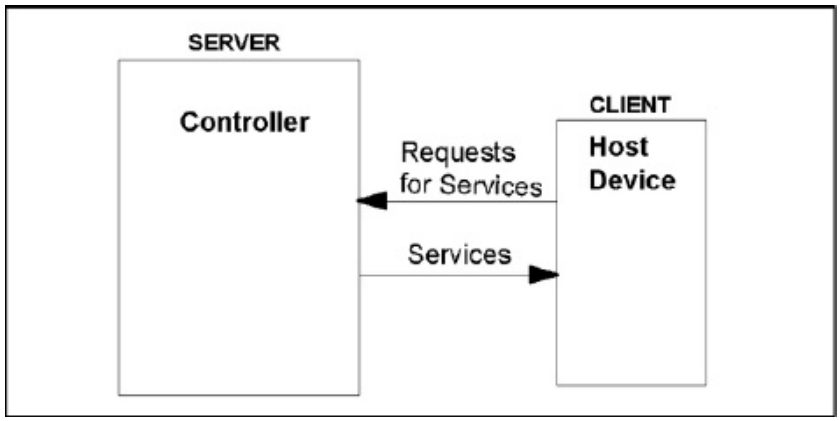



Fig. 1.16.2(b) The controller as a server

## 1.16.3 Devices

A host communications device consists of

- A communications tag
- A communications protocol
- An optional serial port name (not used with FTP)

### Defining a Device

You make communications devices known to the system by defining them. Defining a communications device involves specifying the communications tag and protocol.

Defining a device makes the device known to the system but does not allocate the resources the device needs.

To remove a communications device from the system, you must undefine it. This frees the tag so it can be defined as another device.

### Using a Device

The way in which a device is used depends upon the kind of device it is.

**Client devices C1: through C8:** are used like local file storage devices. Client devices do not have to be started before they are accessed. The devices automatically will be started when opened and stopped when closed. Client devices must be defined before they can be used.

**Server devices S1: through S8:** must be started before any services can be requested. Servers are normally started upon power up and remain running while the controller is powered up. All host devices can be configured to start automatically upon power up.

# 2 SETTING UP TCP/IP

## 2.1 OVERVIEW

You must set up TCP/IP before you can use Internet Protocol Applications. Setup is required in two areas:

- Hardware - includes port initialization and cable and connector requirements
- Software - includes host communication device definition

## 2.2 HARDWARE REQUIREMENTS AND INSTALLATION

### 2.2.1 Overview

This section contains information on hardware requirements and installation for the Ethernet interface. After you have connected the Ethernet interface to the network, you must configure the TCP/IP parameters. Refer to Section 2.4,Section 4.2 for information about installing and configuring FTP and TCP/IP parameters.

### 2.2.2 Hardware Requirements

R-30*i*A and R-30*i*B supports two 10 Base-T or 100 Base-TX interfaces through the RJ45 Ethernet connectors (CD38A and CD38B). R-30*i*B controller has another one RJ45 Ethernet connector which is labeled CD38C. This port is strictly for digital camera for FANUC. By default, each RJ45 Ethernet port will auto-negotiate with the other equipment on the network. Refer to Appendix A for information on the connector and diagnostic LEDs. R-30*i*A Mate supports only one 10 Base-T or 100 Base-TX interfaces through the RJ45 Ethernet connector.
The auto-negotiate feature can be disabled through the $ENETMODE system variable. This should only be needed in special circumstances such as when Full Duplex behavior is desired and the other node does not support auto-negotiation.

NOTE
CD38C port is a dedicated digital camera for FANUC. It is not allowed to use as general-purpose Ethernet port. Please do not connect Ethernet cable to CD38C of R-30*i*B controller other than the above camera.

NOTE
$ENETMODE[1] refers to the first (upper RJ45 on the Main board) interface labeled as CD38A and $ENETMODE[2] refers to the second interface labeled as CD38B. $ENETMODE[2] is not supported on R-30*i*A Mate.

Table 2.2.2 Ethernet configuration setup

| Baud Rate/Duplex | Half Duplex | Full Duplex |
|---|---|---|
| 10 MBPS | $ENETMODE[ ].$SPEED=0<br>$ENETMODE[ ].$FULL_DUPLEX=FALSE | $ENETMODE[ ].$SPEED=0<br>$ENETMODE[ ].$FULL_DUPLEX=TRUE |
| 100 MBPS | $ENETMODE[ ].$SPEED=1<br>$ENETMODE[ ].$FULL_DUPLEX=FALSE | $ENETMODE[ ].$SPEED=1<br>$ENETMODE.$FULL_DUPLEX=TRUE |

NOTE
The default settings of $ENETMODE[ ].$SPEED=2 indicate that auto-negotiation will be used. Normally this variable should not be charged. The baud rate and duplex node will be set to the fastest setting that both devices on the list can support.

See Figure 2.2.2(a) for location of the 10 Base-T/100 Base-TX of R-30*i*B. See Figure 2.2.2(b) for location of the 10 Base-T/100 Base-TX of R-30*i*A. See Figure 2.2.2(c) for location of the 10 Base-T/100 Base-TX of R-30*i*A Mate.

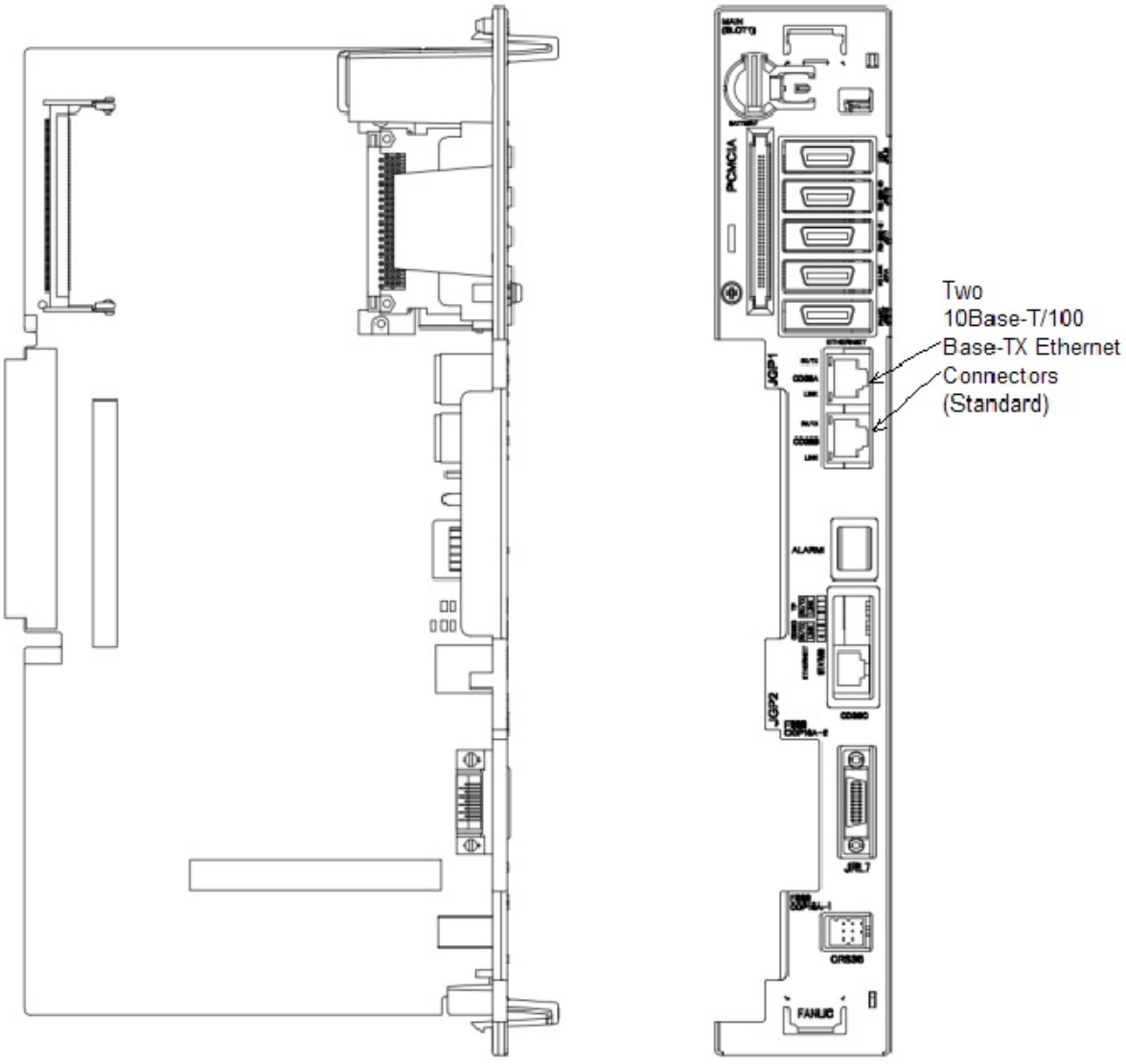



Fig. 2.2.2(a) Main board (R-30*i*B) ethernet connectors

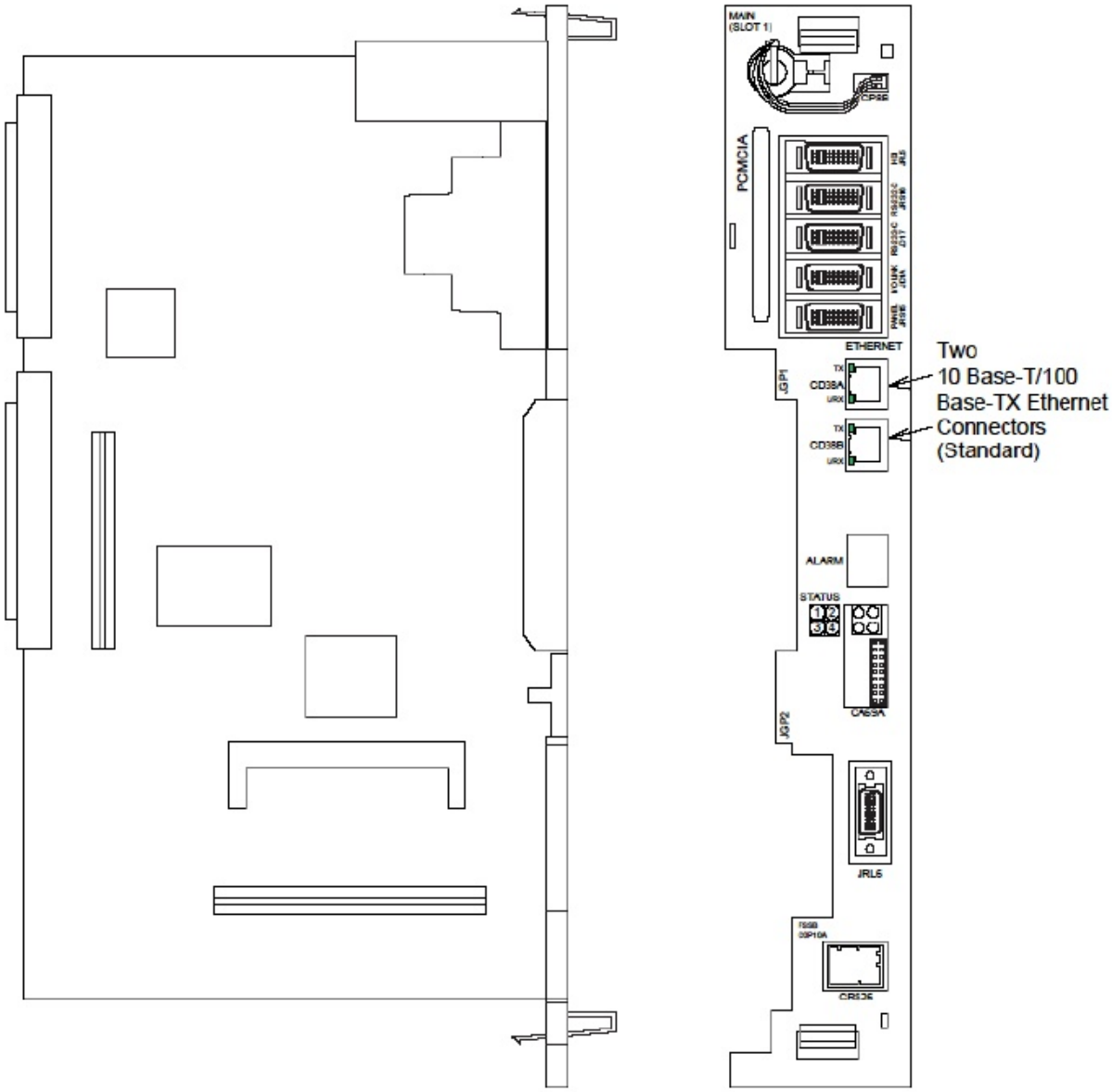



Fig. 2.2.2(b) Main board (R-30*i*A controller) ethernet connectors

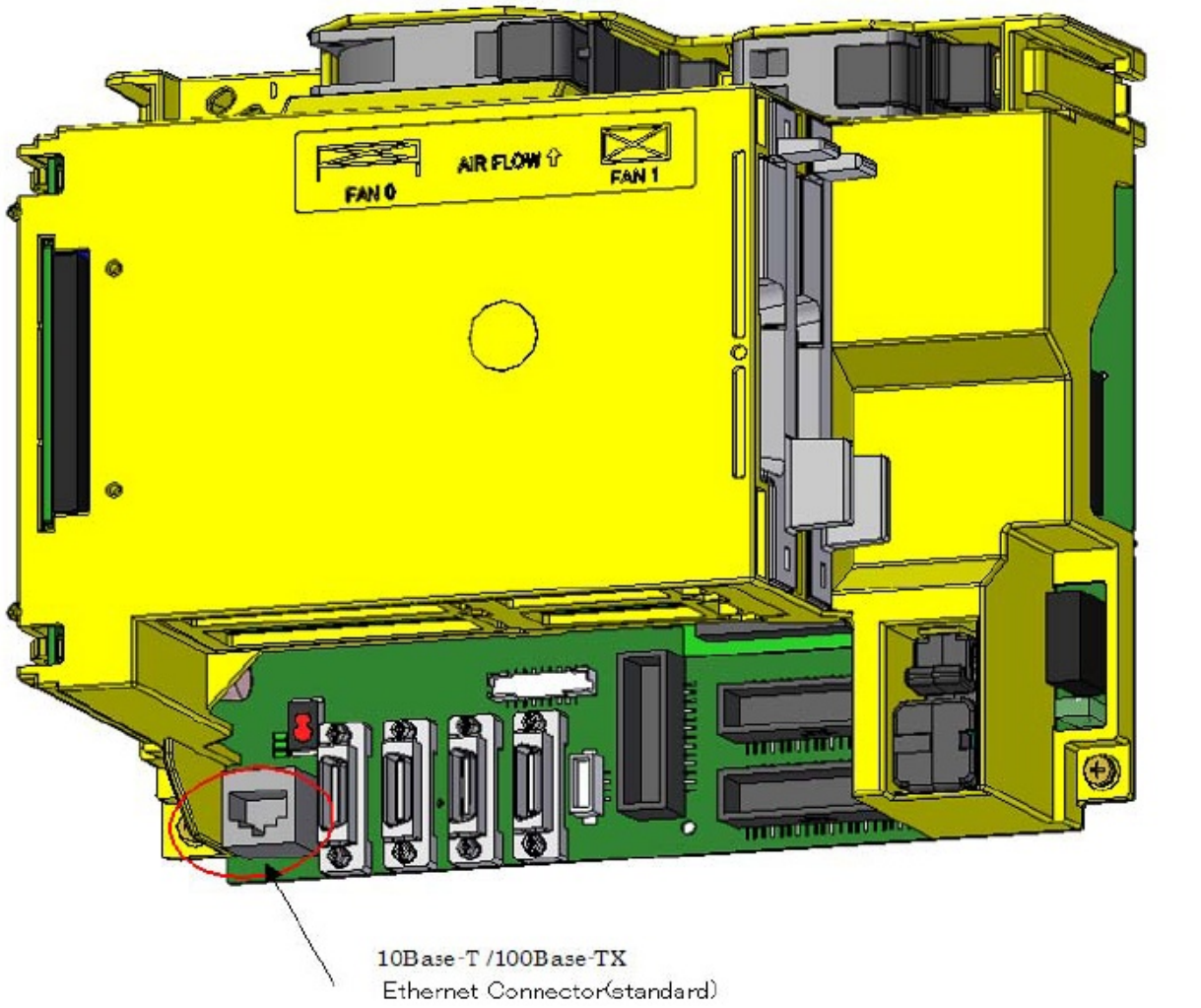



Fig. 2.2.2(c) Main board (R-30*i*A Mate) ethernet connectors

# 2.3 DISPLAYING THE ETHERNET HARDWARE (MAC) ADDRESS

## 2.3.1 Overview

For communications to occur over the Ethernet, the Ethernet Hardware (MAC) Address must be set. This section shows you how to display the Ethernet Hardware address, which might be required in the process of configuring a BOOTP server.

## 2.3.2 Ethernet Hardware (MAC) Address

The Ethernet Hardware Address is set by the manufacturer, and consists of a 6 byte (48 bit) value. The first three bytes are the manufacturer's code, and the last three bytes are a unique serial number for the Ethernet interface.

The Ethernet Hardware (MAC) address can be found on a label attached to the Main board. See Figure 2.3.2(a) for R-30*i*B. See Figure 2.3.2(b) for R-30*i*A. See Figure 2.3.2(c) for R-30*i*A Mate.

Fig. 2.3.2(a) Ethernet hardware (MAC) address on main board for R-30*i*B

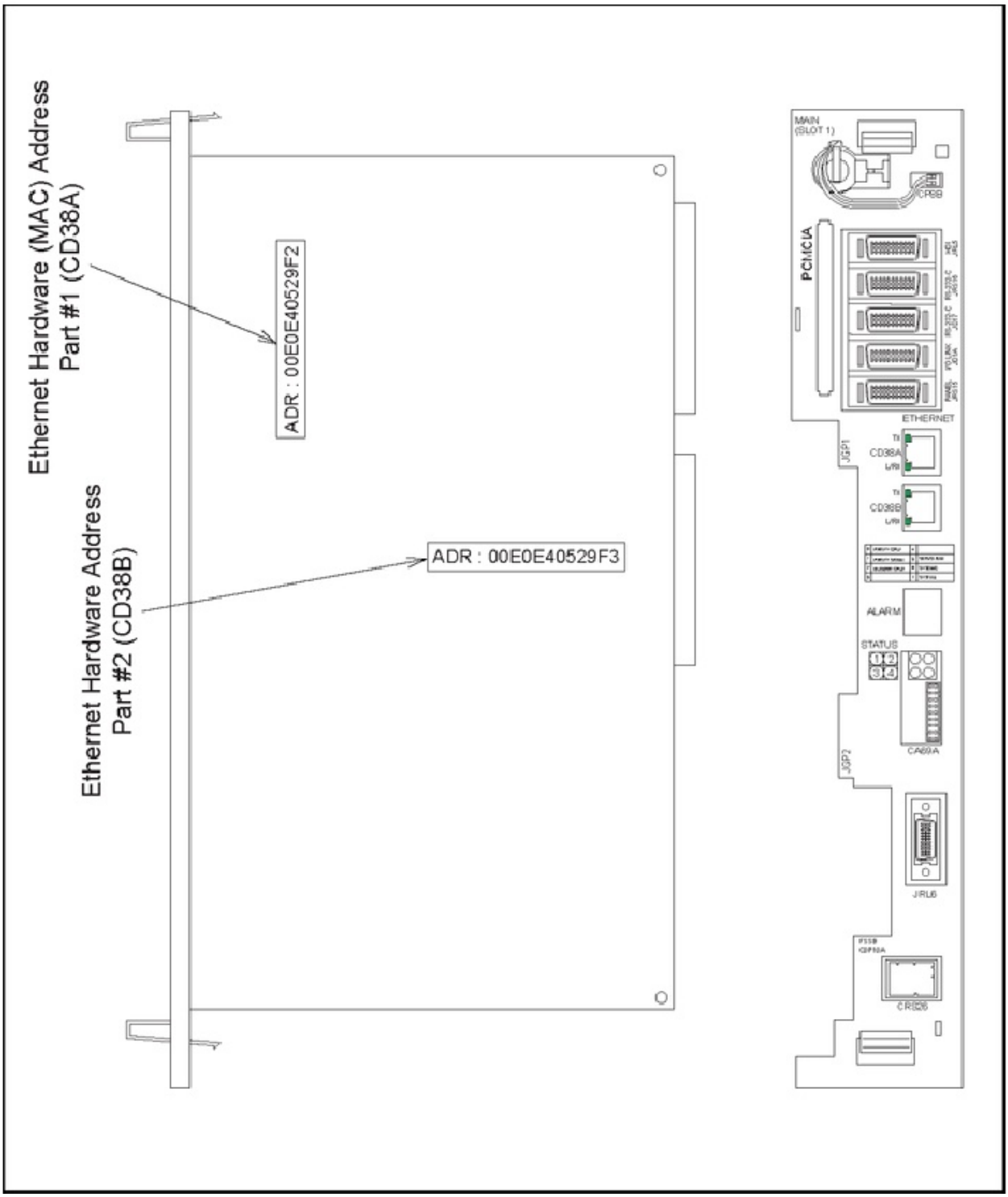



Fig. 2.3.2(b) Ethernet hardware (MAC) address on main board for R-30*i*A

Fig. 2.3.2(c) Ethernet hardware (MAC) address on main board for R-30*i*A Mate

## 2.3.3 Ethernet Hardware (MAC) Address Locations

The Ethernet Hardware (MAC) address can be found in the following locations:

- The physical label on the Main board. See Figure 2.3.2(b) .
- Using SHOW ETHERNET ADDRESS from the BMON Menu. Refer to Procedure 2-1 .
- The Board Address, which can be accessed from the TCP/IP Setup Screen. Refer to Section 2.4 .
- In the system variable $TMI_ETHERAD[x] where x is 1 for port 1, and 2 for port 2.

NOTE

You cannot make changes to the Ethernet Hardware MAC address.

### Procedure 2-1 Displaying the Ethernet Hardware (MAC) Address

### Steps

1. Turn off the controller. Hold the F1 and F5 keys while you turn on the controller. The controller will display the BMON Menu. You will see a screen similar to the following.

```
******* BMON MENU *******
1. Configuration menu
2. All software installation(MC:)
3. INIT start
4. Controller backup/restore
5. Hardware diagnosis
6. Maintenance
7. All software installation(Ethernet)
Select :
```

2. Select Hardware Diagnosis, and press ENTER. You will see a screen similar to the following.

```
****** Hardware Diagnoses Menu ******
1. Show size of RAM/ROM modules
2. Show list of S-BUS modules
3. Dump memory
4. Write memory
5. Check SRAM memory
6. Clear vision SRAM memory
7. Check FROM memory
8. Display MAC address
9. Return to main menu
Select :
```

3. Select Display MAC Address, and press ENTER. You will see a screen similar to the following.

```
MAC ADDRESS Number ? [1-3]  :
```

4. Select 1–3 and press ENTER. The MAC address will be displayed similar to the following.

```
MAC address[1] 00:E0:E4:F0:A1:12
```

Press Enter and then choose Return to Main Menu to display the BMON Menu. Then choose the Configuration Menu. From this menu you can choose to perform a Controlled start, Cold start, or Hot start.

# 2.4 SETTING UP TCP/IP

There are four options for configuring the FTP software and TCP/IP parameters:

- Use Procedure 2-1 through Procedure 2-2 if you want enter all the information necessary for FTP and TCP/IP setup yourself.
- Use the FTPSETUP program to enter the information for you. Refer to Appendix B for information about using the FTPSETUP program
- Use Dynamic Host Configuration Protocol (DHCP) to automatically setup IP address, name, subnet mask, and router.

## TCP/IP Parameters

Several parameters are used to configure and set the functions of the TCP/IP connections. Table 2.4(b) lists and describes the TCP/IP Interface parameters you must define.

Table 2.4(a) SETUP protocols screen items

| ITEM | DESCRIPTION |
|---|---|
| TCP/IP | This item allows you to configure networking parameters. |
| TELNET | This item allows you to configure TELNET parameters. |
| SM | This item allows you to configure socket messaging parameters. |
| RIPE | This item allows robots doing a common job to share information. |
| PROXY | This item allows you to configure proxy server parameters. |

| ITEM | DESCRIPTION |
|---|---|
| PPP | This item allows you to configure Point to Point Protocol. |
| PING | This item allows you to check networking connectivity on the robot. |
| HTTP | This item allows you to configure HTTP parameters. |
| FTP | This item allows you to configure FTP parameters. |
| DNS | This item allows you to configure domain name system parameters. |

Table 2.4(b) TCP/IP interface parameters

| PARAMETERS | DESCRIPTION |
|---|---|
| Robot Name | This item specifies the name of the robot controller. The robot name defaults to ROBOT.<br>This name field is common between Ethernet ports and is local to the robot. |
| Port # IP Address | This item specifies a unique internet (IP) Address for the robot Ethernet interface. Consult your network administrator for the IP address setting. The port # indicates whether you are working with port #1 (TOP RJ45 connection labeled as CD38A) or port #2 (bottom RJ45 connection labeled as CD38B). Use the (F3) port FUNCTION key to change ports to configure. |
| Router IP Address | This item specifies the Internet (IP) Address of the router. This setting is common between Ethernet ports. The router IP address must be on the same subnet as one of the Ethernet ports. |
| Subnet Mask | This item is used to distinguish local hosts from hosts that must be reached across routers. The default is 255.255.255.0. Consult your network administrator for the proper setting. |
| Board Address | This item displays the Ethernet Hardware (MAC) address for the Ethernet Interface. This field is read only. This address conforms to the standards of Ethernet board addresses. |
| Host Name | This item specifies the Internet host name. Entries for any hosts referred to by an FTP client tag are required. This item is case sensitive. |
| Internet Address | This item specifies the corresponding Internet address of each host. |

Use Procedure 2-2 to define TCP/IP parameters.

## Procedure 2-2 Defining TCP/IP Parameters

### Conditions

- You have performed TCP/IP hardware installation. Refer to Section 2.2 if you have not installed the hardware.

### Steps

1 Press MENUS.
2 Select SETUP.
3 Press F1, [TYPE].
4 Select Host Comm. You will see a screen similar to the following.

NOTE
There are two areas in which to enter the Host Name and Internet Address mappings on the TCP/IP Setup screen:

**Local Area** - Data in this area is saved as part of SYSVARS.SV ($HOSTENT[]). SYSVARS.SV should not be shared between robots.

**Shared Area** - can include any Host Name/Internet Address mapping that is to be used as part of the client tag configuration, but should not include robot name or router name entries. Data in this area is saved as part of SYSHOST.SV ($HOST_SHARED[]).

In addition to Host name/Internet Address mapping, SYSHOST.SV ($HOST_SHARED[]) contains information about Telnet and DNS. A SYSHOST.SV can be shared between robots and can be downloaded from one robot to create a complete DNS, Telnet, and Shared host configuration on another robot.

7 Move the cursor to each item and specify the appropriate information:

- **Robot name**— specify the unique name of the robot controller.
- **Port #**— indicates whether you are configuring interface #1 (top RJ45 labeled as CD38A) or interface #2 (bottom RJ45 labeled as CD38B). Use the F3, Port key to change.
- **Robot IP Address**— specify IP address of the robot.
- **Subnet Mask** - This must be set. The default value is 255.255.255.0. Consult your network administrator for guidance in setting this value. Refer to Table 2.4(c) for standard subnet mask settings.
- **Board address**— This is the Ethernet (MAC) address of the robot.
- **Router IP address** — specify IP address of the router. This can be left blank if no router is used. The router address needs to be on the same subnet as interface #1 or interface #2. This is where packets for any destination not on subnet for interface #1 or #2 will be sent.

⚠WARNING
Restrictions for the robot name.

- It is available only one symbol alphabet, numbers, and minus.
  Other symbols can not be used. Also, please do not insert the extra space.
- The first character must be alphabet.
- The last character must not be symbol minus.

⚠WARNING
Please do not insert the extra space in the IP address or 0. The controller will not be able to communicate properly if there is extra space or zero.

NOTE
The board address is displayed and cannot be changed. Refer to Section 2.3 if you want to display the Ethernet Hardware (MAC) address.

NOTE
Robot Name, Router IP address, and the Host Name/Internet Address table are shared between Ethernet interface #1 and interface #2.

- **Host Name/Internet Address** - specify the unique host name and Internet address of each host with which the controller will communicate as a client.

Table 2.4(c) Standard subnet mask settings

| If the first byte of the IP address is between | Set the subnetmask to |
|---|---|
| 0 and 127 (Class A) | 255.0.0.0 |
| ~~128 and 191 (Class B)~~<br>192 and 223 (Class C) | ~~255.255.0.0~~<br>255.255.255.0 |

8 Press F3, LIST, to return to the SETUP Protocols screen.

NOTE
If the controller is connected to an isolated or private network and no routers are used, all equipment must use the same network address in order to communicate. RFC 1597 makes recommendations for setting IP addresses on isolated or private networks. An example of this is the network address 192.168.0 is a Class C address and can support 254 devices, 192.168.0.1 through 192.168.0.254. If you have a private network and have no constraints for setting IP addresses, use the Class C network address 192.168.0.X, where X is a unique number between 1 and 254, for each device on your network.

## 2.4.1 Caution for Setting IP Address

Each IP address for 2 Ethernet ports of the controller must be in network with different subnet. Different subnet means network address which determined by subnet mask is different.
For example, network address is 192.168.1 and host address is 10 when subnet mask is 255.255.255.0 and IP address is 192.168.1.10. "Host-179 IP Address mis-configuration" alarm will be asserted when IP address for 2 Ethernet ports has the same subnet. 2 Ethernet ports of the controller are not able to use as Hub. Controller is not able to set 2 nodes for one network.

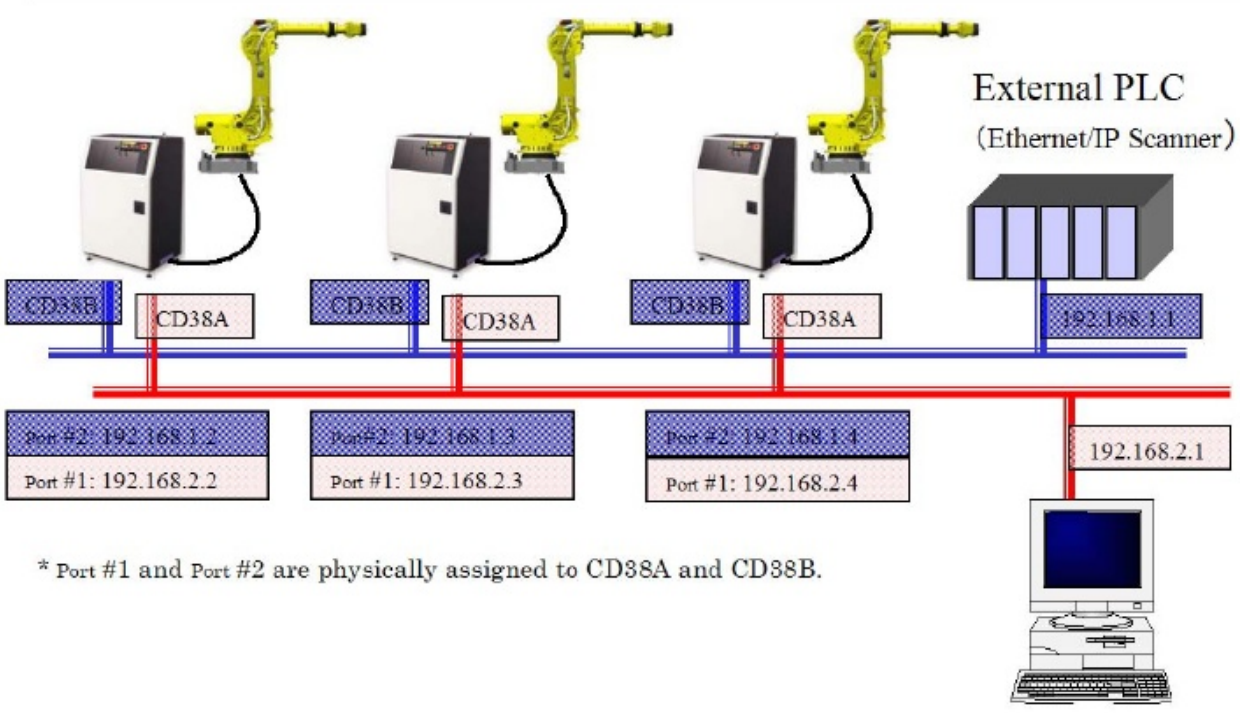



Fig. 2.4.1 Example of correct IP setup (In case, subnet is 255.255.255.0)

# 2.5 FANUC SERVER ACCESS CONTROL (FSAC)

## 2.5.1 Overview

The FANUC Server Access Control (FSAC) feature controls access to the robot communication servers based on the host (client) IP address. FSAC is loaded as part of the FTP option and is disabled by default. The FSAC feature provides no access control at the teach pendant, so properties of this feature can be modified at any time by someone at the teach pendant (variables associated with this feature take effect immediately). Comparing the SYSFSAC.SV file with a known "correct" file on the host system is the intended method to monitor setup. All setup is done directly through system variables.

NOTE
This feature only works if passwords are disabled.

## 2.5.2 Access Levels

Access levels allow you to perform certain kinds of actions, and allow access to specific system areas, based upon the type of access granted. Refer to Table 2.5.2 for descriptions of available FSAC access levels.

Table 2.5.2 FSAC access levels

| Access Level | Description | Type of Access |
|---|---|---|
| 0 | Operator level | Read only access |
| 1 | Program level | Operator level, with additional access to download the following types of files:<br>● TP (teach pendant)<br>● .PC (p-code)<br>● .VR (variable) |
| 2 | Setup level | Program level, with additional access to download the following types of files:<br>● .SV (system)<br>● .IO (i/o config) |
| 3–7 | User-Defined levels | Read-only access |
| 8 | Installation level | Full read/write access |

The access level granted is indicated at login. For example, you might see a message similar to the following:

```
230 User logged in at Operator Level.
```

If an operation is attempted without the appropriate access level, a response is given indicating the required access level. See the following screen for an example.

```
/vob/net/ftp$ ftp sleepy
Connected to sleepy
220 R-J2 FTP server ready
Name (sleepy:huberjf
230 User logged in at Program Level
ftp> binary
200 Type set to 1
ftp put sysfsac.sv
200 PORT command successful
550 Requires SETUP password
ftp>
```

## 2.5.3 Access Denied

If the FSAC feature is enabled and access is not granted, the following response is sent to the FTP client:
**421 Access Denied (FSAC) : closing control connection**

This message is sent in response to the USER portion of the login sequence and will actively close the FTP connection.

## 2.5.4 System Variables

The FSAC feature contains system variables in a file called SYSFSAC.SV. This file can be shared between robots that have the same FTP software installed, and should always be transferred in BINARY mode. Refer to Table 2.5.4 for a description of the system variables contained in SYSFSAC.SV.

Table 2.5.4 System variables contained in SYSFSAC.SV

| Variable Name | Data Type | Description |
|---|---|---|
| $FSAC_ENABLE | Integer | FSAC Enable Flag. This can be set to either:<br>● disabled (any value other than 1 will disable it)<br>● 1, enabled |
| $FSAC_DEF_LV | Integer | FSAC Default Access Level. This can be set to:<br>● 0, operator level<br>● 1, program level<br>● 2, setup level<br>● 3–7, user-defined levels<br>● 8, installation level<br>● any other level is no access |
| $FSAC_LIST[].$CLNT_NAME | String | The name of the host system. Example: MYPC<br>NOTE<br>The name must be in the LOCAL/SHARED host table or DNS must be installed to resolve names. |
| $FSAC_LIST[1-20].$IP_ADDRESS | String | The IP Address of the host system. Example: 199.5.148.62 |
| $FSAC_LIST[]. $ACCESS_LVL | Integer | The access level for the specific host set in $FSAC_LIST.$IP_ADDRESS. Valid values are the same as those used in $FSAC_DEF_LV. |

| | | |
|---|---|---|
| $FSAC_LIST[].$APPS | Integer | Applications that use this entry. The default is 255. Multiple applications can be specified using the following bit mask:<br>● BIT 0: FTP<br>● BIT 1: Telnet<br>● BIT 2: HTTP (Web Server) |

## 2.5.5 Example Configuration

To enable FTP Server Access Control on Robot 1, and give full READ/WRITE access to HOST_1 and READ ONLY access to any other devices trying to use FTP to communicate with Robot 1, set the Robot 1 system variables as follows:

Example 2.5.5 Example system variable configuration

```
$FSAC_ENABLE = 1
$FSAC_DEF_LV = 0
$FSAC_LIST[1].$IP_ADDRESS = '199.5.148.62'
$FSAC_LIST[1].$ACCESS_LVL = 8
```

To configure Robot 2 in the same way, copy the SYSFSAC.SV file from the Robot 1 controller to the Robot 2 controller. See Figure 2.5.5 .

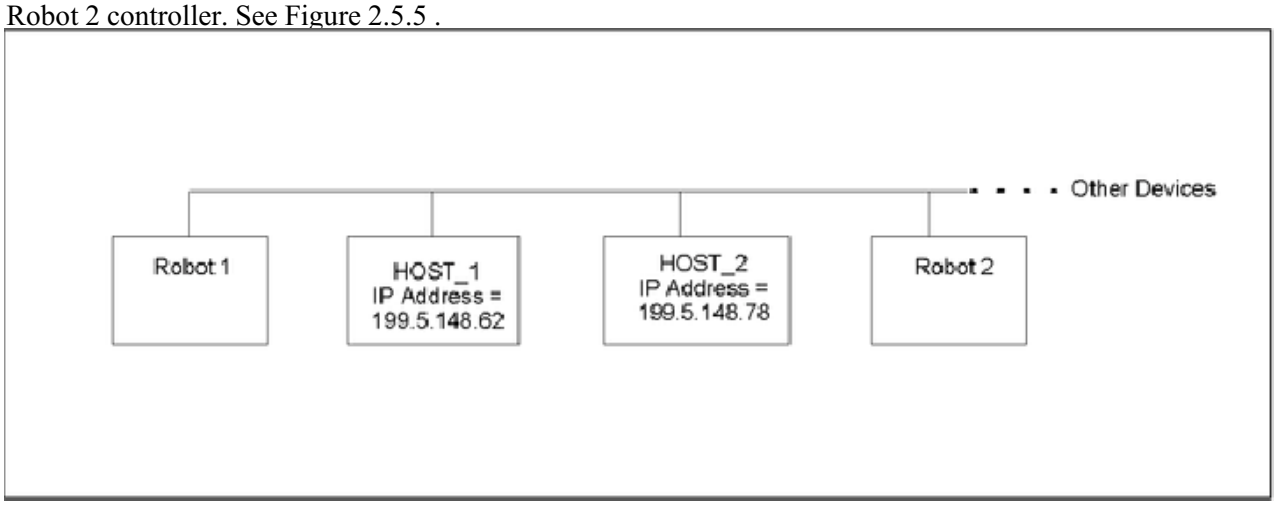



Fig. 2.5.5 Example configuration

# 3 FTP OPERATIONS

## 3.1 OVERVIEW

After you have installed and connected the appropriate Ethernet communications hardware and performed the appropriate device setup procedures, you can use FTP to communicate between the controller and other host devices.
This chapter contains information about the following FTP operations:

- Accessing client devices
- Accessing server devices
- Using the memory device (MD:) specification

## 3.2 SETTING UP AND STARTING FTP

Before you can use the FTP Interface, you must do the following:

- Define TCP/IP parameters ( Section 2.4 )
- Define FTP on a client device ( Procedure 3-1 ) if using FTP client services on the robot.
- Define and start FTP on a server device ( Procedure 3-1 ) if using FTP server services on the robot.

Table 3.2(a) lists and describes the items you must set up to define a client device. Table 3.2(b) lists and describes the items you must set up to define a server device.

NOTE
Two FTP servers are configured and started automatically. If the robot is used as an FTP server only, no further configuration of FTP is needed (TCP/IP still needs to be configured as described in Section 2.4 ).

Table 3.2(a) Client device definition setup items

| ITEM | DESCRIPTION |
|---|---|
| Tag | This item specifies the device name client. Available client tags are C1: through C8:. |
| Comment | This item provides an area for you to include up to 16 characters of information that allow you to label the device for its application use. |
| Protocol* | This item specifies the name of the protocol that will be associated with the tag. For FTP, the protocol name is FTP. |
| Port Name | This item is only displayed when SM (Socket Messaging) is selected as the Protocol, and does not apply to other protocols. |
| Startup State* | This item specifies the desired startup (Power up) state for the selected tag. Three states are possible:<br>• UNDEFINED - the device is not defined.<br>• DEFINED - the device is defined.<br>• STARTED - the device is defined and started.<br>The Startup State should normally be set to defined. When in the defined state the client tag is started automatically from the FILE screen on the teach pendant whenever it is used. |
| Server IP/Hostname* | This item specifies the Hostname or IP address of the remote server to which the connection will be made. |

| ITEM | DESCRIPTION |
|---|---|
| Remote Path/Share* | This item specifies the host path on the server, to be used for file operations, up to 64 characters. This item is case sensitive when using the FTP protocol. When using the PC Share protocol, the Share name must be included. |
| Inactivity Timeout | This item specifies the number of minutes of inactivity on the network before a connection will be closed.<br>• When set to **zero**, no timeouts occur.<br>• When set to a **non-zero value,** Inactivity Timeout specifies the number of minutes of inactivity on the network before a connection will be closed. The default value is 15 minutes. |
| Username* | This item specifies the username to use when logging into the remote server. The username is case sensitive based on the host system that checks it. |
| Password* | This item specifies the password to use when logging into the remote server. The password is case sensitive based on the host system that checks it. |

* This item is normally set up by the user. Other items can normally remain at their default values.

Use Procedure 3-1 to define and start FTP on a client device.

Table 3.2(b) Server device definition setup items

| ITEM | DESCRIPTION |
|---|---|
| Tag | This item specifies the device name server. Available server tags are S1: through S8:. |
| Comment | This item provides an area for you to include up to 16 characters of information that allow you to label the device for its application use. |
| Protocol* | This item specifies the name of the protocol that will be associated with the tag. For FTP, the protocol name is FTP . |
| Port Name | This item is only displayed when SM (Socket Messaging) is selected as the Protocol, and does not apply to FTP. |
| Startup State* | This item specifies the desired startup (Power up) state for the selected tag. Three states are possible:<br>• UNDEFINED - the device is not defined.<br>• DEFINED - the device is defined.<br>• STARTED - the device is defined and started.<br>The Startup State is normally set to Start. |
| Server IP/Hostname | This item is not used at this time. |
| Remote Path/Share | This item is not used at this time. |
| Inactivity Timeout | This item specifies the number of minutes of inactivity on the network before a connection will be closed.<br>• When set to **zero** , no timeouts occur.<br>• When set to a **non-zero value,** Inactivity Timeout specifies the number of minutes of inactivity on the network before a connection will be closed. The default value is 15 minutes. |
| Username | This item is not used at this time. |
| Password | This item is not used at this time. |

* This item is normally set up by the user. Other items can normally remain at their default values.

Use Procedure 3-1 to define and start FTP on a server device.

## Procedure 3-1 Defining and Starting FTP on a Device

### Conditions

- You have connected the Ethernet interface to a network. Refer to Section 2.2 .
- You have defined TCP/IP parameters. Refer to Procedure 2-2 .

### Steps

1 Press MENUS.
2 Select SETUP.
3 Press F1, [TYPE].
4 Select Host Comm. You will see a screen similar to the following.

```
SETUP Protocols
     Protocol       Description
  1  TCP/IP         TCP/IP Detailed Setup
  2  TELNET         Telnet Protocol
  3  PC SHARE       PC Share Setup
  4  PING           Ping Protocol
  5  HTTP           HTTP Authentication
  6  FTP            File Transfer Protocol
  7  DNS            Domain Name System
```

5. To set up a server: (if required)
   a. Press F4, [SHOW].
   b. Select 3, Servers. You will see a screen similar to the following.

```
SETUP Servers

    Tag  Protocol  Port    State
  1 S1:  ********  *****    [UNDEFINED]
  2 S2:  ********  *****    [UNDEFINED]
  3 S3:  ********  *****    [UNDEFINED]
  4 S4:  ********  *****    [UNDEFINED]
  5 S5:  ********  *****    [UNDEFINED]
  6 S6:  ********  *****    [UNDEFINED]
  7 S7:  ********  *****    [UNDEFINED]
  8 S8:  ********  *****    [UNDEFINED]
```

   c. Move the cursor to the server tag you want to set up and press F3, DETAIL. See the following screen for an example.

```
SETUP Tags

 Tag  S1:

  Comment:          ***************
+ Protocol Name:         FTP
  Current State:    UNDEFINED
+ Startup State:    START
  Server IP/Hostname:*****************
  Remote Path/Share: *****************
  Inactivity Timeout:  15 min
  Username:         *****************
  Password          *****************
```

   + These items are normally set up by the user. Other items can typically remain at their default values. A detailed description of the fields in the Setup Tags screen is given in Table 3.2(b) .
   d. To enter a comment, move the cursor to Comment and use the function keys to type a message associated with this configuration and then press ENTER. You are not required to enter a comment.
   e. Move the cursor to Protocol Name and press F4, [CHOICE]. A list of available protocol choices will be displayed.

f. Select FTP and press ENTER.
g. Move the cursor to Startup State and press F4, [CHOICE].

NOTE

By default, all tags come up in the UNDEFINED state. In general, a server should be set to the START startup state.

h. Select the startup state you want and press ENTER.
i. Move the cursor to Inactivity Timeout, type the timeout value you want, in minutes, and press ENTER. The default value is 15 minutes.
j. Press F3, LIST, to display the list of server devices.
k. Repeat Step 5.c through Step 5.j for as many server devices as you are defining.

6. To set up a client: (if required)
   a. Press F4, [SHOW].
   b. Select 2, Clients. You will see a screen similar to the following.

```
SETUP Clients

   Tag  Protocol  Remote    State
 1 C1:  ********  **********[UNDEFINED]
 2 C2:  ********  **********[UNDEFINED]
 3 C3:  ********  **********[UNDEFINED]
 4 C4:  ********  **********[UNDEFINED]
 5 C5:  ********  **********[UNDEFINED]
 6 C6:  ********  **********[UNDEFINED]
 7 C7:  ********  **********[UNDEFINED]
 8 C8:  ********  **********[UNDEFINED]
```

   c. Move the cursor to the client tag you want to set up and press F3, DETAIL. See the following screen for an example.

```
SETUP Tags

 Tag C1:

  Comment:          ****************
+ Protocol Name:         FTP
  Current State:     DEFINED
+ Startup State:     DEFINE
+ Server IP/Hostname:     192.168.1.49
+ Remote Path/Share:   robot/programs/
  Inactivity Timeout:  15 min
+ Username:                    Gary
+ Password              ************
```

   + These items are normally set up by the user. Other items can remain at their default values in most cases. A detailed description of the fields in the Setup Tags screen is given in Table 3.2(a) .
   d. Move the cursor to Comment and use the function keys to enter a message associated with this configuration. You are not required to enter a comment.
   e. Move the cursor to Protocol Name and press F4, [CHOICE]. A list of available protocol choices will be displayed.
   f. Select FTP and press ENTER
   g. Move the cursor to Startup State and press F4, [CHOICE].
   h. Select the startup state you want and press ENTER.

NOTE

By default, all tags come up in the Undefined state. In general, a client should be set to the Define startup state.

i. Move the cursor to the Server IP/Hostname field and enter the remote hostname or IP address. When a hostname is entered, this item is case sensitive and must be defined in the host name table ( Procedure 2-2 ) unless DNS is used.
j. Move the cursor to Remote Path/Share field and use the function keys to enter the remote host path. This item is case sensitive and must end with a /.
k. Move the cursor to Inactivity Timeout, type the timeout value you want, in minutes, and press ENTER. The default value is 15 minutes.
l. Move the cursor to the Username field, and type in the username for the client to use to log into the remote FTP server.
m. Move the cursor to the Password field, and type in the password for the client to use to log into the remote FTP server.
n. Press F3, LIST, to display the list of client devices.
o. Repeat Step 6.c through Step 6.n for as many client devices as you are defining.

7. To define and start FTP on a device:
   a. Press F4, [SHOW].
   b. Select Clients or Servers.
   c. Move the cursor to the client or server you want to define and start.
   d. Press F2, [ACTION].
   e. Select 1, Define.
   f. Press F2, [ACTION], again.
   g. Select Start.
   h. Repeat Step 7.c through Step 7.g for all of the client and server devices you want to define and start.

## 3.3 FTP CLIENT USERNAMES AND PASSWORDS

Each client has the capability to communicate with a different host. Therefore, it is necessary to associate a username, password, and password timer with each client. For a given client, you must set the username, password, and password timer as appropriate.
You must define a password for each username. This password allows a user who enters the username and password the ability to perform communications operations using FTP. This password is case sensitive based on the host system that checks it.
In addition to defining the password, you may set a password timer, which is the number of minutes after which the controller automatically will reset the password to "guest" and set the password timer to zero.
The default client username is **anonymous** . The default client password is **guest** . The default value of a password timer is **zero** , which means the password will not be reset.
A **username** must be from 1 to 12 characters long and must consist of letters, numbers, and punctuation that can be entered using the teach pendant. The username is case sensitive based on the host system that checks it.
A **password** must be from 1 to 12 characters long and must consist of letters numbers, and punctuation that can be entered using the teach pendant. The password is case sensitive based on the host system that checks it.

NOTE
The host computer to which you connect might have restrictions on the characters you can use in the username and password. Refer to your host computer documentation for more information.

Use Procedure 3-2 to set usernames and passwords on client devices.

NOTE
Table 3.3 defines the items needed to set up FTP client tag usernames and passwords. FTP client usernames and passwords may also be configured in the Client Tag Setup screen by following Procedure 3-1 . The password timer can only be configured by following Procedure 3-2 .

Table 3.3 FTP Setup Client Username and Password Items

| ITEM | DESCRIPTION |
|---|---|
| C1, C2 --- C8 | This item is the client tag. There are up to eight client tags available. |
| USERNAME<br>(Default: anonymous) | This item should be set to the username used to authenticate with the remote FTP server of the corresponding FTP client tag. |
| PASSWORD<br>(Default: guest) | This item should be set to the password used to authenticate with the remote FTP server of the corresponding FTP client tag. |
| TIMER (minutes)<br>(Default: 0) | This item should be set to the number of minutes after which the controller automatically will reset the password to "guest". A value of 0 indicates the password will not be reset. |

## Procedure 3-2 Setting Usernames and Passwords on Client Devices

### Conditions

- You have set up FTP. Refer to Procedure 3-1 if you have not set up the FTP client devices.

### Steps

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm. You will see a screen similar to the following.

```
SETUP Protocols
     Protocol       Description
  1  TCP/IP         TCP/IP Detailed Setup
  2  TELNET         Telnet Protocol
  3  PC SHARE       PC Share Setup
  4  PING           Ping Protocol
  5  HTTP           HTTP Authentication
  6  FTP            File Transfer Protocol
  7  DNS            Domain Name System
```

5. Move the cursor to FTP and press ENTER. See the following screen for an example.

```
SETUP Host Comm
FTP
     USERNAME    PASSWORD       TIMER
                                (minutes)
 C1 anonymous     ************     0
 C2 anonymous     ************     0
 C3 anonymous     ************     0
 C4 anonymous     ************     0
 C5 anonymous     ************     0
 C6 anonymous     ************     0
 C7 anonymous     ************     0
 C8 anonymous     ************     0
```

6. Move the cursor to the username you want to change and press ENTER. Use the appropriate function keys to type the username and press ENTER. This item is case sensitive.
7. Move the cursor to the password that corresponds to that username and press ENTER. Use the appropriate function keys to type the username and press ENTER. This item is case sensitive.
8. Move the cursor to the corresponding timer and press ENTER. Use the numeric keys to type the time and press ENTER.

9. Repeat Step 6 through Step 8 for the remaining usernames and passwords you are setting.
10. Press F3, LIST, to return to the SETUP Protocols screen.

# 3.4 ACCESSING AND USING CLIENT DEVICES

## 3.4.1 Access Description

A client device does not have to be started before it is accessed. However, the *tag* must be defined. The device automatically will be started when opened and stopped when closed, returning it to the defined state.
FTP copies files of type .CF, .KL, and .LS, as ASCII files. All other file types are transferred as binary files.

## 3.4.2 File Specification for Client Devices

Client devices are used like local file storage devices. The host communications file specification is as follows:

```
<device_name:><\\host_name\><path_name\>file_name.file_type
```

This is a modified MS-DOS format. The optional **host_name** field is an extension to MS-DOS. The **host_name** is a standard MS-DOS name from one to eight characters long. Single quotes can be used to delimit strings or characters unacceptable to MS-DOS, such as the "¥" character. The full definitions are as follows:

- **device_name** is a two- to five-character optional device name field, followed by a colon. The first character must be a letter; the remaining characters must be alphanumeric. The default device from the system console variable $DEVICE will be used if this field is absent (C1:, for example).
- **host_name** is a file name type consisting of one to eight characters. The optional **host_name** field selects the network node to receive this command. It must be preceded by two backslashes and separated from the remaining fields with a backslash. If a host_name is not present, the string specified for the Remote (Current) will be used as the default host_name. host_name must already have been defined in the host table ( Procedure 2-2 ).
- **path_name** is a recursively defined optional field consisting of one or more **file_names** separated by a backslash. It is used to select the file subdirectory. It can consist of up to a maximum of 64 characters. If a path_name is not present, the string specified for the Path (Current) will be used as the default path_name.
  The root or source directory is handled as a special case. For example, access to the subdirectory **SYS** linked off of the root would have a **path_name** of **'¥SYS'** . The **file_spec** using this **path_name** would be **C1:¥¥HOST¥'¥SYS'¥FILE.KL** .
- **file_name** is from one to eight characters. Note that **file_name** is sent over the network in lower case format, regardless of how it is entered. Therefore, upper case file names on a case-sensitive remote host cannot be retrieved.
- **file _type** is from zero to three characters.

## 3.4.3 Starting and Stopping a Client Device

Use Procedure 3-3 to start, stop, and configure the client device and to start it automatically when the controller is turned on.
Client tags can be turned on in the defined state. They will be started automatically when accessed.

### Procedure 3-3 Starting and Stopping a Client Device

#### Conditions

- The client device you want to start or stop has been defined. ( Procedure 3-1 )

### Steps

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm.
5. Press F4, [SHOW].
6. Select Clients. You will see a screen similar to the following.

```
SETUP Clients

   Tag  Protocol  Remote    State
 1 C1:  ********  **********[UNDEFINED]
 2 C2:  ********  **********[UNDEFINED]
 3 C3:  FTP       **********[UNDEFINED]
 4 C4:  ********  **********[UNDEFINED]
 5 C5:  ********  **********[UNDEFINED]
 6 C6:  ********  **********[UNDEFINED]
 7 C7:  ********  **********[UNDEFINED]
 8 C8:  ********  **********[UNDEFINED]
```

7. Press F2, [ACTION].
8. Select the action you want to perform:

> NOTE
> A device must be in the defined state before it can be started.

- To define a device, select Define.
- To undefine a device, select Undefine.
- To start a device, select Start. The device must be in the defined state.
- To stop a device, select Stop. The device will change to the defined state.

9. To configure the client device to start automatically at power up:
   a. Move the cursor to the client tag you want to start automatically and press F3, DETAIL.
   b. Move the cursor to Startup State and press F4, [CHOICE].
   c. Select Start, and press ENTER.
   The client device will now start automatically when the controller is turned on.

> NOTE
> The host device must be capable of accepting this FTP login at powerup if the tag is set to START AUTOMATICALLY when you turn the robot on. In this case, if the host is not available, the robot controller will wait approximately one minute to timeout before completing powerup. This is why it is recommended to have client tags powerup in the DEFINE state. The controller will automatically start the client tags when used.

## 3.4.4 Teach Pendant File Access

After a client device has been defined, it can be used from the teach pendant.
On the teach pendant, when you set the default device to C1:, you can do the following:

- **From the SELECT screen**
  - Save a program to C1:
  - Load a program from C1:
- **From the FILE screen**

- Generate a directory of files on C1:
- Load or restore files from C1: onto controller memory
- Back up program and system files to C1:
- Copy files to and from C1:
- Delete files from C1:

# 3.5 ACCESSING SERVER DEVICES

## 3.5.1 Overview

This section contains information about accessing server devices. A server device listens for connections that are initiated from the host computer. One server can support one connection. Therefore, you control the number of devices that are connected to the controller by starting only the appropriate number of server tags. **When no server tags have been started, no connections can be received.**
You cannot select which server devices are used for specific connections. This is determined by the TCP/IP Host Communication software.

## 3.5.2 Access Description

Server devices S1: through S8: and server-client devices that perform both functions must be started before any services can be requested. Servers are normally started when the controller is turned on and remain running while the controller is on. All host devices can be configured to start automatically when the controller is turned on via their Startup Mode.

## 3.5.3 Starting and Stopping a Server Device

Use Procedure 3-4 to start, stop, and configure the server to start automatically when the controller is turned on.

### Procedure 3-4 Starting and Stopping a Server Device

#### Conditions

- The server device you want to start or stop has been defined. ( Procedure 3-1 )

#### Steps

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm.
5. Press F4, [SHOW].
6. Select Servers. You will see a screen similar to the following.

```
SETUP Servers

   Tag  Protocol  Port    State
 1 S1:  ********  *****   [UNDEFINED]
 2 S2:  ********  *****   [UNDEFINED]
 3 S3:  FTP       *****   [UNDEFINED]
 4 S4:  ********  *****   [UNDEFINED]
 5 S5:  ********  *****   [UNDEFINED]
 6 S6:  ********  *****   [UNDEFINED]
 7 S7:  ********  *****   [UNDEFINED]
 8 S8:  ********  *****   [UNDEFINED]
```

7. Press F2, [ACTION].
8. Select the action you want to perform:

NOTE
A device must be in the defined state before it can be started.

- To define a device, select Define.
- To undefine a device, select Undefine.
- To start a device, select Start. The device must be in the defined state.
- To stop a device, select Stop. The device will change to the defined state.

9. To configure the server device to start automatically at power up:
   a. Move the cursor to the server tag you want to start up automatically and press F3, DETAIL.
   b. Move the cursor to Startup Mode and press F4, [CHOICE].
   c. Select Start, and press ENTER.
   The server device will now start automatically when the controller is turned on.

## 3.5.4 Blocking Downloads of Certain File Groups

FTP supports preventing certain file groups from being downloaded to the robot from a remote host, using the FTP server on the robot. For example, using this FTP feature, all TP programs can be prevented from being downloaded to the MD device.
This feature is disabled by default and needs to be enabled before use. To enable this feature, set $FTP_CTRL.$DNLD_FILTER = TRUE and turn the controller off and back on.
If the feature is enabled, any file that is in a special table used by FTP will be blocked from being downloaded via the robot FTP server. An FTP error, such as "501 Permission Denied," will be posted.
The table of files that can be blocked is made up of:

- MD:*.TP (all TP files in MD device)
- $FILE_APPBCK[x].$FILE_NAME (contents of this system variable array)

### 3.5.4.1 Features

- For every download request, the FTP server matches the filename with the internal table of files that are to be blocked.
- The match is device-specific and is not case-sensitive.
- Specific files or wildcards can be supplied in $FILE_APPBCK.$FILE_NAME. Device information can also be entered.
- The format for an entry in $FILE_APPBCK.$FILE_NAME is **<device>: {filename.ext} example MD:¥test.pc**
- If the device information is not entered in $FILE_APPBCK.$FILE_NAME, the MD device is assumed.
- If the feature is enabled, all teach pendant programs in MD device (MD:*.TP) are automatically blocked from being downloaded to the robot regardless of $FILE_APPBCK entries.

### 3.5.4.2 Examples

Example 3.5.4.2(a)
Setting $FILE_APPBCK[x].$FILE_NAME to SYSSEAL.SV is equivalent to setting it to MD:¥SYSSEAL.SV and blocks download of SYSSEAL.SV to MD device.

Example 3.5.4.2(b)
Setting $FILE_APPBCK[x].$FILE_NAME to "FR:*.DT" causes downloads of all .DT type files to FR device to be blocked.

# 3.6 FTP SERVICES

## 3.6.1 Overview

The following FTP services are provided:

- Environment services
- File transfer services
- Directory services

These services can be performed only by server devices.

Figure 3.6.1 shows the relationship of host communications to the controller system. It also shows the devices and the services that can be accessed.

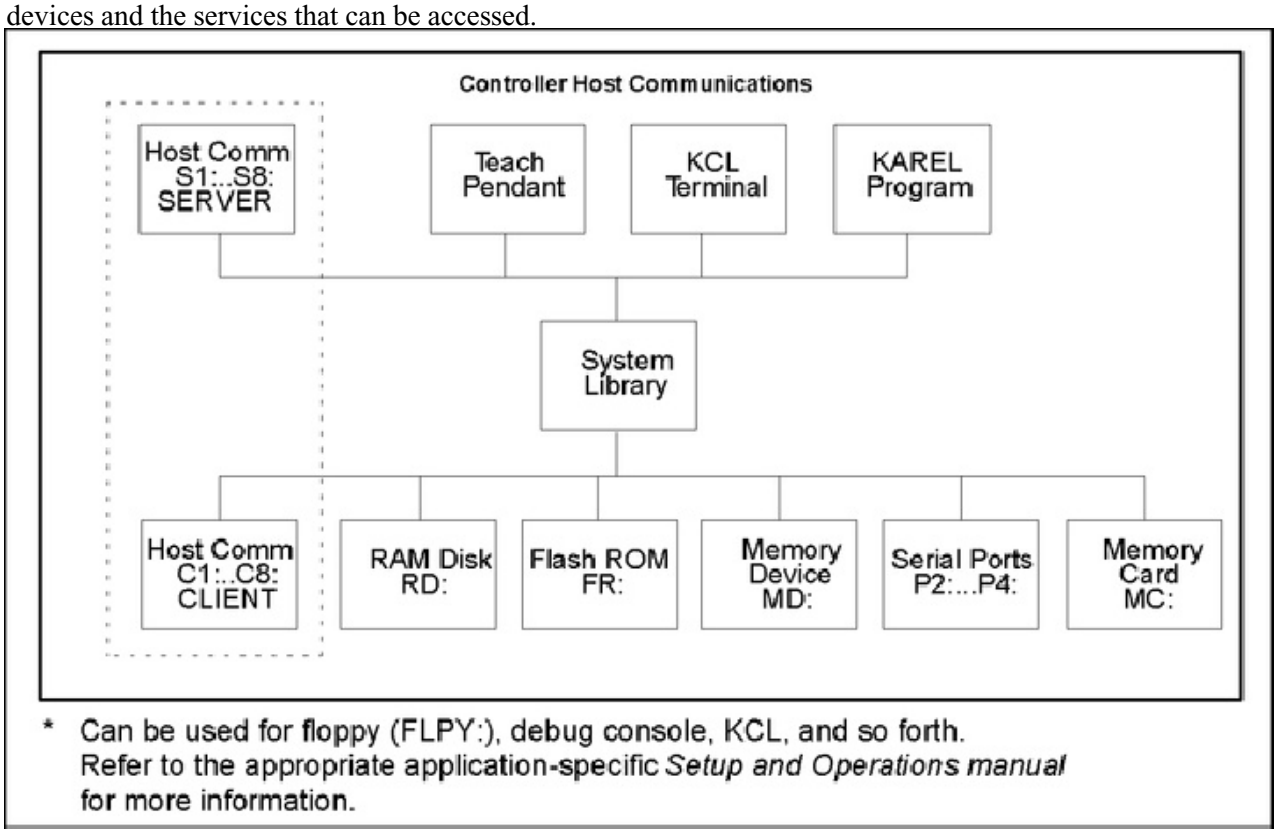



Fig. 3.6.1 Host Communications model

A host device operating as a client will make service requests via the Ethernet cable to the server. All service requests pass through the controller system library functions.

## 3.6.2 Environment Services

FTP provides the following environment services:

- open
- close
- username
- password
- type

### open

Open is used to establish a connection between the host computer and the controller.

### close

The close service is used to close a connection.

When Inactivity times out, the Close request is sent to the attached host computer. You can set the Inactivity Time to zero, which turns off the Inactivity Timer. Timer values can be set between 1 and 99,999 minutes.

### username

Username is checked if the password protection option is installed on the robot controller.

### password

Password is checked only if the password protection option is installed on the robot controller.

> NOTE
> Server passwords require the Password Protection option. The Operator level can upload files and perform a directory. The Program level can perform Operator tasks and download .TP, .PC, .IO, and .VR files. The Setup/Installation level can perform Operator and Program tasks and download .SV files. If the password protection option is not installed, you are placed in the Setup/Installation level by default. You can use the FTP Server Access control feature to modify this behavior.

### type

Type sets the file transfer type to BINARY before transferring binary type files, such as .IO, .PC, .SV, .TP, and .VR.

## 3.6.3 File Transfer Services

FTP provides the following file transfer services:

- get
- put
- mget
- mput

User program and data files can be transferred to and from I/O devices (such as the RAM disk, serial ports, and the memory device). System files can be transferred to and from the memory device only. Refer to Section 3.7 .

The FTP protocol uses the standard input and output services available in the controller. Any device accessible by a KAREL program, except client devices C1: through C8:, can be accessed.

## 3.6.4 Directory Services

FTP provides the following directory services:

- cd
- delete
- dir
- mkdir
- pwd
- rename
- rmdir

### cd

The cd service is used to change the default device.

### delete

The delete service works with devices such as P3: and FLPY:. You can delete all files except system files (such as SYSVARS.SV) with the Memory Device.

### dir

Wildcard operations are allowed for dir using the wildcard character, "*" in the file name or extension. Wildcards can be used as follows:

- **word** No wildcard. The name must match exactly.
- **word*** Matches names that begin with **word** plus zero or more characters.
- ***word** Matches names that end with **word** preceded by zero or more characters.
- ***word*** Matches names that contain **word** in the beginning, middle, or end.

> NOTE
> On some screens, the controller might display the teach pendant file attribute as a file type, such as job (.JB), macro (.MR), or process (.PR). However, these are all stored on external devices as files with the teach pendant file type extension. The controller will not allow teach pendant file names to be used with different attribute types. For example, there cannot be a TEST99.TP with attribute type job (.JB) and a TEST99.TP with attribute type macro (.MR).

### mkdir

The mkdir service allows you to create a directory. NOTE that directories cannot be created on MD: or other memory devices. Refer to Section 3.7 for more information on the memory devices.

### pwd

The pwd service is used to display the default device.

> NOTE
> If you use Distinct FTP on your host computer, the Distinct FTP client cannot change to hidden drives and cannot transfer hidden files. You can transfer only those files that are displayed in the directory of the memory device (MD:), the default device of the FTP server.

### rename

Rename is not available on the memory device (MD:) and memory card (MC:). Refer to Section 3.7 for more information on the memory device.

### rmdir

The rmdir service allows a user to remove or delete a directory. Note that directories cannot be removed from MD: or other memory devices. Refer to Section 3.7 for more information on memory devices.

## 3.6.5 Miscellaneous FTP Information

The FTP implementation on the robot conforms to Internet standard specifications (as given by RFC 959). In particular, the FTP server recognizes the internal commands listed in Table 3.6.5 .

Table 3.6.5 FTP Server Internal Commands

| | | | | |
|---|---|---|---|---|
| ABOR | LIST | PWD | PASV | MODE |
| USER | NLST | CWD | SYST | STRU |
| PASS | RETR | DELE | HELP | XPWD |
| PORT | STOR | RNFR | NOOP | XCUP |
| TYPE | QUIT | RNTO | CDUP | XCWD |
| MKD | RMD | XMKD | XRMD | |

Generally, UNIX based FTP servers are case sensitive, and the robot controller is case insensitive. When using FTP client tags to communicate with a remote UNIX FTP server, by default the robot assumes everything is lower-case. This means the robot will create directories with lower-case names, navigate directory structures assuming all directories have lower-case names, and will access files (read/write/open/create) assuming lower-case file names. When coming across a file or directory with an upper-case name, the robot will be able to display the file or directory when doing a directory listing, but will not be able to access it. Setting the system variable $FTP_CTRL.$SUBDIRCAPS to TRUE reverses this and causes the robot controller to assume all file and directory names are upper-case. When this system variable is changed, the robot controller must be power-cycled to take effect. However, note that the path entered into the Remote Path/Share field of the Client Tag Setup screen is case sensitive. This root path can be mixed case and does not assume either lower or upper case.
In general, the FTP server on the robot is compatible with any FTP client (command-line or GUI-based) that conforms to the standard FTP specification.
In particular, the FTP server has been tested against standard UNIX and Windows-based command line FTP clients and the following graphical FTP clients:

- GlobalScope Inc.'s CuteFTP Version 6.0
- IPSwitch's WS_FTP Pro Version 9.01
- FileZilla Version 3.2.3.1

For newer versions of FileZilla, use Procedure 3-5 to configure FileZilla to work with the robot controller's file system.

**Procedure 3-5 Configure FileZilla FTP Client**

**Steps**

1. Launch FileZilla, select the File menu, and open the Site Manager.
2. Enter appropriate information in the General tab.
3. Select the Advanced tab and set Servertype to DOS.
4. Optionally, enter in a default remote directory.
5. Click on OK.

# 3.7 ACCESSING USER PROGRAM, SETUP, AND DIAGNOSTIC INFORMATION

## 3.7.1 Overview

Access to user program, setup, and diagnostic information can be done over FTP using the following devices:

- MD: provides access to both ASCII and binary versions of user setup and programs along with alarm logs and diagnostic files.
- MDB: provides access to binary versions of user setup and programs (similar to "backup - all of the above" on the teach pendant file menu)
- FMD: (option) provides access to ASCII versions of user setup and programs filtered to include only user settable information (eg. internal timers or time system variables changed by the system are not included) making these files useful for detecting user changes.

When logging into the robot FTP server from a remote client you are defaulted into the MD: device. You can navigate to other robot file devices (FR:, RD:, MC:, MDB:, FMD:) using the change directory service in your remote FTP client. At a command line using the *cd* command where in this example fmd: is the device being used, this might look like :

```
D:\temp>ftp pderob029
Connected to pderob029.frc.com
220 FTP server ready.  [PaintTool Vx.xxP/01]
User <pderob029.frc.com:<none>>:
230 User logged in [NORM].

ftp>cd fmd:
250 CWD command successful.
ftp>
```

The syntax used with MD: is as follows:

```
MD:file_name.file_type
```

- **file_name** is from one to eight characters.
- **file_type** is from zero to three characters.

NOTE

Rename is not supported for MD:.

## Memory Device (MD:)

The memory device (MD:) treats the controller's program memory as if it were a file device. You can access all teach pendant programs, KAREL programs, and KAREL variables loaded in the controller.

## Memory Device Binary (MDB:)

The memory device binary device (MDB:) allows you to copy the same files as provided by the Backup function on the File Menu. This allows you to back up the controller remotely such as from SMON, FTP.
The MDB: device directory function includes only those files that should normally be backed up. When using FTP, a request to the MDB: device such as "mget *.*" (in binary mode) would provide a complete backup of the robot system and application files based on MDB: being configured correctly.
Backing up the binary portion of the memory device (MDB:*.*) provides a complete application backup (analogous to Backup — All of the above on the teach pendant file menu). It does not include the ASCII versions of programs/variables so it is smaller in size and faster to back up. This backup is appropriate for disaster recovery of the application. Note that the controller must be at CTRL start to restore most system files.

## Filtered Memory Device (FMD:)

The Filtered Memory Device option generates text versions of all backup files of user programs and variables that have been changed manually. Included are system and KAREL variables, position and data registers, teach pendant programs, and I/O configuration data.
You can compare these files with previous versions to determine what users or operators have changed. Variables and programs that change without user input are filtered out, and will appear in filter exclusion files.
After the option is installed, it will run automatically whenever you perform an Ethernet backup of the controller from the FMD: device. After you install the Filtered Memory Device option, any of the following filter exclusion files could appear on the FR: device.

⚠CAUTION

Do not delete these files, or filter exclusion data will be lost.

- FR:SVAREEG.DT
- FR:KVAREEG.DT
- FR:POSREEG.DT
- FR:REGEEG.DT
- FR:TPLINEEG.DT

Backing up the filtered memory device (FMD:*.*) provides a set of ASCII files that can be used with an application designed to do comparisons with previous FMD: backups. If differences are detected then specific files which have changed can be backed up.

## 3.7.2 System Files

System files are binary files that store default values for system variables, servo parameter data, and mastering data. They contain information specific to the controller, robot, and software.
You can access the system files listed in Table 3.7.2 by specifying the Memory Device and the reserved file names within the file access services that are supported for Memory Device.

Table 3.7.2 System files accessed through the memory device

| Kind of Information | File Specification |
|---|---|
| Frame information | MD:¥¥TPFDEF¥FRAMEVAR.VR |
| FTP Server Access Control Configuration | MD:[¥¥*SYSTEM*¥]SYSFSAC.SV |
| I/O information | MD:[¥¥*SYSTEM*¥]DIOCFGSV.IO |
| Macro command information | MD:[¥¥*SYSTEM*¥]SYSMACRO.SV |
| Mastering information | MD:[¥¥*SYSTEM*¥]SYSMAST.SV |
| Number registers | MD:[¥¥*NUMREG*¥]NUMREG.VR |
| Password variables | MD:[¥¥*NUMREG*¥]SYSPASS.SV |
| Position registers | MD:[¥¥*POSREG*¥]POSREG.VR |
| Servo parameters | MD:[¥¥*SYSTEM*¥]SYSSERVO.SV |
| Shared Hosts File | MD:[¥¥*SYSTEM*¥]SYSHOST.SV |
| System variables | MD:[¥¥*SYSTEM*¥]SYSVARS.SV |
| [ ] denotes an optional field | |

NOTE
When you perform a DIR listing of the files stored on the MD: device, you will see the system file and its ASCII version. The ASCII version of SYSVARS.SV is SYSVARS.VA, and ASCII versions can be as large as ten times the size of the binary version.

## 3.7.3 Error Log Files

Error log files are ASCII text files that provide a snapshot of the current errors in the system. They can be backed up to the default device, but cannot be restored or loaded into the controller. However, they can be imported to a spreadsheet application, such as Microsoft® Excel. Refer to Table 3.7.3 for a listing of error log files.

Table 3.7.3 Error log files

| File Name | Kind of Information |
|---|---|
| ERRALL.LS | The Error Log (All) file provides a snapshot of the history of errors in the system. |
| ERRACT.LS | The Error Log (Active Alarms Only) file provides a snapshot of active errors in the system. |

### Sample Error Log

See Figure 3.7.3 for an example of an error log entry.

Fig. 3.7.3 Sample error log entry

All of the fields of an error log file are left justified, and are delimited by double quotes (") to simplify importing the file into a spreadsheet.

### Sections of an Error Log

The first line of the error log file is called the header. It consists of the error log name, the robot hostname and the current system time and date stamp.

The next section of the file consists of a sequence number, which is an internal system number that identifies a particular error during consecutive accesses to the error log. The sequence number increases sequentially, although it need not start from 1. The other fields in this section are the time and date stamp of the error, facility name, the error code number, the error code message, the cause code message (if one exists), and the severity text.

ERRALL.LS also has a field to include the active/inactive status of the alarm. Active alarms are denoted by the text "act," and inactive alarms have a null field. Each of the fields, except the cause string field, is set to a fixed width.

## 3.7.4 FTP Transfer Log

The robot records all FTP file transfers in a special log file called FTPLOG.DG available from the MD device.

The log has the following features.

- The log file FTPLOG.DG can be accessed from the Teach Pendant, web browser or retrieved through FTP.
- The number of entries in the log (log size) can be controlled by the system variable $FTP_CTRL.$LOG_ENTRIES.
- The log can be volatile (stored in DRAM) or non-volatile (stored in CMOS). The system variable $FTP_CTRL.$LOG_CMOS controls this behavior.
- The log is a circular buffer of entries, which means that the oldest entry is removed when the log becomes full.
- Each line in the log will contain a record of a specific file transfer in the following format:
  1. Date/time stamp
  2. File operation (U)pload from robot,(D)ownload to robot
  3. Filename
  4. FTP transfer status code
  5. FTP transfer status text
  6. IP address of remote host (optional)
     To save CMOS space, the last field (IP address) is recorded only if the log is stored in DRAM.

```
APR03 2002 14:45:46 U ERRALL.LS      226 Binary Transfer complete.   172.22.192.190
APR03 2002 14:45:54 D TEST2.TP       501 Permission denied.          172.22.192.190
```

Fig. 3.7.4 IP Address

Most users can leave the default configuration which sets the log size to store 50 entries in DRAM.

# 4 DOMAIN NAME SERVICE (DNS)

## 4.1 OVERVIEW

Domain Name Service (DNS) provides a method for a robot controller to communicate with a remote server without having to know the IP address of the server.
You must do the following to be able to use DNS with your robot:

- Install and configure the network components for your Ethernet network. Refer to Section 2.2 .
- Install and configure the FTP software on the servers on your Ethernet network. Refer to Section 3.2.

### Connecting to Servers with DNS

Client side networking applications, such as an FTP client, require an IP address in order to connect to a remote server. DNS provides a way for client applications to obtain the IP address of a remote server if one cannot be found in the local or shared host tables.
When a client application initiates a connection it will first search the local and shared host tables for the IP address of the remote host. If an IP address cannot be found, then DNS will initiate a query to the local DNS server. The server will respond to the query with the IP address that the client needs. DNS will parse the response and return the IP address to the waiting client. When the client receives the needed IP address it will continue with its attempt to establish a connection to the remote server.

## 4.2 DEFINING DNS PARAMETERS

You need to provide the controller with the address of at least one DNS server for your network. The DNS client on the controller is capable of interacting with up to two DNS servers. Your network administrator can provide you with the IP addresses of the DNS servers on your network. You must also provide a local domain name.

### DNS Parameters

Several parameters are used to configure the DNS interface on your robot. Table 4.2 lists and describes the parameters you must define.

Table 4.2 DNS Parameters

| PARAMETERS | DESCRIPTION |
|---|---|
| Primary DNS Server | This item specifies the IP address of the primary DNS server on your network. This server will be contacted by the robot when it is asked to connect to a host whose IP address is unknown. Contact your network administrator for the address of your primary DNS server. DNS will not work if you do not provide the IP address of your primary DNS server. |
| Secondary DNS Server | This item specifies the IP address of the secondary DNS server for your network. This server will be contacted if your primary server is unreachable or not responding. It is not required in order for DNS to work. Not all networks have secondary DNS servers, so you should check with your network administrator to see if your network has one. |
| Local Domain Name | This item is the domain name for your local network. Examples of local domain names are frc.com or aarnet.edu.au. Your network administrator can provide you with the correct local domain name for your network.<br>NOTE<br>DNS will not work if you do not provide a local domain name. |

| PARAMETERS | DESCRIPTION |
|---|---|
| Number of Retries (1,3) | If a DNS server does not respond to a query, the robot will attempt to contact the DNS server again. The number of retries is the number of times a robot will attempt to contact a DNS server after the initial query fails. The number of retries can be set to 1, 2 or 3 retries, and the default is 2 retries. |
| Wait Time (1,7) | This item is the amount of time the robot will wait for a response from a DNS server before trying to initiate another query. You can set the wait time to be between 1 and 7 seconds. The default is 2 seconds. |

Use Procedure 4-1 to define DNS parameters.

## Procedure 4-1 Defining DNS Parameters

### Conditions

- You have installed the DNS software on your robot controllers and remote servers.

### Steps

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm. You will see a screen similar to the following.

```
SETUP Protocols
     Protocol       Description
  1  TCP/IP         TCP/IP Detailed Setup
  2  TELNET         Telnet Protocol
  3  PC SHARE       PC Share Setup
  4  PING           Ping Protocol
  5  HTTP           HTTP Authentication
  6  FTP            File Transfer Protocol
  7  DNS            Domain Name System
```

5. Move the cursor to TCP/IP and configure, if necessary. Refer to chapter of SETTING UP TCP/IP if you have not configured TCP/IP. Otherwise, go to Step 6 .
6. Move the cursor to DNS and press F3, DETAIL. You will see a screen similar to the following.

```
SETUP DNS
DNS
   Primary DNS server:   199.5.148.200
   Secondary server  :   199.5.148.201
QUERY OPTIONS
   Number of retries :    2
   Wait time         :    2

LOCAL DOMAIN NAME
**************************************
**************************************
```

7. Move the cursor to each item and specify the appropriate information:
   - Primary DNS Server - This specifies the unique address of the primary DNS server. Contact your network administrator for the address of your network's primary DNS server.
   - Secondary DNS Server - This specifies the unique address of the secondary DNS server for your network. Your network may or may not have a secondary DNS server. Contact your network administrator for the address of your network's secondary DNS server.
   - Local Domain - This specifies the domain name of your local network.
   - Number of Retries - This specifies the number of times the controller will try to contact a DNS server if its initial query is not answered.
   - Wait Time - specifies the number of seconds the client will wait before attempting another query.

NOTE

The IP addresses of the Primary and Secondary DNS servers, the Number of Retries, and the Wait Time are saved as part of SYSHOST.SV ($DNS_CFG). The local domain name is also saved as part of SYSHOST.SV ($DNS_LOC_DOM).

The SYSHOST.SV file can be shared between robots and can be downloaded to get a complete DNS configuration. In addition to DNS configuration data, the SYSHOST.SV file contains information about Telnet ($TEL_LIST) and shared hosts ($HOST_SHARED).

8. After you have entered the required information, your Domain Name Service Setup screen should look similar to the following.

```
SETUP DNS
DNS
   Primary DNS server:   199.5.148.200
   Secondary server  :   199.5.148.201
QUERY OPTIONS
   Number of retries :    2
   Wait time         :   2
LOCAL DOMAIN NAME
                         aarnet.edu.au
****************************************
```

# 5 TELNET

## 5.1 OVERVIEW

Telnet is a standard protocol designed to work between any host (such as an operating system) and any PC or UNIX terminal. The controller can function as a Telnet server. Remote hosts can use a standard Telnet client to communicate with the server.
Current functionality on the server includes the ability to create teach pendant terminals over the remote Telnet connection. The Telnet screens are under the SETUP Hostcomm menus.

NOTE
The Telnet function is a standard function. Telnet function is loaded by default with all application software packages. You must first define the TCP/IP parameters ( Procedure 2-2 ) for the robot to be active on the network.

## 5.2 SETTING UP TELNET ON YOUR ROBOT

### 5.2.1 Telnet Setup

You will need to configure the Telnet option before you can use your robot as a Telnet server. Use Procedure 5-1 to set up Telnet on your robot.
The Telnet server uses default passwords and access levels to authenticate attempts to log in. These passwords and access levels are in effect until you override them from the Telnet screen. The default passwords and access levels are shown in Table 5.2.1(a) .

Table 5.2.1(a) Telnet default passwords and access levels

| USERNAME | ACCESS LEVEL | DEFAULT PASSWORD |
|---|---|---|
| tpdisplay | Output | rj3_tpd |
| kcl | Input | uninitialized |

NOTE
Login names and passwords are case sensitive .

#### Valid Telnet Devices and Login IDs

Several parameters are used to configure the Telnet option for your robot. Table 5.2.1(b) lists and describes the valid devices and login IDs, which are also parameters you must define on the Telnet Setup Screen.

NOTE
If the robot has an *i*Pendant attached, then you cannot connect to the tpdisplay device on the controller (the login attempt will fail and an error message will be sent to the client).

Table 5.2.1(b) Telnet setup screen items

| USERNAME | DESCRIPTION |
|---|---|
| tpdisplay | This item allows you to log into the teach pendant device and displays the teach pendant output over the remote Telnet connection. |
| kcl | This item has not been supported. |
| help or ? | This item displays a help screen related to the topic you have selected. |

Table 5.2.1(c) SETUP TELNET Screen Items

| ITEM | DESCRIPTION |
|---|---|
| Username | This item is the device on the robot to which users can connect. |
| AccessValues: OUTPUT, INPUT, or NONE | This item is the access level of the device. It can be one of the following:<br>● OUTPUT - outputs from the controller<br>● INPUT - both input and output<br>● NONE - no access to the controller<br>NOTE The TP device doesn't support INPUT access. |
| Password | This item is the password that allows access to the device. To enter a password, move the cursor to this field, press ENTER, and type the password. When you are finished, press ENTER. |
| Timer<br>Units: minutes<br>Range: 0 - 99<br>Default: 0 | This item is an inactivity timeout value. It indicates the number of minutes of inactivity over the TELNET connection before the robot closes the connection. |

Use Procedure 5-1 to set up the Telnet option.

## Procedure 5-1 Setting up Telnet on Your Robot

### Conditions

- You have configured the Ethernet hardware and software on your robot. Refer to Procedure 2-2 .

### Steps

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm. You will see a screen similar to the following.

```
SETUP Protocols
      Protocol          Description
  1   TCP/IP            TCP/IP Detailed Setup
  2   TELNET            Telnet Protocol
  3   PC SHARE          PC Share Setup
  4   PING              Ping Protocol
  5   HTTP              HTTP Authentication
  6   FTP               File Transfer Protocol
  7   DNS               Domain Name System
```

5. Move the cursor to TELNET and press F3, DETAIL. You will see a screen similar to the following.

```
SETUP Telnet
Username      Access    Password    Timer

TP            OUTPUT    ********      0
KCL           OUTPUT    ********      0
CONS          OUTPUT    ********      0
```

6. You can set up passwords and access levels only if you do not want to use the defaults. The timer field is disabled by default (0). If a positive value is set, it determines the number of minutes of inactivity on the connection before the connection is terminated.
   With the SETUP Telnet screen displayed, press F5, HELP. You will see a screen similar to the following.

```
SETUP Telnet
HELP    Arrows to scroll, PREV to exit
TELNET HELP SCREEN
  ACCESS
change the access level of the device, OUTPUT - Outputs
from the controller.
INPUT - Both input and output.
NONE - No access to the controller
The TP device doesn't support input access
```

## 5.2.2 Connecting to a Telnet Server

After you have set up the Telnet feature, you can use it to connect to a Telnet server. Use Procedure 5-2 to connect to a Telnet server.
More security measures, in addition to passwords, are available to control remote access into the robot. Telnet supports the FANUC Server Access Control (FSAC) feature, which decides which remote hosts (PCs) are allowed to connect into the robot. Refer to Section 2.5 for more information on setting up FSAC for Telnet.

### Procedure 5-2 Connecting to a Telnet Server

#### Steps

1. From your PC or UNIX workstation, start a standard Telnet client window, or from a command prompt type the following:

```
C:\>telnet <robothost>
```

Where <robothost> is the host name or IP address of the robot to which you want to connect.

2. After a Telnet connection has been established, you will see the following message on the screen of your PC or UNIX workstation:

```
RJ3 Telnet (Robot: <robothostname> F No: F-xxxxx)
Login:
```

3. From your PC or UNIX workstation, type a valid login name for the device to which you want to connect and press ENTER. Refer to Table 5.2.1(a) for a list of valid login names.
4. Type your password and press ENTER.
5. If you have entered a valid login ID and password, your PC or UNIX workstation will be connected to the device selected in Step 3 .

> NOTE
> Login names and passwords are case sensitive.

# 6 WEB SERVER

## 6.1 OVERVIEW

The *web server* application allows you to access files on the robot using a standard web browser. This includes files on the robot memory device (MD:), as well as other file devices on the robot such as FR: and RD:. The memory device includes error logs, diagnostic data, and ASCII translations of system and program variables. The server can also be customized by including a unique home page.
The main purpose of the web browser is to provide easy access to robot programs and status information.

NOTE
You must first define the TCP/IP parameters ( Procedure 2-2 ) for the robot to be active on the network.

## 6.2 SETTING UP THE WEB SERVER

### 6.2.1 Overview

The web server is a standard feature. The default method for using web server is to have it configured to start automatically when the controller is turned on (it is available at Controlled start mode as well as during normal operation). At this time, configuration of the web server is done directly through system variables. Refer to Table 6.2.1 for the web server system variables and their descriptions.

Table 6.2.1 Web server system variables

| SYSTEM VARIABLE | DESCRIPTION |
|---|---|
| $HTTP_CTRL.$ENABLE | This variable automatically starts the web server when the controller is turned on if the value is greater than 0 (the default value is 1). Reset this variable to zero if you would like to disable the web server when you turn the controller on again. |
| $HTTP_CTRL.$KRL_TIMOUT | This variable defines the maximum number of seconds to wait for a KAREL program to complete which is requested through the web server. Refer to section on “Running KAREL Programs from the web browser. The default value is 10 seconds. |
| $HTTP_CTRL.$HITCOUNT | This variable is incremented each time the web server gets a request. This variable can be modified at any time if, for example if you want to reset the hitcount to 0. This is an integer variable that will roll over at the maximum value (2147483646). |
| $HTTP_CTRL.$BG_COLOR | This variable is the default web page background color (FANUC yellow). It is used in the default header and trailer files. |
| $HTTP_CTRL.$ENAB_TEMPL | This variable indicates whether the HTTP (Web Server) task should use a template file for headers and trailers on any DG/LS/VA files. The default value is 1 (enabled). |

| SYSTEM VARIABLE | DESCRIPTION |
|---|---|
| $HTTP_CTRL.$TEMPLATE | This variable will override the system defined template for LS/DG/VA files if $ENAB_TEMPL is enabled (set to 1). Template files effect the header and trailer HTML around these files so will effect their look on a browser. Note that a query string can also be used to force a particular template for these file types. This variable should not include an extension as this variable really represents two files - the header and trailer. As an example, if $TEMPLATE=FR:MYTEMP, then there should be two files on FR: (FR:MYTEMP.HDR, FR:MYTEMP.TLR). The system template is FRS:DEFAULT. |
| $HTTP_CTRL.$COMMENT | This variable is an available comment field. It can be used in web pages by referencing it directly. This can be changed by the user as desired. |

### 6.2.2 Using FANUC Server Access Control (FSAC) to Control Access to the Web Server

You can use the FANUC Server Access Control (FSAC) feature to control access to the web server. Note that an access level of Program level or above is required to utilize the KAREL/Server Side Include feature within web server, based on the configuration of FSAC. An access level of Operator level or above is required to access other files from the web server.
Access to the *i*Pendant screens is also controlled by FSAC. If $UI_CONFIG.$READONLY[2]=TRUE, then all levels have read-only access. If $UI_CONFIG.$READONLY[2]=FALSE, then the Operator and User-defined levels have read-only access, the Program level will have access to screens used for programming the robot, the Setup level will have access to screens used to set up the system, and the Install level will have read-write access to all the screens.
Refer to Section 2.5 for more information on the FSAC feature.

## 6.3 USING THE WEB SERVER

### 6.3.1 Overview

After you have set up the web server ( Section 6.2 ) , you can use it to connect to a robot's home page, where you can access system variable, teach pendant, error/diagnostic, and binary files.

### 6.3.2 Connecting to a Robot Home Page

The default home page for the robot is a listing of important diagnostic files and links.
The default home page provides a link to the memory device file list (MD:INDEX.HTM). This list is built dynamically each time the page is requested based on the programs and variables loaded in working memory.
The following example URLs (either Figure 6.3.2(a) or Figure 6.3.2(b) ) requests the robot default home page shown in Figure 6.3.2(c) .

```
http://robotname -- if robotname is the name of the robot you want to connect to,
and it is known on the network.
```

Fig. 6.3.2(a) URL example

```
http://192.168.0.1 -- if the robot name is not known on network
```

Fig. 6.3.2(b) URL example

Fig. 6.3.2(c) Default robot home page

The link on the default page called "Active Programs /Variables /Diagnostics (Memory Device)" is the memory device file list (MD:INDEX.HTM) and is shown in Figure 6.3.2(d) .

Fig. 6.3.2(d) Memory device index page

The links at the top of the page shown in Figure 6.3.2(d) are defined in Table 6.3.2 .

Table 6.3.2 Program/Diagnostic link descriptions

| LINK TITLE | DESCRIPTION |
|---|---|
| Variable Files | This link points to a section of this page that provides links to ASCII and binary versions of any .SV file and any .VR file which is loaded (on memory device). |
| TP Program Files | This link points to a section of this page that provides links to ASCII and binary versions of any .TP program loaded on the robot. |
| Error/Diagnostic Files | This link points to a section of this page that provides links to ASCII versions of diagnostic files such as the complete alarm log (errall.ls), the active alarm log (erract.ls), a snapshot of the I/O (iostatus.ls), or a listing of loaded software with memory status and servo information (errcurr.ls, errhist.ls). |

## 6.3.3 Customizing Your Robot Home Page

A customized home page can be loaded to replace the default home page. The file FR:INDEX.HTM will be shown (if it exists on your robot controller) in place of the default home page.
The web server currently is able to return the following kinds of files:

- HTML (.htm extension on robot)
- JPEG (.jpg extension on robot)
- GIF
- TXT
- WAV
- .LS
- .VA
- STM (See Note listed below.)
- PNG
- CLS (Java class files

NOTE
.LS and .VA files are returned with a simple HTML header and trailer appended. Other kinds of files are returned as binary files with a "Content-type" of "application/octet-stream".

If FR:INDEX.HTM is loaded on the controller, it should have a link to the memory device index page (MD:INDEX.HTM). The following code is an example of a link to the memory device INDEX:

<A href=" /md/index.htm"> Program / Diagnostic Files </A>

If the Web Server Enhancements Option is loaded, then the order of files searched to be used as the robot home page is as follows:

- FR: INDEX.HTM
- FR: INDEX.STM
- FRS: INDEX.HTM (internal use- application tool-specific home page)
- FRS: INDEX.STM (internal use - application tool-specific home page)
- FRS: DEFAULT.STM (initial default home page)

NOTE
.STM files are part of the Web Server Enhancements Option support. These are supported on user devices (such as FR:, MC:, and RD:) only if this option is installed.

### General URL Syntax

The general URL syntax to access various files on the robot is :

Example 6.3.3 (a) General URL Syntax

```
http://<robot>[/device]/<filename>
```

The area of the URL indicated by "robot" above is where the name or IP address of the robot is placed. The "device" is optional but corresponds to physical devices on the robot (such as MC, MD, FR, RD). No colon ":" is included in the device identifier within the URL. The "filename" is the actual file to retrieve. An example URL including the device is:

Example 6.3.3 (b) Example URL

```
http://robot1/rd/mypage.htm
```

## 6.3.4 Customizing Diagnostic Files, Variable File Listings, and TP Program Listings

You can customize the way internally generated files are displayed in a browser. Internally generated files are diagnostic files, variable file listings, and teach pendant program listings (anything with an extension of .DG, .LS, or .VA). These files are plain text files with a simple HTML header and trailer added so they display as web pages. You can modify the HTML header and trailer sent in order to change the way these pages look in the browser.
A very simple HTML header might be:

Example 6.3.4 (a) Simple HTML Header

```
<HTML><BODY><PRE>
```

A very simple HTML trailer might be:

Example 6.3.4 (b) Simple HTML Trailer

```
</PRE></BODY></HTML>
```

The above HTML header and trailer are what is sent if $HTTP_CTRL.$ENAB_TEMPL is set to 0. The actual header also includes a META tag to indicate NOCACHE to the browser since these files are generated dynamically each time they are requested :

Example 6.3.4 (c) Header

```
<HTML>
<HEAD><META HTTP-EQUIV=\"PRAGMA\"CONTENT=\"NO-CACHE\"></HEAD>
<BODY><PRE>
```

Example 6.3.4 (d) Trailer

```
</PRE></BODY>
<META HTTP-EQUIV=\"PRAGMA\" CONTENT=\"NO-CACHE\">
</HTML>
```

The system variable $HTTP_CTRL.$ENAB_TEMPL causes a system level dynamic header and trailer to be applied to any .DG/.LS/.VA file when served through the web server. The default value is ENABLED. The system header file used is FRS:DEFAULT.HDR. The system trailer file is FRS:DEFAULT.TLR. These files use "server side include" syntax. This functionality can be disabled by setting $HTTP_CTRL.$ENAB_TEMPL to 0.

Example 6.3.4 (e) $HTTP_CTRL

```
$HTTP_CTRL.$ENABLE Access: RW INTEGER = 1
$HTTP_CTRL.$ENAB_DIAGTP Access: RW INTEGER = 0
$HTTP_CTRL.$ENAB_SMON Access: RW INTEGER = 0
$HTTP_CTRL.$ENAB_SPART Access: RW INTEGER = 0
$HTTP_CTRL.$DBGLVL Access: RW INTEGER = 0
$HTTP_CTRL.$KRL_TIMOUT Access: RW INTEGER = 10
$HTTP_CTRL.$HITCOUNT Access: RW INTEGER = 0
$HTTP_CTRL.$BG_COLOR Access: RW STRING[25] = 'FFF9E3'
$HTTP_CTRL.$ENAB_TEMPL Access: RW INTEGER = 1
$HTTP_CTRL.$TEMPLATE Access: RW STRING[25] = 'FRS:DEFAULT'
$HTTP_CTRL.$COMMENT Access: RW STRING[25] = 'FANUC Web Server
```

The system variable $HTTP_CTRL.$TEMPLATE can be used to define custom header and trailer files. A typical application might be to copy FRS:DEFAULT.HDR to FR:NEWLOOK.HDR and FRS:DEFAULT.TLR to FR:NEWLOOK.TLR, and then modify these two files as desired.

NOTE

The filename (minus extension) of the header and trailer file must be the same. If $HTTP_CTRL.$ENAB_TEMPL is set to 1, and $HTTP_CTRL.$TEMPLATE is set to "FR:NEWLOOK" then the modified files will be used. The filename defined in $HTTP_CTRL.$TEMPLATE does not include an extension.

NOTE

The header and trailer are processed dynamically with the results held internally and the size limited to 4KB each. This is the size of the results of the header and trailer files after any server side include directives have been processed. If either the header and trailer fail to process successfully, the static default header and trailer (shown above) are used.

The system variable $HTTP_CTRL.$BG_COLOR can be used within any server side includes. To affect the background color of the web pages, use the following syntax:

Example 6.3.4 (f) Changing the background color of web pages

```
<BODY bgcolor= #<!-- #echo var=$http_ctrl.$bg_color -->
```

Refer to Section 6.4 for more information about Server Side Includes.
A specific custom header and trailer can be applied to any .DG/.LS/.VA file on the robot by including the template name in the query string. The web server looks for the name "_TEMPLATE" and, if found in the query string, will apply the associated value as the template for that request. For example, to request MD:SUMMARY.DG with a custom header/trailer, the following URL could be issued:
http://my_robot/md/summary.dg?_template=fr:my_templ
This implies that FR:MY_TEMPL.HDR and FR:MY_TEMPL.TLR exist. If either file does not exist, or if there are processing errors, the static (internal) header and trailer are used.

## 6.3.5 Running KAREL Programs from the Web Browser

KAREL programs that do not include any motion can be run from the web browser. The KAREL program must include the %NOLOCKGROUP directive. This capability allows a KAREL programmer to generate a response.htm file based on the execution of the program using data gathered at execution time. A typical example would be generating a production report.
To use this feature write a KAREL program and compile it with the robot version used, and load it on the controller. Use the following guidelines when writing this program:

- The KAREL program can access any program or system variable.

- Use of condition handlers and delays is not recommended, because the program must complete within $HTTP_CTRL.$KRL.TIMOUT. Access to files can be done if it is completed within this time out.
- The program must create a properly formatted HTML file called RD: RESPONSE.HTM for display at the browser. This file is the feedback from running the KAREL program. An understanding of HTML formatting is needed in order to write this kind of program.
- Beginning in V6.22 there is a new device called TD: used for temporary files such as response.htm. Use TD:RESPONSE.HTM for any new applications using KAREL programs and the web server.

Refer to Section 6.5 to setup access to KAREL programs through the web server.

## 6.3.6 Creating Web Pages Based on KAREL Programs

This section contains information about how you can integrate KAREL programs into your robot home page, and how you can use your web browser to pass parameters to a KAREL program. Example 6.3.6(a) through Example 6.3.6(c) in this section contain an example KAREL program (demo.kl) to provide an example of one way you can use KAREL programs to access system variables on your robot from a remote web browser.

Example 6.3.6 (a) Demo.kl -- Example File Access Program

```
%nolockgroup
CONST
  HDR = 'HTTP/1.0 200 OK, request succeeded'
  NAK = 'HTTP/1.0 404 File Not Found'
  SERVER_ERR = 'HTTP/1.0 500 Server Error'
  NOSUPPORT = 'HTTP/1.0 503 Service Unavailable'
  ERRHDR = '<HTML><BODY><P><H2>'
  ERRTRLR = '</H2></BODY></HTML>'
  HDRHTML = 'Content-type: text/html'
  HDRTEXT = 'Content-type: text/plain'
  HDRJPEG = 'Content-type: image/jpeg'
  HDRGIF = 'Content-type: image/gif'
  HDRBIN = 'Content-type: application/octet-stream'
  HDRWAV = 'Content-type: audio/basic'
  DEFAULTFILE = 'INDEX.HTM'
  DEFDEV = 'FR:'
  SCRHDEV = 'RD:'
  SYSDEV = 'FRS:'
  TEXTHDR = '<HTML> <BODY> <PRE>'
  TEXTTRLR = '</PRE> </BODY> </HTML>'
  GETDHDR = '<HTML> <BODY> <H2> Get_data: </H2><BR><BR><OL><LI>'
  GETDTRLR = '</LI></OL><BR> </BODY> </HTML>'
  PAGEHDR = '<HTML> <BODY> <H2> Post data: </H2><BR><BR>'
  PAGETRLR = '<BR> </BODY> </HTML>'
-- graphics and forms used in MD_FILES.HTM
  BACKGROUND = 'FRS/EARTHBG.GIF'
  PIC1 = 'FRS/HLINE.GIF'
VAR
  count1 : integer
  count2 : integer
  file1 :  FILE
  entry : integer
```

These declarations in the KAREL program invoked by the browser will give the KAREL program access to the complete URL (if less than 128 bytes) for debugging and fill in the variable Textbox1 with the data from "Textbox1" from the form..
Note that checkboxes are only sent from a form on the browser if they are checked. Forms can be configured to always send the checkbox value as "false" in a hidden field first, or the KAREL program can always reset the KAREL variable to the default state at the end of the KAREL program. Both methods are shown in the example in Figure 6.3.6(a) through Figure 6.3.6(b) .

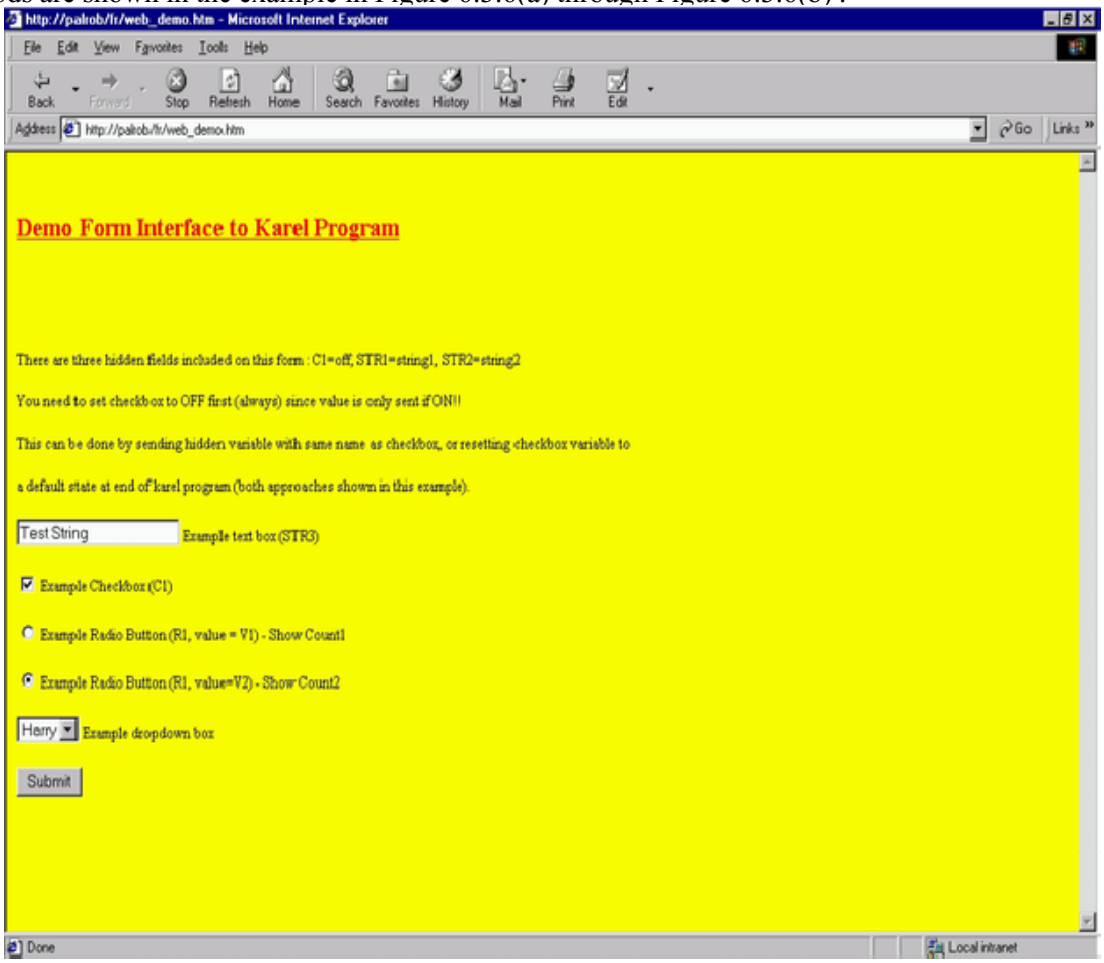



Fig. 6.3.6(a) Example KAREL based web page using parameters

Example 6.3.6 (f) Demo Form Interface to a KAREL Program

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<meta name="GENERATOR" content="Microsoft FrontPage 2.0">
<title>Web Demo</title>
</head>
<body bgcolor="#FFFF00">
<p>&nbspj </p>
<p><font color="#FF0000" size="5"><strong><u>Demo Form Interface
to Karel Program</u></strong></font></p>
<p>  </p>

<form action="http://palrob/karel/web_demo" method="GET" name="uif_demo">
    <input type="hidden" name="C1" value="OFF">

    <input type="hidden" name="str1" value="string1">
    <input type="hidden" name="str2" value="string2">
    <p>&nbspj </p>
    <p>There are three hidden fields included on this form: C1=off, STR1=string1,
```

Example 6.3.6 (g) Example KAREL program

```
-- Example karel program to respond to a form called web_demo.htm created in
-- frontpage.  Note that form data is populated in corresponding karel
-- variables, IF variables are declared.  A string variable called URL should
-- be declared to see exactly what is provided from browser which is useful
-- for debugging.
--
-- Example of received URL :
--
WEB_DEMO?STR1=STRING1&STR2=STRING2&STR3=STRING3&C1=ON&R1=V1&D1=JIMS&B1=SUBMIT
--
-- NOTE : variables which are included in URL are populated each time
-- the program is called.  Some form variables (eg. checkbox) are only
-- sent if they are checked.  This behavior can be handled by always
-- passing a "hidden" variable of same name with default value from
-- form, or by resetting variables with this nature to a default state
-- after program runs (see c1 variable assignment at end of this program).
-- Program variables are uninitialized the first time a program runs
-- (aside from ones which are set by URL, since any variables included in
-- URL are set before program is called).
--
PROGRAM web_demo
%nolockgroup
CONST
  TEXTHDR = '<HTML> <BODY>'
  TEXTTRLR = '</BODY> </HTML>'
  BACKGROUND = 'FRS/EARTHBG.GIF'  -- used in MD_FILES.HTM
  PIC1 = 'FR/PICTURE.GIF' -- some picture for top of response file
VAR
  count1 : integer
  count2 : integer
  file1 :   FILE
  URL : string[128]
  str1 : string[12]
  str2 : string[12]
  str3 : string[12]
  c1 : string[12]
  r1 : string[12]
  d1 : string[12]
```

Example 6.3.6 (h)

```
BEGIN
-- Good practice to check for uninitialized variables before using
--them
  if uninit(count1) then count1 = 0; endif
  if uninit(str1) then str1 = ''; endif
  if uninit(str2) then str2 = ''; endif
  if uninit(str3) then str3 = ''; endif
  if uninit(c1) then c1 = ''; endif
  if uninit(r1) then r1 = ''; endif
  if uninit(d1) then d1 = ''; endif
  if uninit(URL) then url = ''; endif
  count1 = count1 + 1 -- these might be production counts from another program
  count2 = count1 * 2 -- they are just included as examples
  OPEN FILE file1 ('RW', 'RD:RESPONSE.HTM')
  write file1('<HTML><HEAD><TITLE>WEB_DEMO.HTM</TITLE></HEAD>',cr)
  write file1('<BODY BACKGROUND="../')
  write file1(BACKGROUND)
  write file1('">',cr)
-- Could add a graphic to top of response file
--   write file1('<CENTER> <H1><A NAME="TOP"><IMG SRC="../')
--   write file1(PIC1,cr)
--   write file1('" WIDTH="400" HEIGHT="100"></A></H1> </CENTER>',cr)
--   write file1('"></A></H1> </CENTER>',cr)
```

```
  write file1('<H1><CENTER><BOLD>Results of form request :',cr,cr)
  write file1('</CENTER></H1>',cr)
-- checkbox only sent if checked so send default state always
  write file1('<H2><CENTER><BOLD>Received c1 (hidden) : ')
  write file1(c1,cr)

  write file1('</BOLD></CENTER></H2>',cr)
```

Example 6.3.6 (i)

```
  write file1('<H2><CENTER><BOLD>Received str3 : ')
  write file1(str3,cr)
  write file1('</BOLD></CENTER></H2>',cr)
  if (c1='ON') then
    write file1('<H2><CENTER><BOLD>Received Checkbox : ')
    write file1(c1,cr)
    write file1('</BOLD></CENTER></H2>',cr)
  endif
  write file1('<H2><CENTER><BOLD>Received Radio button : ')
  write file1(r1,cr)
  write file1('</BOLD></CENTER></H2>',cr)
  write file1('<H2><CENTER><BOLD>Received dropdown box : ')
  write file1(d1,cr)
  write file1('</BOLD></CENTER></H2>',cr)
  write file1('<H2><CENTER><BOLD>Received URL : ')
  write file1(URL,cr)
  write file1('</BOLD></CENTER></H2>',cr)
  if (r1='V1') then
    write file1('<H2><CENTER><BOLD>Count1 value is : ')
    write file1(count1,cr)
    write file1('</BOLD></CENTER></H2>',cr)
  else
    write file1('<H2><CENTER><BOLD>Count2 value is : ')
    write file1(count2,cr)
    write file1('</BOLD></CENTER></H2>',cr)
  endif
-- If default value of checkbox is not sent as hidden variable, another
-- alternative is to reset checkbox variable to default state after
-- program runs.  As with all karel programs, global variables retain
-- their value between each execution
   c1 ='OFF'
   write file1(TEXTTRLR,cr)
   CLOSE FILE file1
END web_demo
```

# 6.4 SERVER SIDE INCLUDES

## 6.4.1 Overview

The FANUC Robotics web server and server side include (SSI) directives allow you to access web pages on the robot. This provides dynamic information to clients. Such information can include the current value of a program variable (part count, for example), current status of an I/O point, or the current error listing.

SSI directives are directives placed into an HTML file that are replaced by the data they reference each time the file is requested. This allows dynamic data to be included with web pages that are served from the robot controller. SSI capability is included as part of the web server enhancements software option.

It is important to understand that the web server replaces SSI requests with the results of the request before the web page is sent to the client browser. The FANUC Robotics web server will only do this for files on the robot with a .STM file extension. The .STM file will include normal HTML syntax and might also have server side include requests, which must be fulfilled before the page is sent to the client.

The .STM file extension is the indicator to the web server that the file needs to be processed before it is sent to the requestor (client browser).
The following directives are supported through the robot server side include mechanisms.

- Echo - the value of any system variable, program variable, or I/O point
- Exec - request to run a (non-motion) KAREL program. The result of the request is included in what is sent to the browser.
- Include - includes any file in the current file (for example, MD:ERRALL.LS for current error listing).
- If - conditional logic to determine whether blocks of HTML code are included in what is sent to the browser or not (for example, if the robot is faulted display certain things; otherwise display other things.)
- Set - each page can have up to 15 local variables which can be used for display or logic.
- Printenv - diagnostic directive to display values of global and local variables.

The file device called RAM DISK (RD:) on the robot is used as a temporary storage device for .STM file responses. The RD: device must be available and have sufficient space for the response in order for any request to be successful.

## 6.4.2 Syntax

SSI directives are entered as HTML comments. This means that they are placed within HTML comment delimiters. The general syntax of a SSI directive is: <!--#command parameter="argument" -->
where "command" can be one of the following:

- echo: e.g.<!--#echo var="version" -->
- include: e.g.<!--#include file="md:errall.ls" -->
- exec: e.g.<!--#exec cmd="Karel/getdata" -->
- set: e.g.<!--#set var="_ginum" value="$hosts_cfg[1].$tim eout" -->
- if: e.g.<!--#if expr="tpout[1] = on" -->
- elif: e.g.<!--#elif expr="_lvar1 =_lvar3" -->
- else: e.g.<!--#else -->
- endif: e.g.<!--#endif -->
- printenv: eg.<!--#printenv -->

Each line (up to 200 characters long) is processed separately for a .STM file. If the HTML comment delimiters are found and the first character within the comment is a "#" then the comment is interpreted as an SSI directive and an attempt is made to process it as such.
SSI directives cannot be split between lines. The entire command must be placed on a single line. Multiple commands can be used within a single line.
The result of the SSI directive is placed in the response sent to the client browser in place of the SSI directive. There are certain characters that have special meaning within a SSI directive:

- curly bracket: ("{","}") - refer to Section 6.4.5 on string substitution.
- square brackets: ("[","]") - used on the robot to delimit program names and I/O port numbers.
- dollar sign ("$") - used to indicate system variables on the robot.
- underscore ("_") - as the first character of an expression indicates a local/global variable.
- spaces/quotes/equal/# (" ","", "=", "#") - most commands are parsed based on these characters so improper usage will cause errors.

For example, consider the following file called example.stm and placed on the robot FR: device:

Example 6.4.2 (a) example.stm

```
<html>
<head><title>Example SSI file</title></head>
<body>

The value of gpin[1] is <!--#echo var="gpin[1]" -->
</body>
</html>
```

The file is sent to the browser in response to http://<robotname>/fr/example.stm is:

Example 6.4.2 (b) File Sent to Browser Resulting from example.stm

```
<html>
<head><title>Example SSI file</title></head>
<body>
The value of gpin[1] is 3
</body>
</html>
```

This example assumes the value of gpin[1] was 3 when the request was received by the robot web server. If sometime later the value was 5, then the resulting file sent to the browser would indicate that the value was 5 (the SSI directive is evaluated on each request as it occurs).

## 6.4.3 Global Variables

The following global variables are available for use:

- _TIME
- _DATE
- _REMOTE_IP
- _DOC_NAME
- _QUERY_STR
- _URL

The _TIME and _DATE variables provide the current time/date as set on the robot controller.

Example 6.4.3 (a) Time and Date Global Variables

```
<!--#echo var="_TIME" -->        results in        17:36:40
<!--#echo var="_DATE" -->        results in        yy/mm/dd
```

The _REMOTE_IP variable is the IP address of the browser making this request. For example, a request from the browser with an IP address is 192.168.0.1 would have this variable set as follows:

Example 6.4.3 (b) Remote IP Address Global Variable

```
<!--#echo var="_REMOTE_IP" -->        results in        192.168.0.1
```

The _DOC_NAME variable is the name of the document requested. For example, a request from the URL: http://<robot>/fr/example.stm would result in the following:

Example 6.4.3 (c) Document Name Global Variable

```
<!--#echo var="_DOC_NAME" -->        results in        /fr/example.stm
```

The _QUERY_STR variable will be the portion of the URL requested which is after the "?". This indicates data in the request. For example, a request for the URL: http://<robot>/fr/example.stm?myvar=12 would result in the following:

Example 6.4.3 (d) Query String Global Variable

```
<!--#echo var="_QUERY_STR" -->        results in        _myvar=12
```

> **NOTE**
> In the example listed above, a local variable called _myvar would also be set to the value 12. Refer to Section 6.4.4.

The _URL variable contains the entire request as received by the robot. This variable might be useful in debugging. It will be surrounded by HTML preformatting specifiers (<PRE>, </PRE>).

The command #PRINTENV will print out all local and environment variables. It is also useful in debugging. It will be surrounded by HTML preformatting specifiers (<PRE>, </PRE>) also.

## 6.4.4 Local Variables

Each file processed for SSI directives can have up to 15 local variables. These variables must be set each time the file is processed (each time a request is made for the file). A local variable has a name. The name can be up to 12 characters and must start with an underscore ("_"). Local variables also have a value. All local variables are string variables and can be up to 40 characters in length.
Local variables can be set in two ways:

- #SET: eg.<!--#set var="_reqvar" value="$VERSION" -->
- Query String example: http://<robot>/fr/example.stm?_reqvar=$VERSION.

NOTE
The local variable _reqvar is set to the string "$VERSION" in the above examples.

The query string can be part of a request from a client browser. This might typically be created based on providing a HTML form and a submit button. The submit button can make the request and pass the arguments from the form as parameters. If the request is for a .STM file part of initializing the request is to set any variables within the query string which have names beginning with the underscore (other variables within the query string are ignored in terms of setting local variables).
For example, consider the following file called example.stm and placed on the robot FR:device:

Example 6.4.4 (a) example.stm

```
<html>
<head><title>Example SSI file</title></head>
<body>
<!--#set var="_reqvar" value="$VERSION" -->
The value of _reqvar is <!--#echo var="_reqvar" -->
</body>
</html>
```

The file sent to the browser in response to http://<robotname>/fr/example.stm is:

Example 6.4.4 (b) File Sent to Browser Resulting from example.stm

```
<html>
<head><title>Example SSI file</title></head>
<body>
The value of _reqvar is $VERSION
</body>
</html>
```

## 6.4.5 String Substitution

The curly bracket characters ("{","}") are used to indicate that string substitution is required within a SSI directive. The curly brackets indicate that the value of the variable be substituted in the expression. For example, if the local variable _reqvar is equal to $VERSION, then the expression {_reqvar} is equal to Vx.xx where x.xx corresponds to the most recent software version..
Another example to consider is the file called example.stm and placed on the robot FR: device:

Example 6.4.5 (a) example.stm

```
<html>
<head><title>Example SSI file</title></head>
<body>
The value of <!#echo var="_reqvar" --> is <!--#echo var="{_reqvar}" -->
</body>
</html>
```

The file sent to the browser in response to http://<robotname>/fr/example.stm?_reqvar=$version is:

Example 6.4.5 (b) File Sent to Browser Resulting from example.stm

```
<html>
<head><title>Example SSI file</title></head>
<body>
The value of $VERSION is "V6.xx  02/13/xxxx"
</body>
</html>
```

NOTE
In this case, the request could have been generated through a form where any variable is input and the value is echoed back.

## 6.4.6 #ECHO Command

The #ECHO command will replace the argument with the current value of the argument. The argument can be any system variable, program variable, I/O point, local variable, or global variable. The current value of the argument will replace the SSI directive in the response sent to the client browser.

NOTE
Digital I/O values will show as "ON" or "OFF."

### Examples

The following illustrate various uses of this SSI directive:

- <!--#echo var="DIN[1]" --> is replaced by ON
- <!--#echo var="GPIN[2]" --> is replaced by 3
- <!--#echo var="[myprog]partcount" --> is replaced by 72
- <!--#echo var="_1var1" --> is replaced by Fault#1
- <!--#echo var=$numreg[3] --> is replaced by 22
- <!--#echo var="GPIN[_stylenum]" --> is replaced by 8

In each case the argument is evaluated for string substitutions based on curly/square brackets before the final value is placed in the response to the client browser. Also, the I/O points must be configured on the robot.

The I/O type must be one of the following:

- Digital Types (return value is ON/OFF):
  - DIN /*digital input*/
  - DOUT /*digital output*/
  - PLCIN /*PLC input*/
  - PLCOUT /*PLC output*/
  - RDI /*robot digital input*/
  - RDO /*robot digital output*/
  - BRAKE /*brake output*/
  - SOPIN /*operator panels input*/
  - SOPOUT /*operator panels output*/
  - ESTOP /*emergency stop*/
  - TPIN /*teach pendant digital input*/
  - TPOUT /*teach pendant digital output*/
  - WDI /*weld inputs*/
  - WDO /*weld outputs*/
  - UOPIN /*user operator's panel input*/
  - UOPOUT /*user operator's panel output*/

- LDIN /*laser DIN
- LDOUT /*laser DOUT*/
- WSIN /*weld stick input*/
- WSOUT /*weld stick output*/

- Analog/Group Types (return value is the numeric value of the port):
  - GPIN /*grouped inputs*/
  - GPOUT /*grouped outputs*/
  - ANIN /*analog input*/
  - ANOUT /*analog output*/
  - LANIN /*laser AIN*/
  - LANOUT /*laser AOUT*/

## 6.4.7 #INCLUDE Command

The #INCLUDE command places other files from the robot in the current response to the browser. Many files on the controller are generated upon request (such as MD:ERRACT.LS for an active alarms) so these included files can also include dynamic data.

NOTE
Other .STM files can be included and these will also be processed for SSI directives.

The following examples illustrate various uses of the SSI directive:

Table 6.4.7 SSI Directives Examples

| Directive | Description |
|---|---|
| <PRE><!—#INCLUDE FILE=“MD:ERRALL.LS” —></PRE>. | Is replaced by contents of MD:ERRALL.LS |
| <pre><!--#include file="md:errall.ls" --></pre> | Is replaced by ASCII listing of abortit.tp |
| <!--#include file="fr:¥somefile.stm" --> | Is replaced by results of fr:¥somefile.stm |

The HTML preformat specifier is needed when the requested file is not structured as an HTML document. This is because of items such as carriage returns are not interpreted within an HTML document.
There are two considerations to nesting .STM files:

- Nesting is currently allowed to three levels.
- The local variables in one .STM file are not available to another .STM file (even when nested). Global variables are always available and include the query string from the initial request.

## 6.4.8 #EXEC Command

The #EXEC command allows non-motion KAREL programs to be run within processing of the .STM file. The results of these commands are automatically placed in the response to the client browser.

NOTE
The Server Side Include feature uses the same mechanism and follows the same rules but enables the capability within the .STM file processing using the #EXEC command.

### Examples

The following example demonstrates this capability:

Example 6.4.10 (a) example.stm

```
<html>
<head><title>Example SSI file</title></head>
<body>
<!--#if expr="tpout[1] = on" -->
        <P><STRONG>ROBOT IS FAULTED!</STRONG></P>
        <pre><!--#include file="md:erract.ls" --></pre>
<!--#ELSE -->
        <P><STRONG>ROBOT IS NOT FAULTED!</STRONG></P>
<!--#ENDIF -->
</body>
</html>
```

The file sent to the browser in response to http://<robotname>/fr/example.stm (assuming robot is not faulted based on teach pendant fault LED being off) is:

Example 6.4.10 (b) File Sent to the Browser Resulting from example.stm

```
<html>
<head><title>Example SSI file</title></head>
<body>
<P><STRONG>ROBOT IS NOT FAULTED!</STRONG></P>
</body>
</html>
```

## 6.4.11 #PRINTENV Command

The #PRINTENV command is useful for debugging. It outputs all the local and global variables each time it is called. This can be a quick way to identify problems with data being passed into the .STM file through a URL request. Or it can help with problems with handling local variables.

NOTE
The HTML preformat specifier is recommended for more readable results.

### Examples

The SSI directive <PRE><!--#PRINTENV --></PRE>will return the following result:

Example 6.4.11 Result of #PRINTENV SSI Directive

```
_MYVAR : 12
_LVAR1 : MYPROG
_URL : /fr/example.stm?_myvar=12 HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg,
application/vnd.ms-excel, application/msword, application/vnd.mspowerpoint,
*/*
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 5.0; Windows NT; DigExt)
Host: remora
Connection: Keep-Alive
_DOC_NAME : /fr/example.stm
_QUERY_STR : _myvar=12
_REMOTE_IP : 192.168.0.1
_TIME : 16:14:44
_DATE : 01/02/19
```

## 6.4.12 SSI EXAMPLES

See Example 6.4.12(a) for SSI directive examples.

the resource can be accessed. If the resource is locked, then an HTTP Forbidden error (403) is returned indicating that no access is allowed.

NOTE
The only HTTP authentication method supported is BASIC . Basic uses base 64 encoding method for HTTP Authentication.

The HTTP Authentication feature also applies to any external requests through the web server. It does not apply to any requests from the local *i*Pendant web browser.
The following resources can be authenticated:

- *i*Pendant (this expands internally to FRH:¥CGTP¥CGTP.HTM) web server access — This entry limits access to teach pendant screens from your browser. This functionality requires either the web enhancement option or internet connectivity and customization options are loaded.
- KAREL:DEMO— This entry limits access to the demo.pc KAREL program through your browser.
- FR:*.HTM — This entry limits access to any files with a HTM extension on the FR: device.

NOTE
Wild cards can be used within the resource description. However wildcard expansion is limited to an entire field (device, path, name and extension). The first matching entry between the actual request and the protected resource list will apply. This request matching is not case sensitive (but names and passwords are case sensitive).

The HTTP authentication feature is used within the robot controller password option. If the password option is enabled, then the HTTP authentication uses the names and passwords configured within the password option and the associated access levels. If the robot controller password option is not enabled, then the names and passwords are local to HTTP authentication.

## 6.5.2 Operation

### 6.5.2.1 Overview

HTTP Authentication is configured through the HTTP Authentication SETUP screen. This can be found under the SETUP Menu by choosing **Host Comm** .
The following resources require authentication by default.

- iPendant
- KAREL:*
- KCL:*

Refer to Table 6.5.2 for information on the HTTP SETUP Screen Items.

NOTE
Changes to the SETUP screen take effect immediately.

Table 6.5.2.1 HTTP SETUP Screen Items

| ITEM | DESCRIPTION |
|---|---|
| Resource Indicator<br>Values: L, U, or A | This item indicates whether the resource is set to<br>• **L** ocked - No access is allowed<br>• **U** nlocked - Unlimited access is allowed<br>• **A** uthenticate - Name and passwords are required |
| Name | This item is the username. This item is displayed only when the password option is not installed. |
| Pwrd | This item is the password field. This item is displayed only when the password option is not installed. |

| ITEM | DESCRIPTION |
|---|---|
| LevelValues: OPERATOR, PROGRAM, SETUP, or INSTALLDefault: INSTALL | This item is the level associated with the user. It must be at least equivalent to the level set for HTTP authentication of that particular resource. This item is displayed only when the password option is installed. Values can be:<br>• OPERATOR<br>• PROGRAM<br>• SETUP<br>• INSTALL (default)<br>Usernames and passwords that are configured from the password option SETUP screen are used to authenticate the user, and the level field indicates the required minimum level necessary to access the associated resource. |
| Resource | This item indicates the resource. |

### 6.5.2.2 Robot controller password option not enabled

If the controller password option is not enabled, then the HTTP Authentication Setup screen is shown. If the Resource is set to (A)uthenticate then the name and password must match. Names and passwords are limited to 6 characters and are **case sensitive** .

NOTE
The name and password must be set before any resource requiring authentication can be accessed.

```
HTTP Authentication Setup Screen (controller password
option not active)
HTTP Setup
   PROTECTED RESOURCES
  Name  Pwrd  Resource
A ***** ***** iPendant
A ***** ***** KAREL:*
A ***** ***** KCL:*
A ***** ***** **********************
A ***** ***** **********************
A ***** ***** **********************
A ***** ***** **********************
A ***** ***** **********************
```

### 6.5.2.3 Robot controller password option enabled

If the controller password option is enabled then the HTTP Authentication Setup screen shown below will be displayed. See the following screen for an example. If the Resource is set to Authenticate then the name and password entered must correspond to a user defined within the password option setup screens. The level associated with that user must be at least equivalent to the levels set for HTTP Authentication of that resource.
The default level set for all resources is Install. This can be changed by importing a password configuration file from the SETUP Passwords menu.

```
HTTP Authentication Setup Screen (controller password
option active)
HTTP Setup
   PROTECTED RESOURCES
  Resource
A iPendant
A KAREL:*
A KCL:*
A **********************
A **********************
A **********************
A **********************
A **********************
```

## 6.5.2.4 Example configuration

The following example configuration will allow unrestricted access to all files on the FR: device with the .HTM extension., but require authentication for any other files on the FR: device. Since the first match in the list applies, any requests that match FR:*.HTM will use the configuration associated with this item, while other requests to FR: will use the configuration for the FR:*.* item.

NOTE
You must use the UNLOCK setting for FR:*.HTM and the AUTH setting for FR:*.*

```
HTTP Authentication Setup Screen with an Example
Custom Configuration
HTTP Setup
   PROTECTED RESOURCES
  Name  Pwrd  Resource
A*****  ***** iPendant
A*****  ***** KAREL:*
A*****  ***** KCL:*
U*****  ***** FR:*.HTM
A*****  ***** FR:*.*
A*****  ***** **********************
A*****  ***** **********************
A*****  ***** **********************
```

## 6.5.2.5 Accessing *i*Pendant screens through the web server

The robot *i*Pendant screens can be accessed through the robot web server using one of the following URLs:

- http://myrobot/frh/cgtp/echo.htm (non-interactive TP display)
- http://myrobot/frh/cgtp/cgtp.htm (interactive TP display, independent TP session)

Access to cgtp.htm requires a password to be configured for the *i*Pendant resource. By default, all screens are read-only. To enable access, set $UI_CONFIG.$READONLY[2]=FALSE.
Refer to the “Advanced *i*Pendant Functions ” appendix in this manual for more information.

# 7 PROXY SERVER

## 7.1 OVERVIEW

### 7.1.1 Operation of Proxy Server

The proxy server on the robot allows you to browse web servers on the network from the *i*Pendant. For the browser on the *i*Pendant to be able to view web servers on the network, it needs a proxy server to *proxy* web requests from the *i*Pendant to the remote server. The proxy server gets the response from the remote server and forwards it to the browser.
The proxy server operates in three different modes:

- **Mode 1:** Allows access to all web servers on the building network.
- **Mode 2:** Allows access to limited web servers on the building network.
- **Mode 3:** Allows access to all web server on the building network and access to the internet using the building proxy server.

In the first mode (the default when proxy server option is loaded on the robot), a user can access all web servers on the building network from the *i*Pendant. In the second mode, a user has restricted access to web servers on the building network. The servers have to be explicitly specified. Wildcard filtering is allowed.
The third method can be used when internet access from the building network is allowed using a building proxy server ( contact your Information Systems department for details for your building proxy server.) The proxy server on the robot can be set up so that it uses the building proxy server for internet access. You can specify all the web servers that have direct access and no building proxy is required.

NOTE

The *i*Pendant only supports the Basic (base 64 encoding) method for HTTP Authentication. If the building proxy server requires authentication, a pop-up window appears on the *i*Pendant for you to enter the name and password.

### 7.1.2 Requirements for Using Proxy Server

The proxy server is available for use only by the web browser on the *i*Pendant. It cannot be used from Ethernet or PPP Serial/Modem connections.
When browsing a particular web server, the proxy server needs to resolve names to IP addresses. So, the DNS (Domain Name Server) Client option is required. If the DNS option is not installed, you must make sure the web server name (used in the URL for the web browser) is present in the host entry table.

## 7.2 CONFIGURATION OF PROXY SERVER

By default, when the proxy server option is installed, it is ready for use and works in mode 1. In order for mode 2 or 3 use the following procedure.

Procedure 7-1 Installing the Proxy Server Option

1. Press MENUS
2. Select Setup.
3. Press F1, [TYPE], and select HOSTCOMM. You will see a screen similar to the following.

NOTE

You might have to go to the next page of the menu to see this option.

```
SETUP Protocols
     Protocol       Description
  1  TCP/IP         TCP/IP Detailed Setup
  2  TELNET         Telnet Protocol
  3  PROXY          Proxy Server
  4  PPP            Point to Point Protocol
  5  PING           Ping Protocol
  6  HTTP           HTTP Authentication
  7  FTP            File Transfer Protocol
  8  DNS            Domain Name System
```

**4.** Scroll to the **PROXY** protocol and press F3, DETAIL. You will see a screen similar to the following.

```
Proxy/Setup
External Proxy
  Enable : FALSE
  Server : *************************
  Port :   8080

Exceptions:
  1) *********************************
  2) *********************************
  3) *********************************
  4) *********************************
  5) *********************************
  6) *********************************
  7) *********************************
  8) *********************************
```

**5.** To operate in mode 2 (allow limited access to web server on the building network), leave (External Proxy) Enable to be FALSE. Scroll to the Exceptions and enter the host names that you want to allow an *i*Pendant user to access. You can enter wildcard at the beginning or the end of the entry. If no wildcards are used, an exact match is performed. Some examples are *.yahoo.com, 192.168.0.*, www.fanucrobotics.com. In the first case, all host names starting with 192.168.0 will be allowed. In the third case, an exact match for the hostname will be performed.

**6.** To operate in mode 3 (allow access to external web sites through a building proxy server and full access to web servers on the building network), change (External Proxy) Enable to be TRUE. Enter the external proxy server name or IP address (you can obtain this from your Information Systems department). The default port on the external proxy server is 8080 (you are able to change that if necessary). For all the web servers that are to be accessed directly from the robot without contacting the external proxy server, enter the names that would be used in the URL in the *Exceptions* list. For these entries, the robot will contact the web server directly.

NOTE

The Exception list uses string compare for the URL and the exception. It does not resolve the IP address for blocking or redirecting requests.

## 7.3 ERRORS RETURNED BY THE PROXY SERVER

The Proxy Server returns any errors due to configuration to the web browser. The Proxy Server specifically returns the following errors.

- HTTP 400 — Bad Request: The request was not in the expected form. The expected form is http://hostname/...
- HTTP 403 — Forbidden: You are operating in mode 2 and were trying to browse a web server that was not in the exception list.

- HTTP 414 — Request URI Too Long The request (http://hostname/..) was longer than 4 Kbytes. The proxy server can handle requests only up to 4 Kbytes long. The content length can be any size but the URI can only be 4 Kbytes long.
- HTTP 500 — Internal Server Error: There was a problem opening connections as the system is out of resources.
- HTTP 502 — Bad Gateway: The hostname in the web request could not be resolved to an IP address. If you are using an external proxy server, the IP address does not match. Or, the web server you are trying to get does not respond. Verify that you have the DNS option installed or you have the hostname of the web server being used in the URL in the host entry table.

The remote web server or the external proxy server might return one or more of these errors. The errors are standard HTTP errors specified by the RFC documents for the HTTP Protocol. You can contact your Information Systems department if you have any questions regarding these HTTP errors.

# 8 POINT-TO-POINT PROTOCOL CONNECTIVITY

## 8.1 OVERVIEW

Point-to-Point Protocol (PPP) allows devices to connect to each other across a dedicated point to point link.
The controller supports up to one user PPP connection via a serial port or with a modem installed in your controller. All internet options, except Ethernet Image Backup and Restore and BOOTP/DHCP, are available for devices to use over the PPP link.

## 8.2 SETTING UP PPP ON YOUR CONTROLLER

### 8.2.1 Overview

Point-to-Point Protocol (PPP) allows for simple point-to-point connections between network devices that exchange data. PPP allows a PC or other network device to establish a simple point-to-point network connection to your controller either directly through the P2 or P3 serial ports, or through an external modem connected to one of the available serial ports.
You can make remote dial-in PPP connections to your robot, either through external modems installed on the P2 or P3 serial ports.

#### IP Addresses

Table 8.2.1(a) and Table 8.2.1(b) show the default IP Addresses for the P2 and P3 ports.

Table 8.2.1(a) Addresses for P2 Port (Direct Serial Port or External Modem)

| ITEM | IP ADDRESS |
|---|---|
| Robot | 1.1.2.10 |
| Remote (PC) | 1.1.2.11 |
| Subnet Mask | 255.255.255.0 |

Table 8.2.1(b) Addresses for P3 Port (Direct through Serial Port or External Modem)

| ITEM | IP ADDRESS |
|---|---|
| Robot | 1.1.3.10 |
| Remote (PC) | 1.1.3.11 |
| Subnet Mask | 255.255.255.0 |

If possible, you should use the default values in these tables. However, if you need to use different IP addresses for your Robot and Remote device, the IP addresses can be modified by using Procedure 8-3 .

NOTE
If your robot is connected to an Ethernet network, you need make sure that the IP addresses for the PPP connections of both the robot and the remote device are on the same subnet, and that the subnet is different from the Ethernet subnet you are using for your robot.

### Supported Modems

The following external modems are supported:

- US Robotics Sportster, 56K Faxmodem with 2x
- US Robotics Sportster, 28,800 Fax Modem with V.34 and V.32bis

## 8.2.2 Configuring the P2, and P3, Ports

You can configure ports P2 and P3 on the controller to be used as direct serial PPP connections, or you can connect an external modem to ports P2 and P3. Refer to Procedure 8-1 to configure port P2 or P3 for direct serial port connections to your network. Refer to Procedure 8-2 to set up port P2 or P3 for external modem connections to your network.

### Procedure 8-1 Setting up Port P2 or P3 as Direct Serial Port Connections

### Steps

1. Cold start the controller.
   a. On the teach pendant, press and hold the SHIFT and RESET keys. Or, on the operator panel, press and hold RESET.
   b. While still pressing SHIFT and RESET on the teach pendant (or RESET on the operator panel), turn on the power disconnect circuit breaker.
   c. Release all of the keys.
2. Press MENUS.
3. Select SETUP.
4. Press F1, [TYPE].
5. Select Port Init. You will see a screen similar to the following.

```
SETUP Port Init
     Connector   Port    Comment
   1 JRS16 RS-232-C  P2:  [No use     ]
   2 JD17  RS-232-C  P3:  [No Use     ]
```

6. Move the cursor to the port you want to set up, either P2 or P3. Press F3 DETAIL. You will see a screen similar to the following.
7. Move the cursor to Device, and press F4, [CHOICE].
8. Move the cursor to PPP and press ENTER.

> NOTE
> The default and maximum supported baud-rate for the serial connection is 19.2 KB/Sec.

9. If the teach pendant does not show any messages, the port has been initialized for PPP. If the port setting was not displaying a No Use message, turn the controller off, and then on again.

### Procedure 8-2 Setting up Port P2 or P3 for an External Modem

### Steps

1. Cold start the controller
   a. On the teach pendant , press and hold the SHIFT and RESET keys. Or, on the operator panel , press and hold RESET.
   b. While still pressing SHIFT and RESET on the teach pendant (or RESET on the operator panel), turn on the power disconnect circuit breaker.
   c. Release all of the keys.
2. Press MENUS.
3. Select SETUP.
4. Press F1, [TYPE].
5. Select port Init, and press ENTER. You will see a screen similar to the following.

```
SETUP Port Init
    Connector   Port    Comment
  1 JRS16 RS-232-C   P2:  [No use       ]
  2 JD17  RS-232-C   P3:  [No Use       ]
```

6. Move the cursor to the port you want to configure, either P2 or P3, and press F3, DETAIL. You will see a screen similar to the following.

```
SETUP Port Init
PORT B       P3:
  1 Device           [No Use         ]
  2 Speed(Baud rate)            [19200]
  3 Parity bit                 [None ]
  4 Stop bit                  [1bit   ]
  5 Time out value(sec)           [   0]
```

7. Move the cursor to Device and press F4, [CHOICE].
8. Move the cursor to Modem/PPP and press ENTER.

> NOTE
> The default and maximum supported baud rate for serial connections is 19.2 KB/sec.

9 Turn the controller off, and then on again for the changes to take effect.

## 8.2.3 Changing IP Addresses

When assigning IP addresses to ports P2 and P3 you should use the default values listed in Table 8.2.1(a) and Table 8.2.1(b). However, if you need to use different IP addresses for your robot or remote device, the default IP addresses can be changed by performing Procedure 8-3 .

### Procedure 8-3 Changing the Default IP Addresses

#### Conditions

- You have performed a Cold start on your controller

#### Steps

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE]
4. Select Host Comm.
5. Move the cursor to PPP.
6. Press F3, DETAIL.
   You will see a Port initialized for PPP or PPP/Modem message.
7. Press F3, DETAIL. You will see a screen similar to the following.

```
SETUP PPP Port
   P3

Peer IP address  : 1.1.3.11
Robot IP address : 1.1.3.10
Subnet mask      : 255.255.255.0
```

8. Change the IP addresses and the subnet mask as desired.

# 8.3 SETTING UP PPP ON YOUR PC

## 8.3.1 Overview

You can configure your network PC for a Remote Access Server (RAS) dial-up connection. You can establish the dial-up connection to network devices either directly through a serial port. Use Procedure 8-4 to configure the RAS Software on your PC. For detailed information about how to add a dial-up connection to your PC, refer to the operating system software manual for your PC's operating system, or contact your network administrator.

NOTE
RAS is a component of Windows NT/98/2000.

## 8.3.2 Setting up PPP on a Network PC

You should configure your PC for PPP connection. This manual contains step by step instructions on how to configure PC with Windows 2000 and Windows XP operating systems. Refer to Procedure 8-4 , Procedure 8-5 , and Procedure 8-6 .

### Procedure 8-4 Setting Up PPP on a PC with Windows 2000

**1.** Click **Control Panel**, **Network and Dial Up Connections**, and **Make a New Connection**. You will see a screen similar to the following.

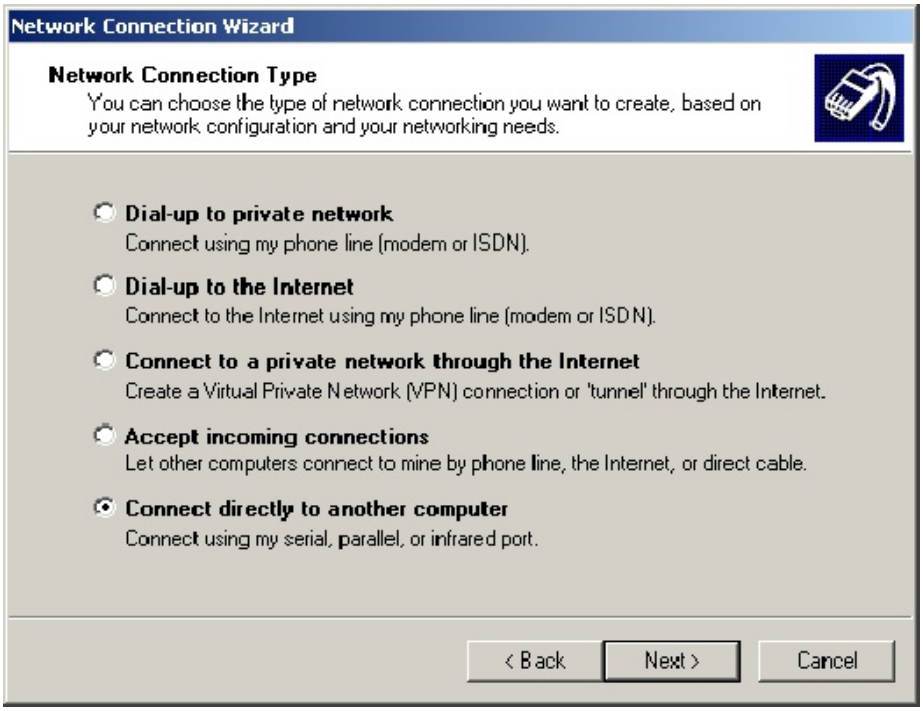



Fig. 8.3.2(a) Control Panel

**2.** In the Network Connection Wizard, select **Connect directly to another computer** and click Next. You will see a screen similar to the following.

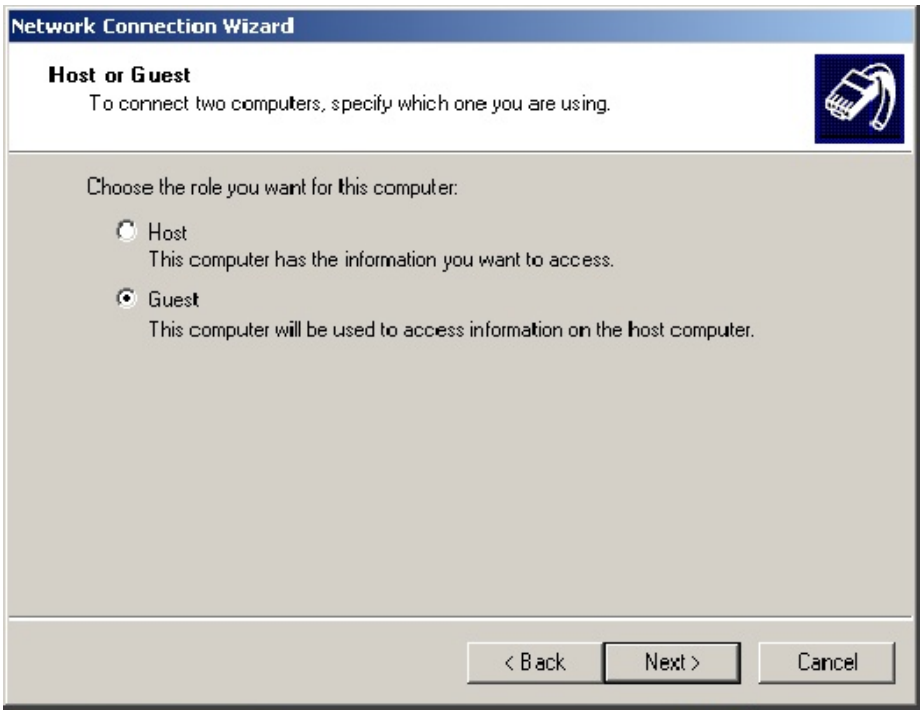



Fig. 8.3.2(b) Network Connection

**3.** Select **Guest**. The robot controller will be the host and you must click Next to continue. You will see a screen similar to the following.

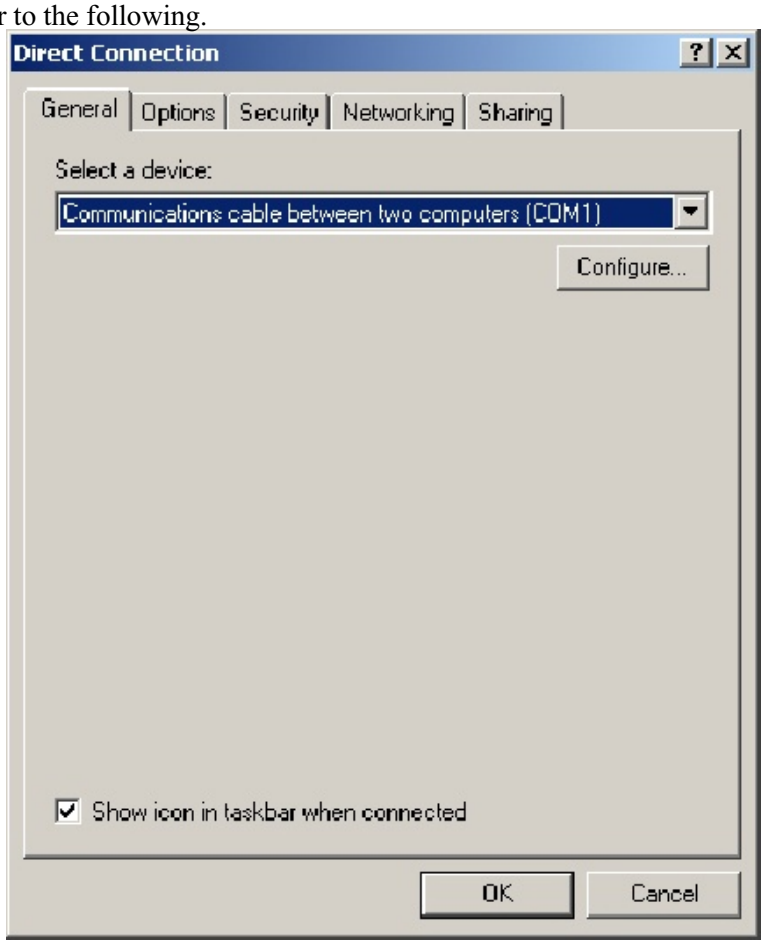



Fig. 8.3.2(c) Guest

**4.** In the **Select a Device Screen,** select **Communications cable between two computers (COMx),** where x is the COM port you will be using for your connection. Click Next to continue. You will see a screen similar to the following.

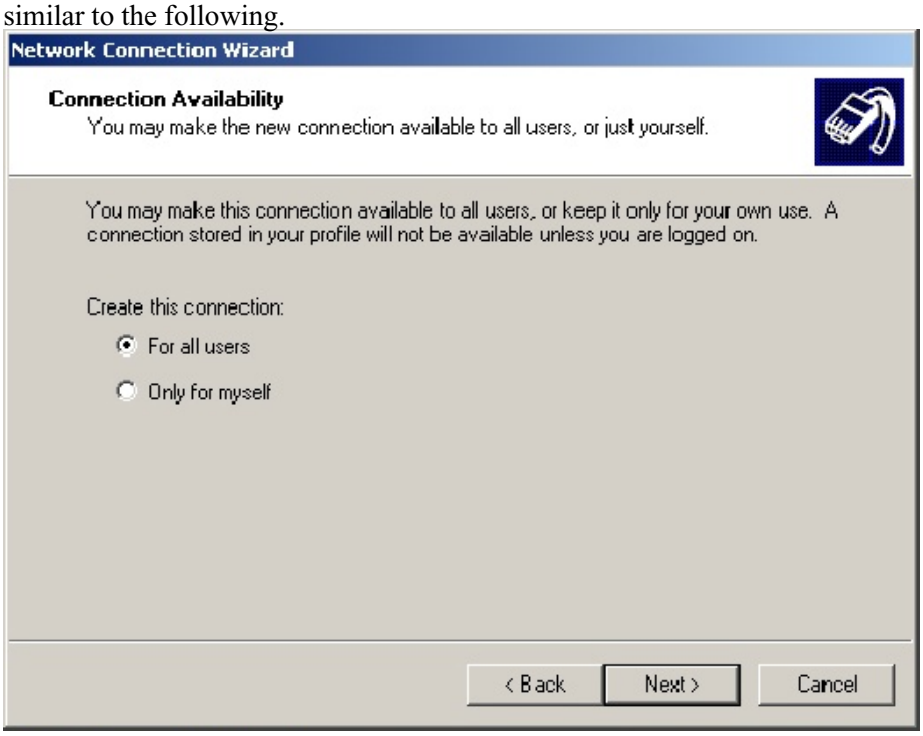



Fig. 8.3.2(d) Select a Device

**5.** If you want all users who log on to your PC to use this connection, select **For all users** and click Next. You will see a screen similar to the following.

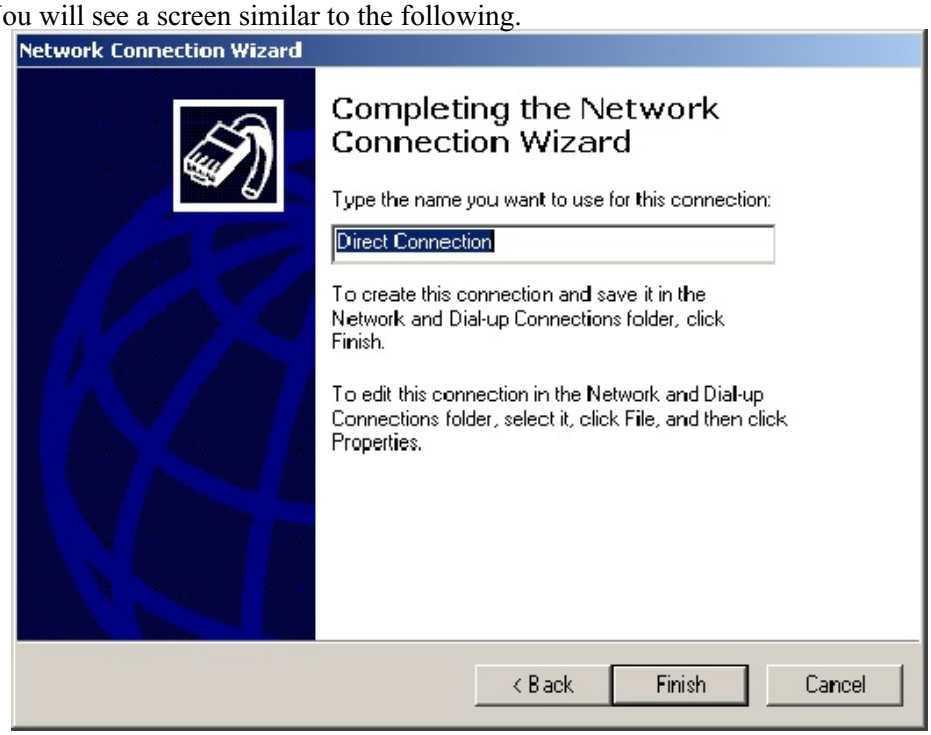



Fig. 8.3.2(e) For All Users

**6.** Type a name for this connection. Select **Finish**. You will see a screen similar to the following.

Fig. 8.3.2(f) Select Finish

A user name and password is not required for this connection. You can leave them blank or ignore the boxes.

**7.** Select **Properties**.

**8.** Select Configure and choose the proper baud rates. Select **OK** to return to the following screen.

Fig. 8.3.2(g) Configure Properties

**9.** Click Next. You will see a screen similar to the following.

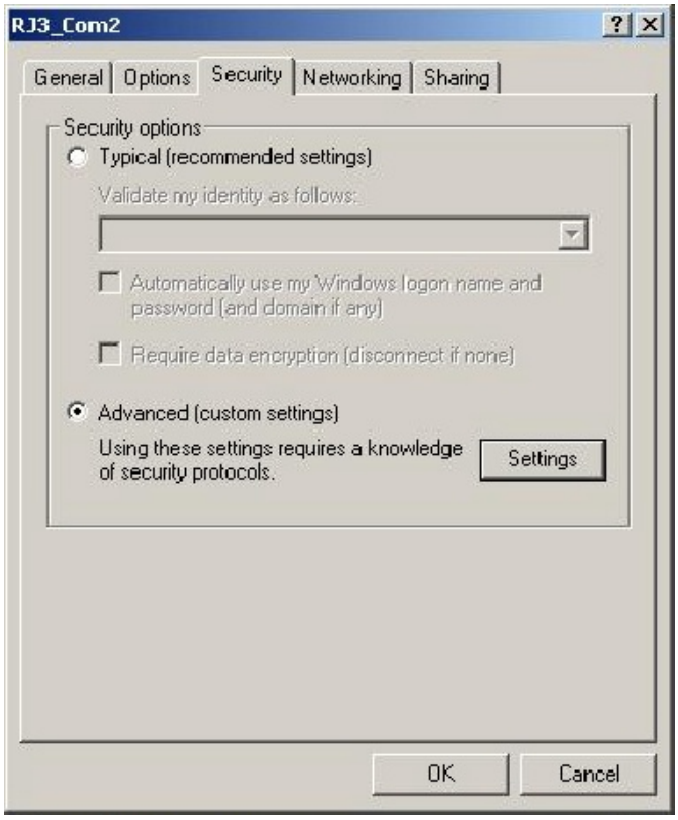



Fig. 8.3.2(h) Security

**10.** Select the **Security** tab and choose **Advanced (Custom Setting)** . Select the Settings button. You will see a screen similar to the following.


Fig. 8.3.2(i) Advanced (Custom Setting)

**11.** Select **Option Encryption** and select the box for Unencrypted password (PAP) is checked. Uncheck all other boxes. Select OK.

**12.** Select the **Networking** tab. You will see a screen similar to the following.

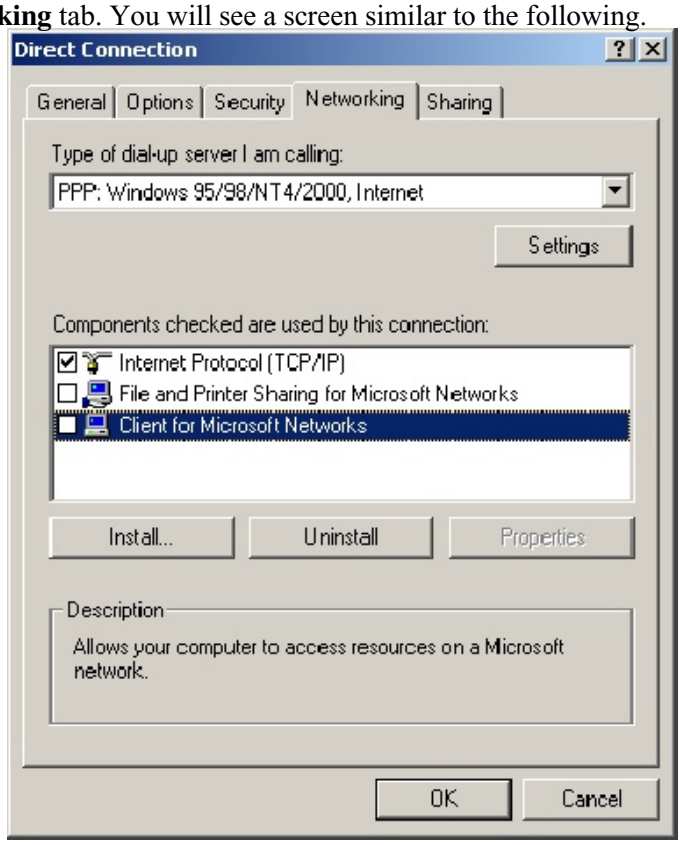



Fig. 8.3.2(j) Networking

**13.** Uncheck **Client for Microsoft® Networks** and **File and Printer Sharing.** You must make sure that Internet Protocol (TCP/IP) is selected.

**14.** Select **Settings** . You will see a screen similar to the following.

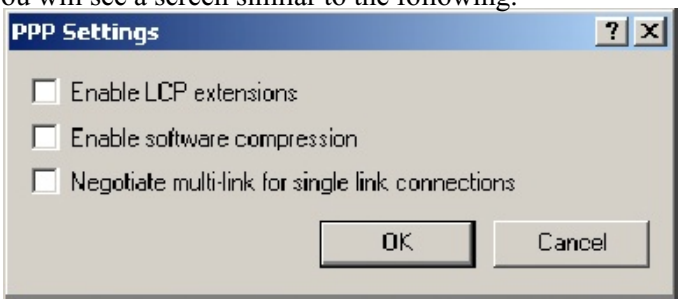



Fig. 8.3.2(k) TCP/IP Settings

Make sure all the boxes are unchecked. Select **OK**.

**15.** Select **Internet Protocol (TCP/IP)** and choose Properties. You will see a screen similar to the following.

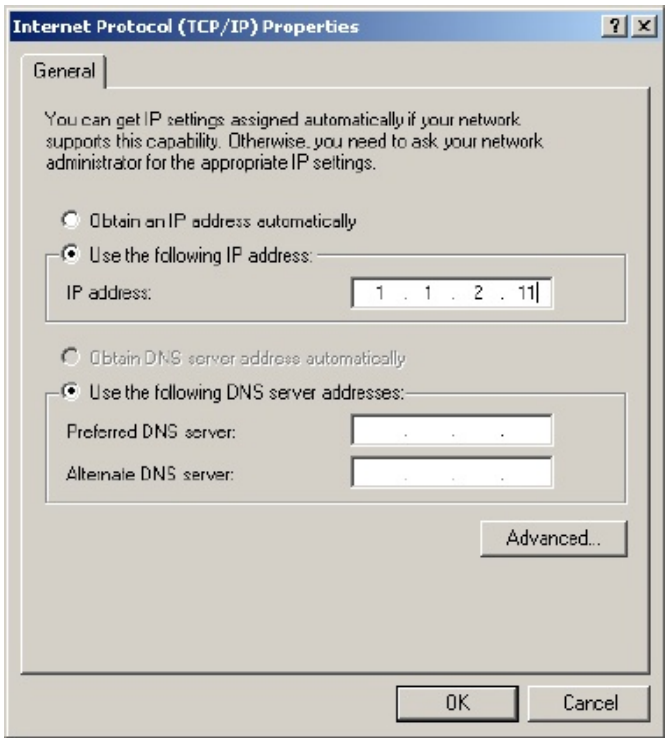



Fig. 8.3.2(l) TCP/IP Settings

**16.** Type the IP address corresponding to the serial port you are using. Leave the entries for the DNS server address blank.

**17.** Select the **Advanced** button. You must make sure the **Use IP header compression** box is checked.

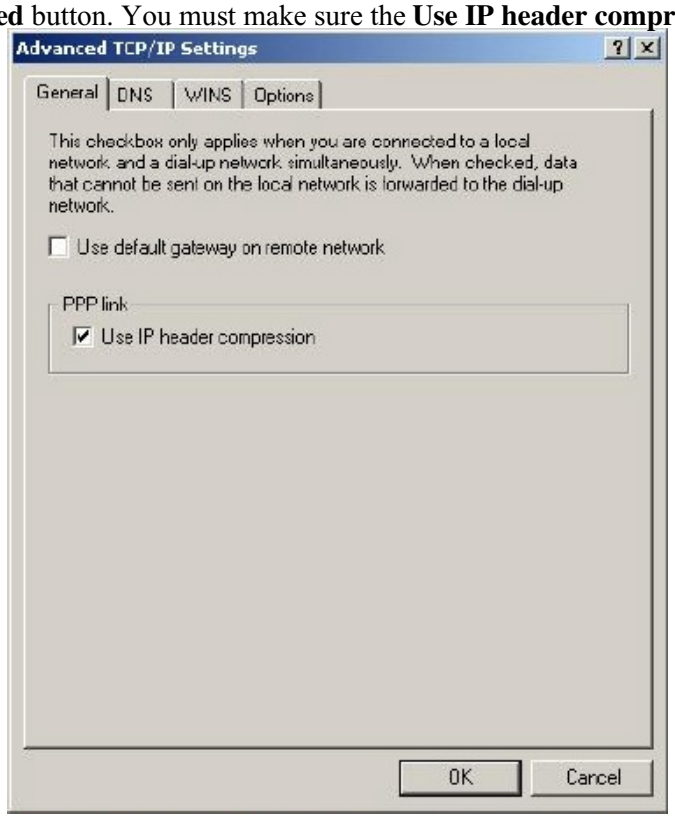



Fig. 8.3.2(m) Advanced Settings

**18.** Select the **DNS** tab. You must make sure that the boxes are checked/unchecked. See the following screen for an example.

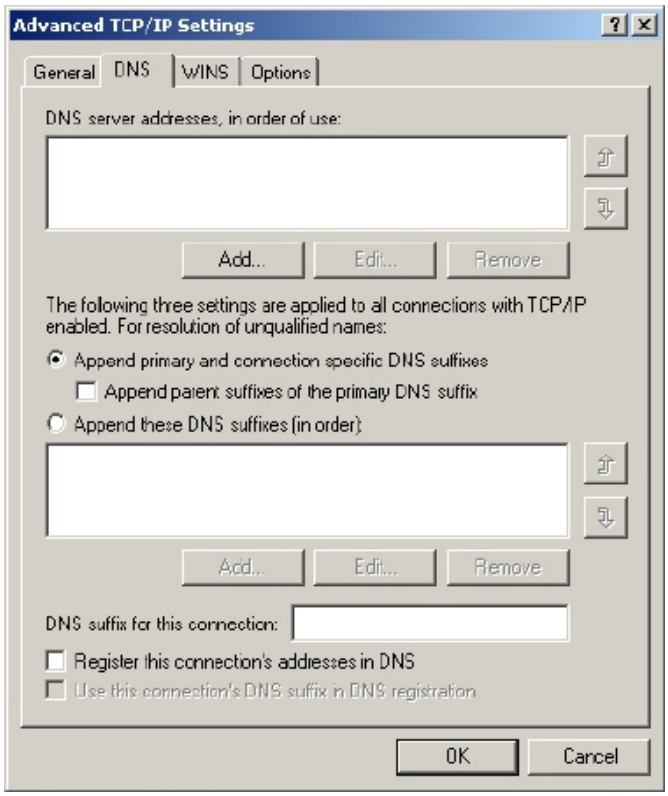



Fig. 8.3.2(n) DNS

**19.** Select the **WINS** tab. Uncheck the box **Enable LMHOSTS lookup.**

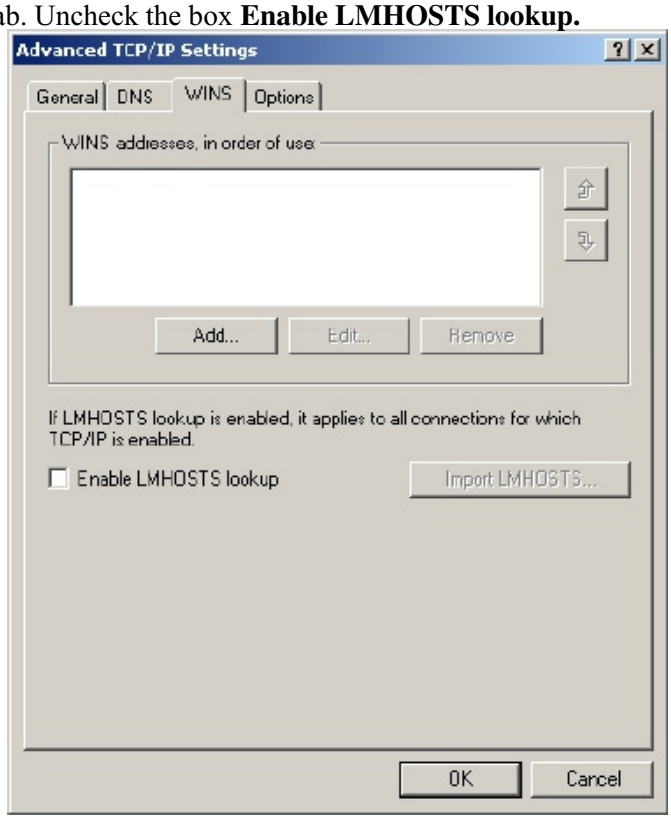



Fig. 8.3.2(o) WINS Enable LMHOSTS

**20.** Select the **Options** tab. You will see a screen similar to the following.

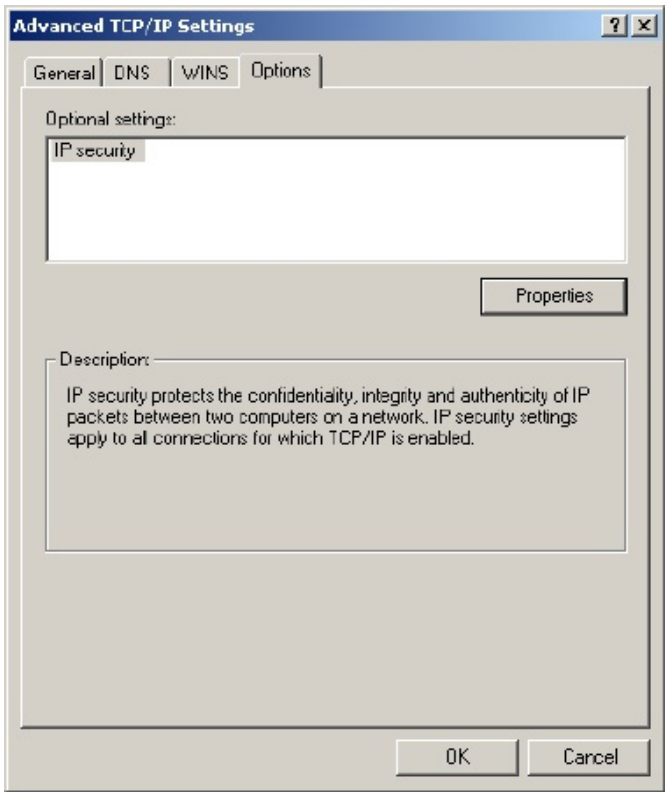



Fig. 8.3.2(p) Options

**21.** Choose IP Security and select the Properties button. You will see a screen similar to the following.

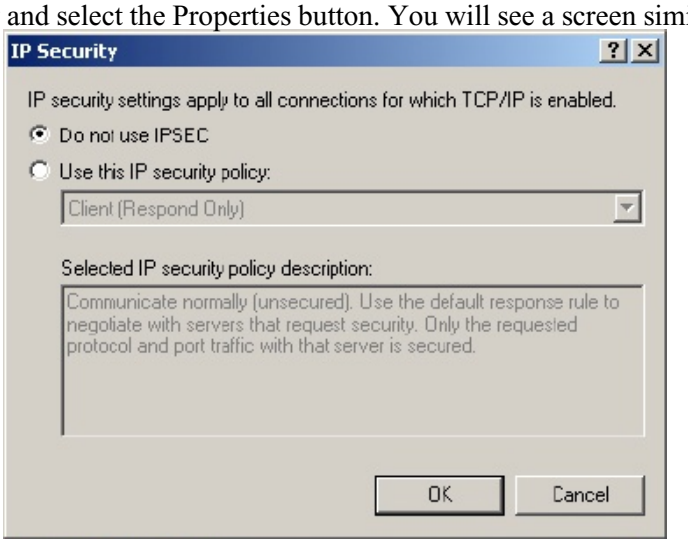



Fig. 8.3.2(q) IP Security

**22.** You must make sure the **Do not use IPSEC** button is selected. Select **OK.**
**23.** If a window pops up with the message **WINS entry is empty** , select OK to ignore the message.

## Procedure 8-5 Setting Up PPP on a PC with Windows XP

**1.** Select Control Panel, Network and Dial Up Connections and Create a New Connection. You will see a screen similar to the following.

Fig. 8.3.2(r) New Connection Wizard

**2.** Click Next. You will see a screen similar to the following.

Fig. 8.3.2(s) New Connection Wizard

**3.** Select **Set up an advanced connection**, and click Next. You will see a screen similar to the following.

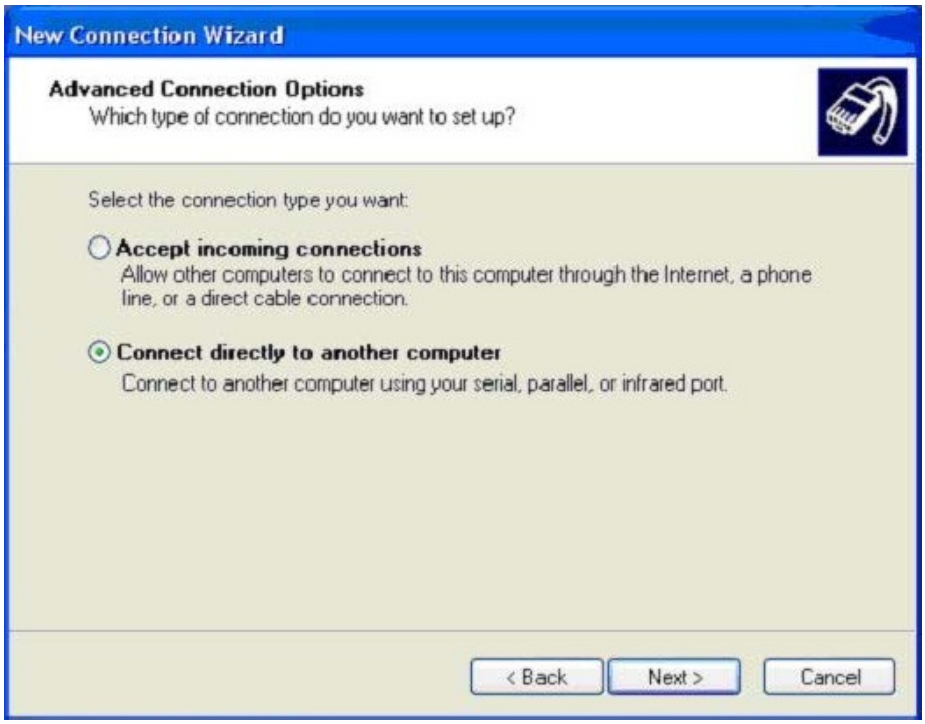



Fig. 8.3.2(t) Advanced Connection Options

**4.** Select **Connect directly to another computer**, and click Next. You will see a screen similar to the following.

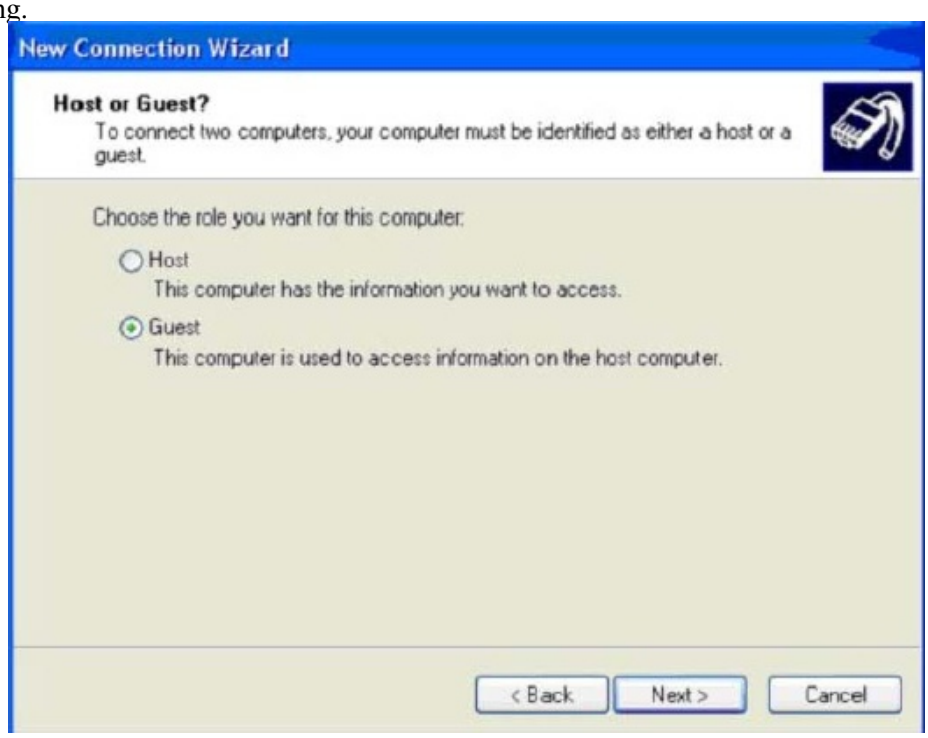



Fig. 8.3.2(u) Host or Guest

**5.** Select **Guest** and click Next. You will see a screen similar to the following.

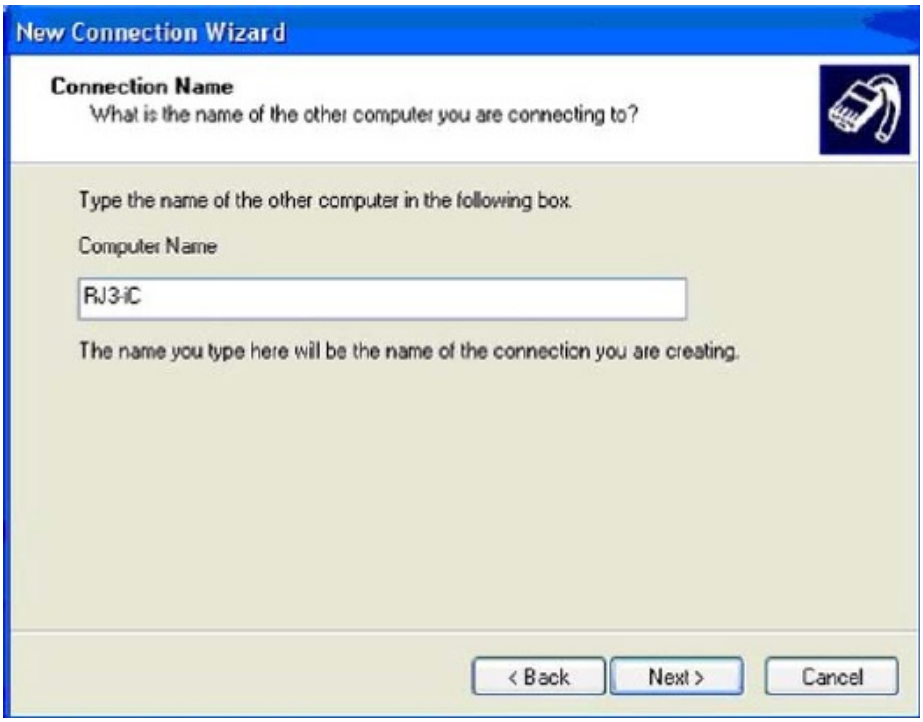



Fig. 8.3.2(v) Connection Name

**6.** Type in Computer Name. For this example, R-30*i*A is used. Click Next and you will see a screen similar to the following.

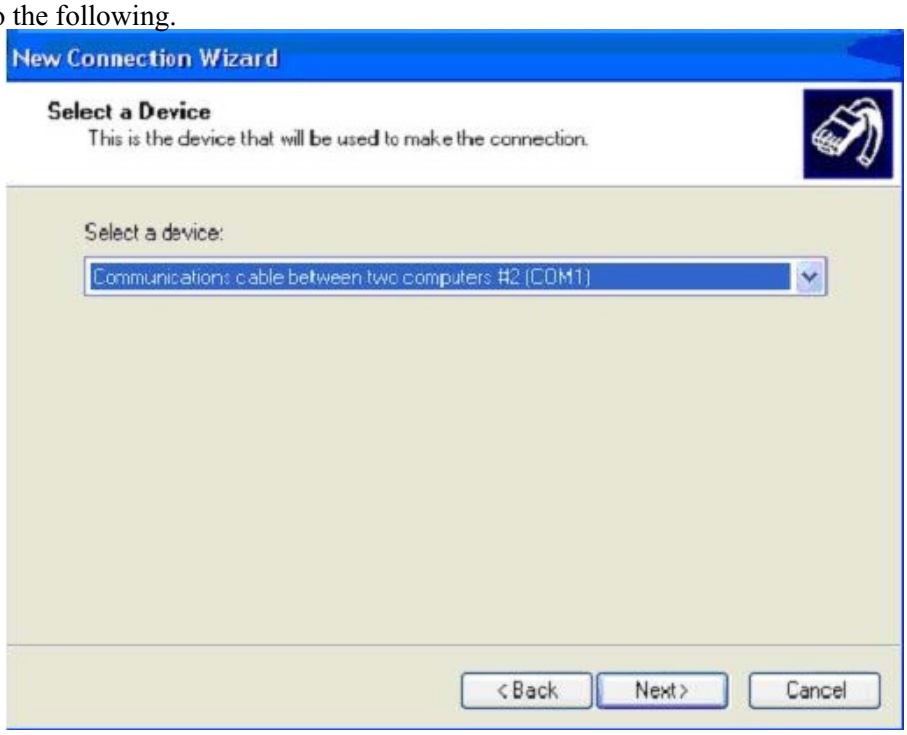



Fig. 8.3.2(w) Select a Device

**7.** Select Communications cable between two computers (COMx), where x is the COM port you will be using for your connection. Click Next to continue. You will see a screen similar to the following.


Fig. 8.3.2(x) Connection Availability

**8.** If you want all users who log on to your PC to use this connection, select **Anyone's use** , and click Next. If you want to be the only one that logs to your PC to use this connection, select **My use only**, and click Next. You will see a screen similar to the following.


Fig. 8.3.2(y) Completing the New Connection Wizard

**9.** Click Finish. You will see a screen similar to the following.

Fig. 8.3.2(af) Advanced TCP/IP Settings

**19.** Select the DNS tab. You must make sure that the boxes are checked/unchecked as shown in Figure 8.3.2(ag) .

Fig. 8.3.2(ag) Advanced TCP/IP Settings

**20.** Select the WINS tab. You will see a screen similar to the following.

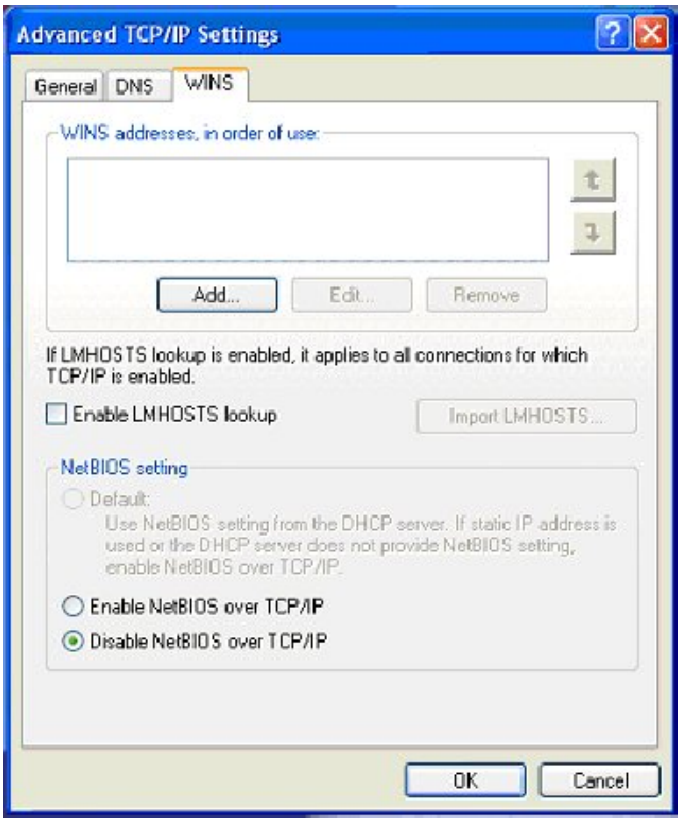



Fig. 8.3.2(ah) Advanced TCP/IP Settings

**21.** Uncheck the box Enable LMHOSTS lookup and check Disable NetBIOS over TCP/IP.

## Procedure 8-6 Setting Up a PPP/Modem on a PC with Windows XP

**1.** Select Control Panel, Network and Dial Up Connections and Create a New Connection. You will see a screen similar to the following.


Fig. 8.3.2(ai) New Connection Wizard

**2.** Click Next, and you will see a screen similar to the following.

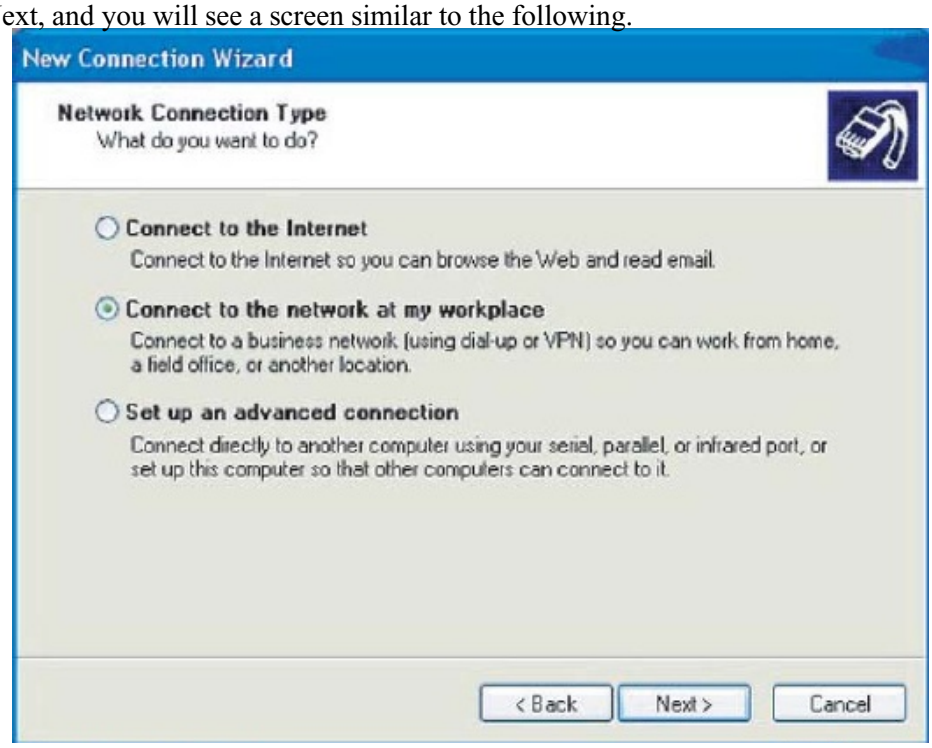



Fig. 8.3.2(aj) Network Connection

**3.** Click **Connect to the network at my workplace**, and click Next. You will see a screen similar to the following.

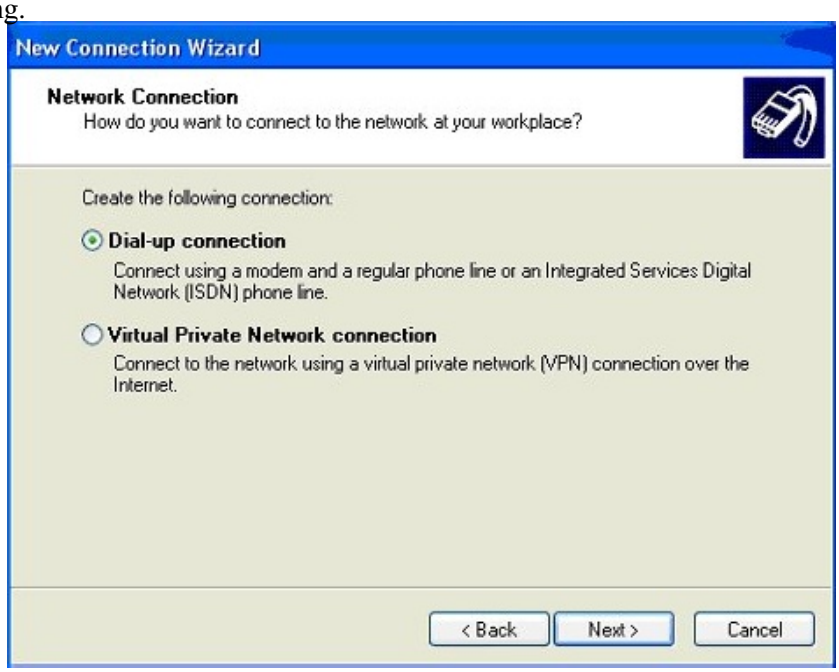



Fig. 8.3.2(ak) Dial Up Connection

**4.** Click **Dial-up connection**, and click Next. You will see a screen similar to the following.

Fig. 8.3.2(al) Connection Name

**5.** Click Next. You will see a screen similar to the following.

Fig. 8.3.2(am) Phone Number to Dial

**6.** Type the phone number of modem where the controller is connected. Click Next. You will see a screen similar to the following.


Fig. 8.3.2(an) Connection Availability

**7.** If you want all users who log on to your PC to use this connection, select Anyone's use and click NEXT. You will see a screen similar to the following.


Fig. 8.3.2(ao) Completing the New Connection Wizard

**8.** Click Finish and you are ready to connect to the controller via a PPP/Modem.

# 9 DYNAMIC HOST CONFIGURATION PROTOCOL

## 9.1 OVERVIEW

### 9.1.1 Introduction to DHCP

DHCP ( **Dynamic Host Configuration Protocol** ) is a service which automates robot configuration on an existing Ethernet network. DHCP is used commonly on PCs to configure them on the network.
The service requires a DHCP server to be present on the network. It returns the various network parameters to the requesting host (DHCP client) which configures it on the network automatically. The network parameters returned by the server typically include at least the **IP address** to be used by the robot, the **subnet mask** of the network, and the **router** or **gateway** used for that network. The server can be configured to return more information such as DNS servers and so forth which can be used to set up the robot as a DNS client.
The DHCP server typically **leases** the IP address to the DHCP client. This means that the robot can use the IP address for a certain period of time called the **lease time** . The lease time period is returned by the DHCP server along with the IP address. The IP address given out by the server is valid for the duration of the lease time. This concept of allocating leases to an IP address is called **Dynamic Allocation** of the IP address. The server typically also returns a **renewal time** for dynamically allocated IP addresses. The renewal time is less than the lease expiration time. When the renewal time expires, the DHCP client typically renews the lease on the IP address (or gets back a new IP address) from the DHCP server.

### 9.1.2 Features of the Robot DHCP Client

The Robot DHCP Client:

- Is used at Controlled and Cold start for network configuration purposes
- RFC2131 and RFC2132 (internet specification) compliant
- Supports leasing of IP address
- Checks IP address first to see if it is in use before using it
- Can act like a PC based DHCP client for seamless integration of the robots into the existing network
- Is easy to set up

## 9.2 SETTING UP DHCP ON THE ROBOT

### 9.2.1 DHCP Setup

The DHCP setup screens are located on the Setup-Hostcomm-TCP/IP screens. The DHCP button on this screen launches the DHCP SETUP screen.

> NOTE
> The DHCP button shows up only when DHCP is installed on the robot.

Table 9.2.1(a) DHCP SETUP Screen Items

| ITEM | DESCRIPTION |
|---|---|
| DHCP enable<br>Values: TRUE or FALSE | This item indicates whether DHCP is enabled. |
| DHCP status | This item indicates the status of the current DHCP operation. |

Table 9.2.1(b) Advanced DHCP SETUP Screen Items

| ITEM | DESCRIPTION |
|---|---|
| Client ID | This item is an optional parameter that the client can send to the server to request specific configuration information. This item can use the Ethernet address of the robot, or any string identifier. If you are typing an Ethernet address, the format must be six bytes separated by colons (for example, 00:E0:E4:F7:94:AC). |
| Set hostname in request | This item allows the robot to function like Windows-based DHCP clients (PCs) in sending out its hostname in the form of a DHCP request. To use the set hostname in request field, you have to make sure that the robot hostname field is set from the TCP/IP screens first. Setting this field sets $DHCP_CTRL.$SETHOST. |
| Retry rate on failure | This item controls the rate (in minutes) at which retries occur if the robot does not get a response back from the server. The DHCP internally tries for a full minute to contact the server before giving up and reporting an error. This retry rate field determines when the next attempt to contact the server must be done. Setting this field sets the system variable $DHCP_CTRL.$RETRATE. |
| Use last valid IP on failure | This item is used in a case where the robot has a previously assigned IP address and the lease is still valid on the IP address. When power is cycled on the robot, the robot on booting contacts the DHCP server to confirm the lease (this is standard DHCP behavior). If the DHCP server does not respond for some reason (network/ server is down or damage to cables), then this field determines if the robot will continue to use the IP address. If set to TRUE, then under these conditions, the robot will continue to use the IP address; if set to FALSE, the robot will not use the IP address. Under no circumstances will the robot use the IP address beyond the lease expiration time, regardless of this setting. Setting this field sets the system variable $DHCP_CTRL.$USEIP. |

## Procedure 9-1 Setting up DHCP on the Robot

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE] and select Host Comm and then TCP/IP. You will see a screen similar to the following.

```
SETUP Host Comm
TCP/IP
    Robot Name:                 ****
    Port#1 IP addr:     **************
    Subnet Mask:         255.255.255.0
    Board address:    08:00:19:02:68:22
    Router IP addr:     **************
Host Name (LOCAL)      Internet Address

  1 ***********       *****************
  2 ***********       *****************
  3 ***********       *****************
```

4. If you want to enable DHCP on Port #1, follow Step 4.a . If you want to enable DHCP on Port#2, follow Step 4.b .
   a. Press F2 (DHCP). You will see a screen similar to the following.

```
DHCP Setup
Port#1
DHCP enable:   FALSE
DHCP status:   *********
```

   b. Press F3(PORT) to display Port#2 Host Comm screen. Press F2(DHCP). You will see a screen similar to the following.

```
DHCP Setup
Port#2
DHCP enable:   FALSE
DHCP status:   **********
```

5. Press F4, TRUE to enable DHCP. The DHCP status shows the status of the DHCP operation.

```
DHCP Setup
Port#1
DHCP enable: TRUE
DHCP status: Success
```

NOTE

With the DHCP server properly configured, most users should be able to use the DHCP service on the robot by pressing the Enable button from the DHCP screens. If you need to reconfigure DHCP Setup (either on the robot side or on the server side) while DHCP is enabled, then you would need to disable DHCP first from the DHCP SETUP screens, make the necessary configuration changes, and re-enable DHCP again from the DHCP SETUP screens.

NOTE

It is recommended that when the system clock on the robot is changed, DHCP is disabled from the DHCP SETUP screen, and re-enabled again.

6. Press F3, ADV to go to the Advanced DHCP SETUP screen. You will see a screen similar to the following.

```
Advanced DHCP Setup
Port#1
Client ID:    **********************
Set hostname in request:      FALSE
Retry rate on failure:       10 min
Use last valid IP on failure: FALSE
```

NOTE

Most users do not need to go to the Advanced DHCP SETUP screen and change the defaults. The screen provides flexibility so that the robot can support different kinds of DHCP server configurations. In some cases it might be necessary to set advanced DHCP options from the Advanced DHCP Setup screen. You must consult with your network administrator if you have any questions.

## 9.2.2 Advanced DHCP Setup

The **client ID** is an optional parameter that the client can send to the server to request specific configuration information. The server needs to be configured to recognize the client ID that the user sets in this field. You can set this field to be the Ethernet address of the robot or to any string identifier. If you are typing an Ethernet address in this field, then the format of the Ethernet address must be 6 bytes separated by colons. An example might include 00:E0:E4:F7:94:AC

NOTE

The Ethernet address of your robot can be viewed from the TCP/IP screens.

Setting the client ID field sets the system variable $DHCP_CLNTID[Port#], where Port# is either 1 or 2.

The **set hostname in request** field allows the robot to function like Windows based DHCP clients (PC's) in sending out its hostname in the form of a DHCP request. Some servers are written explicitly to service Microsoft® clients only so this field allows the robot to function like PCs. To use the set hostname in request field, you have to make sure that the robot hostname field is set from the TCP/IP screens first.

Setting this field sets $DHCP_CTRL[Port#].$SETHOST, where Port# is either 1 or 2.

NOTE
It is up to the DHCP server to update DNS tables when an IP address is given out. If the server does not do this, then it will not be possible to access the robot using the robot hostname, and other hosts will need to use the IP address returned by the server to communicate with the robot. This feature where the DNS server gets informed about the new IP address (via the DHCP mechanism in this case) is called dynamic DNS . For security reasons, it is usually up to the server to do dynamic DNS and inform the DNS server of the IP address changes. The robot DHCP client does not support the dynamic DNS feature.

The **retry rate on failure** field controls the rate (in minutes) at which retries occur if the robot does not get a response back from the server. The DHCP internally tries for a full minute to contact the server before giving up and reporting an error. This retry rate field determines when the next attempt to contact the server must be done. Setting this field sets the system variable $DHCP_CTRL[Port#].$RETRATE, where Port# is either 1 or 2. Set this field to zero to disable retries.
The **last valid IP address on failure** field is used in a case where the robot has a previously assigned IP address, and the lease is still valid on the IP address. When power is cycled on the robot, the robot on booting contacts the DHCP server to confirm the lease (this is standard DHCP behavior). If the DHCP server does not respond for some reason (network/ server is down or damage to cables), then this field determines if the robot will continue to use the IP address or not. If set to TRUE, then under these conditions, the robot will continue to use the IP address, but if set to FALSE, the robot will not use the IP address. Under no circumstances will the robot use the IP address beyond the lease expiration time, regardless of this setting. Setting this field sets the system variable $DHCP_CTRL[Port#].$USEIP, where Port# is either 1 or 2.

NOTE
The Hostcomm TCP/IP screen looks different upon a successful DHCP operation.

```
SETUP HostComm
TCP/IP - DHCP enabled
    Robot name:                   ROBOT

    Port#1 IP addr:       172.22.200.165
    Subnet Mask:           255.255.240.0
    Board address:     08:00:19:02:68:22
    Router IP addr:         172.22.192.1
Host Name (LOCAL)       Internet Address
  1 ***********      ********************
  2 ***********      ********************
  3 ***********      ********************
```

The robot's network information, as returned by the server is reflected in the above screens, but also the first five lines are marked read-only and the user cannot edit these parameters when DHCP is enabled (regardless of whether the DHCP operation succeeded or not). If you must manually set these parameters, DHCP must be disabled.

# 9.3 DHCP SYSTEM VARIABLES

**$DHCP_CTRL_T**[Port#], where Port# is either 1 or 2, structure includes the following fields. This system variable structure is saved in **syshost.sv** and can be copied to a media and moved between robots.

**$ENABLE: BOOLEAN: default FALSE**

This variable enables the robot to start functioning as a DHCP client. The robot tries to configure its Ethernet interface right away. On subsequent powerups, if this variable is set, the robot will try to contact the DHCP server and will use the configuration information returned by the server. If the variable is set, the robot will not use any parameters manually configured by the user from the teach pendant or via system variables on this power cycle or on subsequent power cycles. Enabling DHCP from the DHCP screen causes this field to be set to TRUE.

**Powerup:** The powerup takes effect immediately.

**UIF Location:** DHCP SETUP screen.

**$IPUSE: BOOLEAN default TRUE**

If DHCP is enabled and the robot has a valid lease on an IP address and power is cycled on the robot, then, on powerup, the robot tries to contact the DHCP server to validate its lease. If the server does not respond, the robot might not continue to use the IP address it obtained before. If this variable is set to TRUE, the robot will continue to use the IP address till the lease expires. If this variable is set to FALSE, the robot will shut down the Ethernet interface right away. Under no circumstances will the robot continue to use an IP address after its lease has expired.

**PowerUp:** Cycle power to take effect.

**UIF Location:** DHCP advanced SETUP screen.

**$RETRATE: INTEGER: default 10**

If DHCP is enabled, and the DHCP operation fails, this variable controls the rate (in minutes) at which attempts are made by the robot to contact the DHCP server. DHCP internally tries for a full minute to contact the server before giving up and reporting an error. This retry rate field determines when the next attempt to contact the server must be done.

**PowerUp:** The powerup takes effect immediately.

**UIF Location:** DHCP Advanced SETUP screen

**$SETHOST: BOOLEAN: default FALSE**

This variable sets the hostname field in the DHCP request sent to the server. Some servers require the hostname to be supplied in the hostname field in the request (especially servers serving Microsoft® clients). In this case, you may need to set this field to TRUE. When this field is set to TRUE, the robot hostname ($HOSTNAME) is supplied as the hostname in the DHCP request.

**Powerup:** The powerup takes effect immediately.

**UIF Location:** DHCP Advanced SETUP screen.

**$DHCP_INT_T**[Port#], where Port# is either 1 or 2, structure includes the following fields.

This structure is used internally by DHCP. Users cannot modify this system variable structure (all fields are Read-Only). There is no UIF that displays this structure. This system variable is not saved (not restored) in any .sv files.

**$LEASESTRTIME, ULONG: default 0**

This variable gives the time of start of the lease.

**$LEASESTART: STRING**

Time of start of lease in a readable format.

**$LEASEENDTIME: ULONG: default 0**

This variable gives the time when the lease will expire.

**$LEASEEND: STRING**

This variable is the lease expiration time in a readable format.

**$IPADD: STRING**

This variable indicates that the server returned the IP address.

**$ROUTERIP: STRING**

This variable indicates that the server returned router IP address.

**$SNMASK: STRING**

This variable indicates that the server returned subnet mask.

**$STATUS: STRING**
This variable indicates the status of the DHCP operation.
**$DHCP_CLNTID: STRING: R/W**
Client identifier passed by the robot to the server. This might not have to be supplied, depending on how the DHCP server is configured. You must see your network administrator for more details. Typical use of the client identifier is either to supply an Ethernet address or to supply a string to the server. To use the Ethernet address, the 6 bytes must be separated by colons. Eg: 00:E0:E4:F7:94:DC
**PowerUp:** This variable takes effect immediately.
**UIF Location:** DHCP SETUP screen.

## 9.4 DHCP TROUBLESHOOTING

Some of the DHCP errors that you might receive include the following:

- **The DHCP operation failed with HOST-224 DHCP: No response from the server**
  You must make sure that the robot is connected to the network with a working Ethernet cable. You must contact your network administrator and make sure that the DHCP server is configured and running. The DHCP server must typically be located on the same network as the robot (otherwise, there must be a router on the network that functions as a DHCP relay agent and forwards requests and responses from one network to another). This problem could also happen if the network is having problems (such as heavy traffic). You can check this by looking at the Ethernet diagnostics by pressing DIAG key under the Host-Comm TCP/IP screen.
- **The DHCP operation failed with HOST-225: DHCP duplicate IP <x.x.x>**
  If this error occurs it means that the DHCP server served up an IP address that is already being used by another host on the network. You must inform your network administrator about this problem when it occurs.
- **Ethernet on robot stops working with HOST-226 and HOST-227 errors (lease time expired/shutting down Ethernet)**
  The robot could not renew the DHCP lease and the lease expired. This should not happen under normal circumstances. The robot might not be connected to the network or the network is having problems or the DHCP server might not be running any more.

# 10 SOCKET MESSAGING

## 10.1 OVERVIEW

The User Socket Messaging Option gives you the benefit of using TCP/IP socket messaging from KAREL.
Socket Messaging enables data exchange between networked robots and a remote PC with LINUX, or a UNIX workstation. A typical application of Socket Messaging might be a robot running a KAREL program that sends process information to a monitoring program on the remote PC. The combination of PC-Interface option on the robot and PC-Developers Kit on the PC is recommended for data exchange between the robot and a Windows-based PC.
Socket Messaging uses the TCP/IP protocol to transfer raw data, or data that is in its original, unformatted form across the network. Commands and methods that Socket Messaging uses to transfer data are part of the TCP/IP protocol. Since Socket Messaging supports client and server tags, applications requiring timeouts, heartbeats, or data formatting commands can provide these additional semantics at both the client and server (application) sides of the socket messaging connection.

## 10.2 SYSTEM REQUIREMENTS

### 10.2.1 Overview

This section contains information about the compatibility of socket messaging with some typical network software, transmission protocols, and interface hardware.

### 10.2.2 Software Requirements

Socket Messaging is compatible with all other Internet Options including DNS, FTP, Web Server, and Telnet.

> NOTE
> Client and Server tags are shared between Socket Messaging and FTP. A tag can be set for either FTP operation or for SM (Socket Messaging) operation.

### 10.2.3 Hardware Requirements

Socket Messaging is compatible with all network hardware configurations that use the TCP/IP network protocol. Some of these network hardware configurations include Ethernet, serial PPP connections and PPP modem connections.

## 10.3 CONFIGURING THE SOCKET MESSAGING OPTION

### 10.3.1 Overview

In order to use Socket Messaging, you need to configure the following network hardware and software parameters:

- On the server,
  - The port you want to use for socket messaging
- On the client,

- The IP address or name of your server
- The port on the server that you want to use for socket messaging.

Use Procedure 10-1 to set up a Socket Messaging Server Tag. Use Procedure 10-2 to set up a Socket Messaging Client Tag.

NOTE
The server port at which the server listens on should match the port the client tries to connect on.

## 10.3.2 Setting up a Server Tag

You need configure the server tags you want to use for socket messaging. Use Procedure 10-1 to set up your server tags.

NOTE
If the server tags you want to use are being used by a network protocol other than TCP/IP, you need to undefine the tags before they can be used for socket messaging. After making sure the tag you want to use is not critical to another component of your network, you must undefine the tag.

### Procedure 10-1 Setting up a Server Tag

#### Conditions

- The tag you want to set up is not configured to be used by another device on your network.

#### Steps

1. Cold start the controller.
   a. On the teach pendant, press and hold the SHIFT and RESET keys. Or, on the operator panel, press and hold RESET.
   b. While still pressing SHIFT and RESET on the teach pendant (or RESET on the operator panel), turn on the power disconnect circuit breaker.
   c. Release all of the keys.
2. On the teach pendant, press MENUS.
3. Select SETUP.
4. Press F1, [TYPE].
5. Select Host Comm.
6. Press F4, [SHOW].
7. Choose Servers.
8. Move the cursor to the tag you want set up for Socket Messaging, and press F3, DETAIL. You will see screen similar to the following.

```
SETUP Tags
 Tag S3:

 1 Comment:          ****************
 2 Protocol name:           ********
 3 Port name:                 *****
 4 Mode:     *************************
Current
   State:                  UNDEFINED
 5 Remote:                  ********
 6 Path:  ****************************
Startup
 7 State:
 8 Remote:                ***********
 9 Path:  ****************************
Options
10 Error Reporting:              OFF
11 Inactivity Timeout:         15 min
```

9. Move the cursor to Protocol name, and press F4, [CHOICE].
10. Select SM.
11. Move the cursor to Startup State, and press F4, [CHOICE].
12. Select START.
13. Press F2, [ACTION].
14. Select DEFINE.
15. Press F2, [ACTION].
16. Select START.
17. Set the system variable:
    a. Press MENUS.
    b. Select NEXT.
    c. Select SYSTEM, and press F1, [TYPE].
    d. Select Variables.
    e. Move the cursor to $HOSTS_CFG, and Press ENTER.
    f. Move the cursor to the structure corresponding to the tag selected in Step 8 . For example, if you are setting up tag S3, move the cursor structure element [3], as shown in the following screen.

```
SYSTEM Variables
$HOSTS_CFG
   1   [1]          HOST_CFG_T
   2   [2]          HOST_CFG_T
   3   [3]          HOST_CFG_T
   4   [4]          HOST_CFG_T
   5   [5]          HOST_CFG_T
   6   [6]          HOST_CFG_T
   7   [7]          HOST_CFG_T
   8   [8]          HOST_CFG_T
```

    g. Press ENTER. You will see a screen similar to the following.

```
SYSTEM Variables
$HOSTS_CFG[3]
  1 $COMMENT       *uninit*
  2 $PROTOCOL      'SM'
  3 $PORT          *uninit*
  4 $OPER          3
  5 $STATE         3
  6 $MODE          *uninit*
  7 $REMOTE        *uninit*
  8 $REPERRS       FALSE
  9 $TIMEOUT       15
 10 $PATH          *uninit*
 11 $STRT_PATH     *uninit*
 12 $STRT_REMOTE   *uninit*
 13 $USERNAME      *uninit*
 14 $PWRD_TIMOUT   0
 15 $SERVER_PORT   0
```

h. Move the cursor to $SERVER_PORT. Type in the name of the TCP/IP port you want to use for socket messaging. The server tag is now ready to use from a KAREL program.

## 10.3.3 Setting up a Client Tag

You need configure the client tags you want to use for socket messaging. Use Procedure 10-2 to set up your server tags. You can also use Procedure 10-2 to undefine tags.

> NOTE
> If the client tags you want to use are being used by a network protocol other than TCP/IP, you need to undefine the tags before they can be used for socket messaging.

### Procedure 10-2 Setting up a ClientTag

#### Conditions

- The tag you want to set up is not configured to be used by another device on your network.

#### Steps

1. Cold start the controller.
   a. On the teach pendant, press and hold the SHIFT and RESET keys. Or, on the operator panel, press and hold RESET.
   b. While still pressing SHIFT and RESET on the teach pendant (or RESET on the operator panel), turn on the power disconnect circuit breaker.
   c. Release all of the keys.
2. On the teach pendant, press MENUS.
3. Select SETUP.
4. Press F1, [TYPE].
5. Select Host Comm.
6. Press F4, [SHOW].
7. Choose Clients.
8. Move the cursor to the tag you want set up for Socket Messaging, and press F3, DETAIL. You will see screen similar to the following.

```
SETUP Tags
 Tag C3:

 1 Comment:          ****************
 2 Protocol name:            ********
 3 Port name:                  *****
 4 Mode:     ************************
Current
   State:                  UNDEFINED
 5 Remote:                  ********
 6 Path:  ****************************
Startup
 7 State:
 8 Remote:                ***********
 9 Path:  ****************************
Options
10 Error Reporting:               OFF
11 Inactivity Timeout:          15 min
```

9. Move the cursor to Protocol name, and press F4, [CHOICE].
10. Select SM.
11. Move the cursor to Startup State, press F4, [CHOICE], and choose Define..
12. Move the cursor to Remote, and press ENTER.
13. Type in the of the remote host server you want to use for socket messaging.
14. Press F2, [ACTION], and select DEFINE.

If you are not using DNS, you must add the remote host and its IP address into the host entry table.

15. Set the system variable:
    a. Press MENUS.
    b. Select NEXT.
    c. Select SYSTEM, and press F1, [TYPE].
    d. Select Variables.
    e. Move the cursor to $HOSTC_CFG, and press ENTER.
    f. Move the cursor to the structure corresponding to the tag selected in Step 8 . For example, if you are setting up tag S3, move the cursor structure element [3], as shown in the following screen.

```
SYSTEM Variables
$HOSTC_CFG
  1   [1]          HOST_CFG_T

  2   [2]          HOST_CFG_T
  3   [3]          HOST_CFG_T
  4   [4]          HOST_CFG_T
  5   [5]          HOST_CFG_T
  6   [6]          HOST_CFG_T
  7   [7]          HOST_CFG_T
  8   [8]          HOST_CFG_T
```

    g. Press ENTER. You will see a screen similar to the following.

```
SYSTEM Variables
$HOSTC_CFG[3]
  1 $COMMENT        *uninit*
  2 $PROTOCOL       'SM'
  3 $PORT           *uninit*
  4 $OPER           3
  5 $STATE          3
  6 $MODE           *uninit*
  7 $REMOTE         *uninit*
  8 $REPERRS        FALSE
  9 $TIMEOUT        15
 10 $PATH           *uninit*
 11 $STRT_PATH      *uninit*
 12 $STRT_REMOTE    *uninit*
 13 $USERNAME       *uninit*
 14 $PWRD_TIMOUT    0
 15 $SERVER_PORT    0
```

h. Move the cursor to $SERVER_PORT. Type in the name of the TCP/IP server port you want to use for socket messaging. The client tag is now ready to use from a KAREL program.

# 10.4 SOCKET MESSAGING AND KAREL

## 10.4.1 Overview

Socket messaging is an integrated component of KAREL. When you use socket messaging functions and utilities from a KAREL program, the syntax is similar to other file read and write operations, except that you need to establish a network connection when you use socket messaging functions and utilities.
The following KAREL socket messaging functions and utilities enable the server to establish a connection with a remote host on your network. There are several KAREL program samples in this section that provide examples of how these functions and utilities can be used with KAREL file read and write functions and utilities to write a complete Socket Messaging KAREL client or a server program or application. The Environment flbt statement is required to use any of the listed builtins (%ENVIRONMENT flbt).

## 10.4.2 MSG_CONN( string, integer )

MSG_CONN needs to be called before any tag can be used for socket messaging.
The first parameter of this command contains the tag name ("S1:" for example) and the second parameter is an integer that will contain the status of the operation. If you are using this command to connect to a server tag, this command will return a status value only after a remote client device has established a connection with this server tag.
If you are using this command to connect to a client tag, this command will return a status value only if the remote server is attempting to accept the connection. If the connection was successful, the command will return a value indicating a successful connection was made. If the connection was not successful, the command will return a value indicating that a connection error has occurred.
During a socket messaging session, you must use MSG_DISCO to close the socket connection with a client or server tag before any subsequent attempts to connect to the same client or server tag can be made using MSG_CONN.

## 10.4.3 MSG_DISCO( string, integer )

MSG_DISCO is used to close socket messaging connections. If a connection is lost, perhaps because a READ or WRITE error occurred when the remote server terminated a socket messaging connection, you will need to use MSG_DISCO to close the connection to the remote server. In this case, MSG_DISCO

# 10.6 PROGRAMMING EXAMPLES

## 10.6.1 Overview

This section contains programming examples for a KAREL socket messaging client, and a KAREL socket messaging server. There is also a UNIX-based ANSI C example for a loopback client application, which assumes that you have access to a UNIX-compatible ANSI C compiler, and a basic knowledge of programming in the ANSI C language.

NOTE
The KAREL examples assume the appropriate tags (C2 for client and S3 for Server) have been setup for socket messaging using Procedure 10-1 and Procedure 10-2 .

## 10.6.2 A KAREL Client Application

Example 10.6.2(a) provides code for a basic KAREL client application that can be used to establish a socket messaging connection to a remote host, which could be the KAREL server socket messaging application shown in Example 10.6.3(a) .

Example 10.6.2 (a) A KAREL Client Application

```
--    This material is the joint property of Fanuc Robotics Corporation and
--    FANUC LTD Japan, and must be returned to either Fanuc Robotics
--    Corporation or FANUC LTD Japan immediately upon request. This material
--    and the information illustrated or contained herein may not be
--    reproduced, copied, used, or transmitted in whole or in part in any way
--    without the prior written consent of both Fanuc Robotics and FANUC
--
--            All Rights Reserved
--            Copyright (C) 2000
--            Fanuc Robotics Corporation
--            FANUC LTD Japan
--
--            Karel is a registered trademark of
--            Fanuc Robotics Corporation
--    +
--    Program: loopcl.kl - Program for TCP Messaging
--
--    Description:
--
--    This program serves as an example on how to use TCP messaging
and write
--    a client Karel program.
--
--    Authors: Fanuc Robotics Corporation
--            3900 West Hamlin
--            Rochester Hills, MI 48309
--
--    Modification history:
--
-------------------------------------------------------------------------------
--
PROGRAM loopcl
%RWACCESS
%STACKSIZE = 4000
%NOLOCKGROUP
%NOPAUSE=ERROR+COMMAND+TPENABLE
%ENVIRONMENT uif
%ENVIRONMENT sysdef
```

```
%ENVIRONMENT memo
%ENVIRONMENT kclop
%ENVIRONMENT bynam
%ENVIRONMENT fdev
%ENVIRONMENT flbt

%INCLUDE klevkeys
%INCLUDE klevkmsk
------------------------------------------------------------------------------
--
VAR
        file_var :  FILE
        tmp_int  : INTEGER
         mp_str  : string[128]
         status  : integer
         entry   : integer
         loop1   : BOOLEAN
-------------------------------------------------------
```

Example 10.6.2 (b)

```
BEGIN
       SET_FILE_ATR(file_var, ATR_IA)
       SET_VAR(entry, '*SYSTEM*','$HOSTC_CFG[2].$SERVER_PORT',59002,status)
       -- Connect the tag
       WRITE('Connecting..',cr)
       MSG_CONNECT('C2:',status)
       WRITE(' Connect Status = ',status,cr)
       loop1 = TRUE
       IF status = 0 THEN
             WHILE loop1 = TRUE DO
                    WRITE('Opening File..',cr)

                    OPEN FILE file_var(rw,'C2:')
                    status = io_status(file_var)
                    IF status = 0 THEN
                           FOR tmp_int = 1 TO 100 DO
                                  tmp_str = '0123456789012345'
                                  WRITE file_var(tmp_str::10)
                                  WRITE('Wrote 126 Bytes',cr)
                                  IF status <> 0 THEN
                                         WRITE('Loop Test Fails',cr)
                                         loop1 = FALSE
                                         tmp_int = 100
                                  ELSE
                                         WRITE('Read 126 Bytes',cr)
                                         READ file_var(tmp_str::10)

                           status = io_status(file_var)
                                         WRITE('Read Status ',status,cr)
                                  ENDIF
                           ENDFOR
                           WRITE('Closed File',cr)
                           CLOSE FILE file_var
                           ELSE
                           WRITE('Error Opening File',cr)
                           loop1 = FALSE
                    ENDIF
             ENDWHILE
             WRITE('Disconnecting..',cr)
             MSG_DISCO('C2:',status)
             WRITE('Done.',cr)
       ENDIF
END loopcl
```

## 10.6.3 A KAREL Server Application

Example 10.6.3(a) provide code for a basic KAREL server application that can be used to host a socket messaging connection made by a remote client, which could be the KAREL client socket messaging application shown in Example 10.6.2(a) .

Example 10.6.3 (a) KAREL Server Application

```
--    This material is the joint property of Fanuc Robotics Corporation and
--    FANUC LTD Japan, and must be returned to either Fanuc Robotics
--    Corporation or FANUC LTD Japan immediately upon request.  This material
--    and the information illustrated or contained herein may not be
--    reproduced, copied, used, or transmitted in whole or in part in any way
--    without the prior written consent of both Fanuc Robotics and FANUC.
--
--    All Rights Reserved
--    Copyright (C) 2000
--    Fanuc Robotics Corporation
--    FANUC LTD Japan
--            Karel is a registered trademark of
--            Fanuc Robotics Corporation
--    +
--    Program: tcpserv3.kl - Program for TCP Messaging
--
--    Description:
--
--    This program serves as an example on how to use TCP messaging and write
--    a server Karel program.
--
--    Authors: Fanuc Robotics Corporation
--            3900 West Hamlin
--            Rochester Hills, MI  48309

==    Modification history:
--
-------------------------------------------------------------------------------
--
PROGRAM tcpserv3
%RWACCESS
%STACKSIZE = 4000
%NOLOCKGROUP
%NOPAUSE=ERROR+COMMAND+TPENABLE
%ENVIRONMENT uif
%ENVIRONMENT sysdef
%ENVIRONMENT memo
%ENVIRONMENT kclop
%ENVIRONMENT bynam
%ENVIRONMENT fdev
%ENVIRONMENT flbt
%INCLUDE klevccdf
%INCLUDE klevkeys
%INCLUDE klevkmsk
-------------------------------------------------------------------------------
--
VAR
        file_var : FILE
        tmp_int  : INTEGER
        tmp_int1 : INTEGER
        tmp_str  : string[128]
        tmp_str1 : string[128]
        status   : integer
        entry    : integer
-------------------------------------------------------------------------------
--
```

Example 10.6.3 (b)

```
BEGIN
          SET_FILE_ATR(file_var, ATR_IA)
          -- set the server port before doing a connect
          SET_VAR(entry, '*SYSTEM*','$HOSTS_CFG[3].$SERVER_PORT',59002,status)
          WRITE('Connecting...',cr)
          MSG_CONNECT('S3:',status)
          WRITE(' Connect Status = ',status,cr)
          IF status = 0 THEN
               -- Open S3:
               WRITE ('Opening',cr)
               FOR tmp_int1 = 1 TO 20 DO
                     OPEN FILE file_var ('rw','S3:')
                     status = io_status(file_var)
                     WRITE (status,cr)
                     IF status = 0 THEN
                          -- write an integer
                          FOR tmp_int = 1 TO 1000 DO
                                WRITE('Reading',cr)
                                -- Read 10 bytes
                                BYTES_AHEAD(file_var, entry, status)
                                WRITE(entry, status, cr)
                                READ file_var (tmp_str::10)
                                status = io_status(file_var)
                                WRITE (status, cr)
                                -- Write 10 bytes
                                WRITE (tmp_str::10, cr)
                                status = io_status(file_var)
                                WRITE (status, cr)
                                      ENDFOR
                                      CLOSE FILE file_var
                      ENDIF
               ENDFOR
               WRITE('Disconnecting..',cr)
               MSG_DISCO('S3:',status)
              + WRITE('Done.',cr)
            ENDIF
END tcpserv3
```

## 10.6.4 ANSI C Loopback Client Example

Example 10.6.4(a) provides an example of a UNIX-based loopback client that can be used to establish a connection with a remote host.

Example 10.6.4 (a) ANSI C UNIX-Based Loopback Client Example

```
/* BSD Standard Socket Programming Example - UNIX */
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#define SERV_TCP_PORT 59002
#define SERV_HOST_ADDR "199.5.148.56"
#define MAXLINE 512
int written(int fd, char *ptr, int nbytes);
int readline(int fd, char *ptr, int maxlen);
void str_cli(int sockfd);
char *pname;
int main(int argc, char *argv[])
{
  int sockfd;
  struct sockaddr_in serv_addr;

  pname = argv[0];

  bzero((char *) &serv_addr, sizeof(serv_addr));
  serv_addr.sin_family = AF_INET;
  serv_addr.sin_addr.s_addr = inet_addr(SERV_HOST_ADDR);
  serv_addr.sin_port = htons(SERV_TCP_PORT);

  if((sockfd = socket(AF_INET, SOCK_STREAM,0)) < 0){
    printf("Client: Can't Open Stream Socket\n");
  }

  printf("Client: Connecting...\n");

  if(connect(sockfd,(struct sockaddr *) &serv_addr, sizeof(serv_addr))<0){
    printf("Client: Can't Connect to the server\n");
  }
  else{
    str_cli(sockfd);
  }
  exit(0);
```

Example 10.6.4 (b)

```
void str_cli (int sockfd)
{
  int n, i;
  char sendline[MAXLINE], recvline[MAXLINE + 1];
  while(1)
  {
    memset (sendline, 2, 128);
    if(written(sockfd, sendline, 126)!=126){
       printf("strcli:written error on sock\n");
      }
      i  = readline(sockfd, recvline, 126);
  }
}
int readline(int fd, char *ptr, int maxlen)
{
```

```
  int n, rc;
  char c;
  for(n = 0; n < maxlen; n++){
     if((rc = read(fd, &c, 1)) == 1){
        *ptr++ = c;

        if(c == '\n'){
          break;
         }
         else if(rc == 0) {
            if(n == 0) {
              return (0);
             }
             else{
               break;
            }
       }
    }
    else{
      return (-1);

  }}
*ptr = 0;
return (n);
 }
int written(int fd, char *ptr, int nbytes)
{
  int nleft, nwritten;
  nleft = nbytes;
  while(nleft > 0) {
    nwritten = write(fd, ptr, nleft);
    if(nwritten <= 0) {
       return(nwritten);
     }

     nleft -= nwritten;
     ptr += nwritten;
}
return(nbytes - nleft);
```

# 11 SIMPLE NETWORK TIME PROTOCOL (SNTP)

## 11.1 OVERVIEW

SNTP is a protocol used for synchronizing clocks. A personal computer (PC) acts as a central server, which serves as an accurate reference for the current date and time. SNTP is a subset of NTP (Network Time Protocol), and the protocols are compatible (NTP servers can reply to SNTP clients and vice-versa). The protocol is defined in RFC2030 (SNTP version 4).
The robot (SNTP client) gets the current date and time from a central NTP/SNTP server. The robot system clock need not be set manually on each robot. The current accurate time is received and the system clock is updated. The time is consistent across multiple robots in the cell. Accuracy of time can be useful on alarm timestamps, for instance.
This feature not only saves time and effort during robot installation, but also reduces human errors in setting time manually and keeps the current system clock accurate and consistent across multiple robots. The robot has the capability of using Daylight Saving Time (DST) locally and the local clock is automatically adjusted while DST is in effect.

## 11.2 SETTING UP SNTP

SNTP is installed using the SNTP option. The normal method for using SNTP is to fill out required fields in the SNTP interface screen (refer to Section 11.3 ). Some of the fields can only be configured by setting the system variable, $SNTP_CFG.

NOTE that the SNTP interface screen contains most of the fields in $SNTP_CFG. Unless you want to set the optional variable (i.e. $TIME_WIN) in $SNTP_CFG, you are encouraged to configure SNTP via SNTP user interface instead of setting system variables directly. Refer to Table 11.2(a) for detailed information on $SNTP_CFG.
$SNTP_CUSTOM needs to be filled in for users who meet any of following conditions:

- You live in the area where Daylight Saving Time (DST) policy changes annually. For example, Brazil and Israel determines when DST starts and ends every year.
- Your local DST policy is not same as the default one listed under Timezone in SNTP interface screen. For example, both Athens and Cairo belong to GMT+02:00 timezone but they have different DST policy. NOTE that Athens are chosen by default for GMT+02:00 timezone in SNTP user screen. Suppose that you live in Cairo and set Timezone as GMT+02:00 Athens. (DST is adjusted according to Athens DST policy, not based on Cairo DST policy).

You can set when DST starts and ends by setting $SNTP_CUSTOM. Currently there is no user interface screen provided for setting $SNTP_CUSTOM. You must set $SNTP_CUSTOM using the System Variable screen. Refer to Table 11.2(b) for detailed information on $SNTP_CUSTOM.

Table 11.2(a) $SNTP_CFG Settings

| System Variable | Default Value | Units | Description |
|---|---|---|---|
| $SNTP_CFG. $ENABLE | FALSE | N/A | Enable SNTP |
| $SNTP_CFG. $SERVER | " " | N/A | IP address or host name of NTP server. If DHCP is enabled and configured to provide NTP server address, this field is automatically set. If not, contact your IS department to get NTP server address. |
| $SNTP_CFG. $TIME_WIN* | 4 | Second | Local clock is adjusted only if the difference between the local clock and time server clock is greater than $TIME_WIN seconds. |
| $SNTP_CFG. $TZ_INDEX | 8 | N/A | Current index value of Timezone in user interface screen |
| $SNTP_CFG. $TZ_OFFSET | -300 | Minutes | Current offset from GMT(UTC) timezone in minutes without DST adjustment |
| $SNTP_CFG. $CUR_OFFSET | -300 | Minutes | Current offset from GMT(UTC) timezone in minutes with DST adjustment |
| $SNTP_CFG. $DST | TRUE | N/A | Enable Daylight Saving Time |

* $TIME_WIN is the only optional field that cannot be set from SNTP user interface.

Table 11.2(b) $SNTP_CUST Settings

| System Variable | Default Value | Unit | Description |
|---|---|---|---|
| $SNTP_CUSTOM. $START_MONTH | 4 | N/A | Enter Month when DST starts |
| $SNTP_CUSTOM. $START_DATE | 24 | N/A | Enter date when DST starts |
| $SNTP_CUSTOM. $START_HOUR | 2 | Hour | Enter time (in hour) when DST starts* |
| $SNTP_CUSTOM. $END_MONTH | 10 | N/A | Enter Month when DST ends |
| $SNTP_CUSTOM. $END_DATE | 17 | N/A | Enter date when DST ends |
| $SNTP_CUSTOM. $END_HOUR | 2 | Hour | Enter time (in hour) when DST ends* |

| System Variable | Default Value | Unit | Description |
|---|---|---|---|
| $SNTP_CUSTOM. $LOCAL_TIME | TRUE | N/A | If your DST is based on local time, set it TRUE. If your DST is based on GMT (UTC), set it FALSE** |
| $SNTP_CUSTOM. $NORTH_HEM | TRUE | N/A | If you live in North Hemisphere, set it TRUE. If you live in South Hemisphere, set it FALSE. |

* Set times in 24 hours scale. For example, if DST starts at 4 pm, set it to 16.
** Some countries (most of the countries in Europe) set DST start/end date and times based on GMT (UTC) rather than their local time. For example, DST starts in Berlin (1 am GMT(UTC) on 3/28). In this case, set all system variables in terms of GMT timezone and set $LOCAL_TIME =FALSE.

NOTE
Set the DST end time based on the local standard time not based on local Daylight Saving Time. For example, if your area ends DST in 10/17 3 am, based on the local Daylight Saving Time, type 10/17 2 am, based on the local standard time. $SNTP_CUSTOM variables are based on local stand time not based on Daylight Saving Time.

## 11.3 USING SNTP

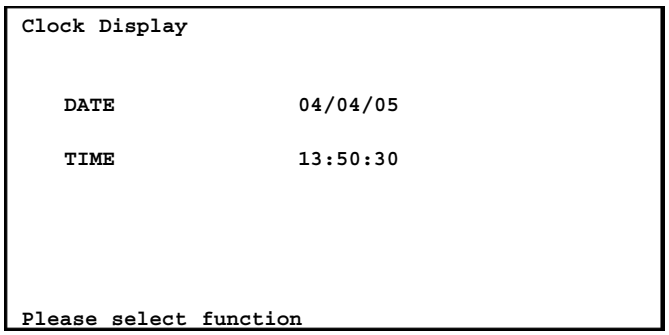
By default, SNTP client is disabled. In order to run SNTP client, use Procedure 11-1 .

### Procedure 11-1 Running the SNTP Client

#### Conditions

- SNTP option is installed.

1. Press MENUS.
2. Select System.
3. Select Clock. You will see a screen similar to the following.




NOTE
ADV appears in Clock screen if SNTP client is installed.

4. Press F2, ADV. You will see a screen similar to the following.

```
Advanced Setting                1/4


NTP enable:           FALSE
Daylight Saving Time:  TRUE

NTP server: 172.22.194.19
Timezone:   GMT-05:00 EST(US)
```

5. If NTP server field is not filled in, please contact your Information System (IS) department to get NTP server address. You can enter either the host name or IP address of NTP server. If the host name is used, ensure that DNS option is installed or the host name is entered in the host entry table.
6. Move the cursor to Timezone field and Press F4, [CHOICE].
7. Browse through Sub-menu and select your timezone. If your area has different DST rule from the default Timezone, please select CUSTOM. Refer to Table 11.3 for timezone and current DST policies. If you select CUSTOM, set the $SNTP_CUSTOM system variable, before you enable SNTP (Refer to Table 11.2(b) to set $SNTP_CUSTOM).
8. Move the cursor to Daylight Saving Time. If your area has DST and wants to enable DST, set it TRUE.
9. Move the cursor to NTP enable and set it TRUE after DST, NTP server, Timezone fields are configured.

Table 11.3 Timezone and Current DST Policies

| Timezone | Current DST Policies |
|---|---|
| GMT-12:00 Date Line | None |
| GMT-11:00 Samoa | None |
| GMT-10:00 Hawaii | None |
| GMT-09:00 Alaska<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT-08:00 PST (US)<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT-07:00 MST (US)<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT-06:00 CST (US)<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT-05:00 EST (US)<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT-04:00 AST (CAN)<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT-03:00 Buenos Aires | None |
| GMT-02:00 Mid-Atl | None |
| GMT-01:00 Azores<br>Starts:<br>Ends: | Last Sunday in March at 1 am GMT (UTC) time<br>Last Sunday in October at 1 am GMT (UTC) time |

| Timezone | Current DST Policies |
|---|---|
| GMT-00:00 London<br>Starts:<br>Ends: | Last Sunday in March at 1 am GMT (UTC) time<br>Last Sunday in October at 1 am GMT (UTC) time |
| GMT+01:00 Berlin<br>Starts:<br>Ends: | Last Sunday in March at 1 am GMT (UTC) time<br>Last Sunday in October at 1 am GMT (UTC) time |
| GMT+02:00 Athens<br>Starts:<br>Ends: | Last Sunday in March at 1 am GMT (UTC) time<br>Last Sunday in October at 1 am GMT (UTC) time |
| GMT+03:00 Moscow<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT+04:00 Baku<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT+05:00 Islamabad | None |
| GMT+06:00 Dhaka | None |
| GMT+07:00 Jakarta | None |
| GMT+08:00 Beijing | None |
| GMT+09:00 Tokyo | None |
| GMT+10:00 Sydney<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT+11:00 Noumea | None |
| GMT+12:00 Auckland<br>Starts:<br>Ends: | First Sunday in April at 2 am local standard time<br>Last Sunday in October at 3 am daylight saving time |
| GMT+13:00 Nukualofa | None |
| CUSTOM<br>Starts:<br>Ends: | The user set these fields<br>The user set these fields |

## 11.4 TROUBLESHOOTING

The robot SNTP is designed to run based on the multicast packets sent by NTP server. However, it is possible that multicast packets might not be delivered to the robot:

- The NTP server might be configured to serve only unicast packets
- Multicast packets could be lost along the hops between NTP server and the robot (for example, switch/hub configurations along the hops)

When multicast packets are not delivered to the robot, the robot SNTP relies on the unicast packets. It sends unicast packet to NTP server every 1 hour to update the clock.

# 12 ETHERNET PACKET SNIFFER

## 12.1 OVERVIEW

The robot controller communicates over Ethernet by sending and receiving messages over the network called *packets*. Capturing, also called *sniffing*, these packets is often helpful in diagnosing a wide variety of Ethernet communication problems.
The Ethernet Packet Sniffer allows for packets to be captured directly on the robot controller and then saved to a file. This file can then be viewed and analyzed offline on a PC using any software supporting the tcpdump capture file format. We recommend using the free, open source Wireshark Network Protocol Analyzer (formerly known as Ethereal) available for download off the Internet for viewing and analyzing the capture file.
Basic filters and triggers can be applied when running the packet sniffer.

NOTE
Enabling the robot Ethernet sniffer may have a small effect on robot Ethernet performance. While running the Ethernet sniffer can be very helpful in debugging and diagnosing problems, take care when enabling the robot Ethernet sniffer in a production environment while running aggressive real-time Ethernet protocols on the robot controller.

## 12.2 SETTING UP THE ETHERNET PACKET SNIFFER

Before you can use the Ethernet Packet Sniffer, you must first define TCP/IP parameters. Table 12.2 describes the Ethernet Packet Sniffer items you can set up. Use Procedure 12-1 to set up the Ethernet Packet Sniffer.

Table 12.2 Ethernet Packet Sniffer Setup Items

| ITEM | DESCRIPTION |
|---|---|
| Status<br>Default: OFFLINE | The status of the sniffer. Possible values are ONLINE, and OFFLINE. This field is not editable. |
| Packet<br>Count<br>Default: 0 | The number of Ethernet packets sniffed. This field is not editable. NOTE that if the ring buffer is used, this will display the total number of packets sniffed and not the number of packets in the current ring buffer. |
| Filter<br>Default: NONE | The name of the filter being used. Only packets matching the filter are captured. |
| Trigger<br>Default: NONE | The name of the trigger being used. When the trigger is matched, the capture is stopped. |
| Interface<br>Default: Port1 | The name of the interface being used. Possible values are Port1 and Port2. |
| Dump file<br>Default fr:sniff.cap | The location where the Ethernet capture file is to be saved. |
| Buffer Size<br>Default: 50000 | The maximum number of bytes used for capturing and storing an Ethernet capture on the controller. This determines the size of the capture buffer. |
| Ring Buffer<br>Default: FALSE | Determines if the capture should run continuously with a ring buffer, or if the capture should automatically stop when the capture buffer is exhausted. |

## Procedure 12-1 Setting Up the Ethernet Packet Sniffer

1. Press MENUS.
2. Select SETUP.
3. Press F1 [TYPE] and select Host Comm.
4. Under Protocol, move the cursor to SNIFF and press F3, [DETAIL]. You will see a screen similar to the following:

```
Ethernet Packet Sniffer Status
                                6/6

Current Monitor
 Status :          OFFLINE
 Packet count:      0
 Filter :          NONE
 Trigger :         NONE
 Interface :       Port1
 Dump File :       fr:sniff.cap
 Buffer Size :     50000
 Ring Buffer :     FALSE
```

5. Select an appropriate Filter and Trigger (or leave at the default: NONE).
6. Select the interface on which to perform the capture. Port 1 corresponds to Port 1 in the TCP/IP SETUP screens and CD38A on the controller, while Port 2 corresponds to Port 2 in the TCP/IP SETUP screens and CD38B on the controller.
7. Type in a file name of where you want to capture file to be saved.
8. Enter an appropriate buffer size, or leave at the default of 50000 bytes.
9. Select whether a ring buffer should be used.
10. Press F2, [START] to start the Ethernet Packet Sniffer.
11. Press F3, [STOP] to stop the Ethernet Packet Sniffer.
12. Press NEXT then F2 [SAVE] to save the capture to an Ethereal file.

> NOTE
> While the Ethernet Packet Sniffer is running, many of the editing and function keys on this screen will be disabled.

> NOTE
> The Ethernet Packet Sniffer does not run in promiscuous mode. That is, the Ethernet Packet Sniffer will only capture Ethernet packets addressed to the robot controller (including broadcast packets and multicast packets from a joined group) and packets sent from the robot controller.

# 12.3 USING THE RING BUFFER AND TRIGGERS

In many instances it is impossible to predict exactly when a communication problem will occur. Some symptoms appear sporadically, which makes troubleshooting all the more difficult.
The ring buffer allows the Ethernet Packet Sniffer to capture packets continuously. When Ring Buffer is enabled, two buffers of identical size are created. When one buffer fills up with captured Ethernet Packets, the other buffer will be overwritten. This will continue until either the capture is stopped, or a trigger is triggered--which in turn stops the capture. With a proper trigger, the communications anomaly can be detected, and the capture can then be stopped immediately. The resulting capture file will contain a snapshot of the Ethernet packets during the occurrence of the communications anomaly.

# 13 ROS INTERFACE PACKETS OVER ETHERNET (RIPE)

## 13.1 OVERVIEW

The Real Time Operating System (ROS) Interface Protocol over Ethernet feature (also called Robot Ring, RIPE or ROSIP) allows robots doing a common job to share information.
This feature also supplies a method of the clocks on multiple robots. Therefore, information can be communicated with respect to a common time base.
The robot ring consists of a single designated master robot and some number of slave robots. The master maintains the master timing information and a different setup requirement. The slaves are all about the same.
The position of the slaves in the ring is important to identify each slave by number. The position determines the *index* of a particular robot in the ring. The position in the ring also determines file access and TELNET connection.
The controller has two Ethernet ports. In a typical application one Ethernet port is connected to a large factory network for backups and other maintenance operations. The second Ethernet port is for dedicated real time protocols. RIPE is intended to be one of those protocols. It is required in order to use RobotLink or *i*RVision.
All of the robots in the RING can be referred to by their designated names. If you do not set the name of the robot in the host setup menus, RIPE will pick a name for you. The default name will be ROBCONT1 - ROBCONTn where n is the number of robots in the ring. It is recommended that you pick a name. The name can be set from the general HOST COMM SETUP screen or the RIPE SETUP screen. For this manual the names MHROB0x are used as an example of a user-defined name.

NOTE
Depending on your RIPE configuration it MAY be necessary to power up the master robot first. On very rare occasions not doing this could lead to long synchronization times or unsuccessful initialization.

## 13.2 RIPE SETUP

Table 13.2 defines the items needed to set up the Master and Slaves in a Ring. Use Procedure 13-1 to set up the Master and Slave in a Ring.

Table 13.2 RIPE Setup Master and Slave Items

| ITEM | DESCRIPTION |
|---|---|
| Displayed on Master and Slave Screen | |
| Robot Name | This item indicates the name of the robot in the ring. |
| Port # | This item indicates the port number to use for RIPE. If possible, use the port that is not already in use for a factory communications link. Typically, port #2 is available for RIPE and other robot to robot communications. |

| ITEM | DESCRIPTION |
|---|---|
| Displayed on Master Screen Only | |
| Master IP Address | This indicates the IP address of the master.<br><br>CAUTION<br>The master IP address must be the same for all robots in the ring. Otherwise, communications will not work properly.<br><br>If the master IP address is set incorrectly, move the cursor to the Master IP address and type in the correct address. |
| Number of Members | This item indicates the number of robots in the ring. |
| Update Interval | This item indicates the heartbeat time in milliseconds (ms). This is how often RIPE checks to see which robot is online. |
| Displayed on Slave Screen Only | |
| Slave IP address | This item indicates the IP address of the slave. |
| Member Index (1 is Master) | This item should be set to a unique sequential number such as 2, 3, 4, or 5. Press F3, AUTO, to configure the robots to WAIT for the config file to be sent from the master |

## Procedure 13-1 Setting Up a Master and Slave in a Ring

1. Press MENUS.
2. Select SETUP.
3. Select Host Comm. The master screen will be displayed. See the following screen for an example.

```
SETUP RIPE
ROS Ethernet Packets (MASTER)     1/17
   Robot Name:              MHROB01
   Port #:                        2
   Master IP addr:   192.168.0.101
   Number of Members:             2
   Update Interval:             400

     Host Name     Internet Address
   1 MHROB01          192.168.0.101
   2 MHROB02          192.168.0.102
   3 MHROB03          192.168.0.103
   4 *******          *************
```

4. Type a new name for your robot. MHROB01, for example.
5. Press F2, SLAVE. The slave screen will be displayed. See the following screen for an example. Press F2 again to display the MASTER screen.

```
SETUP RIPE
ROS Ethernet Packets (SLAVE)      1/14
   Robot Name:               MHROB02
   Port #:                         2
   Master IP addr:    192.168.0.101
   Slave IP addr:     192.168.0.102
   Member Index (1 is Master):    2

     Host Name     Internet Address
   1 MHROB01          192.168.0.101
   2 MHROB02          192.168.0.102
   3 MHROB03          192.168.0.103
   4 **********       *************
```

6. Set up the ring on the slave robots:

a. Select a unique "Member Index" for all slave robots 2 to n where n is the number of robots in the ring.
b. Press F3, AUTO on all of the SLAVES to configure them to WAIT for the config file from the master.

7. Set up the ring on the master robots:
   a. Verify that the port number is correct. If possible, use a port that is not already in use for a factory communications link. Typically, port #2 is available.
   b. Set up the number of members in the ring.
   c. Set the Update Interval.
   d. Press F3, AUTO, on the MASTER to generate the config file and send it to all of the waiting slaves.
8. You can manually edit the ROSIPCONFG.XML file if necessary. Use F4, SEND, to send the files to the Slaves manually.
9. Press F3, LOAD to load the ROSIPCFG.XML file.
10. Press >, NEXT and then F4, RECV on the slave to put the slave into a mode where it will receive the config file from the master.

## 13.3 FILE ACCESS

RIPE provides access to any file on any robot in the RING from any other robot in the ring. There are two ways in general to access a file on another robot.

- By name
- By its member index or position in the RING configuration.

To access a file on MC: on MHROB03 from MHROB01 the following syntax applies:

- RNG:¥MHROB03¥MC¥<filename>
- RNG3:¥MC¥<filename>

For V7.30 DOS directory name size limits exist. For any robot with a name longer than eight characters the first syntax might not work. If possible, the name should be limited to eight characters. In the case of RNG3: access, the robot name does not matter.
All Web server access applies via this interface. So in the case where you want to get to MHROB03 Error and Diagnostic files but the available Ethernet interface is MHROB01 the following access applies:

```
http://MHROB01/rng/MHROB03/MD/INDEX_ER.HTM
http://MHROB01/rng3/MD/INDEX_ER.HTM
```

An FTP example is shown below:

```
ftp> cd rng:\MHROB02
250 CWD command successful.
ftp> cd rng:\MHROB02\fr
250 CWD command successful.
ftp> dir
200 PORT command successful.
150 ASCII data connection.
-rw-rw-rw- 1 none nogroup      1493 apr 23 2007 inactive.htm
```

## 13.4 ASSOCIATED OPTIONS

There are a number of other application which use RIPE internally. The RIPE setup is required for these applications:

- Intelligent Interference Check
- Ethernet Line Tracking
- Multi-application Shell

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 effected 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, EtherNet/IP nodes) by using a separate subnet for the control equipment or by using VLANs (virtual LANs) supported by some higher end switches. If the EtherNet/IP network is completely isolated as a separate control network this is not a concern. However, when connecting into larger networks this becomes important.

Some network environments have a significant amount of multicast traffic. A basic layer 2 switch will treat multicast traffic like broadcast traffic and forward to all ports in the switch wasting network bandwidth and node resources on traffic which is ultimately dropped for the nodes that are not interested in the multicast traffic. Switches that support "IGMP snooping" will selectively send multicast traffic only to the nodes which have joined a particular group. EtherNet/IP UDP packet has a TTL (time to link) value of one. You will not be able to route I/O traffic across more than one switch.

Quality of Service (QOS) techniques provide 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.

It is important to select the proper switch in order for the network to function correctly. The switch should support :

- 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, the switch should support IGMP snooping (multicast aware). Please consider this when Ethernet/IP and/or RIPE (robot ring) traffic exists.

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:

- Cisco 2955 (industrialized version of 2950) – www.cisco.com
- Hischmann MICE (modular industrial switch) – www.hirschmann.de
- Phoenix Contact (managed/unmanaged industrial switch) – www.ethernetrail.com
- N-Tron 508TX-A, 8 port industrial switch with advanced firmware – www.n-tron.com

# 14 PC SHARE

## 14.1 OVERVIEW

PC Share allows the robot controller to connect to a Shared Folder on a PC running a supported version of Microsoft® Windows®, or on a Linux computer running a support version of SAMBA (www.samba.org), using the SMB/CIFS protocol. This "connection" to the Shared Folder allows the robot controller to browse the remote directory tree, and to read and write files remotely. This functionality is similar to the "mapping a drive" concept in Microsoft® Windows® operating systems--the robot is able to "map" a client tag to a directory location on the Windows®-based network. (Microsoft, Windows, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries).
Only client functionality is supported, that is the robot controller can access files on a remote PC; server functionality is not supported on the robot, a PC cannot access files on the robot controller.

## 14.2 SETTING UP AND STARTING PC SHARE

### 14.2.1 Setting Up PC Share Client Tags

Before you can use the PC Share Interface, you must do the following:

- Define TCP/IP parameters (Section 2.4 ).
- Order R558 INTERNET CONN/CUSTO for the robot controller.
- Define PC Share on a client device (Procedure 14-1 ).

Table 14.2.1 lists and describes the items you must set up to define a client device.

Table 14.2.1 Client Device Definition Setup Items

| ITEM | DESCRIPTION |
|---|---|
| Tag | This item specifies the device name client. Available client tags are C1: through C8:. |
| Comment | This item provides an area for you to include up to 16 characters of information that allow you to label the device for its application use. |
| Protocol* | This item specifies the name of the protocol that will be associated with the tag. For PC Share, select PC Share. |
| Port Name | This item is only displayed when SM (Socket Messaging) is selected as the Protocol, and does not apply to other protocols. |
| Startup State* | This item specifies the desired startup (Power up) state for the selected tag. Three states are possible:<br>● UNDEFINED - the device is not defined.<br>● DEFINED - the device is defined.<br>● STARTED - the device is defined and started. |
| Server IP/Hostname* | This item specifies the Hostname or IP address of the remote server to which the connection will be made. |
| Remote Path/Share* | This item specifies the host path on the server, to be used for file operations, up to 64 characters. When using the PC Share protocol, the Share name must be included. |

| ITEM | DESCRIPTION |
|---|---|
| Inactivity Timeout | This item specifies the number of minutes of inactivity on the network before a connection will be closed.<br>● When set to **zero**, no timeouts occur.<br>● When set to a **non-zero value,** Inactivity Timeout specifies the number of minutes of inactivity on the network before a connection will be closed. The default value is 15 minutes. |
| Username* | This item specifies the username to use when logging into the remote server. The username is case sensitive based on the host system that checks it. |
| Password* | This item specifies the password to use when logging into the remote server. The password is case sensitive based on the host system that checks it. |
| * This item is normally set up by the user. Other items can normally remain at their default values. | |

NOTE
For security reasons, the password is not stored in the backup file. You must enter the password each robot in each teach pendant. Password that you enter will not be able to save from memory in the controller. Please always keep a copy in your password that you set.

Use Procedure 14-1 to define and start PC Share on a client device.

## Procedure 14-1 Defining and Starting PC Share on a Client Device

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm. You will see a screen similar to the following.

```
SETUP Protocols
     Protocol       Description
  1  TCP/IP         TCP/IP Detailed Setup
  2  TELNET         Telnet Protocol
  3  PC SHARE       PC Share Setup
  4  PING           Ping Protocol
  5  HTTP           HTTP Authentication
  6  FTP            File Transfer Protocol
  7  DNS            Domain Name System
```

5. Press F4, [SHOW].
6. Select 2, Clients. You will see a screen similar to the following.

```
SETUP Clients
    Tag  Protocol  Remote    State
  1 C1:  ********  **********[UNDEFINED]
  2 C2:  ********  **********[UNDEFINED]
  3 C3:  ********  **********[UNDEFINED]
  4 C4:  ********  **********[UNDEFINED]
  5 C5:  ********  **********[UNDEFINED]
  6 C6:  ********  **********[UNDEFINED]
  7 C7:  ********  **********[UNDEFINED]
  8 C8:  ********  **********[UNDEFINED]
```

7. Move the cursor to the client tag you want to set up and press F3, DETAIL. See the following screen for an example.

## 14.2.2 Configuring PC Share

Table 14.2.2 details the PC Share items that can be configured from the Host Comm screens. Use Procedure 14-2 to configure these items. The robot controller must be power-cycled for any new settings to take effect.

By default, if no response to a robot PC Share request is received from the remote PC Share server, after 8 seconds the connection will timeout with a HOST-281 "SMB: Time-out waiting for PC" alarm. When working with a slower remote PC, this timeout can be adjusted by modifying the value, in seconds, of the system variable SMB_CLNT[X].$RSPTMOUT, where X is the client tag index CX:, for a specific client tag. Power-cycle the robot controller for the new setting to take effect.

When the PC Share option is installed on the controller, hostname resolution is performed in the following order:

- First the Host Name (Local and Shared) tables are checked for the hostname.
- Second, if DNS is installed, the configured DNS server is queried for hostname resolution.
- Third, if a WINS server is configured, the WINS server is queried for hostname resolution.
- Lastly, if PC Share Broadcast Discovery is enabled, a broadcast message is sent to query the local subnet for a PC with the hostname.

Table 14.2.2 PC share configuration items

| ITEM | DESCRIPTION |
|---|---|
| Enable<br>(Default: TRUE) | The PC Share protocol can be completely disabled by setting this item to FALSE. The robot must be power cycled for this setting to take effect. |
| Domain | Optional. A domain may be configured if required by your network. |
| WINS server | Optional. The IP address of the WINS server to be used for hostname resolution. |
| Broadcast Descovery<br>(Default: TRUE) | Enables or disables broadcast discovery. If no WINS server is specified, or if a WINS server is specified but cannot resolve a hostname, the robot controller will send a broadcast discovery query to see if a PC with the specified hostname exists on the local subnet. Set this to FALSE to prevent this broadcast query from being sent. |

### Procedure 14-2 Configuring PC Share

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm. You will see a screen similar to the following.

```
SETUP Protocols
     Protocol        Description
  1  TCP/IP          TCP/IP Detailed Setup
  2  TELNET          Telnet Protocol
  3  PC SHARE        PC Share Setup
  4  PING            Ping Protocol
  5  HTTP            HTTP Authentication
  6  FTP             File Transfer Protocol
  7  DNS             Domain Name System
```

5. Select PC SHARE and press enter. You will see a screen similar to the following.

```
PC Share
 PC Share Client
  Enabled       :   TRUE
  Domain        :   *****************
  WINS Server   :   *****************
  Broadcast Desc :   TRUE
```

6. To enable the PC Share protocol, set Enable to TRUE. To disable the PC Share protocol, set Enable to FALSE.
7. To configure a domain, cursor to Domain and enter a domain name. This may be optional depending on your network.

NOTE
To override the domain set in the PC Share Client screen with a domain to be used only for a specific client tag, set the domain in the system variable $SMB_CLNT[X].$DOMAIN, where X is the client tag index CX:, for the specific client tag. Power-cycle the robot controller for new settings to take effect.

8. To configure a WINS server, curser to WINS and enter the server's IP address.
9. To enable broadcast discovery, set Broadcast Disc to TRUE. Set Broadcast Disc to FALSE to disable broadcast discover.

# 14.3 ACCESSING AND USING PC SHARE CLIENT DEVICES

## 14.3.1 Access Description

A client device does not have to be started before it is accessed. However, the *tag* must be defined. The device will automatically be started when opened and will automatically return to the defined state when closed.

NOTE
Please do not following while accessing the shared folder on a remote PC from the robot. There is a possibility that the contents of the shared folder on the remote PC is damaged or file access processing of the robot will not be interrupted.

NOTE
The time required for saving or reading file is affected by the processing power of the server network traffic load and PC. If you use the function of PC share to store the log files FANUC iRVision, execution time of iRVision will change significantly or slower by saving the log file. Please do not save the log file of iRVision using PC share function while running the robot for production, because there is a possibility that the cycle time of the robot become worse.

NOTE
If you save log files from robots with iRVision using the function PC share to one remote PC, the path name of the shared folder in each robot should be set differently to use a shared folder for each robot is different.

## 14.3.2 File Specification for PC Share Client Devices

Client devices are used like local file storage devices. The file specification for a PC Share client device is as follows:

```
device_name:<path_name\>file_name.file_type
```

This is a modified MS-DOS format. Single quotes can be used to delimit strings or characters unacceptable to MS-DOS, such as the "¥" character. The full definitions are as follows:

- **device_name** is a two to five character device name field, followed by a colon. The first character must be a letter; the remaining characters must be alphanumeric. The default device from the system console variable $DEVICE will be used if this field is absent (C1:, for example).
- **path_name** is a recursively defined optional field consisting of one or more **file_names** separated by a backslash. It is used to select the Share, subdirectory, or Share¥subdirectory. The path_name is appended to the configured Remote Path/Share Client Tag item and the total path can consist of up to a maximum of 64 characters.
- **file_name** is the name of the remote file.
- **file_type** is the file extension.

## 14.3.3 Starting and Stopping a Client Device

Use Procedure 14-3 to start, stop, and configure the client device and to start it automatically when the controller is turned on.
Client tags can be turned on in the defined state. They will be started automatically when accessed.

### Procedure 14-3 Starting and Stopping a PC Share Client Device

1. Press MENUS.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm.
5. Press F4, [SHOW].
6. Select Clients. You will see a screen similar to the following.

```
SETUP Clients
    Tag  Protocol  Remote    State
  1 C1:  ********  **********[UNDEFINED]
  2 C2:  ********  **********[UNDEFINED]
  3 C3:  PC Share  **********[UNDEFINED]
  4 C4:  ********  **********[UNDEFINED]
  5 C5:  ********  **********[UNDEFINED]
  6 C6:  ********  **********[UNDEFINED]
  7 C7:  ********  **********[UNDEFINED]
  8 C8:  ********  **********[UNDEFINED]
```

7. Move the cursor to the tag on which you wish to perform an action.
8. Press F2, [ACTION].
9. Select the action you want to perform:

NOTE
A device must be in the defined state before it can be started.

- To define a device, select DEFINE. The device must be in the undefined state.
- To undefine a device, select UNDEFINE. The device must be in the defined state.
- To start a device, select START. The device must be in the defined state.
- To stop a device, select STOP. The device must be in the started state. The device will change to the defined state.

10. To configure the client device to start automatically at power up:
    a. Move the cursor to the client tag you want to start automatically and press F3, DETAIL.
    b. Move the cursor to Startup State and press F4, [CHOICE].
    c. Select START, and press ENTER.

The client device will now start automatically when the controller is turned on.

> NOTE
> The host device MUST be capable of accepting this PC Share login if the tag is set to START AUTOMATICALLY when you turn the robot on. If the remote host is not available, the robot controller will wait approximately one minute to timeout before completing powerup. For this reason it is recommended to have client tags powerup in the DEFINE state. The controller will automatically start any defined client tag when the tag is used.

## 14.3.4 Teach Pendant File Access

After a client device has been defined, it can be used from the teach pendant.
On the teach pendant, when you set the default device to C1:, you can do the following:

- **From the SELECT screen**
  - Save a program to C1:
  - Load a program from C1:
- **From the FILE screen**
  - Generate a directory of files on C1:
  - Load or restore files from C1: onto controller memory
  - Back up program and system files to C1:
  - Copy files to and from C1:
  - Delete files from C1:

## 14.3.5 Saving FANUC iRVision log to PC share folder

You will be able to save iRVision log files to a shared folder on a remote PC by setting the location for log files as the PC share client device.

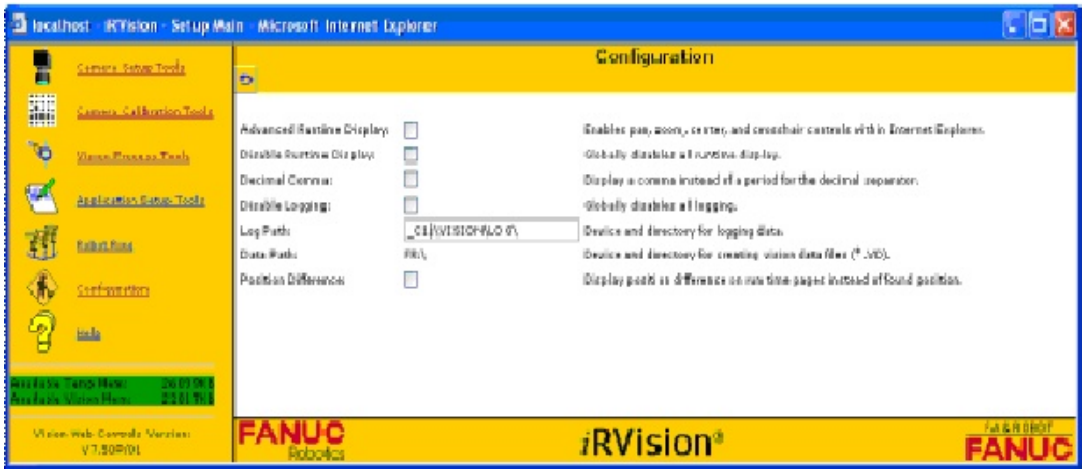



Fig. 14.3.5 Using PC share function for iRVision log

# 14.4 PASSWORDS AND SECURITY

## 14.4.1 Passwords

It is recommended that the robot controller(s) be assigned its own username and password for the Windows®-based network to which it will be connecting. Doing so illustrates good security practice by

reducing the chance of a user's personal password being compromised, and allowing file and Share permission to be set appropriately for the robot controller on the remote PC(s).
However, it is expected that some users will instead choose to enter their corporate or otherwise sensitive authentication information into the robot from time to time to allow the robot to access files and Shares on their Windows®-based network via PC Share. To this end, the following security precautions have been taken on the robot controller to prevent PC Share passwords from being accidentally or maliciously compromised.
Passwords entered into a Client Tag configured for the PC Share protocol are not stored on the robot controller. Rather, the password in read in, and converted to a 21-byte NTLM session key using the RSA MD-4 hash algorithm. Then the hashed password is salted with an additional 3-bytes of random data, and the resulting 24-bytes are encrypted using the DES (Data Encryption Standard) algorithm using an encryption key unique to the specific robot controller. Any DRAM used to read in the original password is cleared. The original plain text password cannot be recovered even by the robot controller software.
Storing (or forgetting) the password in this manner works because only the 21-byte NTLM key is needed by the client device to connect and authenticate to the remote server. The additional salting and encrypting of the NTLM key is an additional layer of protection to prevent anyone from loading the NTLM key on a different robot. As such, PC Share passwords cannot be backed-up and restored onto a different controller.

NOTE
The password must be entered through the Teach Pendent on each robot using PC Share client tags. A PC Share password cannot be backed-up or restored.

## 14.4.2 Supported Security Protocols

The robot PC Share option supports the NT LM 0.12 CIFS dialect. PC Share supports NTLM authentication (NTLMv2 is not supported). And, if required by the server, PC Share also supports Server Message Block (SMB) signing, also known as security signatures (see Article ID: 887429 at Microsoft's support web site, support.microsoft.com, for more information on configuring security signatures). PC Share has been tested against Windows® XP, Windows Vista®, and Windows® 2003 server, as well as Linux SAMBA versions 2.2.5, 3.0.24, and 3.2.5.

# 14.5 CONFIGURE THE REMOTE PC

Procedure 14-4 will walk through the steps required to setup a Share in Windows® XP. See also Figure 14.3.5 .
Procedure 14-5 will walk through the steps required to setup a Share in Windows Vista®.

### Procedure 14-4 Sharing a folder in Windows® XP

1. Open Windows® Explorer (or double click on My Computer) and browse the folder you wish to share.
2. Right click on the folder, and select Properties.
3. Select the Sharing tab.
4. Click the "Share this folder" radio button.
5. Type in a Share name and optionally a comment.
6. Optionally, set User limits.
7. Click on Permissions. Make sure the proper Allow permissions are set. If you will be backing up files to the Share from the robot, allow Full Control so the robot can create the files. Click on OK.
8. Verify that your Firewall settings allow the folder to be shared with other computers on the network. If not, make a firewall exception for File and Printer Sharing.
9. Click on OK.
10. The robot may now connect to this folder using the Share name that was input in step 5.

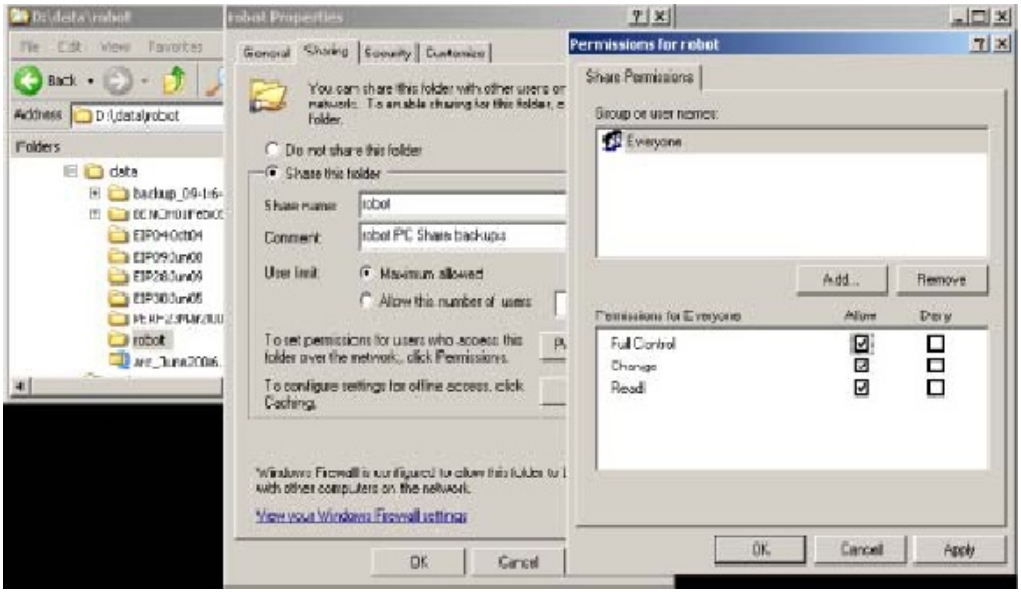



Fig. 14.5(a) Sharing a Folder in Windows® XP

## Procedure 14-5 Sharing a folder in Windows Vista®

1. From the Control Panel, go to the Network and Sharing Center. See Figure 14.5(b) .
2. Turn on "Network discovery". This allows your computer to be visible to other network devices.
3. Turn on "File sharing". This allow you to share files on the computer.
4. Turn on "Password protected sharing". This forces users to log onto the computer with a username and password.
5. Browse to the folder you wish to share and right click on it. Select "Share..." See Figure 14.5(c) .
6. Select the users to share with, set the appropriate Permission Level, and click on the Share button. See Figure 14.5(d) .
7. When you are prompted with the "Your folder is shared" window, Click on the Done button.
8. The robot may now connect to this folder using the name of the shared folder as the Share name.
9. For advanced sharing options, right click on the folder you wish to share, select Properties, select the Sharing tab, then select Advanced Sharing.

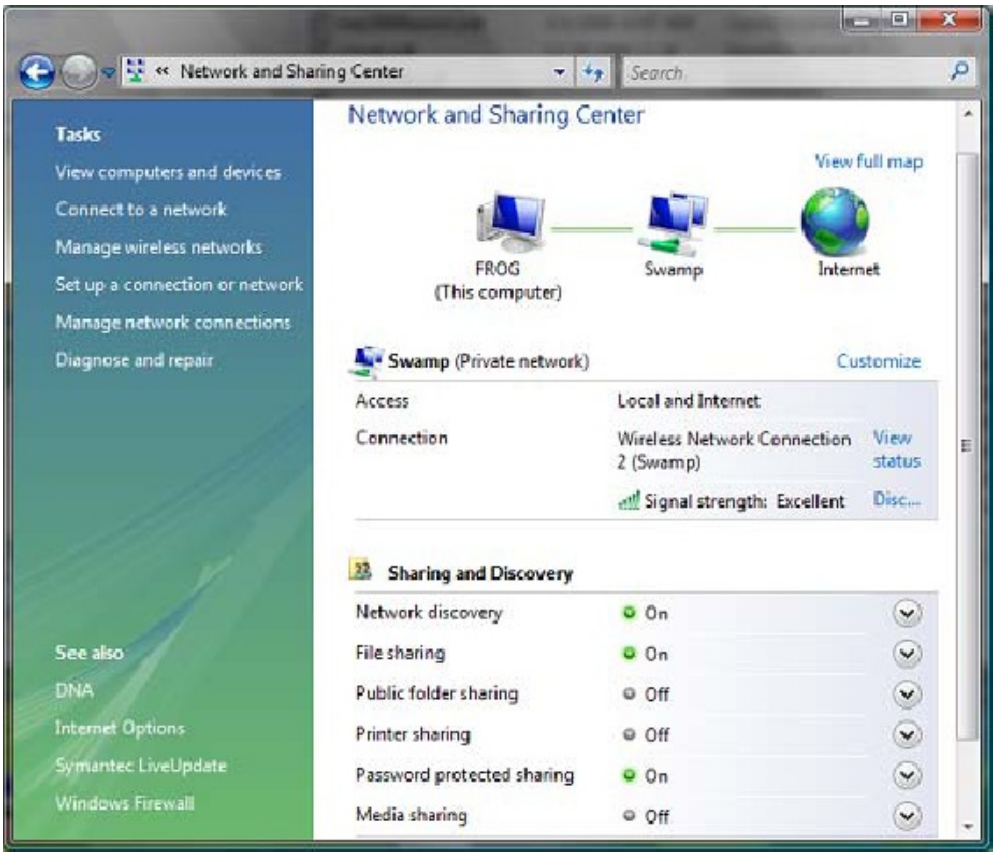



Fig. 14.5(b) Microsoft Vista® Network and Sharing Center

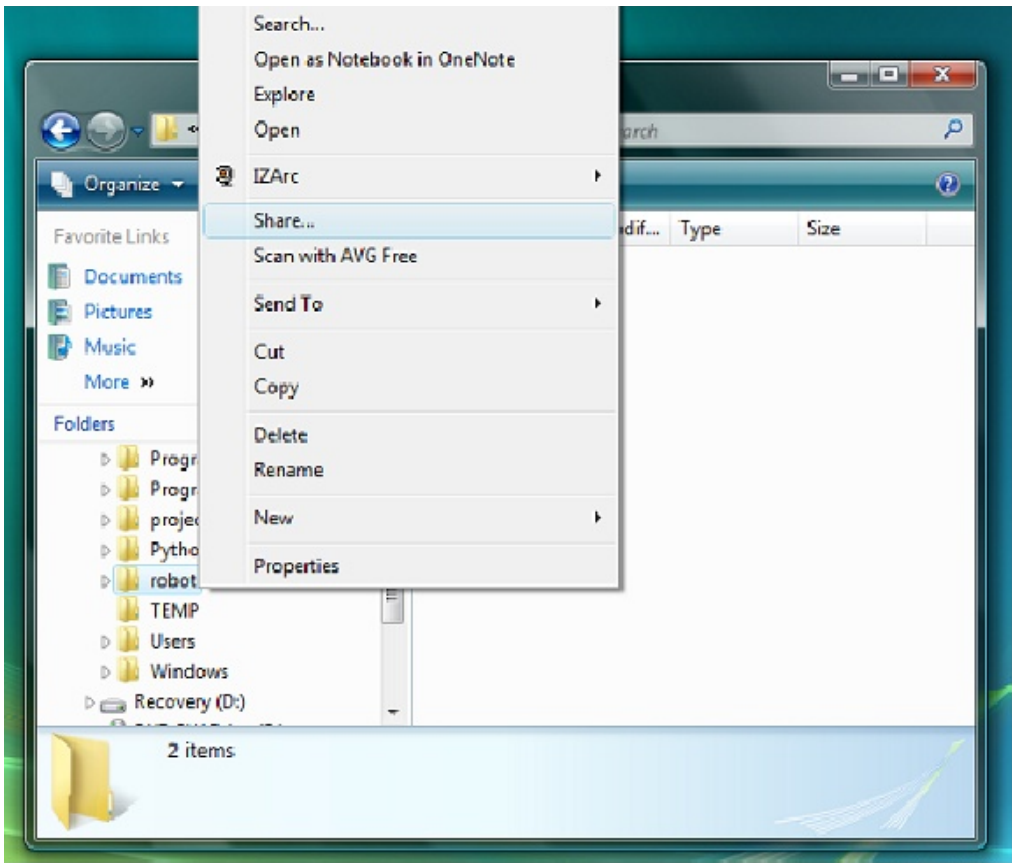



Fig. 14.5(c) Selecting Share... in Windows Vista®

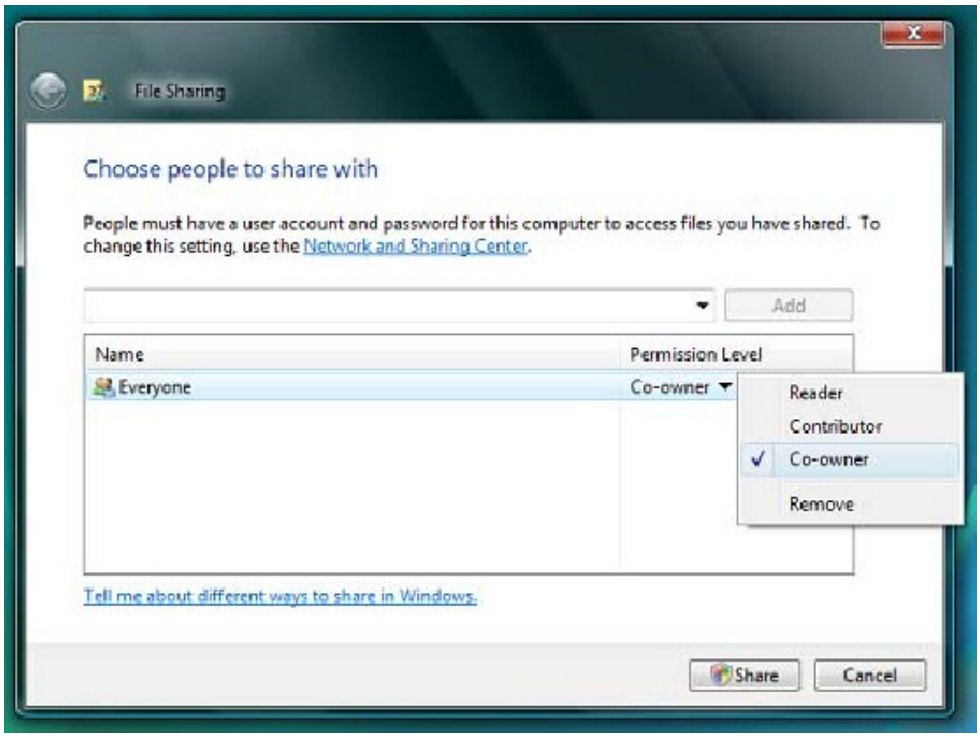



Fig. 14.5(d) Choose People to Share With in Microsoft Vista®

# 15 ADVANCED *i*Pendant FUNCTIONS

## 15.1 OVERVIEW

This chapter provides information about the advanced connectivity and customization features of the *i*Pendant. These include:

- **Installing i*Pendant Controls*** This allows you to install the components necessary to perform any *i*Pendant advanced functions.
- **Remote Monitoring** This allows you to monitor the *i*Pendant screens remotely using a PC and Microsoft® Internet Explorer.
- **Remote Operation** This allows you to display an *i*Pendant-like screen remotely, using a PC and Microsoft® Internet Explorer, and navigate the various menus available in the system. This provides the capability to diagnose system problems remotely.

## 15.2 *i*PENDANT CONTROLS INSTALLATION

Some of the *i*Pendant advanced functions require you to have the FANUC Robotics *i*Pendant Controls loaded on the PC that you will be using. The manual for *i*Pendant Screen Display Software, which attached to CD for A08B-9410-J803, describes how and where to obtain these controls and install them on your PC.

## 15.3 REMOTE MONITORING

### 15.3.1 Overview

Remote monitoring provides you with the capability to display and monitor the current *i*Pendant screens and operations on a PC using Microsoft® Internet Explorer. It is meant as a DISPLAY ONLY mode and therefore, the remote connection normally cannot interact with the screens or affect the operation of the *i*Pendant or robot controller. See Section 15.3.4 for limitations.
The following can be displayed remotely:

- All *i*Pendant screens available from the MENUS and [TYPE] keys.
- All popup menus, and windows.
- Multiple window configurations (Double and Triple modes on the *i*Pendant for example).
- Any input from the *i*Pendant numeric keypad, Function keys or cursor movement.
- Any custom screens that are accessible from the [TYPE] menu in the BROWSER Screen.

### 15.3.2 Setup

Setup consists of:

- Identifying requirements
- Configuring Internet Explorer ®
- Testing the Network Connection

#### 15.3.2.1 Requirements

The following are the requirements for remote display of the *i*Pendant screens.

- The PC must have Microsoft® Internet Explorer 5.5 or greater installed.
- The PC must have the *i*Pendant Controls installed. Refer to Section 15.2 for installation instructions.

- The PC must be connected to a network , and be properly configured to allow a TCP/IP connection to the robot controller with the *i*Pendant connected.
- The robot controller must be connected to a network and be properly configured for Network access to the above PC.

### 15.3.2.2 Configuring Internet Explorer ®

The following settings are required in Microsoft® Internet Explorer for proper operation of the Remote Monitoring function.

#### Procedure 15-1 Configuring Internet Explorer

#### Conditions

Set Internet Explorer to prevent Windows from blocking communication with the robot controller. The procedures are almost the same for Window 7, Windows Vista and Windows XP, so screenshots of Windows 7 are used in the explanation below.

**1.** In the Control Panel window, open [Internet Options]. Internet Explorer ®.

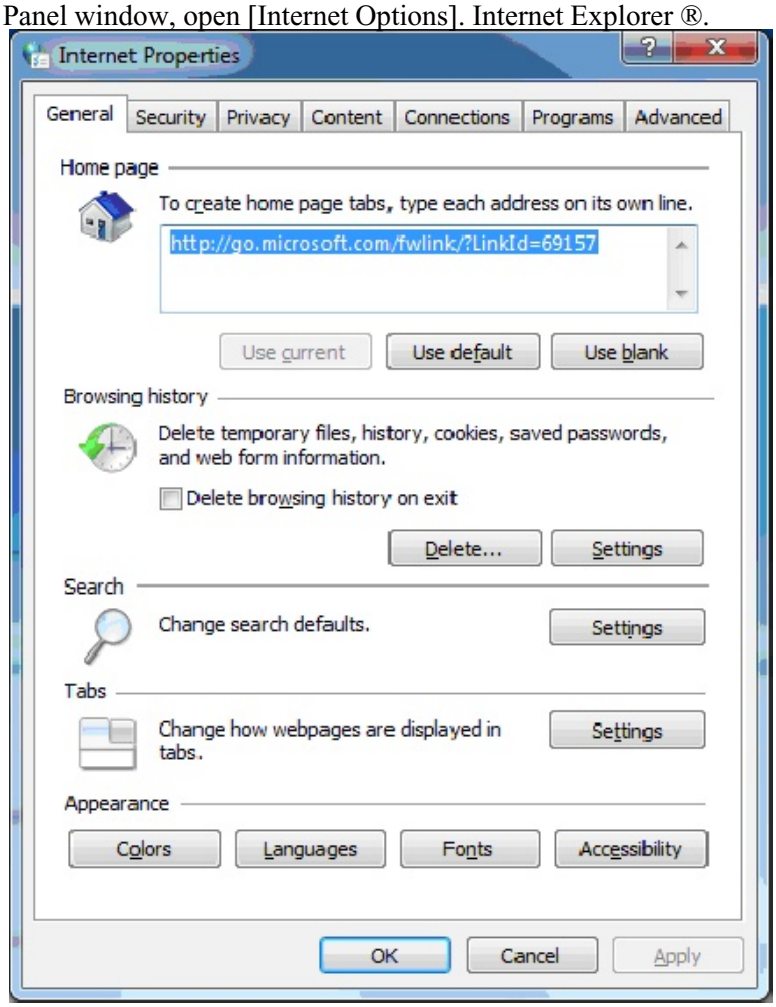



Fig. 15.3.2.2(a) Internet Properties

**2. Trusted Sites:**

**a.** Select the [Security] tab.

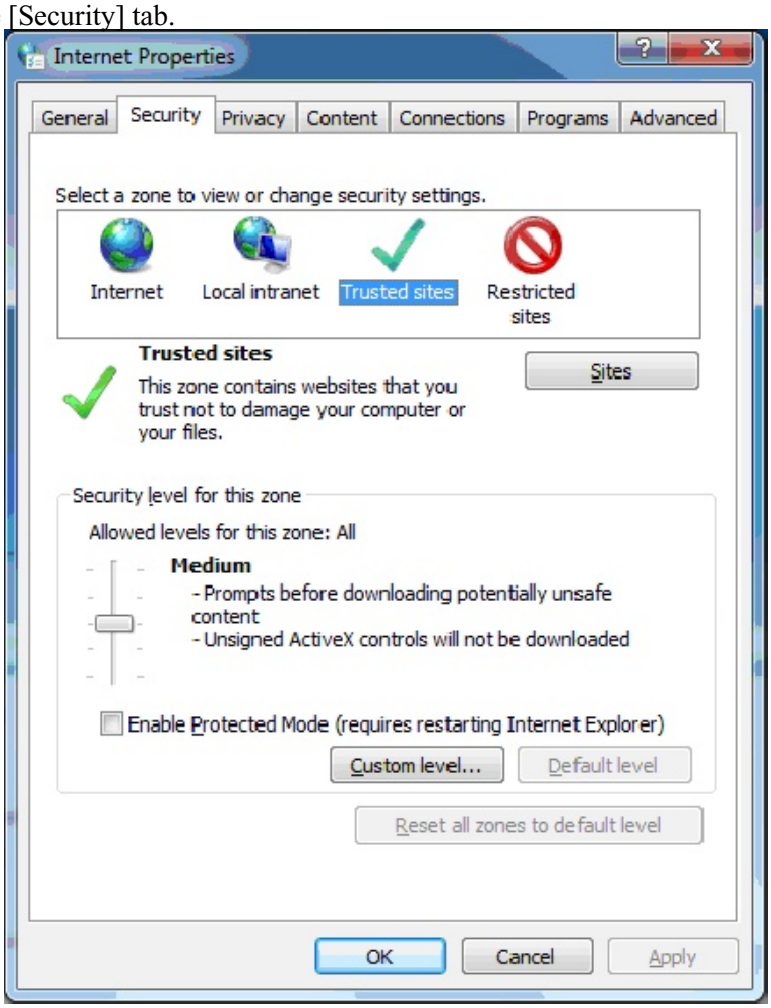



Fig. 15.3.2.2(b) Security

**b.** Select [Trusted Site], and then click the [Sites] button.

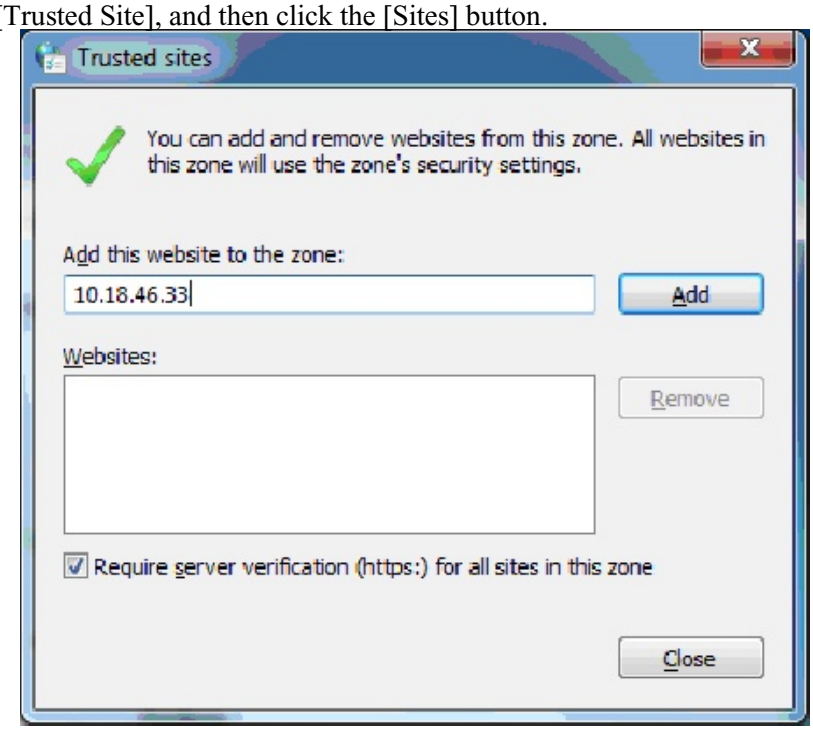



Fig. 15.3.2.2(c) Trusted Sites

**c.** Uncheck the [Require server verification (https:) for all the sites in this zone] box.
**d.** In the [Add this Web site to the zone] textbox, enter the IP address of the robot controller (or the last digit of the IP address can be replaced by *). Then, click the [Add] button.
**e.** Click the [Close] button to close the dialog box.

**3. Popup Blockers**

**a.** Select the [Privacy] tab.

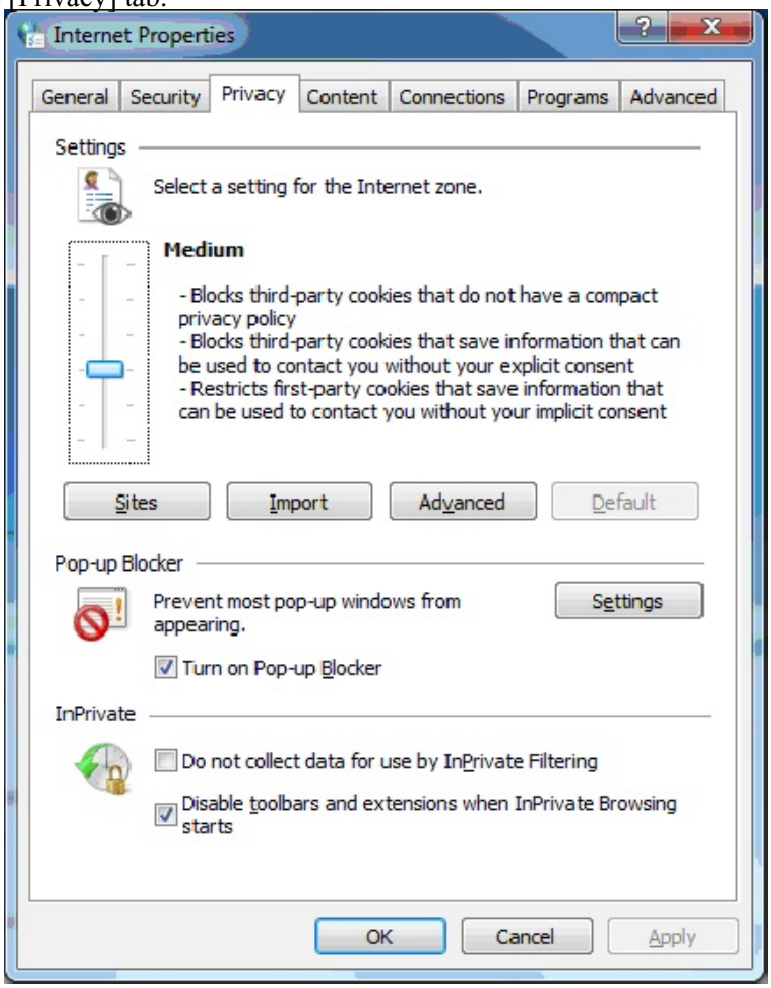



Fig. 15.3.2.2(d) Privacy

**b.** Click the [Settings] button of [Pop-up Blocker].

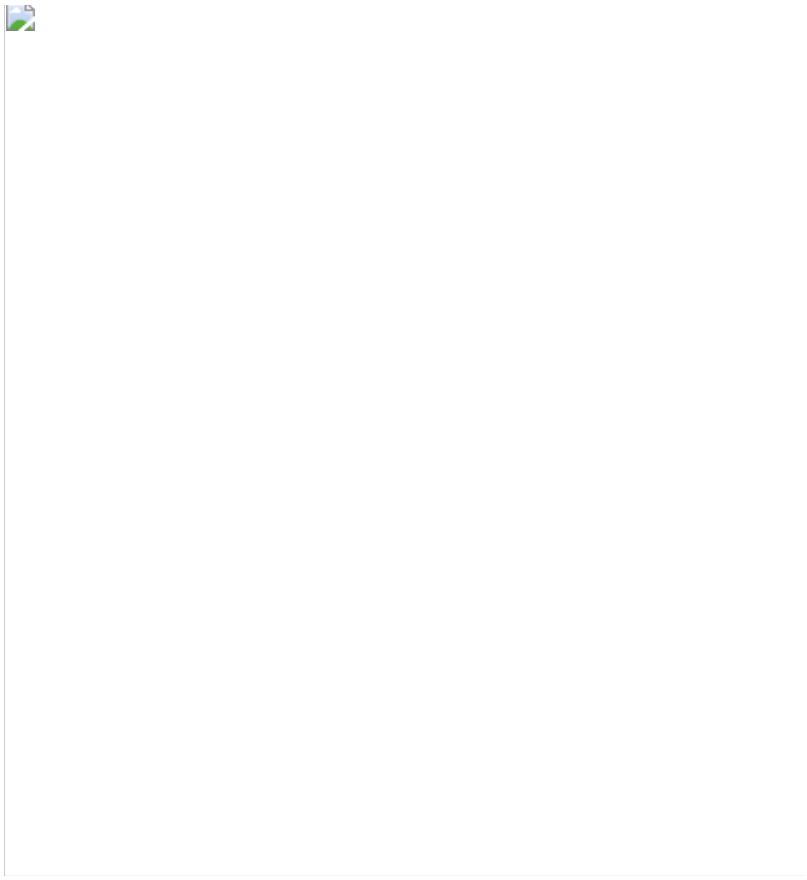


Fig. 15.3.2.2(e) Pop-up Blocker Settings

**c.** Enter the IP address of the robot controller in the [Address of Web site to allow] textbox, and click the [Add] button.
**d.** Click the [Close] button to close the dialog box.

**4. Proxy Setting**

**a.** Select the [Connections] tab.

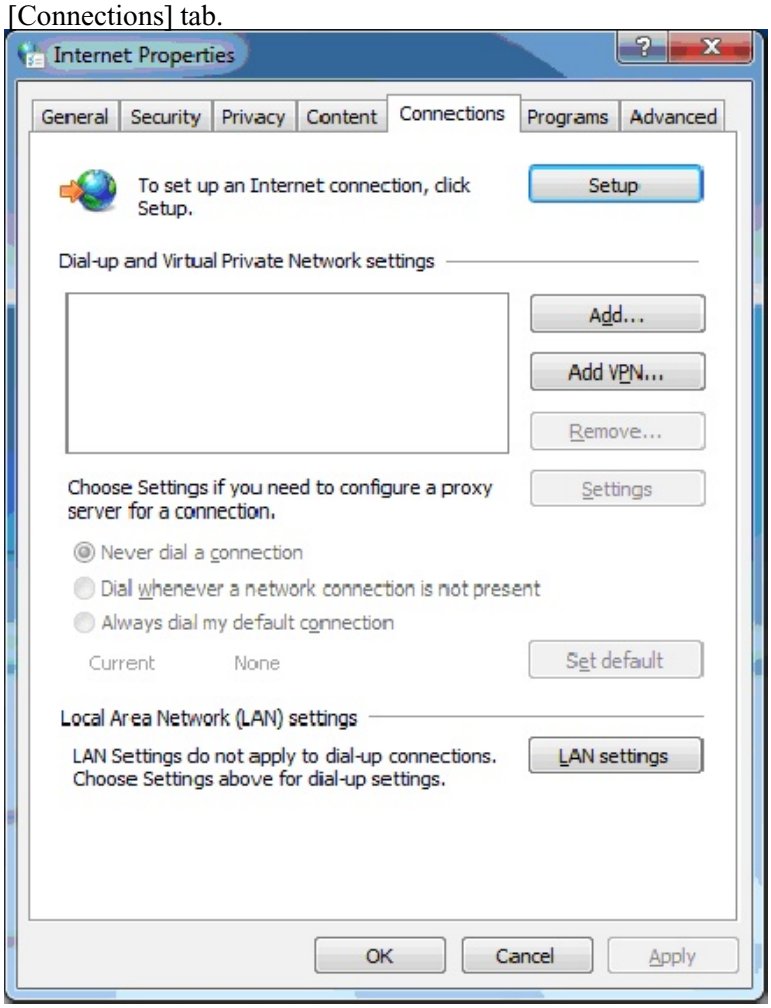



Fig. 15.3.2.2(f) Internet Properties

**b.** Click the [LAN Settings] button.

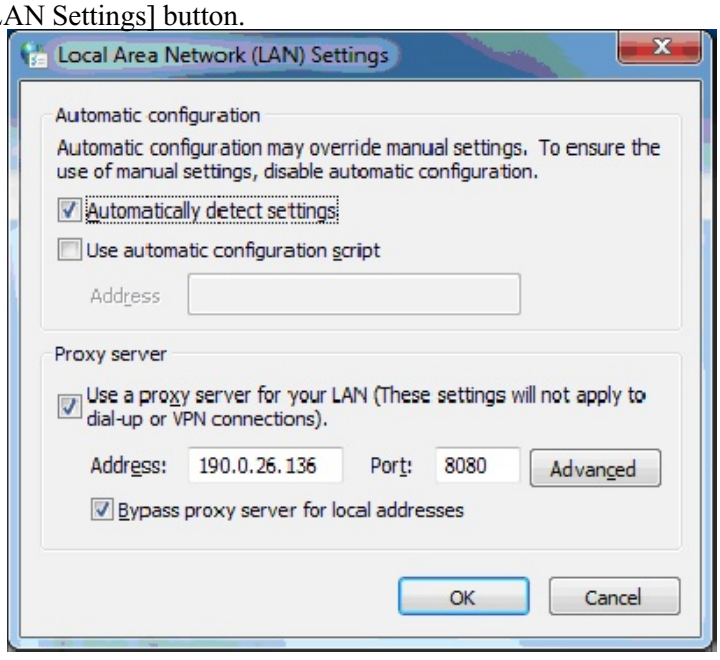



Fig. 15.3.2.2(g) Local Area Network (LAN) Settings

**c.** When the [Use a proxy server for your LAN] check box is not checked, proceed to Step 4.g . When it is checked, perform Step 4.d through Step 4.f .

**d.** Click the [Advanced…] button of [Proxy server].

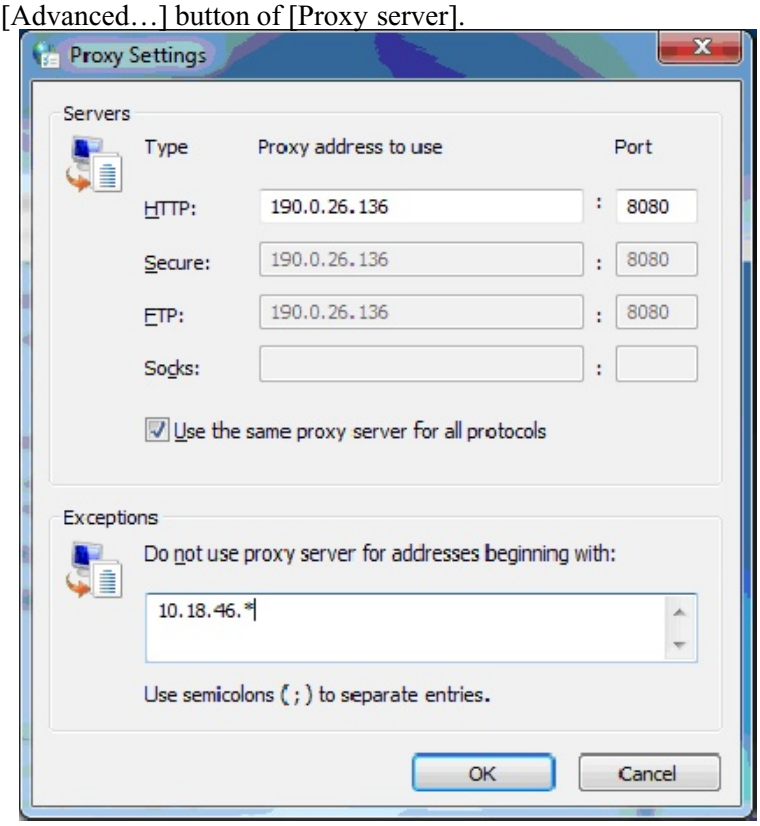



Fig. 15.3.2.2(h) Proxy Settings

**e.** Enter the IP address of the robot controller in the text box under [Exceptions].

**f.** Click the [Close] button to close the dialog box.

**g.** Click the [OK] button to close the Internet property page.

**5.** **Modifying Setting of Windows Firewall** Modify the settings of Windows Firewall to prevent Windows Firewall from blocking communication with the robot controller.

**a.** In Windows 7

**i.** In the Control Panel window, open [Windows Firewall].

Fig. 15.3.2.2(i) Windows Firewall

**ii.** Click [Allow a program or feature through Windows Firewall].

Fig. 15.3.2(j) Allowed Programs

**iii.** Click the [Change settings] button.

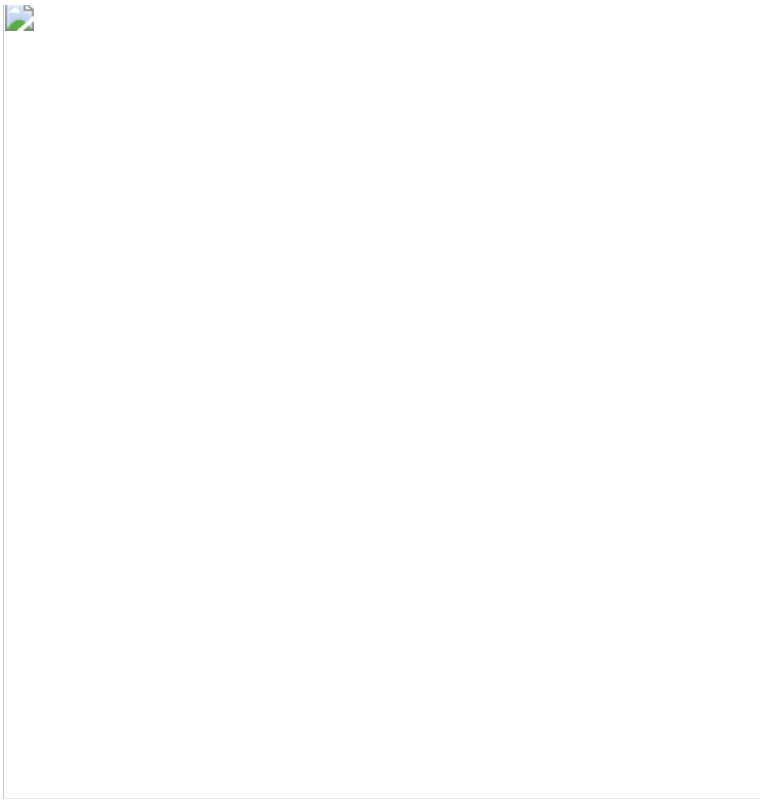


Fig. 15.3.2.2(k) Add a Program

**iv.** Select [Internet Explorer] in the list, and click the [Add] button.

**v.** Click the [OK] button to close the window.

**b.** In Windows XP or Windows Vista,

**i.** In the Control Panel window, open [Windows Firewall].

**ii.** Click the [Exceptions] tab.

Fig. 15.3.2.2(l) Exceptions

**iii.** Click the [Add Program] button.

Fig. 15.3.2.2(m) Add a Program

**iv.** Select [Internet Explorer] from the list, then click the [OK] button.

Fig. 15.3.2.2(n) Exceptions

**v.** Click the [OK] button.

### 15.3.2.3 Testing the Network Connection

This section describes a method to verify that the network connection from the PC to the robot is configured correctly and operational.

#### Procedure 15-2 Testing the Network Connection

#### Conditions

Before performing the test, make sure the following conditions are met:

- The PC is connected to a network that can be used to access the robot.
- The PC has Microsoft® Internet Explorer 5.5 or greater loaded and is properly configured as detailed in Section 15.3.2.2 above.
- The robot is turned on and is connected to a network that is accessible by the above PC.

#### Steps

1. Bring up Internet Explorer on the PC.
2. In the Internet Explorer Address field, Enter “http://<myrobot_name_ or_address>”.
   Where <myrobot_name_ or_address> is either the DNS name of your robot i.e. pderob111.frc.com) or the IP address of your robot. (for example 192.168.1.100)

If the connection is successfully made you will see the HOME page of the robot displayed in Internet Explorer. It will be similar to that shown in Figure 15.3.2 .

Fig. 15.3.2.3 Robot HOME page

If you are unable to make this connection, refer to the SETTING UP TCP/IP section in this manual or contact your System Network Administrator.

## 15.3.3 Operation

After you have properly configured Microsoft® Internet Explorer and verified that you can connect to the robot as detailed above you can now access the Remote *i*Pendant screen. The following sections detail the procedure required for this and the limitations of this feature.

### 15.3.3.1 Remotely monitoring the *i*Pendant

This section will describe the method to connect to the robot controller and display the remote *i*Pendant screen for monitoring the *i*Pendant operation.

#### Procedure 15-3 Remotely Monitoring the *i*Pendant

#### Conditions

- The PC is connected to a network that can be used to access the robot.
- The PC has Microsoft® Internet Explorer 5.5 or greater loaded and is properly configured as detailed in Section 15.3.2.2 above.
- The PC has the *i*Pendant Controls installed as detailed in Section 15.2 .
- The robot is turned on and connected to a network that is accessible by the above PC.
- The robot has a functional *i*Pendant connected and operational.

#### Steps

1. Bring up Internet Explorer on the PC.
2. In the Internet Explorer Address field, Enter “http://<myrobot_name_ or_address>” to access the robot HOME page.
   Where <myrobot_name_ or_address> is either the DNS name of your robot (i.e. pderob111.frc.com) or the IP address of your robot. (i.e. 192.168.1.100)

**3.** From the HOME page, select Monitor *i*Pendant (ECHO).
If the connection is successfully made you will briefly see the LOGIN screen, similar to that shown in Figure 15.3.3(a) , displayed in Internet Explorer.

Fig. 15.3.3.1(a) Remote *i*Pendant LOGON Screen

After the LOGON Screen, you will see a remote display of the current *i*Pendant screen. The display will show the current *i*Pendant screen and any activity that might be occurring on it (popup menus, dynamic data, screen reconfigurations, for example). It might be similar to that shown in Figure 15.3.3(b) .

Fig. 15.3.3.1(b) Remote *i*Pendant Monitoring Screen

#### 15.3.3.2 Troubleshooting Remote Connection

Tips for troubleshooting when the remote *i*Pendant connection does not work.

- Try adding your robot controller IP address as a trusted site in Internet Explorer. Then connect using the IP address instead of the hostname.
- If there is a firewall between the robot controller and the PC, the robot controller must be able to open a TCP connection to port 60005 on the PC. The PC must also be able to open a TCP connection to port 3002 on the robot controller. This is in addition to allowing the basic HTTP request from the PC to the robot controller (TCP connection to port 80 on the robot).
- If Skype IE plugin is installed, please uninstall it.
- Communication with the robot controller might be prevented due to a cause other than the above, which is, for example, a Microsoft Internet Explorer add-on or security software installed in your PC. Try disabling add-on software of Internet Explorer.

### 15.3.4 Limitations

Remote monitoring provides the user with the capability to display and monitor the current *i*Pendant screens and operations on a PC using Microsoft® Internet Explorer.
The following limitations apply to the remotely displayed *i*Pendant screen during the remote monitoring operation:

- It is meant as a DISPLAY ONLY mode, as such the remote connection normally cannot interact with the screens or affect the operation of the *i*Pendant or robot controller except in the following cases:

  WARNING
  If any web page is being displayed, any active link or component (ie *i*Pendant Control) can be selected on the remote *i*Pendant screen and activated. This may cause an interaction with the robot or with the actual *i*Pendant. Care must be taken when viewing web pages remotely.

- The remote display will look similar to the *i*Pendant but will not be an exact duplicate. Fonts, Character sizes, colors, images, and overall format might be different.
- If a popup menu is being displayed on the *i*Pendant when the remote monitoring connection is made to the robot controller, that popup menu will not appear on the initial remote display. Subsequent pop-ups will be displayed on the remote screen as they occur on the *i*Pendant.

## 15.4 REMOTE OPERATION

### 15.4.1 Overview

Remote Operation allows you to display an *i*Pendant-like screen on a PC using Microsoft ® Internet Explorer. The remote user can configure the remote display, navigate the controller menus and screens, and enter data remotely. Remote Operations are only available if the Internet Connectivity and Connection option is loaded.

WARNING
Remote Operation is completely independent from the operation of the actual *i*Pendant therefore it can affect the operation the robot controller. Extreme caution must be exercised when using this feature.

NOTE

The remote operator cannot see what is currently on the actual *i*Pendant or what the operator is doing nor can the *i*Pendant operator see what the remote operator is doing or what screens are being displayed on the remote PC.

For information on the limitations of this feature, refer to Section 15.4.4 .
The following functions can be performed on the remote *i*Pendant operation display:

- All *i*Pendant screens available from the MENUS and/or [TYPE] keys can be displayed as well as any custom screens and HELP or DIAGNOSTIC Screens.
- All popup menus, and windows.
- Multiple window configurations (i.e. Double and Triple modes on the *i*Pendant).
- Any input from the *i*Pendant numeric keypad, Function keys or cursor movement.

## 15.4.2 Setup

### 15.4.2.1 Requirements

The following are the requirements for remote operation of the *i*Pendant screens.

- PC must have Microsoft® Internet Explorer 5.5 or greater installed.
- PC must have the *i*Pendant Controls installed. Refer to Section 15.2 for installation instructions.
- PC must be connected to a network , and be properly configured to allow a TCP/IP connection to the Robot Controller with the *i*Pendant connected.
- The robot controller must be connected to a network and be properly configured for Network access to the above PC.
- The robot controller must have the Internet Protocol Connectivity and Customization (IPCC) Option installed.
- The HTTP Authentication must be properly configured on the robot to allow *i*Pendant access. Refer to the HTTP Authentication section in this manual for configuration information.

### 15.4.2.2 Configuring Microsoft® Internet Explorer

See Section 15.3.2.2 for information on configuring Microsoft® Internet Explorer.

## 15.4.3 Remote *i*Pendant Operation

After you have properly configured Internet Explorer ® and verified that you can connect to the robot, you can now display the remote *i*Pendant Operation screen. The following sections detail the operation and the limitations of this feature.

### 15.4.3.1 Connecting to the controller

This section will describe the method to connect to the robot controller and display the remote *i*Pendant Operation screen.

**Procedure 15-4 Remote *i*Pendant Connection**

**Steps**

1. Bring up Internet Explorer on the PC.
2. In the Internet Explorer Address field, type the following: http://<myrobot_name_ or_address> to view the robot HOME page.
   Where <myrobot_name_ or_address> is either the DNS name of your robot (i.e. pderob111.frc.com) or the IP address of your robot (i.e. 192.168.1.100).

3. From the HOME page, select Navigate *i*Pendant (CGTP). If it greyed out, then you do not have the Internet Protocol Connectivity and Customization (IPCC) option installed.
4. If you have the HTTP Authentication for the *i*Pendant set to AUTHORIZE, the prompt shown in Figure 15.4.3.1(a) will be displayed on the PC.

Fig. 15.4.3.1(a) Remote *i*Pendant Operation Password Screen

Type the USERNAME and PASSWORD that you set on the Robot Controller in the HOSTCOMM Setup>HTTP Authentication for the *i*Pendant.

If the connection is successfully made you will briefly see the LOGIN screen, similar to that shown in Figure 15.4.3.1(b) , displayed in Internet Explorer.

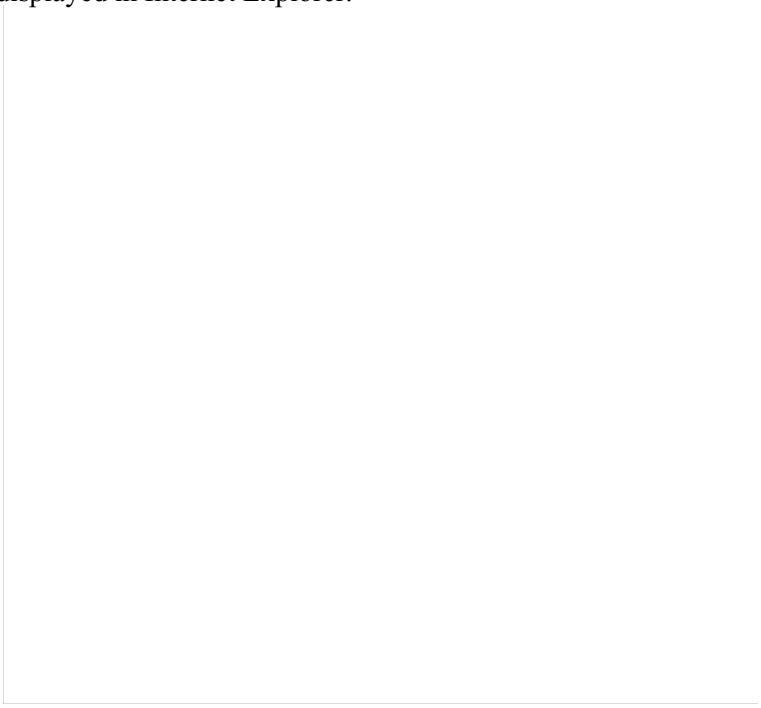


Fig. 15.4.3.1(b) Remote *i*Pendant LOGON Screen

If this is the first time you are logging in, you will see an *i*Pendant screen similar to the one shown in Figure 15.4.3.1(c) . If you have logged in before you might see a different screen configuration, depending on how your controller was set up.

Fig. 15.4.3.1(c) Remote *i*Pendant Operation Screen

If the remote *i*Pendant connection failed, refer to Section 15.3.3.2 for troubleshooting tips.

### 15.4.3.2 Keys

While in the Remote Operation Mode you can use your mouse to select any of the available function keys or the auxiliary keys available below the *i*Pendant screen.

These auxiliary keys allow you access to certain functions available as Hard Keys on the actual *i*Pendant. Some of these keys are also mapped to keys on the PC keyboard as shown in Table 15.4.3.2(a) below:

Table 15.4.3.2(a) Remote Operation Key Mapping

| *i*Pendant Function Key | Description | PC Key Mapping |
|---|---|---|
| PREV | This key moves to Previous Screen or Cancels a pending operation | ESC |
| MENU | This key activates the MENU popup | F8 |
| TOP | This key activates the TOP menu | |
| WND | This key changes the active window | |
| DISP | This key activates the DISPLAY Menu | |
| HELP | This key activates HELP for the current screen | |
| DIAG | This key activates DIAGNOSTICS for the current Alarm | |
| ITEM | Allows you to select an item on the screen by entering an Item Number | |
| FCTN | Activates the FUNCTION Popup Menu | F12 |
| VIEW | Activates the Related View menu if *i* is displayed on the Focus Bar | |
| NEXT | Moves to the next page of softkeys if they are available. | F6 |
| | Enters the SELECT menu | F9 |
| | Enters the EDIT menu | F10 |
| | Enters the DATA menu | F11 |
| | Move the cursor in the appropriate direction | UP, DOWN, LEFT and RIGHT ARROWS |
| | Page Up and Page Down | SHIFT+UP Arrow, SHIFT+DOWN Arrow, PAGE UP and PAGE DOWN |

In addition mouse events are supported in most screens as shown in Table 15.4.3.2(b) below:

Table 15.4.3.2(b) Remote Operation Mouse Events

| Mouse Event | Description |
|---|---|
| Page Up | Mouse click above the lines in any screen. |
| Page Down | Mouse click below the lines in any screen. |

| Mouse Event | Description |
|---|---|
| Page Up and Down | Mouse movement while left button is held. Typically screens will scroll up or down. If you move fast, the automatic scroll feature will be enabled. You can now lift your mouse and the screen will continue to scroll until it reaches the top or bottom of the screen. You can stop the scroll at any time by clicking the screen. If your screen is in zoom mode, you can scroll left or right. |
| Select a line | Mouse click on a line. |
| Select an item | Mouse click on an item. |
| Select Links, or custom controls on Web pages, Help, Diagnostics, and any custom *i*Pendant screen. | Use Mouse click to select. |
| Select a program | Double mouse click on the program. |
| When ENTER is a valid selection. | Double mouse click on the item. |
| When CHOICE is a valid selection. | Right mouse click on the item. |
| Activate [EDCMD] in the TP Editor. | Right mouse click on or to the left of the line number. |
| Select an item in 4D display | Right mouse click on an item. |
| When PREV is a valid selection. | Mouse click to the left of the function keys. |
| When NEXT is a valid selection. | Mouse click on > to the right of the function keys. |
| To display DETAIL for an alarm. | Right mouse click on an alarm in the Alarm Log screen. |
| MENU | Mouse click on the left side of the Focus Bar |
| DISPLAY Menu | Mouse click on the center of the Focus Bar |
| FCTN | Mouse click on the right side of the Focus Bar |
| Related Views | Mouse click on the i shown in the Focus Bar |
| Maximize/Restore | Mouse click on the Maximize/Restore icon shown in the icon menu |
| Zoom | Mouse click on the ^ shown in the Focus Bar |

For further information on navigating *i*Pendant screens and operating the *i*Pendant, refer to Section 15.1

## 15.4.3.3 Editing guidelines

- LOOK/MONITOR mode is available in any window.
- Each window can have a unique default program.
- Selecting a teach pendant program from the SELECT screen in the window will cause that program to be the default program for that window.
- The window will use $UI_DEFPROG[3] to $UI_DEFPROG[8] based on the order of connection.
- The Status line will always show the default program for the left-hand window even if it does not have focus.
- The Editor title line shows the program that is being displayed.
- The program selected will be retained during cycle power. The current line number will not be retained during cycle power.
- The same program can be displayed in multiple windows. The cursor is independent.
- The Editor will not allow editing in the foreground. If you try to make a change to the program, then the warning “TPIF-069 Use background edit” is posted
- The program is open while in the EDIT screen; otherwise the program will be closed.

- Closing a window will cause the selected program to be closed.
- Disconnecting the Internet Explorer session will cause all the programs in all windows to be closed.
- FWD/BWD is not available from Internet Explorer

#### 15.4.3.4 Background editing guidelines

- Background edit is supported in each window. There will be a background edit program unique for each window so multiple edit sessions will not affect each other.
- The background edit programs will be –BCKED4- through –BCKED9- based on the order of connection.
- On Internet Explorer, you have no control over what connection you get. However, if you time out and reconnect, the connection should be the same. If you edit at your desk and stay logged in, then go to the lab you will get a different connection and you will not be able to continue with the same background edit program while in the lab.
- When a program is selected for background edits, then the comment for the background edit program will show the selected program.
- The same program cannot be background edited in multiple windows. If you select the same program already being background edited in another window, then the warning " TPIF-167 Program already in background edit" is posted. You need to End_edit the background session in the other window first.
- Background edit programs are retained during cycle power.
- The three windows within the same connection will share the same copy and paste buffer but they will not share with other Internet Explorer connections. This provides the ability to copy and paste from one teach pendant program to another.
- Each window has its own undo and redo buffers so multiple edit sessions will not affect each other.

### 15.4.4 Limitations

Remote operation allows you to display and monitor any *i*Pendant screen and perform many *i*Pendant operations on a PC using Internet Explorer ® .
The following limitations apply to the Remote Operation of the *i*Pendant screen:

- You cannot make any changes through this connection if read only is enabled. To allow changes, set $UI_CONFIG.$READONLY[2]=FALSE.
- Many operations that are available on the actual *i*Pendant are not allowed through the remote monitoring function. These include:
  - Jogging the Robot
  - Running a Program using the FWD/BWD Keys
  - Changing COORD
  - Changing Speed Override
  - Foreground editing a program (Only background editing is allowed)
  - Functions associated with the application-specific keys on the *i*Pendant such as WIRE+/-, BACKUP, and so forth.

- Several of the *i*Pendant screens can only be displayed by a single device at any given time. The actual *i*Pendant always has precedence. If the *i*Pendant is currently displaying a screen that the Remote Operator has requested, and that screen can only be displayed in one place, the error "TPIF 110– Screen used by other device", will be posted and the screen on the remote device will not be changed. Also, if the remote device is currently displaying a screen that is requested by the actual *i*Pendant operator, the remote display will be forced to the UTILITIES>HINTS screen. However, an error will not be posted.
- The remote display will look similar to the *i*Pendant but will not be an exact duplicate. Fonts, Character sizes, colors, images, and overall format might be different.

# APPENDIX

**2.** Type the following command, replacing the IP address with the IP address you want to PING, and press ENTER.

```
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:\>
```

The following message shows an unsuccessful PING.

```
C:\>ping 172.22.200.246

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:\>
```

If the LINK LED is on but a PING request fails it usually indicates a problem with IP address configuration. Either no IP address is configured, or the combination of IP address and subnet mask is inconsistent for the network. Refer to the “Setting Up TCP/IP” section in the *Internet Options Setup and Operations Manual* for details on configuring the IP address and subnet mask for the robot.

# A.2 ETHERNET LEDS

The Ethernet LEDs are located on the Main board. See Figure A.2 .

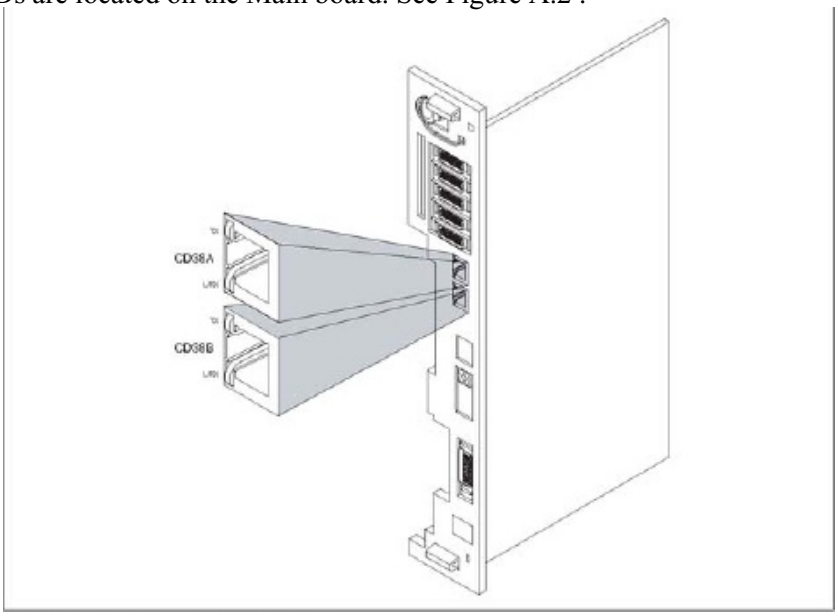



Fig. A.2 Ethernet LEDs

| Description | Label | Color |
|---|---|---|
| Transmit | TX | Green |
| Link/Receive | RX | Green |

This section describes the status of the Ethernet LEDs on the Main board. Refer to Table A.2 for a description of the status LEDs.

Table A.2 LED Status Description

| LED | On | Flashing | Off | Color |
|---|---|---|---|---|
| Link Status | The Ethernet cable connection is plugged into the robot and the link is functional. | The Ethernet interface is receiving packets. | The Ethernet connection is not plugged into the robot or there is a problem on the link (a cable is missing, for example) | Green |
| TX Transmit Status | Indicates that transmit activity has been detected at the Ethernet. | Indicates continuous transmit activities at the selected port. | Indicates no transmit activity at the selected port. | Green |

# A.3 10 BASE-T/100 Base T-X CONNECTOR PIN ASSIGNMENTS

This section contains information about pin assignments for the 10 Base-T/100 Base-TX. See Figure A.3 for 10 Base-T/100 Base-TX connector assignment.

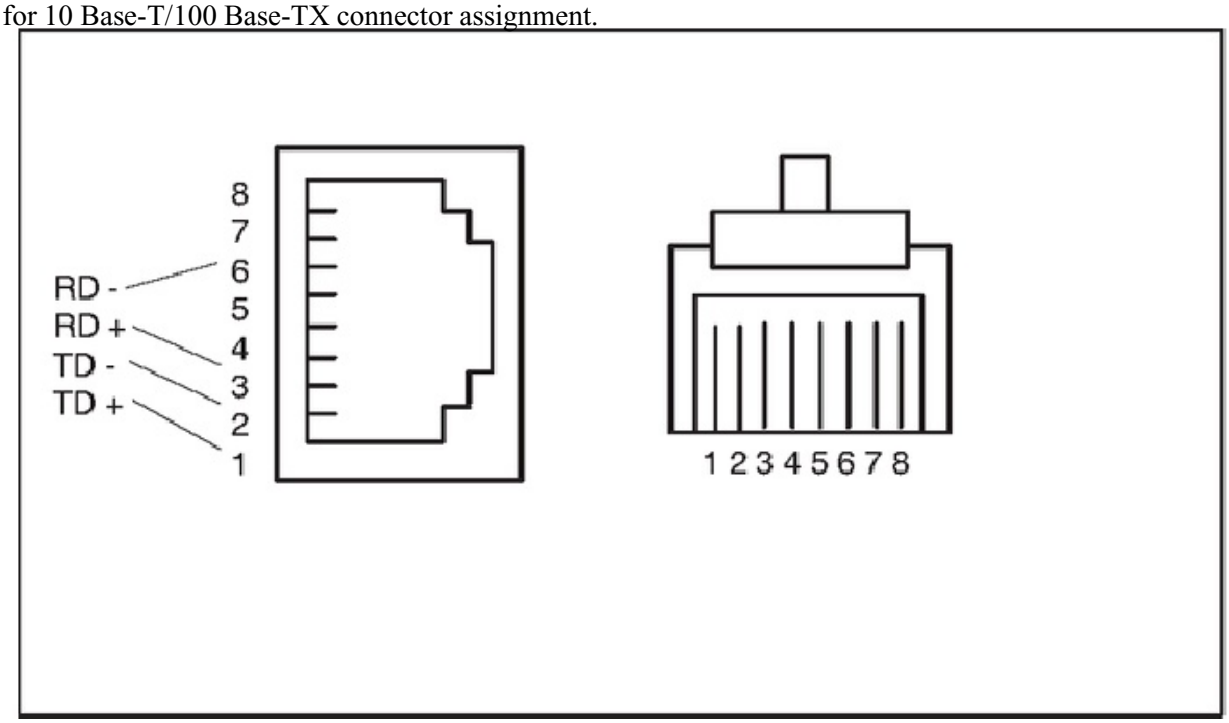



Fig. A.3 10 Base-T/100 Base-TX Connector Pin Assignments

# B CONFIGURE FTP WITH A KAREL COMMAND FILE

## B.1 CONFIGURING NETWORK PARAMETERS WITH A KAREL COMMAND FILE

You can use a command file to set up Ethernet TCP/IP Parameters. The example command file can be run from the KAREL command line with the RUNCF command. You must turn the controller off, and then back on for the settings to take effect. Refer to Table B.1 for an example NETSETUP.CF file.

Table B.1 Example NETSETUP.CF File (for interface #1)

| INSTRUCTION | DESCRIPTION |
|---|---|
| Name and IP address | |
| set var $hostname = 'ROBOT1' | |
| set var $hostent[17].$h_name = 'QUICCO' | Entry 17 in the local host table is reserved for interface #1 on the robot. This string is preset and should not be changed |
| set var $hostent[17].$h_addr = '192.168.0.2' | This is the IP address associated with interface #1 on the robot. |
| set var $tmi_router = 'ROUTER' | |
| set var $hostent[20].$h_name = 'ROUTER' | |
| set var $hostent[20].$h_addr = '192.168.0.1' | Router name and IP address (if router is used) Entry 20 in the local host table is reserved for the router. NOTE that the router IP address should be on the same subnet as interface #1 or #2. There is a single default router for the robot. |
| set var $tmi_snmask[1] = '255.255.255.0' | Robot Subnet Mask (based on network IP address, Class C mask shown) for interface #1. |
| set var $hostent[1].$h_name = 'PC_HOST' | |
| set var $hostent[1].$h_addr = '192.168.0.3' | Additional Entries might be needed in the local robot HOST table to identify remote FTP servers referenced by FTP clients on robot.-- The local HOST table has up to 16 entries. Entries 17–20 are reserved. |
| set var $host_shared[1].$h_name = 'UNIX_HOST' | |
| set var $host_shared[1].$h_addr = '192.168.0.4' | Additional Entries might be needed in sharedrobot HOST table to identify remote FTP servers referenced by FTP clients on robot.The shared HOST table has up to 20 entries and is held in SYSHOST.SV so it can be shared between robots which might share common FTP servers. |

| INSTRUCTION | DESCRIPTION |
|---|---|
| set var $hosts_Cfg[1].$protocol ='FTP' | Up to 8 FTP servers can be configured on the robot. You need to start enough FTP servers to handle the maximum number of simultaneous FTP connections to the robot. Two are started by default. |
| set var $hosts_Cfg[1].$port = '' | |
| set var $hosts_Cfg[1].$oper = 3 | Configure to be STARTED when you turn the controller ON. |
| set var $hosts_Cfg[2].$oper = 3 | |
| set var $hosts_Cfg[2].$protocol = 'FTP' | |
| set var $hosts_Cfg[2].$port = '' | |
| $hosts_Cfg[3].$protocol = 'FTP' | |
| set var $hosts_Cfg[3].$port = '' | |
| set var $hosts_Cfg[3].$oper = 3 | |
| set var $hosts_Cfg[4].$protocol = 'FTP' | |
| set var $hosts_Cfg[4].$port = '' | |
| set var $hosts_Cfg[4].$oper = 3 | |
| set var $hosts_Cfg[5].$protocol = 'FTP' | |
| set var $hosts_Cfg[5].$port = '' | |
| set var $hosts_Cfg[5].$oper = 3 | |
| set var $hosts_Cfg[6].$protocol = 'FTP' | |
| set var $hosts_Cfg[6].$port = '' | |
| set var $hosts_Cfg[6].$oper = 3 | |
| set var $hosts_Cfg[7].$protocol = 'FTP' | |
| set var $hosts_Cfg[7].$port = '' | |
| set var $hosts_Cfg[7].$oper = 3 | |
| set var $hosts_Cfg[8].$protocol = 'FTP' | |
| set var $hosts_Cfg[8].$port = '' | |
| set var $hosts_Cfg[8].$oper = 3 | |
| set var $hostc_Cfg[1].$protocol = 'FTP' | |
| set var $hostc_Cfg[1].$port = '' | |

| INSTRUCTION | DESCRIPTION |
| --- | --- |
| set var $hosts_Cfg[1].$oper = 2 | |
| set var $hostc_Cfg[1].$strt_path = './testing/ftp/' | Up to 8 FTP clients can be configured on the robot. |
| set var $hostc_Cfg[1].$strt_remote = 'UNIX_HOST' | |
| set var $hostc_Cfg[2].$protocol = 'FTP' | |
| set var $hostc_Cfg[2].$port = '' | |
| set var $hostc_Cfg[2].$oper = 2 | Configure to be DEFINED when you turn the controller ON. |
| set var $hostc_Cfg[2].$strt_path = 'C:¥TEMP¥' | |
| set var $hostc_Cfg[2].$strt_remote = 'PC_HOST' | |
| set var $hostc_Cfg[3].$protocol = 'FTP' | |
| set var $hostc_Cfg[3].$port = '' | |
| set var $hostc_Cfg[3].$oper = 2 | |
| set var $hostc_Cfg[3].$strt_path = './testing/ftp/' | |
| set var $hostc_Cfg[3].$strt_remote = 'UNIX_HOST' | |
| set var $hostc_Cfg[4].$protocol = 'FTP' | |
| set var $hostc_Cfg[4].$port = '' | |
| set var $hostc_Cfg[4].$oper = 2 | |
| set var $hostc_Cfg[4].$strt_path = './testing/ftp/' | |
| set var $hostc_Cfg[4].$strt_remote = 'UNIX_HOST' | |
| set var $hostc_Cfg[5].$protocol = 'FTP' | |
| set var $hostc_Cfg[5].$port = '' | |
| set var $hostc_Cfg[5].$oper = 2 | |
| set var $hostc_Cfg[5].$strt_path = './testing/ftp/' | |
| set var $hostc_Cfg[5].$strt_remote = 'UNIX_HOST' | |
| set var $hostc_Cfg[6].$protocol = 'FTP' | |
| set var $hostc_Cfg[6].$port = '' | |
| set var $hostc_Cfg[6].$oper = 2 | |
| set var $hostc_Cfg[6].$strt_path = './testing/ftp/' | |

| INSTRUCTION | DESCRIPTION |
|---|---|
| set var $hostc_Cfg[6].$strt_remote = 'UNIX_HOST' | |
| set var $hostc_Cfg[7].$protocol = 'FTP' | |
| set var $hostc_Cfg[7].$port = '' | |
| set var $hostc_Cfg[7].$oper = 2 | |
| set var $hostc_Cfg[7].$strt_path = './testing/ftp/' | |
| set var $hostc_Cfg[7].$strt_remote = 'UNIX_HOST' | |
| set var $hostc_Cfg[8].$protocol = 'FTP' | |
| set var $hostc_Cfg[8].$port = '' | |
| set var $hostc_Cfg[8].$oper = 2 | |
| set var $hostc_Cfg[8].$strt_path = './testing/ftp/' | |
| set var $hostc_Cfg[8].$strt_remote = 'UNIX_HOST' | |

# C NETWORK DESIGN AND PERFORMANCE

## C.1 GUIDELINES FOR USING ETHERNET

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 autonegotiate 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. However be very careful that both ends of the connection use the same speed and duplex mode. The LEDs near the RJ45 connector on the robot will confirm connection 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 effected 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, EtherNet IP nodes) by using a separate subnet for the control equipment or by using VLANs (virtual LANs) supported by some higher end switches. If the EtherNet I/P network is completely isolated as a separate control network this is not a concern. However, when connecting into larger networks this becomes important.
Some network environments have a significant amount of multicast traffic. A basic layer 2 switch will treat multicast traffic like broadcast traffic and forward to all ports in the switch wasting network bandwidth and node resources on traffic, which is ultimately dropped for the nodes that are not interested in the multicast traffic. Switches that support “IGMP snooping” will selectively send multicast traffic only to the nodes, which have joined a particular group. EtherNet/IP UDP packet has a TTL (time to link) value of one. You will not be able to route I/O traffic across more than one switch.
Quality of Service (QOS) techniques provide mechanisms to prioritize network traffic. Generally on an Ethernet network all packets are equal. Packets can be dropped or delayed within network infrastructure

# CABLE CONNECTION

This section describes information relating to the physical Ethernet connection.

CAUTION

1 Before connecting or disconnecting the cable to or from the robot controller, make sure that the power to the controller is turned off.
2 Please inquire of each manufacturer about the construction of network or the condition of using the equipment except the robot controller (hub, transceiver, cable etc.). When configuring your network, you must take other sources of electrical noise into consideration to prevent your network from being influenced by electrical noise. Make sure that network wiring is sufficiently separated from power lines and other sources of electrical noise such as motors, and ground each of the devices as necessary. Also, a high and insufficient ground impedance may cause interference during communications. After installing the machine, conduct a communications test before you actually start operating the machine.
We cannot ensure operation that is influenced by network trouble caused by a device other than the robot controller.

## D.1 CONNECTING TO Ethernet

R-30*i*A, R-30*i*A Mate, R-30iB is provided with a 100BASE-TX interface.
Prepare a hub for connecting the FAST Ethernet board to the Ethernet trunk. The following shows an example of a general connection.

Some devices (hub, transceiver, etc.) that are needed for building a network do not come in a dust-proof construction. Using such devices in an atmosphere where they are subjected to dust or oil mist will interfere with communications or damage the FAST Ethernet or FAST Data Server. Be sure to install such devices in a dust-proof cabinet.

HUB

:
:
:

Twisted pair
cable

Robot Controller

FAST
Ethernet or
FAST Data
Server

Max. 100m

(Except for the cable recommended by FANUC)

# D.2 LEADING OUT THE Ethernet CABLE

For this type of control unit, the Ethernet cable is drawn out only from the front of the control unit. See the outline drawing of each type of board for the location of the connector.

The radius of the cable must be 70 mm or more.

Twisted-pair cable

Clamp

Ground plate

The Ethernet cable must be fastened by a cable clamp to prevent tension being applied to the modular connector (RJ-45) that connects the cable to the control unit even if the Ethernet cable is pulled directly. This clamp is also used to ground the cable shield.

# D.3 100BASE-TX CONNECTOR (CD38A/CD38B) PIN ASSIGNMENTS

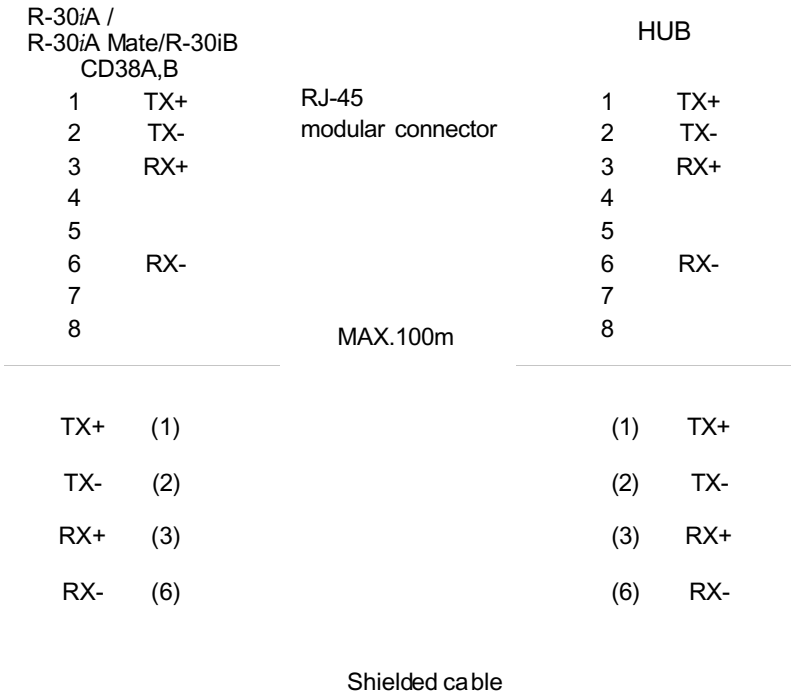
CD38R

| Pin No. | Signal name | Description |
|---|---|---|
| 1 | TX+ | Send + |
| 2 | TX- | Send - |
| 3 | RX+ | Receive + |
| 4 | | Not used |
| 5 | | Not used |
| 6 | RX- | Receive - |
| 7 | | Not used |
| 8 | | Not used |

# D.4 TWISTED-PAIR CABLE SPECIFICATION

## D.4.1 Cable Connection

The cable used for connection between the 100BASE-TX interface, CD38R, of the Ethernet board/Data Server board and the hub is connected as follows:




- Keep the total cable length within 100 m.
  Do not extend the cable more than is necessary. (Except for the cable recommended by FANUC)
- The figure above shows the cable connection when cables are crossed in the hub.
  "X" is usually indicated at the port of the hub to signify that cables are crossed in the hub.

| R-30*i*A / R-30*i*A Mate | | HUB | |
|---|---|---|---|
| | | X | Cross-connected cables |
| 1 TX+ | | TX+ 1 | |
| 2 TX- | | TX- 2 | |
| 3 RX+ | | RX+ 3 | |
| 6 RX- | | RX- 6 | |

## D.4.2 Cable Materials

CAUTION
Unshielded cable (UTP cable) is commercially available as 100BASE-TX twisted-pair cable: You should, however, use shielded Category 5 twisted-pair cable (STP cable) to improve the resistance to electrical noise in an FA environment.

### Recommended Cables

| Manufacturer | Specification | Remarks |
|---|---|---|
| FURUKAWA ELECTRIC CO., LTD. | DTS5087C-4P | Twisted-pair cable |
| NISSEI ELECTRIC CO., LTD. | F-4PFWMF | Single-conductor cable |

NOTE
The recommended cables cannot be connected to moving parts.

### Recommended cable (for movable parts)

| Manufacturer | Specification | Remarks |
|---|---|---|
| Oki Electric Cable Co., Ltd. | AWG26 4P TPMC-C5-F(SB) | Dedicated to FANUC |
| Shinko Electric Industrial Co., Ltd. | FNC-118 | |

### Specification

- Electric characteristics:
  Conforms to EIA/TIA 568A Category 3 and Category 5.
  From the viewpoint of attenuation performance, ensure that the length to the hub is 50 m or less.
- Structure:
  Group shielded (braided shield). A drain wire is available.
  The conductor is an AWG26 annealed copper twisted wire, with a sheath thickness of 0.8 mm and an outer diameter of 6.7 mm ±0.3 mm.
- Fire retardancy
  UL1581 VW-1
- Oil resistance
  Conforms to the FANUC internal standards (equivalent to the conventional oil-resistant electric cables).
- Flexing resistance:
  1,000,000 times or more with a bending radius of 50 mm (U-shaped flex test)
- UL style No.
  AWM 20276 (80°C/30V/VW-1)

NOTE
Be sure to use the connector TM21CP-88P(03) manufactured by HIROSE ELECTRIC CO., LTD. for this cable.

Cable assembly

Oki Electric Cable Co., Ltd. can also supply the cable assembly mentioned above.
Contact Oki Electric directly to determine the specifications (length, factory test, packing, and so forth) for purchase.

## D.4.3 Connector Specification

Use an 8-pin modular connector (RJ-45) with the twisted-pair cable for the Ethernet connection. The following connectors or equivalents must be used.

| For general use | Specification | Manufacturer | Remarks |
|---|---|---|---|
| Solid wire | 5-569530-3 | Tyco Electronics AMP K.K.. | |
| Solid wire | MS8-RSZT-EMC | SK KOHKI CO., LTD. | Special tools required |
| Twisted-pair cable | 5-569552-3 | Tyco Electronics AMP K.K.. | |
| Twisted-pair cable | TM11AP-88P | HIROSE ELECTRIC CO., LTD. | Special tools required |

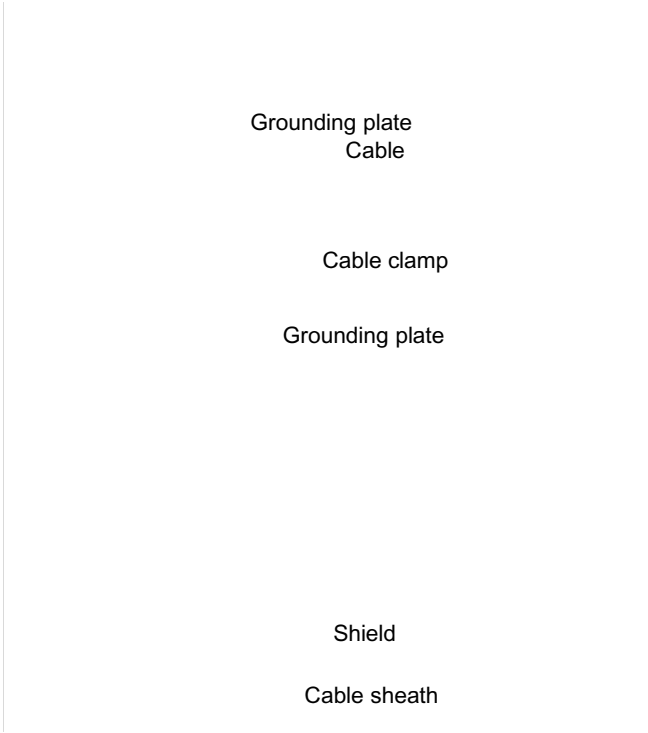
| For movable parts | Specification | Manufacturer | Remarks |
|---|---|---|---|
| For cable AWG26 4P TPMC-C5-F(SB) or FNC-118 | TM21CP-88P(03) | HIROSE ELECTRIC CO., LTD. | NOTE |

NOTE
Information about TM21CP-88P(03):
Connector (standard product of the manufacturer)
Drawing number: A63L-0001-0823#P
Manufacturer: HIROSE ELECTRIC CO., LTD.
Manufacturer type number: TM21CP-88P(03)
Conforms to EIA/TIA 568A Category 3 and Category 5.
For assembly with a cable, contact HIROSE ELECTRIC CO., LTD. directly.
(From HIROSE ELECTRIC CO., LTD., "TM21CP-88P(03) Connection Procedure Manual (Technical Specification No. ATAD-E2367)" is available as a technical document.)

# D.5 ANTI-NOISE MEASURES

## D.5.1 Clamping and Shielding of Cables

The Ethernet twisted pair cable needs to be clamped in the same way as the cables need to be shielded, as shown below. The clamping is required to shield and fix the cable. Be sure to perform the clamping to ensure the stable operation of the system.
As shown in the figure, strip a part of the cable sheath to expose the metal shield and push the shield against the grounding plate with the clamping hardware.




NOTE
Be sure to clamp and shield the cable to ensure the stable operation of the system.

NOTE
1 Upon completion of cabling, perform a communication test sufficiently not only before but also after system operation to ensure anti-noise measures.

A-Cabinet

Cramp

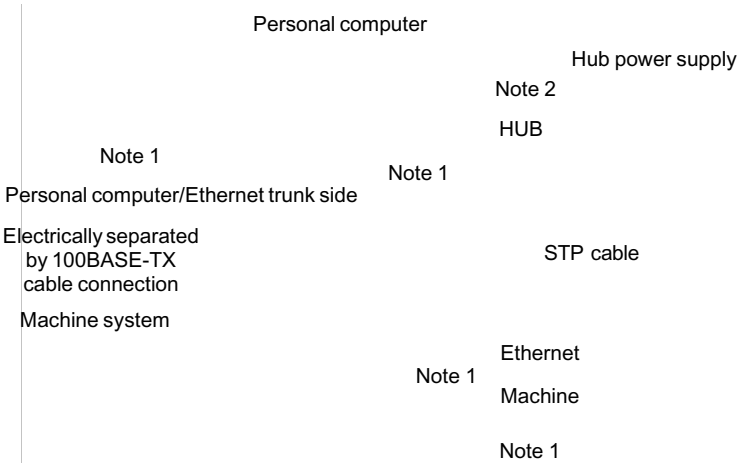
R-30*i*B

R-30*i*A

B-cabinet

Cramp

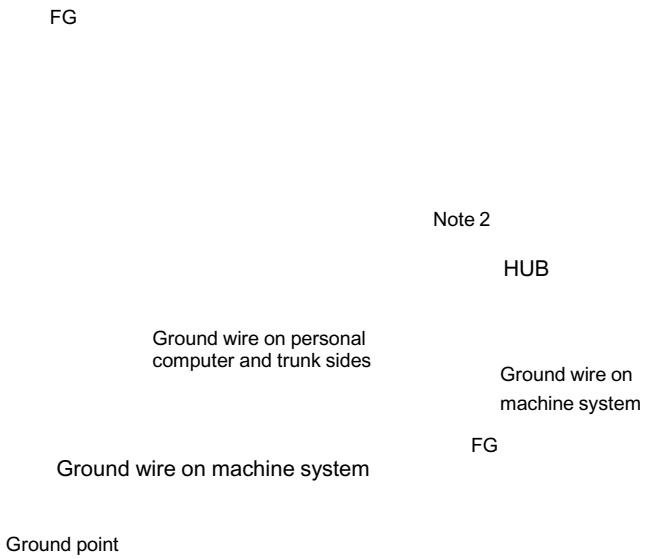
R-30*i*B




Small-Scale Network

NOTE

1 The ground between PC/HUB side and robot system side must be separated. If it is impossible to separate the ground because there is only one grounding point, connect the ground cable for each system to the grounding point independently. (See figure below.)
The resistance for grounding must be less than 100-ohm (Class D). The thickness of the robot controller's ground cable is the same as the thickness of AC power cable or more. At least thickness of 5.5mm$^2$ is necessary.
2 NOTE that the number of allowable hub-to-hub connections depends on the type of hub.
3 There is possibility that noise makes the obstacle of communication even if the ground is separated using the 100BASE-TX. In the case of using the FAST Ethernet/FAST Data Server under the worst environment, please separate between the PC/Trunk line side and robot system side completely using the 100BASE-FX (Optical fiber media).




Wiring on a single ground point

# D.6 CHECK ITEMS AT INSTALLATION

The following table lists check items at installation.

| Check item | | Description | Check |
|---|---|---|---|
| Ethernet cable | | | |
| | Type | Use cables which satisfies all the following conditions: | |
| | | 1) With shielding | |
| | | 2) Twisted-pair cable | |
| | | 3) Category 5 | |
| | Length | The cable length shall be within 100 m (50 m for a movable cable recommended by FANUC). | |
| | Connection | For a twisted-pair cable, the following pins shall be paired: | |
| | | | |
| | | 1) Pin No. 1 (TX+) – pin No. 2 (TX-)<br>2) Pin No. 3 (RX+) – pin No. 6 (RX-) | |
| | Separation | The Ethernet cables shall be bound separately from the following cables or covered with an electromagnetic shield: | |
| | | 1) Group A: AC power lines, power lines for motors, and others | |
| | | 2) Group B: Current DC (24 VDC) and others | |
| | Shielding | For a shielded cable, the part of which outer coating is peeled off and exposed shall be fixed to the ground plate with a clamp fixture. | |
| | Connectors | Any cable connector shall not be pulled (to prevent poor contact of the connector). | |
| | Wiring | No cable shall be laid under a heavy object. | |
| | Bending radius | Please confirm the specification of the cable. | |
| | For movable part | For a movable part, a cable for a movable part shall be used. | |
| HUB | | | |
| | Use conditions | The "cautions on use" of the hub shall be observed (A terminating resistor shall be mounted properly if required). | |
| | Grounding | The hub shall be grounded. | |
| | Cabinet | The hub shall be installed in an enclosed cabinet. | |
| | Vibration | The hub shall be installed so that it is not affected by vibration. | |
| | Bending radius | The bending radius shall be at least four times as long as the diameter of the cable. | |

# INDEX

# REVISION RECORD

| Edition | Date | Contents |
|---|---|---|
| 02 | Oct., 2012 | • Supported R-30*i*B controller. |
| 01 | Aug., 2008 | |