

FANUC Robot **series**

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

Ethernet Function OPERATOR'S MANUAL

B-82974EN/04

- **Original Instructions**

Thank you very much for purchasing FANUC Robot.

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

- No part of this manual may be reproduced in any form.
- The appearance and specifications of this product 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 endeavor to include all pertinent matters. There are, however, a very large number of operations that must not or cannot be performed, and if the manual contained them all, it would be enormous in volume. It is, therefore, requested to assume that any operations that are not explicitly described as being possible are "not possible".

SAFETY PRECAUTIONS

This chapter must be read before using the robot.

For detailed functions of the robot operation, read the relevant operator's manual to understand fully its specification.

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

For safe use of FANUC robots, you must read and follow the instructions in “FANUC Robot SAFETY HANDBOOK (B-80687EN)”.

1 DEFINITION OF USER

The personnel can be classified as follows.

Operator:

- Turns the robot controller power on/off
- Starts the robot program from operator panel

Programmer or Teaching operator:

- Operates the robot
- Teaches the robot inside the safety fence

Maintenance engineer:

- Operates the robot
 - Teaches the robot inside the safety fence
 - Maintenance (repair, adjustment, replacement)
-
- Operator is not allowed to work in the safety fence.
 - Programmer/Teaching operator and maintenance engineer is allowed to work in the safety fence. Works carried out in the safety fence include transportation, installation, teaching, adjustment, and maintenance.
 - To work inside the safety fence, the person must be trained on proper robot operation.

Table 1 lists the work outside the safety fence. In this table, the symbol “○” means the work allowed to be carried out by the worker.

Table 1 List of work outside the fence



	Operator	Programmer or Teaching operator	Maintenance engineer
Turn power ON/OFF to Robot controller	○	○	○
Select operating mode (AUTO, T1, T2)		○	○
Select remote/local mode		○	○
Select robot program with teach pendant		○	○
Select robot program with external device		○	○
Start robot program with operator's panel	○	○	○
Start robot program with teach pendant		○	○
Reset alarm with operator's panel		○	○
Reset alarm with teach pendant		○	○
Set data on teach pendant		○	
Teaching with teach pendant		○	
Emergency stop with operator's panel	○	○	○
Emergency stop with teach pendant	○	○	○
Maintain for operator's panel		○	
Maintain for teach pendant			○

In the robot operating, programming and maintenance, the operator, programmer/teaching operator and maintenance engineer take care of their safety using at least the following safety protectors.

- Use clothes, uniform, overall adequate for the work
- Safety shoes
- Helmet

2 DEFINITION OF SAFETY NOTATIONS

To ensure the safety of users 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 using the robot.

Symbol	Definitions
 WARNING	Used if hazard resulting in the death or serious injury of the user will be expected to occur if he or she fails to follow the approved procedure.
 CAUTION	Used if a hazard resulting in the minor or moderate injury of the user, or equipment damage may be expected to occur if he or she fails to follow the approved procedure.
NOTE	Used if a supplementary explanation not related to any of WARNING and CAUTION is to be indicated.

- Check this manual thoroughly, and keep it handy for the future reference.

TABLE OF CONTENTS

SAFETY PRECAUTIONS.....	s-1
1 OVERVIEW	1
1.1 OVERVIEW	1
1.2 BACKUP AND RESTORE	2
1.2.1 Overview	2
1.2.2 Application File Backup and Restore.....	2
1.3 FILE TRANSFER PROTOCOL (FTP).....	2
1.4 TCP/IP PROTOCOL.....	3
1.5 BOOTP AND TFTP PROTOCOLS	3
1.6 TELNET	3
1.7 DOMAIN NAME SERVICE (DNS)	3
1.8 WEB SERVER.....	3
1.9 PROXY SERVER	4
1.10 POINT-TO-POINT PROTOCOL CONNECTIVITY.....	4
1.11 DYNAMIC HOST CONFIGURATION PROTOCOL	4
1.12 SOCKET MESSAGING	4
1.13 SIMPLE NETWORK TIME PROTOCOL (SNTP).....	4
1.14 ETHERNET PACKET SNIFFER.....	4
1.15 ROS INTERFACE PACKETS OVER ETHERNET (RIPE)	4
1.16 HOST COMMUNICATIONS	5
1.16.1 Overview	5
1.16.2 Architecture.....	5
1.16.3 Devices	6
2 SETTING UP TCP/IP.....	7
2.1 OVERVIEW	7
2.2 HARDWARE REQUIREMENTS AND INSTALLATION	7
2.2.1 Overview	7
2.2.2 Hardware Requirements	7
2.3 DISPLAYING THE ETHERNET HARDWARE (MAC) ADDRESS	13
2.3.1 Overview	13
2.3.2 Ethernet Hardware (MAC) Address.....	13
2.3.3 Ethernet Hardware (MAC) Address Locations	18
2.4 SETTING UP TCP/IP.....	19

2.4.1	Caution for Setting IP Address.....	23
2.5	FANUC SERVER ACCESS CONTROL (FSAC).....	24
2.5.1	Overview	24
2.5.2	Access Levels	24
2.5.3	Access Denied	25
2.5.4	System Variables.....	25
2.5.5	Example Configuration	26
3	FTP OPERATIONS	27
3.1	OVERVIEW	27
3.2	SETTING UP AND STARTING FTP	27
3.3	FTP CLIENT USERNAMES AND PASSWORDS	32
3.4	ACCESSING AND USING CLIENT DEVICES	33
3.4.1	Access Description	33
3.4.2	File Specification for Client Devices	33
3.4.3	Starting and Stopping a Client Device	34
3.4.4	Teach Pendant File Access.....	35
3.5	ACCESSING SERVER DEVICES	35
3.5.1	Overview	35
3.5.2	Access Description	35
3.5.3	Starting and Stopping a Server Device.....	36
3.5.4	Blocking Downloads of Certain File Groups	36
3.5.4.1	Features.....	37
3.5.4.2	Examples	37
3.6	FTP SERVICES.....	37
3.6.1	Overview	37
3.6.2	Environment Services.....	38
3.6.3	File Transfer Services.....	39
3.6.4	Directory Services	39
3.6.5	Miscellaneous FTP Information	40
3.7	ACCESSING USER PROGRAM, SETUP, AND DIAGNOSTIC INFORMATION.....	41
3.7.1	Overview	41
3.7.2	System Files	42
3.7.3	Error Log Files	43
3.7.4	FTP Transfer Log	44
4	DOMAIN NAME SERVICE (DNS).....	45
4.1	OVERVIEW	45

4.2	DEFINING DNS PARAMETERS	45
5	TELNET	48
5.1	OVERVIEW	48
5.2	SETTING UP TELNET ON YOUR ROBOT	48
5.2.1	Telnet Setup.....	48
5.2.2	Connecting to a Telnet Server	50
6	WEB SERVER.....	51
6.1	OVERVIEW	51
6.2	SETTING UP THE WEB SERVER	51
6.2.1	Overview	51
6.2.2	Using FANUC Server Access Control (FSAC) to Control Access to the Web Server	52
6.3	USING THE WEB SERVER	52
6.3.1	Overview	52
6.3.2	Connecting to a Robot Home Page	52
6.3.3	Customizing Your Robot Home Page	54
6.3.4	Customizing Diagnostic Files, Variable File Listings, and TP Program Listings ..	55
6.3.5	Running KAREL Programs from the Web Browser	57
6.3.6	Creating Web Pages Based on KAREL Programs	57
6.4	SERVER SIDE INCLUDES.....	63
6.4.1	Overview	63
6.4.2	Syntax.....	64
6.4.3	Global Variables.....	65
6.4.4	Local Variables.....	66
6.4.5	String Substitution.....	66
6.4.6	#ECHO Command.....	67
6.4.7	#INCLUDE Command	68
6.4.8	#EXEC Command	68
6.4.9	#SET Command	69
6.4.10	#IF, #ELIF, #ELSE, #ENDIF.....	69
6.4.11	#PRINTENV Command.....	70
6.4.12	SSI EXAMPLES	71
6.5	HTTP AUTHENTICATION	71
6.5.1	Overview	71
6.5.2	Operation	72
6.5.2.1	Overview	72
6.5.2.2	Robot controller password option not enabled	73

6.5.2.3	Robot controller password option enabled	73
6.5.2.4	Example configuration.....	74
6.5.2.5	Accessing <i>i</i> Pendant screens through the web server	74
7	PROXY SERVER	75
7.1	OVERVIEW	75
7.1.1	Operation of Proxy Server.....	75
7.1.2	Requirements for Using Proxy Server.....	75
7.2	CONFIGURATION OF PROXY SERVER	76
7.3	ERRORS RETURNED BY THE PROXY SERVER	77
8	POINT-TO-POINT PROTOCOL CONNECTIVITY.....	78
8.1	OVERVIEW	78
8.2	SETTING UP PPP ON YOUR CONTROLLER.....	78
8.2.1	Overview	78
8.2.2	Configuring the P2, and P3	79
8.2.3	Changing IP Addresses	80
8.3	SETTING UP PPP ON YOUR PC	81
8.3.1	Overview	81
8.3.2	Setting up PPP on a Network PC	81
9	DYNAMIC HOST CONFIGURATION PROTOCOL.....	103
9.1	OVERVIEW	103
9.1.1	Introduction to DHCP	103
9.1.2	Features of the Robot DHCP Client	103
9.2	SETTING UP DHCP ON THE ROBOT	103
9.2.1	DHCP Setup	103
9.2.2	Advanced DHCP Setup	105
9.3	DHCP SYSTEM VARIABLES	107
9.4	DHCP TROUBLESHOOTING.....	108
10	SOCKET MESSAGING.....	109
10.1	OVERVIEW	109
10.2	SYSTEM REQUIREMENTS	109
10.2.1	Overview	109
10.2.2	Software Requirements	109
10.2.3	Hardware Requirements	109
10.3	CONFIGURING THE SOCKET MESSAGING OPTION	109
10.3.1	Overview	109
10.3.2	Setting up a Server Tag	110

10.3.3	Setting up a Client Tag	111
10.4	SOCKET MESSAGING AND KAREL	113
10.4.1	Overview	113
10.4.2	MSG_CONNECT(string, integer).....	113
10.4.3	MSG_DISCO(string, integer).....	114
10.4.4	MSG_PING(string, integer)	114
10.4.5	Exchanging Data during a Socket Messaging Connection.....	114
10.5	NETWORK PERFORMANCE.....	114
10.5.1	Overview	114
10.5.2	Guidelines for a Good Implementation	114
10.6	PROGRAMMING EXAMPLES.....	115
10.6.1	Overview	115
10.6.2	A KAREL Client Application	115
10.6.3	A KAREL Server Application.....	117
10.6.4	ANSI C Loopback Client Example	119
11	SIMPLE NETWORK TIME PROTOCOL (SNTP)	121
11.1	OVERVIEW	121
11.2	SETTING UP SNTP	121
11.3	USING SNTP.....	123
11.4	TROUBLESHOOTING.....	125
12	ETHERNET PACKET SNIFFER	126
12.1	OVERVIEW	126
12.2	SETTING UP THE ETHERNET PACKET SNIFFER	126
12.3	USING THE RING BUFFER AND TRIGGERS.....	127
13	ROS INTERFACE PACKETS OVER ETHERNET (RIPE)	128
13.1	OVERVIEW	128
13.2	RIPE SETUP	128
13.3	FILE ACCESS	130
13.4	ASSOCIATED OPTIONS	131
13.5	XML CONFIGURATION FILE.....	131
13.6	TELNET	132
13.7	VARIABLE ACCESS	132
13.8	SYNCHRONIZED TIMING	132
13.9	NETWORK DESIGN CONSIDERATIONS.....	132
14	PC SHARE	135
14.1	OVERVIEW	135

14.2	SETTING UP AND STARTING PC SHARE	135
14.2.1	Setting Up PC Share Client Tags	135
14.2.2	Configuring PC Share	138
14.3	ACCESSING AND USING PC SHARE CLIENT DEVICES	139
14.3.1	Access Description	139
14.3.2	File Specification for PC Share Client Devices.....	140
14.3.3	Starting and Stopping a Client Device	140
14.3.4	Teach Pendant File Access.....	141
14.3.5	Saving FANUC iRVision log to PC share folder.....	141
14.4	PASSWORDS AND SECURITY	142
14.4.1	Passwords	142
14.4.2	Supported Security Protocols	142
14.5	CONFIGURE THE REMOTE PC	142
15	ADVANCED iPendant FUNCTIONS	146
15.1	OVERVIEW	146
15.2	SETUP	146
15.2.1	Installation of iPendant Controls	146
15.2.2	Configuring Internet Explorer ®	146
15.2.3	Testing the Network Connection.....	156
15.2.3.1	Troubleshooting remote connection	157
15.3	MONITOR iPendant FUNCTION	157
15.3.1	Overview	157
15.3.2	Operation	158
15.3.2.1	PC connection to the controller	158
15.4	NAVIGATE iPendant FUNCTION.....	159
15.4.1	Overview	159
15.4.2	Operation.....	160
15.4.2.1	PC connection to the controller	160
15.4.2.2	Keys	161
15.4.2.3	Editing guidelines	163
15.4.2.4	Background editing guidelines	163
15.5	JOGGING iPendant FUNCTION	164
15.5.1	Overview	164
15.5.2	Operation	164
15.5.2.1	PC connection to the controller	164
15.5.2.2	Keys	167
15.5.2.3	Troubleshooting jogging iPendant connection	167
15.6	DISPLAYING iPendant KEY SHEET	167

15.6.1	Vertical <i>i</i> Pendant Key Sheet.....	168
15.6.2	Horizontal <i>i</i> Pendant Key Sheet	168
15.6.3	Limitations of <i>i</i> Pendant Key Sheet.....	169
15.7	LIMITATIONS(GENERAL).....	169
15.7.1	Limitations(Monitor <i>i</i> Pendant function).....	169
15.7.2	Limitations(Navigate <i>i</i> Pendant function)	170
15.7.3	Limitations(Jogging <i>i</i> Pendant function)	170
15.8	OPTION CONFIGURATION	171
16	REMOTE <i>i</i>Pendant FUNCTION.....	172
16.1	OVERVIEW	172
16.2	CONNECTION TO REMOTE <i>i</i> Pendant	173
16.2.1	Settings	173
16.2.2	Connection	173
16.3	MONITOR <i>i</i> Pendant (JECHO)	175
16.3.1	Overview	175
16.3.2	Connection	175
16.4	NAVIGATE <i>i</i> Pendant (JCGTP)	176
16.4.1	Overview	176
16.4.2	Connection	176
16.4.3	Screens.....	177
16.4.3.1	Keys	177
16.4.3.2	Show keys.....	179
16.5	JOGGING <i>i</i> PENDANT (JITP)	179
16.5.1	Overview	179
16.5.2	Connection	180
16.5.3	Screens.....	182
16.5.3.1	Screens (without <i>i</i> PendantControls)	182
16.5.3.2	Screens (with <i>i</i> PendantControls)	183
16.5.4	Jog Operation	185
16.6	Limitations	186
16.6.1	Limitations (general)	186
16.6.2	Limitations of Monitor <i>i</i> Pendant Function (JECHO)	187
16.6.3	Limitations of Navigate <i>i</i> Pendant Function (JCGTP)	187
16.6.4	Limitations of Jogging <i>i</i> Pendant Function (JITP)	187
17	<i>i</i>RCONNECT	189
17.1	OVERVIEW	189
17.2	REQUIREMENTS.....	189

17.3	SETUP AND CONFIGURATION	189
17.3.1	iRConnect Screen	190
17.3.2	iRConnect Emails Screen	191
17.3.3	Configuration File	192
17.3.4	Matching Alarms	195
17.3.5	Attachments	196
17.4	TEACH PENDANT PROGRAMS	196
17.4.1	Program: IRC_STATUS	196
17.4.2	Program: IRC_COUNTER	196
17.4.3	Program: IRC_MSG	196
17.4.4	Program: IRC_STLABEL	197
17.5	VARIABLE CHANGE SETUP AND USE	197
17.5.1	Variable Change Setup	198
17.5.2	Variable Change Output	199

APPENDIX

A	DIAGNOSTIC INFORMATION	203
A.1	VERIFYING NETWORK CONNECTIONS	203
A.1.1	Overview	203
A.1.2	Ethernet Status LEDs	203
A.1.3	PING Utility	203
A.2	ETHERNET LEDS	205
A.3	10 BASE-T/100 Base T-X CONNECTOR PIN ASSIGNMENTS	206
B	CONFIGURE FTP WITH A KAREL COMMAND FILE.....	207
B.1	CONFIGURING NETWORK PARAMETERS WITH A KAREL COMMAND FILE	207
C	NETWORK DESIGN AND PERFORMANCE.....	211
C.1	GUIDELINES FOR USING ETHERNET	211
D	CABLE CONNECTION	213
D.1	CONNECTING TO Ethernet	213
D.2	LEADING OUT THE Ethernet CABLE	214
D.3	100BASE-TX CONNECTOR (CD38A/CD38B) PIN ASSIGNMENTS	215
D.4	TWISTED-PAIR CABLE SPECIFICATION	215
D.4.1	Cable Connection	215
D.4.2	Cable Materials.....	216
D.4.3	Connector Specification	217

D.5 ANTI-NOISE MEASURES 217

 D.5.1 Clamping and Shielding of Cables 217

 D.5.2 Grounding the Network 221

D.6 CHECK ITEMS AT INSTALLATION 223

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.

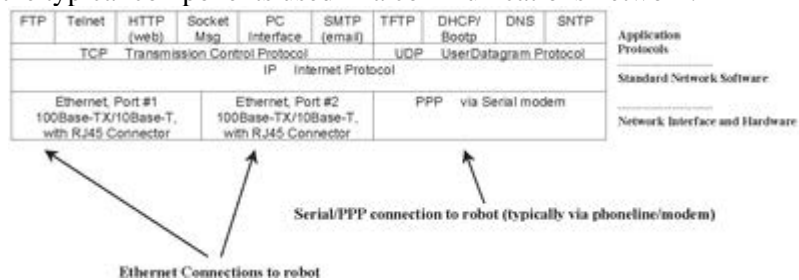


Fig. 1.1 Network components

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

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(MAC) 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 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 *iPendant*. For the browser on the *iPendant* to be able to view web servers on the network, it needs a proxy server to proxy web requests from the *iPendant* 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 (**D**ynamic **H**ost **C**onfiguration **P**rotocol) 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. Ethernet Sniffer Function(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.

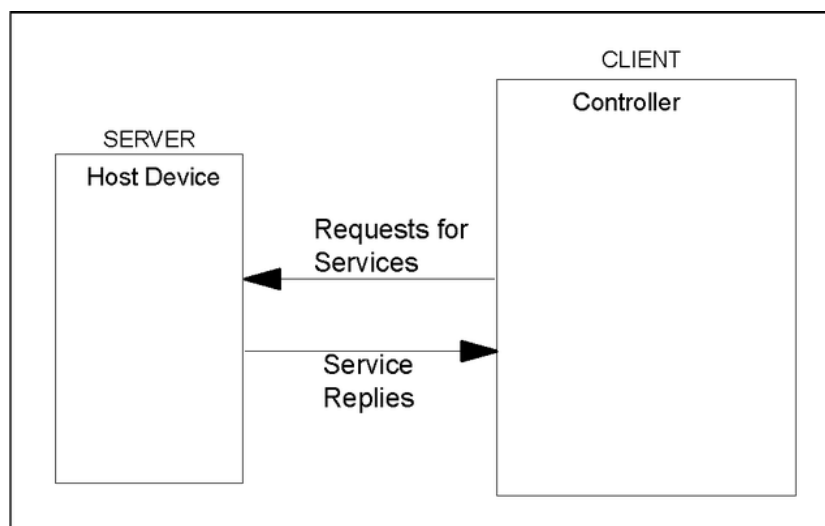


Fig. 1.16.2(a) The controller as a client

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).

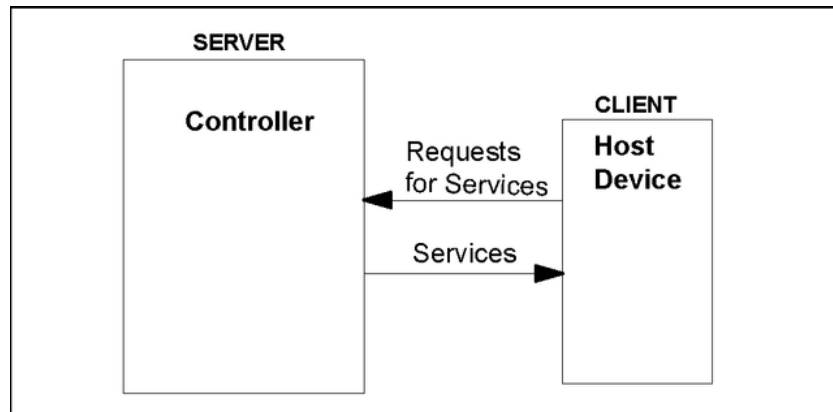


Fig. 1.16.2(b) The controller as a server

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-30iA, R-30iB, R-30iB Plus supports two 10 Base-T or 100 Base-TX interfaces through the RJ45 Ethernet connectors (CD38A and CD38B). R-30iB or R-30iB Plus 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-30iB Mate and R-30iB Mate Plus supports one or two 10 Base-T or 100 Base-TX interfaces. (Main board A supports one port. Main board B and Main board C supports two ports. Please refer table 2.2.2(a),(b) for each order number.)

Table 2.2.2(a) Order number for R-30iB Mate

Name	Order number for standard	Order number for open air
Main board A	A05B-2650-H001	A05B-2655-H001
Main board B	A05B-2650-H002	A05B-2655-H002
Main board C	A05B-2650-H003	A05B-2655-H003

Table 2.2.2(b) Order number for R-30iB Mate Plus

Name	Order number
Main board A	A05B-2680-H001
Main board B	A05B-2680-H002
Main board C	A05B-2680-H003

R-30iA 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-30iB 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 the controller which support only 1 interface.

Table 2.2.2(c) Ethernet configuration setup

Baud Rate/Duplex	Half Duplex	Full Duplex
10 MBPS	\$ENETMODE[].\$SPEED=0 \$ENETMODE[].\$FULL_DUPLEX=FALSE	\$ENETMODE[].\$SPEED=0 \$ENETMODE[].\$FULL_DUPLEX=TRUE
100 MBPS	\$ENETMODE[].\$SPEED=1 \$ENETMODE[].\$FULL_DUPLEX=FALSE	\$ENETMODE[].\$SPEED=1 \$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 changed. The baud rate and duplex mode will be set to the fastest setting that both devices on the list can support.

See Figure 2.2.2(a) for R-30iB. See Figure 2.2.2(b) for R-30iB Plus. See Figure 2.2.2(c) for R-30iA. See Figure 2.2.2(d) for R-30iB Mate. See Figure 2.2.2(e) for R-30iB Mate Plus. See Figure 2.2.2(f) for R-30iA Mate.

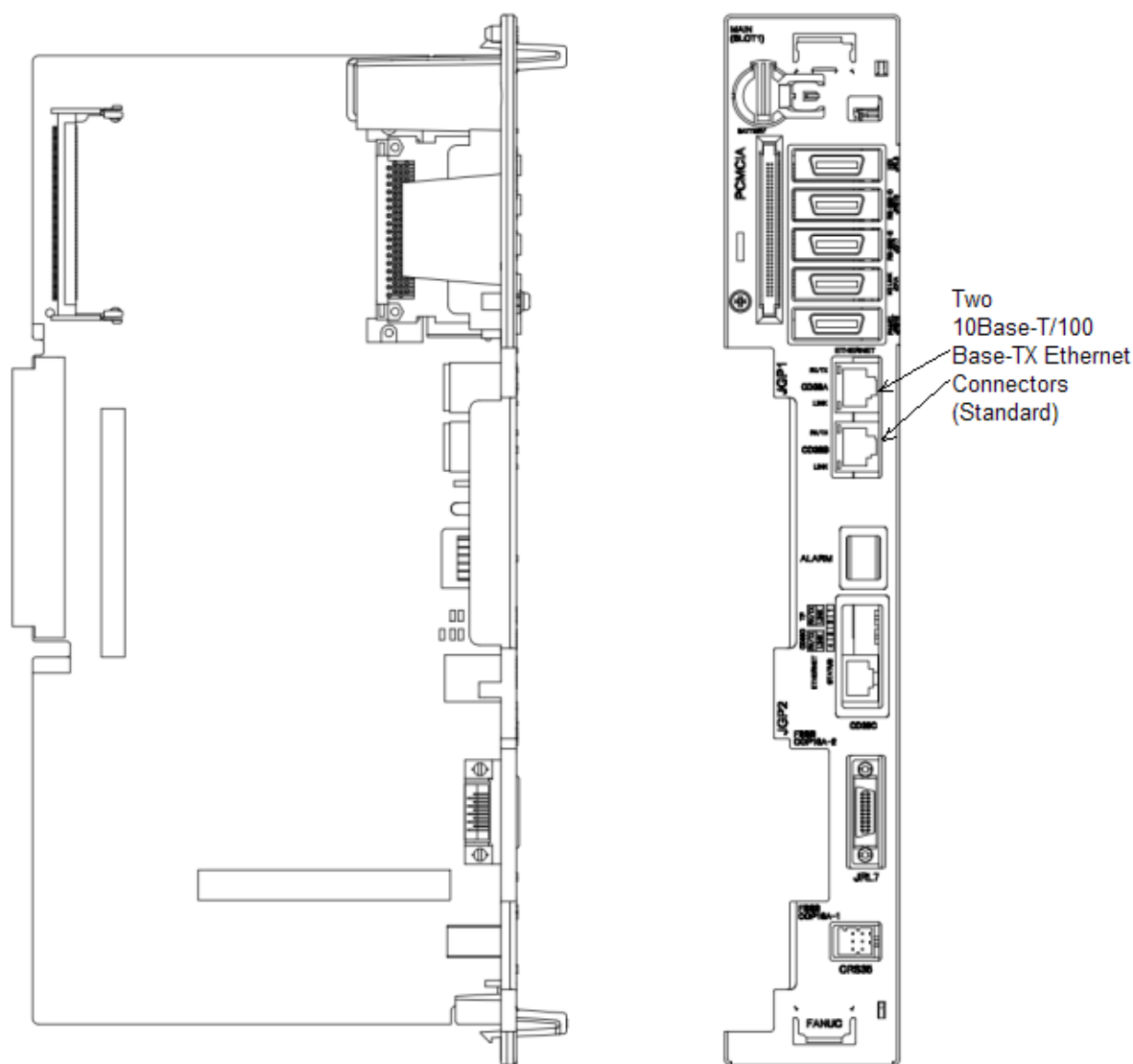


Fig. 2.2.2(a) Main board (R-30iB) Ethernet connectors

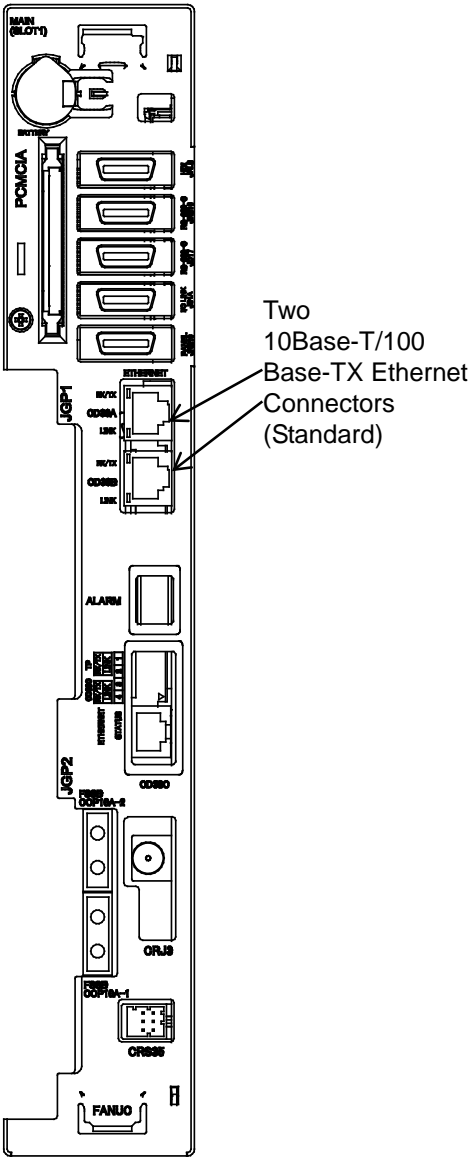


Fig. 2.2.2(b) Main board (R-30iB Plus) Ethernet connectors

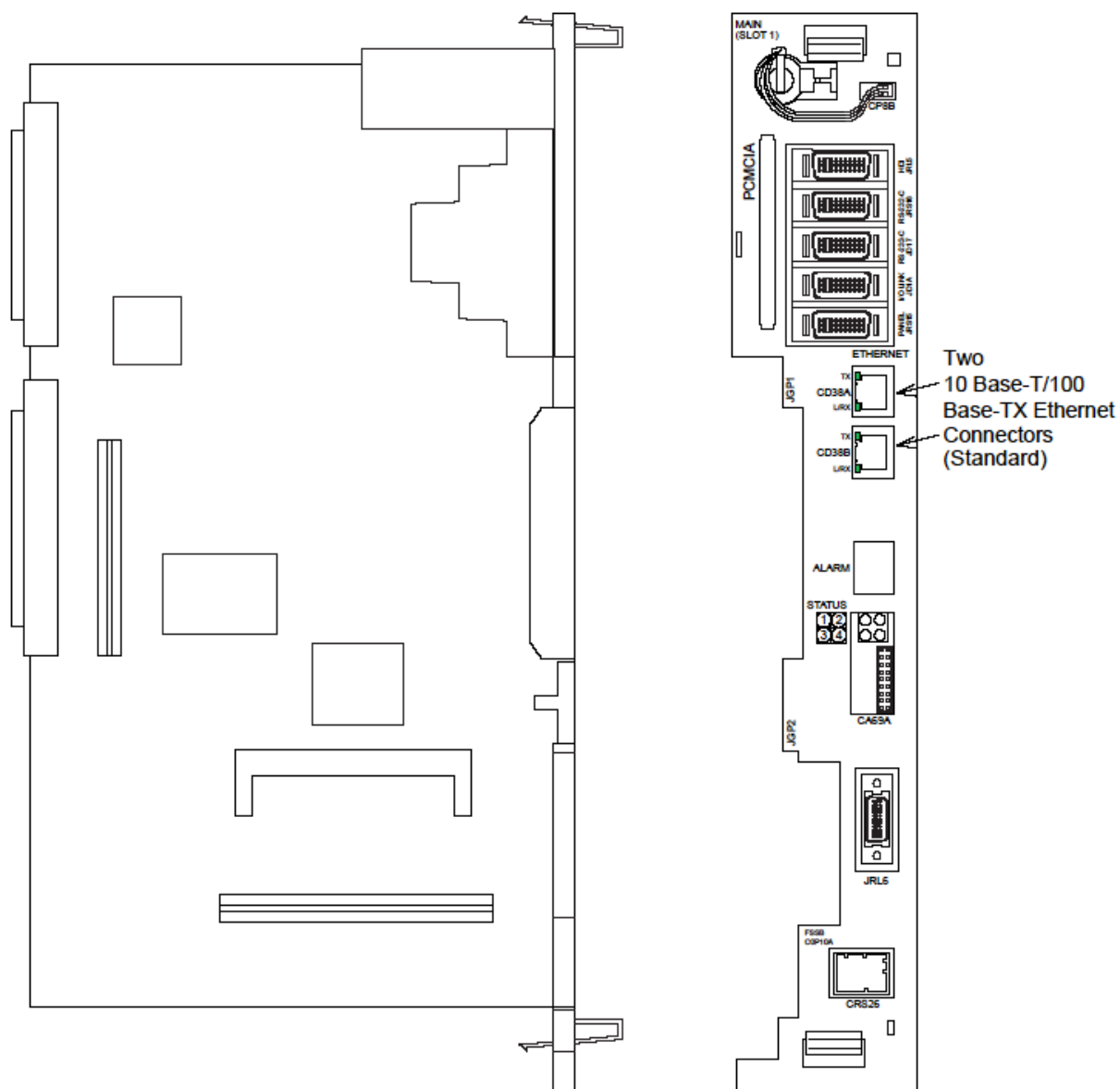
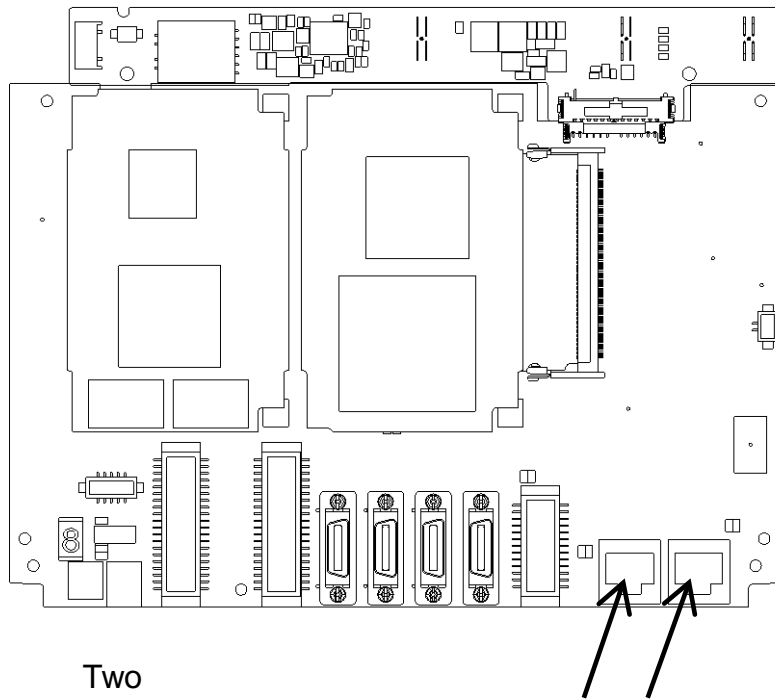


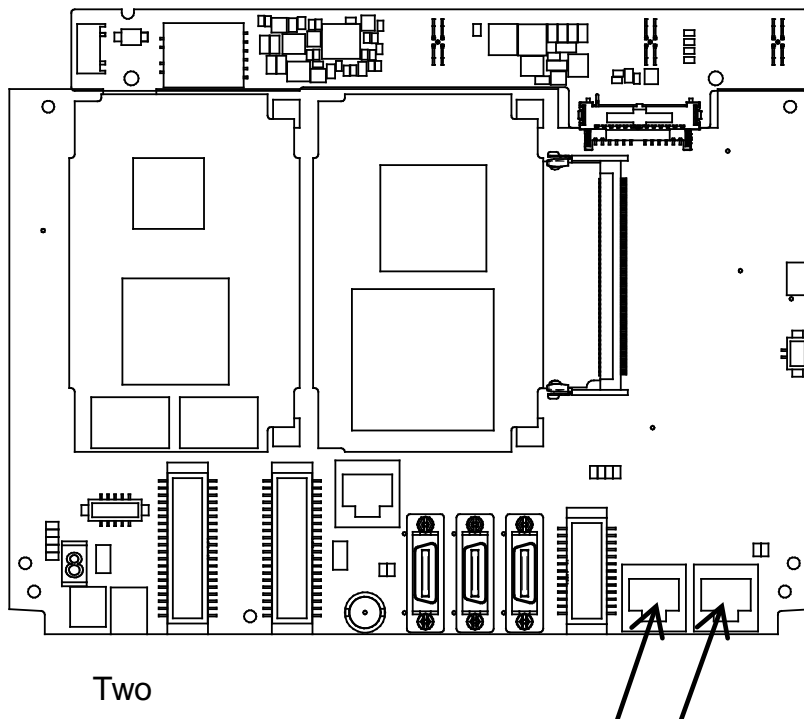
Fig. 2.2.2(c) Main board (R-30iA controller) Ethernet connectors



Two

10Base-T/100Base-TX Ethernet Connectors (Standard)

Fig. 2.2.2(d) Main board (R-30iB Mate) Ethernet connectors



Two

10Base-T/100Base-TX Ethernet Connectors (Standard)

Fig. 2.2.2(e) Main board (R-30iB Mate Plus) Ethernet connectors

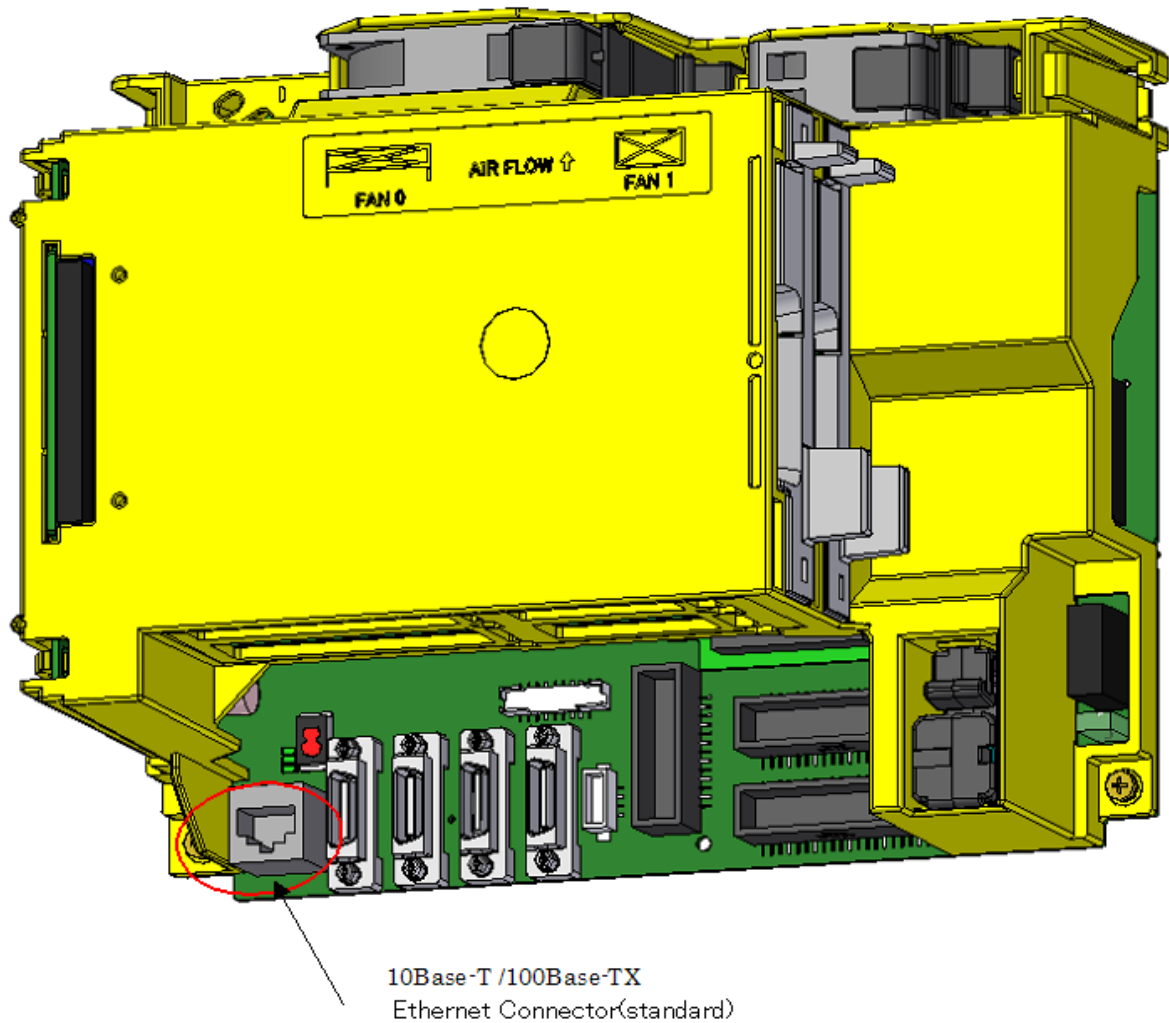


Fig. 2.2.2(f) Main board (R-30iA Mate) Ethernet connectors

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-30iB. See Figure 2.3.2(b) for R-30iB Plus. See Figure 2.3.2(c) for R-30iA. See Figure 2.3.2(d) for R-30iB Mate. See Figure 2.3.2(e) for R-30iB Mate Plus. See Figure 2.3.2(f) for R-30iA Mate.

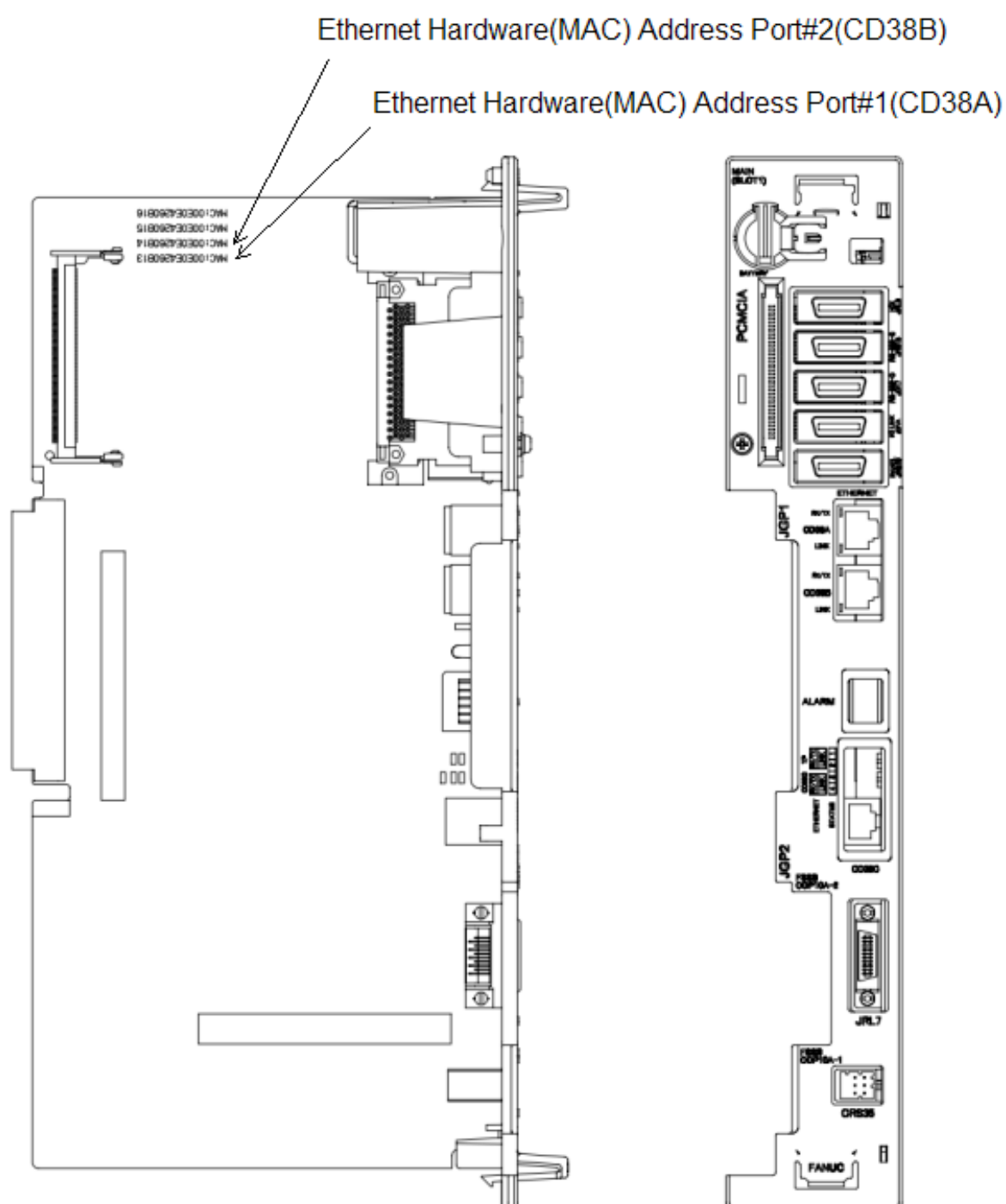


Fig. 2.3.2(a) Ethernet hardware (MAC) address on main board for R-30iB

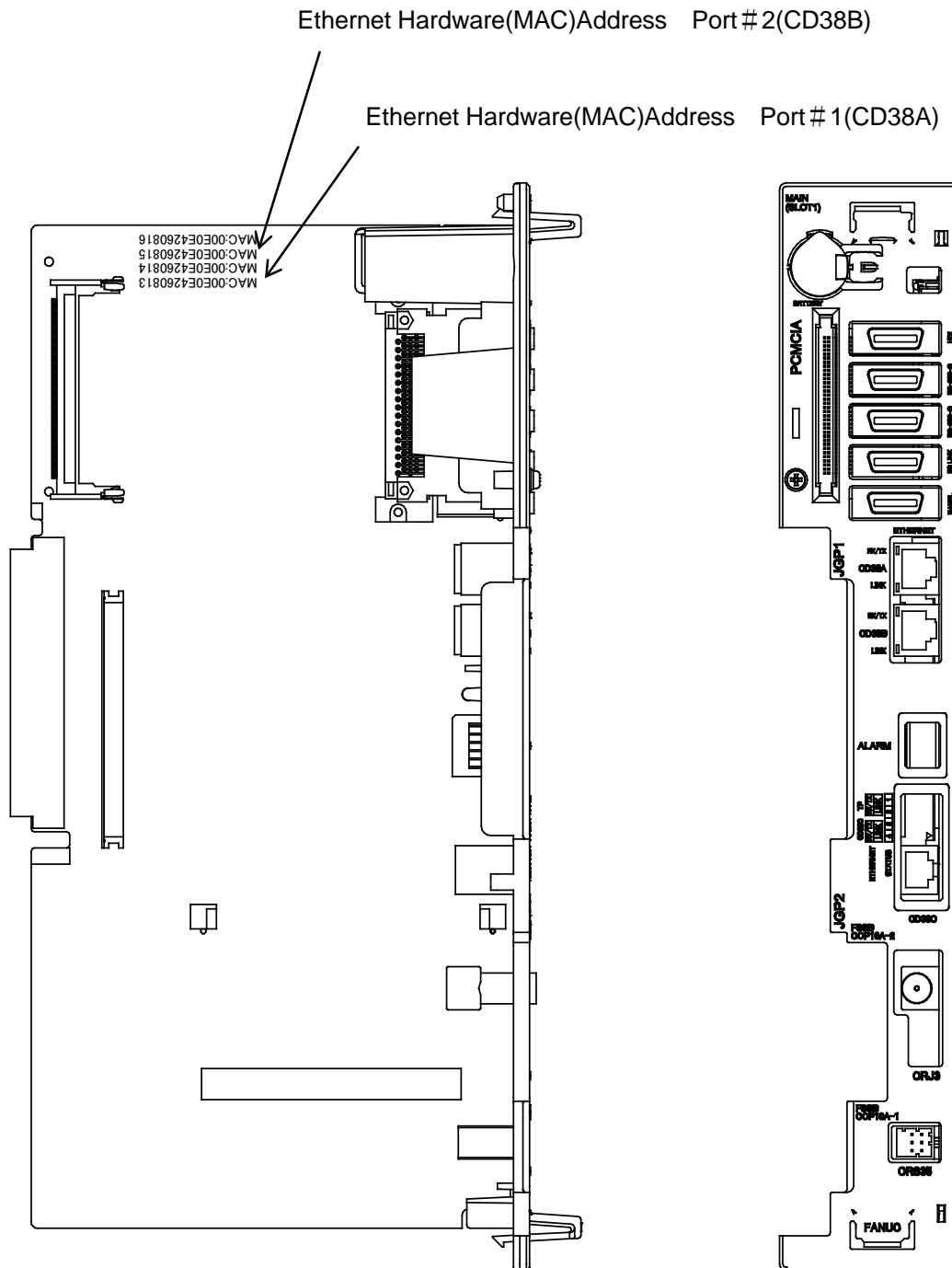


Fig. 2.3.2(b) Ethernet hardware (MAC) address on main board for R-30iB Plus

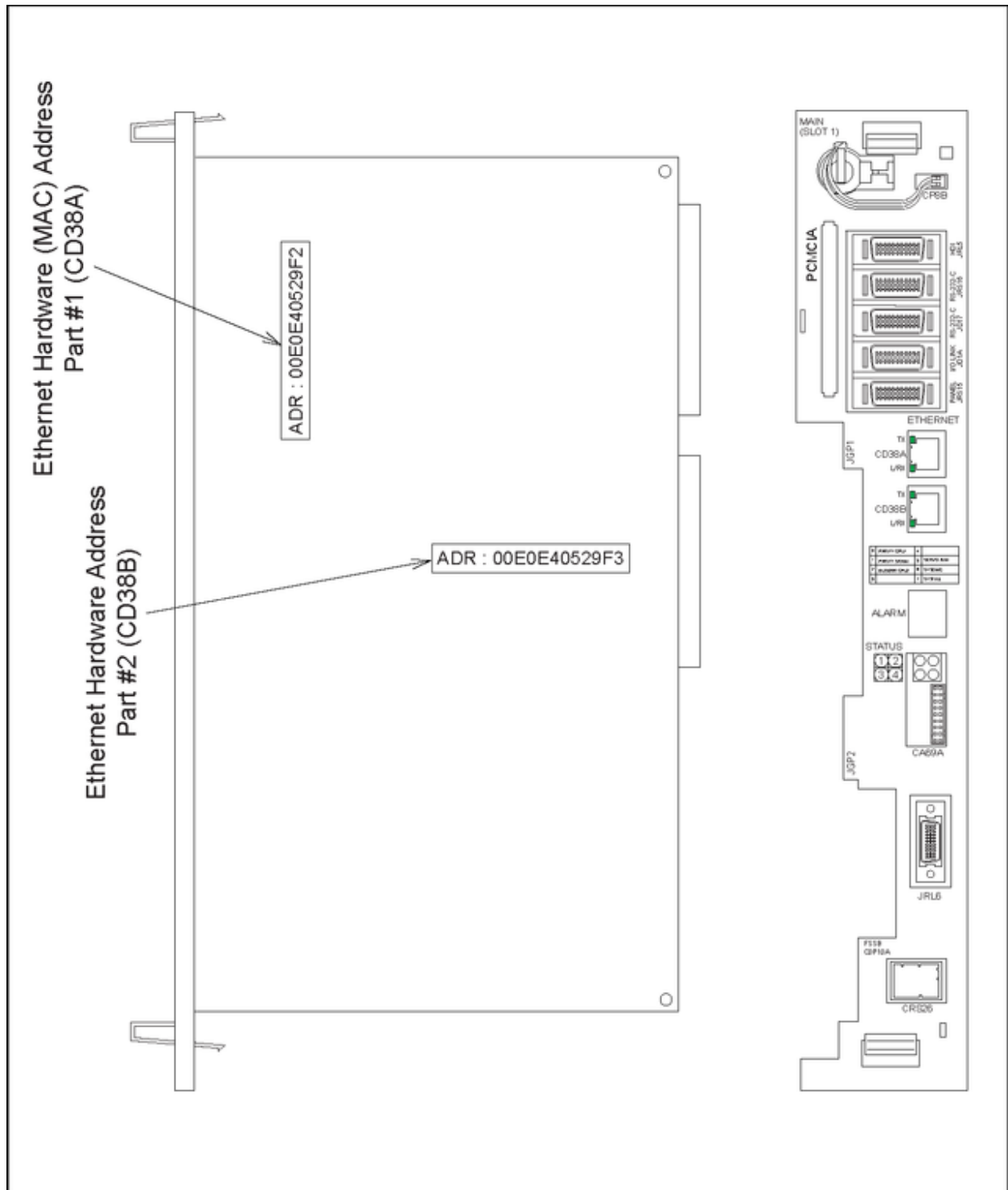
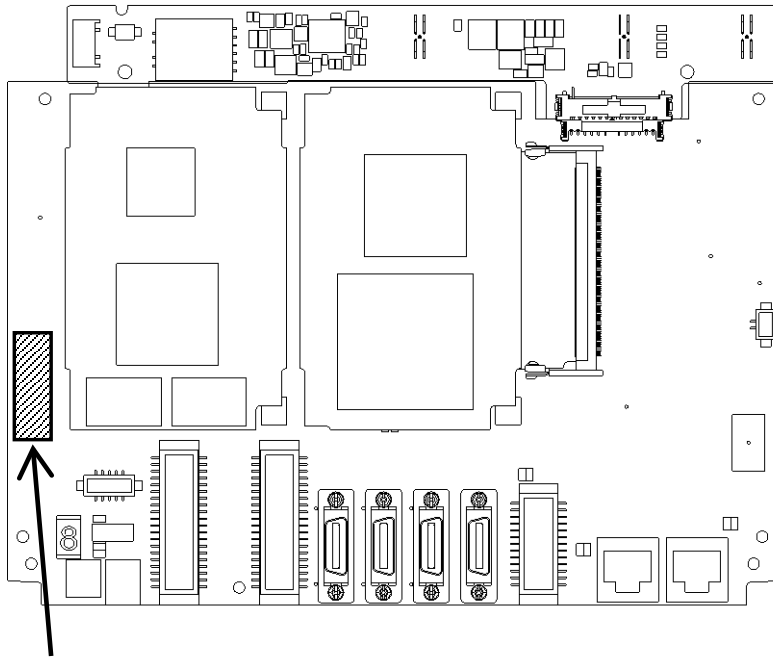


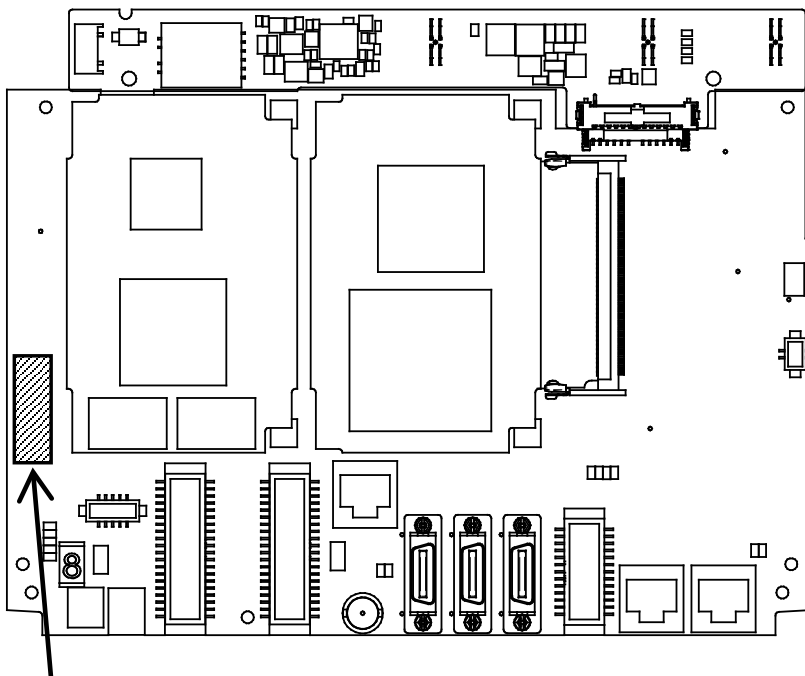
Fig. 2.3.2(c) Ethernet hardware (MAC) address on main board for R-30iA



Ethernet Hardware(MAC)Address Port #1(CD38A)

Ethernet Hardware(MAC)Address Port #2(CD38B)

Fig. 2.3.2(d) Ethernet hardware (MAC) address on main board for R-30i/B Mate



Ethernet Hardware(MAC)Address Port #1(CD38A)

Ethernet Hardware(MAC)Address Port #2(CD38B)

Fig. 2.3.2(e) Ethernet hardware (MAC) address on main board for R-30i/B Mate Plus

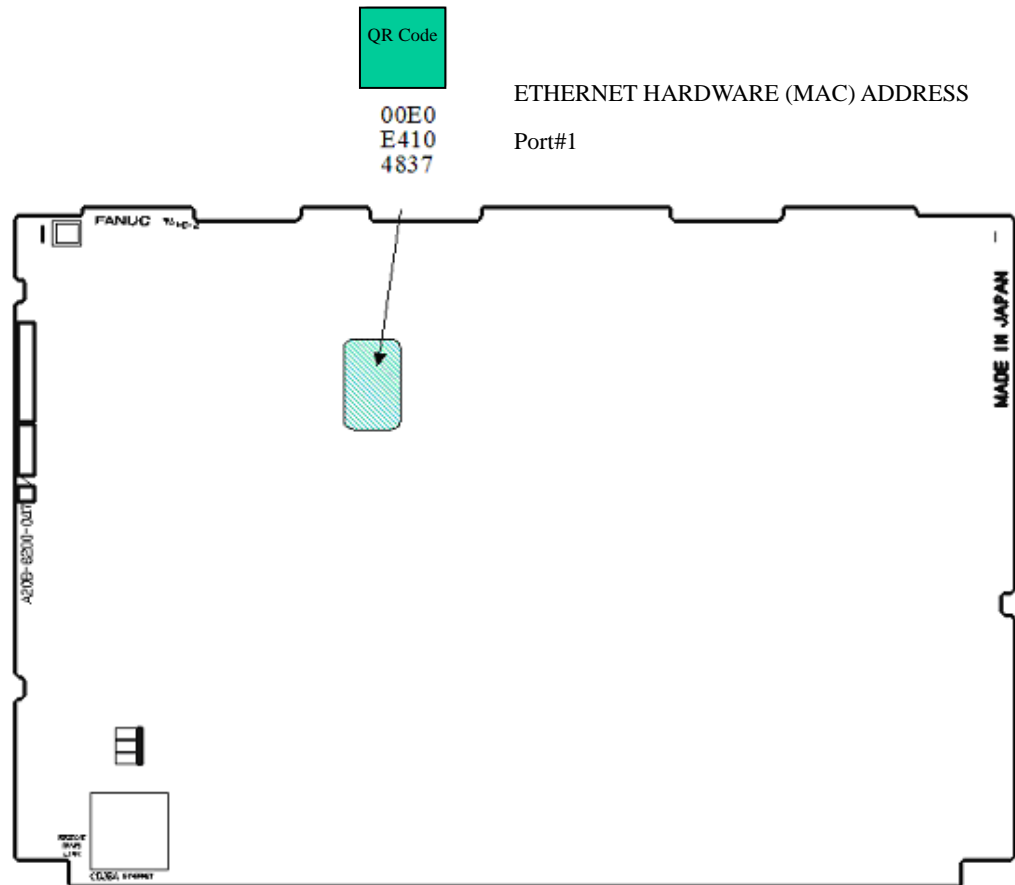


Fig. 2.3.2(f) Ethernet hardware (MAC) address on main board for R-30iA Mate

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) .
- DISPLAY MAC 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. The following screen will be displayed

```
***** 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. The following screen will be displayed

```

***** 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. The following screen will be displayed

```
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.

If the Enhanced Vision Ethernet Port option (R825) is loaded then the Vision Ethernet Port (CD38C) can also be configured using Procedure 2-2. It is referred to as Port#3 in the TCP/IP setup screen.

If this option is loaded and the Ethernet port is configured it supports FTP (file transfer) and HTTP (web) functionality.

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.
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.

ITEM	DESCRIPTION
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. 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. If the Enhanced Vision Ethernet Port option (R825) is loaded then the Vision Ethernet Port (CD38C) is also available and is referred to as Port #3.
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. If the Enhanced Vision Ethernet Port option (R825) is loaded then the Vision Ethernet Port (CD38C, referred to as Port #3) router setting is independent of the router for the first two 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

- Press the [MENU] key.
- Select SETUP.
- Press F1, [TYPE].
- Select Host Comm. The following screen will be displayed

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 Make sure TCP/IP is selected.
- 6 Press F3, DETAIL. The following screen will be displayed

SETUP HOST COMM		
TCP/IP		
Robot name:	PDEROB024	
Port # IP addr:	172.22.194.24	
Subnet mask:	255.255.240.0	
Board address:	08:00:19:02:F2:22	
Router IP addr:	172.22.192.1	
Host Name (LOCAL)		Internet Address
1	*****	*****
2	*****	*****
3	*****	*****
4	*****	*****
5	*****	*****
6	*****	*****
7	*****	*****
8	*****	*****
9	*****	*****
10	*****	*****
11	*****	*****
12	*****	*****
13	*****	*****
14	*****	*****
15	*****	*****
16	*****	*****
Host Name (SHARED)		Internet Address
1	*****	*****
2	*****	*****
3	*****	*****
4	*****	*****
5	*****	*****
6	*****	*****
7	*****	*****
8	*****	*****
9	*****	*****
10	*****	*****
11	*****	*****
12	*****	*****
13	*****	*****
14	*****	*****
15	*****	*****
16	*****	*****
17	*****	*****
18	*****	*****
19	*****	*****
20	*****	*****

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 (RJ45 labeled as CD38A) or interface #2 (RJ45 labeled as CD38B). Use the F3, Port key to change. If the Enhanced Vision Ethernet Port option (R825) is loaded then the Vision Ethernet Port(CD38C) is also available and is referred to as Port #3.
 - **Robot IP Address** — specify IP address of the robot. You can't change this when DHCP is enabled, if you need to change IP address, press F2 DHCP and change DHCP to FALSE.
 - **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 0.

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)	255.255.0.0
192 and 223 (Class C)	255.255.255.0

- 8 To enable changes, cycle power or initialize from press NEXT key and press F3 INIT.

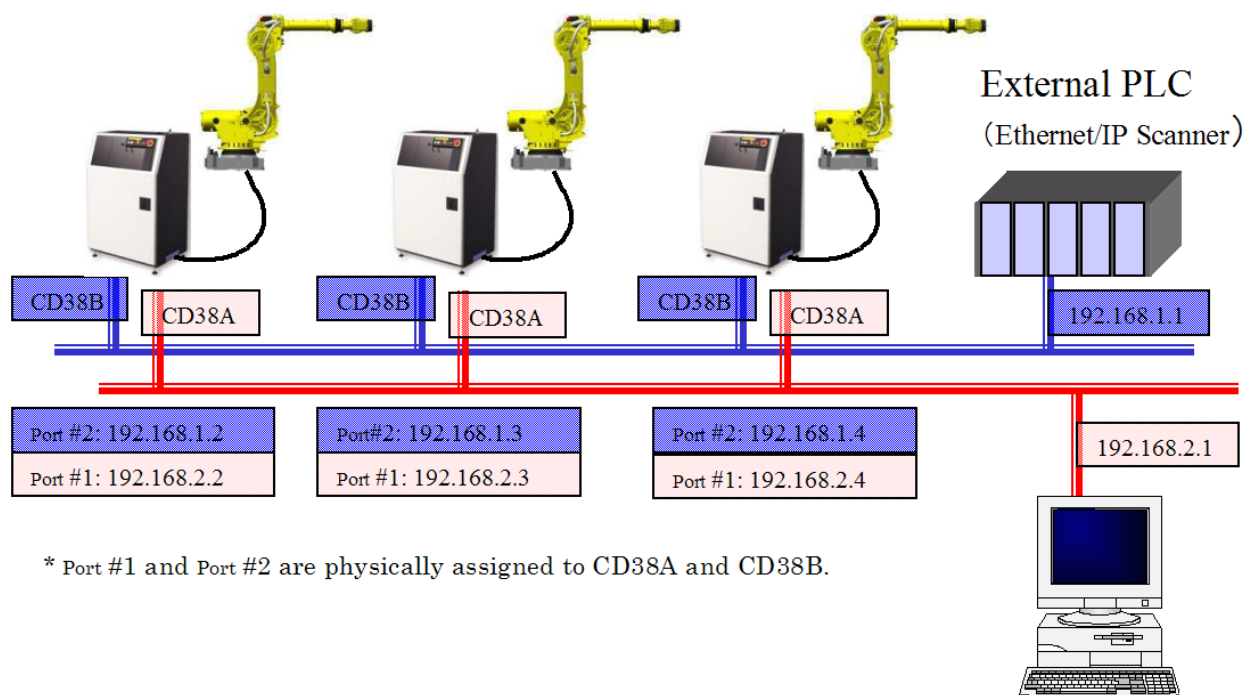
NOTE

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.



* Port #1 and Port #2 are physically assigned to CD38A and CD38B.

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: <ul style="list-style-type: none"> ● TP (teach pendant) ● .PC (p-code) ● .VR (variable)
2	Setup level	Program level, with additional access to download the following types of files: <ul style="list-style-type: none"> ● .SV (system) ● .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: <ul style="list-style-type: none"> ● 0, disabled (any value other than 1 will disable it) ● 1, enabled
\$FSAC_DEF_LV	Integer	FSAC Default Access Level. This can be set to: <ul style="list-style-type: none"> ● 0, operator level ● 1, program level ● 2, setup level ● 3–7, user-defined levels ● 8, installation level ● any other level is no access
\$FSAC_LIST[].\$CLNT_NAME	String	The name of the host system. Example: MYPC <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>NOTE The name must be in the LOCAL/SHARED host table or DNS must be installed to resolve names.</p> </div>
\$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: <ul style="list-style-type: none"> ● BIT 0: FTP ● BIT 1: Telnet ● 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 .

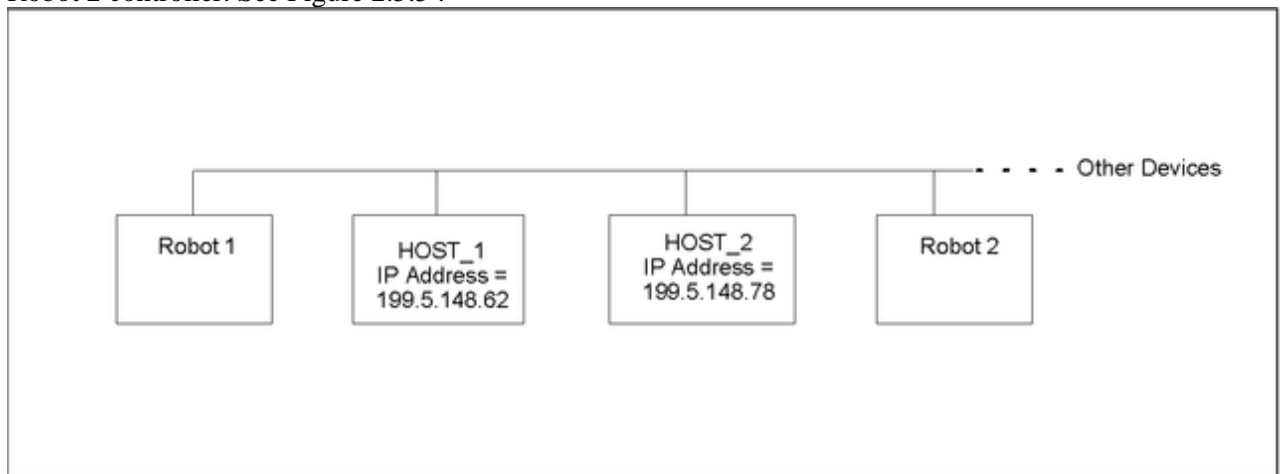


Fig. 2.5.5 Example configuration

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

NOTE

We recommend that you restrict access by setting the password of the FTP server to increase security.

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*	<p>This item specifies the desired startup (Power up) state for the selected tag. Three states are possible:</p> <ul style="list-style-type: none"> • UNDEFINED - the device is not defined. • DEFINED - the device is defined. • STARTED - the device is defined and started. <p>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.</p>

ITEM	DESCRIPTION
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. 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. <ul style="list-style-type: none"> When set to zero, no timeouts occur. 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.
Use Vision Ethernet Port	This item is only available if the Enhanced Vision Ethernet Port option (R825) is loaded. If TRUE then the tag will use the vision Ethernet port (CD38C). If FALSE then one of the first two Ethernet ports will be used.
* 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: <ul style="list-style-type: none"> UNDEFINED - the device is not defined. DEFINED - the device is defined. STARTED - the device is defined and started. 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. <ul style="list-style-type: none"> When set to zero, no timeouts occur. 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.

ITEM	DESCRIPTION
Username	This item is not used at this time.
Password	This item is not used at this time.
Use Vision Ethernet Port	This item is only available if the Enhanced Vision Ethernet Port option (R825) is loaded. If TRUE then the tag will use the vision Ethernet port (CD38C). If FALSE then one of the first two Ethernet ports will be use
* 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 the [MENU] key.
- 2 Select SETUP.
- 3 Press F1, [TYPE].
- 4 Select Host Comm. The following screen will be displayed.

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. The following screen will be displayed.

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) .
- 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.
 - Move the cursor to Protocol Name and press F4, [CHOICE]. A list of available protocol choices will be displayed.
 - Select FTP and press ENTER.
 - 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.

- Select the startup state you want and press ENTER.
 - Move the cursor to Inactivity Timeout, type the timeout value you want, in minutes, and press ENTER. The default value is 15 minutes.
 - Press F3, LIST, to display the list of server devices.
 - Repeat Step 5.c through Step 5.j for as many server devices as you are defining.
- To set up a client: (if required)
 - Press F4, [SHOW].
 - Select 2, Clients. The following screen will be displayed.

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]

- 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) .
- 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.
 - Move the cursor to Protocol Name and press F4, [CHOICE]. A list of available protocol choices will be displayed.
 - Select FTP and press ENTER
 - Move the cursor to Startup State and press F4, [CHOICE].
 - 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.

- 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.
 - 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 /.
 - Move the cursor to Inactivity Timeout, type the timeout value you want, in minutes, and press ENTER. The default value is 15 minutes.
 - Move the cursor to the Username field, and type in the username for the client to use to log into the remote FTP server.
 - Move the cursor to the Password field, and type in the password for the client to use to log into the remote FTP server.
 - Press F3, LIST, to display the list of client devices.
 - 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:
- Press F4, [SHOW].
 - Select Clients or Servers.
 - Move the cursor to the client or server you want to define and start.
 - Press F2, [ACTION].
 - Select 1, Define.
 - Press F2, [ACTION], again.
 - Select Start.
 - 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 (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 (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) (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

- Press the [MENU] key.
- Select SETUP.
- Press F1, [TYPE].

4. Select Host Comm. The following screen will be displayed.

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

- 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.
- Move the cursor to the password that corresponds to that username and press ENTER. Use the appropriate function keys to type the password and press ENTER. This item is case sensitive.
- Move the cursor to the corresponding timer and press ENTER. Use the numeric keys to type the time and press ENTER.
- Repeat Step 6 through Step 8 for the remaining usernames and passwords you are setting.
- 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 the [MENU] key.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm.
5. Press F4, [SHOW].
6. Select Clients. The following screen will be displayed.

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

- Press the [MENU] key.
- Select SETUP.
- Press F1, [TYPE].
- Select Host Comm.
- Press F4, [SHOW].
- Select Servers. The following screen will be displayed.

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]

- Press F2, [ACTION].
- 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.
- To configure the server device to start automatically at power up:
 - Move the cursor to the server tag you want to start up automatically and press F3, [DETAIL].
 - Move the cursor to Startup Mode and press F4, [CHOICE].
 - 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.

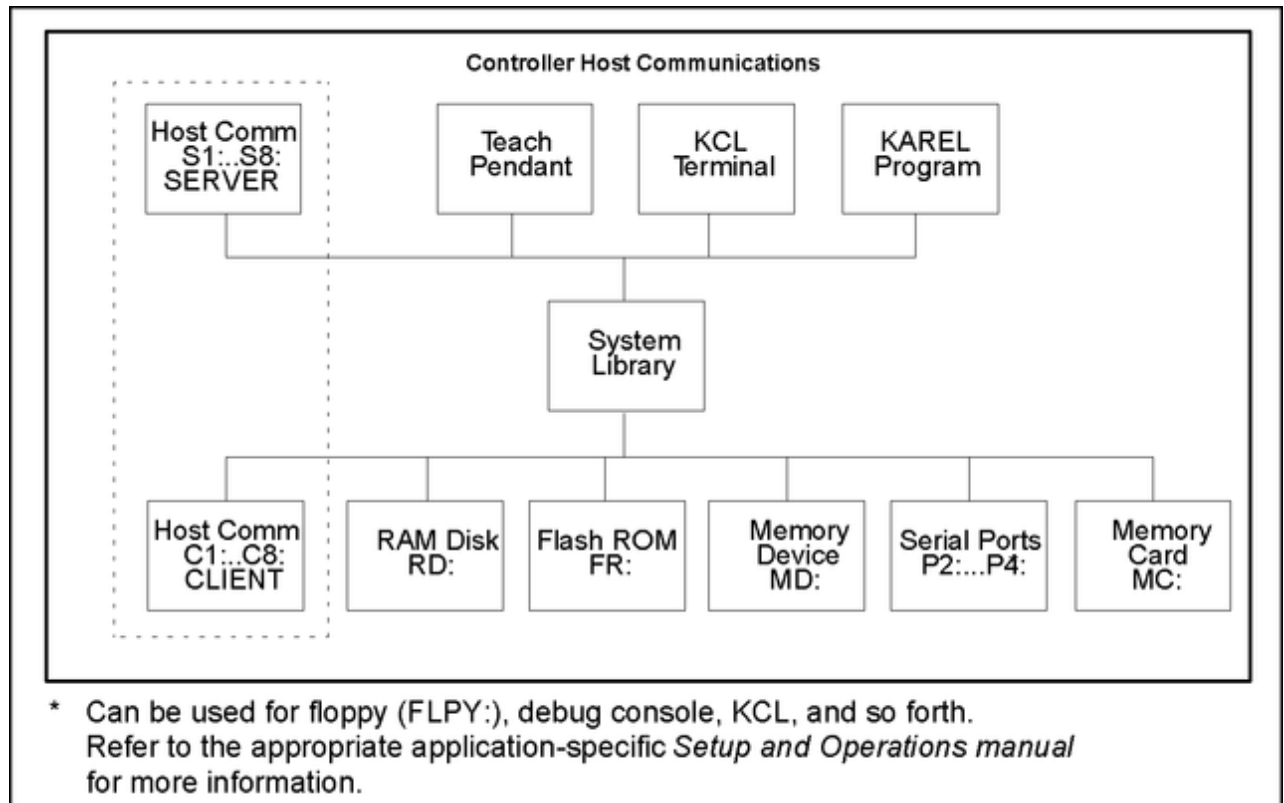


Fig. 3.6.1 Host Communications model

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:REGREEG.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 versions of SYSVARS.SV are SYSTEM.VA and SYMOTN.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.

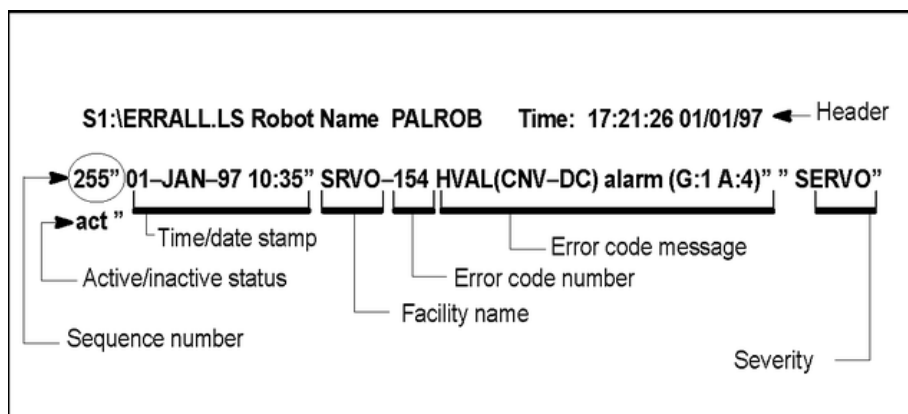


Fig. 3.7.3 Sample error log entry

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)

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	<p>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.</p> <div style="border: 1px solid black; padding: 5px;"> <p>NOTE DNS will not work if you do not provide a local domain name.</p> </div>

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

- Press the [MENU] key.
- Select SETUP.
- Press F1, [TYPE].
- Select Host Comm. The following screen will be displayed.

```

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

```

- 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 .
- Move the cursor to DNS and press F3, DETAIL. The following screen will be displayed.

```

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
*****
*****

```

- 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: <ul style="list-style-type: none"> ● OUTPUT - outputs from the controller ● INPUT - both input and output ● NONE - no access to the controller 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 Units: minutes Range: 0 - 99 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

- Press the [MENU] key.
- Select SETUP.
- Press F1, [TYPE].
- Select Host Comm. The following screen will be displayed.

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

- Move the cursor to TELNET and press F3, DETAIL. The following screen will be displayed.

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. The following screen will be displayed.

```

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 <robohost>
```

Where <robohost> 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: <robohostname> F No: F-xxxxxx)
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_TIMEOUT	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 iPendant 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(R-30iB and R-30iB Mate)

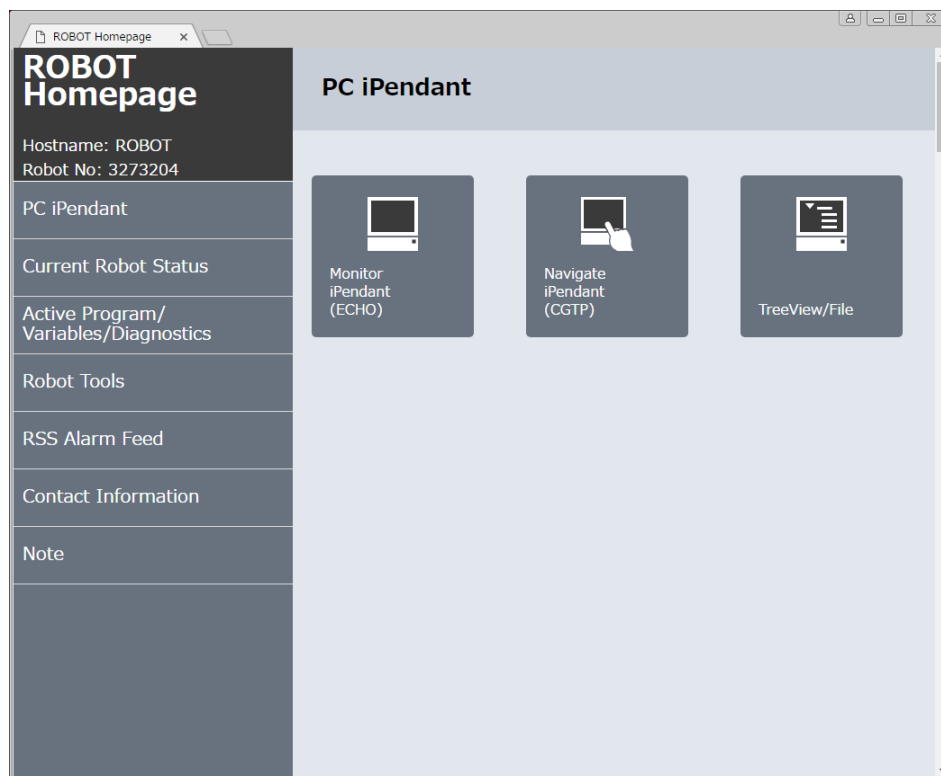


Fig. 6.3.2(d) Default robot home page(R-30iB Plus and R-30iB Mate Plus)

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) .

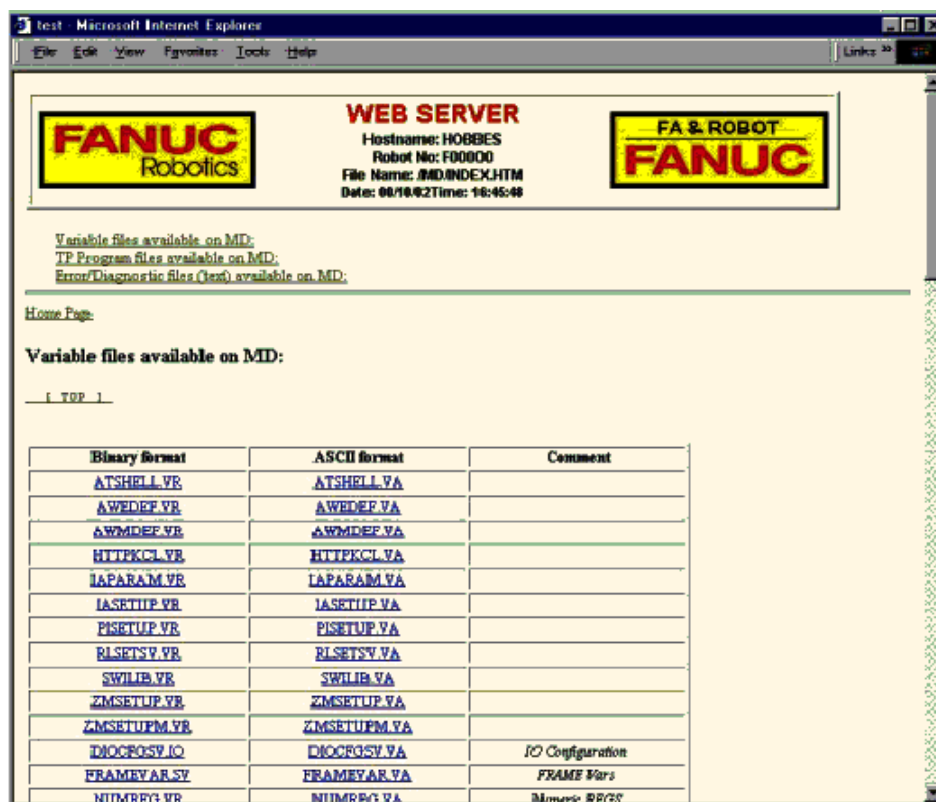


Fig. 6.3.2(e) Memory device index page

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 (MD:)	This link points to a section of this page that provides links to .SV file and any .VR file which is loaded.
TP Program Files (MD:)	This link points to a section of this page that provides links to .TP program loaded on the robot.
Error/Diagnostic Files (MD:)	This link points to a section of this page that provides links to 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" (8-bit binary data string that can take any value).

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
$HTTP_CTRL.$DBGVLV Access: RW: INTEGER = 0
$HTTP_CTRL.$KRL_TIMEOUT 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

```

Example 6.3.6 (b)

```

BEGIN
  if uninit(count1) then
    count1 = 300
  endif
  if uninit(count2) then
    count2 = 2
  endif
  count1 = count1 + 1
  if (count1 MOD 10 = 0) then
    count2 = count2 + 1
  endif
  OPEN FILE file1 ('RW', 'RD:RESPONSE.HTM')
  write file1('<HTML><HEAD><TITLE>ASG_DEMO.HTM</TITLE></HEAD>',cr)
--
-- *** Example of adding some graphics content to page ***
-- *** Be sensitive to file sizes! ***
--
-- write file1('<BODY BACKGROUND="../'')
-- write file1(BACKGROUND)
-- write file1('>',cr)
-- write file1('<CENTER> <H1><A NAME="TOP"><IMG SRC="../'')
-- write file1(PIC1,cr)
-- write file1('" WIDTH="593" HEIGHT="153"></A></H1> </CENTER>',cr)
write file1('<H1><CENTER><BOLD>IMPORTANT CUSTOMER ',cr,cr)
write file1('</CENTER></H1>',cr)
write file1('<H1><CENTER><BOLD>Production Counts for : ')
write file1('PRESS1',cr)
write file1('</CENTER></H1>',cr)

```

Example 6.3.6 (c)

```

write file1('<H1><CENTER><BOLD>ProductionCount: ')
write file1(count1,cr)
write file1('</BOLD></CENTER></H1>',cr)
write file1('<H1><CENTER><BOLD>Error Count: ')
write file1(count2,cr)
write file1('</BOLD></CENTER></H1>',cr)
-- write file1(TEXTTRLR,cr)
CLOSE FILE file1
END demo

```

To use this feature from a browser, set KAREL as the device and the program to run as the filename (demo in this example) (in brief, access like <http://robot1/KAREL/demo>).

Example 6.3.6 (d) Integrating KAREL Programs into the Web Browser

```
http://robot1/KAREL/demo
```

Using a Web Page to Pass Parameters to a KAREL Program

You can create applications that can pass parameters from a form in the browser to the KAREL program. The KAREL program is invoked based on the "submit" action in the form and parameters included in the form are passed with the URL. The web server complies with standards found in the HTTP 1.0 Specification. Note that only the HTTP "GET" method is supported at this time. The KAREL program must declare string variables whose names match any parameter names being passed from the form in order to access it. An additional string variable called "URL" should be declared to see the complete URL request sent from the browser (for debugging).

Example 6.3.6 (e) Variable Declaration for Using a Web Page to Pass Parameters to a KAREL Program

```
var
  URL : string[128]
  Textbox1 : string[12]
```

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) .

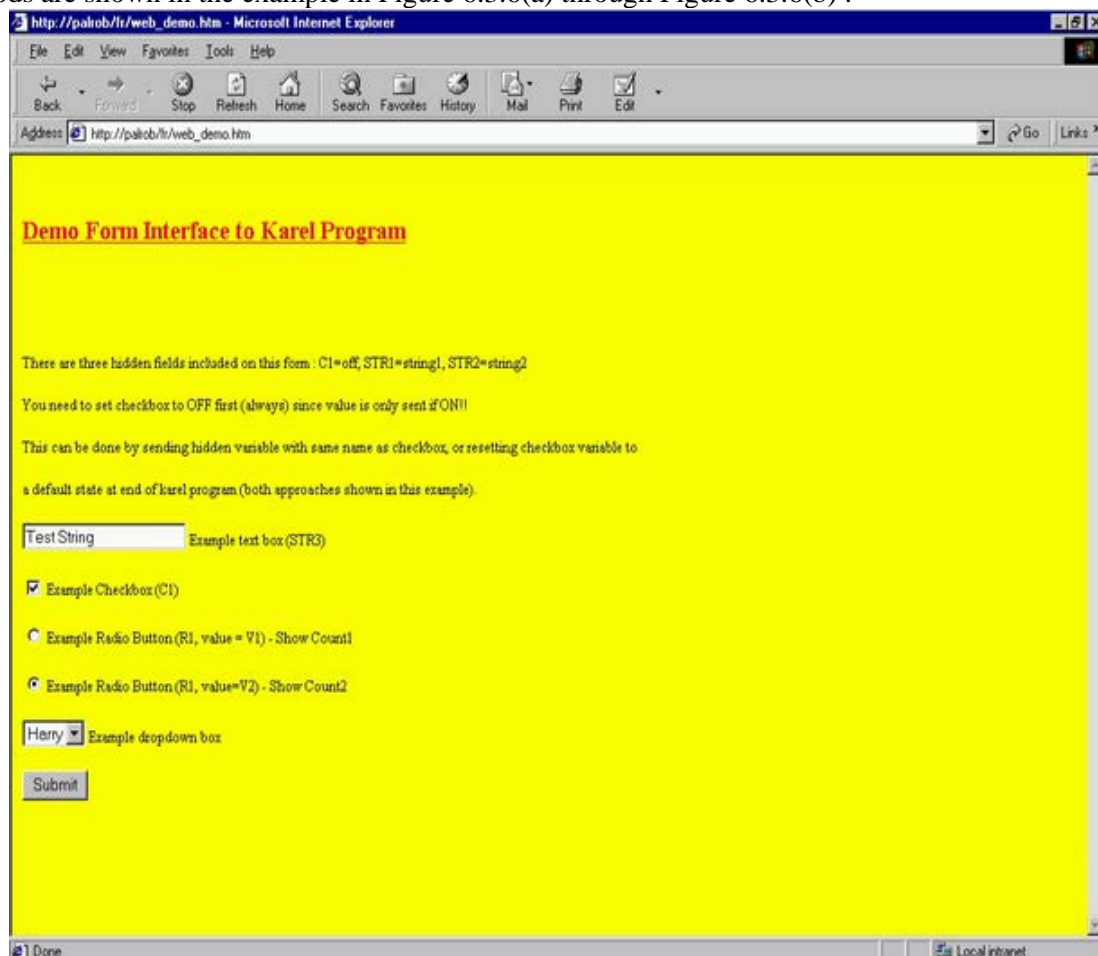


Fig. 6.3.6(a) Example KAREL based web page using parameters

```
<!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>&nbsp; </p>
<p><font color="#FF0000" size="5"><strong><u>Demo Form Interface
to Karel Program</u></strong></font></p>
<p>&nbsp; </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>&nbsp; </p>
  <p>There are three hidden fields included on this form : C1=off, STR1=string1,
STR2=string2</p>
  <p>You need to set checkbox to OFF first (always) since value is only sent if
ON!!</p>
  <p>This can be done by sending hidden variable with same name as checkbox, or
resetting checkbox variable to</p>
  <p>a default state at end of karel program (both approaches shown in this
example).</p>
  <p><input type="text" size="20" name="str3" value="string3">Example text box
(STR3)</p>
  <p><input type="checkbox" name="C1" value="ON"> Example Checkbox (C1)</p>
  <p><input type="radio" checked name="R1" value="V1"> Example Radio Button (R1,
value = V1) - Show Count1</p>
  <p><input type="radio" name="R1" value="V2"> Example Radio Button (R1, value=V2)
- Show Count2</p>
  <p><select name="D1" size="1">
    <option>Jim</option>
    <option>Joe</option>
    <option>Harry</option>
  </select> Example dropdown box</p>
  <p><input type="submit" name="B1" value="Submit"></p>
</form>
</body>
</html>
```

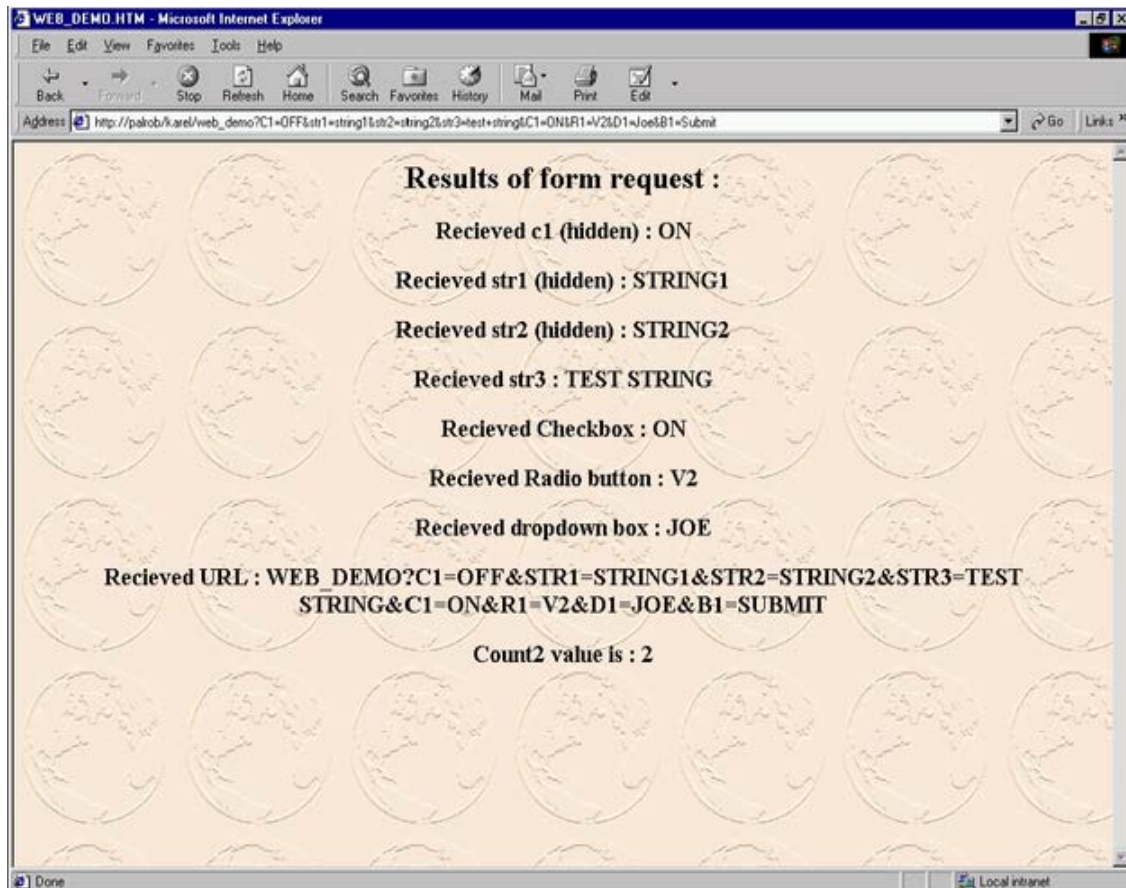


Fig. 6.3.6(b) Results of the demo form interface to a KAREL program

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=JIM&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(TEXTTRGLR,cr)
CLOSE FILE file1
END web_demo

```

6.4 SERVER SIDE INCLUDES

6.4.1 Overview

The 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 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: e.g. `<!--#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*/
 - TOOL /*tool 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:

- The "%20" is a space (ASCII value of a space is 20). This is needed since spaces are not considered valid URL syntax. If you enter spaces in a URL some browsers (IE) will automatically change them to the % encoding, while other browsers (Netscape) return an error indicating the URL is invalid. The code of a "\$" is %24.
- `<!--#exec cmd ="KAREL/web_demo?{_Query_STR}" -->`
- Run the non-motion KAREL program web_demo.pc and passes in arguments from the global variable _QUERY_STR.
- The KAREL program is responsible for writing RD:RESPONSE.HTM. This file will be included automatically and an error is generated if it does not exist.

The system variable associated with running KAREL programs must be set to allow execution of these commands. Refer to Section 6.2 for information on setting up the web server.

6.4.9 #SET Command

The #SET command is used to set the name and value of a local variable. Refer to Section 6.4.4 for details on maximum lengths and naming requirements.

NOTE

If the variable name already exists and the #SET command is issued, the value of the local variable is modified.

Examples

See the following #SET Command examples:

- For example, the following command will set the name of a local variable to _lvar1 and the value to 12:

Example 6.4.9 (a) Set the Name of a Local Variable

```
<!--#set var="_lvar1" value="12" -->
```

- A subsequent command to set _lvar1 will update the value:

Example 6.4.9 (b) Update the Value of the Local Variable

```
<!--#set var="_lvar1" value="15" -->
```

NOTE

Values are always set to uppercase.

6.4.10 #IF, #ELIF, #ELSE, #ENDIF

Conditional expressions allow results sent to the browser to be based on some condition. The conditional expression can allow the active alarms to be included only if the robot were faulted as one example.

Conditionals can be nested three levels deep. All comparisons are string comparisons unless both arguments are strictly numeric in which case a numeric compare is done. Three comparison operators are supported: >, <, =. String comparisons are not case sensitive. All conditional blocks must begin with #IF.

All conditional blocks must end with #ENDIF. The #ELSE command must come after #IF and #ELIF if it is used.

Examples

The following file called example.stm is placed on the robot FR: device:

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.

Example 6.4.12 (a) SSI Example

```
<HTML>
<HEAD><META HTTP-EQUIV="PRAGMA"CONTENT="NO-CACHE">
</HEAD>
<BODY>
<!--This is a comment-->
<P><H1>Hi - this is a test</H1>
<p>The value of $version is : <!--#echo var=$version -->
<p>The value of $acc_maxlmt is : <!--#echo var=$acc_maxlmt -->
<p>The value of $rmt_master is : <!--#echo var=$rmt_master -->
<p>The value of $io_auto_cfg is : <!--#echo var=$io_auto_cfg -->
<p>The value of $http_ctrl.$dbglvl is :
<!--#echo var=$http_ctrl.$dbglvl -->
<p>The value of $hosts_cfg[1].$state -->
    <!--#echo var=$hosts_cfg[1].$state -->
<p><strong><pre>The value of $hosts_cfg[1] is:
    <!--#echo var=$hosts_cfg[1] --></pre><strong>
<p>The value of $SCR.$TEMPER_LIMS[1]=<!--#echo var=$SCR.$TEMPER_LIMS[1] -->
```

Example 6.4.12 (b)

```
<p><pre>$SCR.$TEMPER_LIMS:<!--#echo var=$SCR.$TEMPER_LIMS --></pre>
<p>The value of din[6] is : <!--#echo var=din[6] -->
<p>The value of dout[5] is : <!--#echo var=dout[5] -->
<p>The value of tpin[TP_ENBL] is : <!--#echo var=tpin[249] -->
<p>The value of tpin[TP_ENBL] is : <!--#echo var=tpin[249] -->
<p>The value of tpout[FAULT] is : <!--#echo var=tpout[1] -->
<p>The value of anin[1] is : <!--#echo var=anin[1] -->
<p>The value of anout[1] is : <!--#echo var=anout[1] -->
<p>The value of [myprog]myint is : <!--#echo var=[myprog]myint -->
<p>The value of [myprog]myreal is : <!--#echo var=[myprog]myreal -->
<p>The value of [myprog]mystring is : <!--#echo var=[myprog]mystring -->
<p>The value of [myprog]mybool is : <!--#echo var=[myprog]mybool -->
<p><pre>The value of [myprog]mypos is : <!--#echo var=[myprog]mypos --></pre>
<br>This is the next line.
<p>The value of numreg[1]=<!--#echo var=$numreg[1] -->
<p>The value of posreg[1,1] is <pre><!--#echo var=$posreg[1,1] --></pre>
<p>
<pre><!--#include file=md:errall.ls --></pre>
<p>
<p>
<pre><!--#exec cmd=kcl/show%20mem --></pre>
<p>
</BODY>
<META HTTP-EQUIV="PRAGMA"CONTENT="NO-CACHE">
</HTML>
```

6.5 HTTP AUTHENTICATION

6.5.1 Overview

The HTTP Authentication feature can restrict access to certain resources (files) on the robot through the web server. This capability is included with the Web Server Enhancements option. A resource can be restricted to require authentication (name/password) or can be completely locked so no access is available through the web server.

If a resource must be authenticated then the robot web server will respond with a HTTP Authentication error (401). Then the browser will prompt you for a name and password. When you provide the name and password the request is sent back to the robot with the credentials provided. If the credentials match, then 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 iPendant web browser.

The following resources can be authenticated:

- iPendant (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 6.5.2.1 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 Values: L, U, or A	This item indicates whether the resource is set to <ul style="list-style-type: none"> • Locked - No access is allowed • Unlocked - Unlimited access is allowed • Authenticate - 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.
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: <ul style="list-style-type: none"> • OPERATOR • PROGRAM • SETUP • INSTALL (default) 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.

```

HTTPAuthenticationSetupScreen(controllerpassword
option not active)
HTTP Setup
  PROTECTED RESOURCES
  Name Pwrd Resource
A ***** ipendant
A ***** KAREL:*
A ***** KCL:*
A ***** *****
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:*. item.

```

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 *iPendant* screens through the web server

The robot *iPendant* 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 *iPendant* resource. By default, all screens are read-only. To enable access, set \$UI_CONFIG.\$READONLY[2]=FALSE.

Refer to the “Advanced *iPendant* 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 *iPendant*. For the browser on the *iPendant* to be able to view web servers on the network, it needs a proxy server to proxy web requests from the *iPendant* 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 *iPendant*. 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 *iPendant* 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 *iPendant* 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 *iPendant*. 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 the [MENU] key.
2. Select Setup.
3. Press F1, [TYPE], and select HOSTCOMM. The following screen will be displayed.

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. The following screen will be displayed.

```

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 *iPendant* 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 second 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.

NOTE

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.

NOTE

Point-to-Point Protocol connectivity does not function on R-30iB Plus and R-30iB Mate Plus.

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 port.

IP Addresses

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

Table 8.2.1(a) Addresses for P2(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(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

You can configure P2 and P3 on the controller to be used as direct serial PPP connections, or you can connect an external modem to P2 and P3. Refer to Procedure 8-1 to configure 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 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 the [MENU] key.
3. Select SETUP.
4. Press F1, [TYPE].
5. Select Port Init. The following screen will be displayed.

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. The following screen will be displayed.
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 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 the [MENU] key.
3. Select SETUP.
4. Press F1, [TYPE].
5. Select port Init, and press ENTER. The following screen will be displayed.

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. The following screen will be displayed.

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 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 the [MENU] key.
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. The following screen will be displayed.

```
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**. The following screen will be displayed.

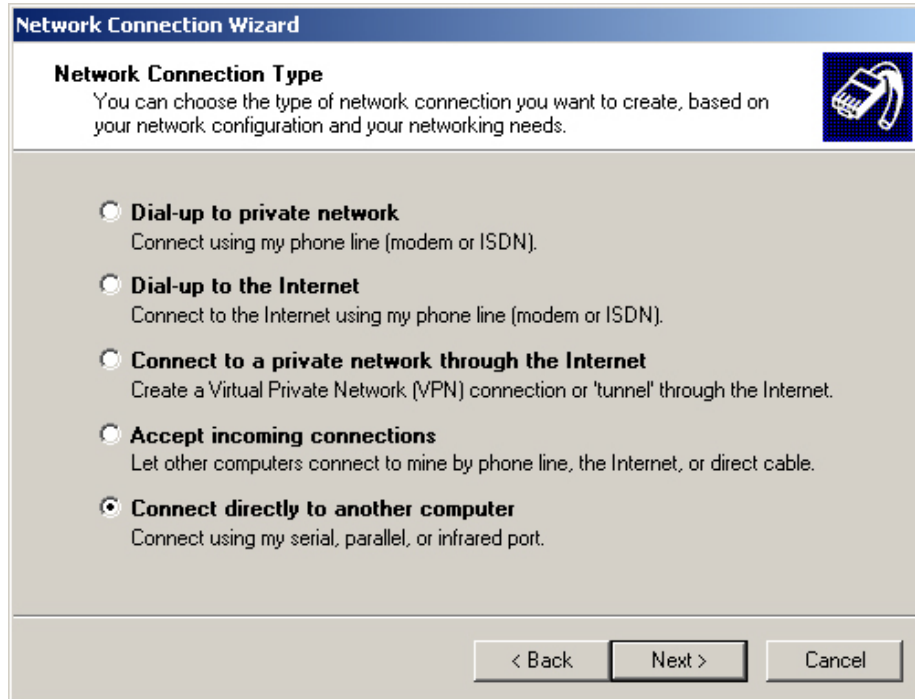


Fig. 8.3.2(a) Control Panel

2. In the Network Connection Wizard, select **Connect directly to another computer** and click Next. The following screen will be displayed.



Fig. 8.3.2(b) Network Connection

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

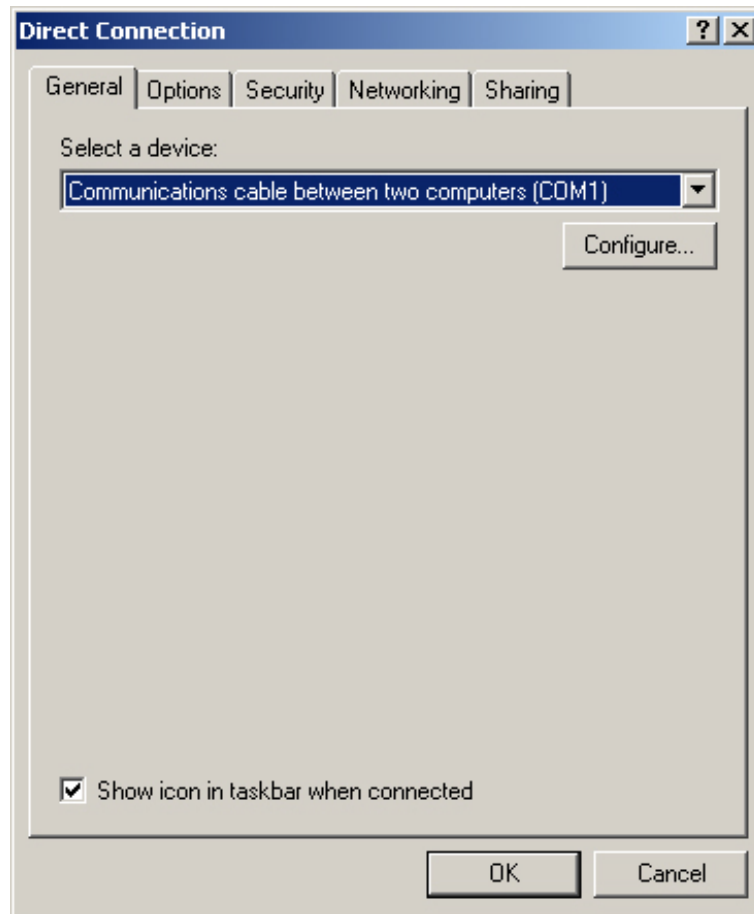


Fig. 8.3.2(c) Guest

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. The following screen will be displayed.

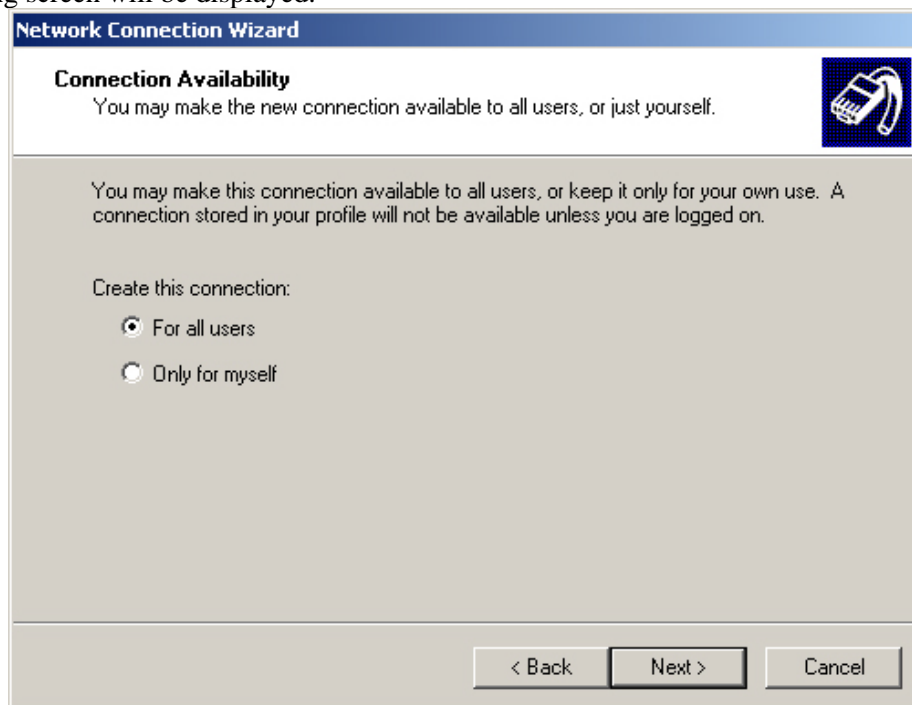


Fig. 8.3.2(d) Select a Device

5. If you want all users who log on to your PC to use this connection, select **For all users** and click Next. The following screen will be displayed.



Fig. 8.3.2(e) For All Users

6. Type a name for this connection. Select **Finish**. The following screen will be displayed.

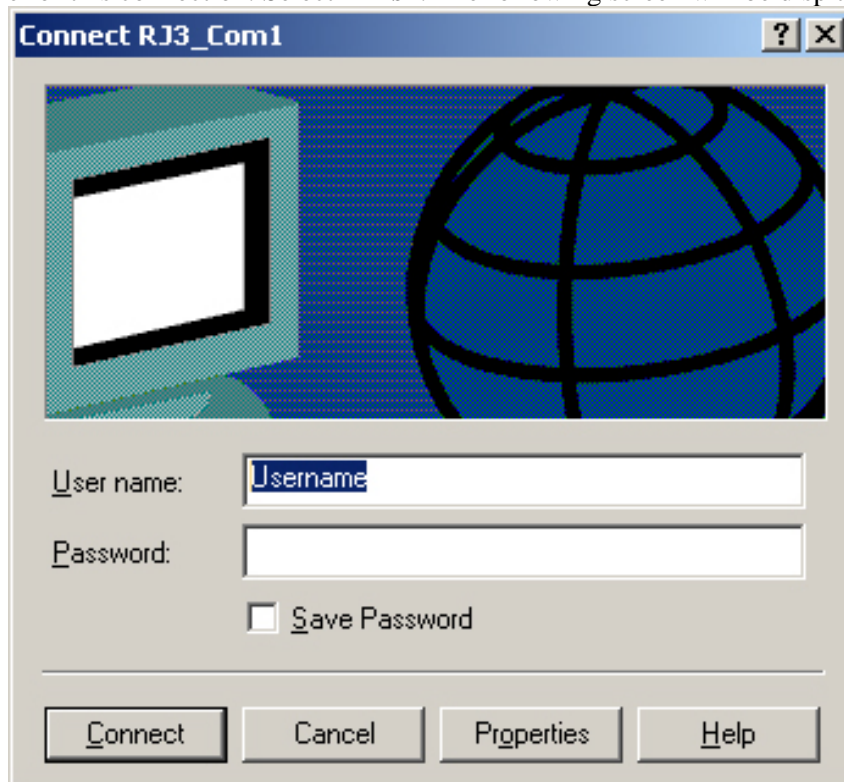


Fig. 8.3.2(f) Select Finish

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.

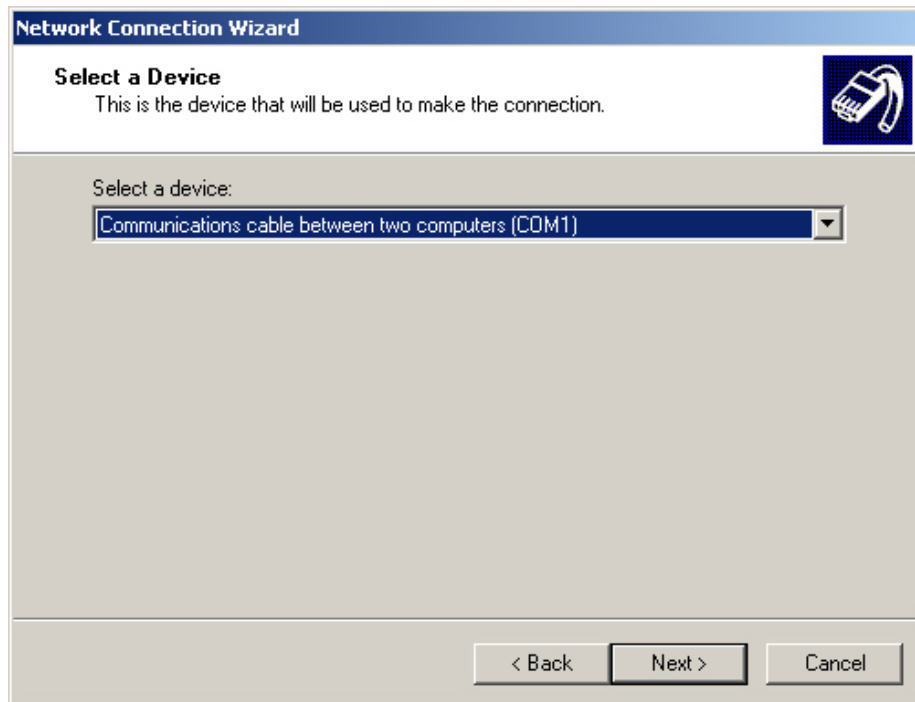


Fig. 8.3.2(g) Configure Properties

9. Click Next. The following screen will be displayed.



Fig. 8.3.2(h) Security

10. Select the **Security** tab and choose **Advanced (Custom Setting)** . Select the Settings button. The following screen will be displayed.

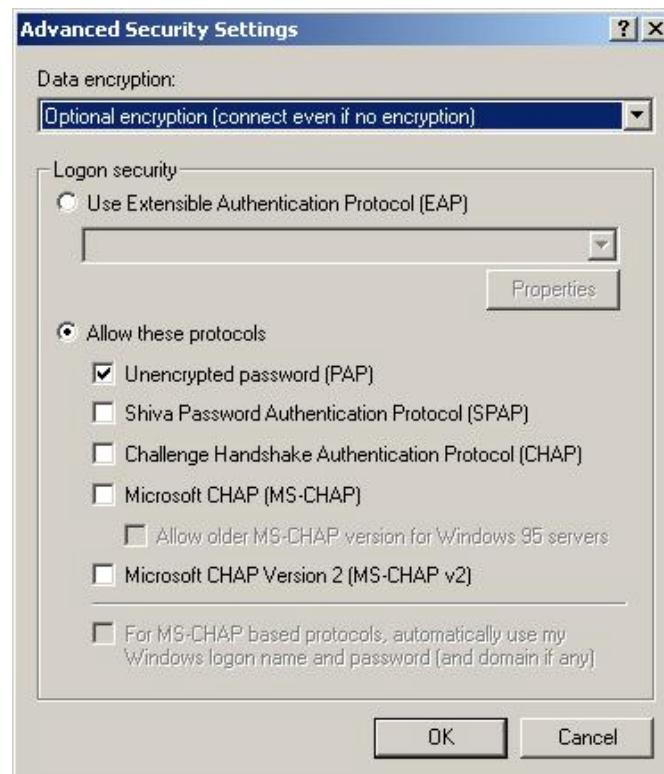


Fig. 8.3.2(i) Advanced (Custom Setting)

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. The following screen will be displayed.

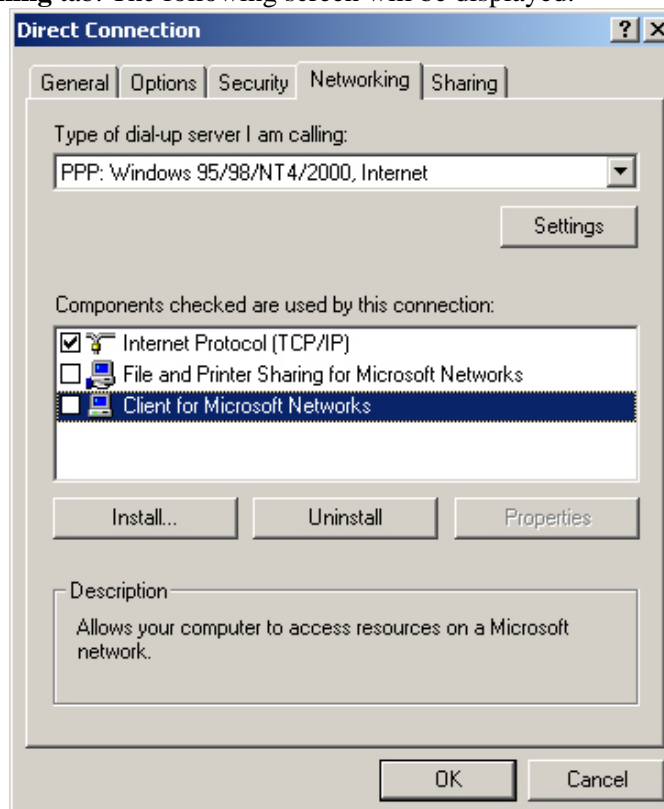


Fig. 8.3.2(j) Networking

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**. The following screen will be displayed.

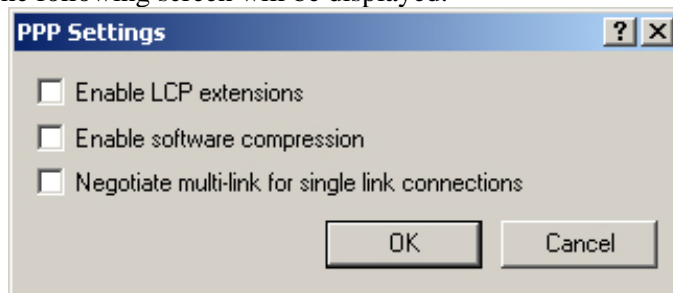


Fig. 8.3.2(k) TCP/IP Settings

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

15. Select **Internet Protocol (TCP/IP)** and choose Properties. The following screen will be displayed.

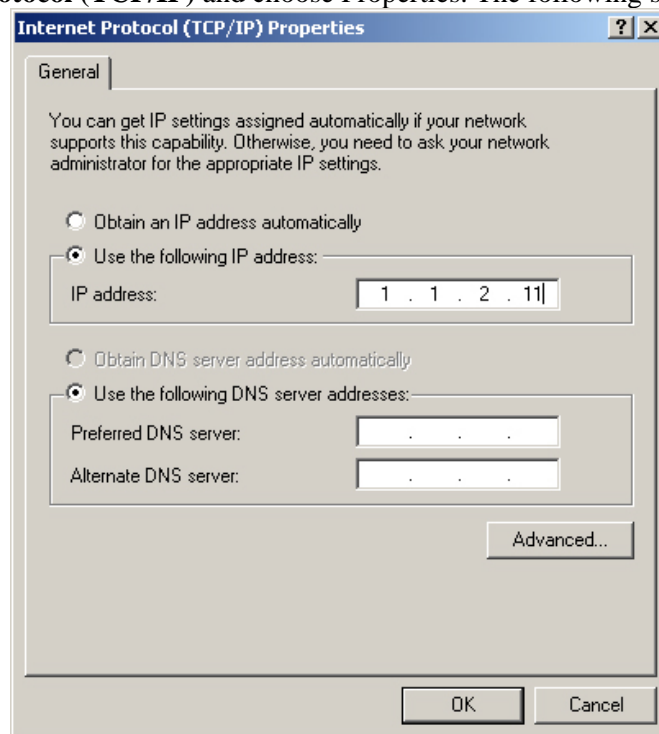


Fig. 8.3.2(l) TCP/IP Settings

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.

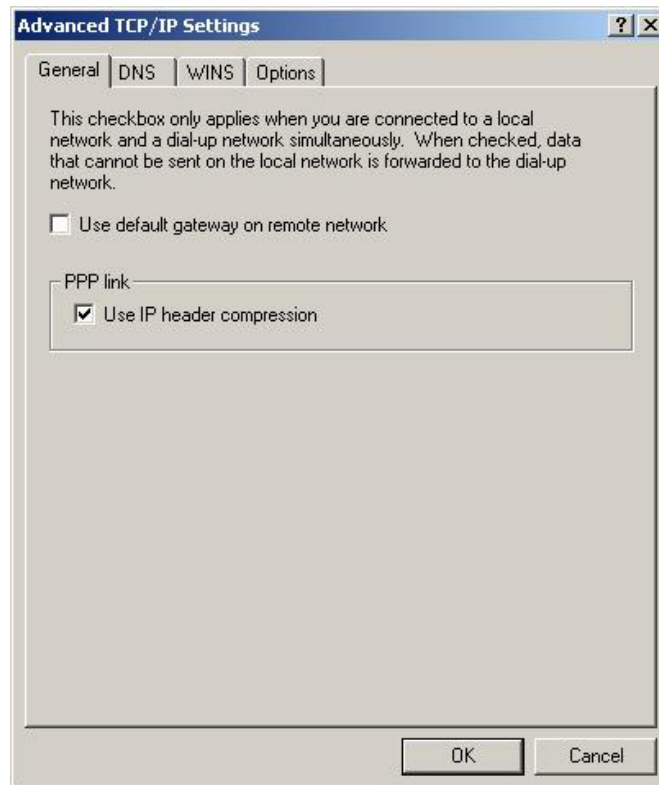


Fig. 8.3.2(m) Advanced Settings

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

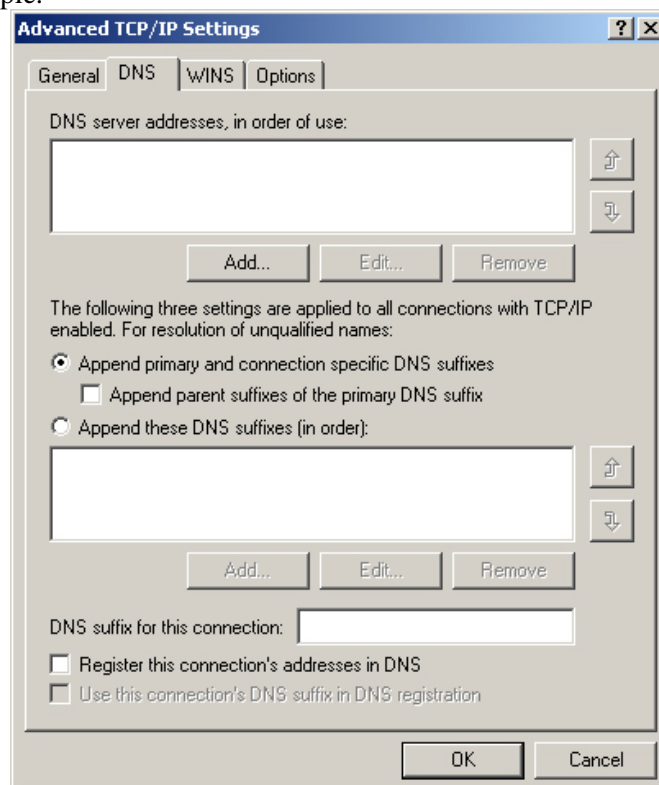


Fig. 8.3.2(n) DNS

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

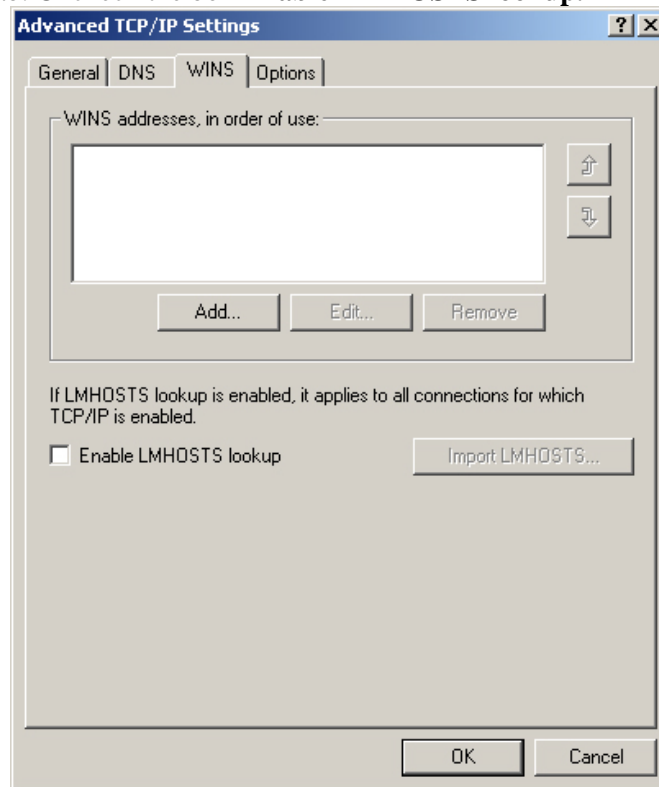


Fig. 8.3.2(o) WINS Enable LMHOSTS

20. Select the **Options** tab. The following screen will be displayed.

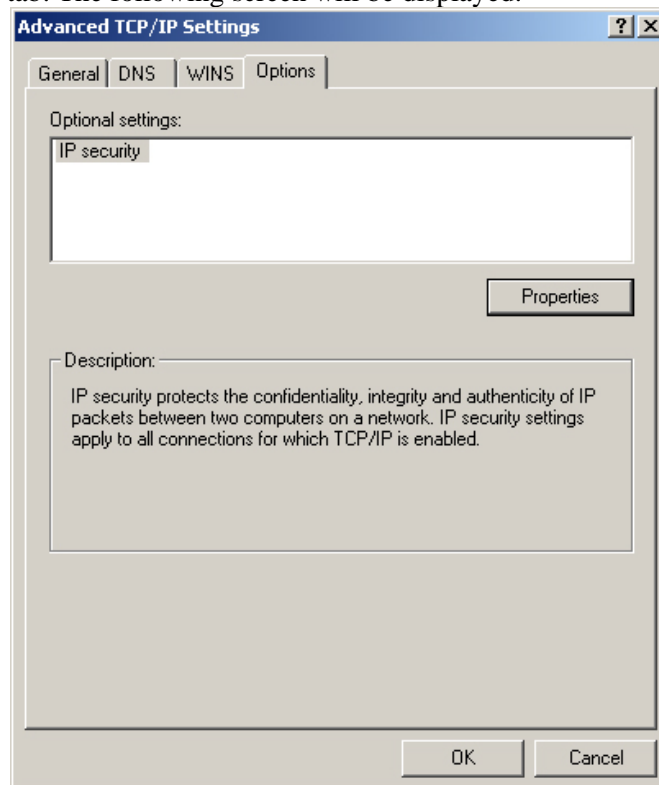


Fig. 8.3.2(p) Options

21. Choose IP Security and select the Properties button. The following screen will be displayed.



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(After SP2)

1. Select Control Panel, Network and Dial Up Connections and Create a New Connection. The following screen will be displayed.



Fig. 8.3.2(r) New Connection Wizard

2. Click Next. The following screen will be displayed.

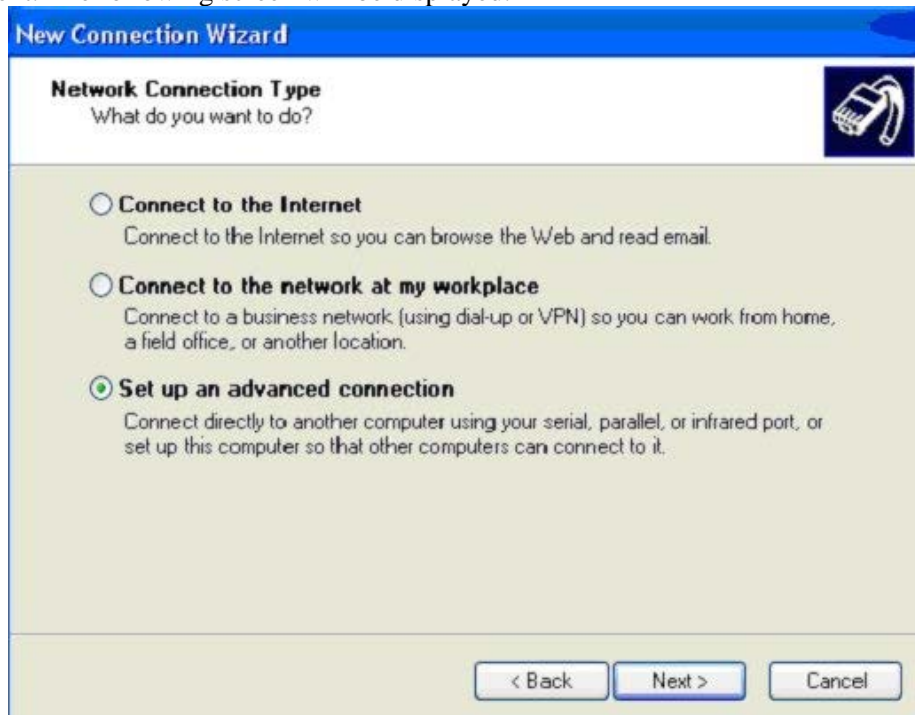


Fig. 8.3.2(s) New Connection Wizard

3. Select **Set up an advanced connection**, and click Next. The following screen will be displayed.

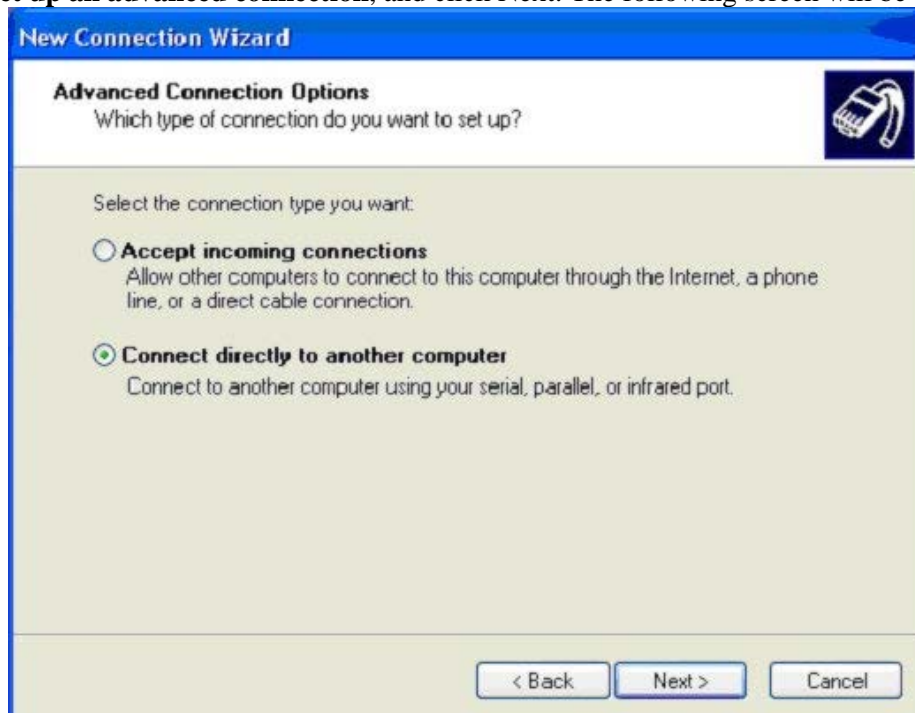


Fig. 8.3.2(t) Advanced Connection Options

4. Select **Connect directly to another computer**, and click Next. The following screen will be displayed.

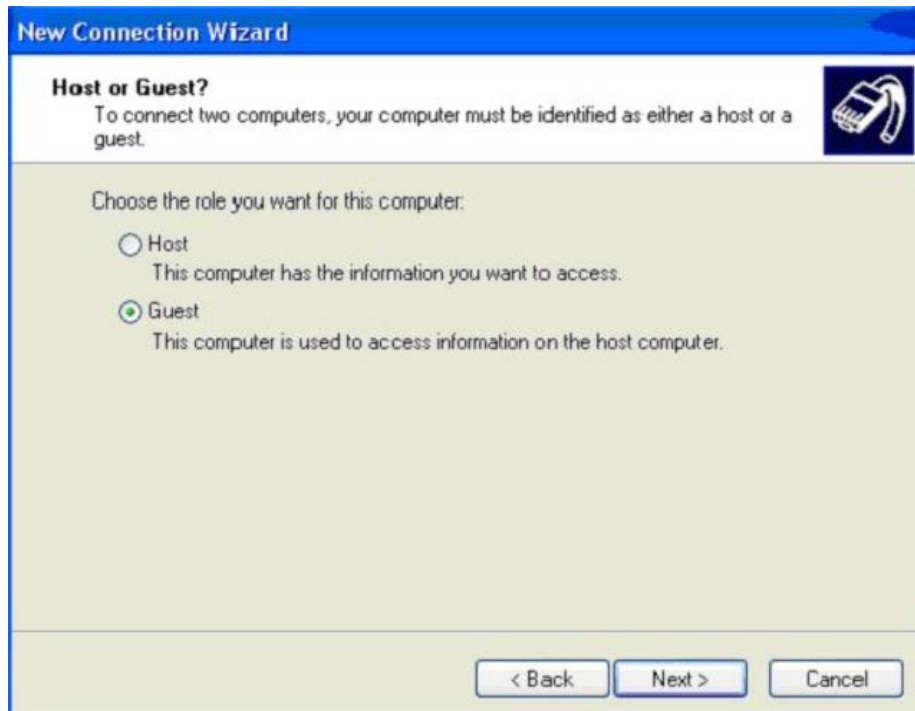


Fig. 8.3.2(u) Host or Guest

5. Select **Guest** and click Next. The following screen will be displayed.



Fig. 8.3.2(v) Connection Name

6. Type in Computer Name. For this example, R-30iA is used. Click Next and The following screen will be displayed.

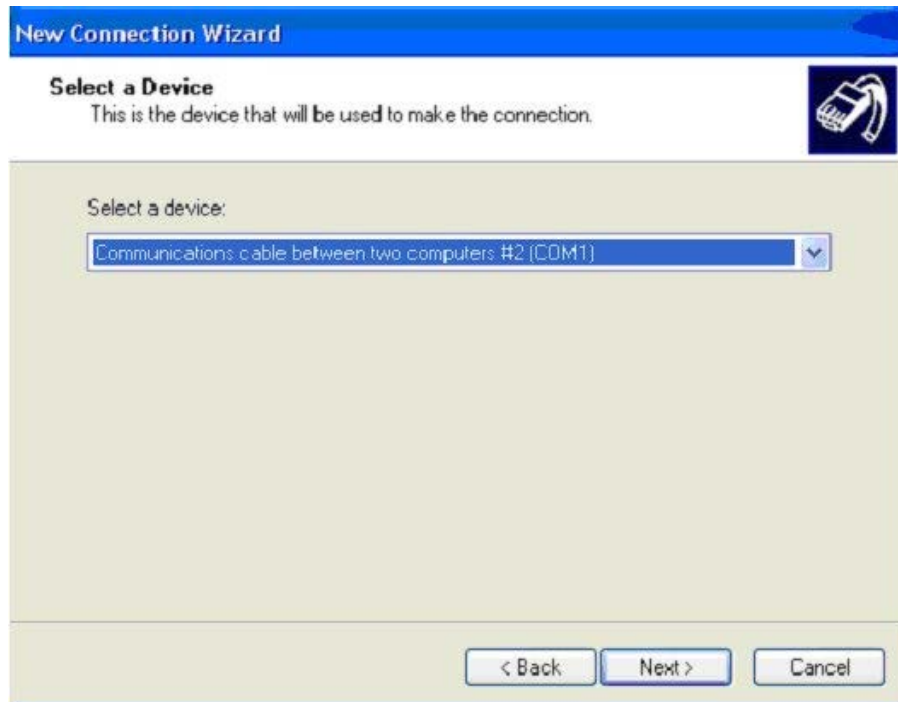


Fig. 8.3.2(w) Select a Device

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. The following screen will be displayed.



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. The following screen will be displayed.



Fig. 8.3.2(y) Completing the New Connection Wizard

9. Click Finish. The following screen will be displayed.



Fig. 8.3.2(z) Connect Screen

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

10. Click Properties. The following screen will be displayed.

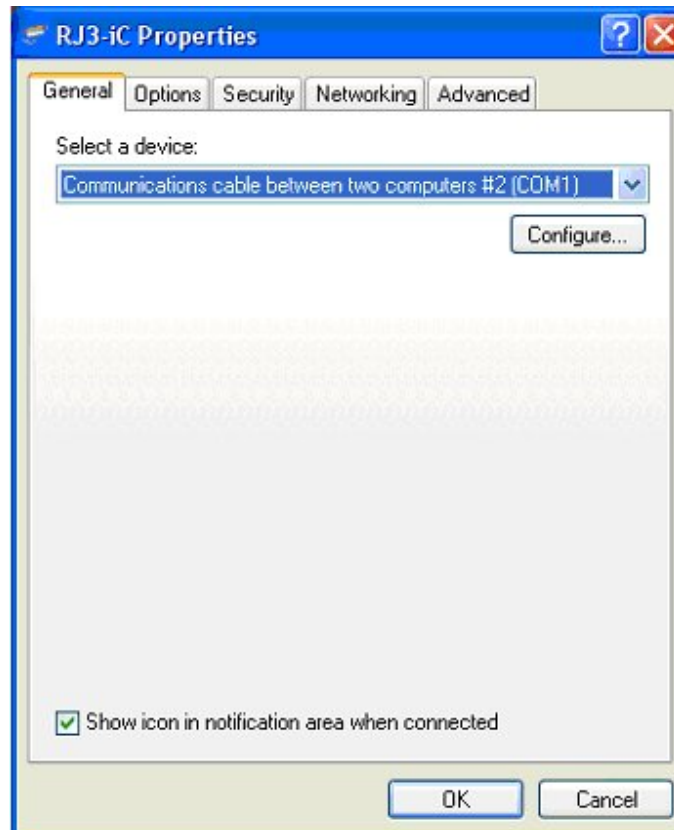


Fig. 8.3.2(aa) Properties

11. Select the Security tab and choose Advanced (Custom Setting). Click Settings. The following screen will be displayed.

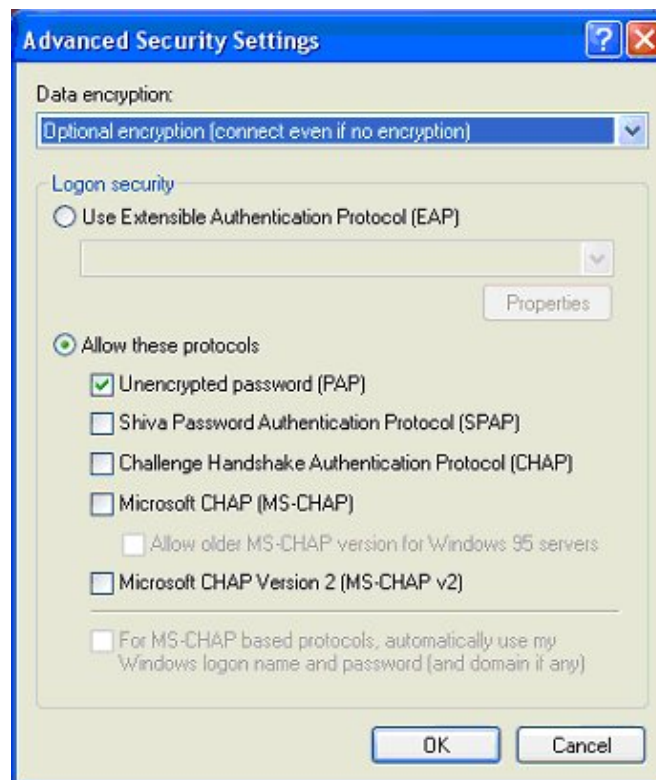


Fig. 8.3.2(ab) Advanced Security Settings

12. Select Optional Encryption and check the box for Unencrypted password (PAP). Uncheck all other boxes. Click OK.
13. Select the Networking tab. The following screen will be displayed.

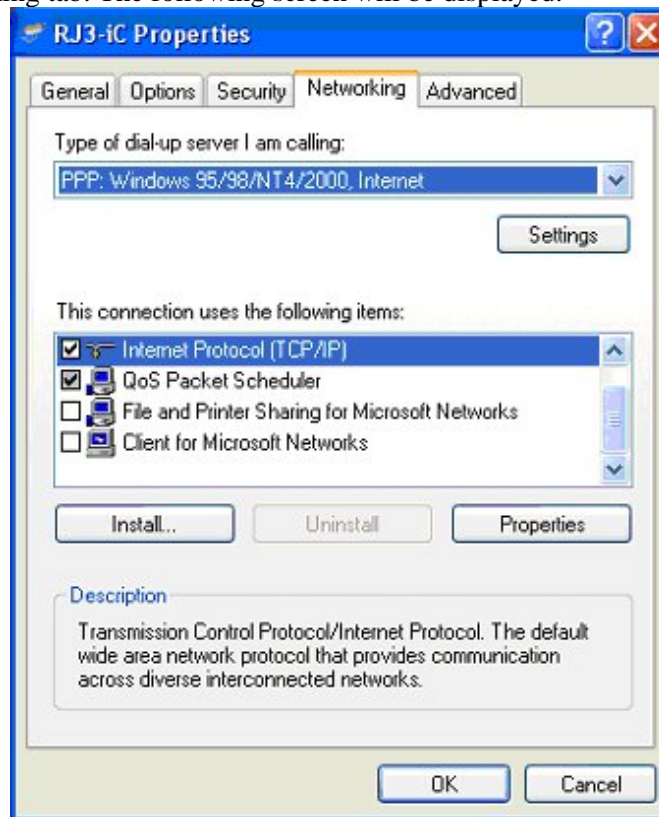


Fig. 8.3.2(ac) Properties

14. Uncheck Client for Microsoft® Networks and File and Printer Sharing. You must make sure that Internet Protocol (TCP/IP) is checked.
15. Click Settings . The following screen will be displayed.

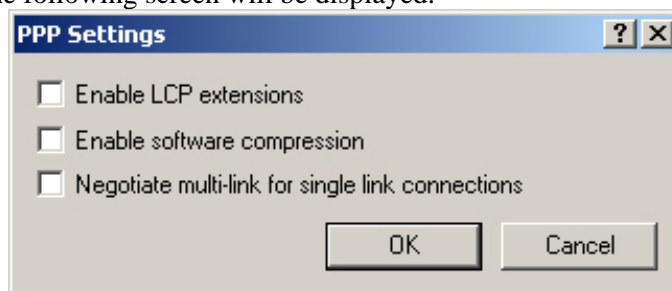
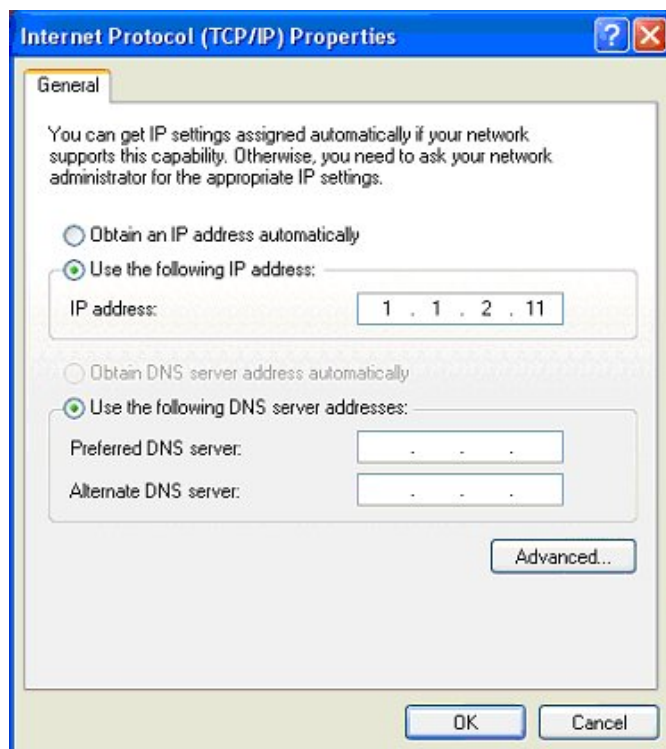


Fig. 8.3.2(ad) TCP/IP Settings

Make sure all the boxes are unchecked. Click OK.

16. Select Internet Protocol (TCP/IP), and choose Properties. The following screen will be displayed.

**Fig. 8.3.2(ae) Internet Protocol (TCP/IP) Properties**

17. Type the IP address corresponding to the serial port you are using. Leave the entries for the DNS server address blank.
18. Click Advanced. You must make sure the Use IP header compression box is checked.

**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) .

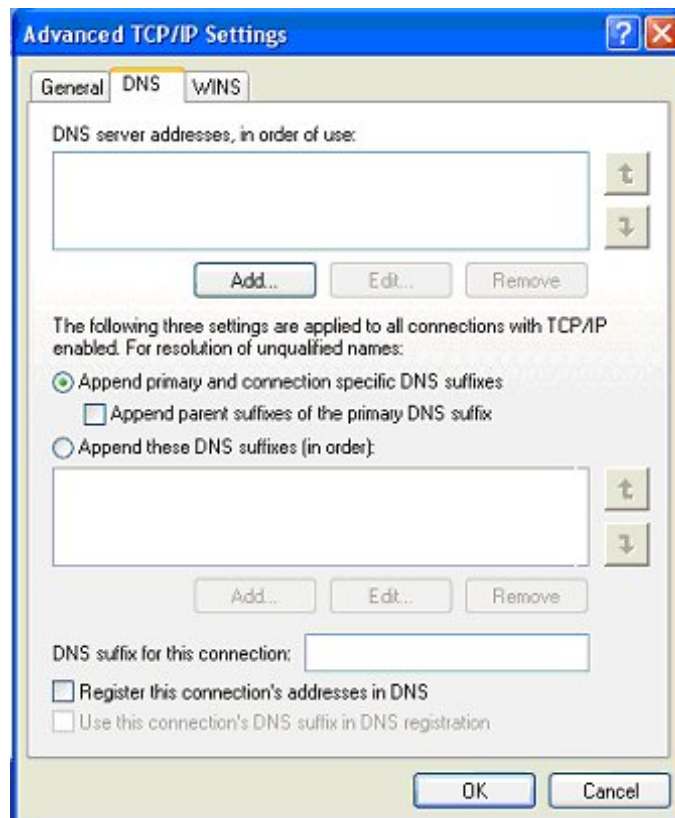


Fig. 8.3.2(ag) Advanced TCP/IP Settings

20. Select the WINS tab. The following screen will be displayed.

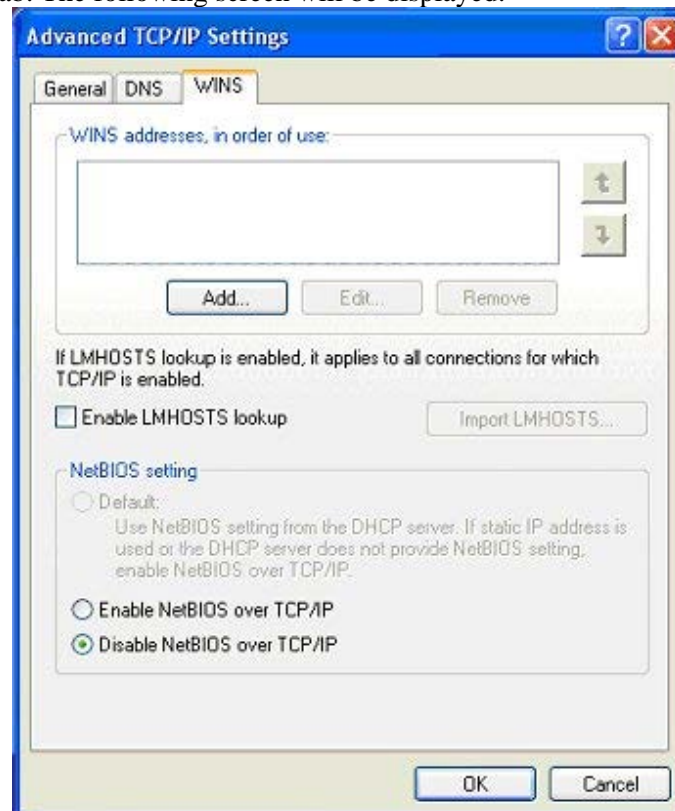


Fig. 8.3.2(ah) Advanced TCP/IP Settings

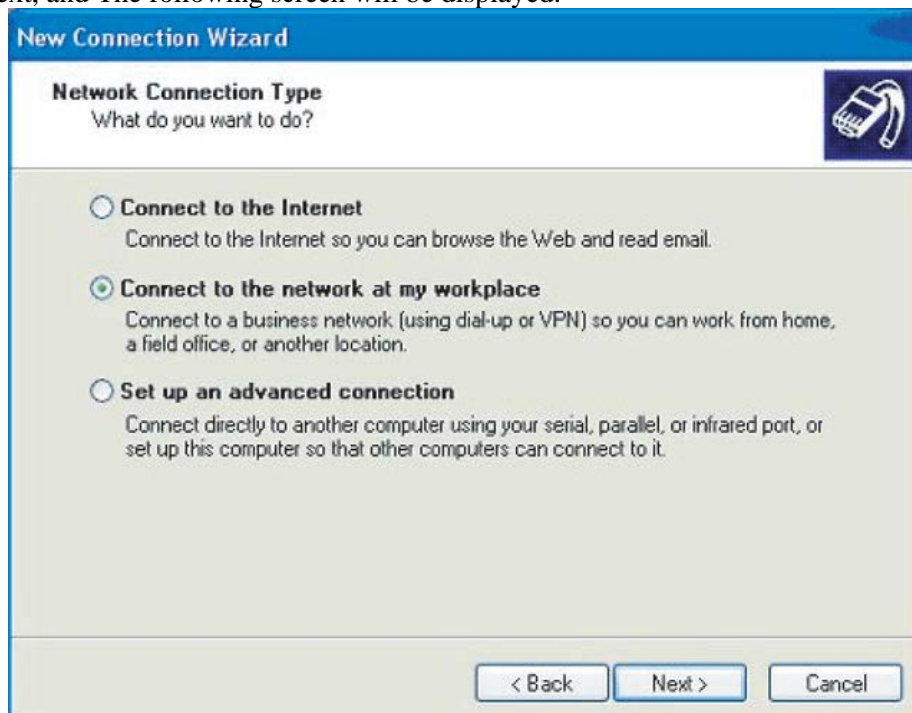
21. Uncheck the box Enable LMHOSTS lookup and check Disable NetBIOS over TCP/IP.

Procedure 8-6 Setting Up a PPP/Modem on a PC with Windows XP(After SP2)

1. Select Control Panel, Network and Dial Up Connections and Create a New Connection. The following screen will be displayed.

**Fig. 8.3.2(ai) New Connection Wizard**

2. Click Next, and The following screen will be displayed.

**Fig. 8.3.2(aj) Network Connection**

3. Click **Connect to the network at my workplace**, and click Next. The following screen will be displayed.



Fig. 8.3.2(ak) Dial Up Connection

4. Click **Dial-up connection**, and click Next. The following screen will be displayed.

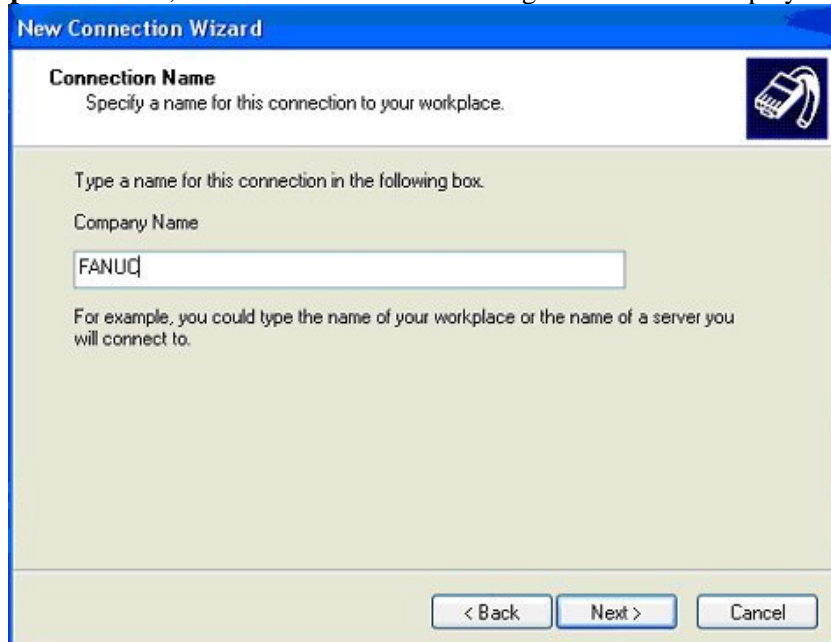


Fig. 8.3.2(al) Connection Name

5. Click Next. The following screen will be displayed.



Fig. 8.3.2(am) Phone Number to Dial

6. Type the phone number of modem where the controller is connected. Click Next. The following screen will be displayed.



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. The following screen will be displayed.



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. Press F3[DETAIL].

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 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 (MAC) of the robot, or any string identifier. If you are typing an Ethernet address (MAC), 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 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 the [MENU] key.
2. Select SETUP.
3. Press F1, [TYPE] and select Host Comm and then TCP/IP. The following screen will be displayed.

```

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). The following screen will be displayed.

```

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

```

- b. Press F3(PORT) to display Port#2 Host Comm screen. Press F2(DHCP). The following screen will be displayed.

```

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. The following screen will be displayed.

```

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 (MAC) of the robot or to any string identifier. If you are typing an Ethernet address(MAC) in this field, then the format of the Ethernet address(MAC) must be 6 bytes separated by colons. An example might include 00:E0:E4:F7:94:AC

NOTE

The Ethernet address(MAC) 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 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(MAC) or to supply a string to the server. To use the Ethernet address(MAC), 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 NEXT and then pressing F2[DIAG] 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 or UDP 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 or UDP 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 or UDP 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 or UDP 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 socket messaging, 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. On the teach pendant, press the [MENU] key.
2. Select 6. SETUP.
3. Press F1, [TYPE].
4. Select Host Comm.
5. Press F4, [SHOW].
6. Choose Servers.
7. 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:

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

8. Move the cursor to Protocol, and press F4, [CHOICE].
9. Select SM.
10. Move the cursor to Startup State, and press F4, [CHOICE].
11. Select START.
12. Press F2, [ACTION].
13. Select DEFINE.
14. Press F2, [ACTION].
15. Select START.
16. Set the system variable:

- a. Press the [MENU] key.
- b. Select 0 – NEXT --.
- c. Select 6. 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. The following screen will be displayed.

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	\$REFERRS	FALSE
9	\$TIMEOUT	15
10	\$PATH	*uninit*
11	\$STRT_PATH	*uninit*
12	\$STRT_REMOTE	*uninit*
13	\$USERNAME	*uninit*
14	\$PWD_TIMEOUT	0
15	\$SERVER_PORT	0

- h. Move the cursor to \$SERVER_PORT. Type in the positive number of the TCP/IP or UDP port you want to use for socket messaging. The server tag is now ready to use from a KAREL program.
- i. If you want to communicate with UDP, set \$USE_UDP to 1.
Set ATR_BUFSIZE or 23 to 2nd argument of SET_FILE_ATR, which is KAREL built-in, and set the size you want to send in 1 record to 3rd argument, then you can change the size sent in 1 record. Default setting is 128.

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 socket messaging, 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. On the teach pendant, press the [MENU] key.
2. Select 6. SETUP.
3. Press F1, [TYPE].
4. Select Host Comm.
5. Press F4, [SHOW].
6. Choose Clients.
7. 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:

Comment:          *****
Protocol:         *****
Current State:    UNDEFINED
Startup State:
Server IP/Hostname: *****
Remote Path/Share: *****
Inactivity Timeout: 15 min
Username:         anonymous
Password:         *****

```

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

NOTE

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

14. Set the system variable:
 - a. Press the [MENU] key.
 - b. Select 0 – NEXT --.
 - c. Select 6. 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 C2, move the cursor structure element [2], 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. The following screen will be displayed.

SYSTEM Variables		
\$HOSTC_CFG[2]		
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	\$PWD_TIMEOUT	0
15	\$SERVER_PORT	0

- h. Move the cursor to \$SERVER_PORT. Type in the positive number of the TCP/IP or UDP server port you want to use for socket messaging. The client tag is now ready to use from a KAREL program.
- i. If you want to communicate with UDP, set \$USE_UDP to 1.
Set ATR_BUFSIZE or 23 to 2nd argument of SET_FILE_ATR, which is KAREL built-in, and set the size you want to send in 1 record to 3rd argument, then you can change the size sent in 1 record. Default setting is 128.

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_CONNECT(string, integer)

MSG_CONNECT 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_CONNECT.

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 must be used to close the connection at the client side before MSG_CONNECT can be used to establish another connection to the remote server.

The first parameter of this command contains the tag name (e.g. "S1:") and the second parameter is an integer value that indicates the status of closing the connection on the client side.

10.4.4 MSG_PING(string, integer)

MSG_PING is a utility command used to check network connections with a remote host, so that you can determine if it is currently connected to the network. The MSG_PING command sends ping packets to the remote host and waits for a reply. (Ping packets are chunks of data that are transferred between hosts on a network.) If there is no reply from the remote host, this usually means that you will not be able to use other network protocols like FTP, TELNET, or Socket Messaging to connect to that host. If you have attempted without success to use Socket Messaging to connect to a remote host, the MSG-PING utility is a good place to start in trying to diagnose the problem.

The first parameter of this command contains the name of the remote host to ping. If you are not using DNS on your network, the host name and IP address of the remote host will have to exist in the Host Entry table.

10.4.5 Exchanging Data during a Socket Messaging Connection

After you have successfully established a socket messaging connection, you can use KAREL commands to exchange data between connected devices. KAREL has several commands that can be used for data exchange operations:

- OPEN FILE
- WRITE
- READ
- BYTES_AHEAD

10.5 NETWORK PERFORMANCE

10.5.1 Overview

Performance of socket messaging on your network will vary depending upon the number of devices connected to the network, the number of applications being run on the controller, the network cabling configuration, and number of *hops* that the message will have to make to reach its destination device.

NOTE

Hops is a term that indicates the number of routers between the source host and destination host. In general, the fewer the number of hops the data makes from router to router, the faster the data is transmitted between the source and destination hosts. Data transfer is fastest between hosts on the same network.

10.5.2 Guidelines for a Good Implementation

Use the following guidelines when implementing a solution for any application using socket messaging.

- You must not transfer small data separately, instead gather the data and transfer as a larger packet. This applies to both read and write.
- You must understand that the rate of data does not flood the remote side.
- You must understand that there are other nodes on the Ethernet network so performance cannot be guaranteed.
- You must understand that other applications on robot also use TCP/IP or UDP and performance can be affected.

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 klevccdf
%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)
            ENDIF
        ENDWHILE
    ENDIF
```

```

                                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:writen 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.

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 the [MENU] key.
- 2. Select System.
- 3. Select Clock. The following screen will be displayed.

Clock Display

DATE

04/04/05

TIME

13:50:30

Please select function

NOTE

ADV appears in Clock screen if SNTP client is installed.

4. Press F2, ADV. The following screen will be displayed.

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 Starts: Ends:	First Sunday in April at 2 am local standard time Last Sunday in October at 3 am daylight saving time
GMT-08:00 PST (US) Starts: Ends:	First Sunday in April at 2 am local standard time Last Sunday in October at 3 am daylight saving time
GMT-07:00 MST (US) Starts: Ends:	First Sunday in April at 2 am local standard time Last Sunday in October at 3 am daylight saving time
GMT-06:00 CST (US) Starts: Ends:	First Sunday in April at 2 am local standard time Last Sunday in October at 3 am daylight saving time
GMT-05:00 EST (US) Starts: Ends:	First Sunday in April at 2 am local standard time Last Sunday in October at 3 am daylight saving time
GMT-04:00 AST (CAN) Starts: Ends:	First Sunday in April at 2 am local standard time 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 Starts: Ends:	Last Sunday in March at 1 am GMT (UTC) time Last Sunday in October at 1 am GMT (UTC) time

Timezone	Current DST Policies
GMT-00:00 London Starts: Ends:	Last Sunday in March at 1 am GMT (UTC) time Last Sunday in October at 1 am GMT (UTC) time
GMT+01:00 Berlin Starts: Ends:	Last Sunday in March at 1 am GMT (UTC) time Last Sunday in October at 1 am GMT (UTC) time
GMT+02:00 Athens Starts: Ends:	Last Sunday in March at 1 am GMT (UTC) time Last Sunday in October at 1 am GMT (UTC) time
GMT+03:00 Moscow Starts: Ends:	First Sunday in April at 2 am local standard time Last Sunday in October at 3 am daylight saving time
GMT+04:00 Baku Starts: Ends:	First Sunday in April at 2 am local standard time 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 Starts: Ends:	First Sunday in April at 2 am local standard time Last Sunday in October at 3 am daylight saving time
GMT+11:00 Noumea	None
GMT+12:00 Auckland Starts: Ends:	First Sunday in April at 2 am local standard time Last Sunday in October at 3 am daylight saving time
GMT+13:00 Nukualofa	None
CUSTOM Starts: Ends:	The user set these fields 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.

SNTP packets may not be routed depending on router configuration when you use SNTP over the router. Your network administrator can provide you the correct router configuration.

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 Default: OFFLINE	The status of the sniffer. Possible values are ONLINE, and OFFLINE. This field is not editable.
Packet Count 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 Default: NONE	The name of the filter being used. Only packets matching the filter are captured.
Trigger Default: NONE	The name of the trigger being used. When the trigger is matched, the capture is stopped.
Interface Default: Port1	The name of the interface being used. Possible values are Port1 and Port2.
Dump file Default fr:sniff.cap	The location where the Ethernet capture file is to be saved.
Buffer Size Default: 5000	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 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 the [MENU] key.
2. Select SETUP.
3. Press F1 [TYPE] and select Host Comm.
4. Under Protocol, move the cursor to SNIFF and press F3, [DETAIL]. The following screen will be displayed:

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 :	5000
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 5000 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 iRVision.

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.

13. ROS INTERFACE PACKETS OVER ETHERNET (RIPE)

ITEM	DESCRIPTION
Displayed on Master Screen Only	
Master IP Address	<p>This indicates the IP address of the master.</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p>⚠ CAUTION</p> <p>The master IP address must be the same for all robots in the ring. Otherwise, communications will not work properly.</p> </div> <p>If the master IP address is set incorrectly, move the cursor to the Master IP address and type in the correct address.</p>
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 the [MENU] key.
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 ROSIPCONFIG.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 applications which use RIPE internally. The RIPE setup is required for these applications:

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

These applications require multicasting or broadcasting for sending messages to multiple controllers. In this case the multi-cast address is based on the RIPE master address. All of the RIPE slaves will join this address.

In the case of just two robots RIPE uses unicast. This simplifies the Network setup. RIPE supports broadcast based on the non-default setting of system variables. It is recommended that multicast be used since any application that works for broadcast will also work for multicast. Multicast is much more flexible approach. See section Section 13.9 for details on network configuration.

13.5 XML CONFIGURATION FILE

The objective of the ROSIPCFG.XML file determines the name and IP addresses of all the robots in the RING. It must be named rosipcfg.xml, and it must be installed on FRS. It is read at every powerup. In the following example, MHROB01 is the master. The ring has two slaves, MHROB02 and MHROB03.

```
<?xml version=" 1.0" ?>
<!-- Multi-Robot configuration file - >
<ROSIPCFG>
<ROBOTRING count=" 3" timeslot=" 400" >
<MEMBER name=" MHROB01" ipadd=" 192.168.0.101" />
<MEMBER name=" MHROB02" ipadd=" 192.168.0.102" />
<MEMBER name=" MHROB03" ipadd=" 192.168.0.103" />
</ROBOTRING>
</ROSIPCFG>
```

This line should be at the start of your XML file:

```
<?xml version=" 1.0" ?>
```

This is a comment. All comments are included within <!-- -->:

```
<!-- Multi-Robot configuration file - >
```

This must be the first tag. Only one of these can exist in your file:

```
<ROSIPCFG>
```

This starts the robot ring definition. Only one of these can exist in your file:

```
<ROBOTRING count=" 3" timeslot=" 400" >
```

This tag defines the member and can contain several attributes:

```
<MEMBER name=" mhrob02"  ipadd=" 192.168.0.102"  focusbg="8388863" />
```

This ends the robot ring definition. Only one of these can exist in your file:

```
</ROBOTRING>
```

13.6 TELNET

For the case of MHROB01 with two slaves and one Ethernet connection for all three robots you can access telnet on the slaves if RIPE is configured.

```
MHROB02 129.168.0.102  --> telnet Port # 2302
MHROB03 129.168.0.103  --> telnet Port # 2303
```

The port number is based on the Member Index. Therefore, if you have 16 robots telnet can connect to the 16th robot via port 2316.

13.7 VARIABLE ACCESS

The program name for GET_VAR and SET_VAR is expanded to optionally include the name of the RING member. So to access \$TP_DEFPROG on MHROB03 the GET_VAR syntax is:

```
GET_VAR(entry,  '//MHROB03/*system*' ,  '$TP_DEFPROG' ,  strvar, status)
```

13.8 SYNCHRONIZED TIMING

In conjunction with this the system variable \$PH_ROSIP.\$NETTICK OFF is provided. This variable is automatically updated by the system. This value provides the offset of \$FAST_CLOCK to that of the master robot.

This means that on all robots the exact same value on any robot can be obtained by added \$FAST_CLOCK and \$PH_ROSIP.\$NETTICK OFF at any instant in time. This is the key to tightly synchronizing the members of the ring.

13.9 NETWORK DESIGN CONSIDERATIONS

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. M12 4-pin D-coded Connectors are included in the Ethernet/IP specification.
- 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 multicast aware switches (support for IGMP snooping).
- 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 LEDs near the RJ45 connector on the robot will confirm a connection link (refer to the appendix titled “Diagnostic Information” in the *Internet Options Setup and Operations Manual* for details on the LEDs). Link State can be confirmed using the TCP/IP status Host Comm screen by following Procedure 13-2.

Procedure 13-2 Verifying Link State

1. Press the [MENU] key.
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. We 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
- Hirschmann 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). This "connection" to the Shared Folder allows the robot controller to browse the remote directory tree, and to read and write files remotely. Robot is able to map a client tag to a directory location on the 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: <ul style="list-style-type: none"> ● UNDEFINED - the device is not defined. ● DEFINED - the device is defined. ● 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. <ul style="list-style-type: none"> ● When set to zero, no timeouts occur. ● 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 the [MENU] key.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm. The following screen will be displayed.

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. The following screen will be displayed.

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.

```

SETUP Tags
Tag C1:
Comment:      *****
Protocol Name: PC Share
Current State: DEFINED
Startup State: DEFINE
* Server IP/Hostname: 192.168.1.49
Remote Path/Share:  robot¥programs¥
Inactivity Timeout: 0 min
Username:      Gary
Password      *****

```

* An IP Address in dotted-decimal notation (as shown) or a hostname can be entered. When using a hostname, the hostname must either be entered into the robot controller host table, DNS must be configured, or a WINS server must be configured.

8. 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.
9. Move the cursor to Protocol Name and press F4, [CHOICE]. A list of available protocol choices will be displayed.
10. Select PC Share and press ENTER
11. Move the cursor to Startup State and press F4, [CHOICE].
12. 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.

13. 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 or a WINS server is used.
14. Move the cursor to Remote Path/Share field and use the function keys to enter the name of the remote "Share" to connect to, optionally followed by a backslash (¥) and a path existing on the Share. For example, if connecting to the Share "robot", enter "robot". If connecting to the "programs" subdirectory in the Share "robot", enter "robot¥programs".
This field may be left blank to get a listing of all available Shares on the remote PC; however, file operations cannot be performed unless a Share is specified.
15. Move the cursor to Inactivity Timeout, type the timeout value you want in minutes, and press ENTER. Please set 0 for PC share.
16. Move the cursor to the Username field, and type in the username for the client to use to log into the remote server.
17. Move the cursor to the Password field, and type in the password for the client to use to log into the remote server.
18. Press F3, LIST, to display the list of client devices.
19. To define and start PC Share on a device:
 - a. Press F4, [SHOW].
 - b. Select 2, Clients.
 - 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 3, START.

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 (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 Discovery (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 the [MENU] key.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm. The following screen will be displayed.

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. The following screen will be displayed.

```

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, cursor 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 *iR*Vision, execution time of *iR*Vision will change significantly or slower by saving the log file. Please do not save the log file of *iR*Vision 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 *iR*Vision 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 the [MENU] key.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm.
5. Press F4, [SHOW].
6. Select Clients. The following screen will be displayed.

```

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 is 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 Pendant 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.

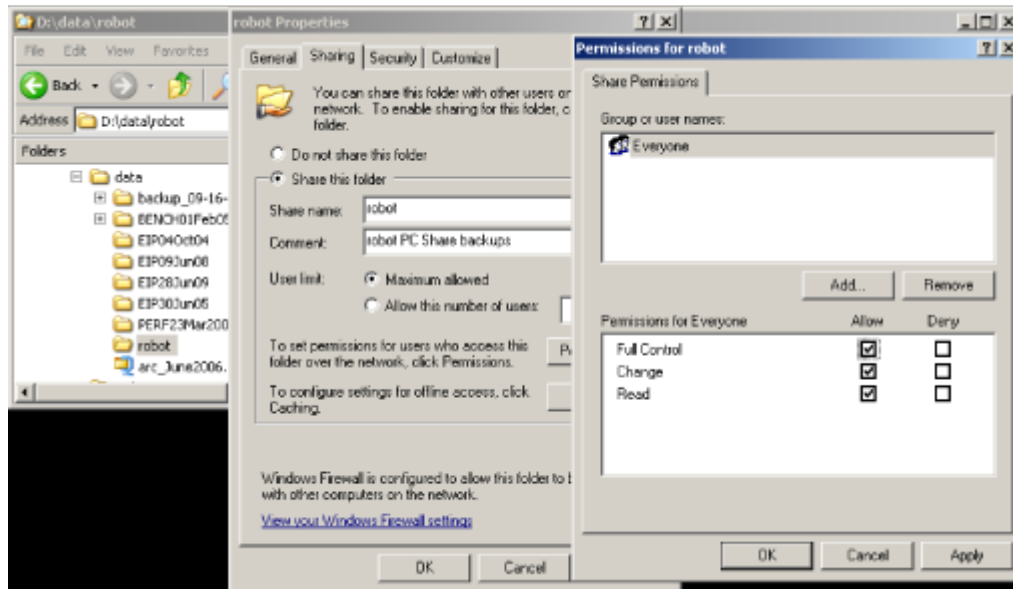


Fig. 14.5(a) Sharing a Folder in Windows® XP

Procedure 14-5 Sharing a folder in Windows 7®

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.

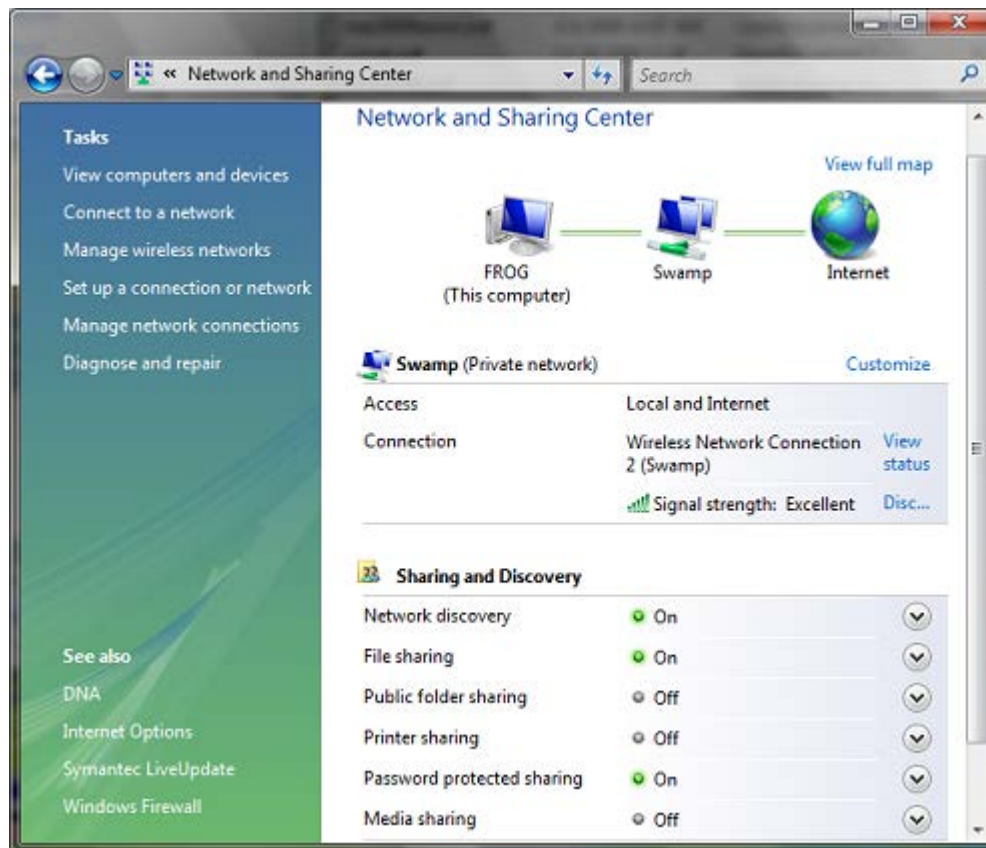


Fig. 14.5(b) Microsoft Vista® Network and Sharing Center

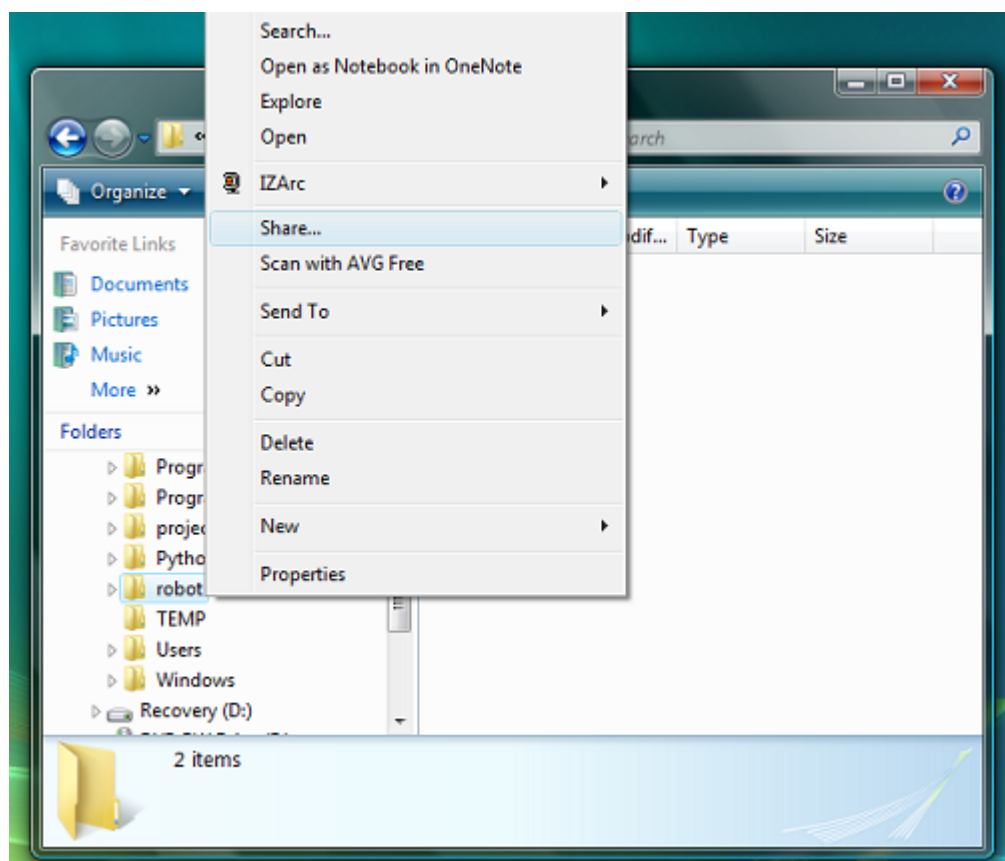


Fig. 14.5(c) Selecting Share... in Windows Vista®

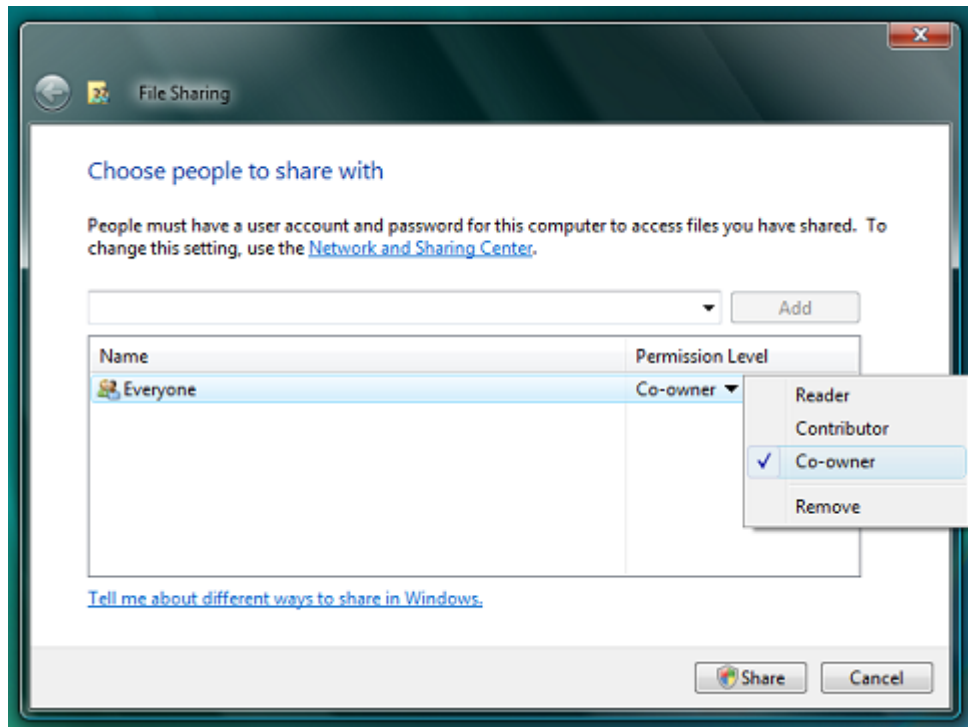


Fig. 14.5(d) Choose People to Share With in Microsoft Vista®

15 ADVANCED iPendant FUNCTIONS

15.1 OVERVIEW

This chapter provides information about the advanced connectivity and customization features of the iPendant. These include:

- **Monitor iPendant function** This allows you to monitor the iPendant screens remotely using a PC and Microsoft® Internet Explorer. This screen follows real iPendant screen.
- **Navigate iPendant function** This allows you to display an iPendant-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.
- **Jogging iPendant function** This allows you to Navigate iPendant function. And this function also allows you to jogging and TP program execution by PC Browser.

15.2 SETUP

The following are the requirements for advanced iPendant functions.

- The PC must have Microsoft® Internet Explorer 5.5 or greater installed.
- The PC must have the iPendant Controls installed. Refer to Section 15.2.1 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 iPendant connected.
- The robot controller must be connected to a network and be properly configured for Network access to the above PC.

15.2.1 Installation of iPendant Controls

Advanced iPendant functions require you to have the FANUC Robotics iPendant Controls loaded on the PC that you will be using. The manual for iPendant 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.

15.2.2 Configuring Internet Explorer ®

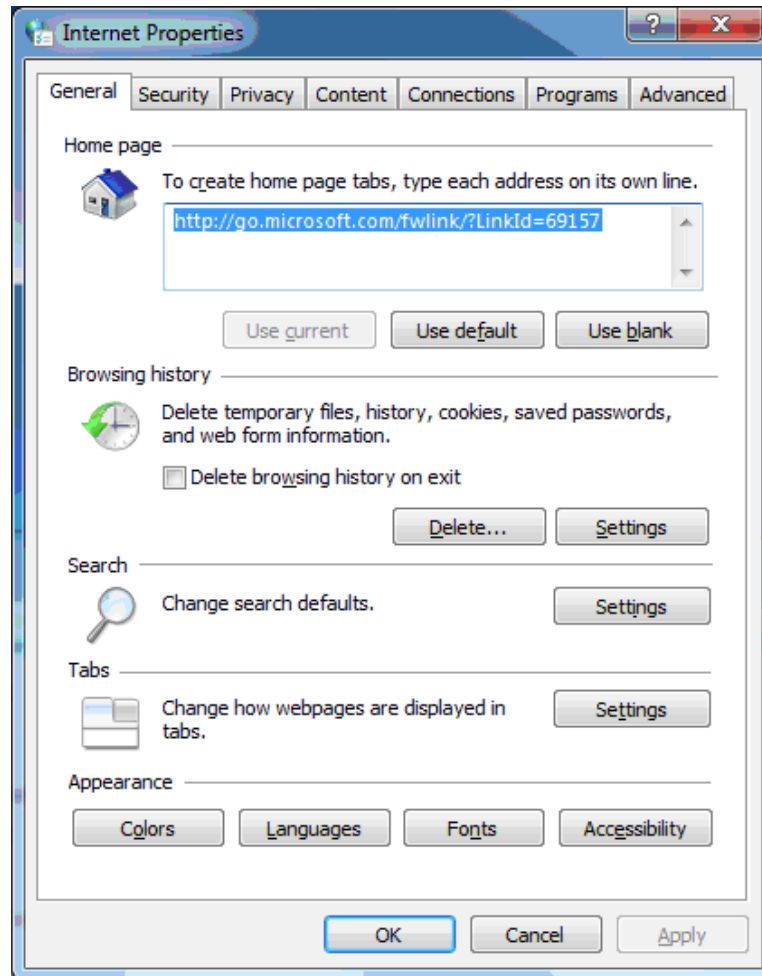
The following settings are required in Microsoft® Internet Explorer for proper operation of the Advanced iPendant function.

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. 15.2.2(a) Internet Properties**

2. Trusted Sites:

- a. Select the [Security] tab.

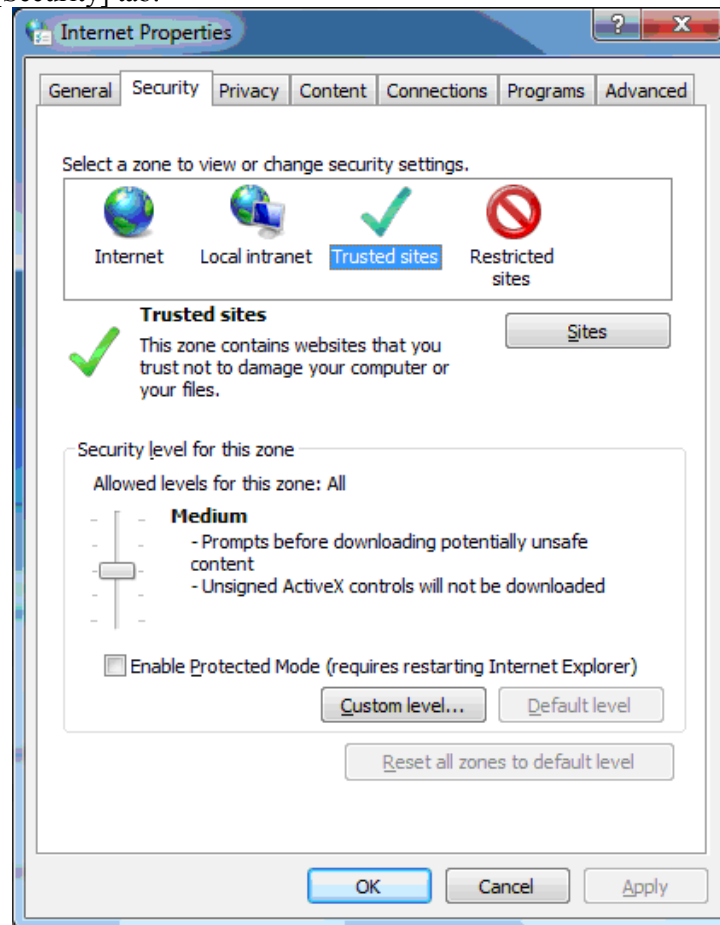


Fig. 15.2.2(b) Security

- b. Select [Trusted Site], and then click the [Sites] button.

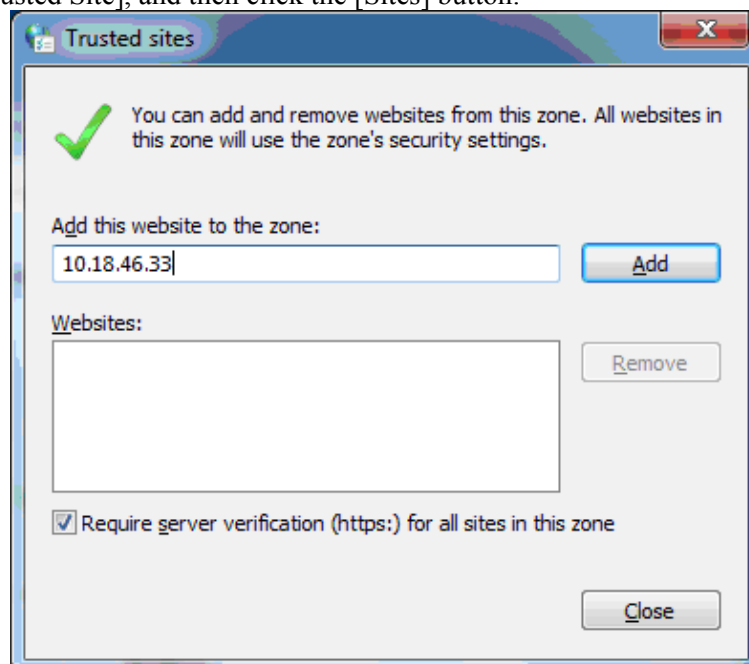


Fig. 15.2.2(c) Trusted Sites

- 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 two digits 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.

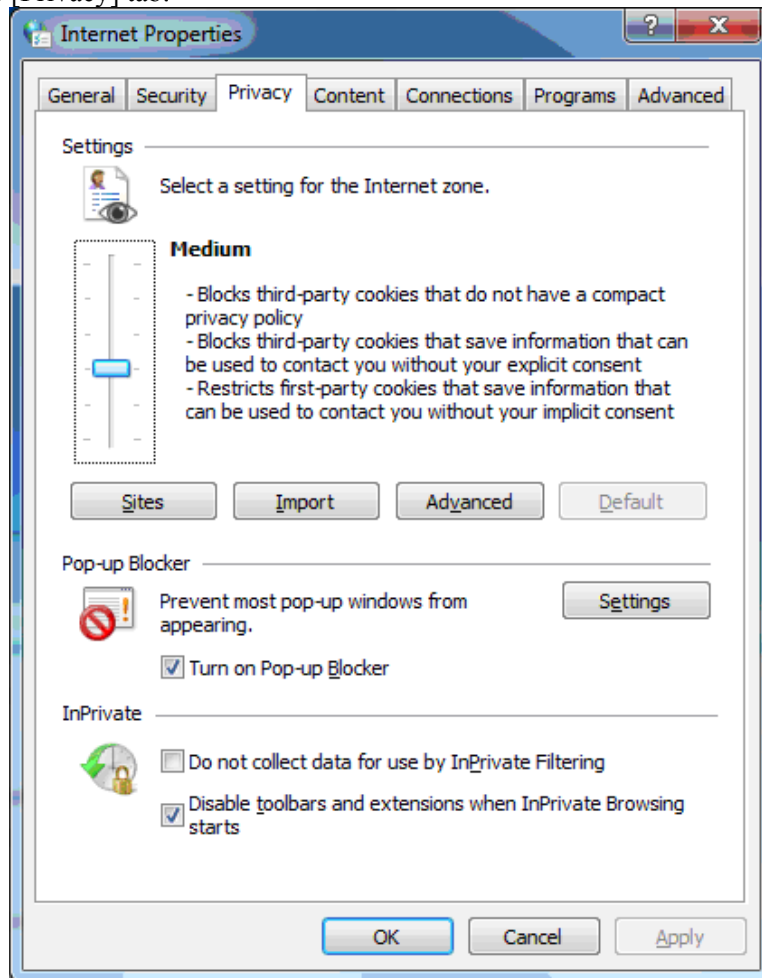


Fig. 15.2.2(d) Privacy

- b. Click the [Settings] button of [Pop-up Blocker].

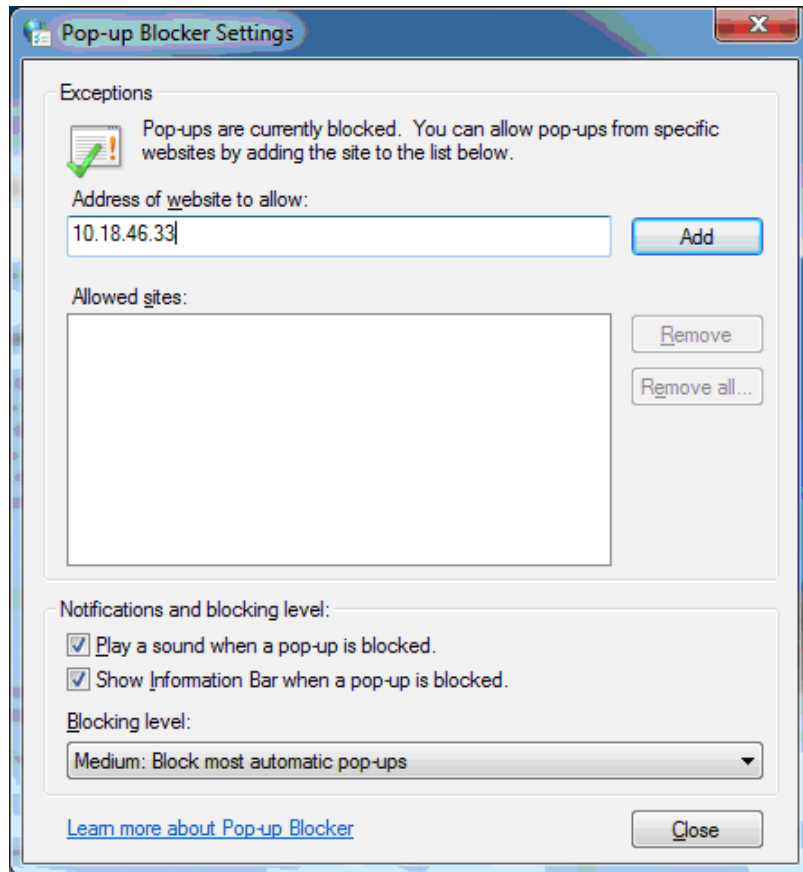


Fig. 15.2.2(e) Pop-up Blocker Settings

- 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.

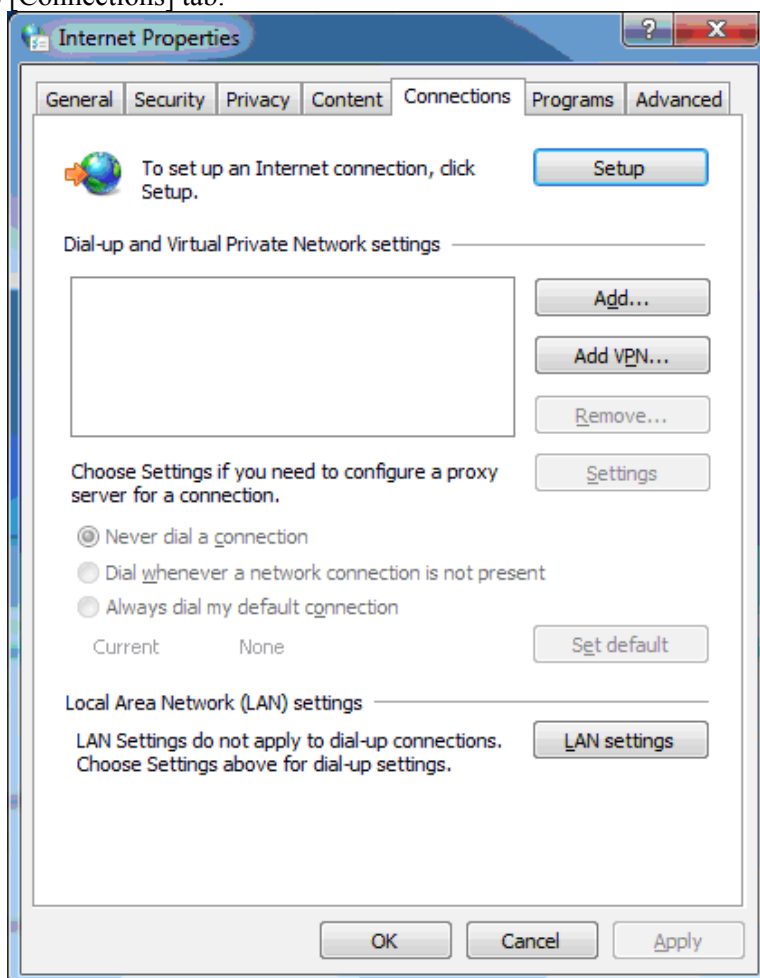


Fig. 15.2.2(f) Internet Properties

- b. Click the [LAN Settings] button.

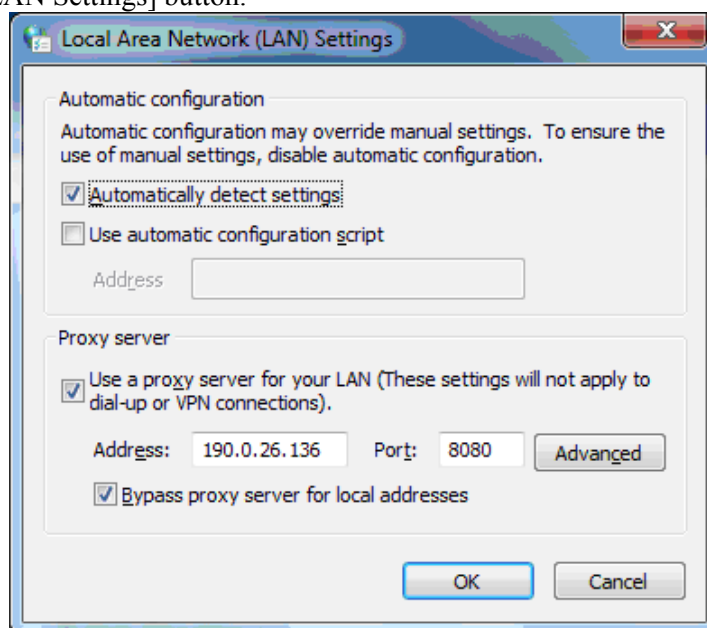


Fig. 15.2.2(g) Local Area Network (LAN) Settings

- 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].

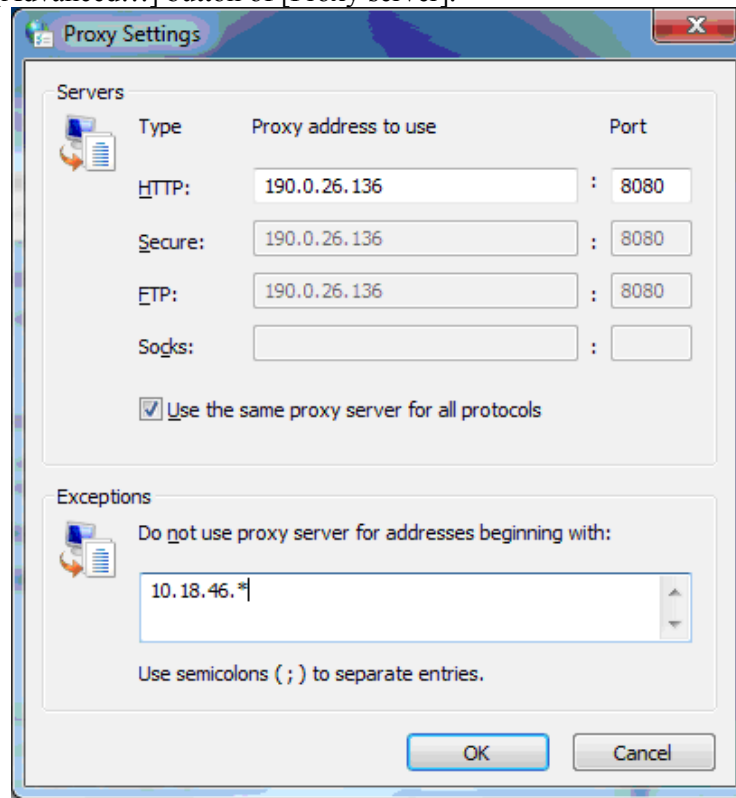


Fig. 15.2.2(h) Proxy Settings

- 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].

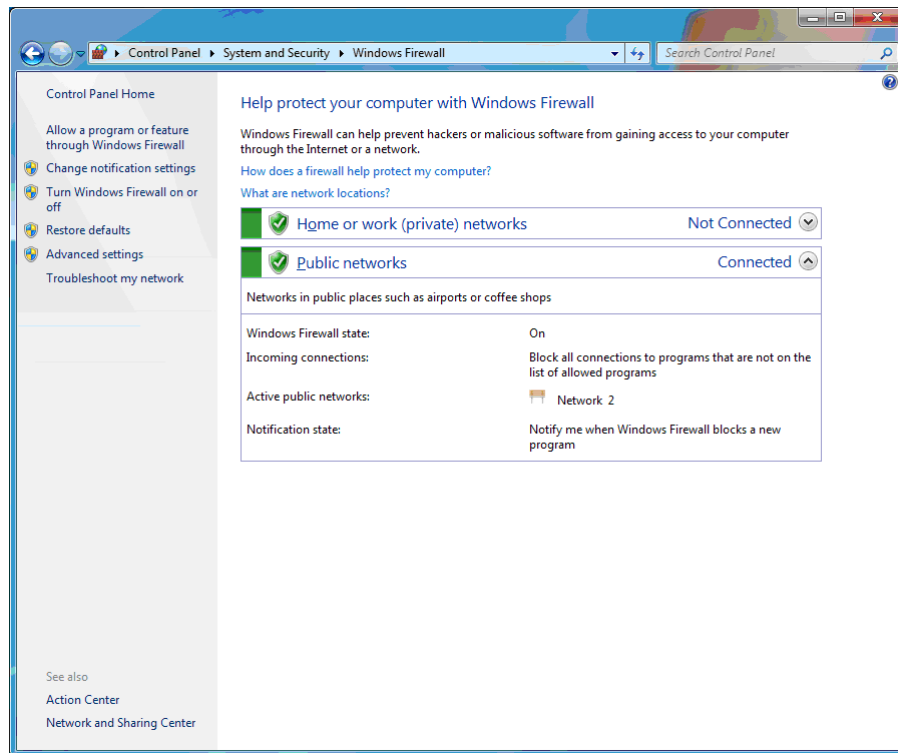


Fig. 15.2.2(i) Windows Firewall

- ii. Click [Allow a program or feature through Windows Firewall].

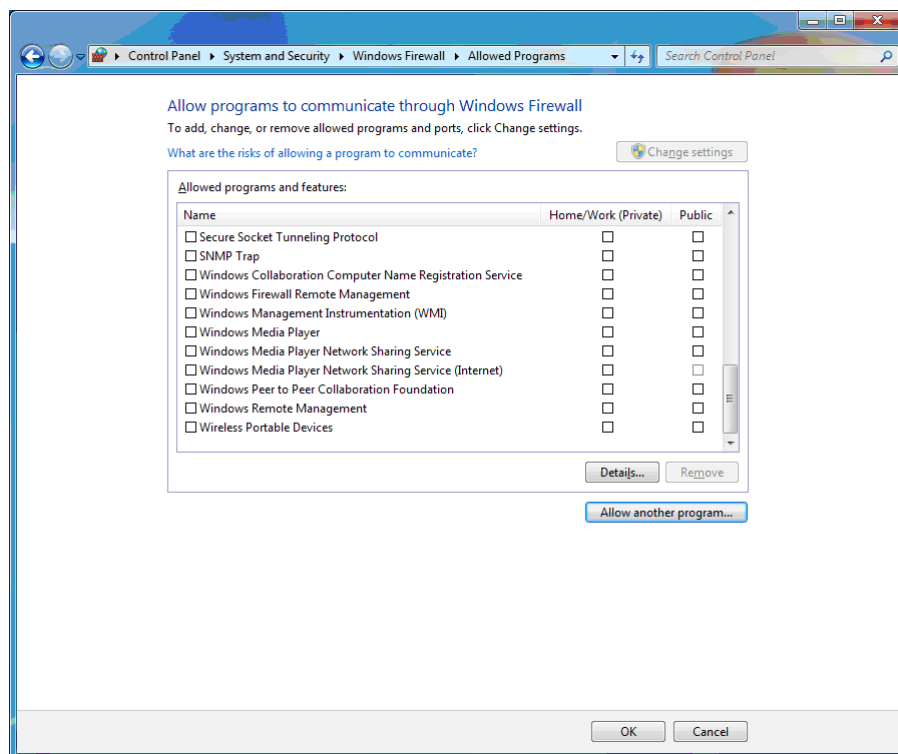
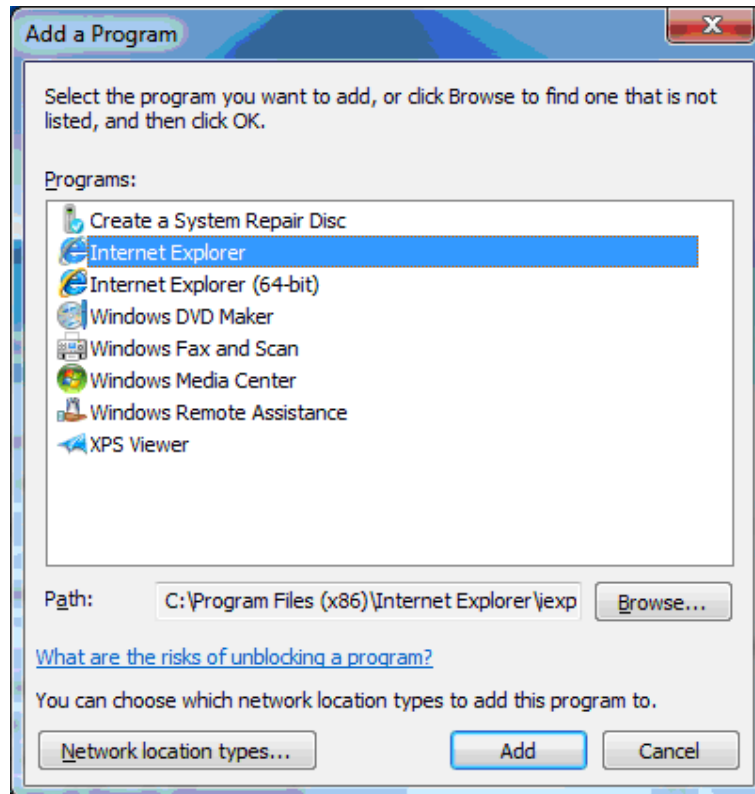


Fig. 15.2.2(j) Allowed Programs

- iii. Click the [Change settings] button.

**Fig. 15.2.2(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.

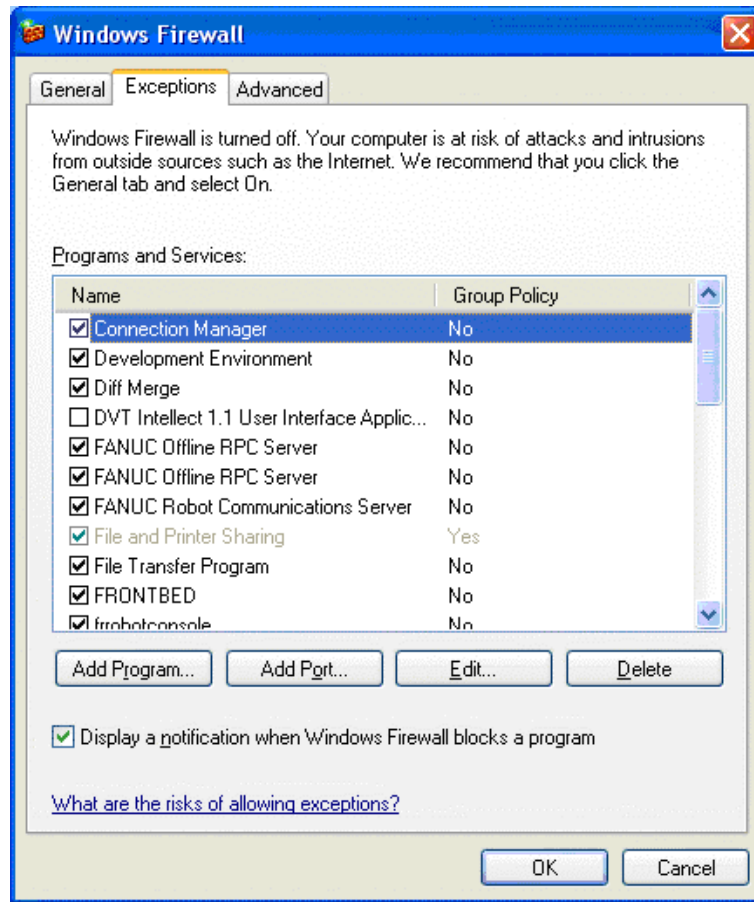


Fig. 15.2.2(l) Exceptions

iii. Click the [Add Program] button.

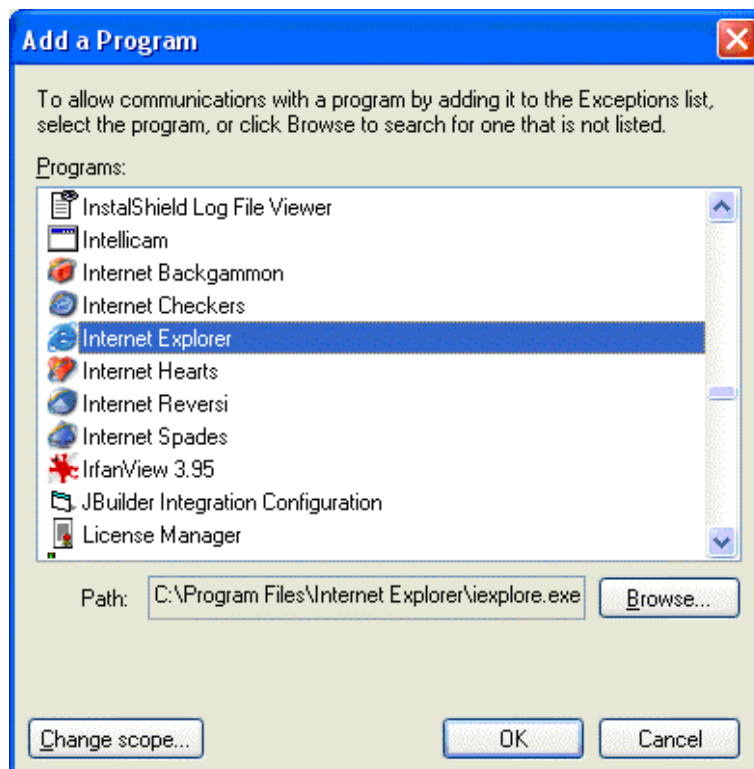


Fig. 15.2.2(m) Add a Program

iv. Select [Internet Explorer] from the list, then click the [OK] button.

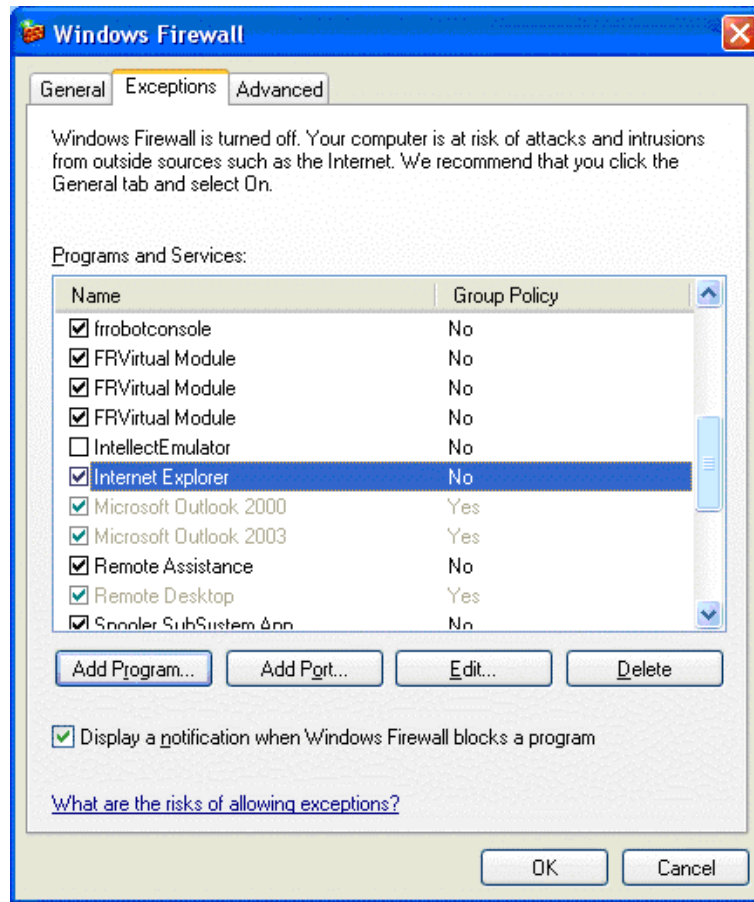


Fig. 15.2.2(n) Exceptions

- v. Click the [OK] button.

15.2.3 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 15.2.2 above.

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 15.2.3 .

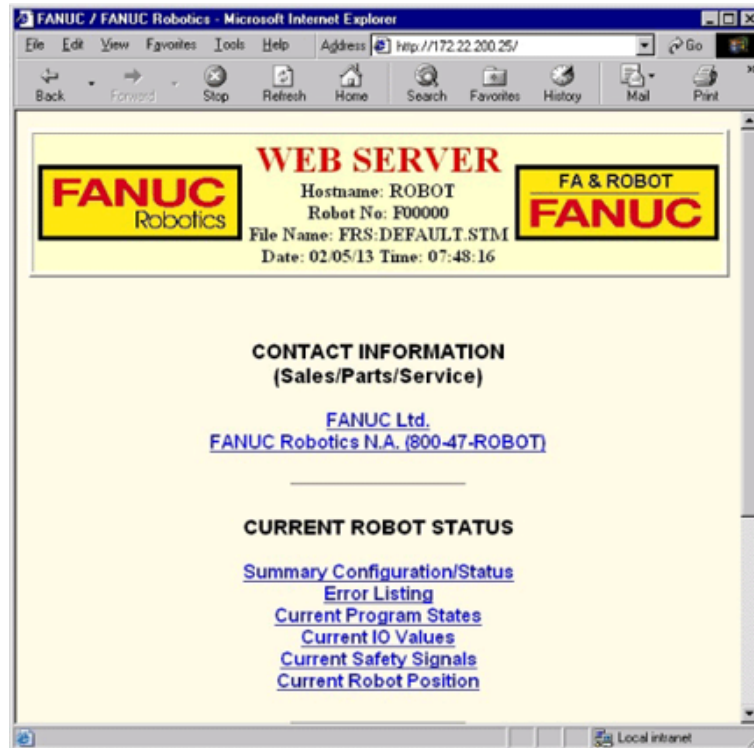


Fig. 15.2.3 Robot HOME page

If you are unable to make this connection, refer to the SETTING UP TCP/IP section in this manual or refer to the section 15.2.3.1 Troubleshooting Remote Connection or contact your System Network Administrator.

15.2.3.1 Troubleshooting remote connection

Tips for troubleshooting when the advanced iPendant 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 MONITOR iPendant FUNCTION

15.3.1 Overview

Monitor iPendant function provides you with the capability to display and monitor the current iPendant screens 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 iPendant or robot controller.

For information on the limitations of this feature, refer to Section 15.7.1 Limitations(Monitor iPendant function).

15.3.2 Operation

The following sections detail the procedure required for connection of this feature.

15.3.2.1 PC connection to the controller

This section will describe the method to connect to the robot controller and display the screen for monitoring the *iPendant* operation.

Procedure 15-3 Remotely Monitoring the *iPendant*

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 *iPendant* (ECHO).

If the connection is successfully made you will briefly see the LOGIN screen, similar to that shown in Figure 15.3.2.1(a) , displayed in Internet Explorer.

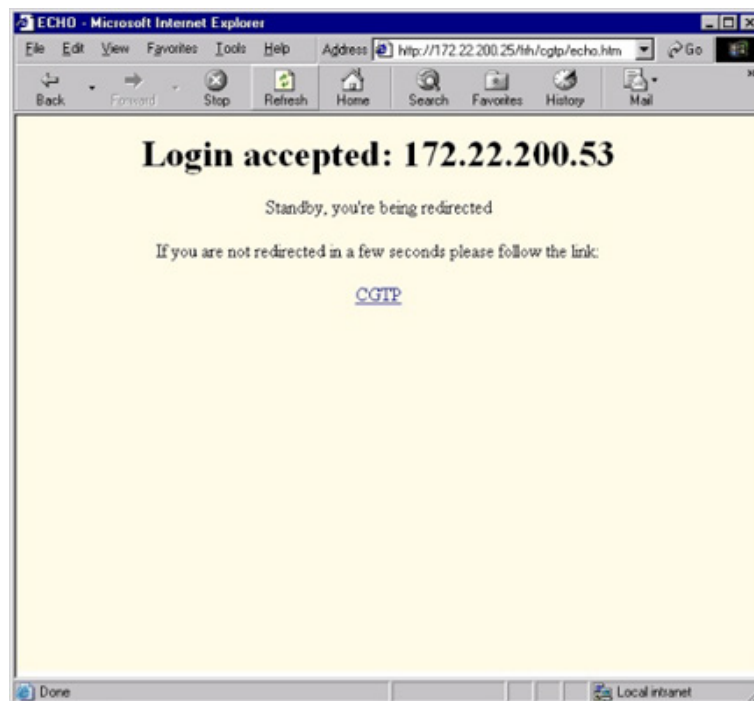


Fig. 15.3.2.1(a) Monitor *iPendant* LOGIN Screen

After the LOGIN Screen, you will see a remote display of the current *iPendant* screen. The display will show the current *iPendant* 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 15.3.2.1(b) .

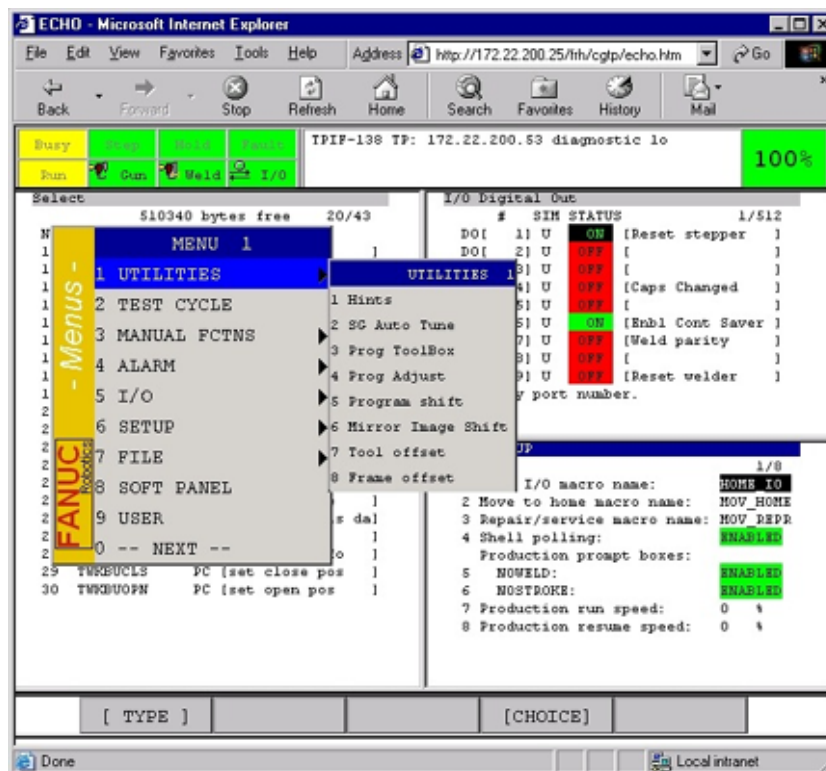


Fig. 15.3.2.1(b) Remote iPendant Monitoring Screen

15.4 NAVIGATE iPendant FUNCTION

15.4.1 Overview

Navigate iPendant function allows you to display an iPendant-like screen on a PC using Microsoft ® Internet Explorer. The PC user can configure on the PC Browser, navigate the controller menus and screens, and enter data remotely. Remote Operations are only available if the R558 Internet Connectivity and Connection option or R850 Remote iPendant function are loaded.



CAUTION

Remote Operation is completely independent from the operation of the actual iPendant 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 iPendant or what the operator is doing nor can the iPendant 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 15.7.2 Limitations(Navigate iPendant Function) .

The following functions can be performed on the navigate iPendant display:

- All iPendant screens available from the MENU 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 iPendant).
- Any input from the iPendant numeric keypad, Function keys or cursor movement.

15.4.2 Operation

The following sections detail the operation of this feature.

15.4.2.1 PC connection to the controller

This section will describe the method to connect to the robot controller and display the navigate iPendant screen.

Procedure 15-4 Navigate iPendant 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 iPendant (CGTP). If it greyed out, then you do not have the Internet Protocol Connectivity and Customization (R558) option or Remote iPendant function (R850) installed.

NOTE

You can use Navigate iPendant by the following addresses.

- `http://<myrobot_name_ or_address>/frh/cgtp/cgtp.htm`
- `http://<myrobot_name_ or_address>/frh/cgtp/remote.stm`

The remote.stm is a redirection web page.

The page select Joggin iPendant or Navigate iPendant by the real iPendant is connected or not.

If,real iPendant is connected, the page will redirect to Navigate iPendant.

And if real iPendant is not connected, the page will redirect to Jogging iPendant.

4. If you have the HTTP Authentication for the iPendant set to AUTHORIZE, the prompt shown in Figure 15.4.2.1(a) will be displayed on the PC.



Fig. 15.4.2.1(a) Remote iPendant Operation Password Screen

Type the USERNAME and PASSWORD that you set on the Robot Controller in the HOSTCOMM Setup>HTTP Authentication for the iPendant.

If the connection is successfully made you will briefly see the LOGIN screen, similar to that shown in Figure 15.4.2.1(b) , displayed in Internet Explorer.

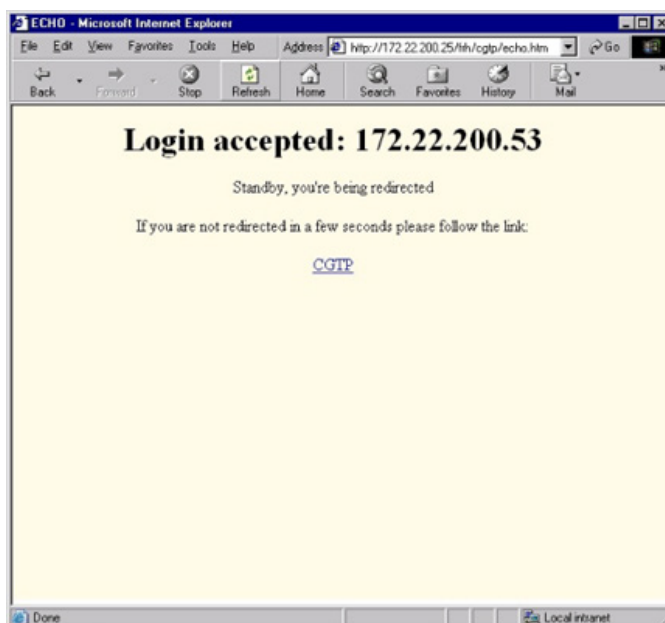


Fig. 15.4.2.1(b) Remote iPendant LOGON Screen

If this is the first time you are logging in, you will see an iPendant screen similar to the one shown in Figure 15.4.2.1(c). If you have logged in before you might see a different screen configuration, depending on how your controller was set up.

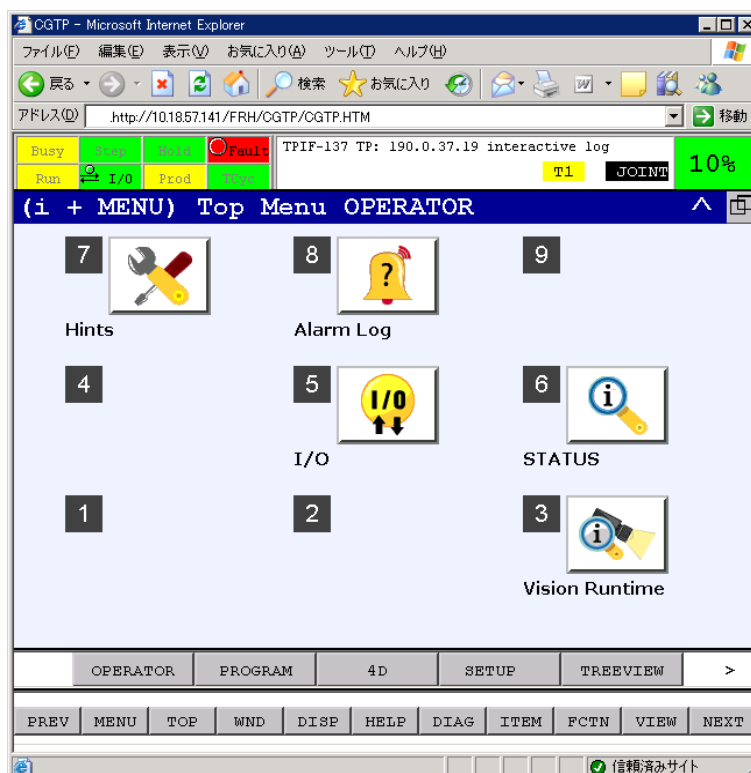


Fig. 15.4.2.1(c) Remote iPendant Operation Screen

If the Navigate iPendant connection failed, refer to Section 15.2.3.1 for troubleshooting tips.

15.4.2.2 Keys

While in the Navigate iPendant connection or Jogging iPendant connection you can use your mouse to select any of the available function keys or the auxiliary keys available below the iPendant screen.

These auxiliary keys allow you access to certain functions available as Hard Keys on the actual *iPendant*. Some of these keys are also mapped to keys on the PC keyboard as shown in Table 15.4.2.2(a) below:

Table 15.4.2.2(a) Remote Operation Key Mapping

<i>iPendant</i> 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	
VIEW	Activates the Related View menu if <i>i</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
	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 15.4.2.2(b) below:

Table 15.4.2.2(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.
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.

Mouse Event	Description
Select Links, or custom controls on Web pages, Help, Diagnostics, and any custom iPendant 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>i</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

15.4.2.3 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 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 only allow to edit in the background. If you try to make a change to the program without background editing, then the warning "TPIF-069 Use background edit" is posted
- FWD/BWD is not available from Internet Explorer in standard option. R850 Remote iPendant function is required to use FWD/BWD from Internet Explorer.

15.4.2.4 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.

15.5 JOGGING iPendant FUNCTION

15.5.1 Overview

Jogging iPendant function allows you to display an iPendant-like screen on a PC using Microsoft ® Internet Explorer. The PC user can configure on the PC Browser, navigate the controller menus and screens, and enter data remotely. Remote Operations are only available if the R850 Remote iPendant function are loaded.



WARNING

Jogging iPendant function enables to move the robot by PC. Extreme caution must be exercised when using this feature.

Emergency stop button must be prepared near the PC to use any time.

Jogging iPendant function must be used in the sight of the robot.

Jogging iPendant function is used to auto mode. You must check the person does not stay in the safety fence.

For information on the limitations of this feature, refer to Section 15.7.3 Limitations(Jogging iPendant Function) .

All functions are available to operate by PC such as following functions can be performed on the Jogging iPendant operation display:

- All iPendant screens available from the MENU 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 iPendant).
- Any input from the iPendant numeric keypad, Function keys or cursor movement.
- Robot movement operation such as Jogging , TP program execution.
- Foreground TP editing.
- Change coordinate, change override.
- Applying DCS parameter.

15.5.2 Operation

The following sections detail the operation of this feature.

15.5.2.1 PC connection to the controller

This section will describe the method to connect to the robot controller and display the jogging iPendant screen.

Procedure 15-5 Jogging iPendant Connection

Steps

1. Press menu key and select SETUP -> HOST comm. Enter the TCP/IP detail menu.
2. Input PC ipaddress to PC Jog IP address field.
3. Turn off the TP enable key and change AUTO mode.
4. Bring up the Internet Explorer.
5. 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).
6. From the HOME page, select Jogging iPendant .
If the connection is successfully made you will briefly see the LOGIN screen, similar to that shown in Figure 15.5.2.1(a) , displayed in Internet Explorer.

NOTE

You can use Jogging iPendant by the following addresses.

- `http://<myrobot_name_or_address>:3080/frh/cgtp/itp.htm`
- `http://<myrobot_name_or_address>/frh/cgtp/remote.stm`

The remote.stm is a redirection web page.

The page select Joggin iPendant or Navigate iPendant by the real iPendant is connected or not.

If,real iPendant is connected, the page will redirect to Navigate iPendant.

And if real iPendant is not connected, the page will redirect to Jogging iPendant.

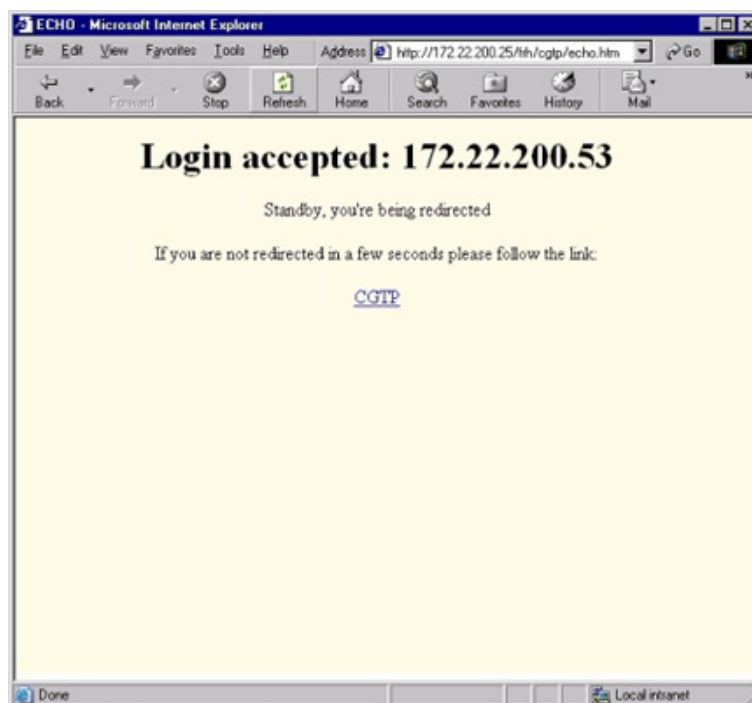


Fig. 15.5.2.1(a) Jogging iPendant LOGON Screen

If this is the first time you are logging in, you will see an iPendant screen similar to the one shown in Figure 15.5.2.1(b) . If you have logged in before you might see a different screen configuration, depending on how your controller was set up.

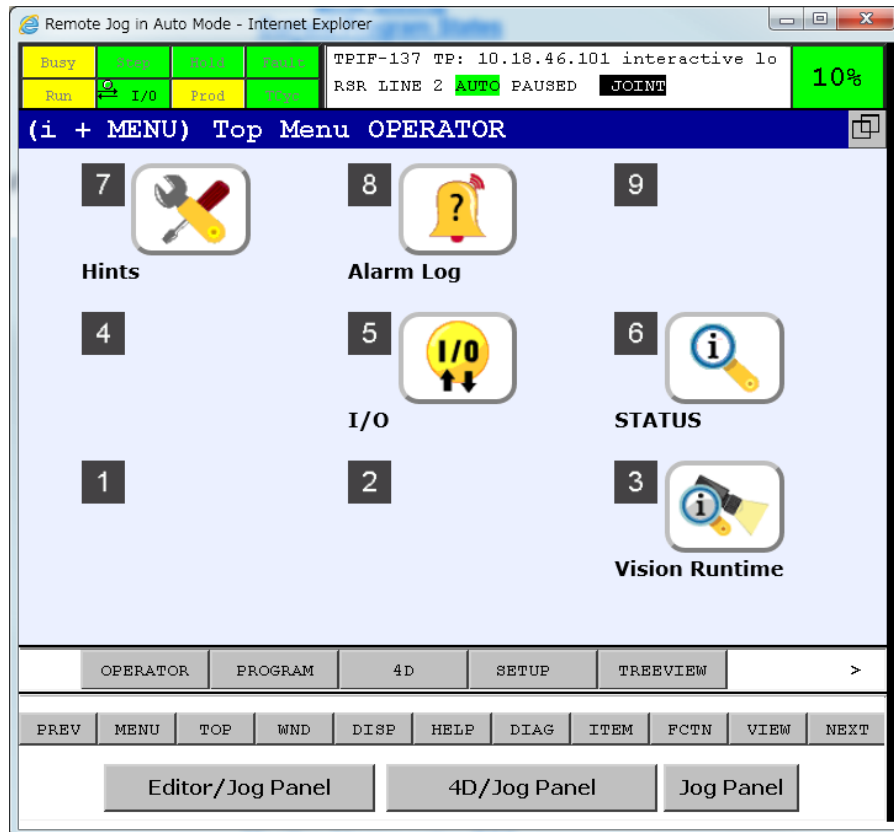


Fig.15.5.2.1(b) Jogging iPendant Function

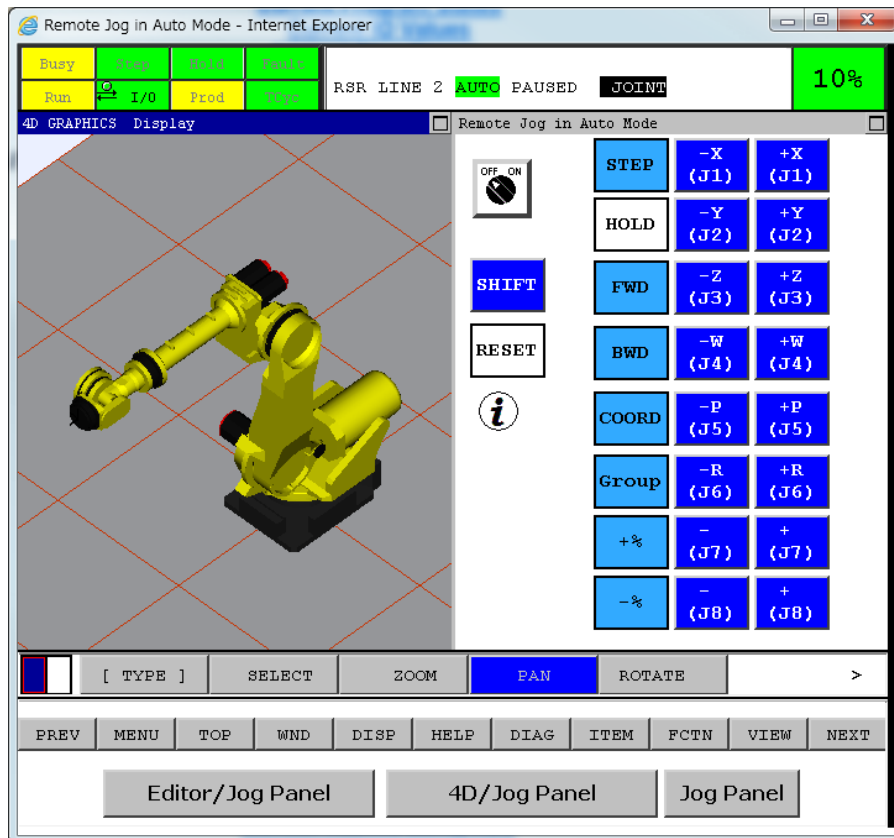


Fig.15.5.2.1(c) Jogging iPendant Function

If the jogging iPendant connection failed, refer to Section 15.5.2.3 for troubleshooting tips.

7. Real iPendant will be disconnect and you will see an iPendant screen shown below when the real iPendant is connected.



Fig15.5.2.1(d) iPendant screen while jogging iPendant connected.

If you want to connect real iPendant, press menu key to reconnect real iPendant and disconnect iPendant jogging connection.

15.5.2.2 Keys

Key assignment of Jogging iPendant function is same as Navigate iPendant function. Refer to Section 15.4.2.2 keys.

15.5.2.3 Troubleshooting jogging iPendant connection

Tips for troubleshooting when the Jogging iPendant connection does not work.

- Confirm that the TP enable switch is turned off when the real iPendant is connected.
- Confirm the mode switch is AUTO mode.
- Confirm PC's ip address is set properly PCjogIP address field in the host comm menu in setup.
- Press f4[PING] key on PC jog IP address field, and check the PING is succeeded. When PING is timed out, network setting may be wrong.
- Refer to section 15.2 and confirm that Internet Explorer settings and iPendant control is properly installed.

15.6 DISPLAYING iPendant KEY SHEET

Actual iPendant key sheet can be displayed for Remote iPendant operation.

By changing system variable “\$JOG_IN_AUTO” you can use iPendant key sheet. Depending on the changed value, a vertical or horizontal key sheet is displayed. You can use it according to the size and resolution of the monitor to be displayed.

NOTE

iPendant key sheet available on version 7DF1/05(V9.10P/05) or greater.

15.6.1 Vertical iPendant Key Sheet

Set the system variable “\$JOG_IN_AUTO” to 2.

When opening the remote iPendant screen, the following screen is displayed.

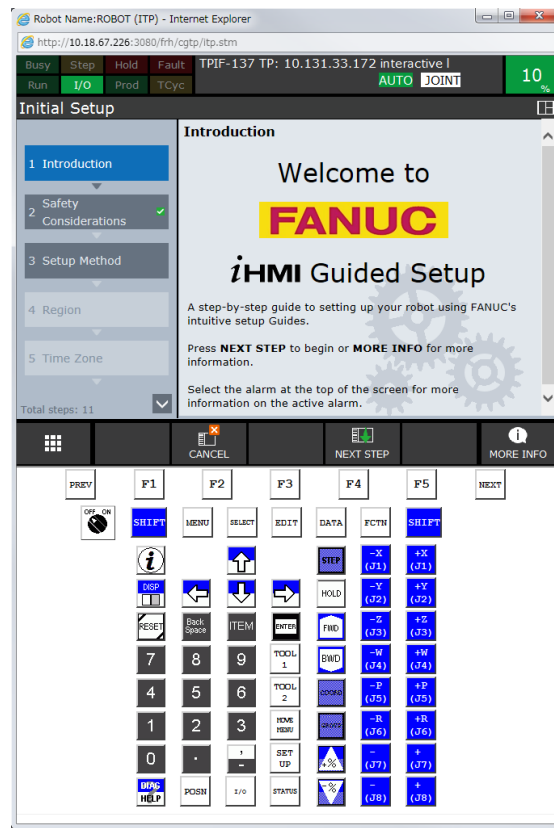


Fig.15.6.1 Vertical iPendant key sheet screen

15.6.2 Horizontal iPendant Key Sheet

Set the system variable “\$JOG_IN_AUTO” to 3.

When opening the remote iPendant screen, the following screen is displayed.



Fig.15.6.2 Horizontal iPendant key sheet screen

15.6.3 Limitations of iPendant Key Sheet

There are some limitations for the iPendant key sheet as follows.

- By setting to display the iPendant key sheet, the key sheet is displayed on all Remote iPendant screens of Monitor iPendant screen, Navigate iPendant screen and Jogging iPendant screen.
- Unusable keys, such as jog keys on the Navigate iPendant screen, are displayed in gray and cannot be used.



Fig.15.6.3 Key sheet on Navigate iPendant screen

15.7 LIMITATIONS(GENERAL)

Advanced iPendant function allows you to display and monitor any iPendant screen and perform many iPendant operations on a PC using Internet Explorer®.

The following limitations apply to the advanced iPendant function.

- The remote display will look similar to the iPendant but will not be an exact duplicate. Fonts, Character sizes, colors, images, and overall format might be different.
- OpenGL 2.0 supported graphic board is required to use 4D graphics function in PC browser. If 4D graphics does not work, please update the device driver of the graphic board.

CAUTION

If any web page is being displayed, any active link or component (ie iPendant Control) can be selected on the remote iPendant screen and activated. This may cause an interaction with the robot or with the actual iPendant. Care must be taken when viewing web pages remotely.

15.7.1 Limitations(Monitor iPendant function)

The following limitations apply to the remotely displayed iPendant 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 iPendant or robot controller except in the following cases:
- The remote display will look similar to the iPendant 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 iPendant 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 iPendant.

15.7.2 Limitations(Navigate iPendant function)

Navigate iPendant function allows you to display and monitor any iPendant screen and perform many iPendant operations on a PC using Internet Explorer ® .

The following limitations apply to the Remote Operation of the iPendant 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 iPendant 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 iPendant such as WIRE+/-, BACKUP, and so forth.
- Several of the iPendant screens can only be displayed by a single device at any given time. The actual iPendant always has precedence. If the iPendant 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 iPendant operator, the remote display will be forced to the UTILITIES>HINTS screen. However, an error will not be posted.

NOTE

Navigate iPendant connection is closed 5 minutes by default setting. To keep connection, set 0 to \$ui_config.\$timeout in system variable.

15.7.3 Limitations(Jogging iPendant function)

Jogging iPendant function allows you to not only display and monitor any iPendant screen and perform many iPendant operations but also jogging operation, TP program execution on a PC using Internet Explorer ® .

The following limitations apply to the Jogging iPendant function:

- iPendant jogging connection is refused when the real TP enable switch is turned on.
- iPendant jogging connection is refused when the mode switch is T1 or T2. The connection immediately disconnect by change mode switch to T1 or T2 while iPendant jogging connection.
- Real iPendant has mastership therefore iPendant jogging connection refused when real iPendant is started to connect.
- Multiple iPendant jogging connection is not supported.
- iPendant jogging connection is refused when PCJogIP address is not set properly in Setup Host comm menu

15.8 OPTION CONFIGURATION

- Monitor *iPendant* Function is standard option.
- Navigate *iPendant* Function is optional software, Please order any one of the following options.
 - R558 Internet Connectivity and Connection function
 - R850 Remote *iPendant* function.
- Jogging *iPendant* Function is optional software, Please order the following option.
 - R850 Remote *iPendant* function.

R850 Remote *iPendant* is available 7dc3/03(V8.30P03) or later.

16 REMOTE iPendant FUNCTION

This chapter provides information about R843 Remote iPendant function.

Contents of this chapter

- 16.1 Overview
- 16.2 Connection to Remote iPendant
- 16.3 Monitor iPendant (JECHO)
- 16.4 Navigate iPendant (JCGTP)
- 16.5 Jogging iPendant (JITP)
- 16.6 Limitations

16.1 OVERVIEW

R843 Remote iPendant function provides monitoring and operating iPendant screen and jogging the robot from a browser on external device like a PC or tablet device.

There is a similarity named option “R850 PC Remote iPendant function”. Both R843 Remote iPendant and R850 PC Remote iPendant can monitor and operate iPendant screen and jog the robot from external device. But R850 PC Remote iPendant function can be used only on Internet Explorer on Windows PC and it requires installing FANUC PC software iPendant Controls (or ROBOGUIDE) in the Windows PC. R843 Remote iPendant are implemented using JavaScript so it can be used on common browser and doesn’t require iPendant Controls (or ROBOGUIDE).

PC Remote iPendant function (R850)

- This function is available only on Windows PC.
- This function is available only on Internet Explorer.
- This function requires installing FANUC PC software iPendant Controls (or ROBOGUIDE) in the PC.

Remote iPendant function (R843)

- This function is available on common external device like PC and tablet device.
- This function is available on common browser. (Recommend Google Chrome.)
- This function does not require installing any additional software in the device.

But R843 Remote iPendant has a limitation that there are some screens can’t be displayed on a browser. Please refer to “16.6.1 Limitations (general)” for details of the screen.

In this chapter, external device like PC and tablet device that can be connected with robot controller using R843 Remote iPendant function is called “HMI (Human Machine Interface) device”.

R843 Remote iPendant option provides the following 3 features:

- Monitor iPendant (JECHO)
- Navigate iPendant (JCGTP)
- Jogging iPendant (JITP)

NOTE

R843 Remote iPendant function is a software option available only on V8.30P/20 or greater.

16.2 CONNECTION TO REMOTE *iPendant*

Accessing to each Remote *iPendant* function from HMI device is operated from robot homepage. This section describes the way to access to robot homepage.

16.2.1 Settings

TCP/IP settings like IP address are the requirements on your robot controller for Remote *iPendant* function. About settings of TCP/IP please refer to the chapter “2 SETTING UP TCP/IP” in for details.

16.2.2 Connection

This subsection provides the sequence to connect to Remote *iPendant* function from robot homepage.

Procedure 16-1 Connection to Remote *iPendant*

Conditions

- 1 IP address, subnet mask and router IP address must be set in setting up TCP/IP.
- 2 The HMI device is connected to a network that can be used to access the robot.

Steps

- 1 Activate your browser on HMI device.
- 2 In the Address field of the browser, input “http://<Robot IP Address>” to open the robot Homepage.
Ex) http://192.168.1.100

If the connection is successfully made you will briefly see the robot homepage, similar to that shown in Fig. 16.2.2(a), displayed on the HMI device.

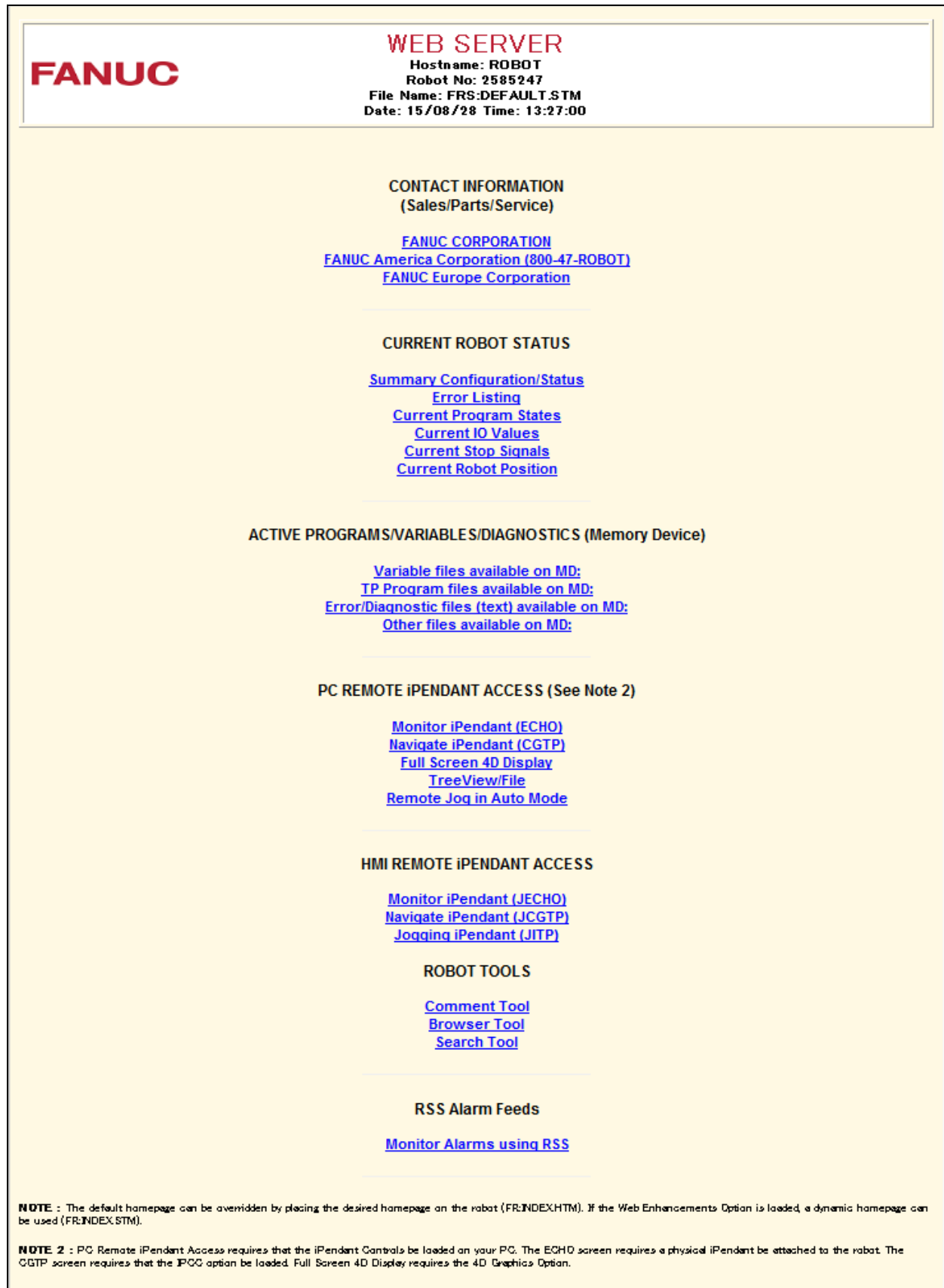


Fig. 16.2.2(a) Robot homepage screen

- 3 Select the link you want to use from the following HMI REMOTE iPENDANT ACCESS list.

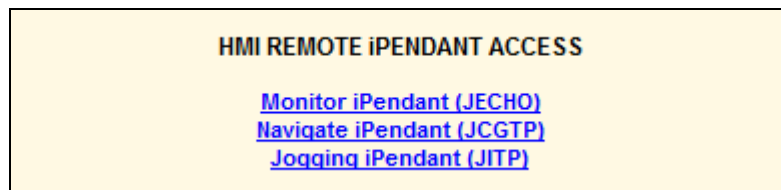


Fig. 16.2.2(b) HMI remote iPendant link

NOTE

Monitor iPendant (JECHO) and Navigate iPendant (JCGTP) function use the standard port No.80. Jogging iPendant (JITP) uses special port 3080 and 4080.

16.3 MONITOR iPendant (JECHO)

16.3.1 Overview

Monitor iPendant function provides you with the capability to display and monitor the current actual iPendant screens on a HMI device using a browser. 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 actual iPendant or robot controller.

16.3.2 Connection

This subsection provides the sequence to connect to Monitor iPendant function from robot homepage.

Procedure 16-2 Connection to Monitor iPendant

Select the Monitor iPendant (JECHO) on the robot homepage on your HMI device.

If the connection is successfully made you will briefly see the Monitor iPendant screen, similar to that shown in Fig. 16.3.2, displayed on the browser on HMI device.

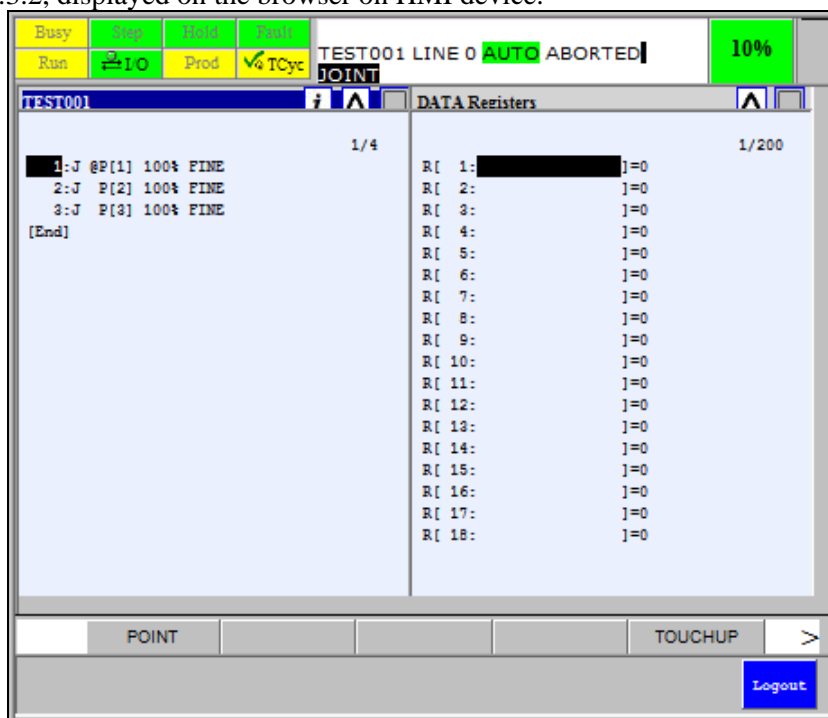


Fig. 16.3.2 Monitor iPendant screen

Procedure 16-3 Disconnection from Monitor iPendant(JECHO)

To disconnect the Monitor iPendant function, press “Logout” key in the bottom of the screen, or close the window of the browser.

16.4 NAVIGATE iPendant (JCGTP)

16.4.1 Overview

Navigate iPendant function allows you to display a iPendant-like screen on a browser on your HMI device. Navigate iPendant provides iPendant screens independent of the actual iPendant. So the HMI device user can display screens, navigate menus and enter data independent from actual iPendant.

**CAUTION**

Remote Operation is completely independent from the operation of the actual iPendant 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 iPendant or what the operator is doing nor can the iPendant operator see what the remote operator is doing or what screens are being displayed on the remote iPendant.

16.4.2 Connection

This subsection provides the sequence to connect to Navigate iPendant function from robot homepage.

Procedure 16-4 Connection to Navigate iPendant(JCGTP)

Select the Navigate iPendant (JCGTP) on the robot homepage on your HMI device. If the connection is successfully made you will briefly see the Navigate iPendant screen, similar to that shown in Fig. 16.4.2, displayed on the browser on your HMI device.

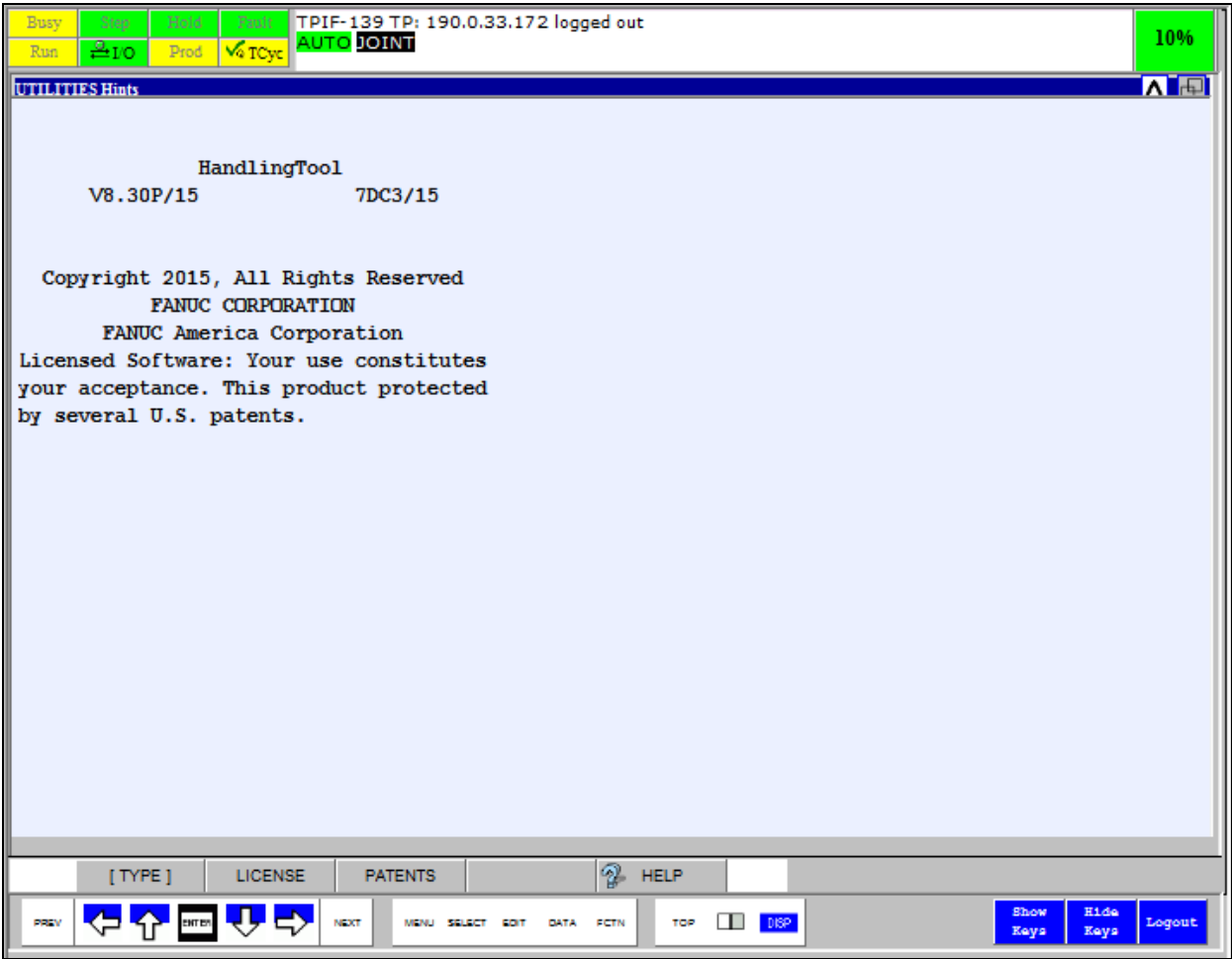


Fig. 16.4.2 Navigate iPendant screen

Procedure 16-5 Disconnection from Monitor iPendant

To disconnect the Navigate iPendant function, press “Logout” key in the bottom of the screen, or close the window of the browser.

16.4.3 Screens













This subsection explains about Monitor iPendant screen and operation of the screen.

16.4.3.1 Keys

While in the Navigate iPendant connection, you can use a mouse and touch operation to select any of the available function keys or the auxiliary keys available below the iPendant screen. These auxiliary keys allow you to access certain functions available as Hard Keys on the actual iPendant. (Table 16.4.3.1(a))

Table 16.4.3.1(a) Navigate iPendant (JCGTP) Keys

Keys	Description
PREV	This key moves to Previous Screen or Cancels a pending operation.
	The cursor key moves the cursor in the direction of an arrow.
ENTER	The ENTER key enters a numeral or selects a menu.

Keys		Description
		This key to switch the function key menu on the next page.
		The MENU key to display the menu screen.
		The SELECT key to display the program selection screen.
		The EDIT key display the program edit screen.
		The DATA key to display the program data screen.
		The FCTN key to display the function menu.
		The TOP key to display the Top Menu OPERATOR screen.
		In case that the screen on the teach pendant is split, when this key is pressed, the operation target screen is changed.
		When this key is pressed, the menu to split the screen is displayed.
		When this key is pressed, teach pendant keys are displayed to the right side of the screen.
		When this key is pressed, keys displayed to the right side of the screen hide.
		The Logout key to logout Navigate iPendant (JCGTP) screen.

In addition mouse events are supported. (Table 16.4.3.1(b))

If your HMI device has touch-screen, touch screen operation serve as these mouse click operation.

Table 16.4.3.1(b) Navigate iPendant (JCGTP) mouse event

Mouse Event	Description
Page Up	Mouse click above the lines
Page Down	Mouse click below the lines
Select a line	Mouse click on a line
Select an item	Mouse click on an item
PREV operation	Mouse click to the left of the function keys
NEXT operation	Mouse click on ">" to the right of the function keys
Display MENU screen	Mouse click on the left side of the Focus Bar
DISPLAY menu list	Mouse click on the center of the Focus Bar
FCTN menu list	Mouse click on the right side of the Focus Bar
Related Views	Mouse click on the "f" shown in the Focus Bar
Zoom	Mouse click on the Square shown in the Focus Bar
Maximize/Restore	Mouse click on the "^" shown in the Focus Bar

16.4.3.2 Show keys

When you press “Show Keys” on lower part of Navigate iPendant screen, key sheet shown in Fig. 16.4.3.2 will be displayed the right side of the screen. Keys removed motion keys from actual iPendant are located on the key sheet.

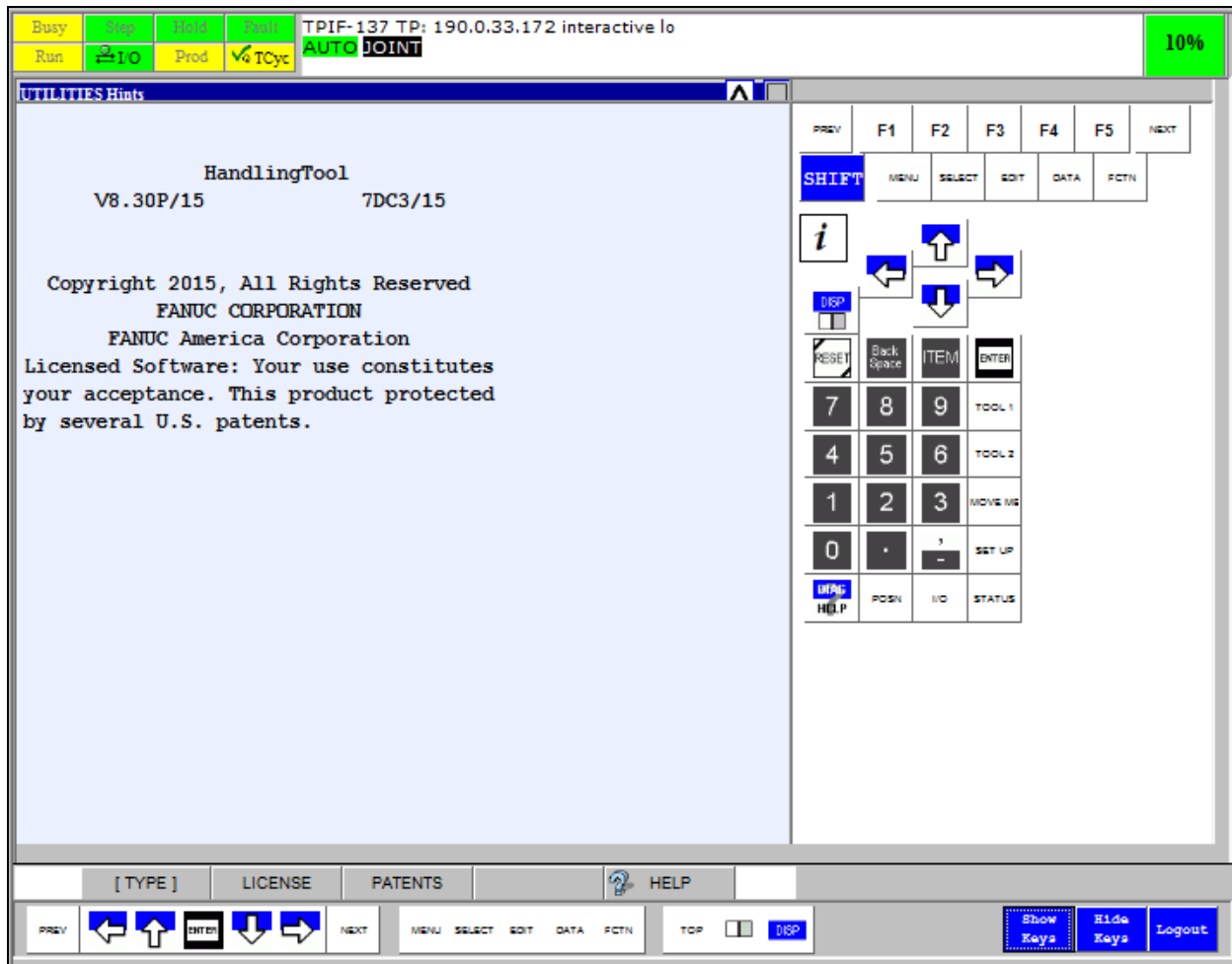


Fig. 16.4.3.2 Operation key sheet

When you press “Hide keys” on lower part of Navigate iPendant screen, key sheet hides and the screen changes to single.

16.5 JOGGING iPENDANT (JITP)

16.5.1 Overview

Jogging iPendant function allows you to display an iPendant-like screen on a browser on you HMI device. This function provides the following functions.

- Robot movement operation such as Jogging, TP program execution.
- Foreground TP editing.
- Change coordinate.
- Change override.
- Change TP Enable/Disable.
- Applying DCS parameter.
- Clearing DCS alarms.

**WARNING**

Jogging *iPendant* function enables to move the robot by HMI device. Extreme caution must be exercised when using this function. When you execute jog operation by Jogging *iPendant* function, you must strictly adhere to the following precautions.

- Set Mode Switch to AUTO mode.
- Jog operation must be executed from outside of the safety fence.
- You must check the person does not stay in the safety fence.
- Emergency stop button must be prepared near the operator to use any time.
- Jog operation must be executed in the sight of the robot.

16.5.2 Connection

This subsection provides the sequence to connect to Jogging *iPendant* function from robot homepage.

Procedure 16-6 Connection to Navigate *iPendant*

Steps

- 1 Press MENU key and select SETUP -> HOST comm. Enter the TCP/IP detail menu.
- 2 Input IP address of the HMI device to "PC Jog IP addr" field.
- 3 Turn off the TP enable key.
- 4 Change to AUTO mode.
- 5 Bring up a browser on your HMI device.
- 6 In the browser address field, type "http://<Robot IP Address>/" to access robot homepage.
Ex) http://192.168.1.100/
If the connection is successfully made you will briefly see the robot homepage, similar to that shown in Fig. 16.2.2(a), displayed on the HMI device
- 7 Select the Jogging *iPendant* (JITP) on the robot homepage on your HMI device.
If the connection is successfully made you will briefly see the Jogging *iPendant* screen, similar to that shown in Fig. 16.5.2(a), displayed on the browser on your HMI device.



Fig. 16.5.2(a) Jogging iPendant screen

Actual iPendant will be disconnected automatically and you will see an iPendant screen shown in Fig. 16.5.2(b) when a HMI device is connected to the robot controller by Jogging iPendant function.



Fig. 16.5.2(b) Disconnected iPendant screen

Troubleshooting Jogging iPendant Connection

Tips for troubleshooting when the jogging iPendant connection does not work.

- Confirm that the TP enable switch is turned off when the actual iPendant is connected.
- Confirm the mode switch is AUTO mode.
- Confirm HMI's IP address is set properly "PC Jog IP addr" field in the host comm menu in setup.
- Press F4[PING] key on "PC Jog IP addr" field, and check the PING is succeeded. When PING is timed out, network setting may be wrong.

Procedure 16-7 Disconnection from Jogging iPendant

To disconnect the Jogging iPendant function, press "Logout" key in the bottom of the screen, or close the window of the browser.

NOTE

Even if you disconnect Jogging iPendant, actual iPendant is not connected automatically. If you want to connect the actual iPendant to the robot, please execute the reconnection operation described in "Procedure 16-8 Reconnection of Actual iPendant".

Procedure 16-8 Reconnection of Actual iPendant

To reconnect actual iPendant to the robot controller, press MENU key on the actual iPendant.

NOTE

While Jogging iPendant is connected to the robot, when you press MENU key on actual iPendant, the connection of Jogging iPendant is disconnected and actual iPendant is connected preferentially to the robot controller.

16.5.3 Screens




There are two kinds of screens for Jogging iPendant function. They are separated by the kinds of HMI device and browser, and the state of the HMI device.

If your HMI device is Windows PC installed iPendantControls and the browser is Internet Explorer, refer to the subsection "16.5.3.2 Screens (with iPendantControls)". If your HMI device and browser is not one described above, refer to the subsection "16.5.3.1 Screens (without iPendantControls)".

16.5.3.1 Screens (without iPendantControls)

Jogging iPendant screen has the auxiliary keys available below the Jogging iPendant screen same as Navigate iPendant screen. In addition to the keys, Jogging iPendant screen has some special keys for Jogging iPendant function described in Table 16.5.3.1.

Table 16.5.3.1 Jogging iPendant (JITP) keys

Keys		Description
		When this key is pressed, position screen is displayed to the left side of the screen and jog keys are displayed to the right side of the screen.
		When this key is pressed, jog keys are displayed to the right side of the screen.
		The TP enable switch key to change ENABLE/DISABLE the teach pendant of jogging iPendant.

When you press “Pos/Jog Panel”, Position screen is displayed and Jog panel will be displayed in the right hand of the screen shown in Fig. 16.5.3.1. Jog Panel consist of motion keys on actual *iPendant* key sheet.

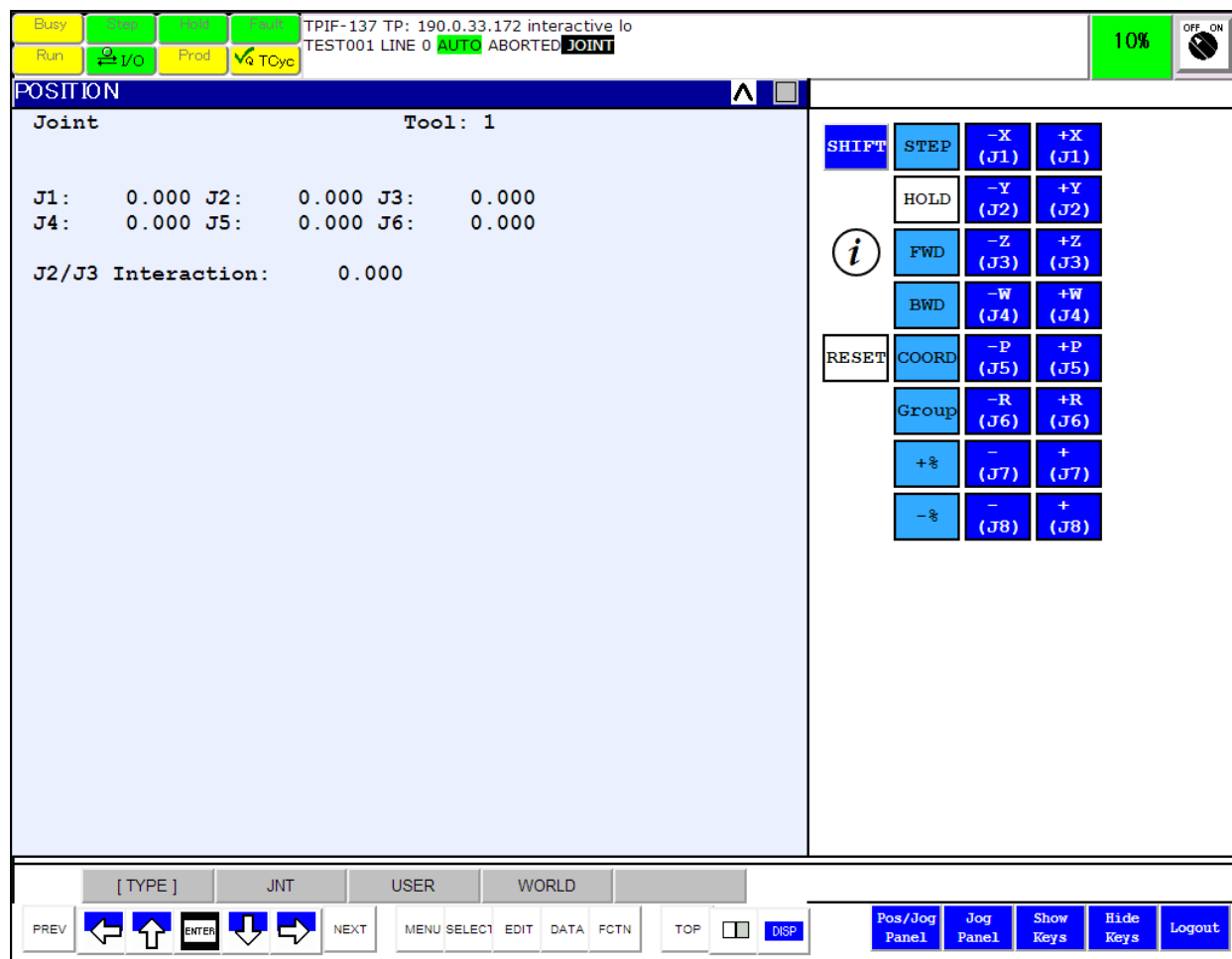


Fig. 16.5.3.1 Pos/Jog Panel screen

When you press “Hide Keys”, Jog Panel hides and the left screen goes full-screen.

16.5.3.2 Screens (with *iPendantControls*)

On Jogging *iPendant* function, if your HMI device is Windows PC installed *iPendantControls* and the browser is Internet Explorer, 4D graphics display screen can be displayed. In this case, “4D/Jog Panel” key is displayed instead of “Pos/Jog Panel”.

Table 16.5.3.2 Jogging *iPendant* (JITP) key

Keys	Description
4D/Jog Panel	When this key is pressed, 4D graphics screen is displayed to the left side of the screen and jog keys are displayed to the right side of the screen.



Fig. 16.5.3.2(a) Jogging iPendant screen with iPendantControls

When you press “4D/Jog Panel”, 4D Graphics screen will be displayed and Jog panel will be displayed in the right hand of the screen shown in Fig. 16.5.3.2(b).

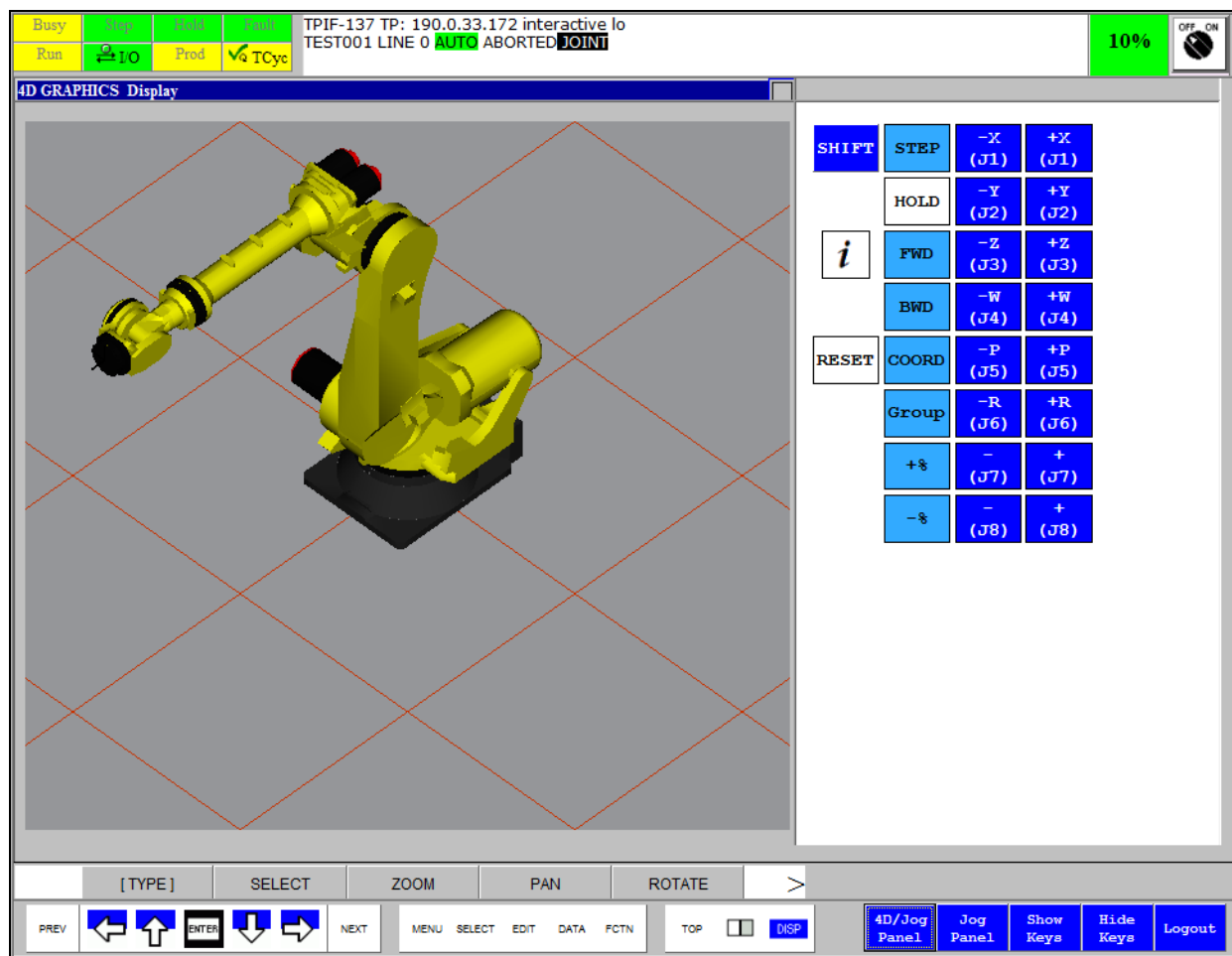


Fig. 16.5.3.2(b) 4D/Jog Panel screen

When you press “Hide Keys”, Jog Panel hides and the left screen goes full-screen.

NOTE

4D Graphics Function (R764) option is required to use 4D graphics functions. If this option is not installed, the screen Fig. 16.5.3.2(b) is displayed but other 4D graphics functions are not available.

16.5.4 Jog Operation

This subsection provides the sequence to jog the robot from Jogging iPendant screen. All steps described in Procedure 16-8 is operated on the HMI device.

Procedure 16-9 Robot Jog Operation

Steps

- 1 Show Jog Panel by pressing “Pos/Jog Panel”, “4D/Jog Panel” or “Jog Panel” keys.
- 2 Press the TP enable/disable switch located in the upper right of the screen to turn it ON.
- 3 Press RESET key to clear “SYST-045 TP enabled in AUTO mode” alarm.
- 4 Press SHIFT key.

When you press SHIFT key, color of the key is changed from blue to aqua. It means SHIFT key is press and hold state (latched state). When you press SHIFT key again, the color turns to blue. It means SHIFT key is released. Refer to Table 16.5.4 for details.

Table 16.5.4 SHIFT key state

Keys	State
SHIFT	Released SHIFT key.
SHIFT	Pressed and held SHIFT key.

- 5 When press and hold a jog key with latching SHIFT key, the robot moves.

**WARNING**

When jogging the robot from HMI device, please strictly adhere to the description content at the WARNING in subsection 16.5.1.

16.6 Limitations

This section provides the limitation of Remote iPendant functions.

16.6.1 Limitations (general)

The following limitations apply to the Monitor iPendant, Navigate iPendant and Jogging iPendant function.

- The remote display will not be an exact duplicate to the Teaching Pendant. Fonts, Line space, Character size, color, images, and overall format might be different from each HMI device or browser.
- On R843 Remote iPendant function, there are some screens can't be displayed on the browser. This will exclude any graphical screens such as followings.

Common

- PMC Ladder Display screen
- Power Consumption Monitor screen
- 4D Graphics Display screen
- Icon Editor
- Panel Wizard
- Interface Panel
- iPendant General Setup
- TreeView display
- Vision Setup and Monitor screen

Arc tool

- Arc Chart screen
- Analog Meter screen
- Process Logger screen
- Process Logger Chart screen
- DMON Chart screen

Spot tool

- Cell Diagnostics screen
- Production Monitor screen
- Status Process screen
- Tip Dresser screen

Machine tool

- Initial Setting Load screen
- Machine Tool Operation screen
- CNC Actual Position screen
- Utilities Machine Tool screen

If you want to display above screens on external device, please use R558 Internet Conn/Custo and R850 PC remote iPendant function. These functions require to use Internet Explorer on Windows PC and to install FANUC PC software iPendant Controls (or ROBOGUIDE) to the PC. Please refer to the chapter “15 Advanced iPendant functions” for details.

- Screen size of HMI device is recommended more than 9 inch.
If the screen size of HMI device is smaller than this size, remote iPendant screen extend far beyond the HMI device screen and scroll bar will be displayed. And if the screen size of HMI device is not enough, some remote iPendant keys displayed lower part of the screen hide. And please don't use smaller HMI device like smart phone for safety.

16.6.2 Limitations of Monitor iPendant Function (JECHO)

The following limitations apply to the Monitor iPendant function.

- When Teaching Pendant is disconnected by jogging iPendant function, Monitor iPendant function is not used.
- Monitor iPendant function can't monitor Navigate iPendant screen. It can monitor only Teaching Pendant screen.
- A partial popup screens like Software Key Board can't be displayed.

16.6.3 Limitations of Navigate iPendant Function (JCGTP)

The following limitations apply to the Navigate iPendant function.

- You can't make any changes from Navigate iPendant if read only is enabled. To allow changes, set \$UI_CONFIG.\$READONLY[2] to FALSE.
- About editing TP program only Background Editing is allowed. Please refer to “BACKGROUND EDITING” in “OPERATOR'S MANUAL (Basic Operation) (B-83284EN)” for details of Background Editing.
- Following operations are not available on Navigate iPendant.
 - Jogging the Robot
 - Running a Program using the FWD/BWD Keys
 - Changing COORD
 - Changing Override
 - Foreground editing a program (Only background editing is allowed)
- Several of the iPendant screens are only be displayed by a single device at any given time. The Teaching Pendant always has precedence. When Navigate iPendant open a screen that is opened by Teaching Pendant, “TPIF-110 Screen used by other device” will be posted and the screen on the Navigate iPendant will not be changed. Also, if Navigate iPendant is currently displaying a screen that is requested by the Teaching Pendant operator, the Navigate iPendant will be forced to the UTILITIES>HINTS screen. However, an error will not be posted.

NOTE

Navigate iPendant connection is closed 5 minute by default setting. To keep connection, set 0 to \$UI_CONFIG.\$TIMEOUT in system variable.

16.6.4 Limitations of Jogging iPendant Function (JITP)

The following limitations apply to the Jogging iPendant function.

- Multiple Jogging iPendant connection is not supported.
- Jogging iPendant connection is refused when the Teaching Pendant TP enable switch is turned ON.
- Jogging iPendant connection is refused when the mode switch in T1 or T2.
- The connection immediately change mode switch to T1 or T2 while Jogging iPendant connection.
- Jogging iPendant connection is refused when Teaching Pendant is started to connect.

- Jogging iPendant connection is refused when PC Jog IP addr is not set properly in Setup Host comm menu.
- Robot jog by jogging iPendant function stops immediately with “SYST-066 Teach Pendant communication error” when the communication between robot and HMI device is disconnected. Robot watches closely the communication whether the communication with HMI device is normal or not. So the robot might stop by communication delay. If the communication error occurs frequently, please check your network environment.
- Press and Hold operation is recognized as right click on Windows 7 PC with touch panel (including tablet PC). It is the specification of Windows 7. And it takes about three times to recognize the Press and Hold operation as right click. So jog operation cannot be executed from Windows 7 touch panel PC by default. By following settings on Windows 7 PC you can remove a large portion of this problem.
 - Open “Control Panels” and select “Pen and Touch” to open “Pen and Touch” screen.
 - Open “Touch” tab and select “Press and hold” item, and press “Settings...” button.
 - Set the slider of “Speed:” and “Duration:” to the extreme left like Fig. 16.6.4.
 - Uncheck the checkbox for “Enable press and hold for right-click”.
 - Press “OK” to apply the settings and close the all windows.

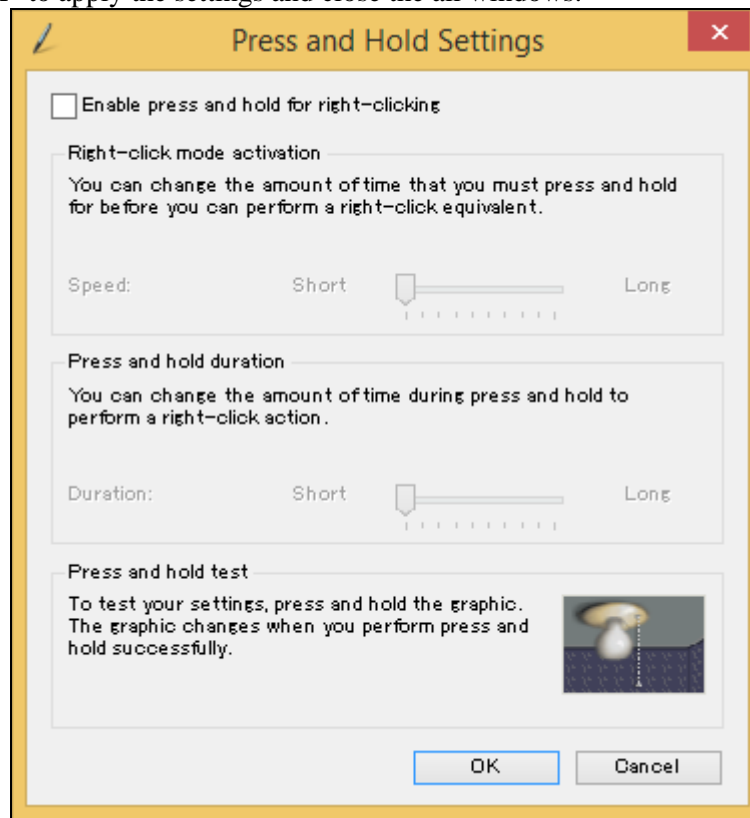


Fig. 16.6.4 Press and Hold Settings

17 iRCONNECT

17.1 OVERVIEW

iRConnect is a software option (R818), available only on v8.20 or greater, that facilitates sending E-mails to the phone which can be viewed with the iRConnect mobile application. The E-mails are based on user configuration settings provided via an XML file or the Teach Pendant iRConnect menu. It provides several functions:

- iRConnect menu with related view links to other applications which support iRConnect.
- Sending E-mail based on user configuration.
- KAREL programs which can be called from TP programs to send E-mail.
- KAREL Built-ins that can be used to send user defined messages and alerts, e.g. production status, part counts, summary etc.

iRConnect provides a queue timer and a queue size to send E-mail whichever occurs first. There is a periodic status E-mail which is sent on power up and each period subsequently. It can include summary information of what options are loaded and last power off time, etc. There is an iRConnect mobile application for the phone that presents all messages sent out from the controller in a friendly format.

17.2 REQUIREMENTS

iRConnect requires the following components:

- Robot Controller with the iRConnect option (see Section 17.3)
- SMTP server that the robot controller is configured to send to
- E-mail inbox server accessible using IMAP or POP3 protocols
- Wireless network access the e-mail inbox for mobile devices
- iPhone (iPhone 4S or greater recommended) or iPad (3rd generation or greater)
- iRConnect Application from the iPhone Application Store

17.3 SETUP AND CONFIGURATION

Option R818: iRConnect is required to load in order to use this feature. iRConnect is also available as a Demonstration Option. Once this option is loaded, go to MENU >> SETUP >> iRConnect. The iRConnect screen is shown as in Figure 17.3 . By default *All Emails* are disabled. Please enable to activate the feature.

The screenshot shows the 'SETUP iRConnect' screen. At the top, there are status indicators: 'Busy', 'Stop', 'Halt', 'Fault', 'Run', 'I/O', 'Prod', and 'Stop'. The main title is 'MAIN LINE 0 AUTO ABORTED JOINT' with a '10%' indicator. Below the title, there are two tabs: 'SETUP iRConnect' (selected) and 'SETUP iRConnect Emails'. The 'SETUP iRConnect' tab shows a list of 9 items, each with a status: 1 All Emails: ENABLED, 2 Periodic Status: <*DETAIL*>, 3 Alarms: ENABLED, 4 TP Change Log: DISABLED, 5 Var Change Log: DISABLED, 6 Process Logger Monitor: Not Loaded, 7 Vision Images: Not Loaded, 8 iRDiagnosics: <*DETAIL*>, 9 System Clock: <*DETAIL*>. The 'SETUP iRConnect Emails' tab shows a list of 14 items: 1 Ethernet Port: Port #1, 2 Mail Server: 172.20.12.214, 3 To Addr: my_server@xyz.com, 4 Alert Addr: 2473777000@txt.att., 5 Cc Addr: my_email@xyz.com, 6 From Addr: my_robot@xyz.com, 7 Subject: iRConnect Message, 8 Plant Name: FANUC, 9 Line Name: PDE, 10 Cluster Name: SYSTEM, 11 Queue Timer: 900 sec, 12 Queue Size: 10, 13 Alrm Throttle: 30 per min, 14 Msg Throttle: 6 per min. At the bottom, there are buttons: [TYPE], DISP_CFG, DISP_VAR, ENABLED, and DISABLED.

Fig. 17.3 Enable iRConnect

17.3.1 iRConnect Screen

Several parameters are used to configure iRConnect. Table 17.3.1 lists and describes the parameters.

Table 17.3.1 iRConnect Screen Items

ITEM	DESCRIPTION
All Emails	This item will enable iRConnect E-mails. You will have to cycle power the first time this item is set. If you wish to disable all E-mails, then set to disabled. To view and set the E-mail settings, press i + FCTN and select iRConnect Details from the Related View menu. See iRConnect Emails screen.
Periodic Status	This item will bring up the related view to configure the periodic status E-mail which is sent on power up and each period subsequently. The related view menu allows you to set the subject and status timer in seconds.
Alarms	This item allows you to disable all alarms from being sent. Other iRConnect E-mails will continue to be sent. When this item is enabled, then alarms will be sent according to the related view menu or the configuration file. The related view menu allows you to enable or disable entire categories of alarms. The categories are the same as that shown in the ALARM Logs.
TP Change Log	This item allows you to enable or disable the TP change log. If enabled, then changes made to TP programs are queued and periodically sent. See Queue Timer and Queue Size in the iRConnect Emails screen.
Var Change Log	This item allows you to enable or disable the Var change log. Refer to Section 17.5.
Process Logger Monitor	This item will bring up the related view to configure Process Logger Monitor for iRConnect. If the option is not loaded, then it will show "Not Loaded".
Vision Images	This item will bring up the related view to configure Images for iRConnect. You will be sent E-mails when the IRVICONN macro call is executed. If iRVision is not loaded, then it will show "Not Loaded".
iRDiagnosics	This item will bring up the related view to configure iRDiagnosics Emails for iRConnect. iRConnect will always load iRDiagnosics.

ITEM	DESCRIPTION
System Clock	This item will bring up the related view to configure the System Clock. To send messages to the mobile application with the correct time, verify the System Clock is set correctly and use the ADV function key to verify the Daylight Saving Time and Timezone settings. If NTP server is not enabled and set, then you must manually set the correct local time and correct time zone.
DISP_CFG	The DISP_CFG function key will show the current configuration file. If a power cycle has occurred since the file was changed, then it will also reflect the current configuration being used by iRConnect.
DISP_VAR	The DISP_VAR function key will show the current Variable Change configuration file, if found. If a power cycle has occurred since the file was changed, then it will also reflect the current Variable Change configuration being used by iRConnect.

17.3.2 iRConnect Emails Screen

Several parameters are used to configure iRConnect Emails. Table 17.3.2 lists and describes the parameters. If rcmcfg.xml is used to configure iRConnect, then most of these items are configured in the XML file and cannot be changed in this screen.

Table 17.3.2 iRConnect Emails Screen Items

ITEM	DESCRIPTION
Ethernet Port	Ethernet Port to be used
Mail Server	SMTP/MAIL server IP address
To Addr	Receiving E-mail address
Alert Addr	SMS gateway E-mail address which arrives as a text message on phone
Cc Addr	Copy to E-mail address (optional)
From Addr	Address of sender (optional). If not provided the default address is FRA_ROBOT.
Subject	Subject line for E-mail
Plant Name	Plant name (optional)
Line Name	Line name in the plant (optional)
Cluster Name	Cluster name of the line in the plant (optional)
Queue Timer	E-mails are queued and sent only when the queue timer or the queue size is reached, whichever is first. The queue timer is in seconds. E-mails are language dependent so when the controller language changes, any queued E-mails are sent immediately.
Queue Size	The number of E-mails that should be queued before the E-mails are sent.
Alarm Throttle	The Alarm Throttle determines the number of alarms that can be sent each minute. When the limit is reached, the alarm is posted, "HOST-013 iRConnect email throttled". Throttling will continue silently until the current minute (from first send) is over. Then the throttle resets itself and starts the process over. Each alarm to be sent increases the count even if it is put into the queue to be sent later. To turn off the Alarm Throttle, set the value to 0.
Msg Throttle	The Message Throttle determines the number of E-mails, excluding alarms, that can be sent each minute from each source. Each source of E-mails, such as iRVision, will have its own throttle. When the limit is reached, it is up to the source to determine if an alarm is posted. Throttling will continue until the current minute (from first send) is over. Then the throttle resets itself and starts the process over. Each E-mail to be sent increases the count for that source even if it is put into the queue to be sent later. To turn off the Message Throttle for all sources except alarms, set the value to 0.

17.3.3 Configuration File

iRConnect has a default rcmcfg.xml file. The file will be copied to your default device if you press FUNCTION SAVE while in the iRConnect menu.

To set up iRConnect using a custom configuration file, modify rcmcfg.xml, and copy to the FR:¥ device. Your custom file will be used instead of the default file after you cycle power on the controller. Once you have a custom file, then FUNCTION SAVE will now copy FR:¥rcmcfg.xml to your default device. This file is also backed up and restored. Each time you modify FR:¥rcmcfg.xml, you must cycle power for the changes to take effect.

To remove your custom configuration file, delete the file from FR:¥ and the default rcmcfg.xml file will be used after the next power cycle.

The XML tag definitions are described in Table 17.3.3 .

Table 17.3.3 Tag Definitions

Tag	Parent Tag	Depth	Definition
<RCMCFG>	None	0	This is root tag.
<PLANT>	<RCMCFG>	1	Tag to specify the plant information. It has one attribute: 1. <i>name</i> unique across plants
<LINE>	<PLANT>	2	Tag to specify the line information in the plant. It is child of PLANT tag. It has one attribute: 1. <i>name</i> attribute to specify name of the lines in the plant.
<CLUSTER>	<LINE>	3	Tag to specify cluster information of the line in the plant. It is child of LINE tag. It has three attributes: 1. <i>name</i> attribute to specify name of the cluster. 2. <i>ipgroup</i> attribute to specify the first 3 octets of IP address. 3. <i>ipset</i> defines the SET of last octet of IP addresses. The <i>ipset</i> could be specified in two ways, either in range if all robots in cluster has continuous IP addresses e.g. <i>ipset</i> ="200-300" or in set of last octets otherwise e.g. <i>ipset</i> ="123, 178, 200, 201, 223".
<ETHPORT>	<RCMCFG>	1	Tag to specify the Ethernet port to be used. Port%d is expected.
<SMTP>	<RCMCFG>	1	Tag to specify the E-mail configuration including MAIL Server information.
<SERVER>	<SMTP>	2	This tag defines SMTP/MAIL server IP address
<SUBJECT>	<SMTP>	2	Subject Line for E-mail to be send on errors
<TO_ADDRESS>	<SMTP>	2	Receiving E-mail address. It can carry multiple E-mail addresses separated by semicolon (;).
<ALERT_ADDRESS>	<SMTP>	2	SMS gateway E-mail address which arrives as a text message on phone. It can carry multiple E-mail addresses separated by semicolon (;).
<FROM_ADDRESS>	<SMTP>	2	Address of sender (optional) if not provided from address is default address which is FRA_ROBOT

Tag	Parent Tag	Depth	Definition
<CC_ADDRESS>	<SMTP>	2	Copy to E-mail address (optional). This field also can carry multiple E-mail addresses separated by semicolon (;).
<SEVERITY>	<RCMCFG>	1	Defines ALL errors of this severity. It has four attributes: 1. <i>name</i> of severity - "NONE", "WARN", "PAUSE", "STOP", "SERVO", "SERVO2", "ABORT", "SYSTEM", "NO DISP", " " 2. <i>global</i> - If TRUE then all errors which have global severity. If FALSE then all errors with local severity. This attribute applies only to "PAUSE", "STOP", and "ABORT" severities, otherwise the attribute is ignored. 3. <i>send</i> - If TRUE then E-mail is sent immediately on that error. 4. <i>alert</i> - If TRUE then E-mail is sent immediately on that error and alert text message is sent to phone.
<CATEGORY>	<RCMCFG>	1	Defines ALL errors of this category. It has three attributes: 1. 'name' of category - "MOTION", "APPL", "PWD", "COMM", "SYSTEM" 2. <i>send</i> - If TRUE then E-mail is sent immediately on that error. 3. <i>alert</i> - If TRUE then E-mail is sent immediately on that error and alert text message is sent to phone.
<FILE>	<SEVERITY> <CATEGORY> <ERROR>	2/3	Defines files to be attached with error. File name is included in tag text not in attribute.
<EMBED>	<SEVERITY> <CATEGORY> <ERROR>	2/3	Defines files to be embedded in E-mail body with specified error. File name is included in tag text not in attribute.
<ERTYPE>	<RCMCFG>	1	Defines specific Error category i.e. facility code. It has <i>name</i> attribute to specify facility code.
<ERROR>	<ERTYPE>	2	Defines specific errors. It has seven attributes: 1. <i>code</i> to specify error number. It could be '*' if <i>sev</i> is specified. 2. <i>sev</i> to define error severity. 3. <i>global</i> - If TRUE then all errors which have global severity. If FALSE then all errors with local severity. 4. <i>send</i> - If TRUE then E-mail is sent immediately on that error. 5. <i>alert</i> - If TRUE then E-mail is sent immediately on that error and alert text message is sent to phone. 6. <i>freq</i> to define frequency of an alarm before sending. 7. <i>dur</i> to define duration of time before sending an alarm.

Below is sample configuration file. It must be named *rcmcf.xml*.

```
<!-- This file is processed one time at startup. Tags are always English. - >
<?xml version=" 1.0" ?>
<RCMCFG>
<!-- Plant, Line, and Cluster configuration details - >
  <PLANT name=" shap" >
    <LINE name = "west-paint">
      <CLUSTER name ="basecoat" ipgroup="172.22.194"
        ipset="100-300" />
      <CLUSTER name ="clearcoat"
        ipgroup="172.22.200" ipset="5-14" />
    </LINE>
    <LINE name = "west-body">
      <CLUSTER name ="Left Fender"
        ipgroup="192.168.2" ipset="100,102,104,106" />
      <CLUSTER name ="Right Fender"
        ipgroup="172.22.200" ipset="101,103,107,222" />
    </LINE>
  </PLANT>
  <!-- Ethernet port in use. This does not mean you force
  E_mail client to send E-mail via configured port. It is only
  for port information ROBOT is identified among other robots- >
  <ETHPORT> Port 1</ETHPORT>
  <!-- - Email Information - >
  <SMTP>
    <SERVER>10.1.2.4</SERVER>
    <SUBJECT> RCM: iRConect Message </SUBJECT>
    <!-- TO_ADDRESS field is compulsory otherwise E-mail
    is not delivered - >
    <TO_ADDRESS>junk@junk.com</TO_ADDRESS>
    <CC_ADDRESS>junk_cc@junk.com</CC_ADDRESS>
    <FROM_ADDRESS>junk_fr@junk.com</FROM_ADDRESS>
  </SMTP>
  <!-- Severity based configuration - >
  <!-- All errors of SYSTEM severity and global scope.
  Set alert to TRUE.- >
  <SEVERITY name ="SYSTEM" global ="TRUE" alert=" TRUE" >
    <FILE>md:errall.ls />
    <FILE>td:systest.dt />
  </SEVERITY>
  <!-- This tells iRConnect to send E-mail immediately if error of STOP
  severity and local scope occurs - >
  <SEVERITY name ="STOP" global ="FALSE" send=" TRUE" >
    <FILE md:errcomm.ls />
    <EMBED name="td:sysmsg.emb"/>
  </SEVERITY>
  <!-- This tells iRConnect to send E-mail if error of PAUSE severity
  and local scope occurs - >
  <SEVERITY name ="PAUSE" global ="FALSE">
    <FILE name="MD:errcomm.ls" />
    <FILE name="MD:errall.ls" />
  </SEVERITY>
```

Continued...

```

<!-- Category based configuration -->
<CATEGORY name ="APPL" send="TRUE" alert="TRUE">
  <FILE name="MD:errall.ls" />
  <FILE name="MD:conslog.dg" />
</CATEGORY>
<CATEGORY name ="SYSTEM">
  <FILE name="MD:errall.ls" />
  <FILE name="MD:conslog.dg" />
</CATEGORY>
<!-- Power message startup, do not include if you
      don' t need power up message - >
<ERTYPE name ="SYST">
  <ERROR code ="26">
    <FILE name="MD:errhist.ls"/>
    <FILE name="MD:errmot.ls"/>
  </ERROR>
</ERTYPE>
<!-- Error based configuration- >
<!-- This tells to send email immediately if SRVO-223
      occurs and set alert - >
<ERTYPE name ="SRVO">
<ERROR code ="223" alert=" TRUE" send=" TRUE" >
  <FILE name="MD:errall.ls"/>
  <FILE name="MD:conslog.ls"/>
</ERROR>
<!-- Wants if SRVO-58 occurs - >
<ERROR code ="58">
  <FILE name="MD:hist.ls"/>
</ERROR>
</ERTYPE> <ERTYPE name ="PRIO">
  <ERROR code=" *" >
  </ERROR>
  <!-- Send flag enables iRConnect to send E_mail immediately
        on this error- >
  <ERROR code ="230" send =" TRUE" alert=" TRUE" >
    <FILE name="MD:summary.dg"/>
    <EMBED name="mc:eiptest.emb" />
  </ERROR>
</ERTYPE>
<!-- Wants all SYST errors which has local scope - >
<ERTYPE name=" SYST" >
  <ERROR code ="*" sev="SERVO" global="FALSE">
    <FILE name="MD:ethernet.dg"/>
  </ERROR>
</ERTYPE>
</RCMCFG>

```

17.3.4 Matching Alarms

A message is queued for send when at least one of the above XML tags match the given alarm. The attachments and embed files are pulled for each match. So if an alarm matches a severity tag and a category tag and an ertype/error tag, then 1 message is queued for the given alarm with attachments and embed files for each match. The attachments and embed files are checked for duplicates so only 1 copy of each is queued. If any of the matches have the *send* attribute true, then the E-mail is sent.

17.3.5 Attachments

Attachments are different. They are sent by E-mail task (SMTP). Therefore, attachment files will not be deleted until the actual E-mail is sent. All files (except for MD: device) are deleted after the send occurs. This was added special for iRConnect because normally attachments are not deleted by SMTP.

17.4 TEACH PENDANT PROGRAMS

Teach pendant programs can send data to iRConnect through a call to one of four KAREL programs; IRC_STATUS, IRC_COUNTER, IRC_MSG, and IRC_STLABEL. These programs create and send an e-mail in the format that iRConnect expects.

17.4.1 Program: IRC_STATUS

The IRC_Status program updates the status indicator for the robot controller in iRConnect.

Table 17.4.1 IRC_STATUS Parameters

Name	Type	Description
Status	INTEGER	One of three status colors: 1 = Green 2 = Orange 3 = Red
Label	STRING[32]	Label text to describe the current status

```
1: CALL IRC_STATUS(2, 'Paused')
```

17.4.2 Program: IRC_COUNTER

The IRC_Counter program allows a teach pendant program to sent production data to iRConnect. Each call to IRC_COUNTER updates the current count for the counter name and current hourly period. iRConnect can support up to three different counters.

Table 17.4.2 IRC_COUNTER Parameters

Name	Type	Description
Hourly Count	INTEGER	Total hourly count for the current period
Counter Name	STRING[32]	Counter name displayed in iRConnect

```
1: CALL IRC_COUNTER(634, 'Part 1')
```

17.4.3 Program: IRC_MSG

The IRC_MSG program allows a general message to be sent from a teach pendant. General messages appear in iRConnect's general messages dashboard gadget.

Table 17.4.3 IRC_MSG Parameters

Name	Type	Description
Message Type	STRING[32]	Message label used by iRConnect to group messages.
Message Text	STRING[32]	Message text to display
Background Color	Integer	One of four colors: 0 = White 1 = Green 2 = Orange 3 = Red

```
1: CALL IRC_MSG('Process 1', 'Process 1 restarted', 0)
```

17.4.4 Program: IRC_STLABEL

The IRC_STLABEL program sends a message that updates the details of the controller status. The program accepts a background color and two lines of text.

Table 17.4.4 IRC_STLABEL Parameters

Name	Type	Description
Line 1 Background Color	INTEGER	One of four background colors: 0 = Clear 1 = Green 2 = Orange 3 = Red
Line 1 Text	STRING[32]	Text to display in line one
Line 2 Background Color	INTEGER	One of four background colors: 0 = Clear 1 = Green 2 = Orange 3 = Red
Line 2 Text	STRING[32]	Text to display in line one

```
1: CALL IRC_STLABEL(2, 'Program SPOT1 Paused', 3, 'Invalid IO Config')
```

17.5 VARIABLE CHANGE SETUP AND USE

The variable change monitor is used to scan certain variables and notify the user when there is a change. The list of variables to be monitored is defined in an XML file. On startup this file is parsed and the monitor structure built. A scan utility is called periodically to watch for changes to the listed variables. When one is changed, an E-mail describing the change is queued. iRConnect provides a queue timer and queue size to send E-mail whichever occurs first.

The types of variables that can be monitored include:

- System Variables
- Numeric Registers
- String Registers
- Position Registers
- Frames

- Program Variables

17.5.1 Variable Change Setup

Setup is required to use this service. First, all necessary iRConnect options must be enabled. In the iRConnect menu, 'All Emails' and 'Var Change Log' must be enabled.

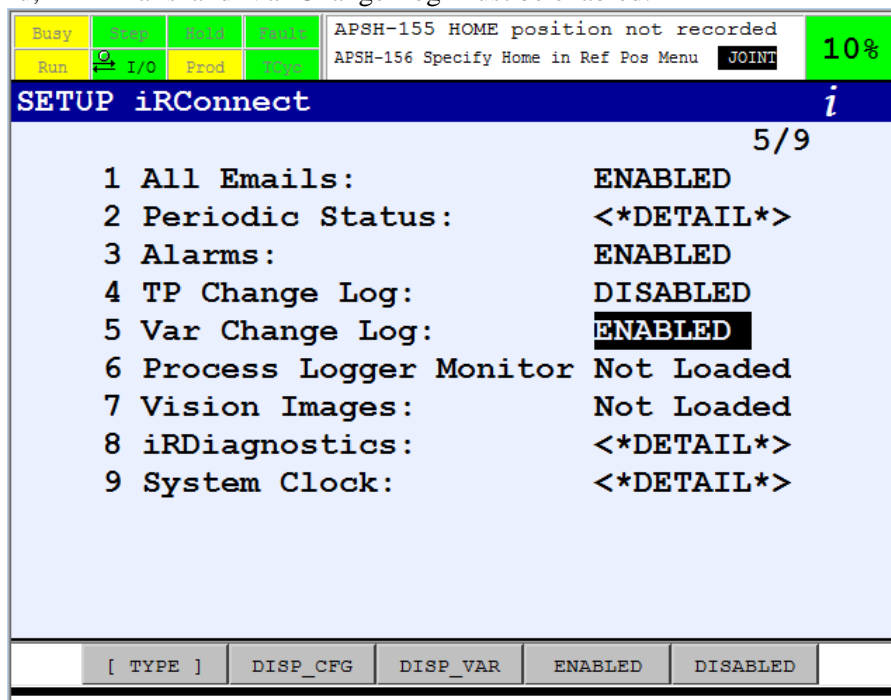


Fig. 17.5.1 iRConnect Setup Menu

Second, FR:¥varchg.xml must be configured. This is a configuration file that defines the variables to be monitored and specifies the output. Create varchg.xml offline and copy to the FR:¥ device. This file will be backed up and restored. Each time you modify FR:¥varchg.xml, you must cycle power for the changes to take effect.

<CHANGELOG> is the root tag. For each variable, a VARCHG tag is defined.

Each VARCHG tag has the following attributes:

- var — This attribute is REQUIRED. It is the system-defined name of the variable.
- program — This attribute is OPTIONAL if the VARCHG tag is describing a system variable, position register, numeric register, or string register. Otherwise it is REQUIRED. This is the name of the program the variable belongs to.
- title — This attribute is OPTIONAL. It is used to define the title, a user defined description of the variable. The default title is [prog_name]var_name.
- dict — This attribute is OPTIONAL. It is used along with ele_no to define a dictionary entry. This entry can be used instead of title to describe the title in multiple languages. You must omit the title attribute to use a dictionary entry.
- ele_no — This attribute is OPTIONAL. It is an integer. See the description for dict.
- subtype — This attribute is OPTIONAL. It is used to define subtype, which is used to classify the change notification in the iRConnect application. The default subtype is the title.
- subt_dict — This attribute is OPTIONAL. It is used along with subt_ele to define a dictionary entry. This entry can be used instead of subtype to describe the subtype in multiple languages. You must omit the subtype attribute to use a dictionary entry.
- subt_ele — This attribute is OPTIONAL. It is an integer. See the description for subt_dict.
- alert — This attribute is OPTIONAL. It can be either 0 or 1. It is used to describe the alert status in the iRConnect application. Unlike other messages, alert does not immediately send the change nor send an alert text message to the phone. It defaults to 0.

This service also includes the ability to exclude monitoring particular elements of an array or complex data structure. This can be useful in certain situations. To configure, an EXCLUDE tag is needed. EXCLUDE tags are children of a VARCHG tag.

Each EXCLUDE tag has one attribute:

- **field** — This attribute is REQUIRED. It is the system-defined name of the excluded element. The definition of field must be some sub-element of the variable described in the parent VARCHG tag. If the variable is an array of structure, then the fields are excluded from all structures in the array.

IMPORTANT: EXCLUDE tags can only be used on arrays or complex variables.

Below is an example of what FR:¥varchg.xml could look like.

```
<?xml version="1.0" ?>
<CHANGELOG>
<VARCHG program="my_prog" var="my_var" />
<VARCHG program="PKMAIN" var="parts_picked"
    dict="PICK" ele_no="12"
    subt_dict="PICK" subt_ele="11"/>
<VARCHG var="$POSREG[1,1]" title="Job 1 Positions"
    subtype="Job Vars" alert="1"/>
<VARCHG var="$NUMREG[1]" dict="X0PT" ele_no="6"/>
<VARCHG var="$STRREG[1]" dict="X0PT" ele_no="7"
    subt_dict="X0PT" subt_ele="20"/>
<VARCHG var="$UI_CONFIG.$TIMEOUT"
    title="iPendant Time Out"
    subtype="UI Configuration"/>
<VARCHG var="$UI_CONFIG.$READONLY[2]"
    title="Navigate iPendant RW Access"
    subtype="UI Configuration"/>
<VARCHG var="$STYLE_NAME">
<EXCLUDE field="$STYLE_NAME[2]"/>
<EXCLUDE field="$STYLE_NAME[3]"/>
<EXCLUDE field="$STYLE_NAME[9]"/>
<EXCLUDE field="$STYLE_NAME[10]"/>
</VARCHG>
<VARCHG var="$SCR_GRP" title="Motion Parameters" >
<EXCLUDE field="$MCH_POS_X"/>
<EXCLUDE field="$MCH_POS_Y"/>
<EXCLUDE field="$MCH_POS_Z"/>
<EXCLUDE field="$MCH_POS_W"/>
<EXCLUDE field="$MCH_POS_P"/>
<EXCLUDE field="$MCH_POS_R"/>
<EXCLUDE field="$MCH_ANG"/>
<EXCLUDE field="$MCH_SPD"/>
<EXCLUDE field="$DST_MIR_P"/>
<EXCLUDE field="$DPOS_DST"/>
<EXCLUDE field="$DST_POS_X"/>
<EXCLUDE field="$DST_POS_Y"/>
<EXCLUDE field="$DST_POS_Z"/>
<EXCLUDE field="$DAC_AXMODE"/>
</VARCHG>
</CHANGELOG>
```

17.5.2 Variable Change Output

Changes made to the monitored variables are queued and periodically sent via E-mail to the iRConnect mobile application. See Queue Timer and Queue Size in the iRConnect Emails screen. The iRConnect mobile application will display the changes, including new values of the variables and in most cases the old values.

The variables are scanned every 60 seconds. To change the scan time, edit \$RCMCFG.\$VARCHG_TIME from the System Variables menu. The minimum scan time is 15 seconds.

APPENDIX

A

DIAGNOSTIC INFORMATION

A.1 VERIFYING NETWORK CONNECTIONS

A.1.1 Overview

There are two basic tools for verifying network connections:

- Ethernet status LEDs
- PING

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

Refer to Section A.1.2 for more information about Ethernet status LEDs.

Refer to Section A.1.3 for more information about the PING utility.

A.1.2 Ethernet Status LEDs

The Ethernet status LEDs at the Ethernet RJ45 connector on the robot will indicate if the robot is connected to anything based on the link LED. Most Ethernet switches and other equipment will have similar LEDs indicating a physical connection. If the LINK LED is off then there is no Ethernet connectivity at all. This generally implies a disconnected or bad cable or bad connections. For more information about the Ethernet status LEDs, refer to Section A.2. The robot will auto-negotiate by default. The speed and duplex used on either interface is shown under the STATUS screen for interface #1 or interface #2. To do this, Press the [MENU] key, select Setup, press F1, [TYPE], select Host Comm, select TCP/IP, press NEXT, and then press Status for the desired interface.

A.1.3 PING Utility

PING is a network utility that sends a request to a specific IP address and expects a response. The request is essentially "Can you hear me?" The destination node will send a response that it received the request. The requesting node will either receive the response or timeout. PING is a basic network utility that is included with most operating systems, such as Windows and Unix, and is also supported on the robot. Even devices that do not support generating PING requests will normally respond to the PING request.

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

The PING utility is also available from any windows PC. Use Procedure A-2 .

Procedure A-1 Using PING on the Robot

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

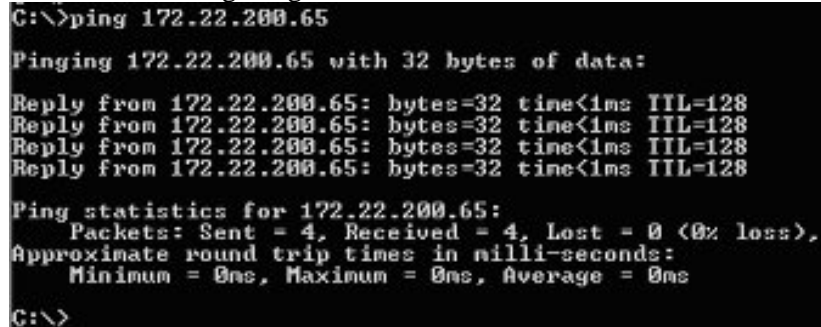
The prompt line on the teach pendant will indicate if the PING was successful, or if the PING request timed out.

Procedure A-2 Using PING on a Windows PC

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

PING 192.168.0.10

The following image shows a successful PING.



```
C:\>ping 172.22.200.65

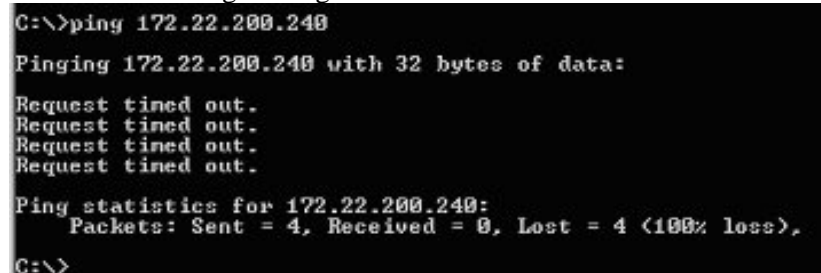
Pinging 172.22.200.65 with 32 bytes of data:

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 172.22.200.240

Pinging 172.22.200.240 with 32 bytes of data:

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

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

C:\>
```

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 2.4 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 Fig. A.2 .

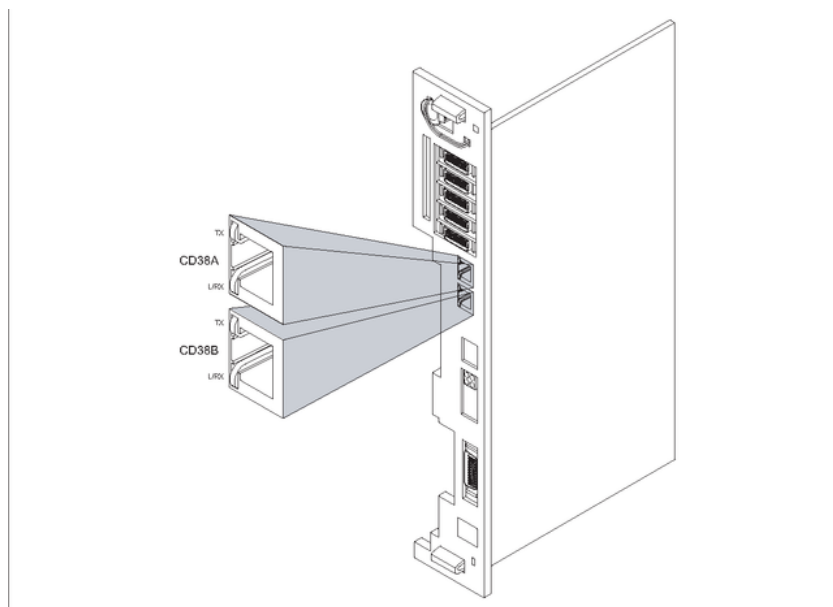


Fig. A.2 Ethernet LEDs

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.

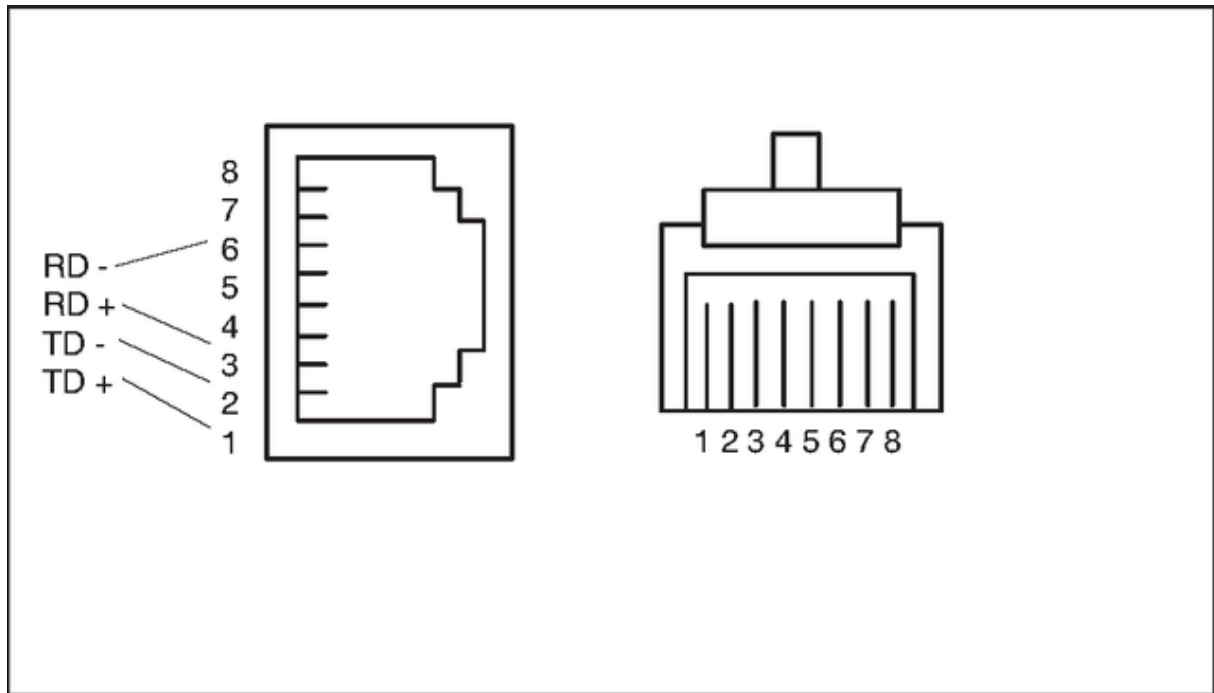


Fig. A.3 10 Base-T/100 Base-TX Connector Pin Assignments

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. We recommends limiting the broadcast domain to only the control devices 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.

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 EtherNet I/P 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).

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 port managed switch – <http://www.cisco.com/web/JP>
- Cisco 2955 (industrialized version of 2950) – <http://www.cisco.com/web/JP>
- Phoenix Contact (managed/unmanaged industrial switch) – <http://www.phoenixcontact.co.jp>
- SIEMENS (managed/unmanaged industrial switch) – <http://www.siemens.co.jp>

NOTE

Overload of Ethernet network on robot controller will cause the alarm "HOST-244 Ethernet Throttle". Overload of Ethernet network will cause the alarm "SRVO-352 DCS COUNT 1 alarm". Please verify network design and make an action referring this section.

D

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.

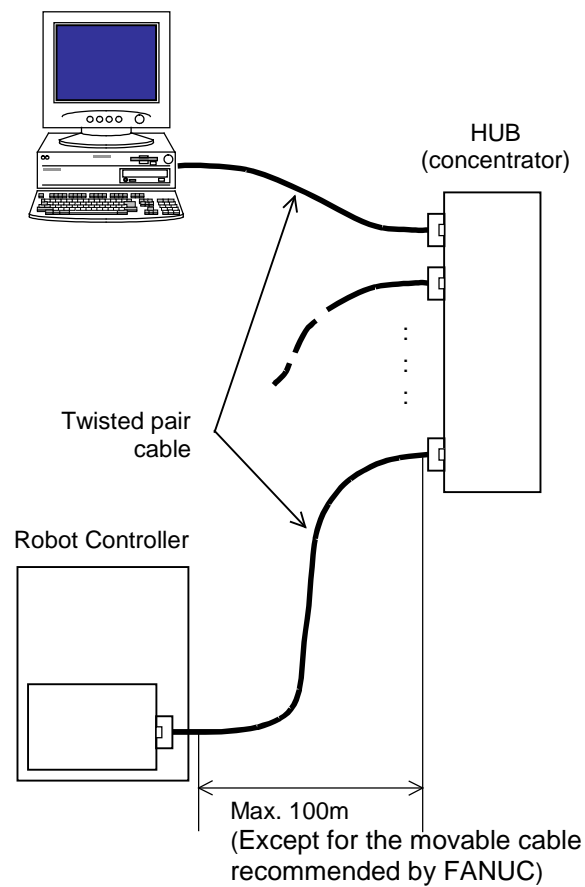
We cannot ensure operation that is influenced by network trouble caused by a device other than the robot controller.

D.1 CONNECTING TO Ethernet

Main board of controller is provided with a 100BASE-TX interface.

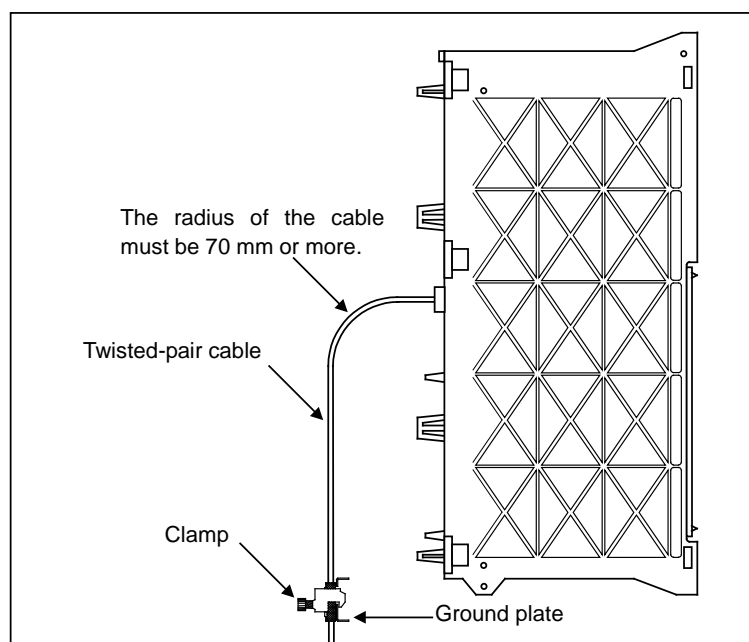
Prepare a hub for connecting the FAST Ethernet board to the Ethernet trunk. The following shows an example of a general connection.

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.



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 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.

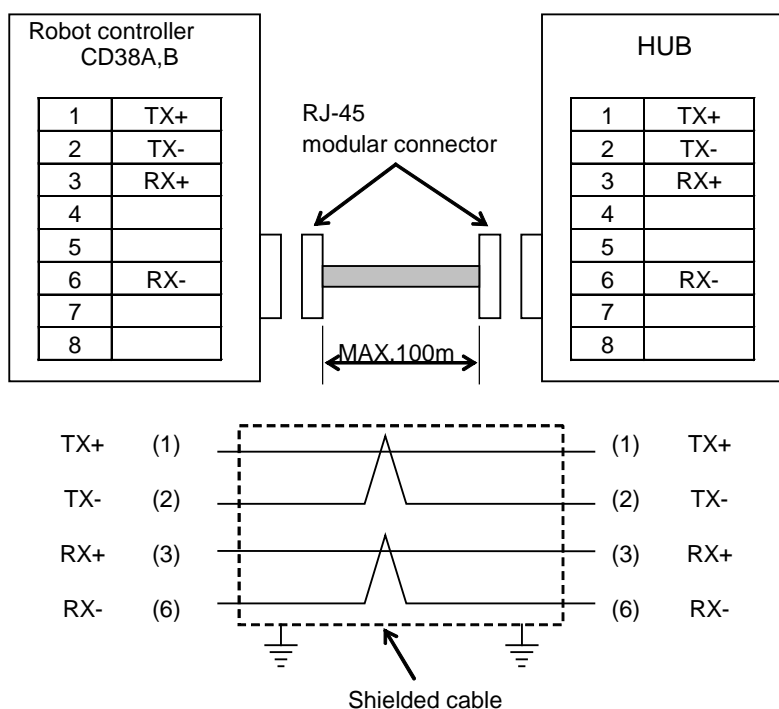
D.3 100BASE-TX CONNECTOR (CD38A/CD38B) PIN ASSIGNMENTS

CD38A CD38B		
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

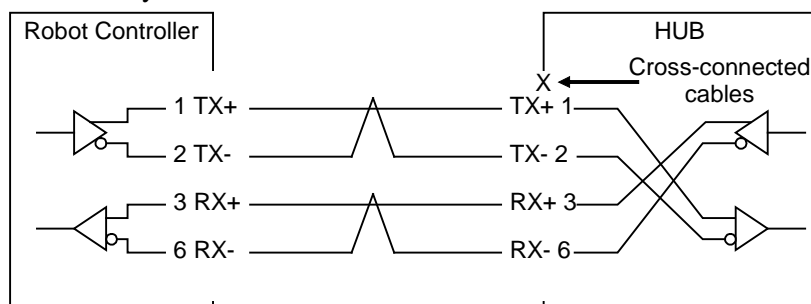
D.4.1 Cable Connection

The cable used for connection between the 100BASE-TX interface, CD38A or CD38B 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 following figure shows the cable connection when cables are crossed in the hub. Controllers other than R-30iA Mate has Auto MDI/MDI-X function so cable selection of which straight cable or cross cable is unnecessary.



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 or more twisted-pair cable (STP cable) to improve the resistance to electrical noise in an FA environment.

Recommended Cables

Manufacturer	Specification	Remarks
NISSEI ELECTRIC CO., LTD.	F-4PWVWMF	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 or more.
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 \pm 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 must be used.

Specification	Manufacturer	Remarks
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 or more.

For assembly with a cable, contact HIROSE ELECTRIC CO., LTD. directly.

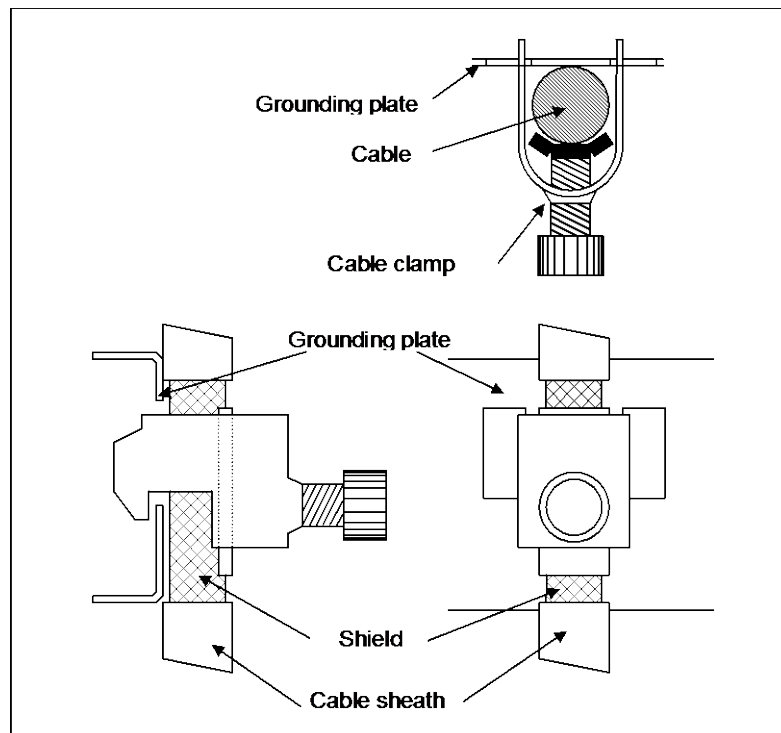
(From HIROSE ELECTRIC CO., LTD., "TM21CP-88P(03) Connection Procedure Manual (Technical Specification No. ATAD-E2367)" is available as a technical document.)

D.5 ANTI-NOISE MEASURES

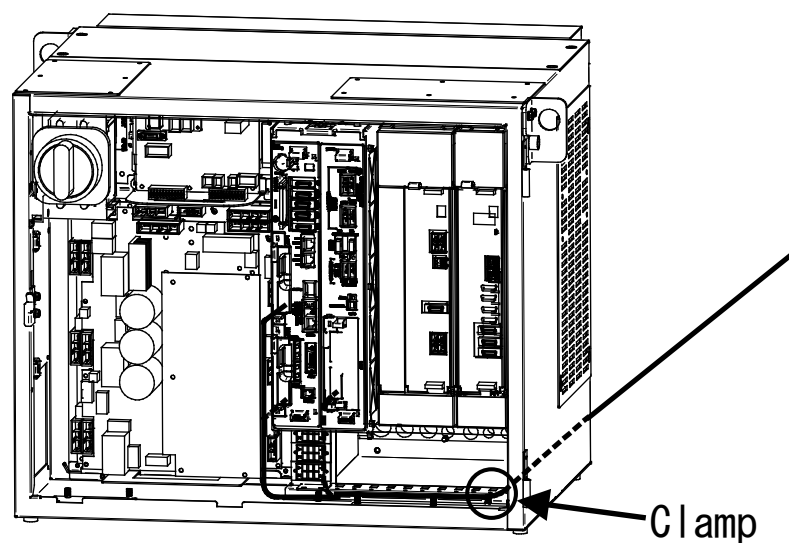
D.5.1 Clamping and Shielding of Cables

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.

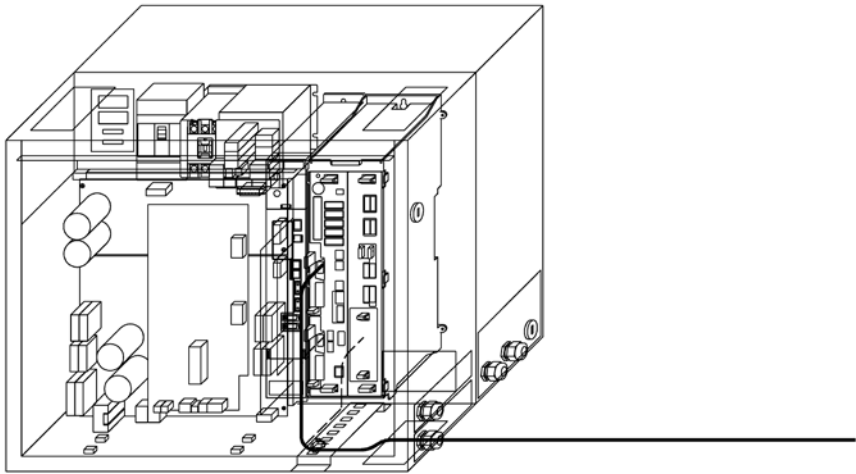
As shown in the figure, strip a part of the cable sheath to expose the metal shield and push the shield against the grounding plate with the clamping hardware.

**NOTE**

- 1 Be sure to clamp and shield the cable to ensure the stable operation of the system.
- 2 Upon completion of cabling, perform a communication test sufficiently not only before but also after system operation to ensure anti-noise measures.

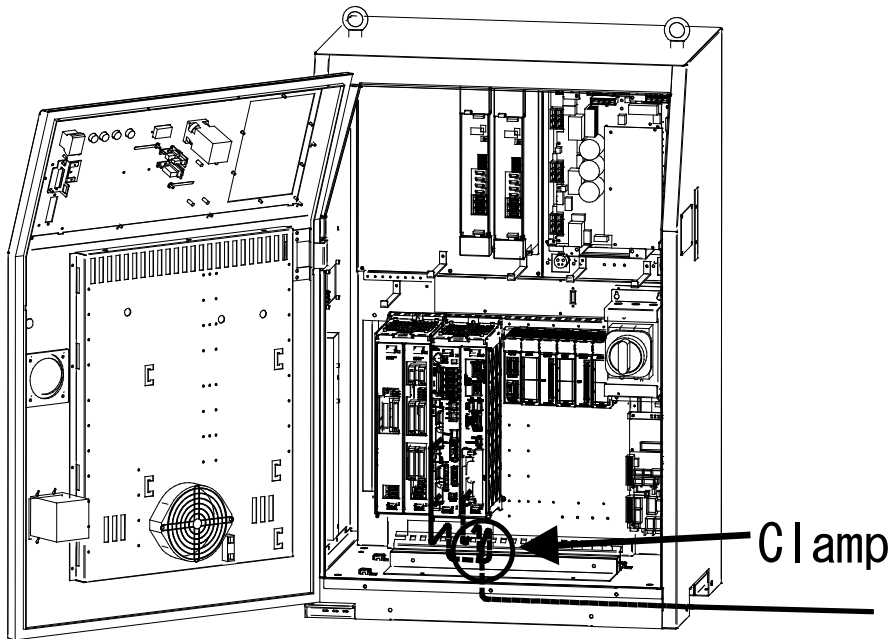
A-Cabinet

R-30iB/ R-30iB Plus

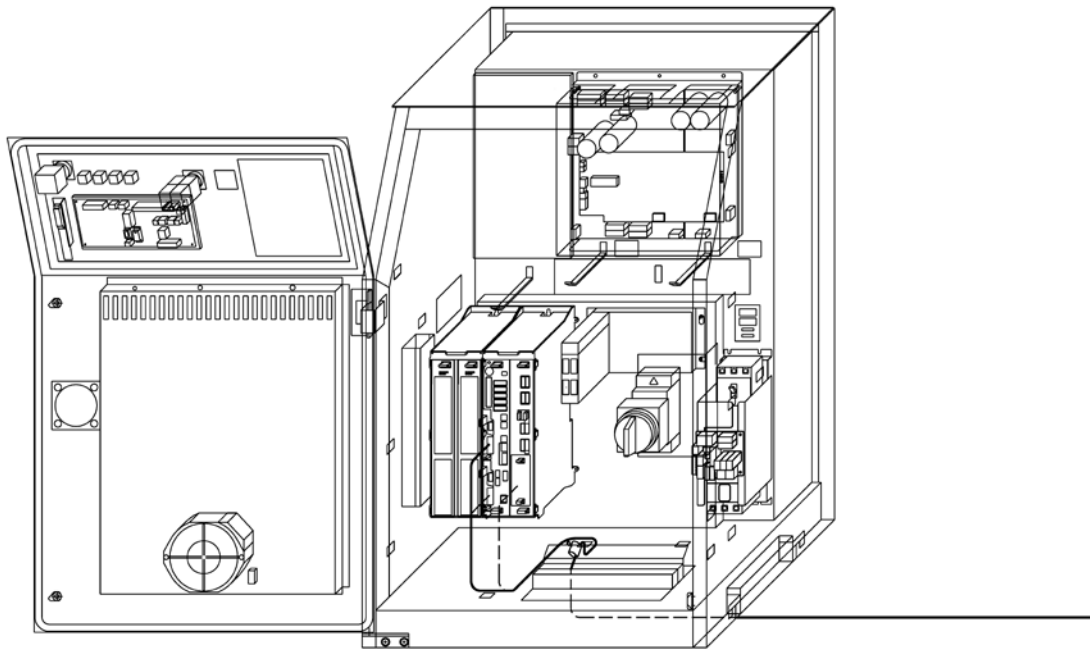
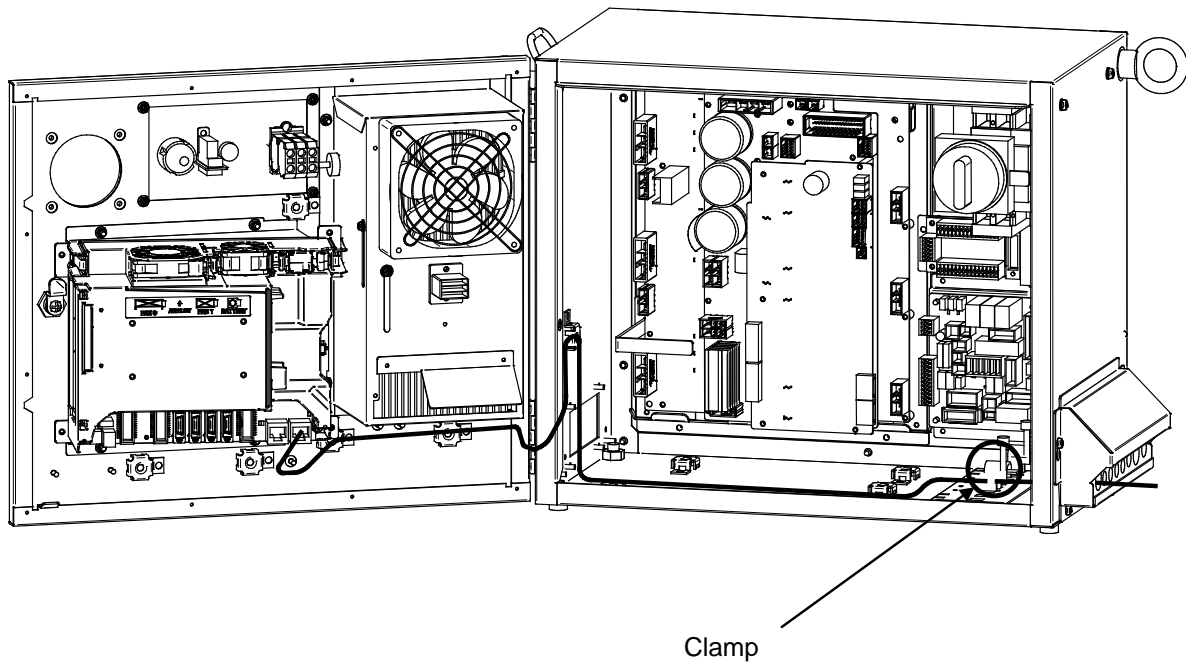


B-cabinet

R-30iA

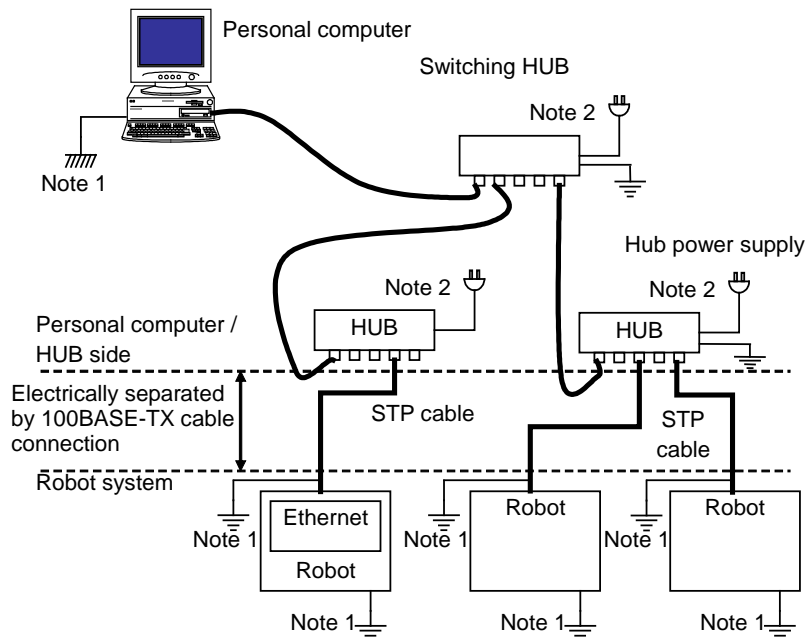


R-30iB/ R-30iB Plus

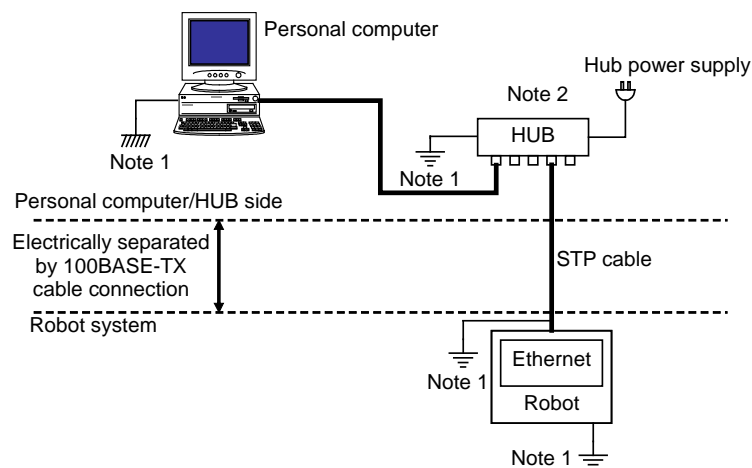
**R-30iA****Mate-cabinet****R-30iB Mate/R-30iB Mate Plus/R-30iA Mate**

D.5.2 Grounding the Network

Even if the grounding condition on the machine side is satisfied, the communication line can pick up noise from the machine, depending on the machine installation condition and environment, thus resulting in a communication error. To protect against such noise, the robot / HUB / personal computer should be separated and insulated. Examples of connection are given below.



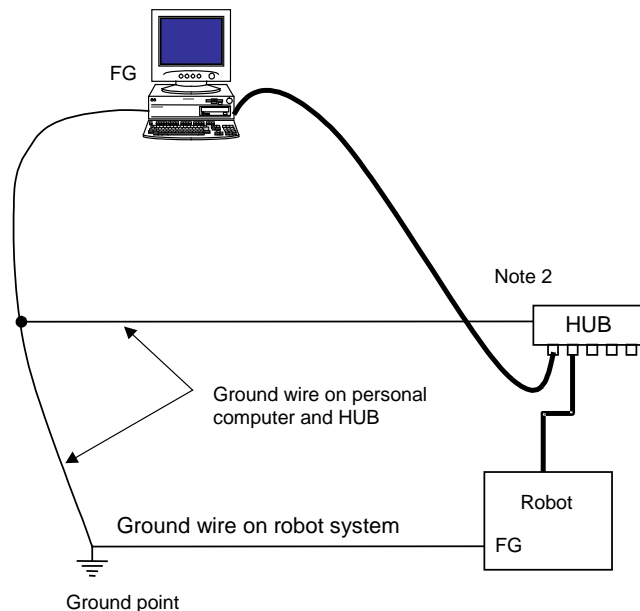
Large-Scale Network



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 or more	
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-)	
	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.	

INDEX

<Number>

10 BASE-T/100 Base T-X CONNECTOR PIN ASSIGNMENTS	206
100BASE-TX CONNECTOR (CD38A/CD38B) PIN ASSIGNMENTS	215

<#>

#ECHO Command	67
#EXEC Command	68
#IF, #ELIF, #ELSE, #ENDIF	69
#INCLUDE Command	68
#PRINTENV Command	70
#SET Command	69

<A>

A KAREL Client Application	115
A KAREL Server Application	117
Access Denied	25
Access Description	33,35,139
Access Levels	24
ACCESSING AND USING CLIENT DEVICES	33
ACCESSING AND USING PC SHARE CLIENT DEVICES	139
Accessing iPendant screens through the web server	74
ACCESSING SERVER DEVICES	35
ACCESSING USER PROGRAM, SETUP, AND DIAGNOSTIC INFORMATION	41
Advanced DHCP Setup	105
ADVANCED iPendant FUNCTIONS	146
ANSI C Loopback Client Example	119
ANTI-NOISE MEASURES	217
Application File Backup and Restore	2
Architecture	5
ASSOCIATED OPTIONS	131
Attachments	196

Background editing guidelines	163
BACKUP AND RESTORE	2
Blocking Downloads of Certain File Groups	36
BOOTP AND TFTP PROTOCOLS	3

<C>

CABLE CONNECTION	213,215
Cable Materials	216
Caution for Setting IP Address	23
Changing IP Addresses	80
CHECK ITEMS AT INSTALLATION	223
Clamping and Shielding of Cables	217
Configuration File	192
CONFIGURATION OF PROXY SERVER	76
CONFIGURE FTP WITH A KAREL COMMAND FILE	207

CONFIGURE THE REMOTE PC	142
Configuring Internet Explorer ®	146
CONFIGURING NETWORK PARAMETERS WITH A KAREL COMMAND FILE	207
Configuring PC Share	138
Configuring the P2, and P3	79
CONFIGURING THE SOCKET MESSAGING OPTION	109
Connecting to a Robot Home Page	52
Connecting to a Telnet Server	50
CONNECTING TO Ethernet	213
Connection	173,175,176,180
CONNECTION TO REMOTE iPendant	173
Connector Specification	217
Creating Web Pages Based on KAREL Programs	57
Customizing Diagnostic Files, Variable File Listings, and TP Program Listings	55
Customizing Your Robot Home Page	54

<D>

DEFINING DNS PARAMETERS	45
Devices	6
DHCP Setup	103
DHCP SYSTEM VARIABLES	107
DHCP TROUBLESHOOTING	108
DIAGNOSTIC INFORMATION	203
Directory Services	39
DISPLAYING iPendant KEY SHEET	167
DISPLAYING THE ETHERNET HARDWARE (MAC) ADDRESS	13
DOMAIN NAME SERVICE (DNS)	3,45
DYNAMIC HOST CONFIGURATION PROTOCOL4,103	

<E>

Editing guidelines	163
Environment Services	38
Error Log Files	43
ERRORS RETURNED BY THE PROXY SERVER	77
Ethernet Hardware (MAC) Address	13
Ethernet Hardware (MAC) Address Locations	18
ETHERNET LEDS	205
ETHERNET PACKET SNIFFER	4,126
Ethernet Status LEDS	203
Example Configuration	26,74
Examples	37
Exchanging Data during a Socket Messaging Connection	114

<F>

FANUC SERVER ACCESS CONTROL (FSAC)	24
Features	37
Features of the Robot DHCP Client	103
FILE ACCESS	130

- File Specification for Client Devices33
- File Specification for PC Share Client Devices..... 140
- FILE TRANSFER PROTOCOL (FTP).....2
- File Transfer Services39
- FTP CLIENT USERNAMES AND PASSWORDS.....32
- FTP OPERATIONS27
- FTP SERVICES37
- FTP Transfer Log44
- <G>**
 - Global Variables65
 - Grounding the Network.....221
 - Guidelines for a Good Implementation114
 - GUIDELINES FOR USING ETHERNET211
- <H>**
 - Hardware Requirements7,109
 - HARDWARE REQUIREMENTS AND
INSTALLATION7
 - Horizontal *i*Pendant Key Sheet168
 - HOST COMMUNICATIONS.....5
 - HTTP AUTHENTICATION71
- <I>**
 - Installation of *i*Pendant Controls.....146
 - Introduction to DHCP103
 - i*RCONNECT189
 - i*RConnect Emails Screen.....191
 - i*RConnect Screen.....190
- <J>**
 - Jog Operation185
 - JOGGING *i*PENDANT (JITP).....179
 - JOGGING *i*Pendant FUNCTION.....164
- <K>**
 - Keys161,167,177
- <L>**
 - LEADING OUT THE Ethernet CABLE.....214
 - Limitations186
 - Limitations (general).....186
 - Limitations of *i*Pendant Key Sheet.....169
 - Limitations of Jogging *i*Pendant Function (JITP)187
 - Limitations of Monitor *i*Pendant Function (JECHO) ...187
 - Limitations of Navigate *i*Pendant Function (JCGTP) ..187
 - LIMITATIONS(GENERAL).....169
 - Limitations(Jogging *i*Pendant function)170
 - Limitations(Monitor *i*Pendant function).....169
 - Limitations(Navigate *i*Pendant function)170
 - Local Variables66
- <M>**
 - Matching Alarms.....195
 - Miscellaneous FTP Information40
 - MONITOR *i*Pendant (JECHO)175
 - MONITOR *i*Pendant FUNCTION157
 - MSG_CONNECT(string, integer).....113
 - MSG_DISCO(string, integer)114
 - MSG_PING(string, integer)114
- <N>**
 - NAVIGATE *i*Pendant (JCGTP).....176
 - NAVIGATE *i*Pendant FUNCTION159
 - NETWORK DESIGN AND PERFORMANCE211
 - NETWORK DESIGN CONSIDERATIONS132
 - NETWORK PERFORMANCE114
- <O>**
 - Operation72,158,160,164
 - Operation of Proxy Server75
 - OPTION CONFIGURATION171
 - OVERVIEW1,2,5,7,13,24,27,35,37,41,45,48,
51,52,63,71,72,75,78,81,103,109,113,114,115,121,126,
128,135,146,157,159,164,172,175,176,179,189,203
- <P>**
 - Passwords.....142
 - PASSWORDS AND SECURITY142
 - PC connection to the controller.....158,160,164
 - PC SHARE.....135
 - PING Utility.....203
 - POINT-TO-POINT PROTOCOL CONNECTIVITY .4,78
 - Program: IRC_COUNTER196
 - Program: IRC_MSG.....196
 - Program: IRC_STATUS196
 - Program: IRC_STLABEL.....197
 - PROGRAMMING EXAMPLES115
 - PROXY SERVER.....4,75
- <R>**
 - REMOTE *i*Pendant FUNCTION172
 - REQUIREMENTS189
 - Requirements for Using Proxy Server.....75
 - RIPE SETUP.....128
 - Robot controller password option enabled.....73
 - Robot controller password option not enabled.....73
 - ROS INTERFACE PACKETS OVER ETHERNET
(RIPE)4,128
 - Running KAREL Programs from the Web Browser57
- <S>**
 - SAFETY PRECAUTIONSs-1
 - Saving FANUC *i*RVision log to PC share folder.....141
 - Screens177,182
 - Screens (with *i*PendantControls).....183
 - Screens (without *i*PendantControls).....182
 - SERVER SIDE INCLUDES.....63
 - Setting up a Client Tag.....111
 - Setting up a Server Tag.....110
 - SETTING UP AND STARTING FTP27
 - SETTING UP AND STARTING PC SHARE135
 - SETTING UP DHCP ON THE ROBOT.....103
 - Setting Up PC Share Client Tags135

Setting up PPP on a Network PC	81
SETTING UP PPP ON YOUR CONTROLLER	78
SETTING UP PPP ON YOUR PC	81
SETTING UP SNTP	121
SETTING UP TCP/IP	7,19
SETTING UP TELNET ON YOUR ROBOT	48
SETTING UP THE ETHERNET PACKET SNIFFER	126
SETTING UP THE WEB SERVER	51
Settings	173
SETUP	146
SETUP AND CONFIGURATION	189
Show keys	179
SIMPLE NETWORK TIME PROTOCOL (SNTP)	4,121
SOCKET MESSAGING	4,109
SOCKET MESSAGING AND KAREL	113
Software Requirements	109
SSI EXAMPLES	71
Starting and Stopping a Client Device	34,140
Starting and Stopping a Server Device	36
String Substitution	66
Supported Security Protocols	142
SYNCHRONIZED TIMING	132
Syntax	64
System Files	42
SYSTEM REQUIREMENTS	109
System Variables	25

<T>

TCP/IP PROTOCOL	3
Teach Pendant File Access	35,141
TEACH PENDANT PROGRAMS	196
TELNET	3,48,132
Telnet Setup	48
Testing the Network Connection	156
TROUBLESHOOTING	125
Troubleshooting jogging iPendant connection	167
Troubleshooting remote connection	157
TWISTED-PAIR CABLE SPECIFICATION	215

<U>

Using FANUC Server Access Control (FSAC) to Control Access to the Web Server	52
USING SNTP	123
USING THE RING BUFFER AND TRIGGERS	127
USING THE WEB SERVER	52

<V>

VARIABLE ACCESS	132
Variable Change Output	199
Variable Change Setup	198
VARIABLE CHANGE SETUP AND USE	197
VERIFYING NETWORK CONNECTIONS	203
Vertical iPendant Key Sheet	168

<W>

WEB SERVER	3,51
------------------	------

<X>

XML CONFIGURATION FILE	131
------------------------------	-----

REVISION RECORD

Edition	Date	Contents
04	Apr., 2018	<ul style="list-style-type: none">Supported R-30iB Plus/R-30iB Mate Plus controllerSocket message supports UDPMSG_CONN is fixed to MSG_CONNECTAdd chapter 16 Remote iPendant function
03	Feb., 2014	<ul style="list-style-type: none">Supported R-30iB Mate controller, iRConnect, and enhanced vision Ethernet port.
02	Oct., 2012	<ul style="list-style-type: none">Supported R-30iB controller
01	Aug., 2008	

B-82974EN/04

