FANUC America Corporation SYSTEM R-30*i*B Controller Internet Options Setup and Operations Manual

MAROBIN8304141E REV D

Version V8.30

FANUC America Corporation
3900 W. Hamlin Road
Rochester Hills, Michigan 48309–3253

About This Manual

Copyrights and Trademarks

This new publication contains proprietary information of FANUC America Corporation, furnished for customer use only. No other uses are authorized without the express written permission of FANUC America Corporation.

FANUC America Corporation
3900 W. Hamlin Road
Rochester Hills, MI 48309-3253

The descriptions and specifications contained in this manual were in effect at the time this manual was approved. FANUC America Corporation, hereinafter referred to as FANUC America, reserves the right to discontinue models at any time or to change specifications or design without notice and without incurring obligations.

FANUC America's manuals present descriptions, specifications, drawings, schematics, bills of material, parts, connections and/or procedures for installing, disassembling, connecting, operating and programming FANUC America Corporation's products and/or systems. Such systems consist of robots, extended axes, robot controllers, application software, the KAREL® programming language, INSIGHT® vision equipment, and special tools.

FANUC America recommends that only persons who have been trained in one or more approved FANUC America Training Course(s) be permitted to install, operate, use, perform procedures on, repair, and/or maintain FANUC America's products and/or systems and their respective components. Approved training necessitates that the courses selected be relevant to the type of system installed and application performed at the customer site.

Warning

This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. As temporarily permitted by regulation, it has not been tested for compliance with the limits for Class A computing devices pursuant to subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference. Operation of the equipment in a residential area is likely to cause interference, in which case the user, at his own expense, will be required to take whatever measure may be required to correct the interference.

FANUC America conducts courses on its systems and products on a regularly scheduled basis at its headquarters in Rochester Hills, Michigan. For additional information contact

FANUC America Corporation
3900 W. Hamlin Road
Rochester Hills, MI 48309-3253

www.fanucamerica.com

For customer assistance, including Technical Support, Service, Parts & Part Repair, and Marketing Requests, contact the Customer Resource Center, 24 hours a day, at 1-800-47-ROBOT (1-800-477-6268). International customers should call 011-1-248-377-7159.

Send your comments and suggestions about this manual to: product.documentation@fanucamerica.com

Patents

One or more of the following U.S. patents might be related to the FANUC America products described in this manual.

FRA Patent List

4,630,567 4,639,878 4,707,647 4,708,175 4,708,580 4,942,539 4,984,745 5,238,029 5,239,739 5,272,805 5,293,107 5,293,911 5,331,264 5,367,944 5,373,221 5,421,218 5,434,489 5,644,898 5,670,202 5,696,687 5,737,218 5,823,389 5,853,027 5,887,800 5,941,679 5,959,425 5,987,726 6,059,092 6,064,168 6,070,109 6,086,294 6,122,062 6,147,323 6,204,620 6,243,621 6,253,799 6,285,920 6,313,595 6,325,302 6,345,818 6,356,807 6,360,143 6,378,190 6,385,508 6,425,177 6,477,913 6,490,369 6,518,980 6,540,104 6,541,757 6,560,513 6,569,258 6,612,449 6,703,079 6,705,361 6,726,773 6,768,078 6,845,295 6,945,483 7,149,606 7,149,606 7,211,978 7,266,422 7,399,363

FANUC LTD Patent List

4,571,694 4,626,756 4,700,118 4,706,001 4,728,872 4,732,526 4,742,207 4,835,362 4,894,596 4,899,095 4,920,248 4,931,617 4,934,504 4,956,594 4,967,125 4,969,109 4,970,370 4,970,448 4,979,127 5,004,968 5,006,035 5,008,834 5,063,281 5,066,847 5,066,902 5,093,552 5,107,716 5,111,019 5,130,515 5,136,223 5,151,608 5,170,109 5,189,351 5,267,483 5,274,360 5,292,066 5,300,868 5,304,906 5,313,563 5,319,443 5,325,467 5,327,057 5,329,469 5,333,242 5,337,148 5,371,452 5,375,480 5,418,441 5,432,316 5,440,213 5,442,155 5,444,612 5,449,875 5,451,850 5,461,478 5,463,297 5,467,003 5,471,312 5,479,078 5,485,389 5,485,552 5,486,679 5,489,758 5,493,192 5,504,766 5,511,007 5,520,062 5,528,013 5,532,924 5,548,194 5,552,687 5,558,196 5,561,742 5,570,187 5,570,190 5,572,103 5,581,167 5,582,750 5,587,635 5,600,759 5,608,299 5,608,618 5,624,588 5,630,955 5,637,969 5,639,204 5,641,415 5,650,078 5,658,121 5,668,628 5,687,295 5,691,615 5,698,121 5,708,342 5,715,375 5,719,479 5,727,132 5,742,138 5,742,144 5,748,854 5,749,058 5,760,560 5,773,950 5,783,922 5,799,135 5,812,408 5,841,257 5,845,053 5,872,894 5,887,122 5,911,892 5,912,540 5,920,678 5,937,143 5,980,082 5,983,744 5,987,591 5,988,850 6,023,044 6,032,086 6,040,554 6,059,169 6,088,628 6,097,169 6,114,824 6,124,693 6,140,788 6,141,863 6,157,155 6,160,324 6,163,124 6,177,650 6,180,898 6,181,096 6,188,194 6,208,105 6,212,444 6,219,583 6,226,181 6,236,011 6,236,896 6,250,174 6,278,902 6,279,413 6,285,921 6,298,283 6,321,139 6,324,443 6,328,523 6,330,493 6,340,875 6,356,671 6,377,869 6,382,012 6,384,371 6,396,030 6,414,711 6,424,883 6,431,018 6,434,448 6,445,979 6,459,958 6,463,358 6,484,067 6,486,629 6,507,165 6,654,666 6,665,588 6,680,461 6,696,810 6,728,417 6,763,284 6,772,493 6,845,296 6,853,881 6,888,089 6,898,486 6,917,837 6,928,337 6,965,091 6,970,802 7,038,165 7,069,808 7,084,900 7,092,791 7,133,747 7,143,100 7,149,602 7,131,848 7,161,321 7,171,041 7,174,234 7,173,213 7,177,722 7,177,439 7,181,294 7,181,313 7,280,687 7,283,661 7,291,806 7,299,713 7,315,650 7,324,873 7,328,083 7,330,777 7,333,879 7,355,725 7,359,817 7,373,220 7,376,488 7,386,367 7,464,623 7,447,615 7,445,260 7,474,939 7,486,816 7,495,192 7,501,778 7,502,504 7,508,155 7,512,459 7,525,273 7,526,121

VersaBell, ServoBell and SpeedDock Patents Pending.

Conventions

This manual includes information essential to the safety of personnel, equipment, software, and data. This information is indicated by headings and boxes in the text.

Warning

Information appearing under WARNING concerns the protection of personnel. It is boxed and in bold type to set it apart from other text.

Caution

Information appearing under **CAUTION** concerns the protection of equipment, software, and data. It is boxed to set it apart from other text.

Note Information appearing next to **NOTE** concerns related information or useful hints.

Contents

List of Figures

List of Tables

Safety

FANUC America Corporation is not and does not represent itself as an expert in safety systems, safety equipment, or the specific safety aspects of your company and/or its work force. It is the responsibility of the owner, employer, or user to take all necessary steps to guarantee the safety of all personnel in the workplace.

The appropriate level of safety for your application and installation can best be determined by safety system professionals. FANUC America Corporation therefore, recommends that each customer consult with such professionals in order to provide a workplace that allows for the safe application, use, and operation of FANUC America Corporation systems.

According to the industry standard ANSI/RIA R15-06, the owner or user is advised to consult the standards to ensure compliance with its requests for Robotics System design, usability, operation, maintenance, and service. Additionally, as the owner, employer, or user of a robotic system, it is your responsibility to arrange for the training of the operator of a robot system to recognize and respond to known hazards associated with your robotic system and to be aware of the recommended operating procedures for your particular application and robot installation.

Ensure that the robot being used is appropriate for the application. Robots used in classified (hazardous) locations must be certified for this use.

FANUC America Corporation therefore, recommends that all personnel who intend to operate, program, repair, or otherwise use the robotics system be trained in an approved FANUC America Corporation training course and become familiar with the proper operation of the system. Persons responsible for programming the system-including the design, implementation, and debugging of application programs-must be familiar with the recommended programming procedures for your application and robot installation.

The following guidelines are provided to emphasize the importance of safety in the workplace.

CONSIDERING SAFETY FOR YOUR ROBOT INSTALLATION

Safety is essential whenever robots are used. Keep in mind the following factors with regard to safety:

- The safety of people and equipment
- Use of safety enhancing devices
- Techniques for safe teaching and manual operation of the robot(s)
- Techniques for safe automatic operation of the robot(s)
- Regular scheduled inspection of the robot and workcell
- Proper maintenance of the robot

Keeping People Safe

The safety of people is always of primary importance in any situation. When applying safety measures to your robotic system, consider the following:

- External devices
- Robot(s)
- Tooling
- Workpiece

Using Safety Enhancing Devices

Always give appropriate attention to the work area that surrounds the robot. The safety of the work area can be enhanced by the installation of some or all of the following devices:

- Safety fences, barriers, or chains
- Light curtains
- Interlocks
- Pressure mats
- Floor markings
- Warning lights
- Mechanical stops
- EMERGENCY STOP buttons
- DEADMAN switches

Setting Up a Safe Workcell

A safe workcell is essential to protect people and equipment. Observe the following guidelines to ensure that the workcell is set up safely. These suggestions are intended to supplement and **not** replace existing federal, state, and local laws, regulations, and guidelines that pertain to safety.

- Sponsor your personnel for training in approved FANUC America Corporation training course(s) related to your application. Never permit untrained personnel to operate the robots.
- Install a lockout device that uses an access code to prevent unauthorized persons from operating the robot.
- Use anti-tie-down logic to prevent the operator from bypassing safety measures.
- Arrange the workcell so the operator faces the workcell and can see what is going on inside the cell.

- Clearly identify the work envelope of each robot in the system with floor markings, signs, and special barriers. The work envelope is the area defined by the maximum motion range of the robot, including any tooling attached to the wrist flange that extend this range.
- Position all controllers outside the robot work envelope.
- Never rely on software or firmware based controllers as the primary safety element unless they comply with applicable current robot safety standards.
- Mount an adequate number of EMERGENCY STOP buttons or switches within easy reach of the operator and at critical points inside and around the outside of the workcell.
- Install flashing lights and/or audible warning devices that activate whenever the robot is operating, that is, whenever power is applied to the servo drive system. Audible warning devices shall exceed the ambient noise level at the end-use application.
- Wherever possible, install safety fences to protect against unauthorized entry by personnel into the work envelope.
- Install special guarding that prevents the operator from reaching into restricted areas of the work envelope.
- Use interlocks.
- Use presence or proximity sensing devices such as light curtains, mats, and capacitance and vision systems to enhance safety.
- Periodically check the safety joints or safety clutches that can be optionally installed between the robot wrist flange and tooling. If the tooling strikes an object, these devices dislodge, remove power from the system, and help to minimize damage to the tooling and robot.
- Make sure all external devices are properly filtered, grounded, shielded, and suppressed to prevent hazardous motion due to the effects of electro-magnetic interference (EMI), radio frequency interference (RFI), and electro-static discharge (ESD).
- Make provisions for power lockout/tagout at the controller.
- Eliminate *pinch points* . Pinch points are areas where personnel could get trapped between a moving robot and other equipment.
- Provide enough room inside the workcell to permit personnel to teach the robot and perform maintenance safely.
- Program the robot to load and unload material safely.
- If high voltage electrostatics are present, be sure to provide appropriate interlocks, warning, and beacons.
- If materials are being applied at dangerously high pressure, provide electrical interlocks for lockout of material flow and pressure.

Staying Safe While Teaching or Manually Operating the Robot

Advise all personnel who must teach the robot or otherwise manually operate the robot to observe the following rules:

- Never wear watches, rings, neckties, scarves, or loose clothing that could get caught in moving machinery.
- Know whether or not you are using an intrinsically safe teach pendant if you are working in a hazardous environment.
- Before teaching, visually inspect the robot and *work envelope* to make sure that no potentially hazardous conditions exist. The work envelope is the area defined by the maximum motion range of the robot. These include tooling attached to the wrist flange that extends this range.
- The area near the robot must be clean and free of oil, water, or debris. Immediately report unsafe working conditions to the supervisor or safety department.
- FANUC America Corporation recommends that no one enter the work envelope of a robot that is on, except for robot teaching operations. However, if you must enter the work envelope, be sure all safeguards are in place, check the teach pendant DEADMAN switch for proper operation, and place the robot in teach mode. Take the teach pendant with you, turn it on, and be prepared to release the DEADMAN switch. Only the person with the teach pendant should be in the work envelope.

Warning

Never bypass, strap, or otherwise deactivate a safety device, such as a limit switch, for any operational convenience. Deactivating a safety device is known to have resulted in serious injury and death.

- Know the path that can be used to escape from a moving robot; make sure the escape path is never blocked.
- Isolate the robot from all remote control signals that can cause motion while data is being taught.
- Test any program being run for the first time in the following manner:

Warning

Stay outside the robot work envelope whenever a program is being run. Failure to do so can result in injury.

  — Using a low motion speed, single step the program for at least one full cycle.

  — Using a low motion speed, test run the program continuously for at least one full cycle.

  — Using the programmed speed, test run the program continuously for at least one full cycle.

- Make sure all personnel are outside the work envelope before running production.

Staying Safe During Automatic Operation

Advise all personnel who operate the robot during production to observe the following rules:

- Make sure all safety provisions are present and active.
- Know the entire workcell area. The workcell includes the robot and its work envelope, plus the area occupied by all external devices and other equipment with which the robot interacts.
- Understand the complete task the robot is programmed to perform before initiating automatic operation.
- Make sure all personnel are outside the work envelope before operating the robot.
- Never enter or allow others to enter the work envelope during automatic operation of the robot.
- Know the location and status of all switches, sensors, and control signals that could cause the robot to move.
- Know where the EMERGENCY STOP buttons are located on both the robot control and external control devices. Be prepared to press these buttons in an emergency.
- Never assume that a program is complete if the robot is not moving. The robot could be waiting for an input signal that will permit it to continue activity.
- If the robot is running in a pattern, do not assume it will continue to run in the same pattern.
- Never try to stop the robot, or break its motion, with your body. The only way to stop robot motion immediately is to press an EMERGENCY STOP button located on the controller panel, teach pendant, or emergency stop stations around the workcell.

Staying Safe During Inspection

When inspecting the robot, be sure to

- Turn off power at the controller.
- Lock out and tag out the power source at the controller according to the policies of your plant.
- Turn off the compressed air source and relieve the air pressure.
- If robot motion is not needed for inspecting the electrical circuits, press the EMERGENCY STOP button on the operator panel.
- Never wear watches, rings, neckties, scarves, or loose clothing that could get caught in moving machinery.
- If power is needed to check the robot motion or electrical circuits, be prepared to press the EMERGENCY STOP button, in an emergency.
- Be aware that when you remove a servomotor or brake, the associated robot arm will fall if it is not supported or resting on a hard stop. Support the arm on a solid support before you release the brake.

Staying Safe During Maintenance

When performing maintenance on your robot system, observe the following rules:

- Never enter the work envelope while the robot or a program is in operation.
- Before entering the work envelope, visually inspect the workcell to make sure no potentially hazardous conditions exist.
- Never wear watches, rings, neckties, scarves, or loose clothing that could get caught in moving machinery.
- Consider all or any overlapping work envelopes of adjoining robots when standing in a work envelope.
- Test the teach pendant for proper operation before entering the work envelope.
- If it is necessary for you to enter the robot work envelope while power is turned on, you must be sure that you are in control of the robot. Be sure to take the teach pendant with you, press the DEADMAN switch, and turn the teach pendant on. Be prepared to release the DEADMAN switch to turn off servo power to the robot immediately.
- Whenever possible, perform maintenance with the power turned off. Before you open the controller front panel or enter the work envelope, turn off and lock out the 3-phase power source at the controller.
- Be aware that an applicator bell cup can continue to spin at a very high speed even if the robot is idle. Use protective gloves or disable bearing air and turbine air before servicing these items.
- Be aware that when you remove a servomotor or brake, the associated robot arm will fall if it is not supported or resting on a hard stop. Support the arm on a solid support before you release the brake.

Warning

Lethal voltage is present in the controller WHENEVER IT IS CONNECTED to a power source. Be extremely careful to avoid electrical shock. HIGH VOLTAGE IS PRESENT at the input side whenever the controller is connected to a power source. Turning the disconnect or circuit breaker to the OFF position removes power from the output side of the device only.

- Release or block all stored energy. Before working on the pneumatic system, shut off the system air supply and purge the air lines.
- Isolate the robot from all remote control signals. If maintenance must be done when the power is on, make sure the person inside the work envelope has sole control of the robot. The teach pendant must be held by this person.
- Make sure personnel cannot get trapped between the moving robot and other equipment. Know the path that can be used to escape from a moving robot. Make sure the escape route is never blocked.

- Use blocks, mechanical stops, and pins to prevent hazardous movement by the robot. Make sure that such devices do not create pinch points that could trap personnel.

Warning

Do not try to remove any mechanical component from the robot before thoroughly reading and understanding the procedures in the appropriate manual. Doing so can result in serious personal injury and component destruction.

- Be aware that when you remove a servomotor or brake, the associated robot arm will fall if it is not supported or resting on a hard stop. Support the arm on a solid support before you release the brake.
- When replacing or installing components, make sure dirt and debris do not enter the system.
- Use only specified parts for replacement. To avoid fires and damage to parts in the controller, never use nonspecified fuses.
- Before restarting a robot, make sure no one is inside the work envelope; be sure that the robot and all external devices are operating normally.

KEEPING MACHINE TOOLS AND EXTERNAL DEVICES SAFE

Certain programming and mechanical measures are useful in keeping the machine tools and other external devices safe. Some of these measures are outlined below. Make sure you know all associated measures for safe use of such devices.

Programming Safety Precautions

Implement the following programming safety measures to prevent damage to machine tools and other external devices.

- Back-check limit switches in the workcell to make sure they do not fail.
- Implement “failure routines” in programs that will provide appropriate robot actions if an external device or another robot in the workcell fails.
- Use *handshaking* protocol to synchronize robot and external device operations.
- Program the robot to check the condition of all external devices during an operating cycle.

Mechanical Safety Precautions

Implement the following mechanical safety measures to prevent damage to machine tools and other external devices.

- Make sure the workcell is clean and free of oil, water, and debris.
- Use DCS (Dual Check Safety), software limits, limit switches, and mechanical hardstops to prevent undesired movement of the robot into the work area of machine tools and external devices.

KEEPING THE ROBOT SAFE

Observe the following operating and programming guidelines to prevent damage to the robot.

Operating Safety Precautions

The following measures are designed to prevent damage to the robot during operation.

- Use a low override speed to increase your control over the robot when jogging the robot.
- Visualize the movement the robot will make before you press the jog keys on the teach pendant.
- Make sure the work envelope is clean and free of oil, water, or debris.
- Use circuit breakers to guard against electrical overload.

Programming Safety Precautions

The following safety measures are designed to prevent damage to the robot during programming:

- Establish *interference zones* to prevent collisions when two or more robots share a work area.
- Make sure that the program ends with the robot near or at the home position.
- Be aware of signals or other operations that could trigger operation of tooling resulting in personal injury or equipment damage.
- In dispensing applications, be aware of all safety guidelines with respect to the dispensing materials.

Note Any deviation from the methods and safety practices described in this manual must conform to the approved standards of your company. If you have questions, see your supervisor.

ADDITIONAL SAFETY CONSIDERATIONS FOR PAINT ROBOT INSTALLATIONS

Process technicians are sometimes required to enter the paint booth, for example, during daily or routine calibration or while teaching new paths to a robot. Maintenance personal also must work inside the paint booth periodically.

Whenever personnel are working inside the paint booth, ventilation equipment must be used. Instruction on the proper use of ventilating equipment usually is provided by the paint shop supervisor.

Although paint booth hazards have been minimized, potential dangers still exist. Therefore, today’s highly automated paint booth requires that process and maintenance personnel have full awareness of the system and its capabilities. They must understand the interaction that occurs between the vehicle moving along the conveyor and the robot(s), hood/deck and door opening devices, and high-voltage electrostatic tools.

Caution

Ensure that all ground cables remain connected. Never operate the paint robot with ground provisions disconnected. Otherwise, you could injure personnel or damage equipment.

Paint robots are operated in three modes:

- Teach or manual mode
- Automatic mode, including automatic and exercise operation
- Diagnostic mode

During both teach and automatic modes, the robots in the paint booth will follow a predetermined pattern of movements. In teach mode, the process technician teaches (programs) paint paths using the teach pendant.

In automatic mode, robot operation is initiated at the System Operator Console (SOC) or Manual Control Panel (MCP), if available, and can be monitored from outside the paint booth. All personnel must remain outside of the booth or in a designated safe area within the booth whenever automatic mode is initiated at the SOC or MCP.

In automatic mode, the robots will execute the path movements they were taught during teach mode, but generally at production speeds.

When process and maintenance personnel run diagnostic routines that require them to remain in the paint booth, they must stay in a designated safe area.

Paint System Safety Features

Process technicians and maintenance personnel must become totally familiar with the equipment and its capabilities. To minimize the risk of injury when working near robots and related equipment, personnel must comply strictly with the procedures in the manuals.

This section provides information about the safety features that are included in the paint system and also explains the way the robot interacts with other equipment in the system.

The paint system includes the following safety features:

- Most paint booths have red warning beacons that illuminate when the robots are armed and ready to paint. Your booth might have other kinds of indicators. Learn what these are.
- Some paint booths have a blue beacon that, when illuminated, indicates that the electrostatic devices are enabled. Your booth might have other kinds of indicators. Learn what these are.
- EMERGENCY STOP buttons are located on the robot controller and teach pendant. Become familiar with the locations of all E-STOP buttons.
- An intrinsically safe teach pendant is used when teaching in hazardous paint atmospheres.
- A DEADMAN switch is located on each teach pendant. When this switch is held in, and the teach pendant is on, power is applied to the robot servo system. If the engaged DEADMAN switch is released during robot operation, power is removed from the servo system, all axis brakes are applied, and the robot comes to an EMERGENCY STOP. Safety interlocks within the system might also E-STOP other robots.

Warning

An EMERGENCY STOP will occur if the DEADMAN switch is released on a bypassed robot.

- Overtravel by robot axes is prevented by software limits. All of the major and minor axes are governed by software limits. DCS (Dual Check Safety), limit switches and hardstops also limit travel by the major axes.
- EMERGENCY STOP limit switches and photoelectric eyes might be part of your system. Limit switches, located on the entrance/exit doors of each booth, will EMERGENCY STOP all equipment in the booth if a door is opened while the system is operating in automatic or manual mode. For some systems, signals to these switches are inactive when the switch on the SOC is in teach mode.When present, photoelectric eyes are sometimes used to monitor unauthorized intrusion through the entrance/exit silhouette openings.
- System status is monitored by computer. Severe conditions result in automatic system shutdown.

Staying Safe While Operating the Paint Robot

When you work in or near the paint booth, observe the following rules, in addition to all rules for safe operation that apply to all robot systems.

> **Warning**
>
> **Observe all safety rules and guidelines to avoid injury.**

> **Warning**
>
> **Never bypass, strap, or otherwise deactivate a safety device, such as a limit switch, for any operational convenience. Deactivating a safety device is known to have resulted in serious injury and death.**

> **Warning**
>
> **Enclosures shall not be opened unless the area is known to be nonhazardous or all power has been removed from devices within the enclosure. Power shall not be restored after the enclosure has been opened until all combustible dusts have been removed from the interior of the enclosure and the enclosure purged. Refer to the Purge chapter for the required purge time.**

- Know the work area of the entire paint station (workcell).
- Know the work envelope of the robot and hood/deck and door opening devices.
- Be aware of overlapping work envelopes of adjacent robots.
- Know where all red, mushroom-shaped EMERGENCY STOP buttons are located.
- Know the location and status of all switches, sensors, and/or control signals that might cause the robot, conveyor, and opening devices to move.
- Make sure that the work area near the robot is clean and free of water, oil, and debris. Report unsafe conditions to your supervisor.
- Become familiar with the complete task the robot will perform BEFORE starting automatic mode.
- Make sure all personnel are outside the paint booth before you turn on power to the robot servo system.
- Never enter the work envelope or paint booth before you turn off power to the robot servo system.
- Never enter the work envelope during automatic operation unless a safe area has been designated.
- Never wear watches, rings, neckties, scarves, or loose clothing that could get caught in moving machinery.

- Remove all metallic objects, such as rings, watches, and belts, before entering a booth when the electrostatic devices are enabled.
- Stay out of areas where you might get trapped between a moving robot, conveyor, or opening device and another object.
- Be aware of signals and/or operations that could result in the triggering of guns or bells.
- Be aware of all safety precautions when dispensing of paint is required.
- Follow the procedures described in this manual.

Special Precautions for Combustible Dusts (powder paint)

When the robot is used in a location where combustible dusts are found, such as the application of powder paint, the following special precautions are required to insure that there are no combustible dusts inside the robot.

- Purge maintenance air should be maintained at all times, even when the robot power is off. This will insure that dust can not enter the robot.
- A purge cycle will not remove accumulated dusts. Therefore, if the robot is exposed to dust when maintenance air is not present, it will be necessary to remove the covers and clean out any accumulated dust. Do not energize the robot until you have performed the following steps.

1. Before covers are removed, the exterior of the robot should be cleaned to remove accumulated dust.
2. When cleaning and removing accumulated dust, either on the outside or inside of the robot, be sure to use methods appropriate for the type of dust that exists. Usually lint free rags dampened with water are acceptable. Do not use a vacuum cleaner to remove dust as it can generate static electricity and cause an explosion unless special precautions are taken.
3. Thoroughly clean the interior of the robot with a lint free rag to remove any accumulated dust.
4. When the dust has been removed, the covers must be replaced immediately.
5. Immediately after the covers are replaced, run a complete purge cycle. The robot can now be energized.

Staying Safe While Operating Paint Application Equipment

When you work with paint application equipment, observe the following rules, in addition to all rules for safe operation that apply to all robot systems.

Warning

When working with electrostatic paint equipment, follow all national and local codes as well as all safety guidelines within your organization. Also reference the following standards: *NFPA 33 Standards for Spray Application Using Flammable or Combustible Materials* , and *NFPA 70 National Electrical Code* .

- **Grounding**: All electrically conductive objects in the spray area must be grounded. This includes the spray booth, robots, conveyors, workstations, part carriers, hooks, paint pressure pots, as well as solvent containers. Grounding is defined as the object or objects shall be electrically connected to ground with a resistance of not more than 1 megohms.
- **High Voltage**: High voltage should only be on during actual spray operations. Voltage should be off when the painting process is completed. Never leave high voltage on during a cap cleaning process.
- Avoid any accumulation of combustible vapors or coating matter.
- Follow all manufacturer recommended cleaning procedures.
- Make sure all interlocks are operational.
- No smoking.
- Post all warning signs regarding the electrostatic equipment and operation of electrostatic equipment according to NFPA 33 Standard for Spray Application Using Flammable or Combustible Material.
- Disable all air and paint pressure to bell.
- Verify that the lines are not under pressure.

Staying Safe During Maintenance

When you perform maintenance on the painter system, observe the following rules, and all other maintenance safety rules that apply to all robot installations. Only qualified, trained service or maintenance personnel should perform repair work on a robot.

- Paint robots operate in a potentially explosive environment. Use caution when working with electric tools.
- When a maintenance technician is repairing or adjusting a robot, the work area is under the control of that technician. All personnel not participating in the maintenance must stay out of the area.
- For some maintenance procedures, station a second person at the control panel within reach of the EMERGENCY STOP button. This person must understand the robot and associated potential hazards.
- Be sure all covers and inspection plates are in good repair and in place.
- Always return the robot to the “home” position before you disarm it.

- Never use machine power to aid in removing any component from the robot.
- During robot operations, be aware of the robot’s movements. Excess vibration, unusual sounds, and so forth, can alert you to potential problems.
- Whenever possible, turn off the main electrical disconnect before you clean the robot.
- When using vinyl resin observe the following:
  - Wear eye protection and protective gloves during application and removal
  - Adequate ventilation is required. Overexposure could cause drowsiness or skin and eye irritation.
  - If there is contact with the skin, wash with water.
  - Follow the Original Equipment Manufacturer’s Material Safety Data Sheets.
- When using paint remover observe the following:
  - Eye protection, protective rubber gloves, boots, and apron are required during booth cleaning.
  - Adequate ventilation is required. Overexposure could cause drowsiness.
  - If there is contact with the skin or eyes, rinse with water for at least 15 minutes. Then, seek medical attention as soon as possible.
  - Follow the Original Equipment Manufacturer’s Material Safety Data Sheets.

Chapter 1

OVERVIEW

Contents

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.

Refer to the *Cimplicity HMI Operators Manual* for information on the Cimplicity/HMI option.

The FTP option is loaded by default with most 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 a KCL command window or 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)
- Image Backup and Restore over Ethernet, allowing the non-volatile memory areas of the robot to be backed up to a remote TFTP server (Chapter 12 *ETHERNET-BASED LOADING*)
- Remote access to the robot through a serial modem using the point to point protocol (PPP). (Chapter 8 *POINT-TO-POINT PROTOCOL CONNECTIVITY*)

The Advanced Internet Connectivity and Customization option includes :

- *i*Pendant Proxy Server, allowing the iPendant to browse outside of the robot to other web servers across the robot Ethernet connections

  (Chapter 7 *PROXY SERVER*)
- Enhanced Web Server, allowing access to customized web pages on the robot with dynamic content

  (Chapter 6 *WEB SERVER*)
- 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 *DOMAIN NAME SERVICE (DNS)*)
- 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 *DYNAMIC HOST CONFIGURATION PROTOCOL*)

- Simple Network Time Protocol (SNTP), allowing the robot SNTP client to get updated date/time information from a remote SNTP server

  (Chapter 13 *SIMPLE NETWORK TIME PROTOCOL (SNTP)*)

- Simple Mail Transfer Protocol (SMTP), allowing the robot SMTP client to send email to a remote SMTP server

  (Chapter 14 *SIMPLE MAIL TRANSFER PROTOCOL*)

- PC Share, allowing the robot to connect to and perform file operations on remote PC network Shares.

  (Chapter 17 *PC SHARE*)

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 *SOCKET MESSAGING*)

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:)

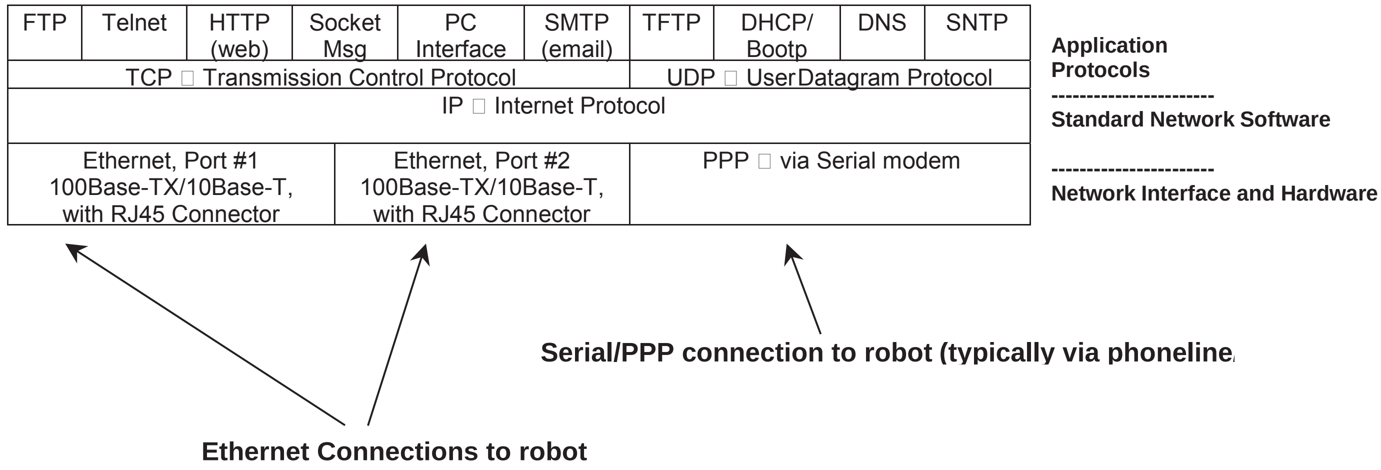
See the “Storage Devices” section under the “Program and File Manipulation” chapter in the application-specific *Setup and Operations Manual* for additional details on these devices.

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 *SETTING UP TCP/IP* 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.

Figure 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.

1.2.3 Image Backup and Restore

Image backup and restore operations provide the ability to back up and restore a copy of the entire contents of non-volatile controller memory in one operation.

The backup and restore operation uses a BOOTP and TFTP server in a host computer to backup and restore all of controller memory from the BMON Menu. Refer to Chapter 12 *ETHERNET-BASED LOADING* for more information.

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.

1.4 TCP/IP PROTOCOL

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

1.5 BOOTP AND TFTP PROTOCOLS

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

1.6 TELNET

The controller can support three Telnet connections. Telnet can be used to establish terminal sessions between a robot controller and a remote PC with Telnet software installed on it. This allows you to access your robot's teach pendant display remotely, use the KAREL Command Language (KCL), CRT/KB options, or a Diagnostic terminal depending on your system's configuration.

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.

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.

1.9 HOST COMMUNICATIONS

1.9.1 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

1.9.2 Architecture

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

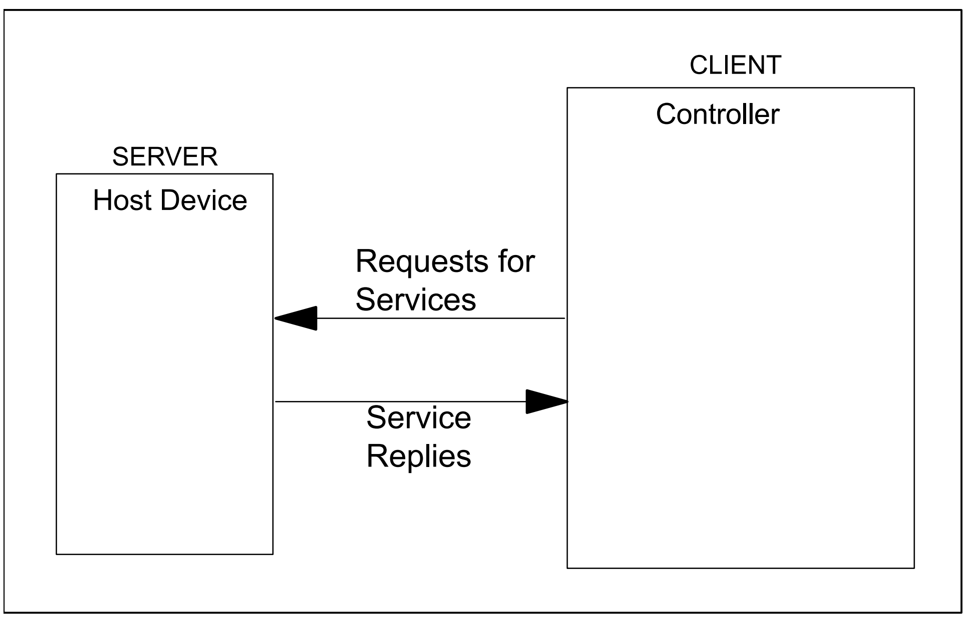
- The **client** is the device that needs a service.
- The **server** is the device that provides the service.

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 1–2 .

Note Client operation is available from the teach pendant and from KCL on a CRT/KB.

Figure 1–2. The Controller as a Client

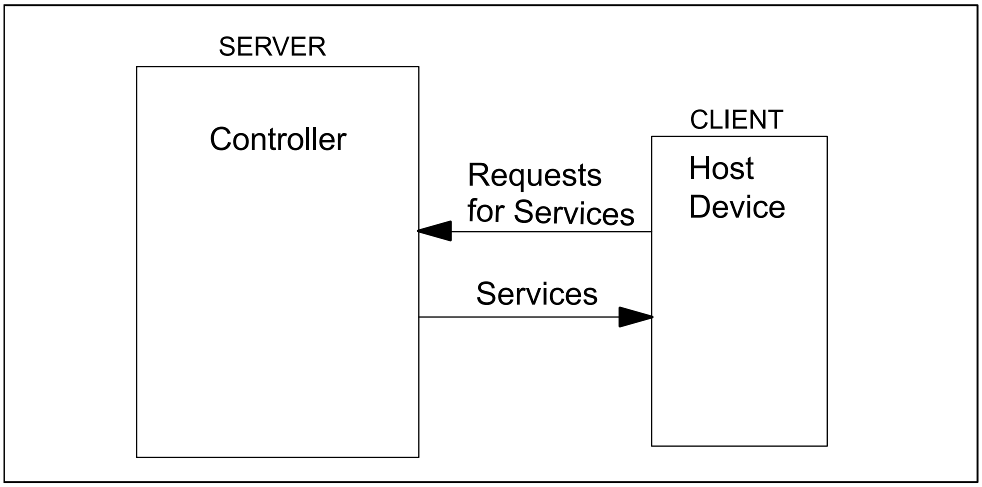


Servers

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 1–3 .

Figure 1–3. The Controller as a Server



1.9.3 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.

Chapter 2

SETTING UP TCP/IP

Contents

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, configure the TCP/IP parameters. Refer to Section 2.3 for information about configuring FTP and TCP/IP parameters.

2.2.2 Hardware Requirements

The R30iB supports two 10 Base-T or 100 Base-TX interfaces through the RJ45 Ethernet connectors (CD38A, and CD38B). 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. There is also a dedicated Ethernet port for vision, labeled CD38C. This port is strictly for vision applications. Please see vision documentation for setup and use.

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 $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.

Table 2–1. 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 charged. The baud rate and duplex node will be set to the fastest setting that both devices on the list can support.

2.3 SOFTWARE INSTALLATION OVERVIEW

This section contains an overview of FTP software installation. Refer to the *Software Installation Manual* for detailed procedures on software installation.

There are three 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 a Command File to enter the information for you. Refer to Appendix B .
- Use DHCP to configure network parameters automatically. Refer to Chapter 9 *DYNAMIC HOST CONFIGURATION PROTOCOL* .

2.4 DISPLAYING THE ETHERNET HARDWARE (MAC) ADDRESS

2.4.1 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.

2.4.2 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 CPU. Port numbering starts from the port close to face plate e.g. port # 1, port #2, etc.

2.4.3 Ethernet Hardware (MAC) Address Locations

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

- The physical labels on the Main board. Port numbering starts from the port close to face plate e.g. port # 1 MAC address, port #2 MAC address, etc.
- Using SHOW ETHERNET ADDRESS from the BMON Menu. Refer to Procedure 2-1 .
- The Board Address, which can be accessed from the TCP/IP Setup Screen. Refer to Section 2.5 .
- In the system variable $TMI_ETHERAD[x] where x is 1 for port 1, and 2 for port 2.

Note You cannot make changes to the Ethernet Hardware MAC address.

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

Steps

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

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

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

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

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

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

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

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

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

2.5 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 a Command File to enter the information for you. Refer to Appendix B .
- 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 2–3 lists and describes the TCP/IP Interface parameters you must define.

Table 2–2. 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.
SNIFF	This item allows you to configure and run the Ethernet Packet Sniffer.
SM	This item allows you to configure socket messaging parameters.

Table 2–2. SETUP Protocols Screen Items (Cont'd)

RIPE	This item allows robots doing a common job to share information efficiently.
PC SHARE	This item allows you to configure global PC Share parameters.
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.
FTP	This item allows you to configure FTP password parameters.
DNS	This item allows you to configure domain name system parameters.

Table 2–3. 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 the first two Ethernet ports. The router IP address must be on the same subnet as one of the first two 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 (LOCAL or SHARED)	This item specifies the Internet hostname. Entries for any hosts referred to by an FTP client tag, for example, are required, unless DNS is used. This item is case sensitive.
Internet Address	This item specifies the corresponding Internet address of each host.

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

Procedure 2-2 Defining TCP/IP Parameters

Conditions

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

Steps

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

```
SETUP Protocols
    Protocol         Description
 1  TCP/IP           TCP/IP Detailed Setup
 2  FTP              File Transfer Protocol
 3  NONE             Connects tag to port
```

5. Make sure TCP/IP is selected.

6. Press F3, DETAIL. You will see a screen similar to the following.

```
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 (top RJ45 labeled as CD38A) or interface #2 (bottom 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. Note that port #1 (CD38A) and port #2 (CD38B) must be configured on different networks. Port # 2 will not get initialized if it is configured as the same network as port #1 and warning message will be posted at powerup (HOST-179, IP Address mis-configuration).
- **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 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.

  Note The board address is displayed and cannot be changed. Refer to Section 2.4 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. 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. Press F3, LIST, to return to the SETUP Protocols screen.

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

2.6 FANUC SERVER ACCESS CONTROL (FSAC)

2.6.1 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.

2.6.2 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 for descriptions of available FSAC access levels.

Table 2–5. 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: • TP (teach pendant) • .PC (p-code) • .VR (variable)
2	Setup level	Program level, with additional access to download the following types of files: • .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>
```

2.6.3 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.

2.6.4 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 2–6 for a description of the system variables contained in SYSFSAC.SV.

Table 2–6. System Variables Contained in SYSFSAC.SV

Variable Name	Data Type	Description
$FSAC_ENABLE	Integer	FSAC Enable Flag. This can be set to either: • disabled (any value other than 1 will disable it) • 1, enabled
$FSAC_DEF_LV	Integer	FSAC Default Access Level. This can be set to: • 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 **Note** The name must be in the LOCAL/SHARED host table or DNS must be installed to resolve names.

Table 2–6. System Variables Contained in SYSFSAC.SV (Cont'd)

Variable Name	Data Type	Description
$FSAC_LIST[1-20].$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: • BIT 0: FTP • BIT 1: Telnet • BIT 2: HTTP (Web Server)

2.6.5 Example Configuration

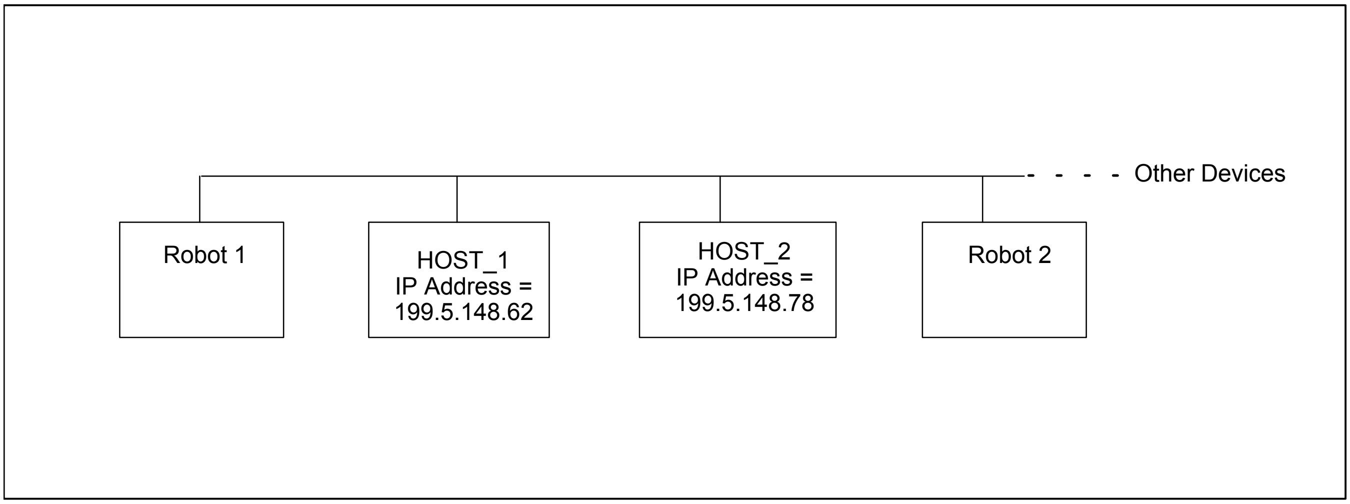
To enable FTP Server Access Control on Robot 1, and give full READ/WRITE access to HOST_1 and READ ONLY access to any other devices trying to use FTP to communicate with Robot 1, set the Robot 1 system variables as follows:

Example 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 2–1 .

Figure 2–1. Example Configuration



Chapter 3

FTP OPERATIONS

Contents

3.1 OVERVIEW

After you have installed and connected the appropriate Ethernet communications hardware and performed the appropriate device setup procedures, you can use FTP to communicate between the controller and other host devices.

This chapter contains information about the following FTP operations:

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

3.2 SETTING UP AND STARTING FTP

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

- Define TCP/IP parameters (Section 2.5)
- Install the FTP Interface software
- 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.

Refer to the *Software Installation Manual* for information on installing host communications options such as the FTP Interface software.

Table 3–1 lists and describes the items you must set up to define a client device. Table 3–2 lists and describes the items you must set up to define a server device.

Note Two FTP servers are configured and started automatically when the FTP option is installed. 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 2.5).

Table 3–1. 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 FTP, the protocol name is **FTP**. For PC Share, select **PC Share**. And so on.
***** This item is normally set up by the user. Other items can normally remain at their default values.	

Table 3–1. Client Device Definition Setup Items (Cont'd)

ITEM	DESCRIPTION
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: • UNDEFINED - the device is not defined. • DEFINED - the device is defined. • STARTED - the device is defined and started. The Startup State should normally be set to defined. When in the defined state the client tag is started automatically from KCL or from the FILE screen on the teach pendant whenever it is used.
Server IP/Hostname*	This item specifies the Hostname or IP address of the remote server to which the connection will be made.
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. • 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. 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: • 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. • 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 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 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 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 MENU.
2. Select SETUP.
3. Press F1, [TYPE].
4. Select Host Comm. You will see a screen similar to the following.

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

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

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

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

```
SETUP Tags
 Tag  S1:
  Comment:             ****************
+ Protocol Name:            FTP
  Current State:       UNDEFINED
+ Startup State:       START
  Server IP/Hostname:*****************
  Remote Path/Share: *****************
  Inactivity Timeout:  15 min
  Username:            *****************
  Password             *****************
```

+ These items are normally set up by the user. Other items can typically remain at their default values. A detailed description of the fields in the Setup Tags screen is given in Table 3–2 .

d. To enter a comment, move the cursor to Comment and use the function keys to type a message associated with this configuration and then press ENTER. You are not required to enter a comment.

e. Move the cursor to Protocol Name and press F4, [CHOICE]. A list of available protocol choices will be displayed.

f. Select FTP and press ENTER.

g. Move the cursor to Startup State and press F4, [CHOICE].

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

h. Select the startup state you want and press ENTER.

i. Move the cursor to Inactivity Timeout, type the timeout value you want, in minutes, and press ENTER. The default value is 15 minutes.

j. Press F3, LIST, to display the list of server devices.

k. Repeat Step 5c through Step 5j for as many server devices as you are defining.

6. To set up a client: (if required)

a. Press F4, [SHOW].

b. Select 2, Clients. You will see a screen similar to the following.

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

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

```
SETUP Tags
 Tag C1:
  Comment:            ****************
+ Protocol Name:           FTP
  Current State:      DEFINED
+ Startup State:      DEFINE
+ Server IP/Hostname:      192.168.1.49
+ Remote Path/Share:    robot/programs/
  Inactivity Timeout:   15 min
+ Username:                        Gary
+ Password                 ************
```

+ These items are normally set up by the user. Other items can remain at their default values in most cases. A detailed description of the fields in the Setup Tags screen is given in Table 3–1 .

d. Move the cursor to Comment and use the function keys to enter a message associated with this configuration. You are not required to enter a comment.

e. Move the cursor to Protocol Name and press F4, [CHOICE]. A list of available protocol choices will be displayed.

f. Select FTP and press ENTER

g. Move the cursor to Startup State and press F4, [CHOICE].

h. Select the startup state you want and press ENTER.

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

i. Move the cursor to the Server IP/Hostname field and enter the remote hostname or IP address. When a hostname is entered, this item is case sensitive and must be defined in the host name table (Procedure 2-2) unless DNS is used.

j. Move the cursor to Remote Path/Share field and use the function keys to enter the remote host path. This item is case sensitive and must end with a /.

k. Move the cursor to Inactivity Timeout, type the timeout value you want, in minutes, and press ENTER. The default value is 15 minutes.

l. Move the cursor to the Username field, and type in the username for the client to use to log into the remote FTP server.

m. Move the cursor to the Password field, and type in the password for the client to use to log into the remote FTP server.

n. Press F3, LIST, to display the list of client devices.

o. Repeat Step 6c through Step 18 for as many client devices as you are defining.

7. To define and start FTP on a device:

a. Press F4, [SHOW].

b. Select Clients or Servers.

c. Move the cursor to the client or server you want to define and start.

d. Press F2, [ACTION].

e. Select 1, Define.

f. Press F2, [ACTION], again.

g. Select Start.

h. Repeat Step 7c through Step 7g 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

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

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

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

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

6. Move the cursor to the username you want to change and press ENTER. Use the appropriate function keys to type the username and press ENTER. This item is case sensitive.
7. Move the cursor to the password that corresponds to that username and press ENTER. Use the appropriate function keys to type the username and press ENTER. This item is case sensitive.
8. Move the cursor to the corresponding timer and press ENTER. Use the numeric keys to type the time and press ENTER.
9. Repeat Step 6 through Step 8 for the remaining usernames and passwords you are setting.
10. Press F3, LIST, to return to the SETUP Protocols screen.

3.4 ACCESSING AND USING CLIENT DEVICES

3.4.1 Access Description

A client device does not have to be started before it is accessed. However, the *tag* must be defined. The device automatically will be started when opened and stopped when closed, returning it to the defined state.

FTP copies files of type .CF, .KL, and .LS, as ASCII files. All other file types are transferred as binary files.

3.4.2 File Specification for Client Devices

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

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

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

- **device_name** is a two- to five-character optional device name field, followed by a colon. The first character must be a letter; the remaining characters must be alphanumeric. The default device from the system console variable $DEVICE will be used if this field is absent (C1:, for example).
- **host_name** is a file name type consisting of one to eight characters. The optional **host_name** field selects the network node to receive this command. It must be preceded by two backslashes and separated from the remaining fields with a backslash. If a host_name is not present, the string specified for the Remote (Current) will be used as the default host_name. host_name must already have been defined in the host table (Procedure 2-2).
- **path_name** is a recursively defined optional field consisting of one or more **file_names** separated by a backslash. It is used to select the file subdirectory. It can consist of up to a maximum of 64 characters. If a path_name is not present, the string specified for the Path (Current) will be used as the default path_name.

  The root or source directory is handled as a special case. For example, access to the subdirectory **SYS** linked off of the root would have a **path_name** of **’\SYS’** . The **file_spec** using this **path_name** would be **C1:\\HOST\’\SYS’\FILE.KL** .

- **file_name** is from one to eight characters. Note that **file_name** is sent over the network in lower case format, regardless of how it is entered. Therefore, upper case file names on a case-sensitive remote host cannot be retrieved.
- **file _type** is from zero to three characters.

Note Generally the host_name and path_name fields can be omitted based on the client tag setup. A typical example of using a client tag from KCL would be:

```
KCL> COPY C1:myprog.tp to MD:
```

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

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

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

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

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

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

   The client device will now start automatically when the controller is turned on.

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

3.4.4 Teach Pendant File Access

After a client device has been defined, it can be used from the teach pendant. Refer to the "Program and File Manipulation" chapter in the appropriate application-specific *Setup and Operations Manual* for information on program and file manipulation.

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:

3.4.5 KCL File Access

After a client device has been defined, it can be used from KCL at the CRT/KB to copy files between the controller and any node device on the network. This node device can be either the host computer or another controller with server tags started.

The general format for copying files is the KCL COPY command with the appropriate host communications definitions for the source and destination file specifications.

Transfer File FROM Controller TO Node Device

To transfer a copy of the source file (**src_file_spec**) to the destination device and destination file (**dst_file_spec**), type the following:

```
KCL> COPY <src_file_spec> TO <dst_file_spec>
```

Transfer File FROM Node Device TO Controller

To transfer a file from a node device, named **FRED** , to the controller, type the following, for example:

```
KCL> COPY C1:\\FRED\TPDEF.VR TO RD:\TPDEF.VR
```

Generate Directory On Node Device

To generate a directory of files on a node device, type the following:

```
KCL> DIR <src_file_spec>
```

Delete File from Node Device

To delete a file on a node device, type the following:

```
KCL> DELETE FILE <src_file_spec>
```

Save Program to Node Device

To save a program to a node device, type the following, for example:

```
KCL> CD C1:
KCL> SAVE TP TEST1
```

Load Program from Node Device

To load a program from a node device, type the following, for example:

```
KCL> CD C1:
KCL> LOAD TP TEST1
```

Note If you use Distinct FTP on your host PC and you want to send a DIR command to a Distinct FTP Server, the file specification must be delimited by single quotation marks. For example

```
DIR '*.*'
```

```
DIR '*.tp'
```

3.5 ACCESSING SERVER DEVICES

3.5.1 Overview

This section contains information about accessing server devices. A server device listens for connections that are initiated from the host computer. One server can support one connection. Therefore, you control the number of devices that are connected to the controller by starting only the appropriate number of server tags. **When no server tags have been started, no connections can be received.**

You cannot select which server devices are used for specific connections. This is determined by the TCP/IP Host Communication software.

3.5.2 Access Description

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

3.5.3 Starting and Stopping a Server Device

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

Procedure 3-4 Starting and Stopping a Server Device

Conditions

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

Steps

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

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

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

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

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

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

   The server device will now start automatically when the controller is turned on.

3.5.4 Blocking Downloads of Certain File Groups

FTP supports preventing certain file groups from being downloaded to the robot from a remote host, using the FTP server on the robot. For example, using this FTP feature, all TP programs can be prevented from being downloaded to the MD device.

This feature is disabled by default and needs to be enabled before use. To enable this feature, set $FTP_CTRL.$DNLD_FILTER = TRUE and turn the controller off and back on.

If the feature is enabled, any file that is in a special table used by FTP will be blocked from being downloaded via the robot FTP server. An FTP error, such as “501 Permission Denied,” will be posted.

The table of files that can be blocked is made up of:

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

3.5.4.1 Features

- For every download request, the FTP server matches the filename with the internal table of files that are to be blocked.

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

3.5.4.2 Examples

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.

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

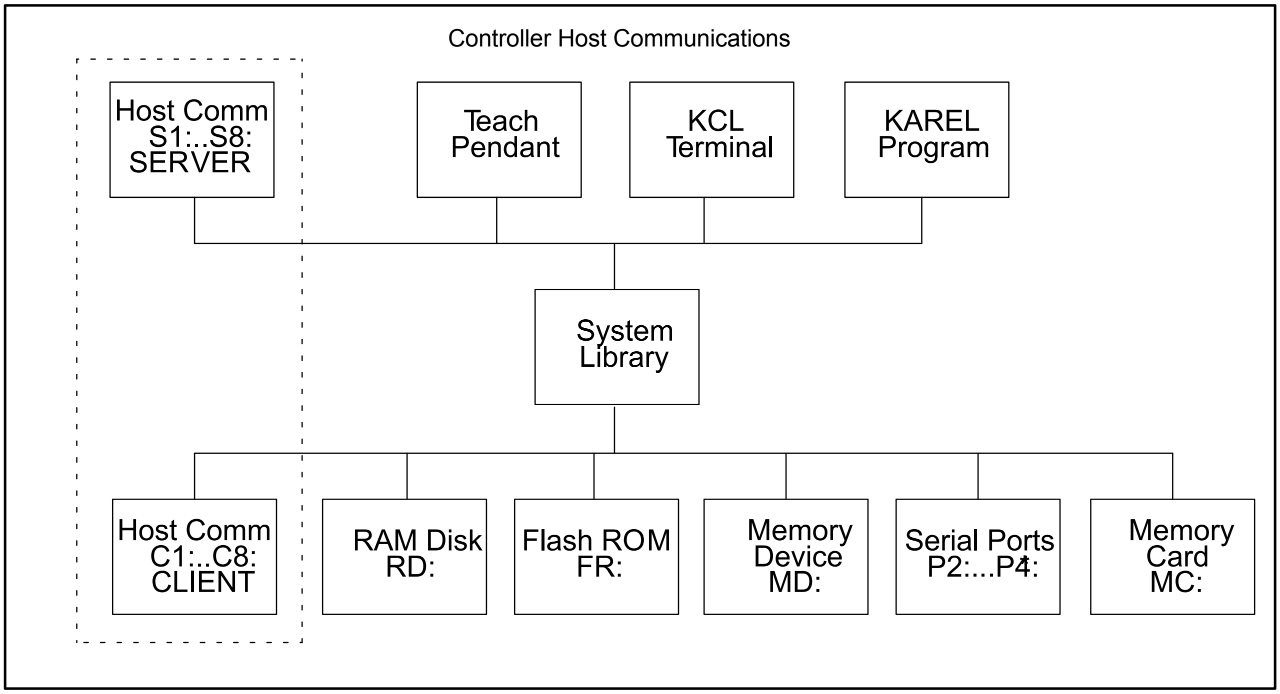
The following FTP services are provided:

- Environment services
- File transfer services
- Directory services

These services can be performed only by server devices.

Figure 3–1 shows the relationship of host communications to the controller system. It also shows the devices and the services that can be accessed.

Figure 3–1. Host Communications Model



* Can be used for floppy (FLPY:), debug console, KCL, and so forth. Refer to the appropriate application-specific Setup and Operations manual for more information.

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 or KCL, 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

The dir service provides the same directory operations as the KCL DIRECTORY command. Refer to the *KAREL Reference Manual* for more information about KCL commands.

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 3–4 .

Table 3–4. 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:.

Refer to the appropriate application-specific *Setup and Operations Manual* for information on specific file operations.

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, or KCL. For example, you could use the MDB: device to copy all teach pendant files (including invisible files) to the memory card (KCL>copy MDB:*.tp TO mc:).

The MDB: device directory function includes only those files that should normally be backed up. When using FTP, a request to the MDB: device such as "mget *.*" (in binary mode) would provide a complete backup of the robot system and application files based on MDB: being configured correctly.

Backing up the binary portion of the memory device (MDB:*.*) provides a complete application backup (analogous to Backup — All of the above on the teach pendant file menu). It does not include the ASCII versions of programs/variables so it is smaller in size and faster to back up. This backup is appropriate for disaster recovery of the application. Note that the controller must be at CTRL start to restore most system files.

Filtered Memory Device (FMD:)

The Filtered Memory Device option generates text versions of all backup files of user programs and variables that have been changed manually. Included are system and KAREL variables, position and data registers, teach pendant programs, and I/O configuration data.

You can compare these files with previous versions to determine what users or operators have changed. Variables and programs that change without user input are filtered out, and will appear in filter exclusion files.

After the option is installed, it will run automatically whenever you perform an Ethernet backup of the controller from the FMD: device. After you install the Filtered Memory Device option, any of the following filter exclusion files could appear on the FR: device.

Caution

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

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

You can view program, variable, or filter exclusion files via KCL. For example:

```
KCL> DIR FMD:*.*.
```

Note Computer systems that perform periodic backups could be modified to use the FMD: device instead of the MD: device for some compare operations, for example. Contact FANUC America Corporation for more information.

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 3–5 by specifying the Memory Device and the reserved file names within the file access services that are supported for Memory Device. Refer to the appropriate application-specific *Setup and Operations Manual* for more information.

Table 3–5. System Files Accessed through the Memory Device

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

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

When transferring system files, you must ensure that the system files have the same core version as the system to which you are transferring these files.

Refer to the appropriate application-specific *Setup and Operations Manual* for more information.

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 3–6 for a listing of error log files.

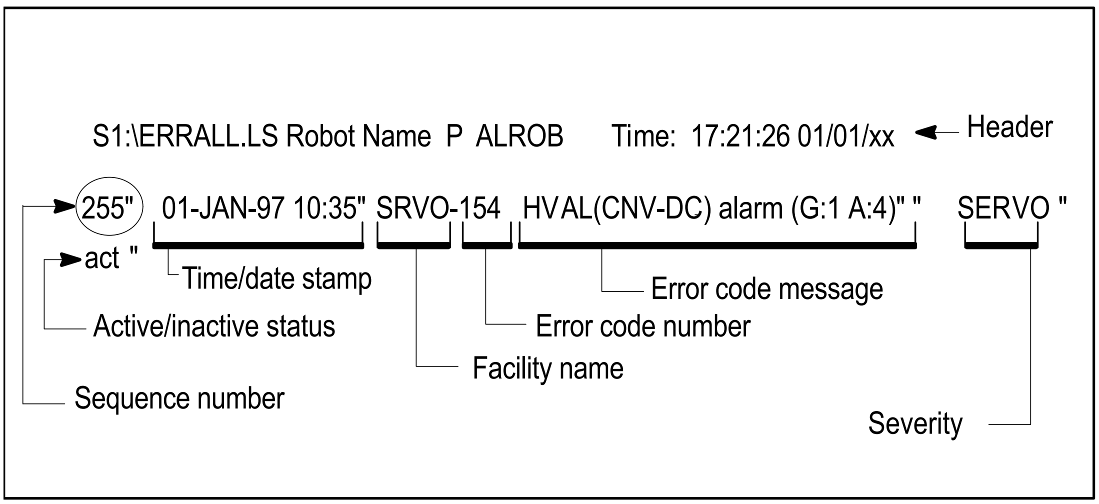
Table 3–6. Error Log Files

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 3–2 for an example of an error log entry.

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

  To save CMOS space, the last field (IP address) is recorded only if the log is stored in DRAM.

Figure 3–3. IP Address

```
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
```

Most users can leave the default configuration which sets the log size to store 50 entries in DRAM. If you want to change the default settings, please refer to $FTP_CTRL system variable in the *Software Reference Manual* for details on configuring the transfer log feature.

Chapter 4

DOMAIN NAME SERVICE (DNS)

Contents

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.

To use DNS with your robot, you must have a properly configured network (Refer to Section 2.5). You must also configure the robot with a DNS server (check with your network administrator for the IP address of your DNS server—see Section 4.2 for more details). Possible uses for DNS include FTP (refer to Chapter 3 *FTP OPERATIONS*), external HTTP proxy (refer to Chapter 7 *PROXY SERVER*), and Vision among others.

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 4–1 lists and describes the parameters you must define.

Table 4–1. 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.

Table 4–1. DNS Parameters (Cont'd)

PARAMETERS	DESCRIPTION
Secondary DNS Server	This item specifies the IP address of the secondary DNS server for your network. This server will be contacted if your primary server is unreachable or not responding. It is not required in order for DNS to work. Not all networks have secondary DNS servers, so you should check with your network administrator to see if your network has one.
Local Domain Name	This item is the domain name for your local network. Examples of local domain names are frc.com or aarnet.edu.au. Your network administrator can provide you with the correct local domain name for your network. While in most instances you will only need to configure a single local domain name, there exist rare instances where multiple local domain names are required. Up to five names can be configured, separated by a semicolon ';'. **Note** DNS will not work if you do not provide a local domain name.
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 client on the controller and DNS server on the remote servers. Refer to Section 2.3 .
- You have configured the FTP client on your robot controller and install FTP server on the remote servers. Refer to Section 2.3 .

Steps

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

```
SETUP Protocols
    Protocol                  Description
  1  TCP/IP                   TCP/IP Detailed Setup
  2  FTP                      File Transfer Protocol
  3  DNS                      Domain Name Service
```

5. Move the cursor to TCP/IP and configure, if necessary. Refer to Section 2.3 if you have not configured TCP/IP. Otherwise, go to Step 6 .
6. Move the cursor to DNS and press F3, DETAIL. You will see a screen similar to the following.

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

7. Move the cursor to each item and specify the appropriate information:
   - **Primary DNS Server -** This specifies the unique address of the primary DNS server. Contact your network administrator for the address of your network's primary DNS server.
   - **Secondary DNS Server -** This specifies the unique address of the secondary DNS server for your network. Your network may or may not have a secondary DNS server. Contact your network administrator for the address of your network's secondary DNS server.
   - **Local Domain Name -** 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
*****************************************
```

Chapter 5

TELNET

Contents

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. Telnet is included in FTP software which is part of standard controller software. So it is always loaded with standard controller software. 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 KCL or CRT/KB (if these options are loaded) and teach pendant terminals over the remote Telnet connection. The Telnet screens are under the SETUP Hostcomm menus.

Note 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 5–1 .

Table 5–1. 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 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. 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 allows you to log into the KCL device. Make sure that the KCL option is loaded first. If the CRT/KB option is loaded and the port is configured for KCL/CRT, logging in via Telnet as KCL will bring up a KCL/CRT display.
help or ?	This item displays a help screen related to the topic you have selected.

Table 5–3. 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: • 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 Telnet option loaded as a part of standard software on your robot controller.
- You have configured the Ethernet hardware and software on your robot. Refer to Procedure 2-2 .

Steps

1. Press MENU.
2. Select SETUP.

3. Press F1, [TYPE].

4. Select Host Comm. You will see a screen similar to the following.

```
SETUP Protocols
     Protocol         Description
  1  TCP/IP           TCP/IP Detailed Setup
  2  TELNET           Telnet Protocol
  3  FTP              File Transfer Protocol
  4  DNS              Domain Name Service
  5  NONE             Connects tag to port
```

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

```
SETUP Telnet
Username      Access   Password    Timer
TP           NONE     ********      0
KCL          NONE     ********      0
CONS         NONE     ********      0
```

Note You must make sure that the password for the KCL device needs to be explicitly set before use.

6. You can set up passwords and access levels only if you do not want to use the defaults. The timer field is disabled by default (0). If a positive value is set, it determines the number of minutes of inactivity on the connection before the connection is terminated.

With the SETUP Telnet screen displayed, press F5, HELP. You will see a screen similar to the following.

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

5.2.2 Connecting to a Telnet Server

After you have set up the Telnet feature, you can use it to connect to a Telnet server. Use Procedure 5-2 to connect to a Telnet server.

More security measures, in addition to passwords, are available to control remote access into the robot. Telnet supports the FANUC Server Access Control (FSAC) feature, which decides which remote hosts (PCs) are allowed to connect into the robot. Refer to Section 2.6 for more information on setting up FSAC for Telnet.

Procedure 5-2 Connecting to a Telnet Server

Steps

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

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

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

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

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

3. From your PC or UNIX workstation, type a valid login name for the device to which you want to connect and press ENTER. Refer to Table 5–1 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**.

Chapter 6

WEB SERVER

Contents

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.

There is access to KCL commands and the ability to run KAREL programs (user programming required). There is also support for *Server Side Include* (SSI) directives. These features (KCL, KAREL, SSI) are part of the Web Server Enhancements Option.

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 6–1 for the web server system variables and their descriptions.

Table 6–1. Web Server System Variables

SYSTEM VARIABLE	DESCRIPTION
$HTTP_CTRL.$ENABLE	This variable automatically starts the web server when the controller is turned on if the value is greater than 0 (the default value is 1). Reset this variable to zero if you would like to disable the web server when you turn the controller on again.
$HTTP_CTRL.$KRL_TIMOUT	This variable defines the maximum number of seconds to wait for a KAREL program to complete which is requested through the web server. Refer to section on “Running KAREL Programs from the web browser. The default value is 10 seconds.
$HTTP_CTRL.$HITCOUNT	This variable is incremented each time the web server gets a request. This variable can be modified at any time if, for example if you want to reset the hitcount to 0. This is an integer variable that will roll over at the maximum value (2147483646).

Table 6–1. Web Server System Variables (Cont'd)

SYSTEM VARIABLE	DESCRIPTION
$HTTP_CTRL.$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).
$HTTP_CTRL.$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/KCL/Server Side Include feature within web server, based on the configuration of FSAC. An access level of Operator level or above is required to access other files from the web server.

Access to the *i* Pendant screens is also controlled by FSAC. If $UI_CONFIG.$READONLY[2]=TRUE, then all levels have read-only access. If $UI_CONFIG.$READONLY[2]=FALSE, then the Operator and User-defined levels have read-only access, the Program level will have access to screens used for programming the robot, the Setup level will have access to screens used to set up the system, and the Install level will have read-write access to all the screens.

Refer to Section 2.6 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 contains the current robot status, a listing of important diagnostic files and links, and robot tools.

The following example URLs (either Figure 6–1 or Figure 6–2) requests the robot default home page shown in Figure 6–3 .

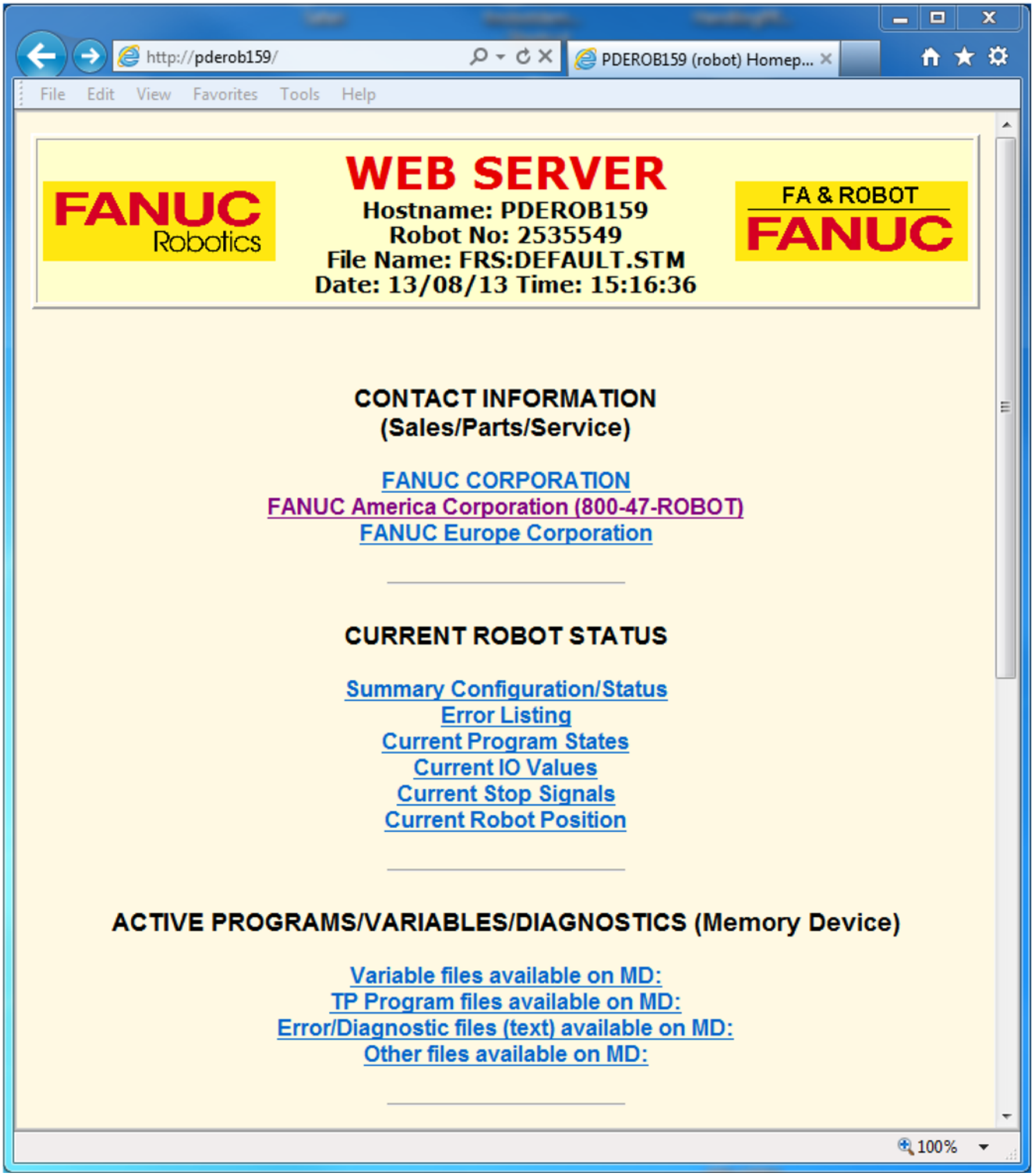
Figure 6–1. URL Example

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

Figure 6–2. URL Example

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

Figure 6–3. Default Robot Home Page



The links on the default home page called "Active Programs /Variables /Diagnostics (Memory Device)" is the memory device file list. This list is built dynamically each time the page is requested based on the programs and variables loaded in working memory. The links are defined in Table 6–2 .

Table 6–2. Program/Diagnostic Link Descriptions

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

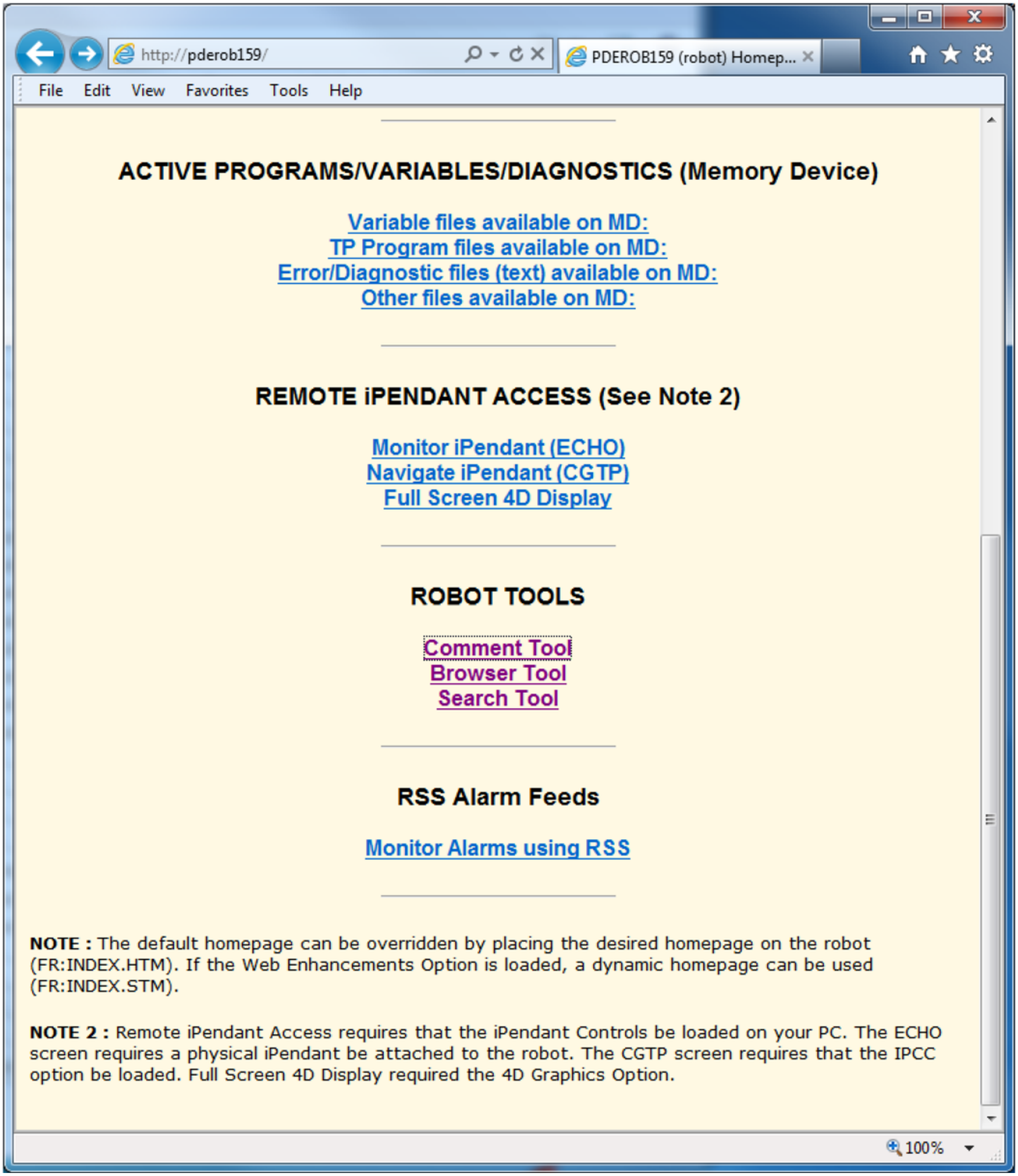
An example of Variable Files on MD: is shown in Figure 6–4 .

Figure 6–4. Variable Files on Memory Device

The links on the default home page called "ROBOT TOOLS" allow you to enter text information easily using a keyboard.

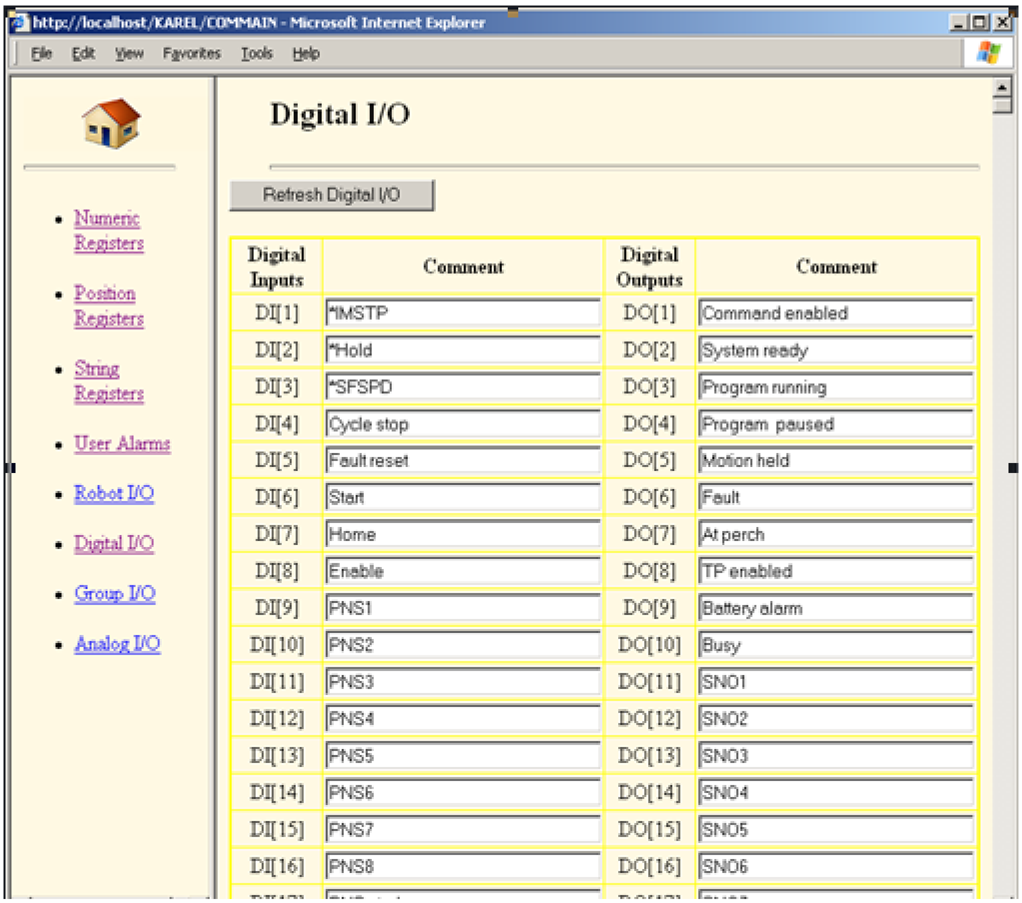
Note The Web Server Enhancements Option must be loaded to use this feature. See the section on HTTP Authentication to set up access to KAREL through the web server.

Figure 6–5. Robot Tools



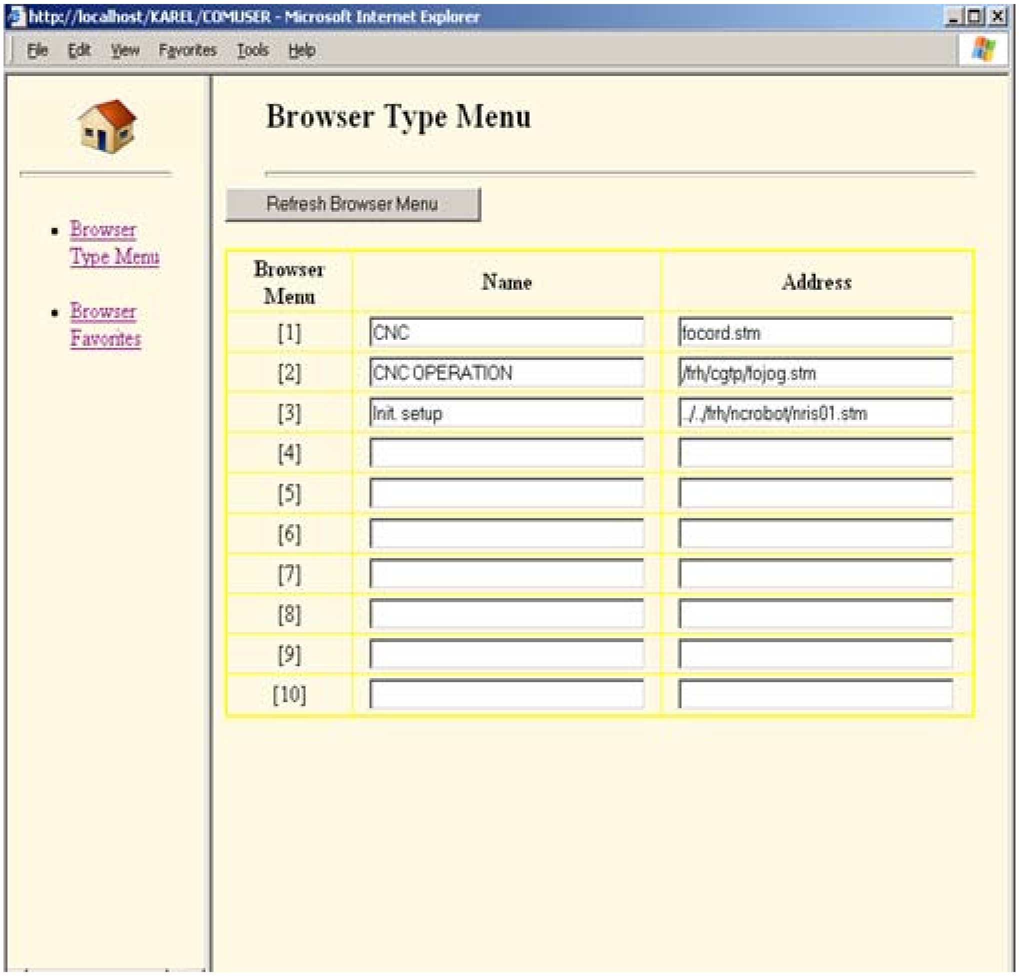
Comment Tool will split the screen as shown in Figure 6–6 . The left side links determine the type of robot information displayed on the right side. The data is retrieved from the robot controller but it is not dynamically updated so you must press the "Refresh" button if you want to refresh the contents of the page. When you are done entering text in the box, click the mouse outside the box or press the tab key. The new data is immediately entered to the robot. When you wish to return home, press the HOME icon on the upper left of the screen.

Figure 6–6. Comment Tool



Browser Tool will split the screen as shown in Figure 6–7 . Browser Type Menu contains the menu entries within the BROWSER menu. Browser Favorites allows you to add links to the FAVORITES page.

Figure 6–7. Browser Tool



Search Tool will split the screen as shown in Figure 6–8 .

Search Programs will search for text in all programs. It generates a new version of MD:*.LS files and searches within those files. Links to the .LS file will be displayed for all matches.

Search All Registers will search for text in all numeric, position, and string registers, including the comments. It generates a new version of MD:*REG*.VA and searches within those files. Links to the .VA files will be displayed for all matches.

Search System Variables will search for text in all system variables, including the non-saved variables. It generates a new version of MD:SY*.VA and searches within those files. Links to the .VA files will be displayed for all matches.

Search KAREL Variables will search for text in all KAREL variables. It generates a new version of MD:*.VA, excluding registers and system variables, and searches within those files. Links to the .VA files will be displayed for all matches.

The search is not case sensitive, but you can match the case if desired. The search for match case is faster.

Figure 6–8. Search Tool

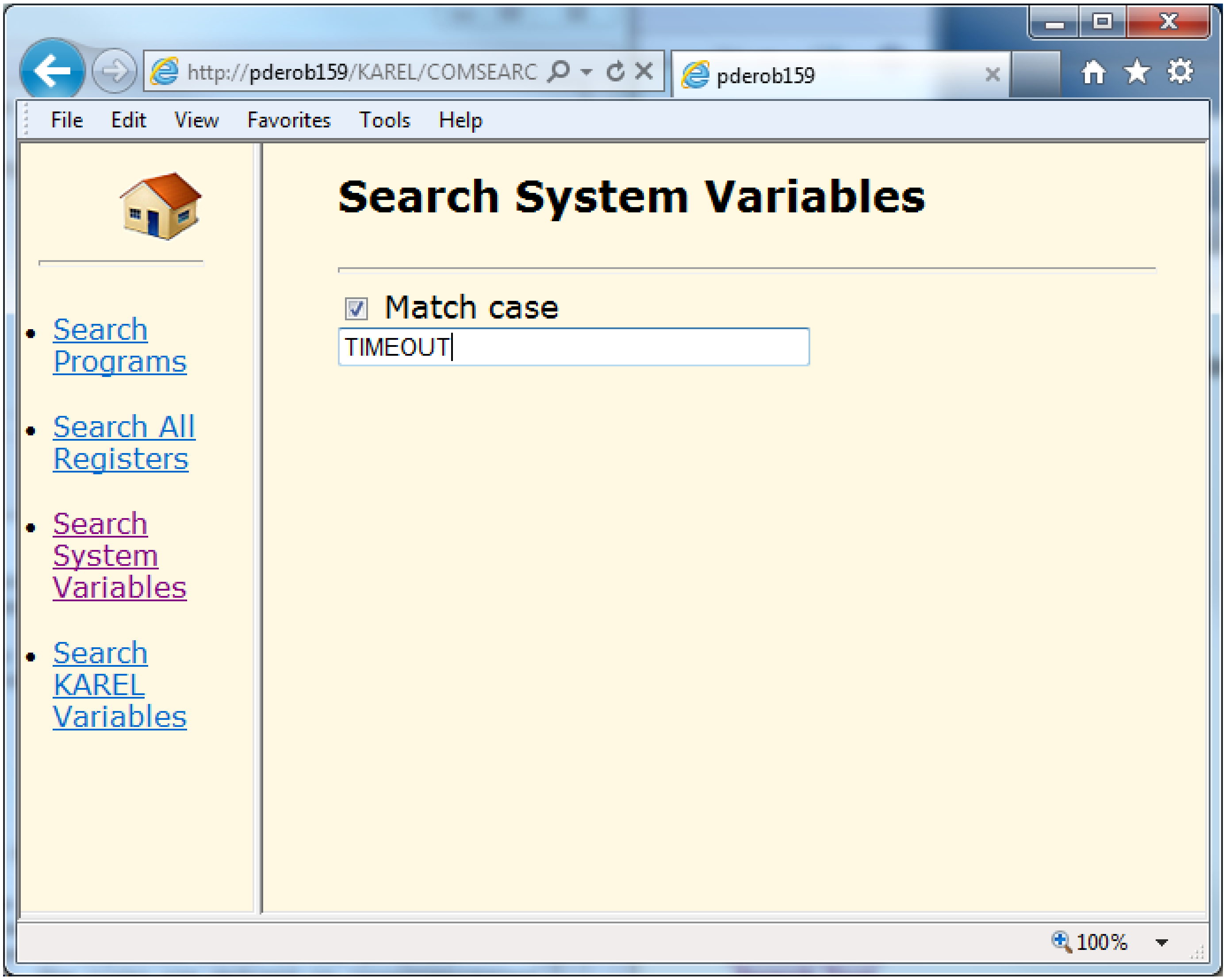
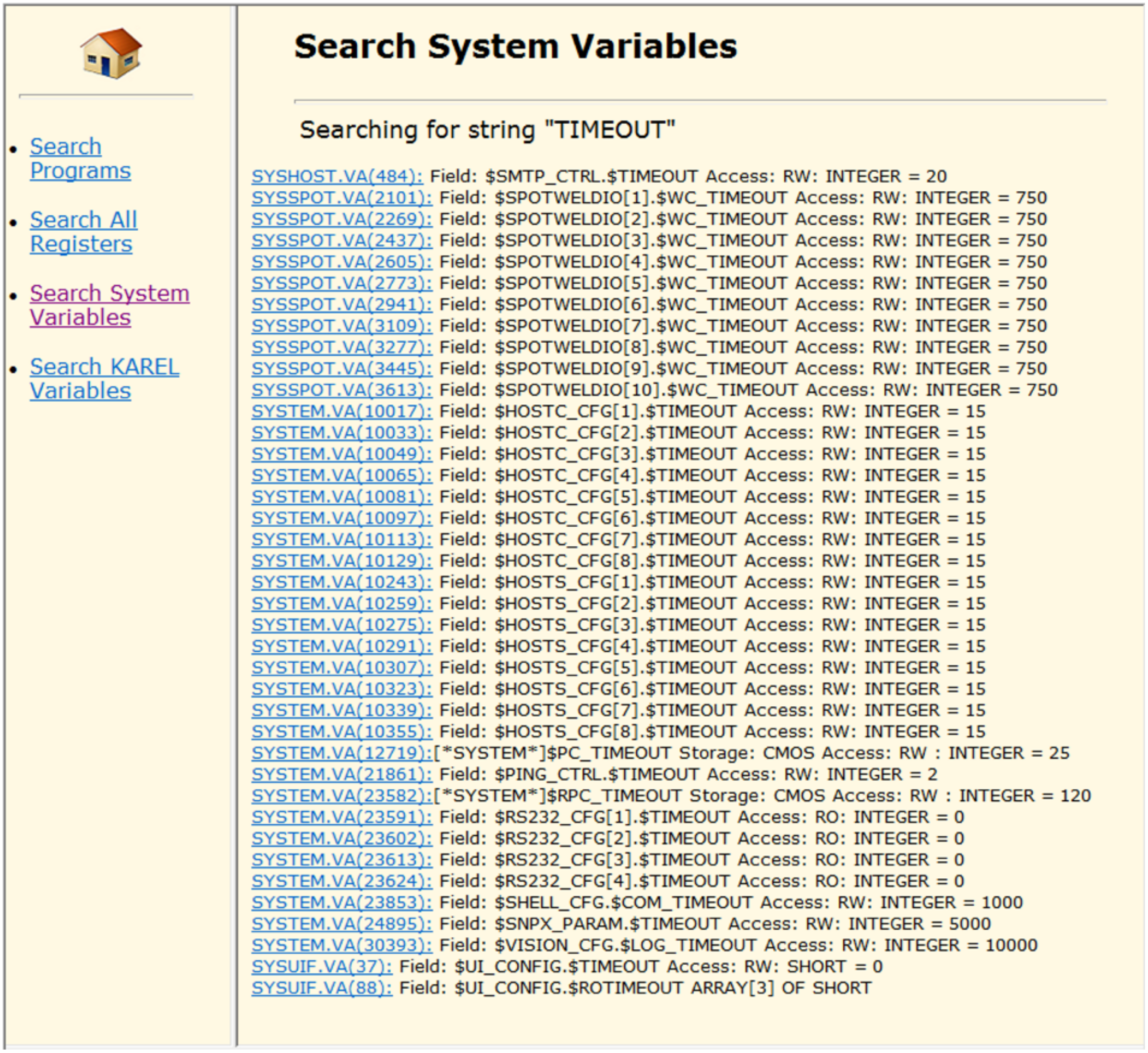


Figure 6–9 is an example of search for TIMEOUT in System Variables.

Figure 6–9. Search Tool Results



6.3.3 Customizing Your Robot Home Page

A customized home page can be loaded to replace the default home page. The file FR:INDEX.HTM will be shown (if it exists on your robot controller) in place of the default home page.

The web server currently is able to return the following kinds of files:

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

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

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

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

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

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

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

General URL Syntax

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

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 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:

Simple HTML Header

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

A very simple HTML trailer might be:

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 :

Header

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

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.

$HTTP_CTRL

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

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

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

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

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

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 Using KCL with Web Server

KCL commands can be run from a web browser. The KCL commands are actually processed through an internal filter on the robot to allow tuning of allowed commands. The following commands are disabled by default:

```
cmds[1] = 'MOVE'
cmds[2] = 'RECORD'
cmds[3] = 'ABORT'
cmds[4] = 'CLEAR'
cmds[5] = 'CONTINUE'
cmds[6] = 'DISMOUNT'
cmds[7] = 'EDIT'
cmds[8] = 'FORMAT'
cmds[9] = 'HOLD'
cmds[10] = 'PAUSE'
cmds[11] = 'PURGE'
cmds[12] = 'RUN'
cmds[13] = 'RUNCF'
cmds[14] = 'SKIP'
cmds[15] = 'STEP'
```

To disable a specific KCL command, include it in [HTTPKCL]cmds[]. To allow a specific KCL command, remove it from this variable. The above list is based on allowing commands to gather data (for example, SHOW is allowed), but not allowing any command that could modify data or cause motion.

Note The Web Server Enhancements Option must be loaded to use this feature See the section on HTTP Authentication to set up access to KCL through the web server. .

To access KCL from a browser use KCL as the device specifier and include the KCL command itself instead of the filename. For example:

Accessing KCL from a Browser

```
http://robot1/KCL/show var $Version
```

Note Some browsers might generate an error if you use spaces within the URL. In this case, do one of the following:

- Use a different browser. At present Internet Explorer tolerates spaces, but Netscape does not.
- Embed the command within a form.
- Use "%20" in the URL for spaces and "%24" for the "$". For example:

Specifying Spaces and “$” in a URL

```
http://robot1/KCL/show%20var%20%24Version
```

Note To send KCL commands without getting a response in your web page use **KCLDO** instead of KCL. **KCLDO** is the same as KCL except that there is no content returned as long as the command is successful. This is useful for buttons on web pages which are intended to cause an action without requiring the web page to be rewritten.

6.3.6 Running KAREL Programs from the Web Browser

Note For R-30*i*B and R-30*i*B Mate controllers, the KAREL option must be installed on the robot controller in order to load KAREL programs.

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.

The Web Server Enhancements Option is required to support this feature. Refer to Section 6.5 to setup access to KAREL programs through the web server.

6.3.7 Creating Web Pages Based on KAREL Programs

Note The KAREL option must be installed on the robot controller in order to load 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. Demo.kl – Example File Access Program through 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.

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




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)




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). For example :

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). For example:

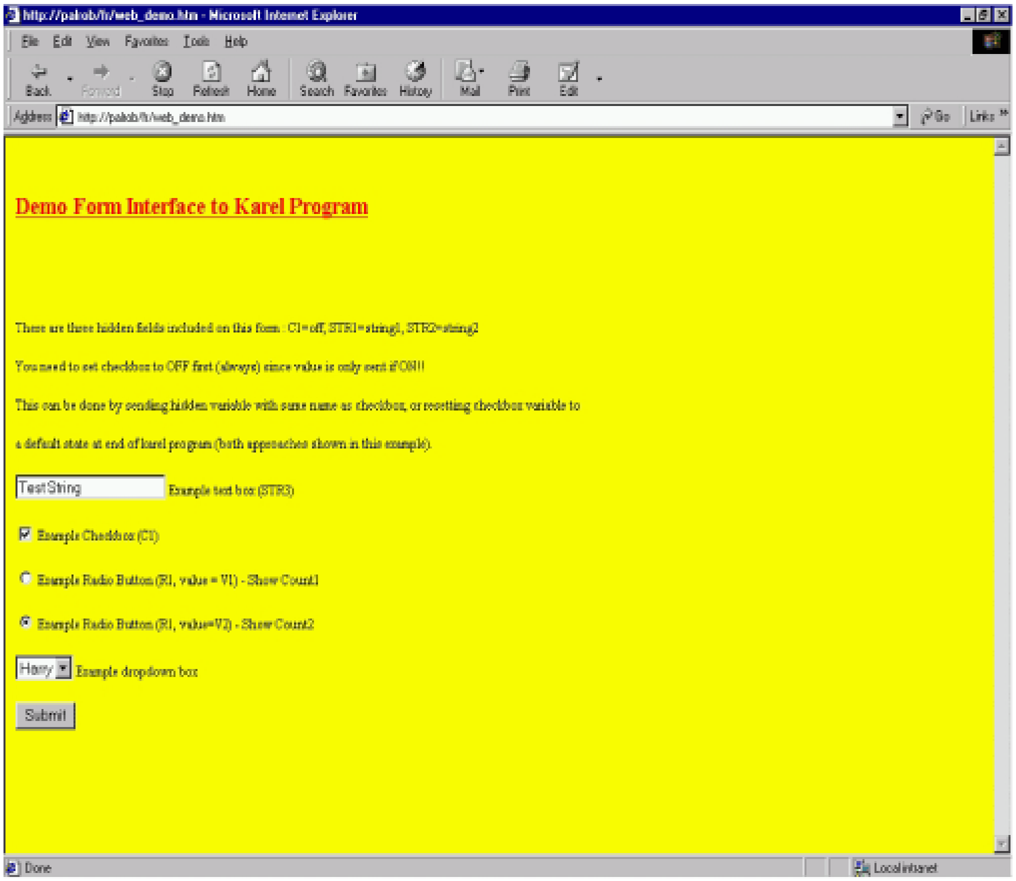
Variable Declaration for Using a Web Page to Pass Parameters to a KAREL Program

```
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 6–10 through Figure 6–11 .

Figure 6–10. Example KAREL Based Web Page Using Parameters

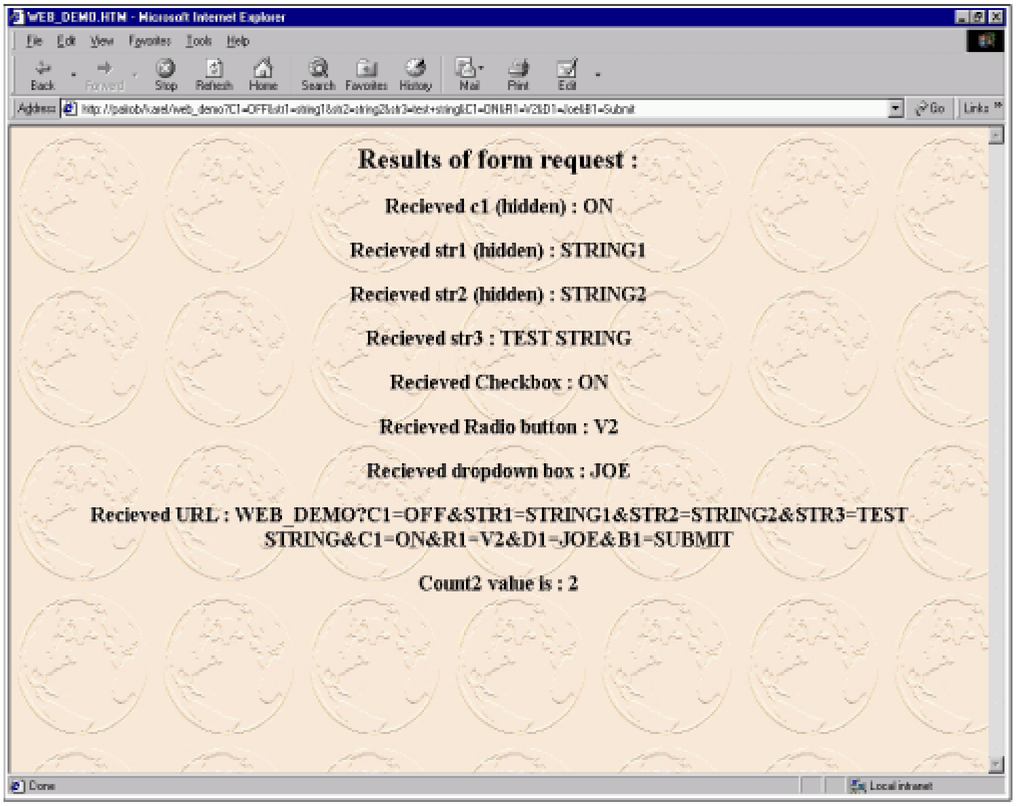


Demo Form Interface to a KAREL Program

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">
<
html>
<
head>
<
meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1">
<
meta name="GENERATOR" content="Microsoft FrontPage 2.0">
<
title>Web Demo</title>
<
/head>
<
body bgcolor="#FFFF00">
<
```

```
p>&nbspj </p>
<
p><font color="#FF0000" size="5"><strong><u>Demo Form Interface
to Karel Program</u></strong></font></p>
<
p>  </p>
<
form action="http://palrob/karel/web_demo" method="GET"
name="uif_demo">
    <input type="hidden" name="C1" value="OFF"><input
    type="hidden" name="str1" value="string1"><input
    type="hidden" name="str2" value="string2"><p>&nbspj </p>
    <p>There are three hidden fields included on this form :
    C1=off, STR1=string1, 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>
```

Figure 6–11. Results of the Demo Form Interface to a KAREL Program



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]



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)




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

6.3.8 Running KAREL Programs to Perform Commands

Note For R-30*i*B and R-30*i*B Mate controllers, the KAREL option must be installed on the robot controller in order to load KAREL programs.

KAREL programs that do not include any motion can be run from the web browser to perform commands. In this case, the web page does not change and no response is returned from the KAREL program.

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 must include the %NOLOCKGROUP directive.
- The KAREL program must specify the no content return code as follows:

```
return_code = 204  -- no content return code
```

- The KAREL program can be passed string arguments as described below.
- 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 Web Server Enhancements Option is required to support this feature. Refer to Section 6.5 to setup access to KAREL programs through the web server.

Here are some examples of its use:

- Inside HTML:

```
<a href="/KARELCMD/progname?name1=value1&name2=value2">KAREL Command</a>
```

- Inside javascript

```
<script language="javaScript">
<!--
function KARELCommand() {
  window.location.href ='/KARELCMD/progname?name1=value1&name2=value2';
}
//-->
</script>
```

- iPendant ButtonChange Control:

```
<param name="PageName" value="/KARELCMD/progname?name1=value1&amp;name2=value2">
```

The URL consists of the following:

- The KAREL program is specified after /KARELCMD/.
- You can use /KAREL/ or /KARELCMD/. The Browser menu has BACK and FORWARD function keys. It keeps a history of URLs and can replay them. If you use /KARELCMD/, then the Browser will not keep a history of the URL so it cannot be accidently replayed by the BACK and FORWARD keys. In fact, the only difference between using /KAREL/ and /KARELCMD/ is that /KAREL/ will store the URL in the history and /KARELCMD/ will not.
- The name/value pairs are specified after the ? and are separated by &
- Each name/value pair is parsed and the corresponding STRING variable is set in the KAREL program if it exists. The web server does not support truncation so your STRING variable must be big enough to hold the value, otherwise the name/value pair is ignored.
- After the variables are set, the web server will run the KAREL program. In the example below, name1='value1' and name2='value2'. You can name your parameters whatever you wish.
- An example KAREL program follows:

```
PROGRAM progname
%NOLOCKGROUP
%NOABORT=ERROR+COMMAND
%NOPAUSE=ERROR+COMMAND+TPENABLE
%NOBUSYLAMP
VAR
name1: STRING[20]
name2: STRING[20]
return_code: INTEGER
BEGIN
IF UNINIT(name1) THEN name1 = ''; ENDIF
IF UNINIT(name2) THEN name2 = ''; ENDIF
-- Perform command
return_code = 204  -- no content return code
END progname
```

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

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

6.4.2 Syntax

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

where "command" can be one of the following:

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

Each line (up to 200 characters long) is processed separately for a .STM file. If the HTML comment delimiters are found and the first character within the comment is a "#" then the comment is interpreted as an SSI directive and an attempt is made to process it as such.

SSI directives cannot be split between lines. The entire command must be placed on a single line. Multiple commands can be used within a single line.

The result of the SSI directive is placed in the response sent to the client browser in place of the SSI directive. There are certain characters that have special meaning within a SSI directive:

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

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

example.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:

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. For example:

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:

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:

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:

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.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:

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.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:

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 6–3. 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 certain KCL commands and 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. Refer to Section 6.3.5 and. Section 6.3.6 for details on how to run KCL and KAREL commands through the web server.

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:

- <pre><!–#exec cmd ="kcl/show%20mem" –></pre>
  - This will place the results of a "KCL>show memory" command in the response to the browser.
  - 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 ="kcl/show%20var%20%24version" –>
- <!–#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 KCL and 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 _1var1 and the value to 12:

Set the Name of a Local Variable

```
<!--#set var="1var1" value="12" -->
```

- A subsequent command to set _1var1 will update the value:

Update the Value of the Local Variable

```
<!--#set var="1var1" 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.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:

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:

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 SSI Example for SSI directive examples.

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] -->



<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.
- KCL:* — This entry limits access to any KCL commands 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** .

Note The Web Enhancements Option must be installed in order to get to this screen.

The following resources require authentication by default.

- iPendant
- KAREL:*
- KCL:*

Refer to Table 6–4 for information on the HTTP SETUP Screen Items.

Note Changes to the SETUP screen take effect immediately.

Table 6–4. HTTP SETUP Screen Items

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

```
HTTP Authentication Setup Screen (controller password option not active)

HTTP Setup
   PROTECTED RESOURCES
  Name  Pwrd  Resource
A ***** ***** iPendant
A ***** ***** KAREL:*
A ***** ***** KCL:*
A ***** ***** *********************
A ***** ***** *********************
A ***** ***** *********************
A ***** ***** *********************
A ***** ***** *********************
```

6.5.2.3 Robot Controller Password Option Enabled

If the controller password option is enabled then the HTTP Authentication Setup screen shown below will be displayed. See the following screen for an example. If the Resource is set to Authenticate then the name and password entered must correspond to a user defined within the password option setup screens. The level associated with that user must be at least equivalent to the levels set for HTTP Authentication of that resource.

The default level set for all resources is Install. This can be changed by importing a password configuration file from the SETUP Passwords menu.

```
HTTP Authentication Setup Screen (controller password option active)

HTTP Setup
   PROTECTED RESOURCES
  Resource
A iPendant
A KAREL:*
A KCL:*
A **********************
A **********************
A **********************
A **********************
A **********************
```

6.5.2.4 Example Configuration

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

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

HTTP Authentication Setup Screen with an Example Custom Configuration

```
HTTP Setup
   PROTECTED RESOURCES
  Name   Pwrd  Resource
A*****  ***** iPendant
A*****  ***** KAREL:*
A*****  ***** KCL:*
U*****  ***** FR:*.HTM
A*****  ***** FR:*.*
A*****  ***** **********************
A*****  ***** ***********************
A*****  ***** ***********************
```

6.5.2.5 Accessing 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 Chapter 18 *ADVANCED iPENDANT FUNCTIONS* for more information.

6.6 REALLY SIMPLE SYNDICATION

6.6.1 Overview

RSS (Really Simple Syndication) is a popular web feed format used for Web syndication. RSS is commonly used by such services as news websites, web blogs, and podcasting. Web feeds provide web content or web content summaries together with links to the full versions of the content. In addition to facilitating syndication, web feeds allow a website's readers to track updates on a site using an aggregator, or an RSS viewer.

The robot controller uses RSS as a convenient way to deliver alarm content to a remote PC. Not only can alarm error and cause code information be displayed, but links to DRC Cause/Remedy information are provided, allowing you to assess quickly the nature of an alarm and the recommended remedy.

Because RSS is a popular open standard used on the Internet today, a plethora of compatible PC software is freely and commercially available, including the latest Mozilla and Microsoft® web browsers.

The Internet Protocol Connectivity and Configuration (IPCC) option must be loaded to use RSS.

6.6.2 Setting Up and Using RSS

RSS is automatically available after the Internet Protocol Connectivity and Configuration (IPCC) option is loaded on the robot controller. No additional setup is needed.

Robot RSS alarm feeds can be viewed or subscribed to using any RSS viewer or aggregator supporting RSS version 2.0. In this section, we will walk through an example of subscribing to an alarm feed using Microsoft® Internet Explorer 7.

First we access the robot controller's default we page. Then we scroll down and click the link to "Monitor Alarms using RSS", as seen in Figure 6–12 and Figure 6–13 .

Figure 6–12. Robot Controller Default Web Page

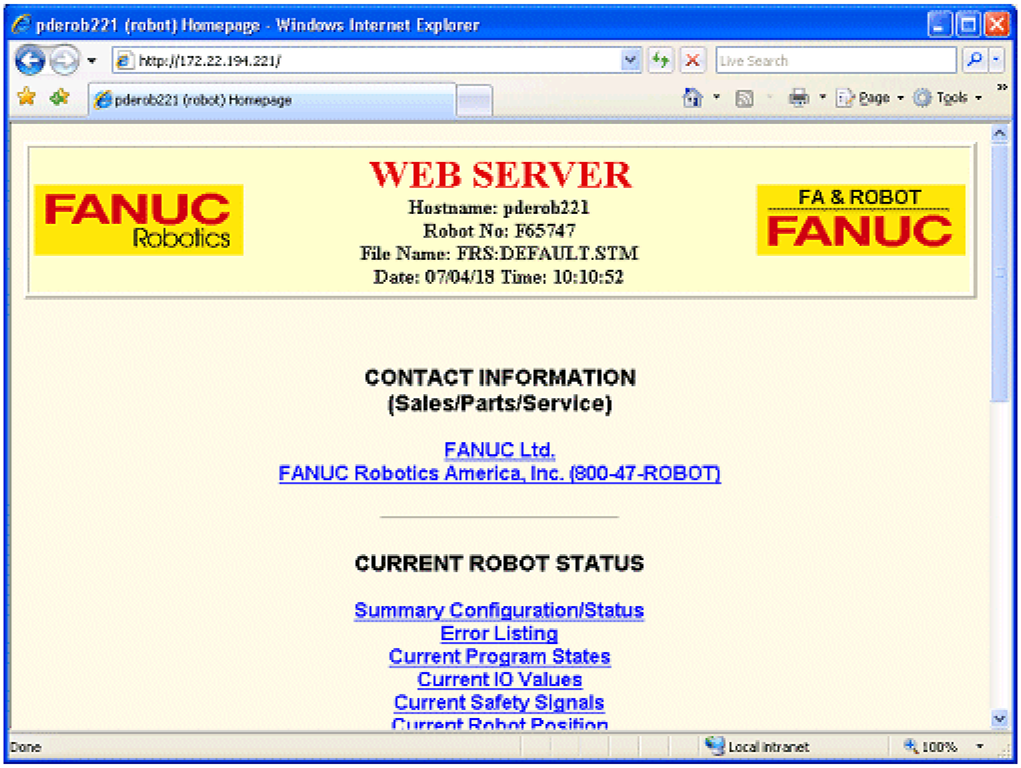


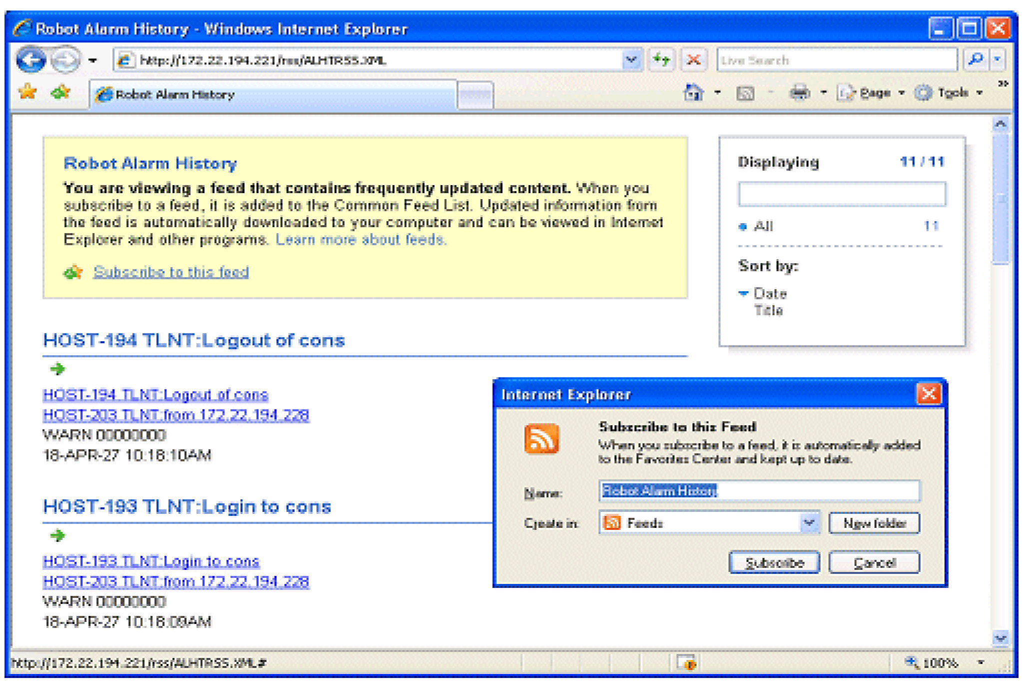
Figure 6–13. Monitor RSS Alarm Feeds Link

After the RSS Alarm Feeds page is displayed, the RSS icon on the right side of the Internet Explorer toolbar will turn orange, indicating RSS Feeds exist on this web page. Clicking on the orange RSS icon gives you a dropdown containing the available robot alarm feeds, as shown in Figure 6–14 .

Figure 6–14. RSS Alarm Feeds Page

After selecting an Alarm Feed, you are given the opportunity to subscribe to the Feed, as seen in Figure 6–15 . To subscribe, click on the “Subscribe to this feed” link in the yellow box.

Figure 6–15. Subscribe to this Feed



After you have successfully subscribed to the feed, you can view and modify the feed properties by clicking on “View feed properties” link on the right side of the web page: see Figure 6–16 .

Figure 6–16. Successfully Subscribed

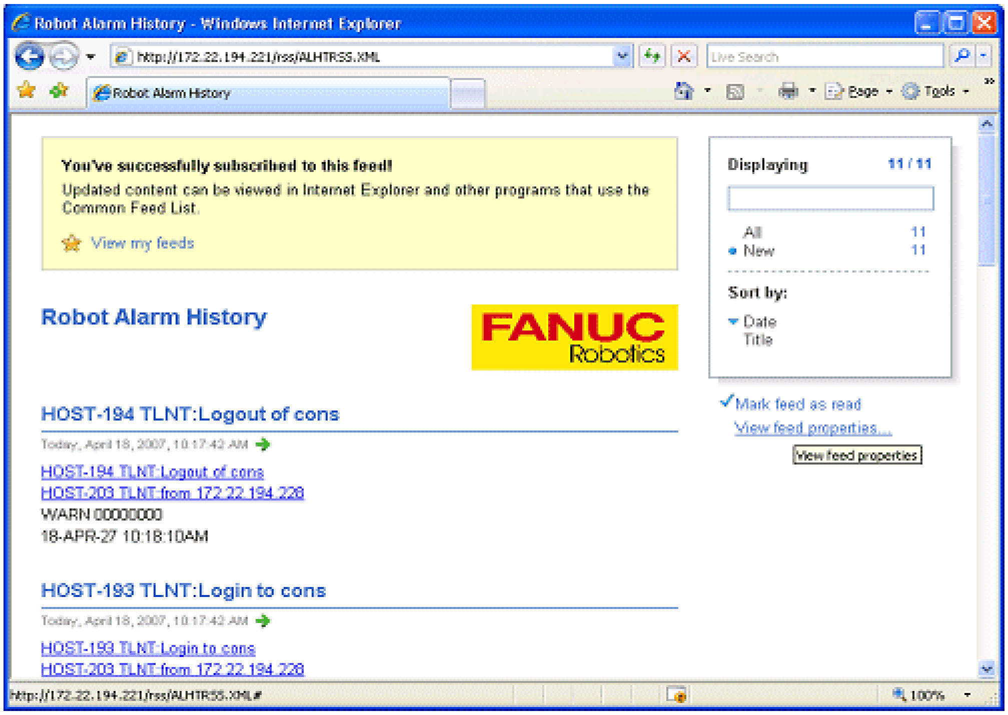
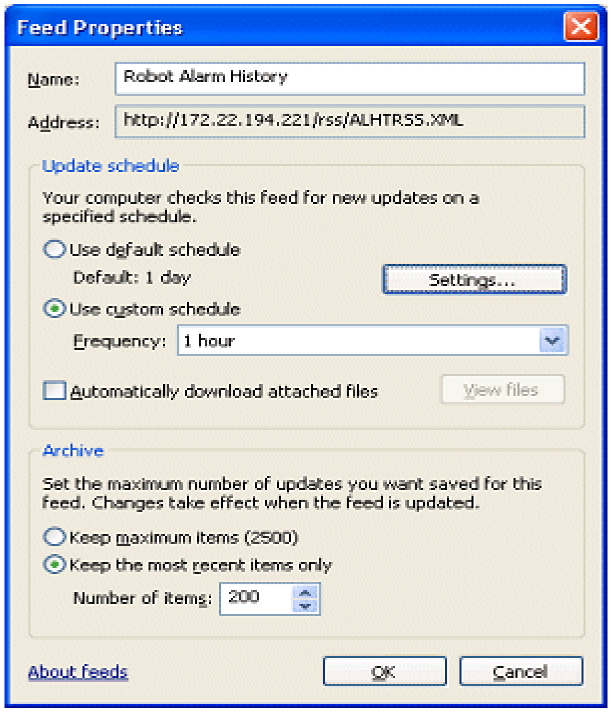


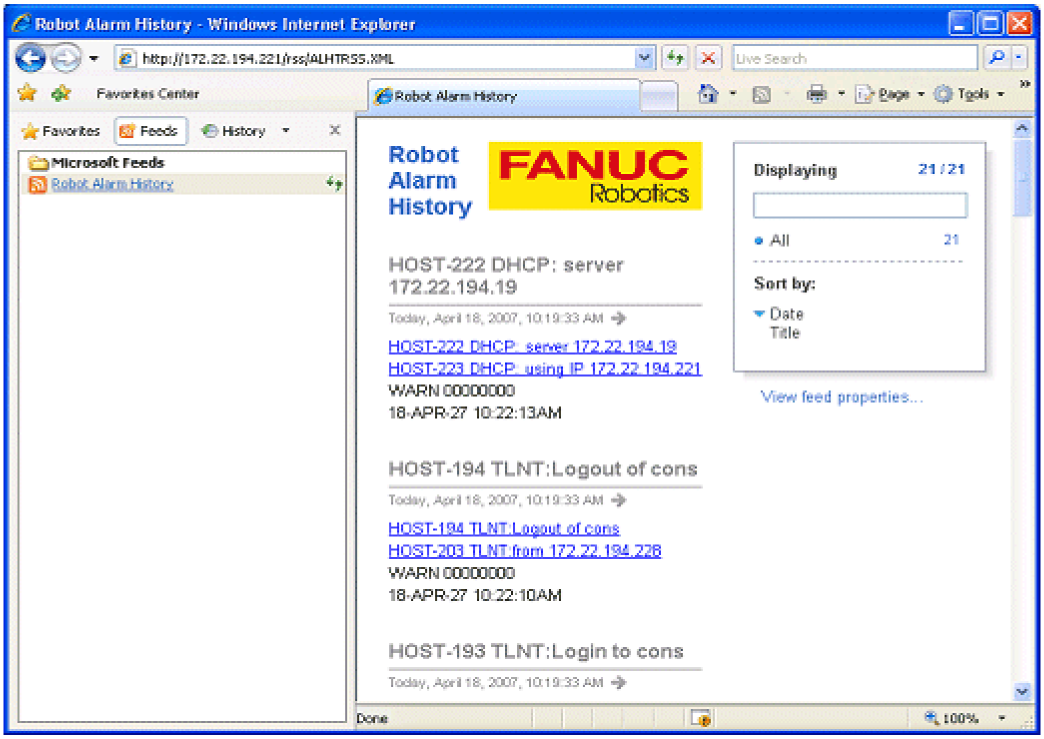
Figure 6–17 shows the configurable feed properties: Update Schedule, and Archive settings.

Figure 6–17. Feed Properties



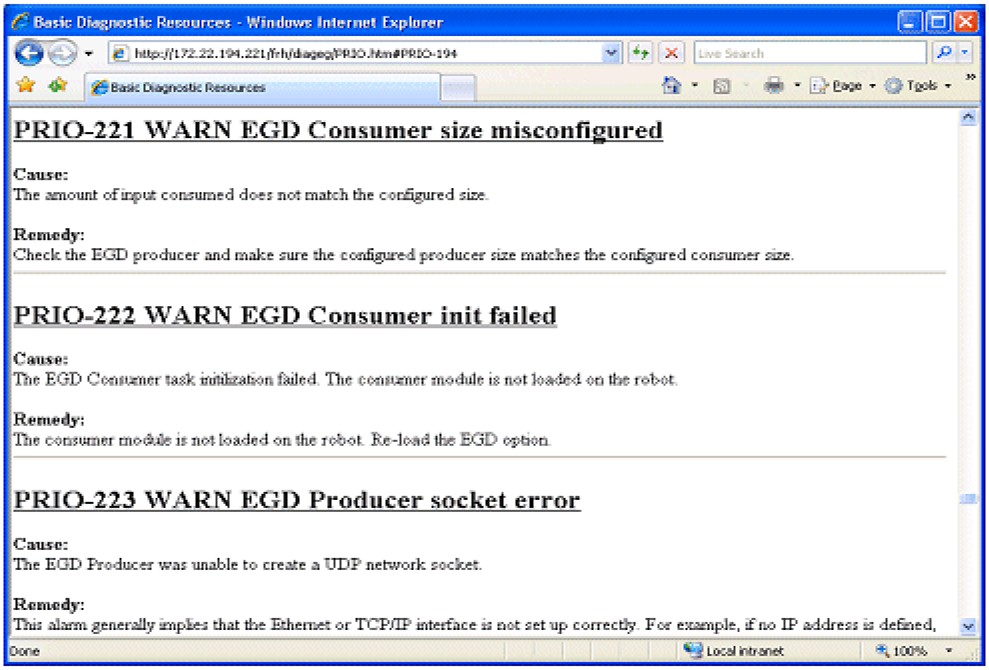
After you have subscribed, you can access your Alarm Feeds from the Favorites Center as shown in Figure 6–18 .

Figure 6–18. Favorites Center Feeds Menu



By clicking on the links provided in the feed, alarm cause remedy information can be quickly accessed. For example, Figure 6–19 shows the cause remedy information of a PRIO-221 alarm as accessed through the RSS Feed.

Figure 6–19. Alarm Cause Remedy Information



Chapter 7

PROXY SERVER

Contents

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

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

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

Note If **PROXY** protocol does not show up on the screen, you do not have the proxy server option installed.

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

```
Proxy Setup
External Proxy
  Enable : FALSE
  Server : **************************
  Port :   8080
Exceptions:
  1 **********************************
  2 **********************************
  3 **********************************
  4 **********************************
  5 **********************************
  6 **********************************
  7 **********************************
  8 **********************************
```

5. To operate in mode 2 (allow limited access to web server on the building network), leave (External Proxy) Enable to be FALSE. Scroll to the Exceptions and enter the host names that you want to allow an 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 first case, all host names starting with 192.168.0 will be allowed. In the third case, an exact match for the hostname will be performed.

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

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

7.3 ERRORS RETURNED BY THE PROXY SERVER

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

- HTTP 400 — Bad Request: The request was not in the expected form. The expected form is http://hostname/...
- HTTP 403 — Forbidden: You are operating in mode 2 and were trying to browse a web server that was not in the exception list.
- HTTP 414 — Request URI Too Long The request (http://hostname/..) was longer than 4 Kbytes. The proxy server can handle requests only up to 4 Kbytes long. The content length can be any size but the URI can only be 4 Kbytes long.
- HTTP 500 — Internal Server Error: There was a problem opening connections as the system is out of resources.
- HTTP 502 — Bad Gateway: The hostname in the web request could not be resolved to an IP address. If you are using an external proxy server, the IP address does not match. Or, the web server you are trying to get does not respond. Verify that you have the DNS option installed or you have the hostname of the web server being used in the URL in the host entry table.

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.

Chapter 8

POINT-TO-POINT PROTOCOL CONNECTIVITY

Contents

8.1 OVERVIEW

Point-to-Point Protocol (PPP) allows devices to connect to each other across a dedicated point to point link.

The controller supports up to one user PPP connection via a serial port or with a modem installed in your controller. All internet options, except Ethernet Image Backup and Restore and BOOTP/DHCP, are available for devices to use over the PPP link.

8.2 SETTING UP PPP ON YOUR CONTROLLER

8.2.1 Overview

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

You can make remote dial-in PPP connections to your robot, either through external modems installed on the P2 or P3 serial ports.

IP Addresses

Table 8–1 and Table 8–2 show the default IP Addresses for the P2 and P3 ports.

Table 8–1. Addresses for P2 Port (Direct Serial Port or External Modem)

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

Table 8–2. Addresses for P3 Port (Direct through Serial Port or External Modem)

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

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

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

Supported Modems

The following external modems are supported:

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

8.2.2 Configuring the P2, and P3, Ports

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

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

Steps

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

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

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

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

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

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

Steps

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

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

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

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

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

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

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

8.2.3 Changing IP Addresses

When assigning IP addresses to ports P2 and P3 you should use the default values listed in Table 8–1 and Table 8–2 . 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 MENU.
2. Select SETUP.
3. Press F1, [TYPE]
4. Select Host Comm.
5. Move the cursor to PPP.
6. Press F3, DETAIL.

You will see a **Port initialized for PPP or PPP/Modem** message.

7. Press F3, DETAIL. You will see a screen similar to the following.

```
SETUP PPP Port
   P3

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

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

8.3 SETTING UP PPP ON YOUR PC

8.3.1 Overview

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

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

8.3.2 Setting up PPP on a Network PC

You should configure your PC for PPP connection. This manual contains step by step instructions on how to configure PC with Windows NT, Windows 2000 and Windows XP operating systems. Refer to Procedure 8-4 , Procedure 8-5 , Procedure 8-6 , and Procedure 8-7 .

Procedure 8-4 Setting up PPP on a PC with Windows NT

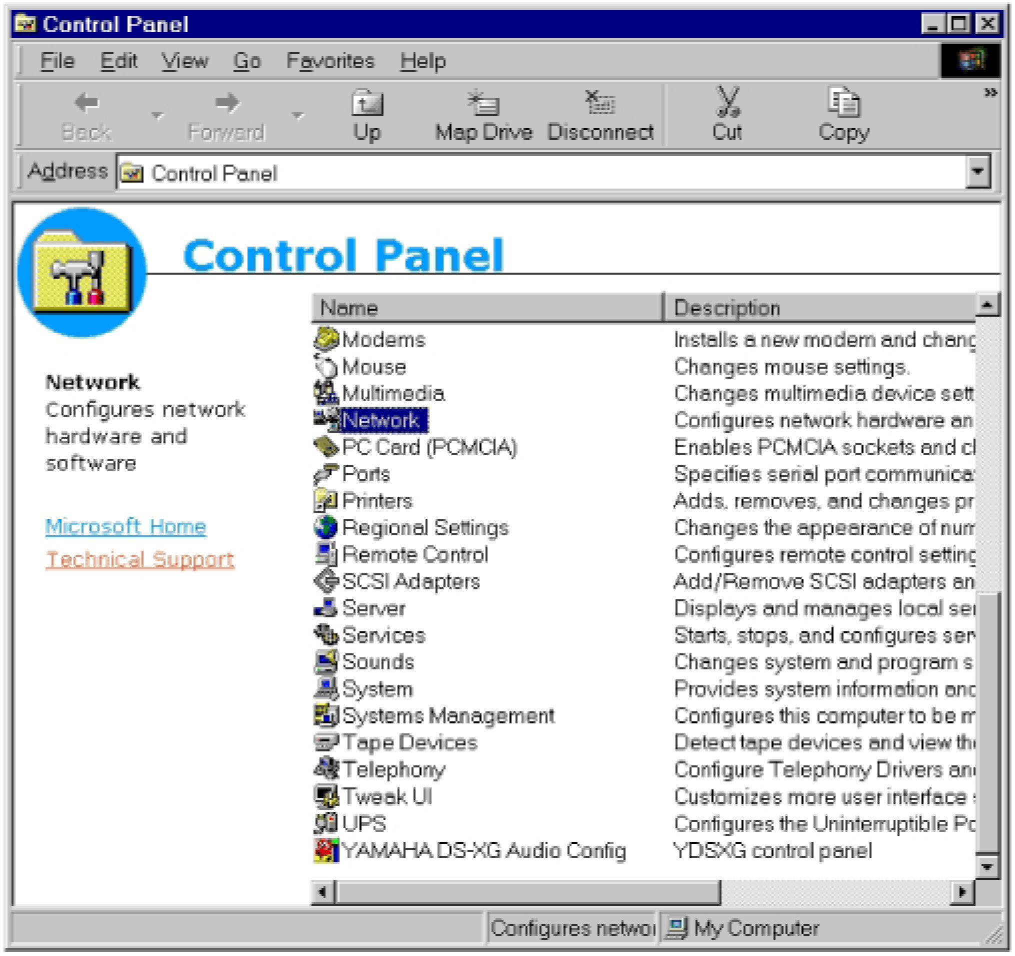
Conditions

- Make sure RAS on the PC is set up for dial-out only.

Steps

1. Click the **My Computer** Icon on the Desktop.
2. Double-click **Control Panel**. The following window will be displayed.

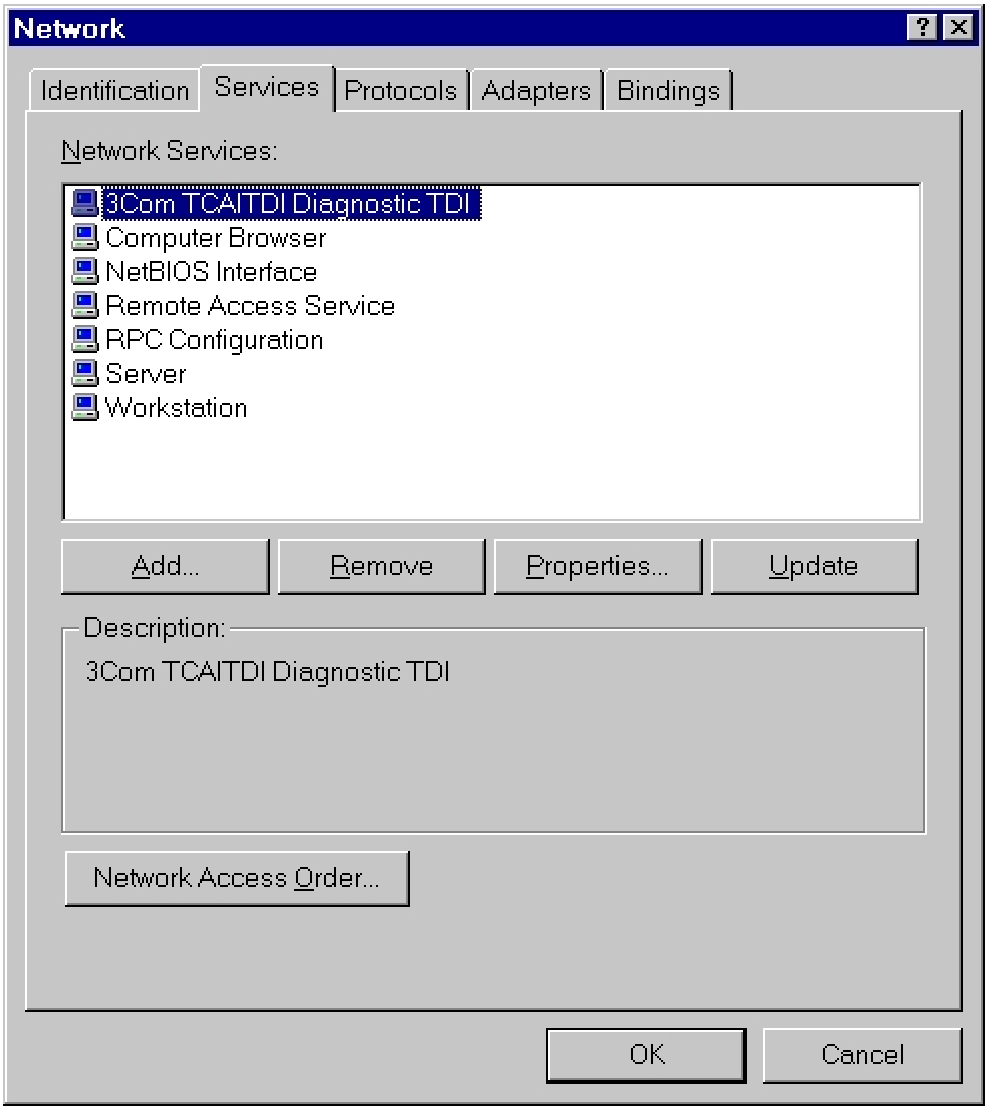
Figure 8–1. Control Panel Screen



3. Double-click **Network** to configure network settings.

4. Select the Services tab as shown in Figure 8–2 .

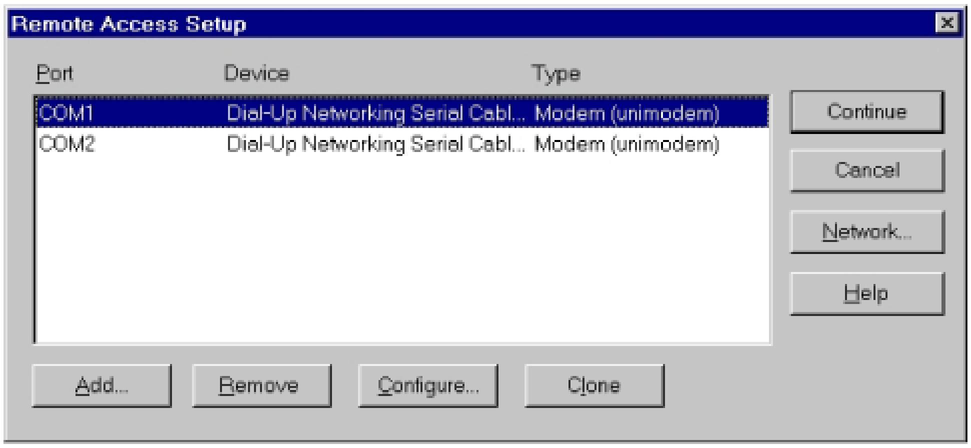
Figure 8–2. Network Screen



5. Select Remote Access Service and click Properties.

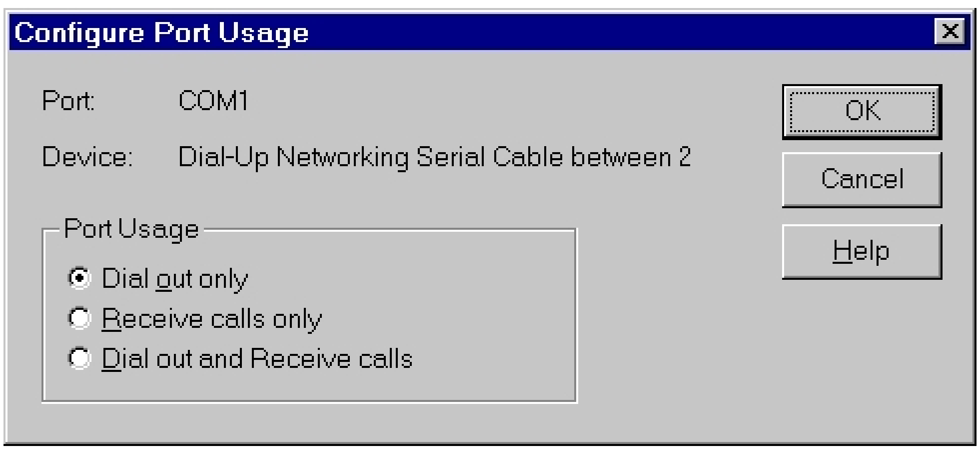
Note "Remote Access Service" should be present. If it is not present, click **Add...** and follow the directions provided by the Windows NT installation manual to install Remote Access Service (RAS). See Figure 8–3 .

Figure 8–3. Remote Access Setup Screen



6. Select the Device to configure (Modem or Direct Serial Connection) and click Configure. See Figure 8–4 .

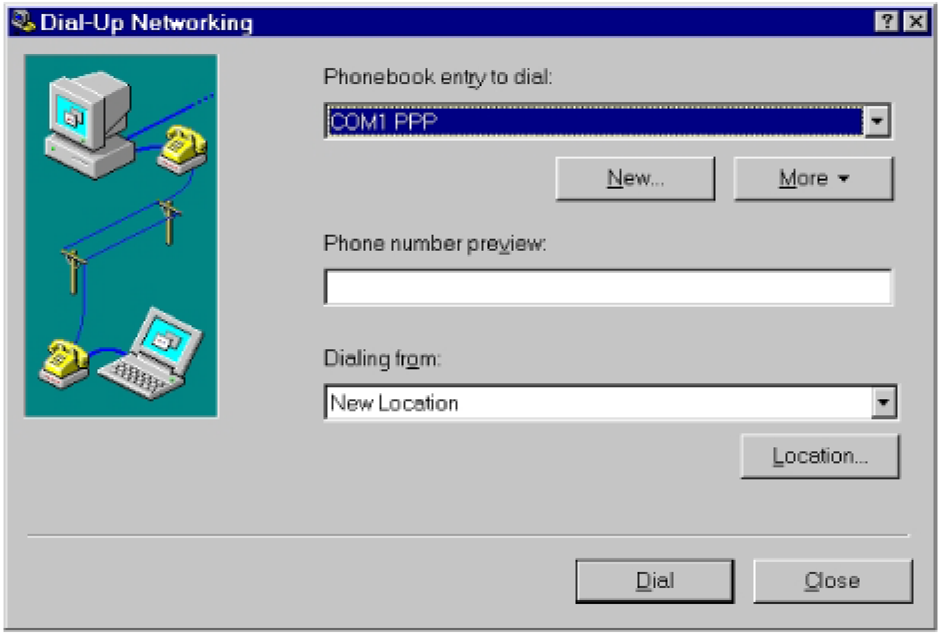
Figure 8–4. Configure Port Usage Screen



7. Select **Dial Out Only**.

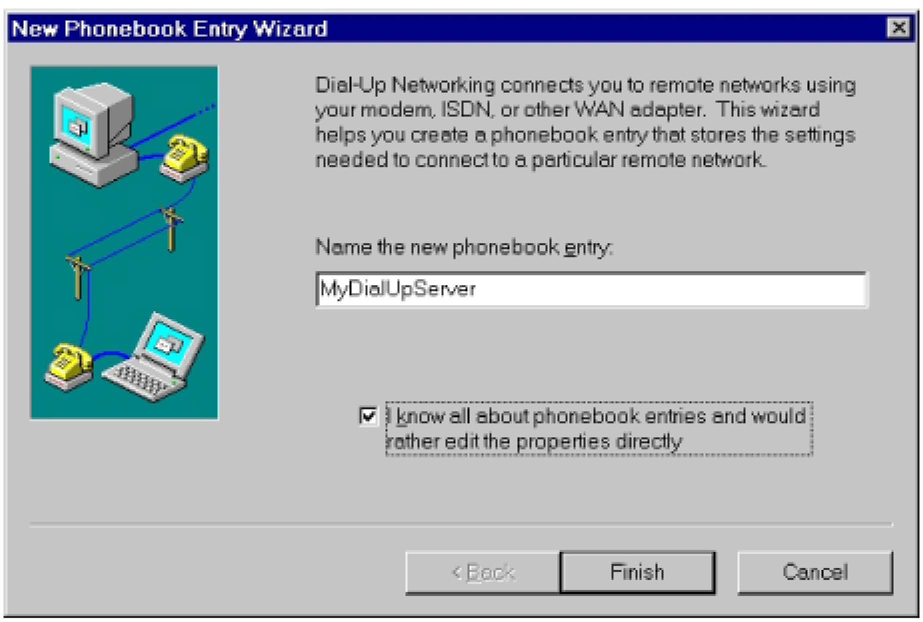
8. Select the icon **My Computer** on the desktop. You must double-click on Dial-Up Networking and select the **New** button. See Figure 8–5 .

Figure 8–5. Dial-Up Networking



9. Type the name you want for this connection in the **New Phone Book Entry Wizard** window that is displayed. You must check the box that states the following: **I know all about phonebook entries and would rather edit the properties directly.**

Figure 8–6. New Phonebook Entry Wizard



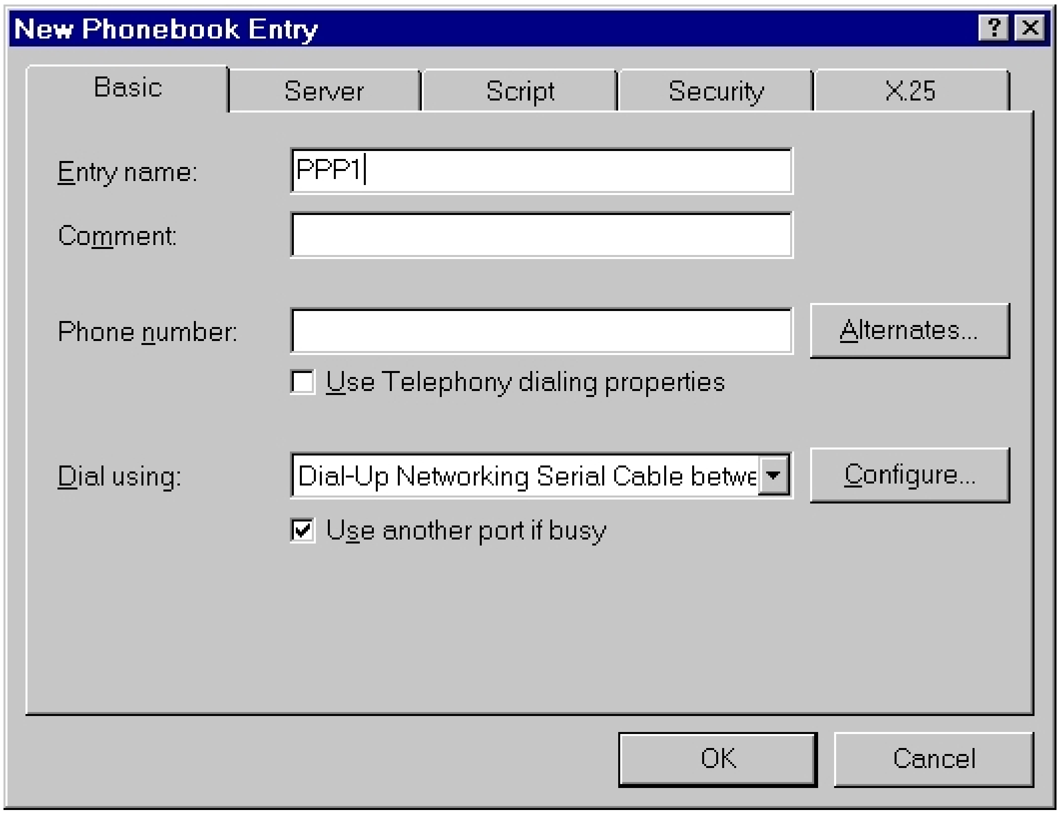
10. Set up the new phone book entry as follows:

a. Type a name for your connection in the **Entry name** box. See Figure 8–7 .

b. Select the network device you want to use for your RAS connection from the **Dial using** box. You can select either a Modem, or Dial-up Networking using a Serial cable.

c. Click Configure, and proceed to Step 11 to configure your network device.

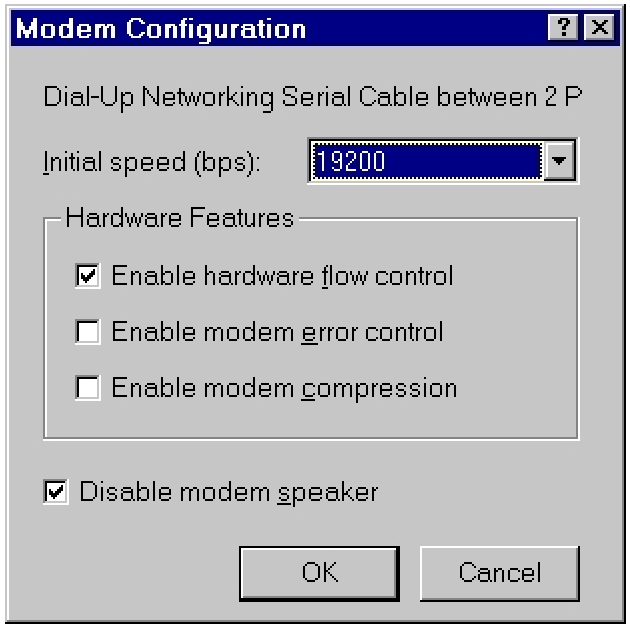
Figure 8–7. New Phonebook Entry



11. Configure the modem or serial port. Do the following from the Modem Configuration dialog box:

a. Turn off all hardware flow control, modem error control, and modem compression by leaving the "Hardware features" check boxes unchecked. See Figure 8–8 .

Figure 8–8. Configure the Serial Device



b. Click OK to save your modem or serial port configuration.

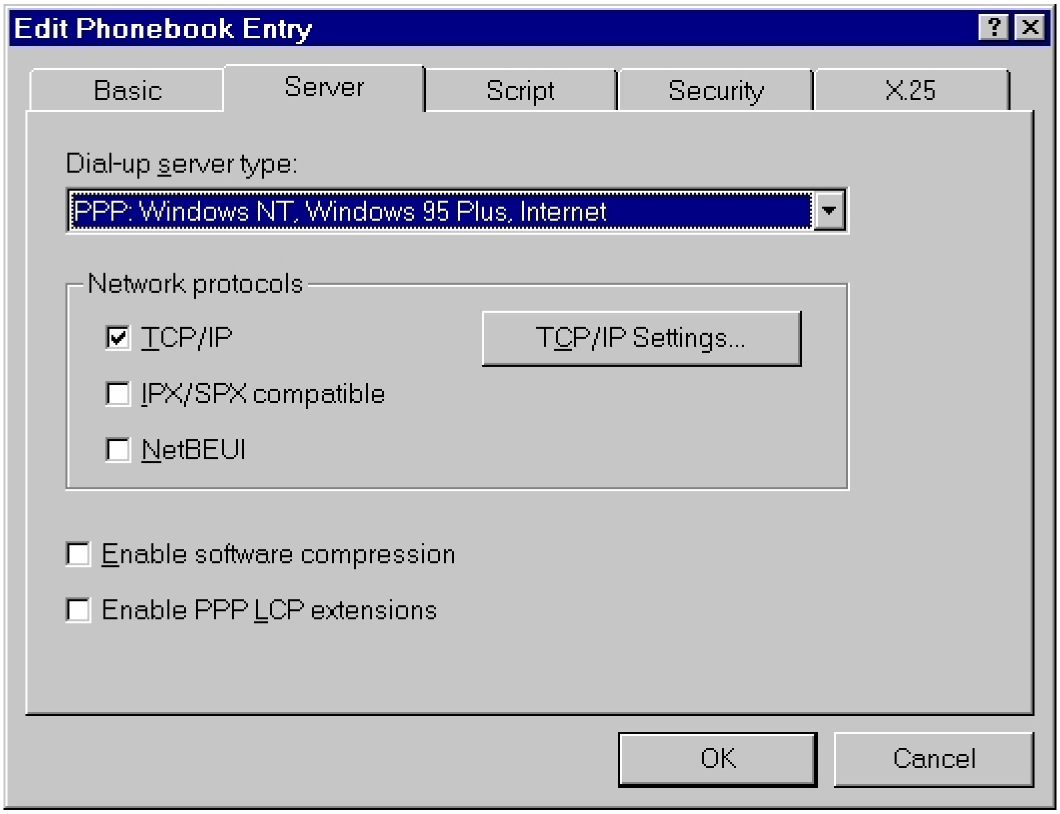
12. Select the network protocol

a. Click the Server tab on the Edit Phonebook Entry dialog box.

b. Use the **Dial up server type** box to select PPP: Windows NT, Windows 95 Plus, Internet, as shown in Figure 8–9 .

c. From the **Network protocols frame**, select TCP/IP, as shown in Figure 8–9 Do not use any of the other available network protocols.

Figure 8–9. Select the Network Protocol



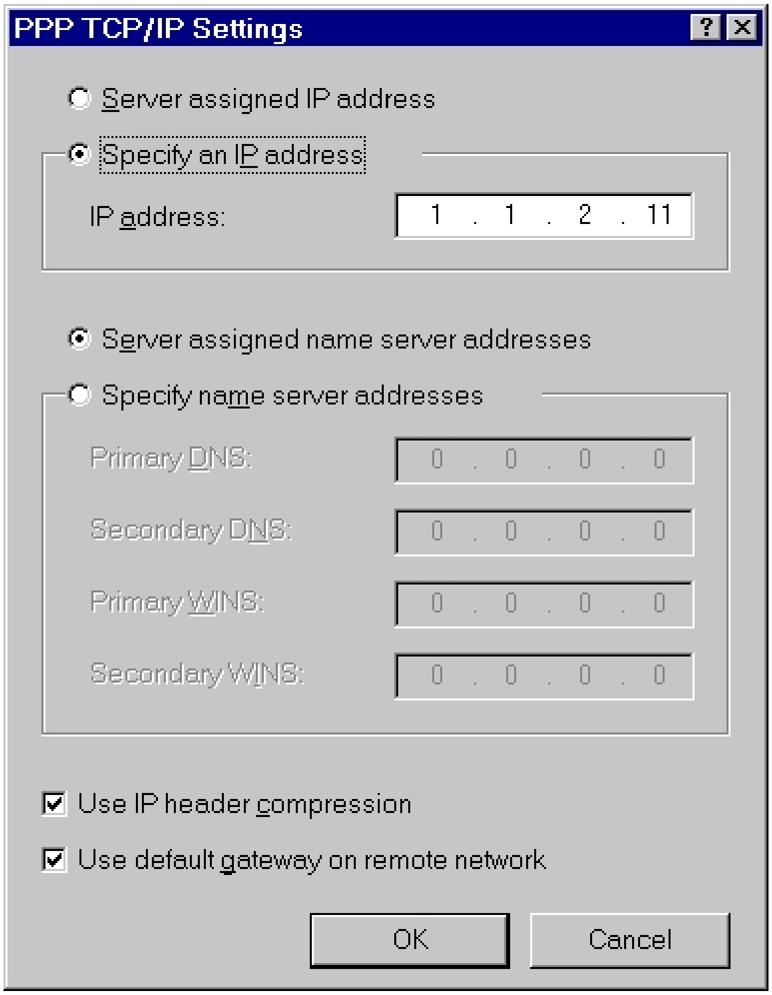
d. Turn off software compression and disable all PPP LCP extensions, as shown in Figure 8–9 Click the TCP/IP Settings button, and proceed to Step 13 to configure the PPP TCP/IP settings.

13. Configure the PPP TCP/IP settings. Do the following from the PPP TCP/IP Settings dialog box:

a. Choose the IP address of the PC from the addresses specified in Table 8–1 in Section 8.2 .

b. Set all DNS and WINS server addresses to zero.

Figure 8–10. Select the Network Protocol



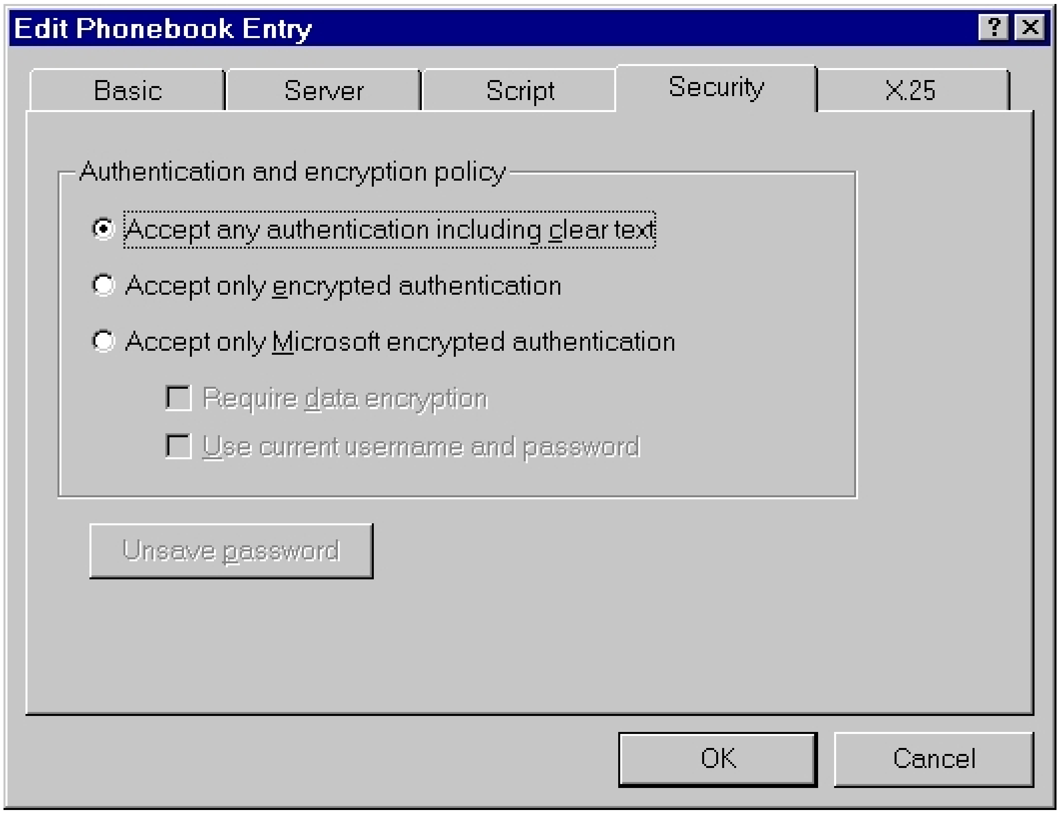
c. Check the **Use IP header compression** and **Use default gateway on remote network** checkboxes, as shown in Figure 8–10 . Click OK to save your PPP TC/IP settings.

14. Click the Security tab on the Edit Phonebook Entry dialog box .

a. In the **Authentication and encryption policy** frame, enable the option to Accept any authentication including clear text, as shown in Figure 8–11 . This will allow for plain text passwords.

Note The controller does not have user name and password authentication capabilities. It will accept any (including blank) username and a blank password.

Figure 8–11. Select the Network Protocol



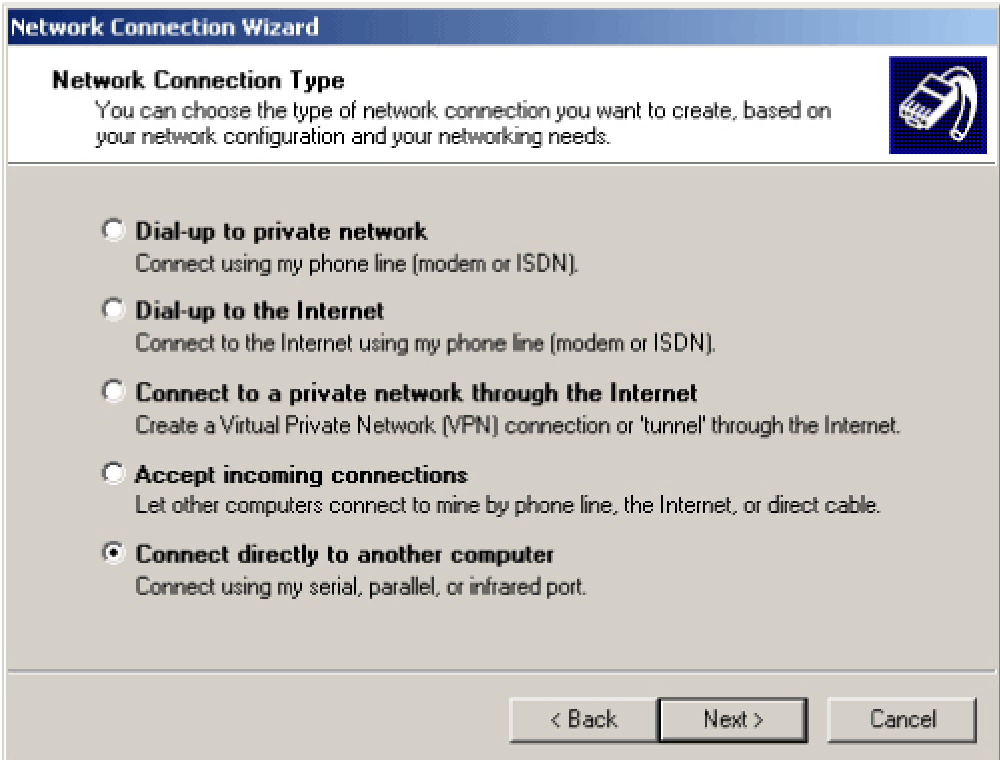
15. Click **OK** and then dial to connect to the robot.

Note Refer to the software documentation for you PC's operating system for detailed information about how to install and set up the RAS software.

Procedure 8-5 Setting Up PPP on a PC with Windows 2000

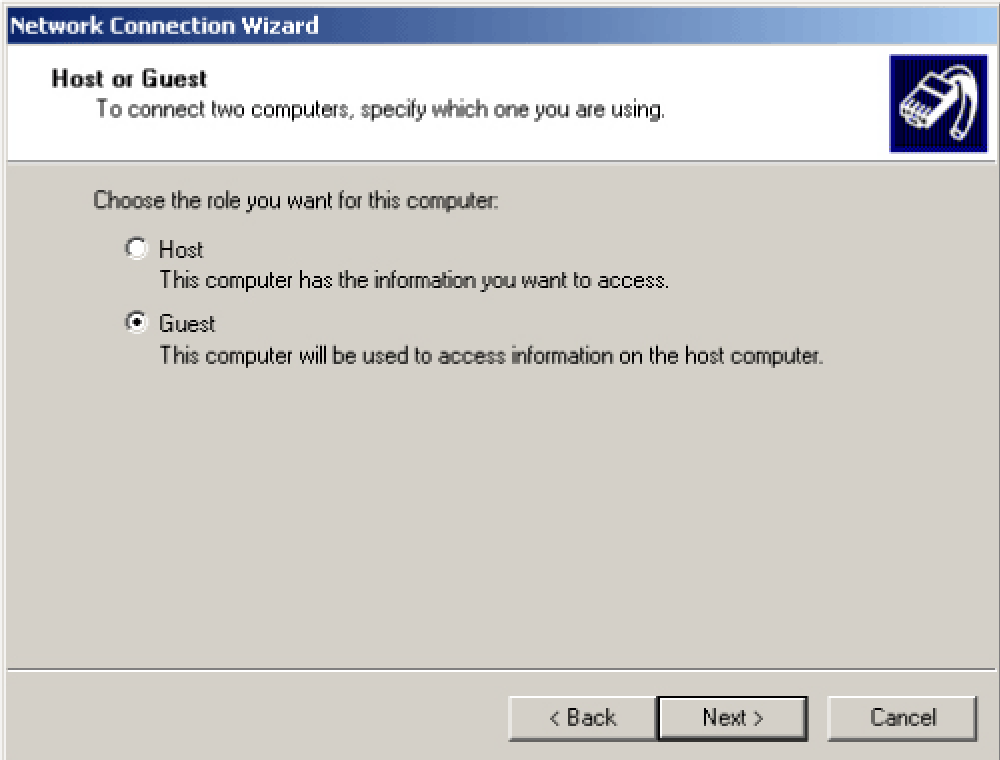
1. Click **Control Panel**, **Network and Dial Up Connections**, and **Make a New Connection**. You will see a screen similar to the following.

Figure 8–12. Control Panel



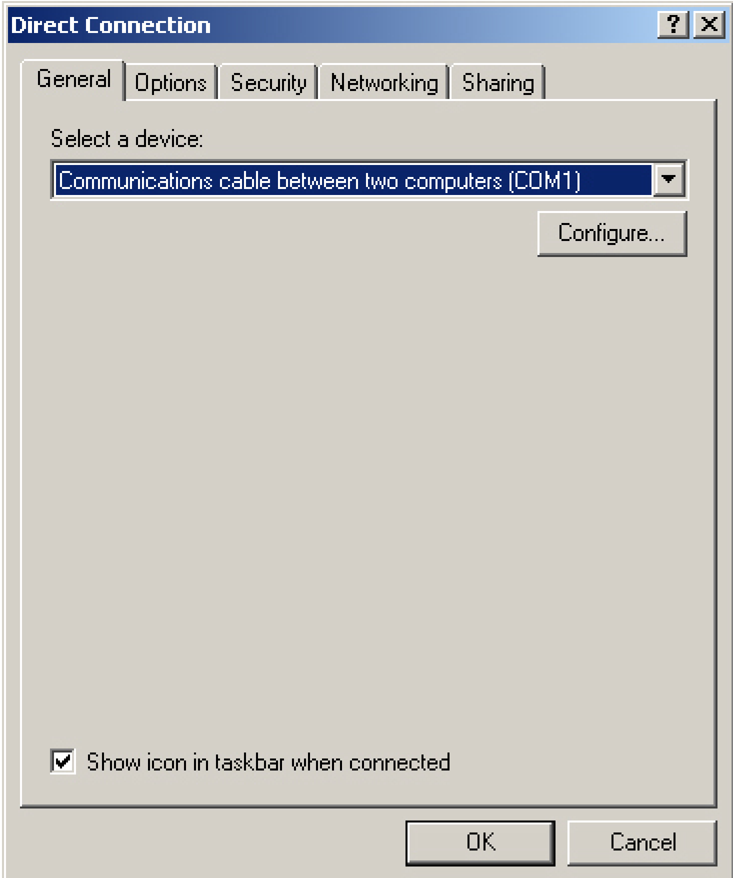
2. In the Network Connection Wizard, select **Connect directly to another computer** and click Next. You will see a screen similar to the following.

Figure 8–13. Network Connection



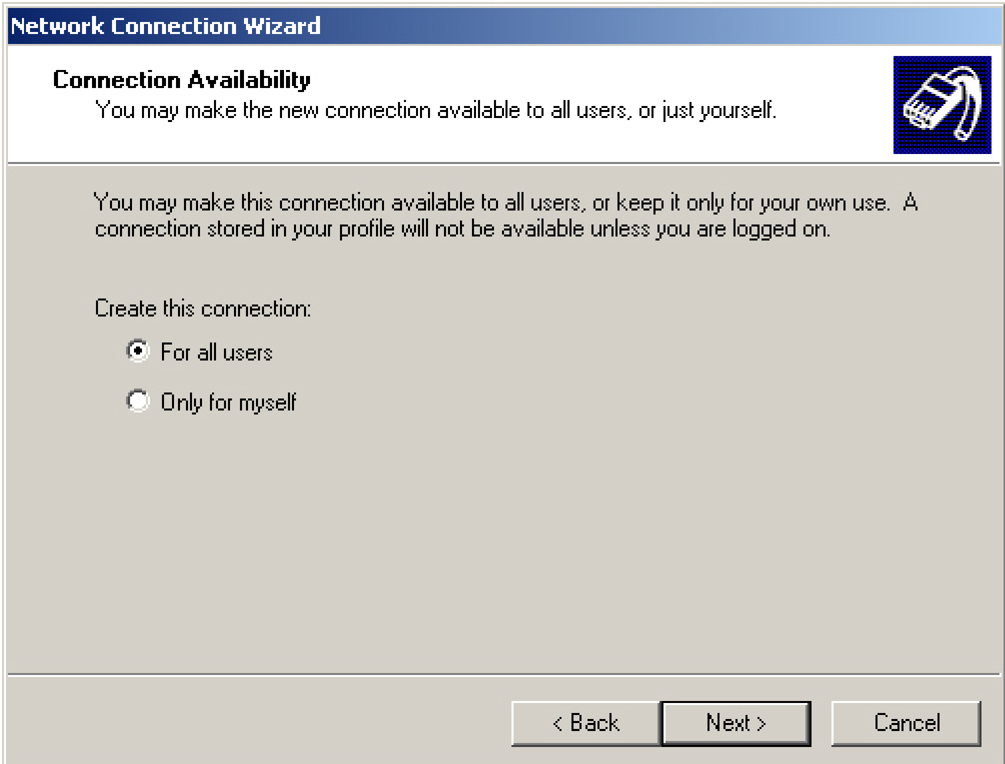
3. Select **Guest**. The robot controller will be the host and you must click Next to continue. You will see a screen similar to the following.

Figure 8–14. Guest



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

Figure 8–15. Select a Device

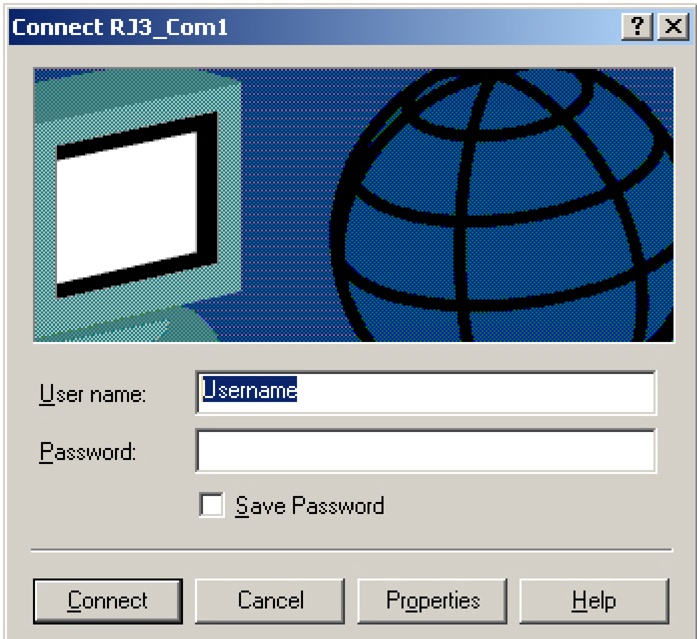


5. If you want all users who log on to your PC to use this connection, select **For all users** and click Next. You will see a screen similar to the following.

Figure 8–16. For All Users

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

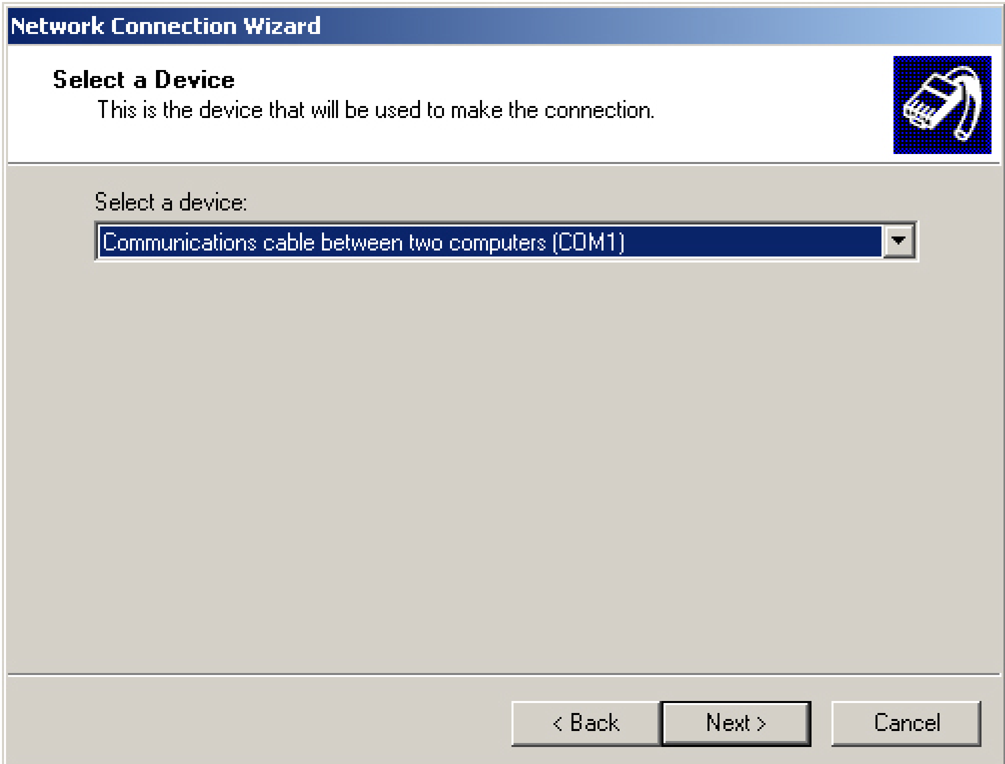
Figure 8–17. Select Finish



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.

Figure 8–18. Configure Properties

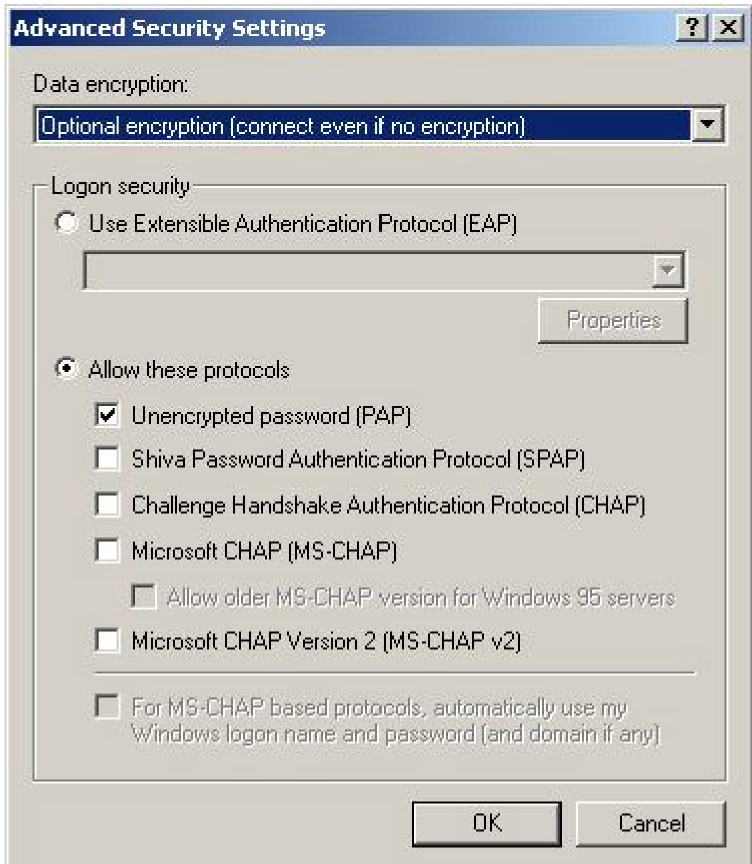


9. Click Next. You will see a screen similar to the following.

Figure 8–19. Security

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

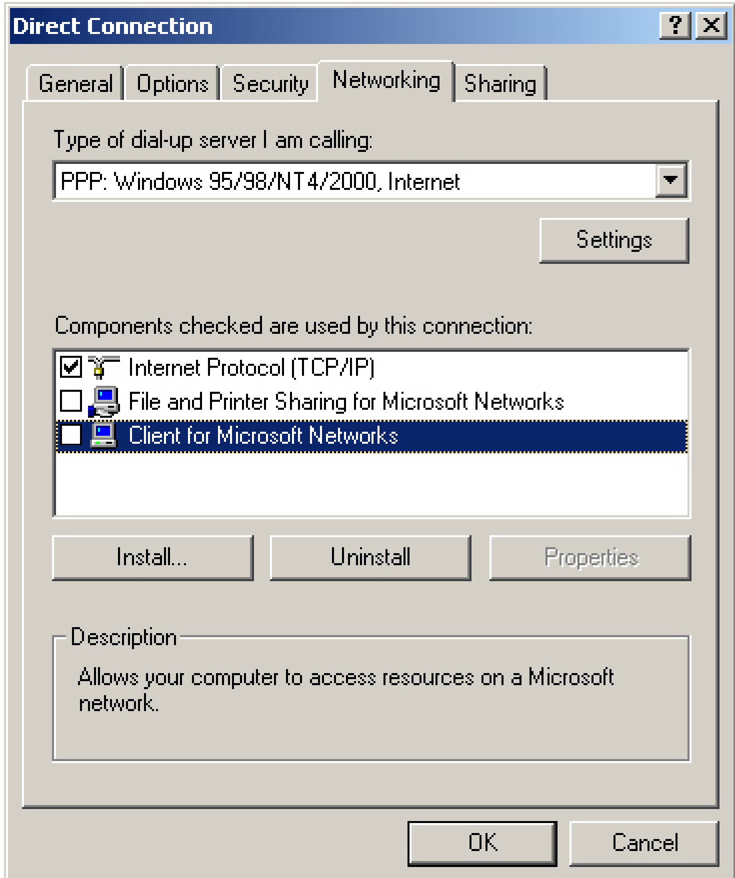
Figure 8–20. Advanced (Custom Setting)



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

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

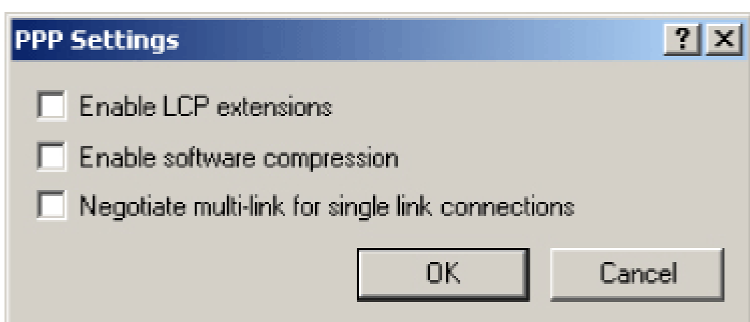
Figure 8–21. Networking



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

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

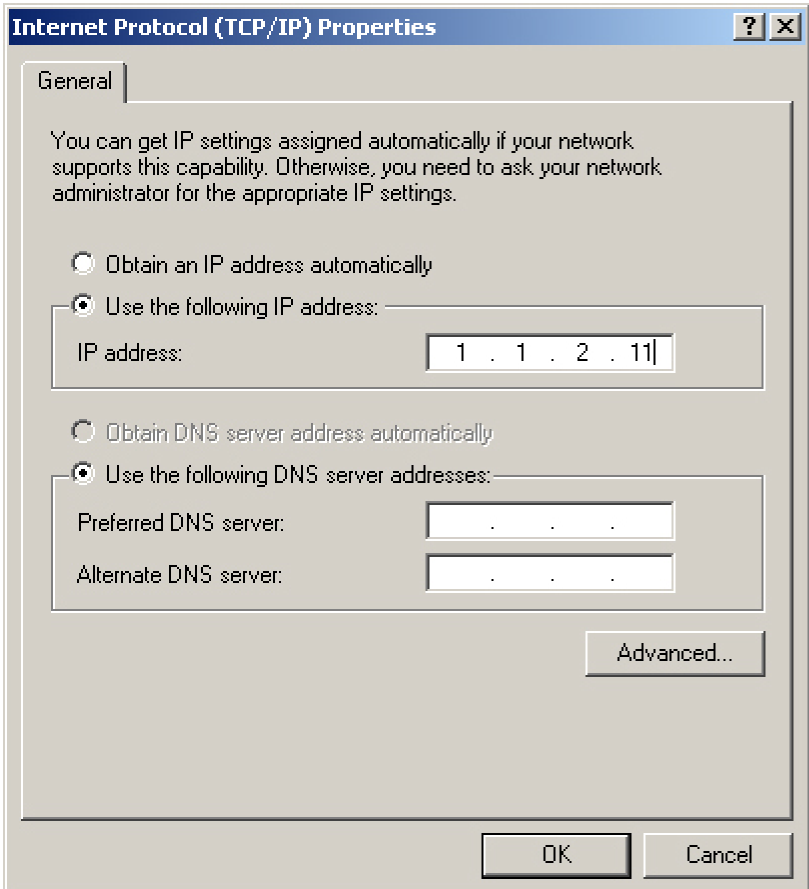
Figure 8–22. TCP/IP Settings



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

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

Figure 8–23. TCP/IP Settings



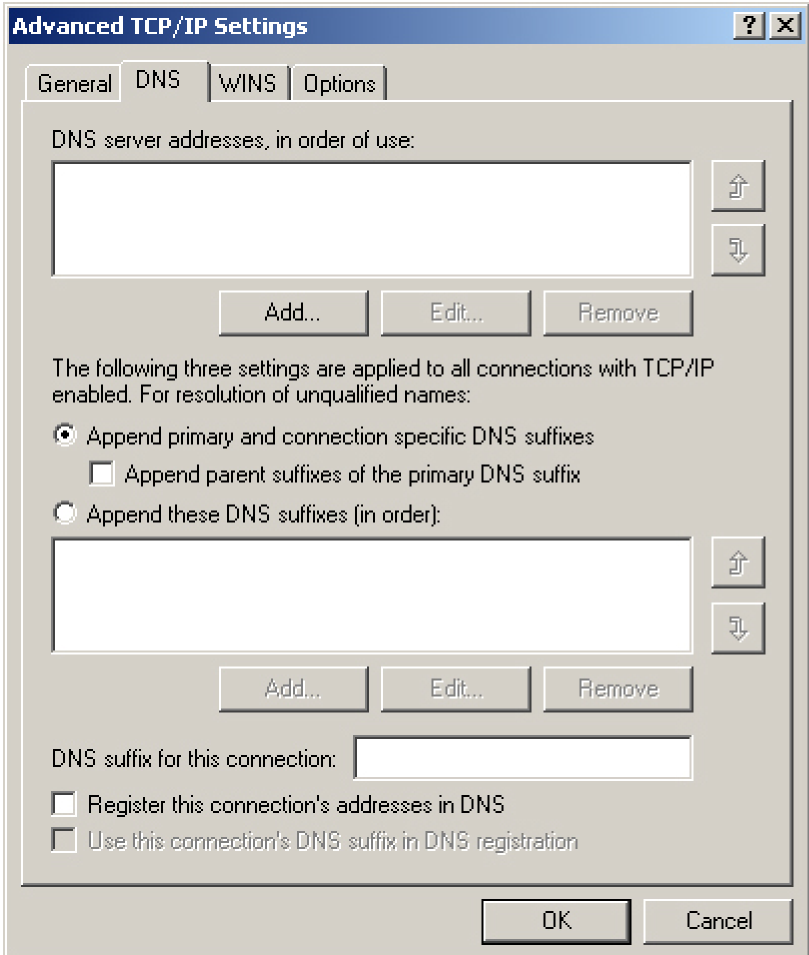
16. Type the IP address corresponding to the serial port you are using. Leave the entries for the DNS server address blank.

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

Figure 8–24. Advanced Settings

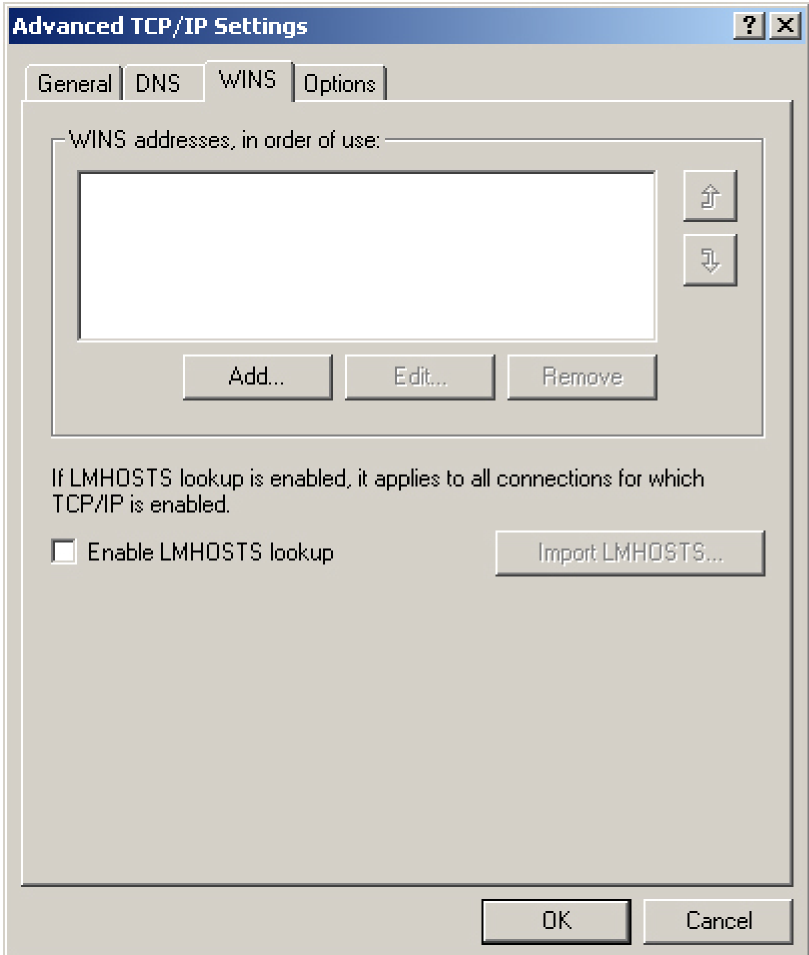
18. Select the **DNS** tab. You must make sure that the boxes are checked/unchecked. See the following screen for an example.

Figure 8–25. DNS



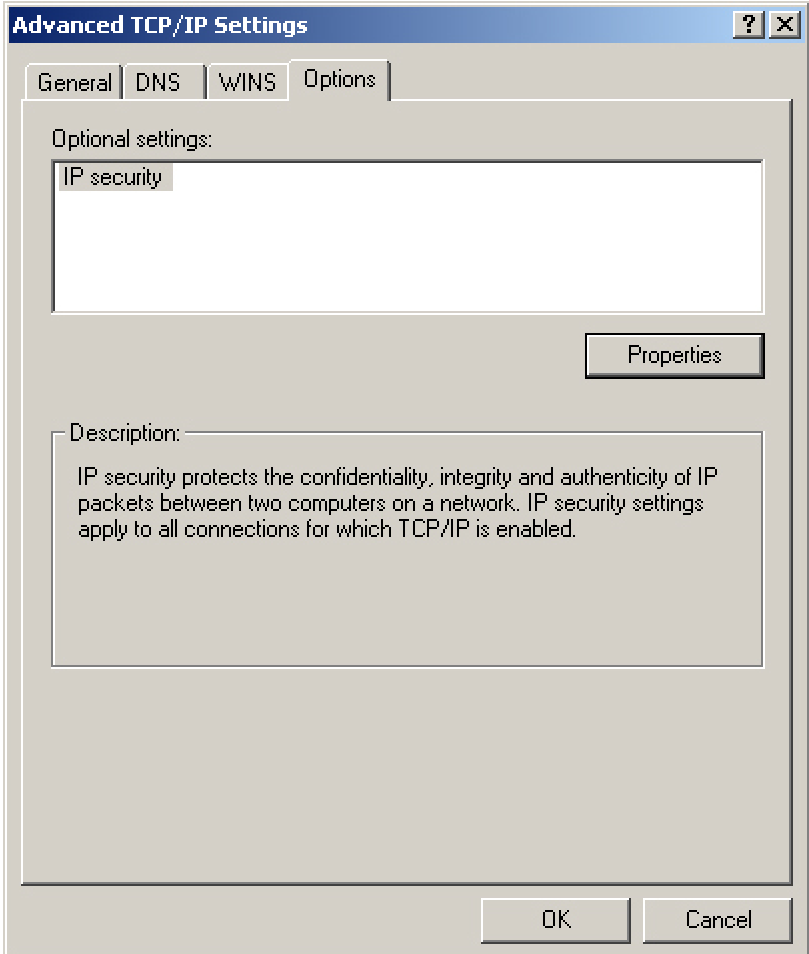
19. Select the **WINS** tab. Uncheck the box **Enable LMHOSTS lookup.**

Figure 8–26. WINS Enable LMHOSTS



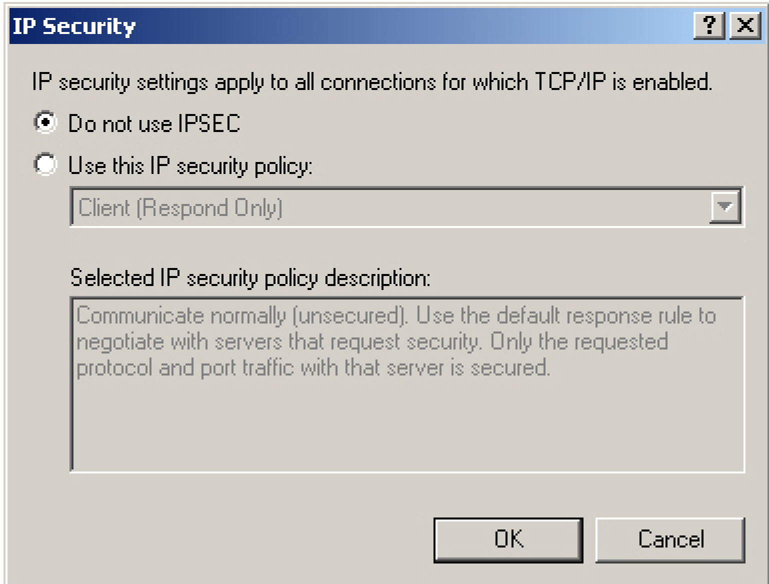
20. Select the **Options** tab. You will see a screen similar to the following.

Figure 8–27. Options



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

Figure 8–28. IP Security

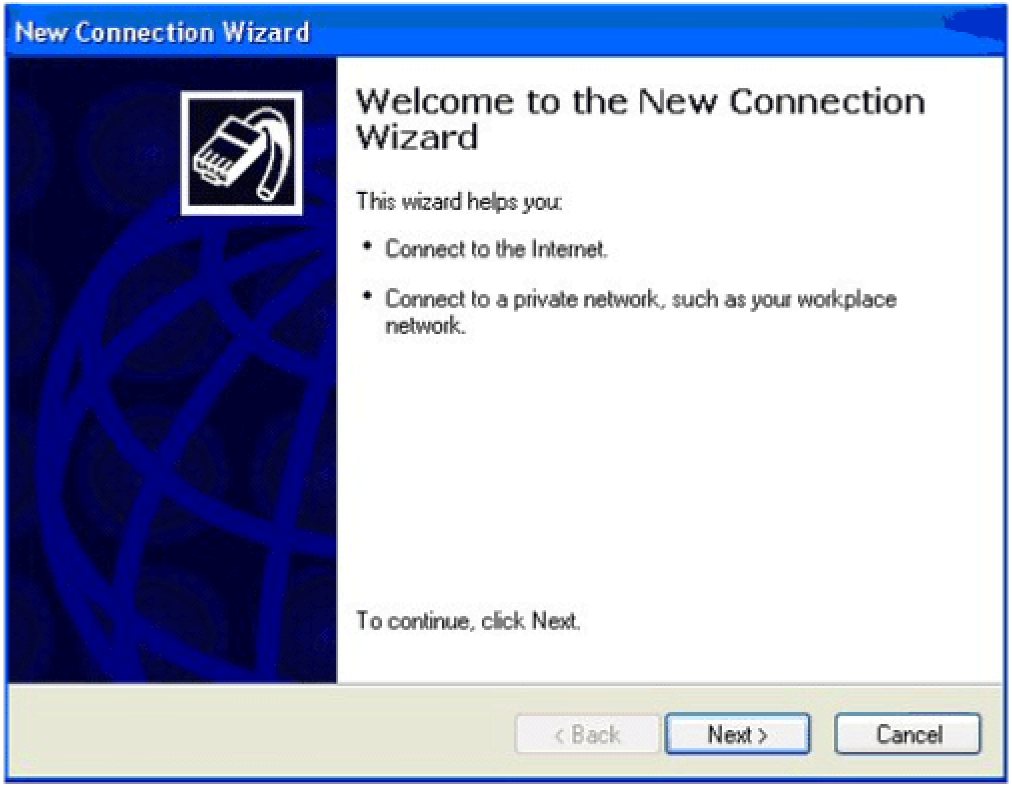


22. You must make sure the **Do not use IPSEC** button is selected. Select **OK.**
23. If a window pops up with the message **WINS entry is empty** , select OK to ignore the message.

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

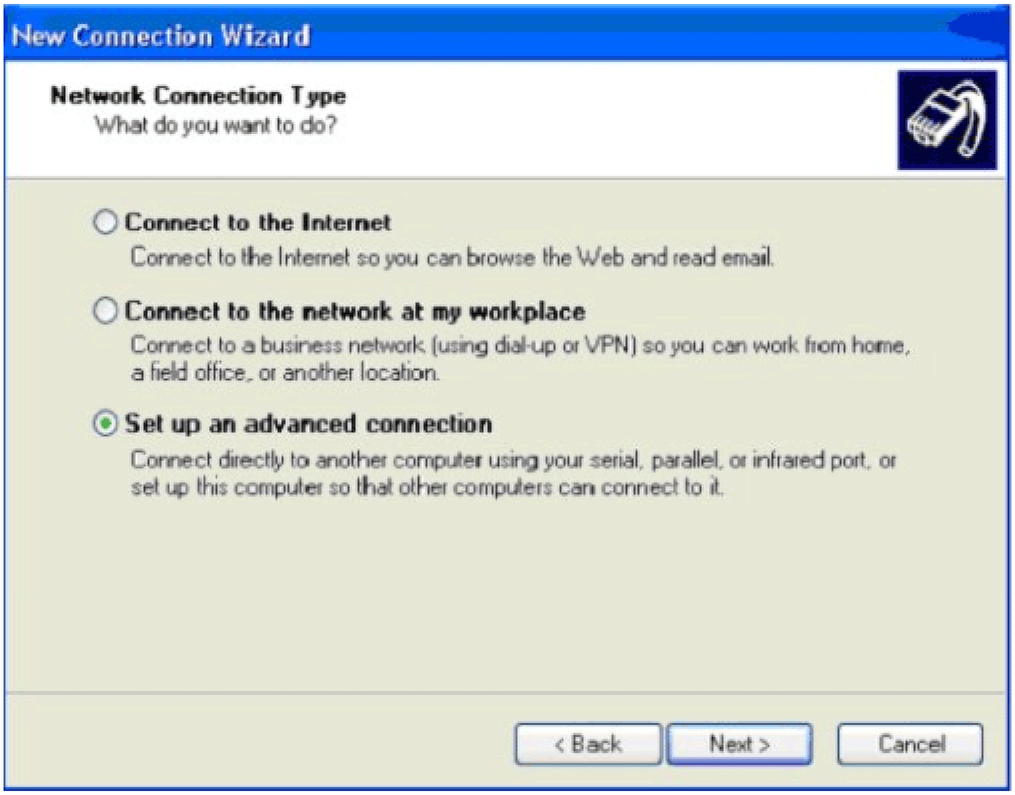
1. Select Control Panel, Network and Dial Up Connections and Create a New Connection. You will see a screen similar to the following.

Figure 8–29. New Connection Wizard



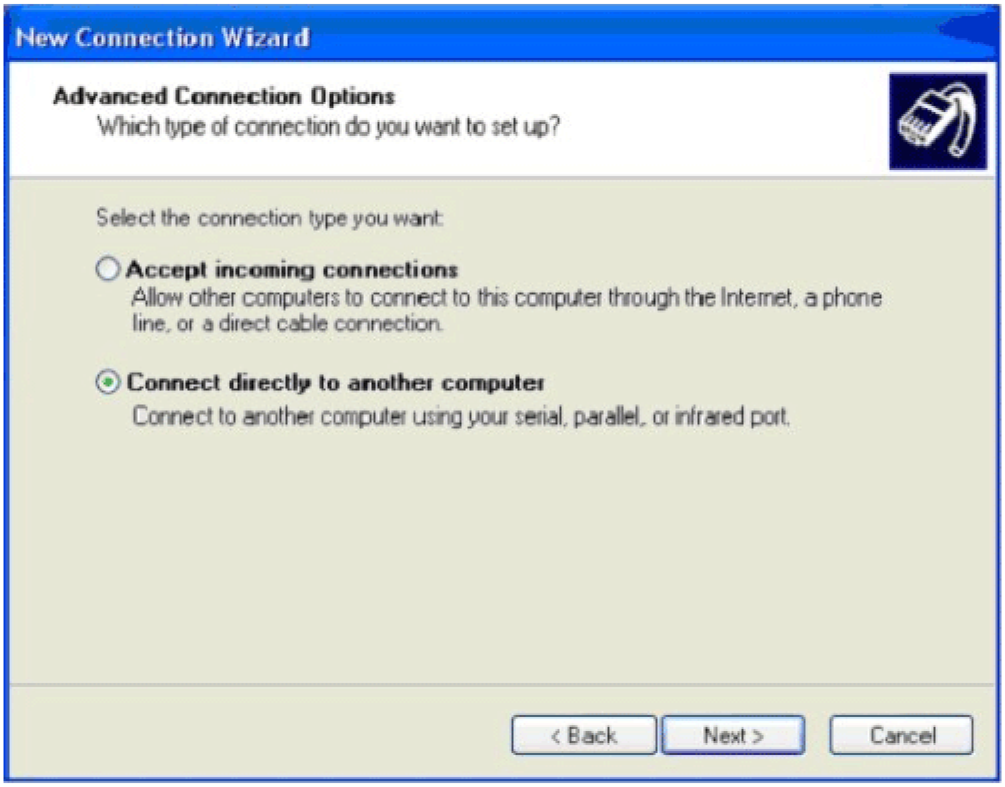
2. Click Next. You will see a screen similar to the following.

Figure 8–30. New Connection Wizard



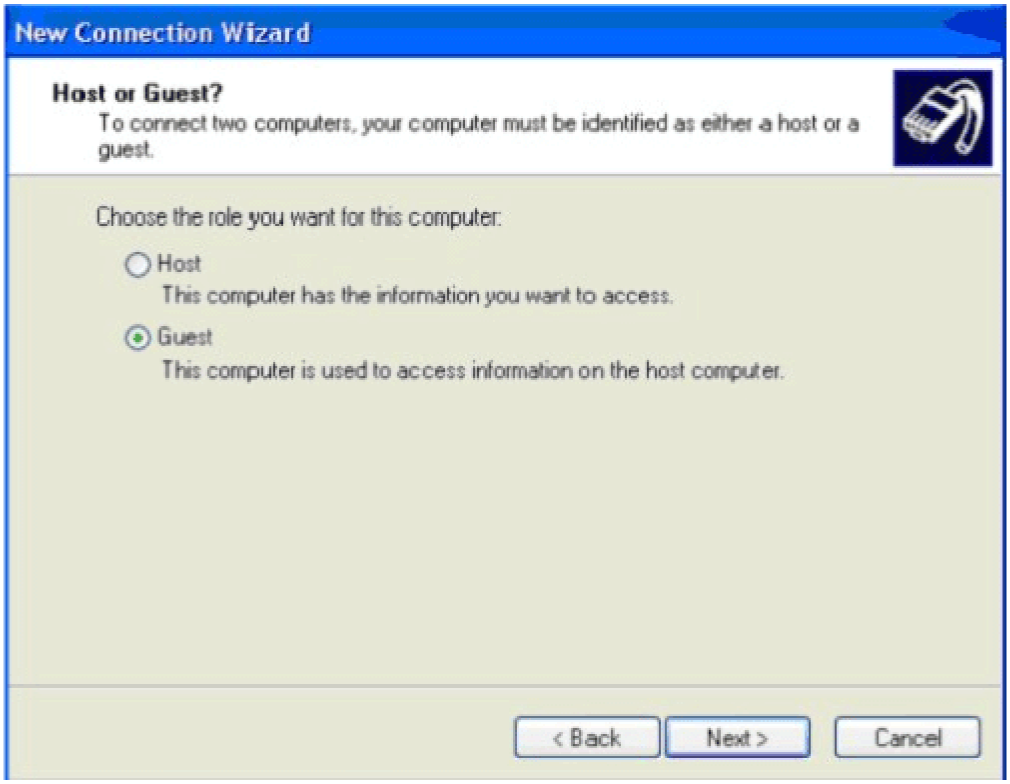
3. Select **Set up an advanced connection**, and click Next. You will see a screen similar to the following.

Figure 8–31. Advanced Connection Options



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

Figure 8–32. Host or Guest

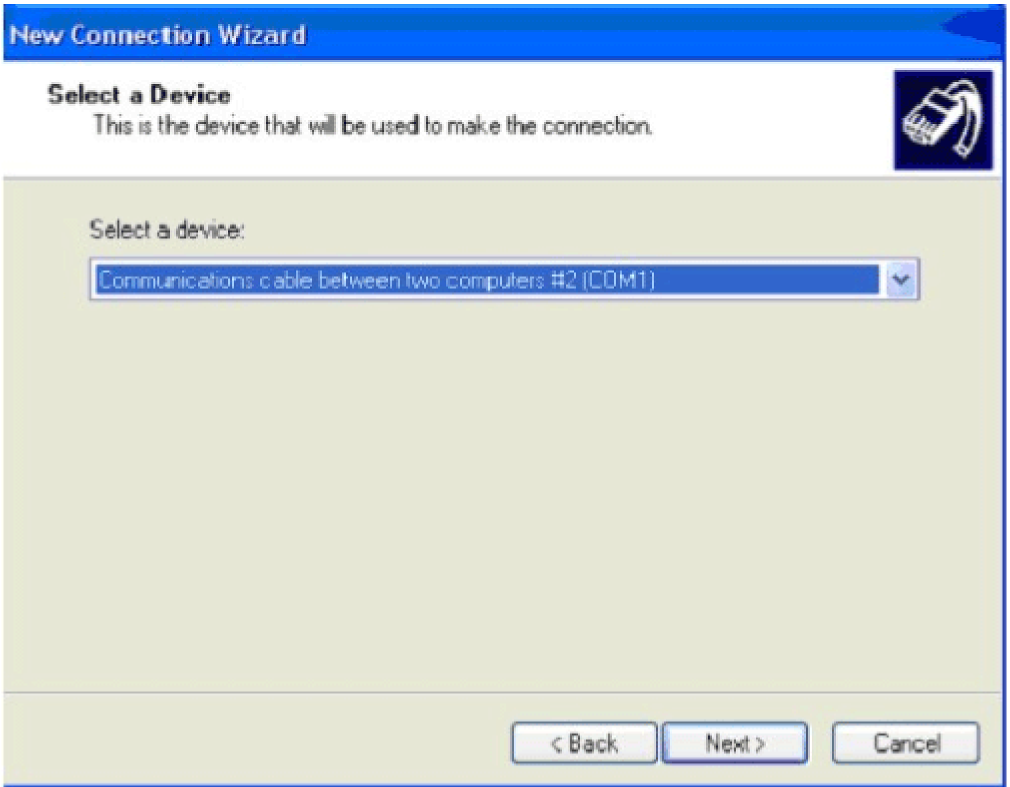


5. Select **Guest** and click Next. You will see a screen similar to the following.

Figure 8–33. Connection Name

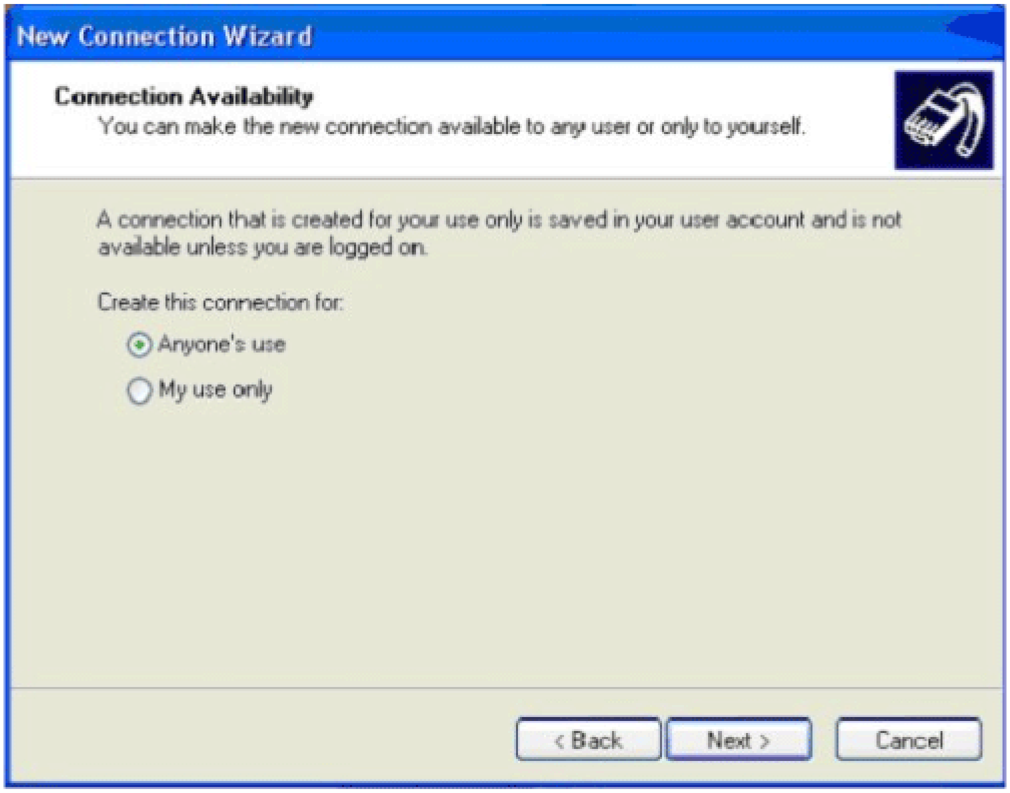
6. Type in Computer Name. For this example, RJ3-iC is used. Click Next and you will see a screen similar to the following.

Figure 8–34. Select a Device



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

Figure 8–35. Connection Availability

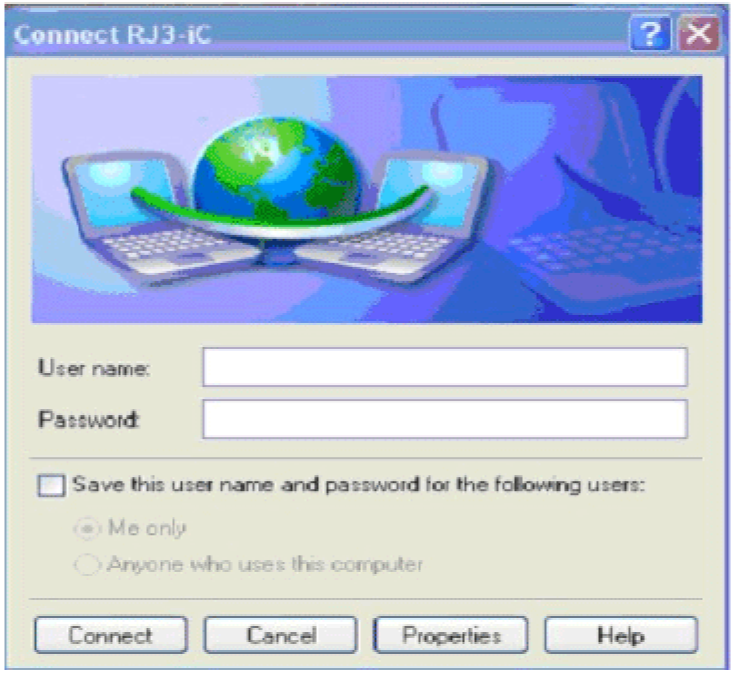


8. If you want all users who log on to your PC to use this connection, select **Anyone's use** , and click Next. If you want to be the only one that logs to your PC to use this connection, select **My use only**, and click Next. You will see a screen similar to the following.

Figure 8–36. Completing the New Connection Wizard

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

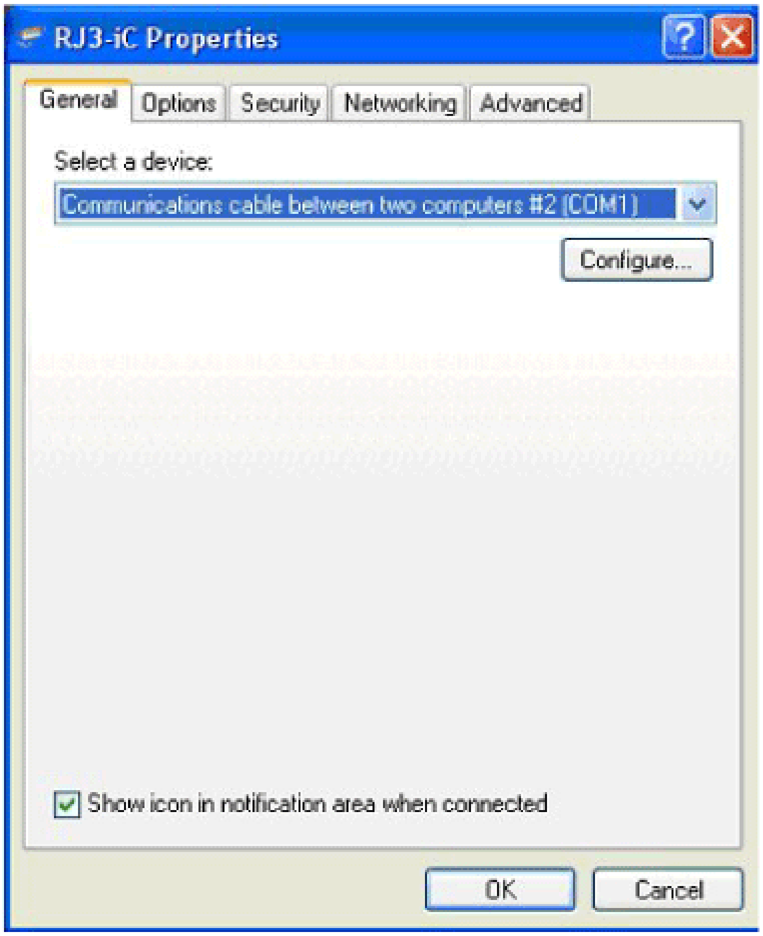
Figure 8–37. Connect Screen



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

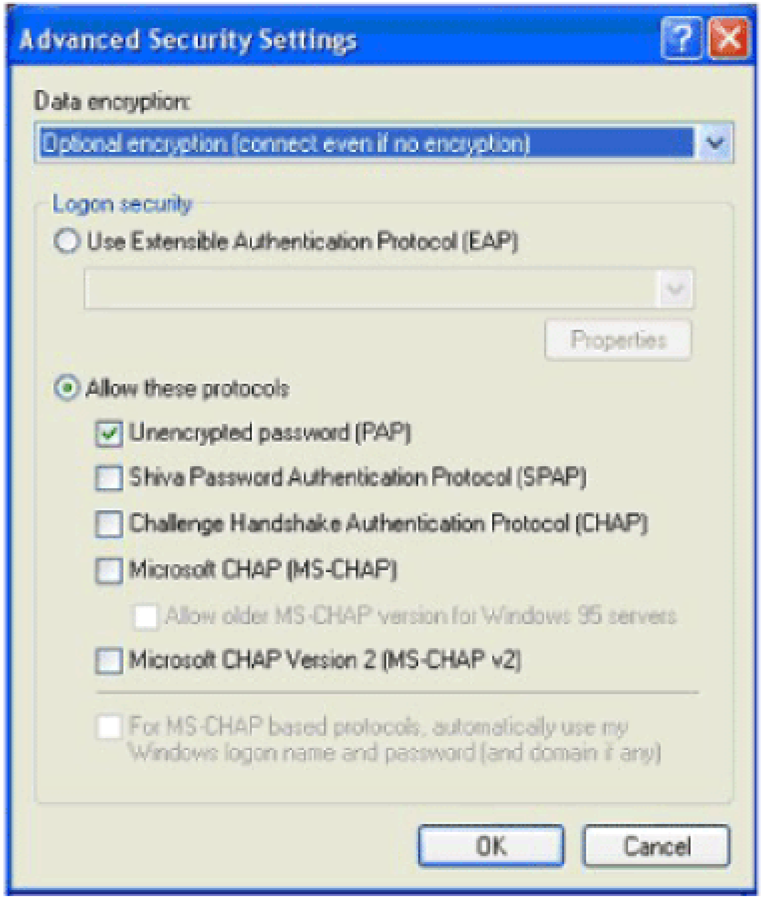
10. Click Properties. You will see a screen similar to the following.

Figure 8–38. Properties



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

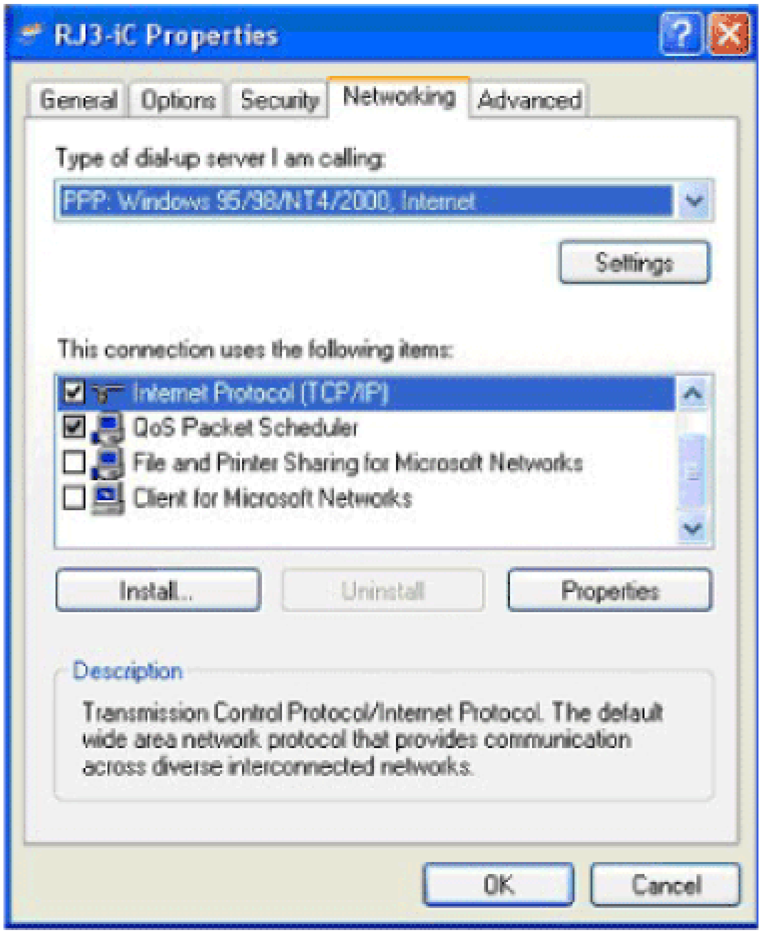
Figure 8–39. Advanced Security Settings



12. Select Optional Encryption and check the box for Unencrypted password (PAP). Uncheck all other boxes. Click OK.

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

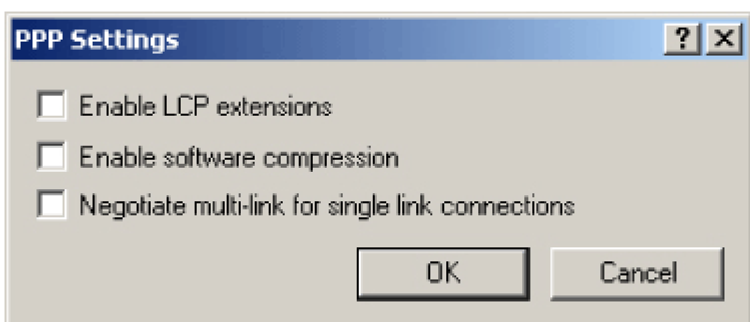
Figure 8–40. Properties



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 . You will see a screen similar to the following.

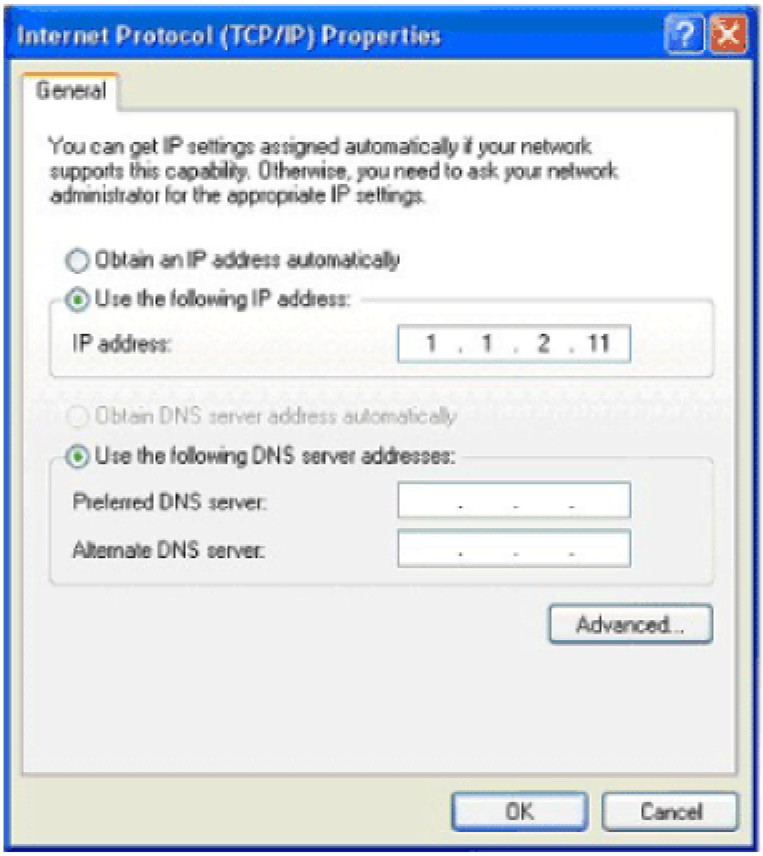
Figure 8–41. TCP/IP Settings



Make sure all the boxes are unchecked. Click OK.

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

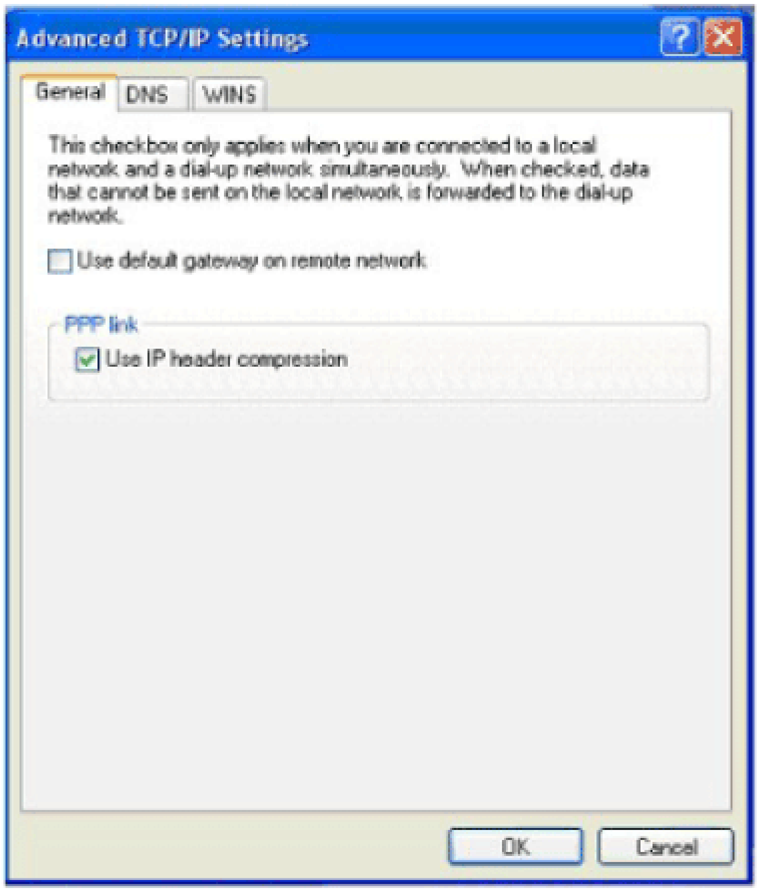
Figure 8–42. Internet Protocol (TCP/IP) Properties



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

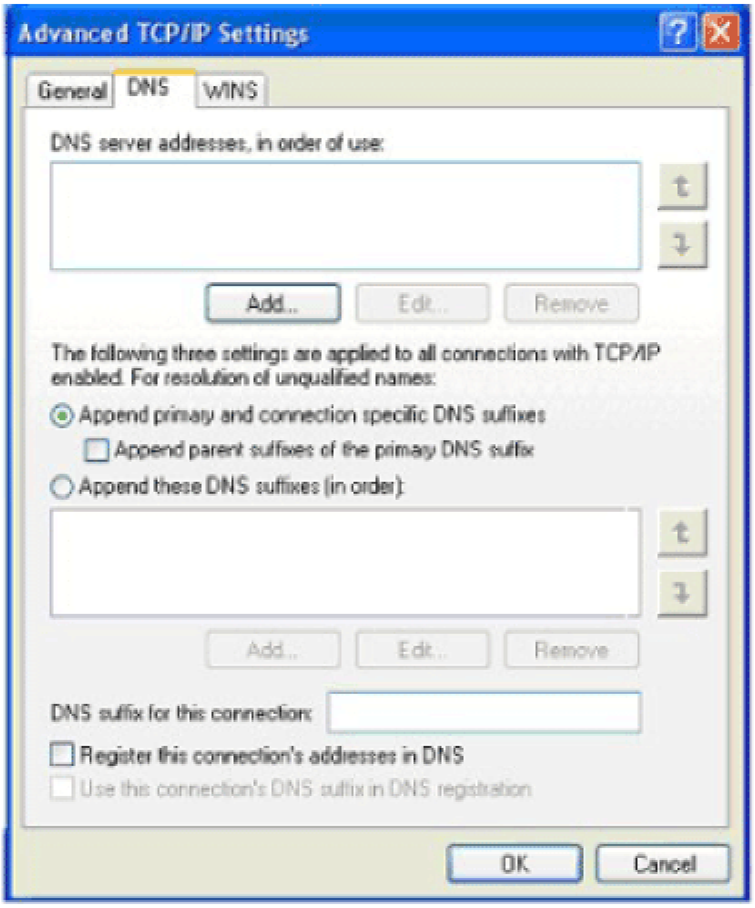
18. Click Advanced. You must make sure the Use IP header compression box is checked.

Figure 8–43. Advanced TCP/IP Settings



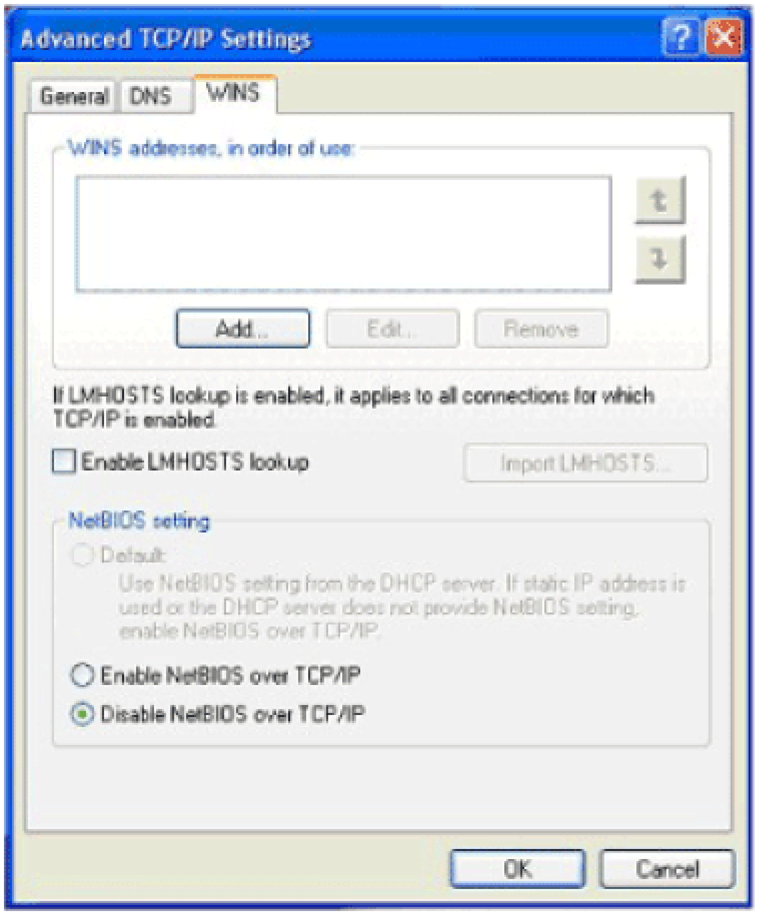
19. Select the DNS tab. You must make sure that the boxes are checked/unchecked as shown in Figure 8–44 .

Figure 8–44. Advanced TCP/IP Settings



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

Figure 8–45. Advanced TCP/IP Settings



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

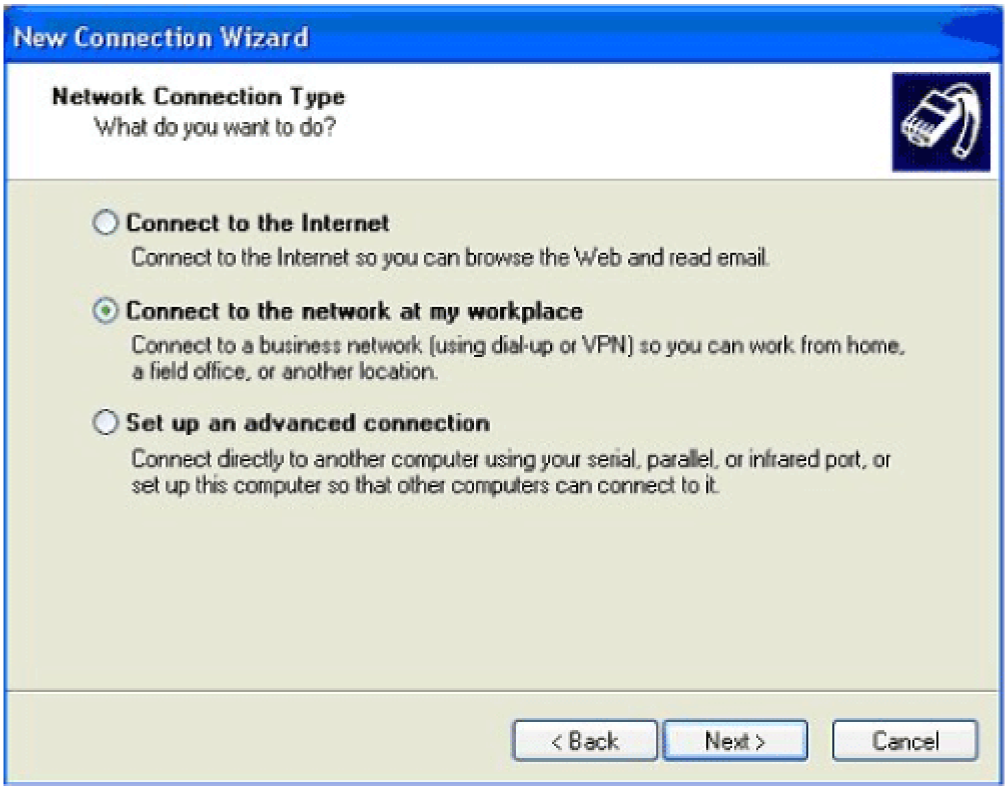
Procedure 8-7 Setting Up a PPP/Modem on a PC with Windows XP

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

Figure 8–46. New Connection Wizard

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

Figure 8–47. Network Connection



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

Figure 8–48. Dial Up Connection

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

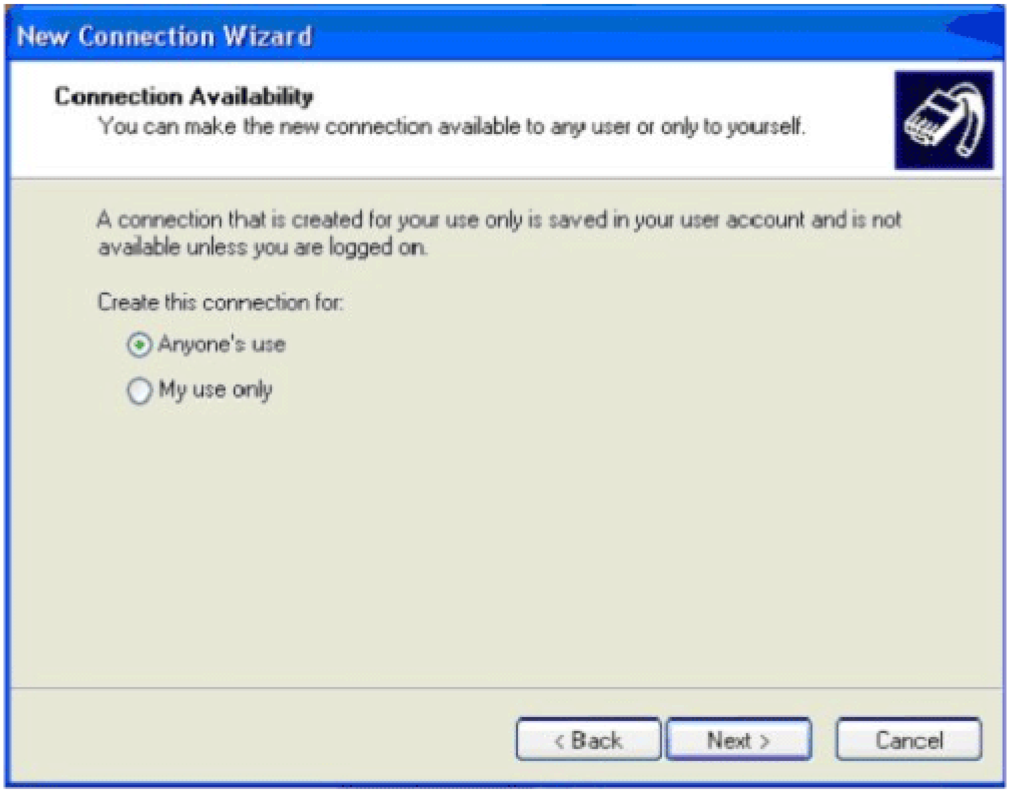
Figure 8–49. Connection Name

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

Figure 8–50. Phone Number to Dial

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

Figure 8–51. Connection Availability



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

Figure 8–52. Completing the New Connection Wizard

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

Chapter 9

DYNAMIC HOST CONFIGURATION PROTOCOL

Contents

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
- RFC1534 compliant, accepts and assigns infinite leases to BOOTP server responses
- Checks IP address first to see if it is in use before using it
- Can act like a PC based DHCP client for seamless integration of the robots into the existing network
- Is easy to set up

9.2 SETTING UP DHCP ON THE ROBOT

9.2.1 DHCP Setup

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

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

Table 9–1. 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. Advanced DHCP SETUP Screen Items

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

Procedure 9-1 Setting up DHCP on the Robot

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

```
SETUP Host Comm
TCP/IP
    Robot Name:                     ****
    Port#1 IP addr:       **************
    Subnet Mask:           255.255.255.0
    Board address:     08:00:19:02:68:22
    Router IP addr:       **************
Host Name (LOCAL)       Internet Address
  1 ***********        *****************
  2 ***********        *****************
  3 ***********        *****************
```

4. If you want to enable DHCP on Port #1, follow Step 4a . If you want to enable DHCP on Port#2, follow Step 4b .

   a. Press F2 (DHCP). You will see a screen similar to the following.

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

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

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

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

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

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

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

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

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

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

9.2.2 Advanced DHCP Setup

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

Note The Ethernet address of your robot can be viewed from the TCP/IP screens. Setting the client ID field sets the system variable $DHCP_CLNTID[Port#], where Port# is either 1 or 2.

The **set hostname in request** field allows the robot to function like Windows based DHCP clients (PC's) in sending out its hostname in the form of a DHCP request. Some servers are written explicitly to service Microsoft® clients only so this field allows the robot to function like PCs. To use the set hostname in request field, you have to make sure that the robot hostname field is set from the TCP/IP screens first. Setting this field sets $DHCP_CTRL[Port#].$SETHOST, where Port# is either 1 or 2.

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

The **retry rate on failure** field controls the rate (in minutes) at which retries occur if the robot does not get a response back from the server. The DHCP internally tries for a full minute to contact the server before giving up and reporting an error. This retry rate field determines when the next attempt to contact the server must be done. Setting this field sets the system variable $DHCP_CTRL[Port#].$RETRATE, where Port# is either 1 or 2. Set this field to zero to disable retries.

The **last valid IP address on failure** field is used in a case where the robot has a previously assigned IP address, and the lease is still valid on the IP address. When power is cycled on the robot, the robot on booting contacts the DHCP server to confirm the lease (this is standard DHCP behavior). If the DHCP server does not respond for some reason (network/ server is down or damage to cables), then this field determines if the robot will continue to use the IP address or not. If set to TRUE, then under these conditions, the robot will continue to use the IP address, but if set to FALSE, the robot will not use the IP address. Under no circumstances will the robot use the IP address beyond the lease expiration time, regardless of this setting. Setting this field sets the system variable $DHCP_CTRL[Port#].$USEIP, where Port# is either 1 or 2.

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

```
SETUP HostComm
TCP/IP - DHCP enabled
    Robot name:                     ROBOT
    Port#1 IP addr:        172.22.200.165
    Subnet Mask:            255.255.240.0
    Board address:      08:00:19:02:68:22
    Router IP addr:          172.22.192.1
Host Name (LOCAL)        Internet Address
  1 ***********      ********************
  2 ***********      ********************
  3 ***********      ********************
```

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

9.3 DHCP SYSTEM VARIABLES

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

$ENABLE: BOOLEAN: default FALSE

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

Powerup: The powerup takes effect immediately.

UIF Location: DHCP SETUP screen.

$IPUSE: BOOLEAN default TRUE

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

PowerUp: Cycle power to take effect.

UIF Location: DHCP advanced SETUP screen.

$RETRATE: INTEGER: default 10

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

PowerUp: The powerup takes effect immediately.

UIF Location: DHCP Advanced SETUP screen

$SETHOST: BOOLEAN: default FALSE

This variable sets the hostname field in the DHCP request sent to the server. Some servers require the hostname to be supplied in the hostname field in the request (especially servers serving Microsoft®

clients). In this case, you may need to set this field to TRUE. When this field is set to TRUE, the robot hostname ($HOSTNAME) is supplied as the hostname in the DHCP request.

Powerup: The powerup takes effect immediately.

UIF Location: DHCP Advanced SETUP screen.

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

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

$LEASESTRTIME, ULONG: default 0

This variable gives the time of start of the lease.

$LEASESTART: STRING

Time of start of lease in a readable format.

$LEASEENDTIME: ULONG: default 0

This variable gives the time when the lease will expire.

$LEASEEND: STRING

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

$IPADD: STRING

This variable indicates that the server returned the IP address.

$ROUTERIP: STRING

This variable indicates that the server returned router IP address.

$SNMASK: STRING

This variable indicates that the server returned subnet mask.

$STATUS: STRING

This variable indicates the status of the DHCP operation.

$DHCP_CLNTID: STRING: R/W

Client identifier passed by the robot to the server. This might not have to be supplied, depending on how the DHCP server is configured. You must see your network administrator for more details.

Typical use of the client identifier is either to supply an Ethernet address or to supply a string to the server. To use the Ethernet address, the 6 bytes must be separated by colons. Eg: 00:E0:E4:F7:94:DC

PowerUp: This variable takes effect immediately.

UIF Location: DHCP SETUP screen.

9.4 DHCP TROUBLESHOOTING

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

- **The DHCP operation failed with HOST-224 DHCP: No response from the server**

  You must make sure that the robot is connected to the network with a working Ethernet cable. You must contact your network administrator and make sure that the DHCP server is configured and running. The DHCP server must typically be located on the same network as the robot (otherwise, there must be a router on the network that functions as a DHCP relay agent and forwards requests and responses from one network to another). This problem could also happen if the network is having problems (such as heavy traffic). You can check this by looking at the Ethernet diagnostics by pressing DIAG key under the Host-Comm TCP/IP screen.

- **The DHCP operation failed with HOST-225: DHCP duplicate IP <x.x.x>**

  If this error occurs it means that the DHCP server served up an IP address that is already being used by another host on the network. You must inform your network administrator about this problem when it occurs.

- **Ethernet on robot stops working with HOST-226 and HOST-227 errors (lease time expired/shutting down Ethernet)**

  The robot could not renew the DHCP lease and the lease expired. This should not happen under normal circumstances. The robot might not be connected to the network or the network is having problems or the DHCP server might not be running any more.

Chapter 10

SOCKET MESSAGING

Contents

The User Socket Messaging Option gives you the benefit of using TCP/IP socket messaging from KAREL.

Socket Messaging enables data exchange between networked robots and a remote PC with LINUX, or a UNIX workstation. A typical application of Socket Messaging might be a robot running a KAREL program that sends process information to a monitoring program on the remote PC. The combination of PC-Interface option on the robot and PC-Developers Kit on the PC is recommended for data exchange between the robot and a Windows-based PC.

Socket Messaging uses the TCP/IP protocol to transfer raw data, or data that is in its original, unformatted form across the network. Commands and methods that Socket Messaging uses to transfer data are part of the TCP/IP protocol. Since Socket Messaging supports client and server tags, applications requiring timeouts, heartbeats, or data formatting commands can provide these additional semantics at both the client and server (application) sides of the socket messaging connection.

Refer to the *KAREL Reference Manual* for more information.

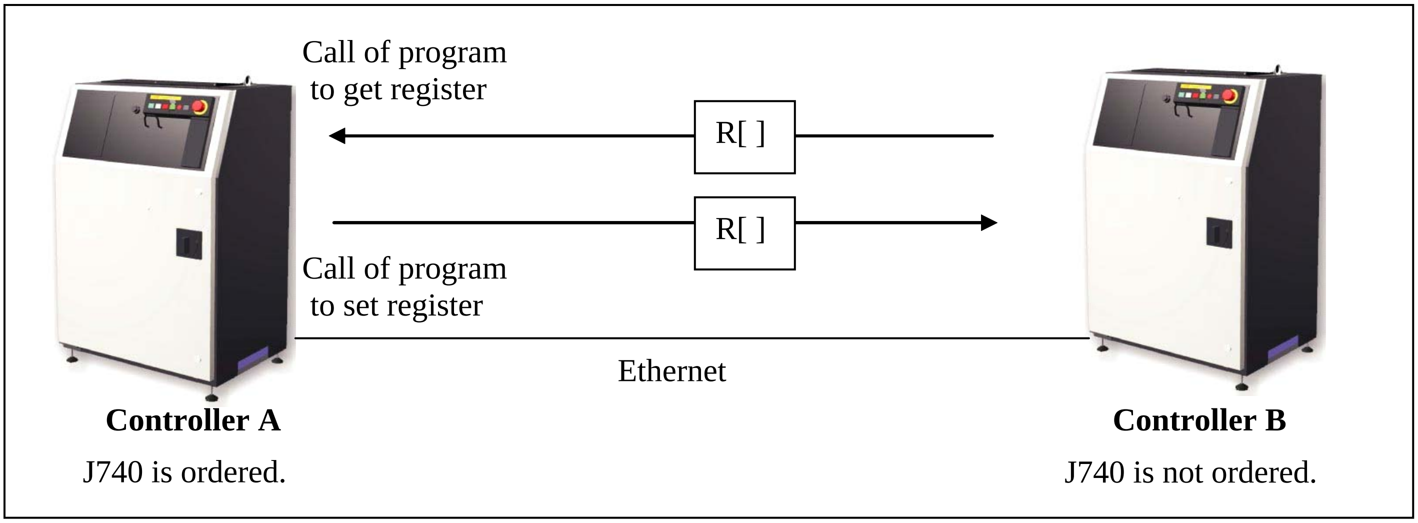
Chapter 11

DATA TRANSFER BETWEEN ROBOTS OVER ETHERNET

Contents

This section explains the optional (J740) data transfer between robots function available in V7.70 or later. This function enables you to transfer data between robots over Ethernet. By calling a KAREL program, you can transfer registers and position registers between robot controllers.

Figure 11–1. Data Transfer Between Robots over Ethernet



In Figure 11–1 , controller A runs the program to get the register from controller B. In addition, Controller A calls the program to transfer the register to controller B. In Figure 11–1 , only controller A starts data transfer. Controller B is just responding to request from controller A. In this case, only controller A needs this option.

Refer to the *KAREL Reference Manual* for more information.

Chapter 12

ETHERNET-BASED LOADING

Contents

12.1 OVERVIEW

The controller supports Ethernet functionality at the BMON mode.

The robot uses standard Internet protocols BOOTP/DHCP and TFTP to communicate with and transfer files from the server. This functionality requires a BOOTP or DHCP server and a TFTP server to be present and configured on the network. The configuration process involves setting up the Ethernet address and network parameters (like IP address, router address, and so forth) for every robot that wants to be loaded over Ethernet.

The server host can be any PC/UNIX system that supports the BOOTP or DHCP, and the TFTP protocols.

The BOOTP/DHCP protocols allow the server to return network information to the robot (like robot hostname, robot IP address, router information, and so forth). The TFTP protocol is used to transfer the files to and from the robot.

An example for configuring the Distinct BOOTP/TFTP servers (running on a PC) is given for reference in Appendix C .

12.2 IMAGE BACKUP AND RESTORE OF CONTROLLER OVER ETHERNET

This feature allows the user to backup or restore snapshot images of the robot. When backing up image files to a remote TFTP server, the robot controller also automatically performs a verification, or integrity check, to ensure that the image files were saved correctly by the TFTP server.

Requirements

- A BOOTP or DHCP server, and a TFTP server must be running and configured to service the robot on the network.
- The backed up images can be restored only on the same robot or on a robot with exactly the same memory configuration.

Procedure 12-1 Backing Up Controller as Images

Steps

1. To display the BMON prompt, press and hold F1 and F5 on the teach pendant while turning on the power disconnect circuit breaker. You will see a screen similar to the following.

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

2. Select “Controller backup/restore” and press ENTER. You will see a screen similar to the following.

```
*********Backup/Restore menu*********
0. Return to Main menu
1. Emergency backup
2. Backup controller as images
3. Restore controller images
4. Bootstrap to CFG menu
Select:
```

3. Select “Backup controller as images” and press ENTER. You will see a screen similar to the following.

```
********Device selection menu***********
1. Memory card (MC:)
2. Ethernet  (TFTP:)
Select:
```

4. To perform a backup over Ethernet, select Ethernet (TFTP:).

```
**** Backup Controller as Images ****
Module size to backup:

 FROM:  XXMb   SRAM:  XXMb

Disk space required on remote host is
 XXMb free space.

 Are you ready ? [Y=1/N=else] :
```

5. Select 1 to start backing up images to the server. The robot controller will attempt to get an IP address from a BOOTP or a DHCP server. A BOOTP request will be first sent out of Ethernet port 1. If no reply is received in 4 seconds, a DHCP request will be sent out of Ethernet port 1. If no reply is received in 4 seconds, a BOOTP request will be sent out of Ethernet port 2. If no reply is received in 4 seconds, a DHCP request will be sent out of Ethernet port 2. If no BOOTP or DHCP server exists on the network, you will see a screen similar to the following:

```
Initializing ethernet.  Please wait...
Trying BOOTP on Port 1....
Trying DHCP on Port 1....
Trying BOOTP on Port 2....
Trying DHCP on Port 2....
Failed to get an IP address!!!
Press enter to continue>
```

When a BOOTP or DHCP server response is received, the details of the response will be displayed, and the user will be allowed to accept or reject the parameters, for example:

```
Trying BOOTP on Port 1 ,.
Hostname   : robot49
IP Address : 192.168.1.49
Subnet Mask: 255.255.255.0
Router     : 192.168.1.1
TFTP Path  : software/v750p01/
Accept? [1=Yes][Other=No] >
```

6. Once boot paramters are accepted, the progress of the backup can be monitored.

```
Writing TFTP:\FROM00.IMG  (1/35)
Writing TFTP:\FROM01.IMG  (2/35)
          .
          .
          .
          .
Writing TFTP:\FROM31.IMG  (32/35)
Writing TFTP:\SRAM00.IMG  (33/35)
Writing TFTP:\SRAM01.IMG  (34/35)
Writing TFTP:\SRAM02.IMG  (35/35)
Integrity check on server succeeded
Done!!
Press ENTER to return >
```

Note The number of image files might vary, depending on the memory configuration of the controller. The above examples are for a 32MB FROM/3MB SRAM system. Backing up and verifying this system configuration takes about three and a half minutes over Ethernet.

Procedure 12-2 Restoring Controller as Images

Steps

1. To display the BMON prompt, press and hold F1 and F5 on the teach pendant while turning on the power disconnect circuit breaker. You will see a screen similar to the following.

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

2. Select “Controller backup/restore” and press ENTER. You will see a screen similar to the following.

```
*********Backup/Restore menu*********
0. Return to Main menu
1. Emergency backup
2. Backup controller as images
3. Restore controller images
4. Bootstrap to CFG menu
Select:-
```

3. Select “Restore controller as images” and press ENTER. You will see a screen similar to the following.

```
********Device selection menu***********
1. Memory card (MC:)
2. Ethernet  (TFTP:)
Select:
```

4. To perform a restore over Ethernet, select Ethernet (TFTP).

Note The BOOTP/DHCP server on the network must be configured correctly and the previously backed up images must be available on the TFTP server.

You will see a message similar to the following.

```
***** Restore Controller Images *****
Current module size:

 FROM:  XXMb   SRAM:  XXMb

CAUTION:  You SHOULD have image files
  from the same size of FROM/SRAM.
  If you don't, this operation causes
  fatal damage to the controller.
Are you ready?  [Y=1/N=else]
```

5. Select 1 to start the restore process. The robot controller will attempt to get an IP address from a BOOTP or a DHCP server. A BOOTP request will be first sent out of Ethernet port 1. If no reply is received in 4 seconds, a DHCP request will be sent out of Ethernet port 1. If no reply is received in 4 seconds, a BOOTP request will be sent out of Ethernet port 2. If no reply is received in 4 seconds, a DHCP request will be sent out of Ethernet port 2. If no BOOTP or DHCP server exists on the network, you will see a screen similar to the following:

```
Initializing ethernet.  Please wait...
Trying BOOTP on Port 1....
Trying DHCP on Port 1....
Trying BOOTP on Port 2....
Trying DHCP on Port 2....
Failed to get an IP address!!!
Press enter to continue>
```

When a BOOTP or DHCP server response is received, the details of the response will be displayed, and the user will be allowed to accept or reject the parameters, for example:

```
Trying BOOTP on Port 1 ,.
Hostname   : robot49
IP Address : 192.168.1.49
Subnet Mask: 255.255.255.0
Router     : 192.168.1.1
TFTP Path  : software/v750p01/
Accept? [1=Yes][Other=No] >
```

6. Once boot paramters are accepted, the progress of the backup can be monitored.

```
***************************************
CAUTION:  NEVER TURN OFF THE POWER
SUPPLY WHILE CLEARING FROM.
***************************************
Reading TFTP:\FROM00.IMG ... Done
Reading TFTP:\FROM01.IMG ... Done
          .
          .
          .
          .
Reading TFTP:\FROM31.IMG ... Done
Clearing SRAM (3M) ... done
Reading TFTP:\SRAM00.IMG ... Done
Reading TFTP:\SRAM01.IMG ... Done
Reading TFTP:\SRAM02.IMG ... Done
 -- Restore complete --

Press ENTER to return >
```

7. Turn the controller off and back on to complete the restore process.

12.3 LOADING SOFTWARE OPTIONS OVER ETHERNET

The controller supports loading software options, updates, and customizations over Ethernet. The load media must be available on the server and there must be BOOTP or DHCP, and TFTP server software configured and running on the server.

Procedure 12-3 tells how to load software options over Ethernet.

Procedure 12-3 Loading Software Options Over Ethernet

Conditions

- The controller is plugged in and is working properly.
- You are trying to do one of the following:
  - Authorize an unauthorized option
  - Install an authorized option
  - Set up an installed option
  - Reinstall an option
  - Authorize, install, and set up an option purchased separately from the application software
- Enough system memory is available to install all application software. Refer to the ssd.html file on the memory card for information on the amount of memory available for your application, and required by each option.

  Note If you try to install an option that is too large for the memory on your controller, you will see a screen similar to the following.

Figure 12–1. Memory Screen

```
More memory is needed than is available.
AVAILABLE:   REQUIRED:
Main SYSTEM :         0 Main
TEMP    :         0        74750
Comm SYSTEM :         0            0
Comm TEMP   : 15305678            0
Main TPP    :   864423         1860
Main PERM   :  1047667            0
Main FR:    :  7846912        57344
Use PREV to exit
```

Contact FANUC America Corporation to reconfigure your system with the help of a FANUC America Corporation representative.

- The teach pendant ON/OFF switch is OFF and the DEADMAN switch is released.
- The REMOTE/LOCAL setup item in the System Configuration Menu is set to LOCAL. Refer to the *application-specific Setup and Operations Manual for more information*.
- **If you are updating controller software**, you have backed up everything that you want to use on this controller.

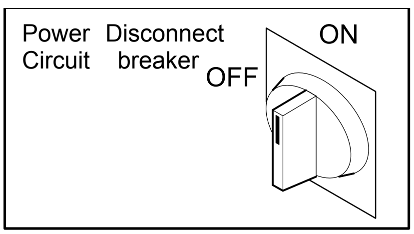
Steps

1. Perform a Controlled start. If the controller is already in Controlled Start mode, go to Step 2

a. If the controller is turned on, turn off the power disconnect circuit breaker.

b.

c. Press and hold the PREV and NEXT keys on the teach pendant. While you hold these keys, turn on the power disconnect circuit breaker.



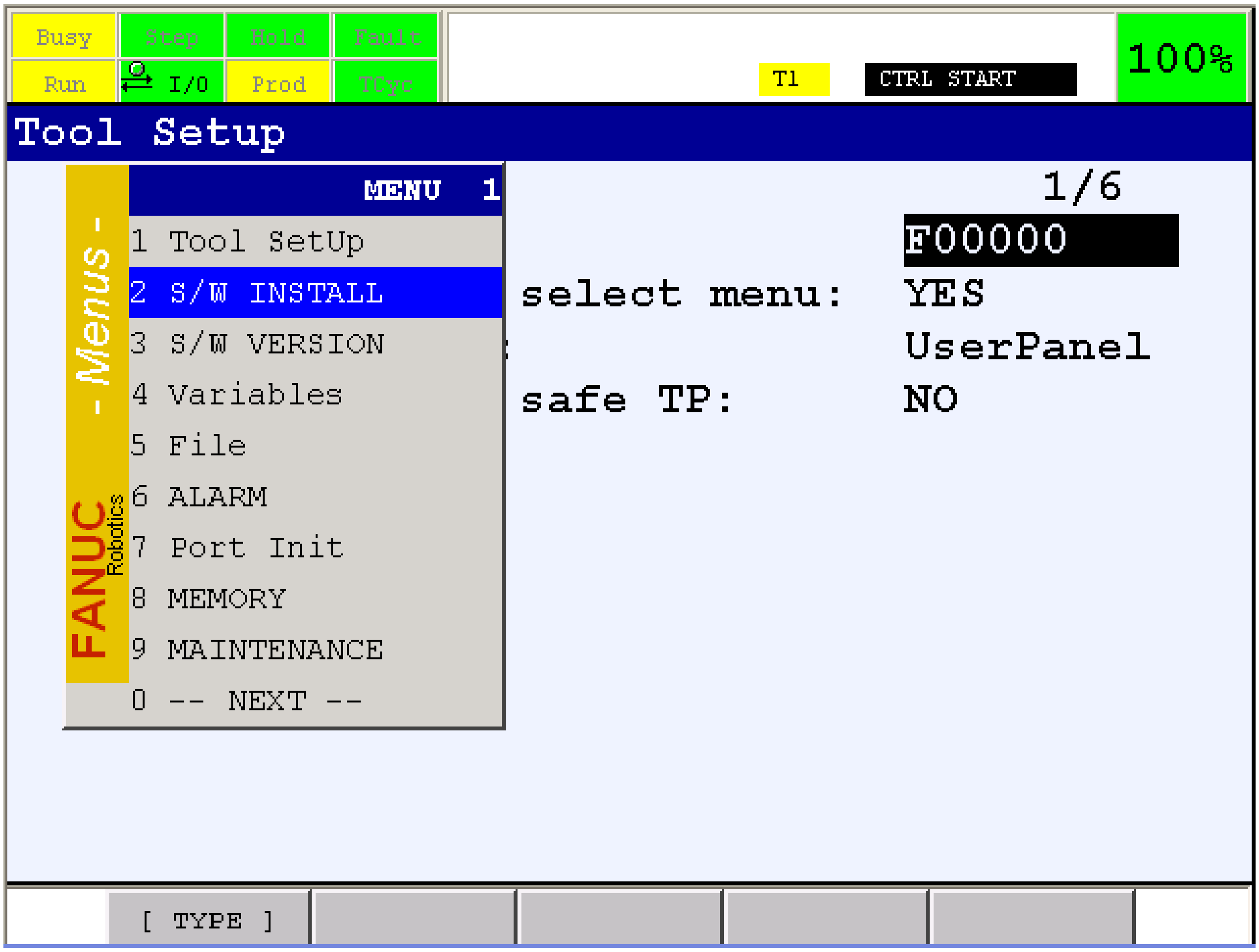
You will see a screen similar to the following.

```
---------- Configuration Menu ----------
1 Hot start
2 Cold start
3 Controlled start
4 Maintenance
Select >
```

d. Select Controlled start and press ENTER. After a few moments, you will see the Controlled start screen for your application.

2. Press MENU. You will see a screen similar to the following.

Figure 12–2. Tool Setup Screen



3. Select S/W INSTALL. You will see a screen similar to the following.

```
S/W INSTALL           CONTROLLED START MENUS
Application : ProductName
      1   Controller ID
      2   Robot Library
      3   Option
      4   Update
      5   Customization
      6   FTP Setup
```

4. Move the cursor to Option and press ENTER. You will see a screen similar to the following.

```
Load Media
Please choose device to load options.
Press F3 (DONE) when done.
Current Load Media:  Ethernet (TFTP)
```

5. Press F4, [CHOICE], and select “Ethernet (TFTP)” from the list of available devices.
6. When you have finished, press F3, [DONE].

Note The following screen is an example. Your option screen will be different depending on the options that are available with your software. The four digits preceding the option name uniquely identify each option.

```
OPTION                   CONTROLLED START MENUS
J631: AccuPath              Authorized
ACAL: AccuCal               Installed
FIND: Auto Normal Utility   Installed
ATCP: Auto TCP              Authorized
J618: CE Mark               Unauthorized
J535: CRT/Keyboard Manager  Authorized
J613: Continuous Turn       Authorized
CNRE: Control Reliable      Unauthorized
CNET: ControlNet Int        Installed
J619: Coordinated Motion    Authorized
```

7. **To change the display from TITLE to ORDER NUMBER** , move the cursor to the item you want to change and press F3, ORD NO. You will see a screen similar to the following.

```
OPTION                   CONTROLLED START MENUS
1   A05B-2400-J631        AUTHORIZED
2   A05B-2400-ACAL        INSTALLED
3   A05B-2400-FIND        INSTALLED
```

8. **To change the display from ORDER NUMBER to TITLE** , press F3, TITLE.

   - **If the option status is Authorized** , it is not a requirement for your application, but it has been preauthorized for you. In this case, it has not yet been installed and you can decide whether or not to install it. The option will have a PAC code already assigned (where a number >1 is valid, and -1 is invalid). If you want to install this option, and you know the PAC code, go to Step 3, Install, in Table 12–1 , to install this option.

Note If this option status is Installed , was purchased with the application software, and is a requirement for your application, the PAC code will already be available and the option will be authorized, installed, and set up. You do not need to do anything more to install or set up this option.

Note If this option status is Unauthorized or empty , it has not yet been authorized for installation. To authorize this option before you install it, refer to Table 12–1 and go to Step 2, Authorize, to authorize this option.

9. To authorize, install, and set up additional options that have been purchased separately from the application software, refer to Table 12–1 .

Table 12–1. Authorizing and Installing Options

Step Order	To do this	You must do this
1.	Obtain the PAC	Obtain the Product Authorization Codes (PAC) for each option you have purchased and want to install from FANUC.
2.	Authorize	1. Press F4, AUTH. 2. Type the PAC for the option and press ENTER. 3. PressF4, AUTH. The option status will be Authorized if you have entered a valid PAC. 4. Go to Step 3, Install, to install the option.
3.	Install	1. Press F2, INSTALL. The software will be installed. **NOTE** The requirements for the option will be evaluated. If requirements have not been met, INSTALL will fail.

Note If you are using PaintTool and have installed options after the initial software installation, you must display the PaintTool Setup screen and set up the PaintTool software. Then, select your Robot No., Applicator Type, Cell I/O Hardware, and Process I/O Hardware. After you have configured these items press F2, SETUP, to set up your PaintTool software properly.

12.4 INSTALLING UPDATES AND CUSTOMIZATIONS

Use Procedure 12-4 to install a software update or a customization.

Procedure 12-4 Installing Customizations

Conditions

- Media with a customizations on it.
- If installing over Ethernet, a BOOTP/DHCP and TFTP server must be configured and running on the network.

- You have the software load media containing the software you want to load, and the media is inserted properly.

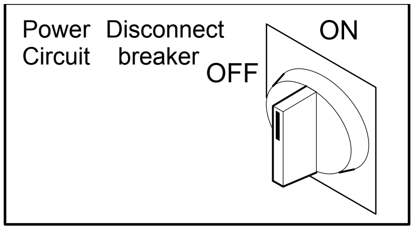
Steps

Note You can perform a customizations over Ethernet.

1. Perform a Controlled start. If the controller is already in Controlled Start mode, go to Step 2

   - If the controller is turned off, **immediately press and hold,** PREV and NEXT keys on the teach pendant.

Figure 12–3. Power Disconnect Breaker



When power is cycled, the second line on the pendant should display “Boot system activating...” in reverse video. You can then release the PREV and NEXT keys.

You will see a screen similar to the following.

```
---------- Configuration Menu ----------
1 Hot start
2 Cold start
3 Controlled start
4 Maintenance
Select >
```

Select Controlled start and press ENTER. After a few moments, you will see the Controlled start screen for your application.

   - If the controller is turned on but in Cold start mode, press FCTN and select CYCLE POWER. While the controller is starting up, **immediately press and hold,** PREV and NEXT keys on the teach pendant. When power is cycled, the second line on the pendant should display “Boot system activating...” in reverse video. You can then release the PREV and NEXT keys.

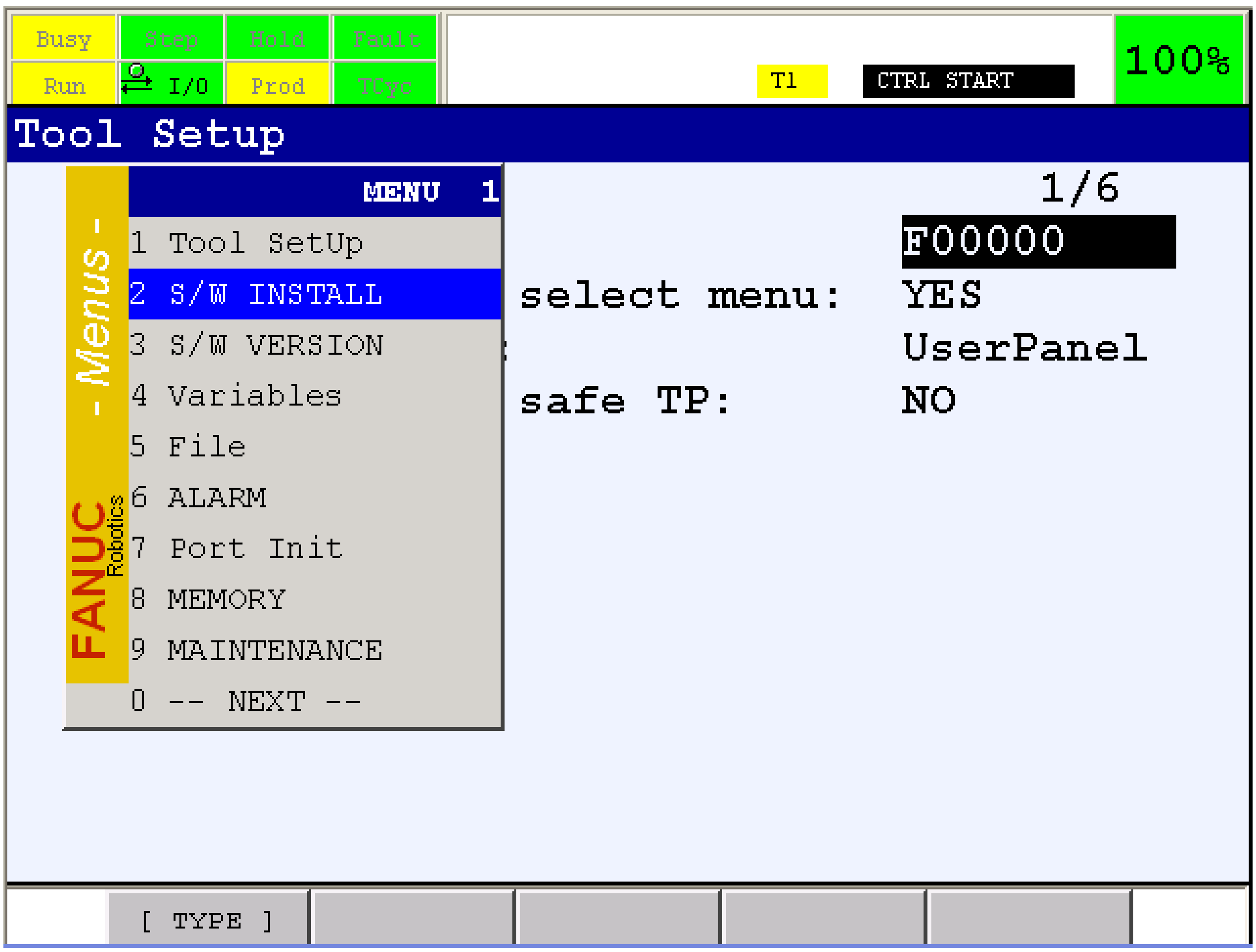
You will see a screen similar to the following.

```
---------- Configuration Menu ----------
1 Hot start
2 Cold start
3 Controlled start
4 Maintenance
Select >
```

Select Controlled start and press ENTER. After a few moments, you will see the Controlled start screen for your application.

2. Press MENU. You will see a screen similar to the following.

Figure 12–4. Tool Setup Screen



3. Select S/W INSTALL. You will see a screen similar to the following.

```
 S/W INSTALL        CONTROLLED START MENUS
Application: ProductName
1  Controller ID
2  Robot Library
3  Option
4  Customization
5  Language
```

4. **To install a customization**, move the cursor to Customization and press ENTER. You will see a screen similar to the following.

```
Load Media
Please choose device to load options.
Press F3 (DONE) when done.
Current Load Media: USB (UD1:)
```

5. To choose the device, press F4, [CHOICE], and select the item from the list of available devices. If you are loading over Ethernet, choose “ Ethernet (TFTP) from the list of available devices.
6. When you have finished, press F3, [DONE]. The files will be copied to the controller. When they are finished, the S/W Install screen will be displayed. You will see a screen similar to the following.

```
S/W INSTALL        CONTROLLED START MENUS
Application: ProductName
     1  Controller ID
     2  Robot Library
     3  Option
     4  Customization
     5  Language
```

Chapter 13

SIMPLE NETWORK TIME PROTOCOL (SNTP)

Contents

13.1 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.

13.2 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 13.3). Some of the fields can only be configured by setting the system variable, $SNTP_CFG. Refer to the application-specific *Setup and Operations Manual* for more information on setting system variables.

Note that the SNTP interface screen contains most of the fields in $SNTP_CFG. Unless you want to set the optional variable (i.e. $TIME_WIN) in $SNTP_CFG, you are encouraged to configure SNTP via SNTP user interface instead of setting system variables directly. Refer to Table 13–1 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 13–2 for detailed information on $SNTP_CUSTOM.

Table 13–1. $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 13–2. $SNTP_CUSTOM 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

Table 13–2. $SNTP_CUSTOM Settings (Cont'd)

System Variable	Default Value	Unit	Description
$SNTP_CUSTOM. $END_HOUR	2	Hour	Enter time (in hour) when DST ends*
$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.

13.3 USING SNTP

By default, SNTP client is disabled. In order to run SNTP client, use Procedure 13-1 .

Procedure 13-1 Running the SNTP Client

Conditions

- SNTP option is installed.

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

```
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. You will see a screen similar to the following.

```
Advanced Setting                   1/4


NTP enable:             FALSE
Daylight Saving Time:   TRUE

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

5. If NTP server field is not filled in, please contact your Information System (IS) department to get NTP server address. You can enter either the host name or IP address of NTP server. If the host name is used, ensure that DNS option is installed or the host name is entered in the host entry table. Refer to Chapter 2 *SETTING UP TCP/IP* for more details on 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 13–3 for timezone and current DST policies. If you select CUSTOM, set the $SNTP_CUSTOM system variable, before you enable SNTP (Refer to Table 13–2 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 13–3. 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
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

Table 13–3. Timezone and Current DST Policies (Cont'd)

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

13.4 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.

Chapter 14

SIMPLE MAIL TRANSFER PROTOCOL

Contents

14.1 OVERVIEW

Simple Mail Transfer Protocol (SMTP) is the protocol used on the Internet to exchange electronic mail, or email. Email is a convenient way to exchange information over networks, and the Internet. Most firewalls allow email to pass through. Services also exist that convert email messages to text messages (for cell phone support), providing a simple mechanism to alert a technician when certain triggers or events occur on a robot. Files can be attached to email messages, providing service personnel with instant data access.

Two SMTP options exist:

- A basic SMTP option (SMTP), which is included on every controller.
- An advanced option (EMLP), which is loaded when either of the following controller options is loaded, Internet Protocol Connectivity and Configuration(IPCC), or DRC *i*Pendant edition (IDRC).

The basic option is loaded on all robots and provides a simple mechanism for sending predefined emails from an *i*Pendant or remote *i*Pendant. This feature provides a simple interface and set of basic email templates for you to send data to the FANUC America Corporation Hotline, or to a distributor for diagnostics and support.

The enhanced email option adds the ability for you to create and modify email templates. This option also gives you more options for accessing the SMTP functionality on the robot. You can not only send email from the *i*Pendant and remote *i*Pendant, but also from KAREL, and teach pendant programs.

If a problem occurs with your robot system, robot email can assist you in sending all required information quickly and easily to the FANUC America Corporation Hotline. By sending email, the FRA representative can quickly understand the software loaded, options, and alarm history and can therefore provide quicker assistance by having a better understanding of the situation. There are multiple email forms that are preset for you to select from, but your FRA representative might ask for a specific form based on the situation. The FRA email address is pre-populated in the form, and you can add your own email address in the CC part if you would like a copy of the information being sent.

Refer to Section 14.5.2 for setup information.

Note To use SMTP to send email, a properly configured SMTP server must be accessible by the robot over an Ethernet network.

14.2 SETTING THE ROBOT F NUMBER

During the setup/install of the robot you must set the Robot F Number the robot for remote connection and for robot email. The F Number is a primary piece of information needed for any service or support call, by having the robot properly identify itself via Robot Email, Diagnostic Files, and remote connections.

You can use either Procedure 14-1 or perform a Controlled Start to set the Robot F Number.

Refer to the application-specific *Setup and Operations Manual,* “System Operations” Appendix for information on performing a Controlled Start.

Procedure 14-1 Setting Up the Robot F Number System Variable

1. Press MENU.
2. Select SYSTEM.
3. Select Variables.
4. Move the cursor to $FNO, and press ENTER.
5. Type the Robot F Number, and press ENTER.

14.3 SETTING UP SMTP

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

- Define TCP/IP parameters
- Configure and set up SMTP on the controller

Refer to the *Software Installation Manual* for information on installing host communications options such as the SMTP Interface software.

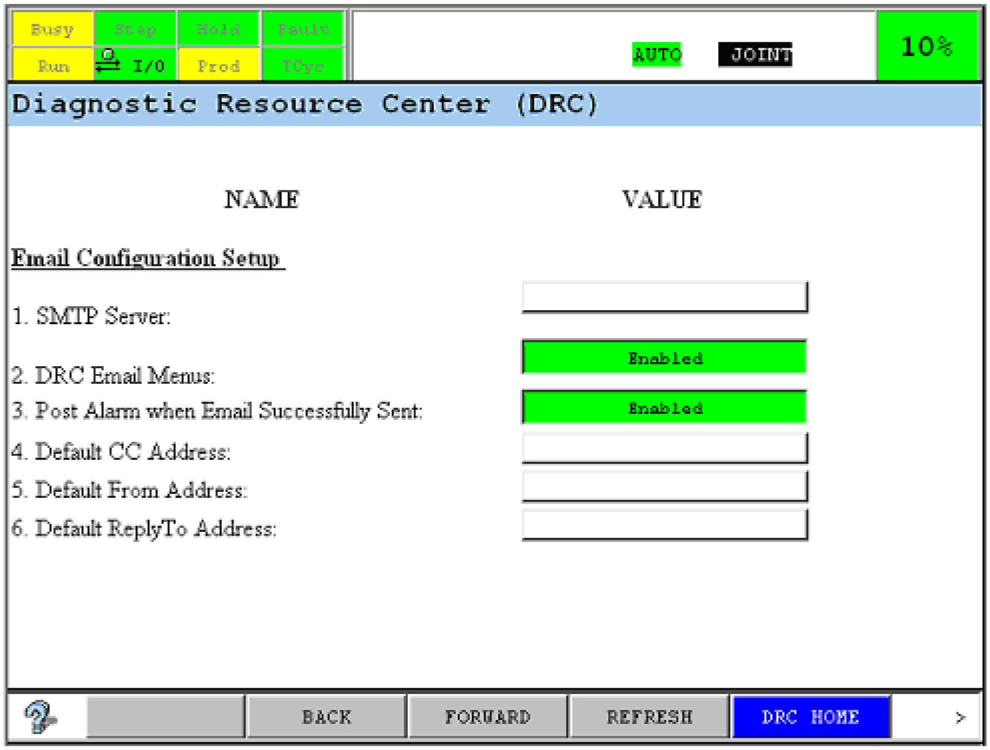
SMTP setup is done through the DRC screens. To display the main DRC screen, press SHIFT and DISP. Select Help/Diagnostics. Select Diagnostics Home. At the main DRC screen, as seen in Figure 14–1 , select Data Services Setup. A Remote *i*Pendant can also be used through a web browser to configure and access the SMTP screens. To launch a remote *i*Pendant, a link entitled “Navigate iPendant (CGTP)” located at the bottom of the default robot web page can be used. Refer to Chapter 18 *ADVANCED iPENDANT FUNCTIONS* for more information.

Figure 14–1. DRC Startup Screen

In the configuration screen, under Email Configuration Setup, type in the address of the SMTP server. If you would like to send a copy of every email message sent by the controller, insert a Default CC Address. A default From Address and a default ReplyTo Address can also be configured here, which might be needed to prevent email messages from being filtered through firewalls.

Due to the potential length of time that an SMTP transaction between the SMTP server and the controller can take, you cannot be immediately notified whether an email message was successfully sent or not. Instead, you will be notified if the message was placed in the email send queue. However, if “Post Alarm when Email Successfully Sent” is enabled, an alarm of WARN severity will be posted when an email message is successfully sent to the SMTP server. An alarm of WARN severity will always be posted on unsuccessful delivery of email messages to the SMTP server. See Figure 14–2

Figure 14–2. Configuration Screen



14.4 BACKING UP SMTP

The SMTP system variable structure is saved in syshost.sv and can be copied to a media and moved between robots.

SMTP Templates are saved in email.dg and can also be copied to a media and moved between robots.

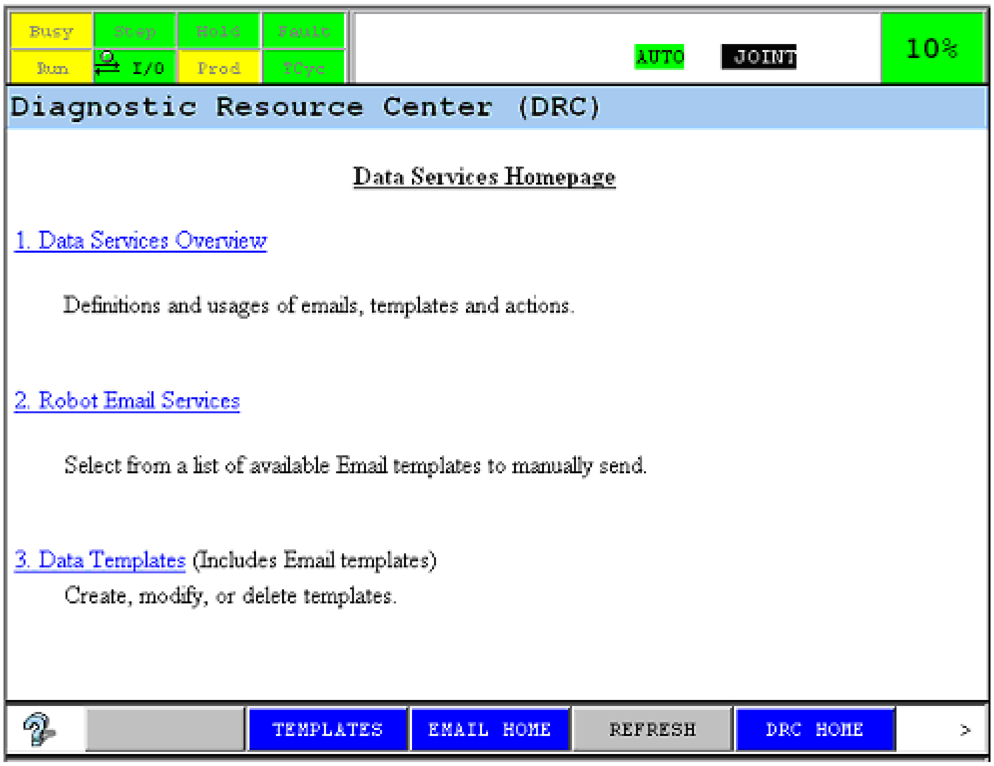
14.5 DATA SERVICES — CREATING AND SENDING EMAIL

14.5.1 Robot Email Services

From the main DRC homepage, select Data Services. On the Data Services Homepage there are three options. See Figure 14–3 . The Data Services Overview provides definitions of the email functions. The Robot Email Services allows you to send email from the *i*Pendant. The Data Templates option allows you to create, modify, or delete templates.

Note The advanced email option must be loaded on the robot to create, modify, or delete templates.

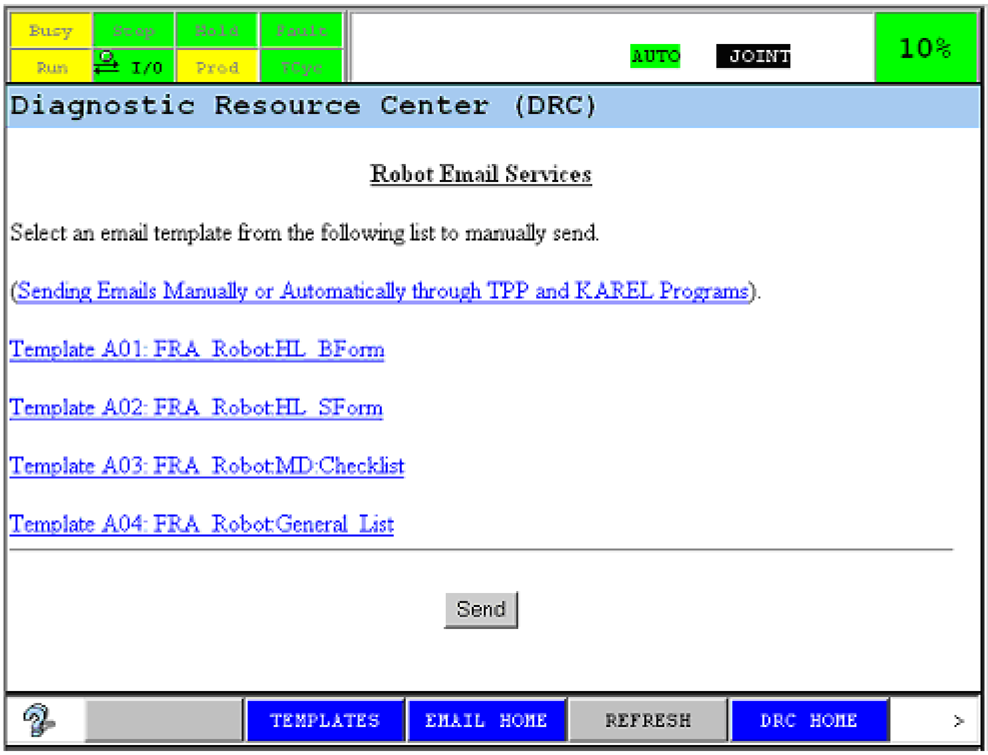
Figure 14–3. Data Services Homepage



14.5.1.1 Robot Email Services

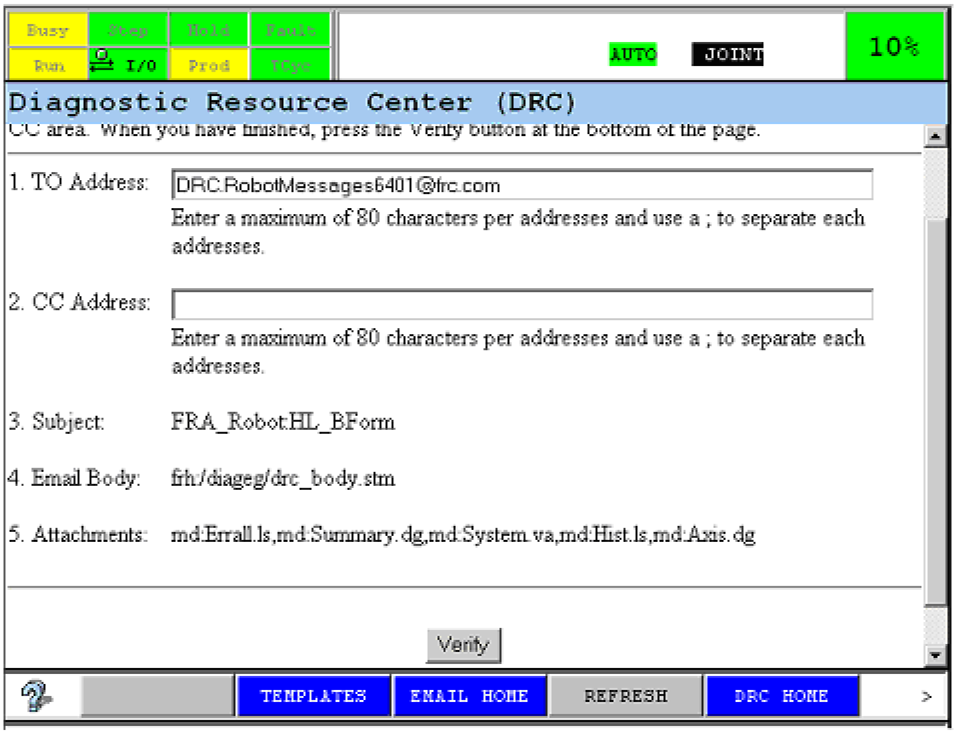
Email Templates are used to pre-define the contents of an email. Email Templates contain information such as the message or body of the email, the subject line, to address(es), copy address(es), and so forth. A number of pre-defined, un-modifiable templates are automatically installed on the robot. These templates can be viewed through the Robot Email Services screen as shown in Figure 14–4 .

Figure 14–4. Robot Email Services Screen



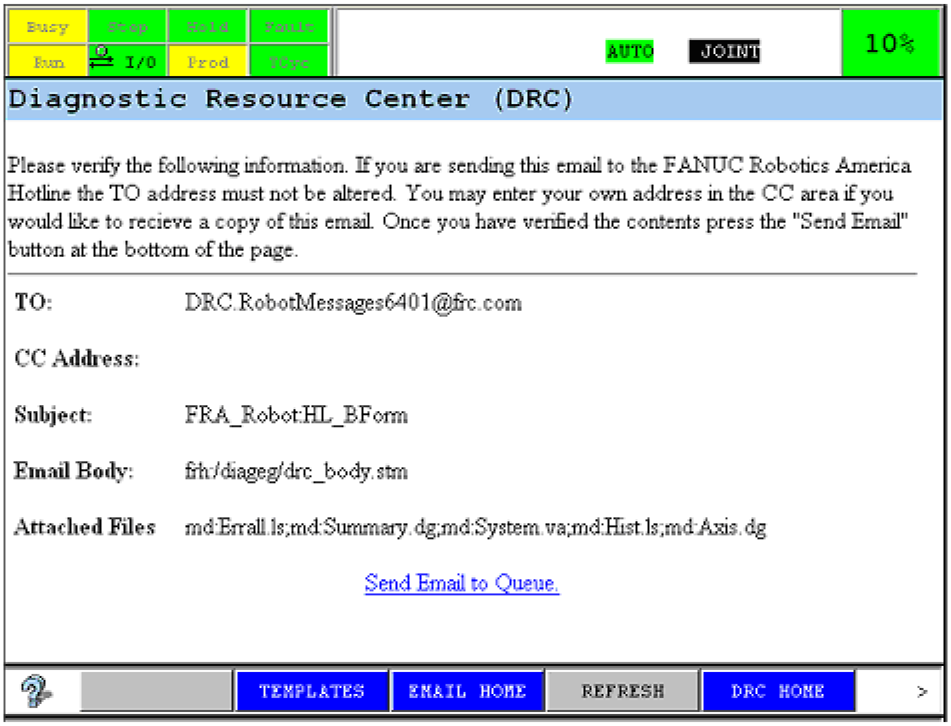
When a template is selected, all information that exists in the template is displayed for you to confirm. You are also given the option of changing the To Address and CC Address for the email on-the-fly. See Figure 14–5 .

Figure 14–5. Changing the Addresses Screen



After you verify that all of the information is correct for the email, select Verify to send the email to the email queue. See Figure 14–6 .

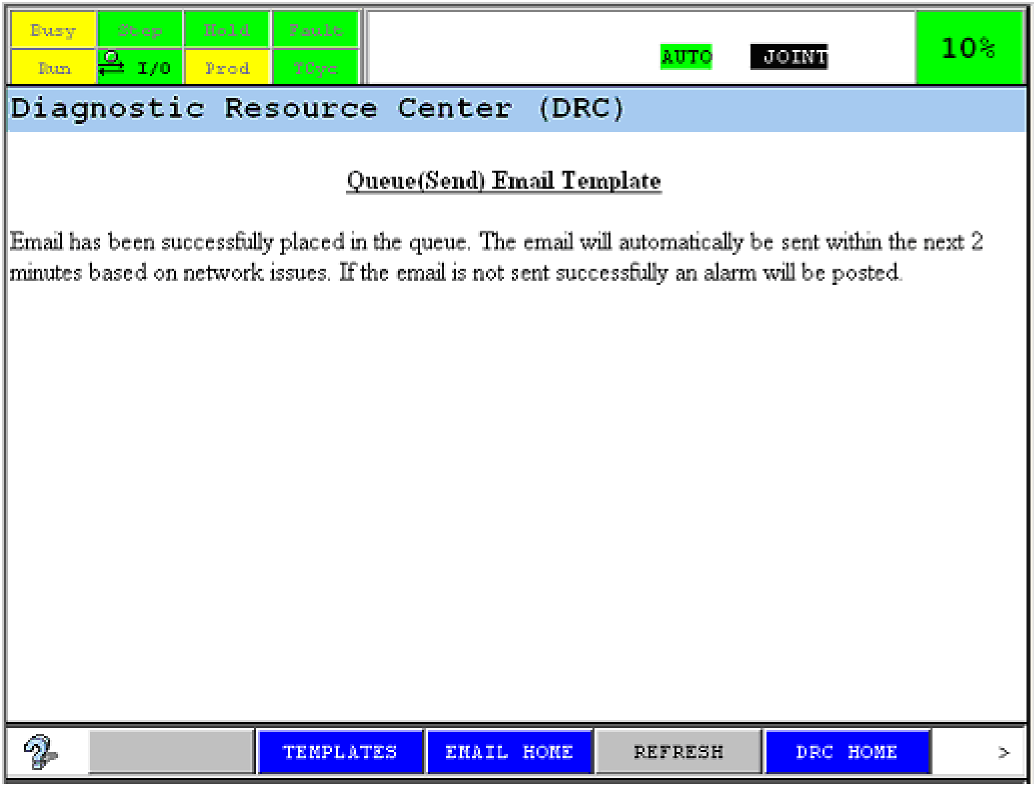
Figure 14–6. Verification Screen



After sending the email, the screen similar to the one shown in is displayed to indicate that the email has been placed in the email queue.

Note If an email send is unsuccessful, the email is deleted and the controller will NOT attempt to re-send the email.

Figure 14–7. Queue(Send) Email Template Screen

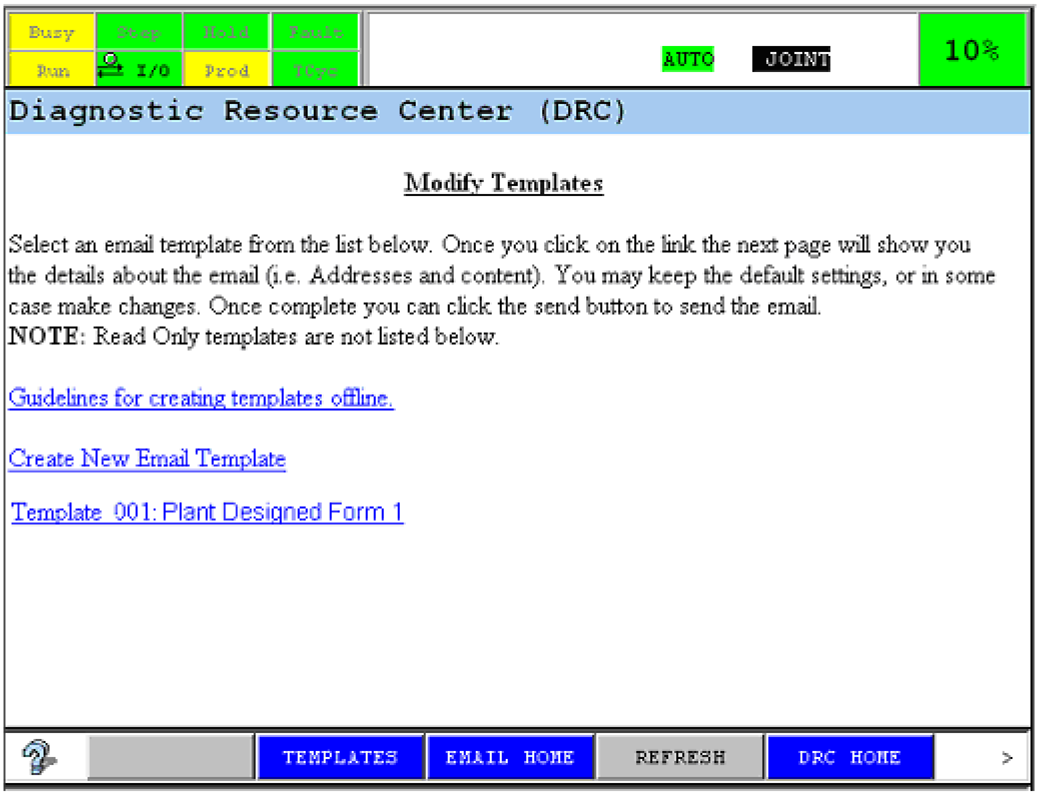


Note With the Basic Email Option installed (SMTP), email can only be sent using this *i*Pendant, or remote *i*Pendant interface.

14.5.2 User Data Templates—Modify, Add, Delete Email Templates

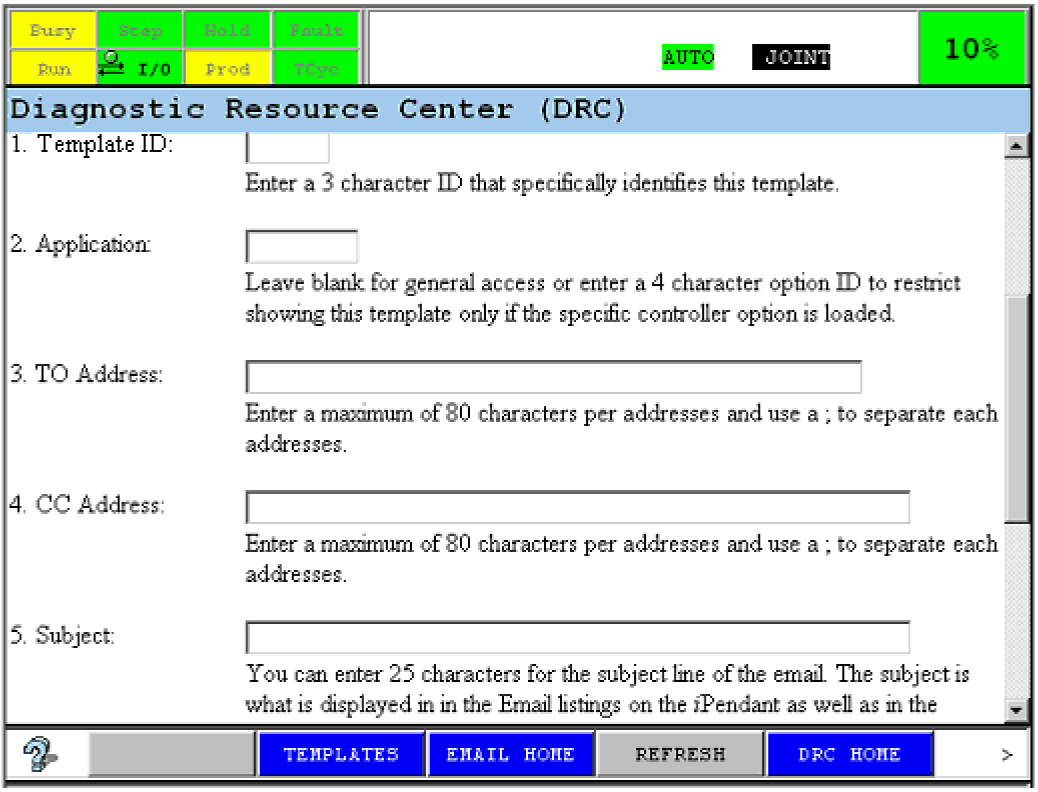
If the advanced email client is loaded on the controller, you can select Data Templates From the Data Services homepage. Then select Create New Email Template. You will see a screen similar to the one shown in Figure 14–8 .

Figure 14–8. Modify Templates



You can then pre-define a custom email template. See Figure 14–9 .

Figure 14–9. Template Setup Screen



After a template has been created, the template will appear on both the Modify Templates screen, as well as the Robot Email Services screen.

To delete a template, go to the Modify Templates screen, select the template to be deleted, and clear out all fields except the template name.

Email templates can be backed up and loaded from the teach pendant. When they are backed up from the teach pendant, the email template file is named email.dg.

14.5.3 Advanced Email Sending Using Teach Pendant or KAREL Programs

With the enhanced email option (EMPL) loaded, you can not only define your own templates, but you can also send email through KAREL and teach pendant programs. When an email template is created, so is an *action* that will send the email template. This action has the same name as the email template and can be executed through a KAREL and/or a teach pendant program.

Note Email has built-in throttle control. Currently only 15 emails can be placed in the email queue at one time. When the 16th email is placed in the queue, a warning alarm will be posted, and the email will be deleted.

Procedure 14-2 Modifying Email Templates

1. Make sure the Email Template to be used is loaded on the controller. This can be verified through the DRC Email Screens. The following example assumes that the template name is "EM1".
2. To send an Email from a teach pendant program:
   a. Create a teach pendent program:
   b. Where you want an email to be sent, make the call: "

```
CALL KLACTION('<Action_Name>')
```

where

```
<Action_Name>
```

is the name of an action. For example, the following line in a teach pendant program runs Action "EM1", which sends email template EM1.

```
1: CALL KLACTION('EM1')
```

3. To send an email from a KARCEL program:

Note The KAREL option must be loaded to support user KAREL programs.

   a. Create a KAREL program and define the routine: "ROUTINE do_klaction(action_str:STRING) FROM klaction".
   b. Make the call:

```
 do_klaction('<Action_Name>')
```

where

```
<Action_Name>
```

is the name of an action. For example, the following KAREL program runs Action “EM1”, which sends the email template **EM1**

```
program mailtst
%NOLOCKGROUP
%NOBUSYLAMP
%NOPAUSESHFT
%NOPAUSE = ERROR + TPENABLE + COMMAND
%NOABORT = ERROR + COMMAND
ROUTINE do_klaction(action_str:STRING) FROM klaction
begin
do_klaction("EM1")
end mailtst
```

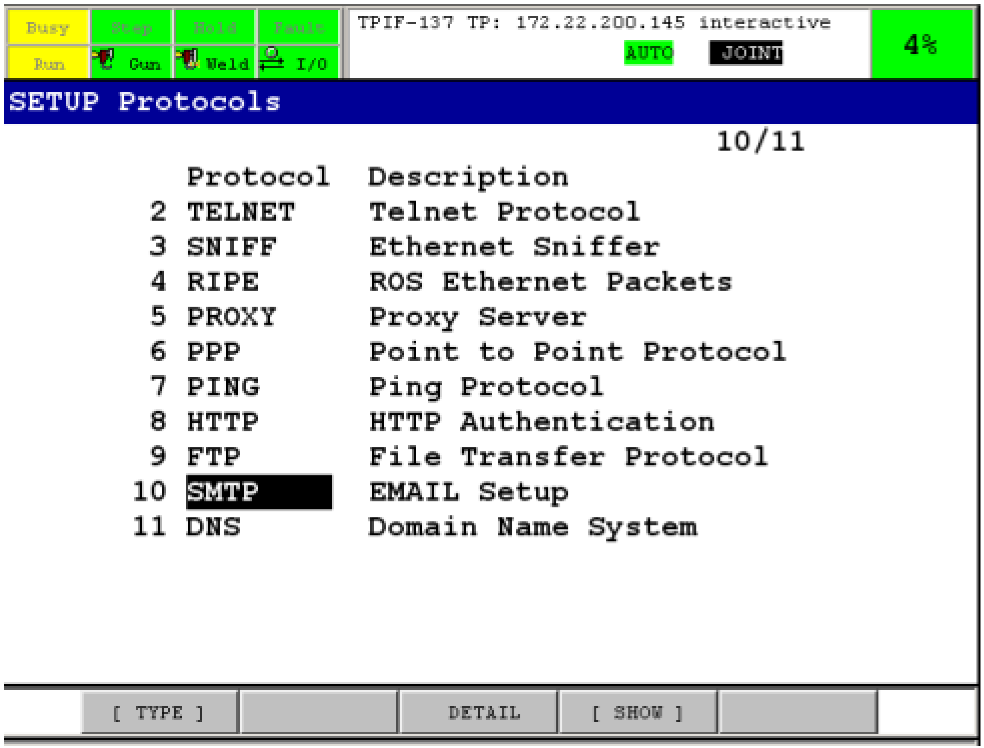
14.5.4 Email Tutorial

Email, or SMTP setup and configuration screens can also be accessed through the HOST COMM Menu when using an *i*Pendant, or when using a remote *i*Pendant in Microsoft® Internet Explorer. For more information on using the remote *i*Pendant, refer to Chapter 18 *ADVANCED iPENDANT FUNCTIONS* .

Procedure 14-3 Accessing SMTP

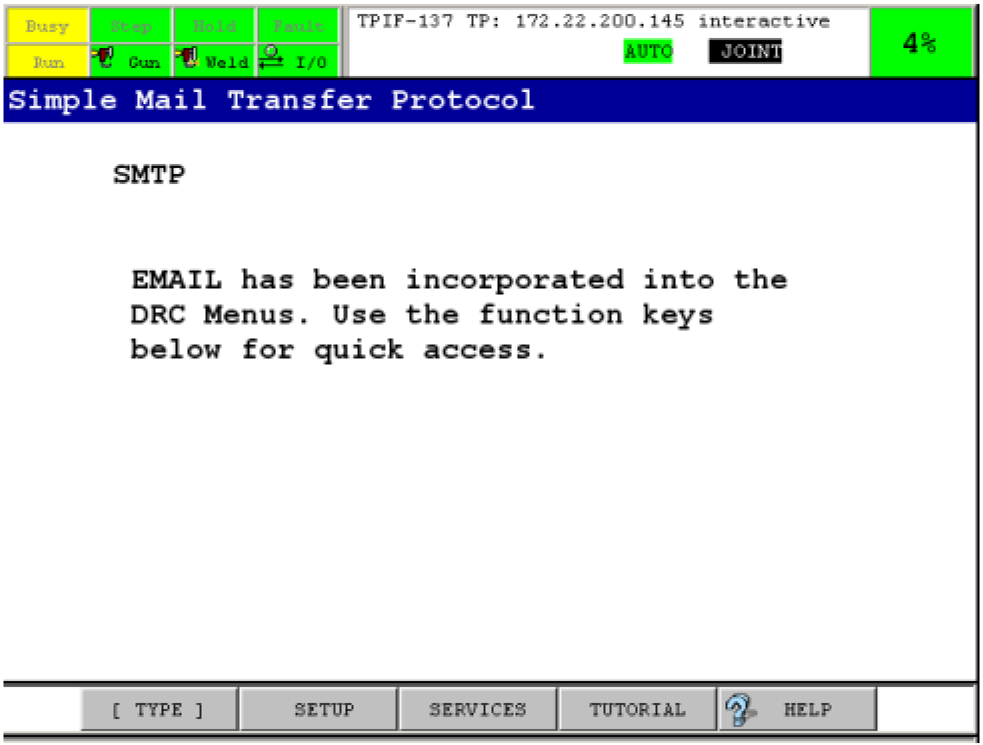
1. To access the SMTP screens in the HOST COMM menu, press Menus -> SETUP -> Host Comm. Then select SMTP in the Protocol list, as shown in Figure 14–10 .

Figure 14–10. Host Comm Protocols List



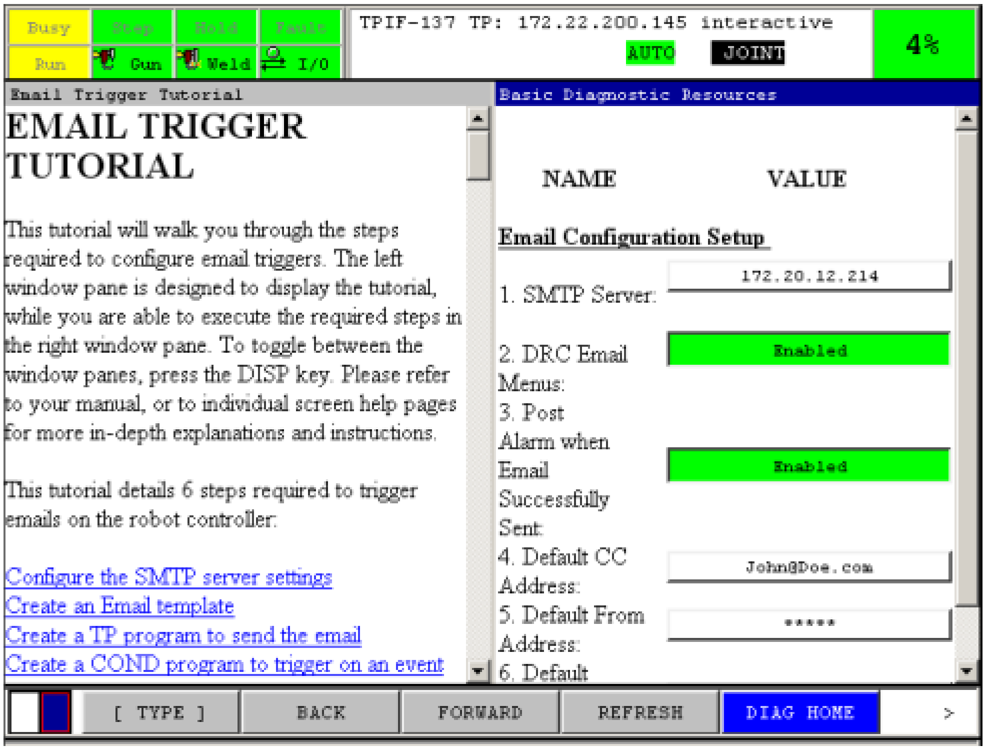
2. The SMTP screen provides function keys used to access the SMTP setup and SMTP services screens incorporated in the DRC Menus. Also provided, is a Tutorial key. See Figure 14–11 for details.

Figure 14–11. Host Comm SMTP screen



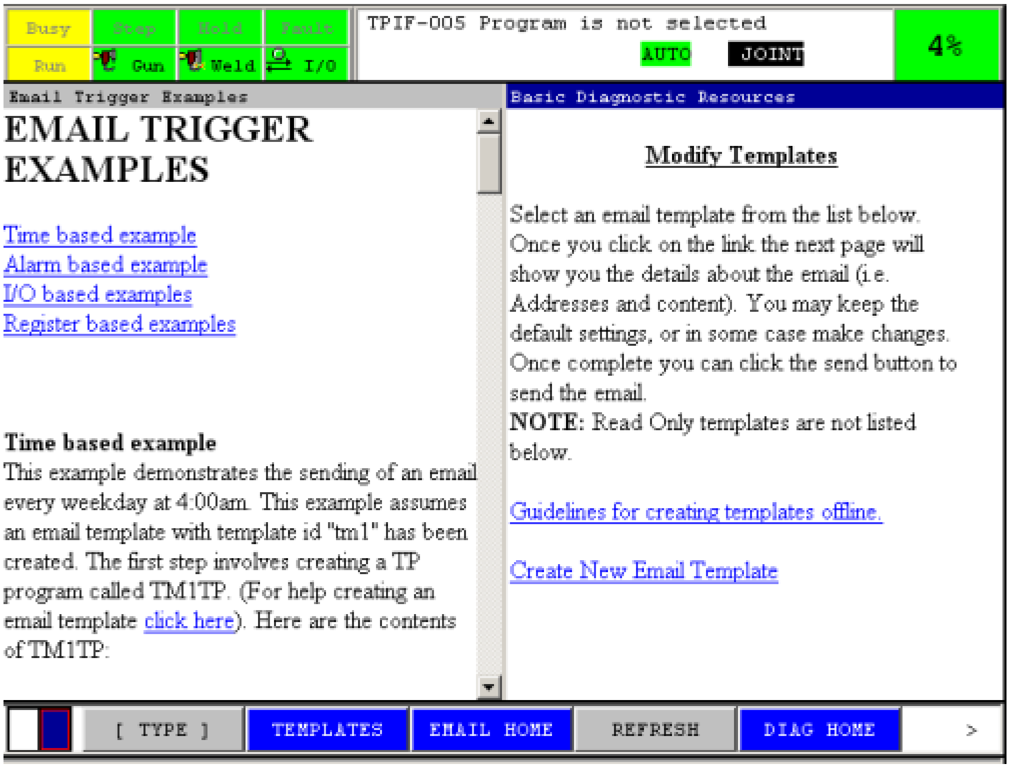
3. A tutorial is included to describe the steps required to configure email triggers. To display the tutorial, press F4 [TUTORIAL]. This will split the screen into two panes. The left window pane is designed to display the tutorial, while you are able to execute the required steps in the right window pane. To toggle between the window panes, press DISP on the *i*Pendant. See Figure 14–12 for an example.

Figure 14–12. Email Tutorial



An examples section is included as part of the tutorial. This section shows examples on how to set up and trigger email messages based on time, alarms, I/O, or registers. See Figure 14–13 .

Figure 14–13. Email examples



Chapter 15

ETHERNET PACKET SNIFFER

Contents

15.1 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.

15.2 SETTING UP THE ETHERNET PACKET SNIFFER

Before you can use the Ethernet Packet Sniffer, you must first define TCP/IP parameters. Table 15–1 describes the Ethernet Packet Sniffer items you can set up. Use Procedure 15-1 to set up the Ethernet Packet Sniffer.

Table 15–1. 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.

Table 15–1. Ethernet Packet Sniffer Setup Items (Cont'd)

ITEM	DESCRIPTION
Buffer Size Default: 50000	The maximum number of bytes used for capturing and storing an Ethernet capture on the controller. This determines the size of the capture buffer.
Ring Buffer 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 15-1 Setting Up the Ethernet Packet Sniffer

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

```
Ethernet Packet Sniffer Status
                                   6/6

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

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

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

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

15.3 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.

15.4 ADVANCED SYSTEM VARIABLES

The following system variables are available for the advanced user. Refer to the *System Software Reference Manual* for more information.

- $SNIFF.$AUTOSAVE
- $SNIFF.$SNDMAIL
- $SNIFF.$MAILADDR
- $SNIFF.$POWERUP
- $SNIFF.$DELAY

Chapter 16

ROS INTERFACE PACKETS OVER ETHERNET (RIPE)

Contents

16.1 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 efficiently. It does this by using very low level services normally provided by an operating system on a bus over the Ethernet. This is designed for high speed and efficiency so messages can be sent typically in less than 4ms.

This network of cooperating robots has a limited number of members; usually less than four but it can support up to sixteen robots. This tightly coupled group of robots is called the *robot ring*. Services are provided to get variable, file and timing related information from any of the ring members.

This feature also supplies a patented synchronization method of the clocks on multiple robots. Therefore, information can be communicated with respect to a common time base. That common time base is the fundamental heartbeat time of the robot. In the case of an R-30*i*B controller, that heartbeat is 2ms.

For all users, several higher level functions are provided.

- A RING based tick time which is a common value among all RING members.
- GET_VAR and SET_VAR allow KAREL variables and TPP registers to be set and read on other robots in the ring.
- Files on member controllers can be read or written to.
- A TELNET proxy allows a TELNET connection to any ring member.

Based on the file capabilities and TELNET, you can configure a system that has just one Ethernet interface to the world at large. Through this one *master* controller all external functions on the *slave* controllers can be exercised. All controllers can be operated through one *i*Pendant. This means that to the outside world a RING of robots can look like a single controller.

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 designed to work with EGD, Ethernet/IP and ArcLink. It is required in order to use RobotLink.

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.

Version Compatibility Issues

Since the robots are connected by RIPE are tightly coupled it is recommended that all of the robots be the exact same version. Up until V8.10P/11 this was a requirement for any RING operations. It is still recommended that you have the same version on all robots in the RING whenever possible.

For V8.10 and later versions a system variable is provided called $RINGVERSION. This variable can be set to the oldest controller version in a RING. This allows the RING to be backward compatible with those older versions. $RINGVERSION is required to be OLDER than the version on all RING members.

For example a V8.10 robot can be in the same ring as V7.70 controllers. The version is actually based on the $VERSION system variable in V7.70 controller.

So to communicate between V8.10 and V7.70 $RINGVERSION is set to the same value as $VERSION on the V7.70 controller. For the case of V7.70/P33 that value is V7.7071. So setting $RINGVERSION to V7.7071 on the V8.10 controller will allow V8.10 to communicate to V7.7071 controllers

This can work for V7.7or V7.5 controllers. However, only ONE older version is supported. This menas that all of the V7.x controllers (R30iA) must be the exact same version. There is no way to communicate between V7.5 and V7.7 controllers with RIPE.

If the variable $RINGVERSION exists on a controller then it can be set. This means that if all controllers are V8.x $RINGVERSION can be set such that they will all communicate. So V8.10P/11 V8.10P/12 and V8.2 can all participate in the same RING.

Note Several options such as Ethernet line tracking, IIC and PickTOOL use RIPE within the options. If the version is not the same these options may not work in a mixed RING. Check the documentation on your specific option of interest.

Each option that relies on RIPE will do version compatibility checking if the $RINGVERSION is not the current version on V8.10 and later software. If it is incompatible RIPE version incompatible error is posted on V8.10. The V7.7x and V7.5x will not post an error in this case but the function will not work

16.2 RIPE SETUP

Table 16–1 defines the items needed to set up the Master and Slaves in a Ring. Use Procedure 16-1 to set up the Master and Slave in a Ring.

Table 16–1. 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.
	Displayed on Master Screen Only	
	Master IP Address	This indicates the IP address of the master. **Caution** The master IP address must be the same for all robots in the ring. Otherwise, communications will not work properly. If the master IP address is set incorrectly, move the cursor to the Master IP address and type in the correct address.
	Number of Members	This item indicates the number of robots in the ring.
	Update Interval	This item indicates the heartbeat time in milliseconds (ms). This is how often RIPE checks to see which robot is online.
	Displayed on Slave Screen Only	
	Slave IP address	This item indicates the IP address of the slave.
	Member Index (1 is Master)	This item should be set to a unique sequential number such as 2, 3, 4, or 5. Press F3, AUTO, to configure the robots to WAIT for the config file to be sent from the master

Procedure 16-1 Setting Up a Master and Slave in a Ring

1. Press MENU.

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

```
SETUP\Host\Comm\\\\\\\\\\\\\\\\\\\\\\\\
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\Host\Comm\\\\\\\\\\\\\\\\\\\\\\\\\
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
    Ring 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 "Ring 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. Refer to the “Setting up a Port” section in the *Software Installation Manual* for more information.

b. Set up the number of members in the ring.

c. Set the Update Interval.

d. Press F3, AUTO, on the MASTER to generate the config file and send it to all of the waiting slaves.

8. You can manually edit the ROSIPCONFG.XML file if necessary. Use F4, SEND, to send the files to the Slaves manually.

9. Press F3, LOAD to load the ROSIPCFG.XML file.

10. Press >, NEXT and then F4, RECV on the slave to put the slave into a mode where it will receive the config file from the master.

16.3 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 ring 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 devices on MHROB03 apply. So you can also access for example MD: to get binary teach pendant programs. For example from KCL:

```
KCL> CD RNG:\MHROB03\MD
KCL> LOAD TP
```

This will load the teach pendant program directly from MHROB03 into memory.

For user KAREL programs, a similar example is valid:

```
KCL> CD RNG:\MHROB03\MD
KCL> LOAD PROG
```

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 no one nogroup       1493 apr 23 2007 inactive.htm
```

Refer to the “File System” section in the *KAREL Reference Manual* for more information on the diagnostic files on MD:.

16.4 ASSOCIATED OPTIONS

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

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

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 broadcase will also work for multicast. Multicast is much more flexible approach. See section Section 16.9 for details on network configuration.

16.5 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>
```

16.6 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 Ring Index. Therefore, if you have 16 robots telnet can connect to the 16th robot via port 2316.

16.7 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)
```

Refer to the *KAREL Reference Manual* for more information.

16.8 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.

16.9 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 robot can be set for a specific connection speed and duplex (refer to the chapter titled “Setting Up TCP/IP” in the *Internet Options Setup and Operations Manual*). However be very careful that both ends of the connection use the same speed and duplex mode. Be careful not to set one end of a connection for autonegotiate and set the other end to a specific speed duplex – both ends must autonegotiate, or both ends must be fixed to the same settings.

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 16-2 .

Procedure 16-2 Verifying Link State

1. Press MENU.
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. Limit 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. EtherN et/IP UDP packet has a TTL (time to link) value of one. You will not be able to route I/O traffic across more than one switch.

Quality of Service (QOS) techniques provide mechanisms to prioritize network traffic. Generally on an Ethernet network all packets are equal. Packets can be dropped or delayed within network infrastructure equipment (for example, switches) in the presence of excessive traffic. Which packets are dropped or delayed is random.

QOS is a term covering several different approaches to prioritizing packets including:

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

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

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

- 100 Mbps baud rate
- Full duplex connections

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

Note If there is a significant amount of multicast traffic, the switch should support IGMP snooping (multicast aware). Please consider this when Ethernet/IP and/or RIPE (robot ring) traffic exists.

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

Some examples of switch products are:

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

Chapter 17

PC SHARE

Contents

17.1 OVERVIEW

PC Share allows the robot controller to connect to a Shared Folder on a PC running a supported version of Microsoft® Windows®, or on a Linux computer running a support version of SAMBA (www.samba.org), using the SMB/CIFS protocol. This "connection" to the Shared Folder allows the robot controller to browse the remote directory tree, and to read and write files remotely. This functionality is similar to the "mapping a drive" concept in Microsoft® Windows® operating systems–the robot is able to "map" a client tag to a directory location on the Windows®-based network. (Microsoft, Windows, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries).

Only client functionality is supported, that is the robot controller can access files on a remote PC; server functionality is not supported on the robot, a PC cannot access files on the robot controller.

17.2 SETTING UP AND STARTING PC SHARE

17.2.1 Setting Up PC Share Client Tags

Before you can use the PC Share Interface, you must do the following:

- Define TCP/IP parameters (Section 2.5).
- Install the PC Share Interface software.
- Define PC Share on a client device (Procedure 17-1).

Refer to the *Software Installation Manual* for information on installing host communications options such as the PC Share interface software.

Table 17–1 lists and describes the items you must set up to define a client device.

Table 17–1. Client Device Definition Setup Items

ITEM	DESCRIPTION
***** This item is normally set up by the user. Other items can normally remain at their default values.	
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**. For PC Share, select **PC Share**. And so on.

Table 17–1. Client Device Definition Setup Items (Cont'd)

ITEM	DESCRIPTION
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: • UNDEFINED - the device is not defined. • DEFINED - the device is defined. • STARTED - the device is defined and started. The Startup State should normally be set to defined. When in the defined state the client tag is started automatically from KCL or from the FILE screen on the teach pendant whenever it is used.
Server IP/Hostname*	This item specifies the Hostname or IP address of the remote server to which the connection will be made.
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. • 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 Procedure 17-1 to define and start PC Share on a client device.

Procedure 17-1 Defining and Starting PC Share on a Client Device

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

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

5. Press F4, [SHOW].
6. Select 2, Clients. You will see a screen similar to the following.

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

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

```
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:  15 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. The default value is 15 minutes. A value of 0 indicates no timeout.
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. Refer to Section 17.4 for a detailed explanation of how PC Share passwords are stored and protected in the robot controller.
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.

17.2.2 Configuring PC Share

Table 17–2 details the PC Share items that can be configured from the Host Comm screens. Use Procedure 17-2 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 17–2. 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 17-2 Configuring PC Share

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

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

5. Select PC SHARE and press enter. You will see a screen similar to the following.

```
PC Share
 PC Share Client
  Enabled         :   TRUE
  Domain          :   *****************
  WINS Server     :   *****************
  Broadcast Disc  :   TRUE
```

6. To enable the PC Share protocol, set Enable to TRUE. To disable the PC Share protocol, set Enable to FALSE.

7. To configure a domain, cursor to Domain and enter a domain name. This may be optional depending on your network.

Note To override the domain set in the PC Share Client screen with a domain to be used only for a specific client tag, set the domain in the system variable $SMB_CLNT[X].$DOMAIN, where X is the client tag index CX:, for the specific client tag. Power-cycle the robot controller for new settings to take effect.

8. To configure a WINS server, curser to WINS and enter the server's IP address.

9. To enable broadcast discovery, set Broadcast Disc to TRUE. Set Broadcast Disc to FALSE to disable broadcast discover.

17.3 ACCESSING AND USING PC SHARE CLIENT DEVICES

17.3.1 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.

17.3.2 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.

Note Generally the path_name field can be omitted based on the client tag setup. A typical example of using a client tag from KCL would be:

```
KCL> COPY C1:myprog.tp to MD:
```

17.3.3 Starting and Stopping a Client Device

Use Procedure 17-3 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 17-3 Starting and Stopping a PC Share Client Device

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

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

7. Move the cursor to the tag on which you wish to perform an action.
8. Press F2, [ACTION].
9. Select the action you want to perform:

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

   - To **define** a device, select DEFINE. The device must be in the undefined state.
   - To **undefine** a device, select UNDEFINE. The device must be in the defined state.
   - To **start** a device, select START. The device must be in the defined state.
   - To **stop** a device, select STOP. The device must be in the started state. The device will change to the defined state.

10. **To configure the client device to start automatically at power up:**

a. Move the cursor to the client tag you want to start automatically and press F3, DETAIL.

b. Move the cursor to Startup State and press F4, [CHOICE].

c. Select START, and press ENTER.

The client device will now start automatically when the controller is turned on.

Note The host device MUST be capable of accepting this PC Share login if the tag is set to START AUTOMATICALLY when you turn the robot on. If the remote host is not available, the robot controller will wait approximately one minute to timeout before completing powerup. For this reason it is recommended to have client tags powerup in the **DEFINE** state. The controller will automatically start any defined client tag when the tag is used.

17.3.4 Teach Pendant File Access

After a client device has been defined, it can be used from the teach pendant. Refer to the "Program and File Manipulation" chapter in the appropriate application-specific *Setup and Operations Manual* for information on program and file manipulation.

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:

17.3.5 KCL File Access

After a client device has been defined, it can be used from KCL to copy files between the controller and any node device on the network. This node device can be either the host computer or another controller with server tags started.

The general format for copying files is the KCL COPY command with the appropriate host communications definitions for the source and destination file specifications.

Transfer File FROM Controller TO Node Device

To transfer a copy of the source file (**src_file_spec**) to the destination device and destination file (**dst_file_spec**), type the following:

```
KCL> COPY <src_file_spec> TO <dst_file_spec>
```

For example:

```
KCL> COPY RD:\TPDEF.VR TO C1:\
```

Transfer File FROM Node Device TO Controller

To transfer a file from a node device to the controller, type the following, for example:

```
KCL> COPY C1:\TPDEF.VR TO RD:\TPDEF.VR
```

Generate Directory On Node Device

To generate a directory of files on a node device, type the following:

```
KCL> DIR <src_file_spec>
```

Delete File from Node Device

To delete a file on a node device, type the following:

```
KCL> DELETE FILE <src_file_spec>
```

Save Program to Node Device

To save a program to a node device, type the following, for example:

```
KCL> CD C1:
KCL> SAVE TP TEST1
```

Load Program from Node Device

To load a program from a node device, type the following, for example:

```
KCL> CD C1:
KCL> LOAD TP TEST1
```

17.4 PASSWORDS AND SECURITY

17.4.1 Passwords

It is recommended that the robot controller(s) be assigned its own username and password for the Windows®-based network to which it will be connecting. Doing so illustrates good security practice by reducing the chance of a user's personal password being compromised, and allowing file and Share permission to be set appropriately for the robot controller on the remote PC(s).

However, it is expected that some users will instead choose to enter their corporate or otherwise sensitive authentication information into the robot from time to time to allow the robot to access files and Shares on their Windows®-based network via PC Share. To this end, the following security precautions have been taken on the robot controller to prevent PC Share passwords from being accidentally or maliciously compromised.

Passwords entered into a Client Tag configured for the PC Share protocol are not stored on the robot controller. Rather, the password in read in, and converted to a 21-byte NTLM session key using the RSA MD-4 hash algorithm. Then the hashed password is salted with an additional 3-bytes of random data, and the resulting 24-bytes are encrypted using the DES (Data Encryption Standard) algorithm using an encryption key unique to the specific robot controller. Any DRAM used to read in the original password is cleared. The original plain text password cannot be recovered even by the robot controller software.

Storing (or forgetting) the password in this manner works because only the 21-byte NTLM key is needed by the client device to connect and authenticate to the remote server. The additional salting and encrypting of the NTLM key is an additional layer of protection to prevent anyone from loading the NTLM key on a different robot. As such, PC Share passwords cannot be backed-up and restored onto a different controller.

Note The password must be entered through the Teach Pendent on each robot using PC Share client tags. A PC Share password cannot be backed-up or restored.

17.4.2 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.

17.5 CONFIGURE THE REMOTE PC

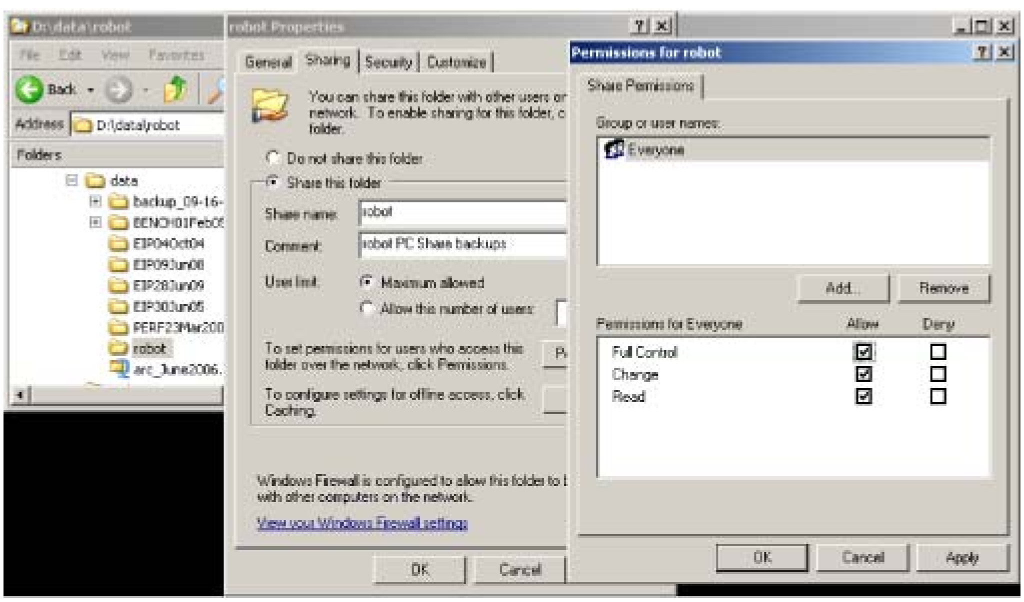
Procedure 17-4 will walk through the steps required to setup a Share in Windows® XP. See also Figure 17–1 .

Procedure 17-5 will walk through the steps required to setup a Share in Windows Vista®.

Procedure 17-4 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.

Figure 17–1. Sharing a Folder in Windows® XP



Procedure 17-5 Sharing a folder in Windows Vista®

1. From the Control Panel, go to the Network and Sharing Center. See Figure 17–2 .
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 17–3 .
6. Select the users to share with, set the appropriate Permission Level, and click on the Share button. See Figure 17–4 .
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.

Figure 17–2. Microsoft Vista® Network and Sharing Center

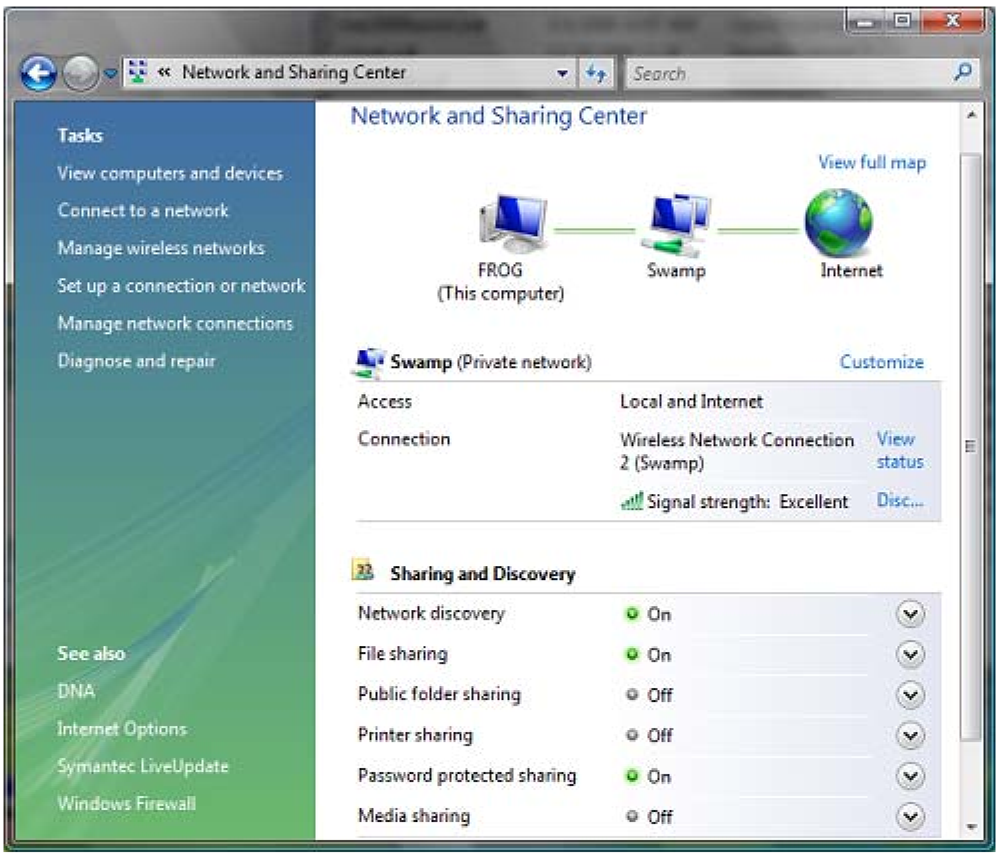


Figure 17–3. Selecting Share... in Windows Vista®

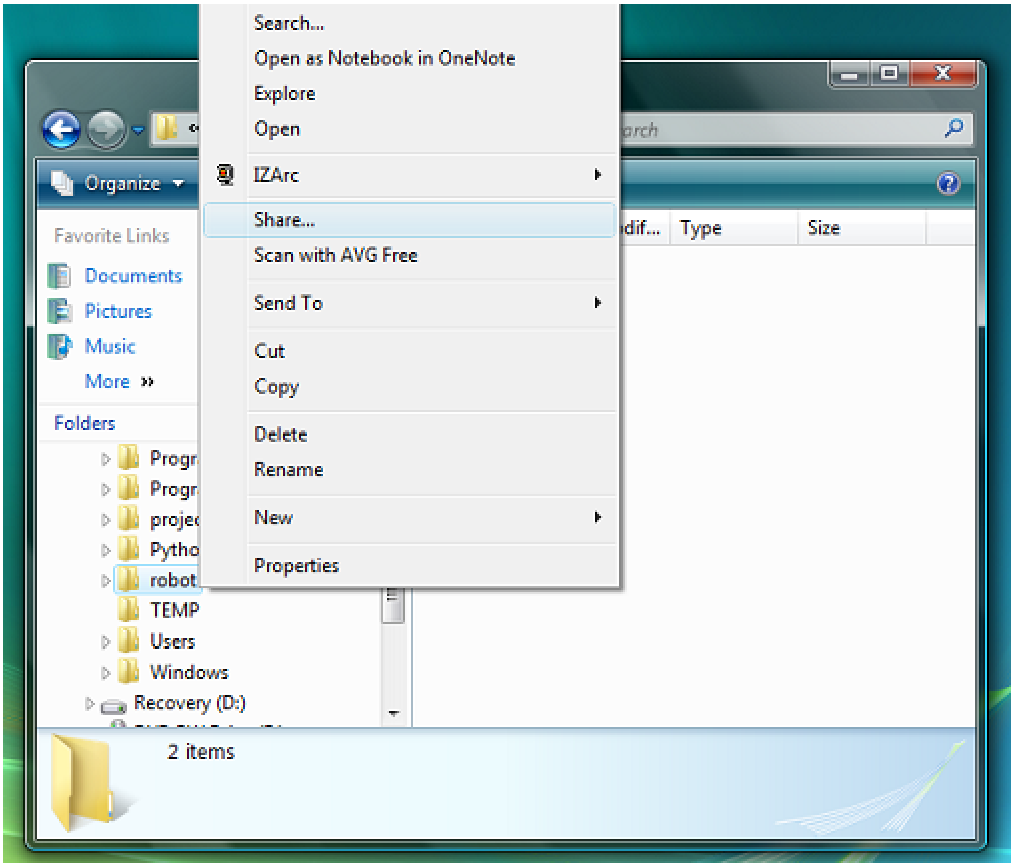
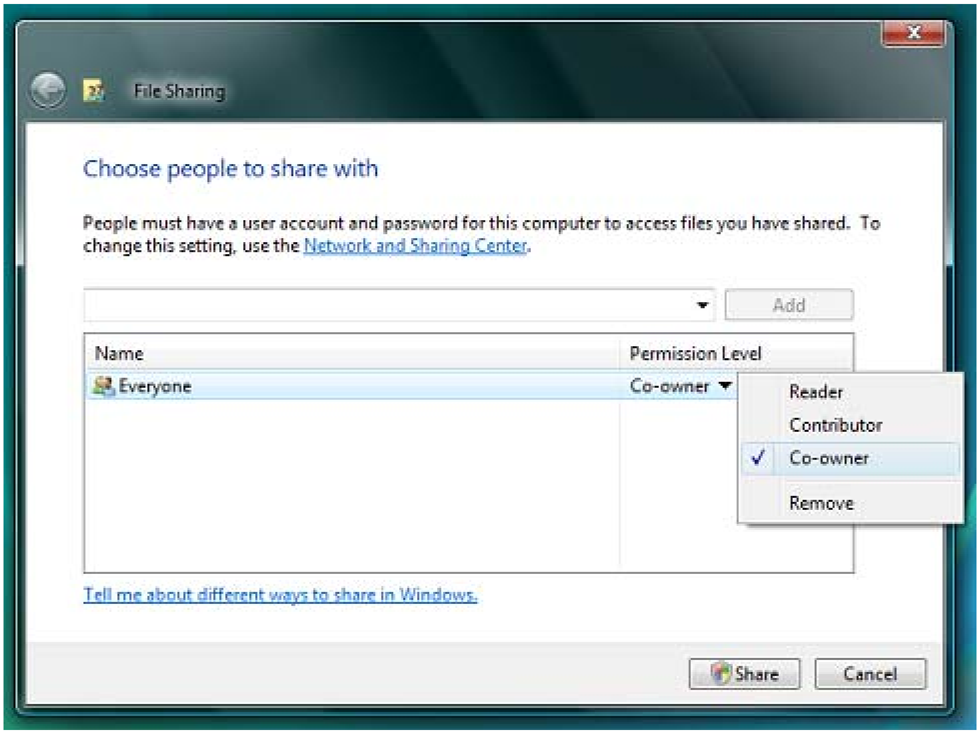


Figure 17–4. Choose People to Share With in Microsoft Vista®



Chapter 18

ADVANCED iPENDANT FUNCTIONS

Contents

18.1 OVERVIEW

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

- **Installing *i*Pendant Controls** This allows you to install the components necessary to perform any *i*Pendant advanced functions.
- **Remote Monitoring** This allows you to monitor the *i*Pendant screens remotely using a PC and Microsoft® Internet Explorer.
- **Remote Operation** This allows you to display an *i*Pendant-like screen remotely, using a PC and Microsoft® Internet Explorer, and navigate the various menus available in the system. This provides the capability to diagnose system problems remotely.
- **Jog in Auto Mode** This allows you to jog the robot remotely using a PC and Microsoft® Internet Explorer. Jogging is only allowed when the teach pendant is disconnected from the robot and the robot is in Auto Mode. The 4D window will display the robot. When used with the 4D Graphics option, you can use visual jog to preview the direction that the robot will move before actually jogging the robot.
- **PC Remote iPendant** This option allows you to connect as the iPendant and jog the robot remotely using a PC and Microsoft® Internet Explorer. The connection is only allowed when the robot is in Auto Mode. The teach pendant, if disabled, will be logically disconnected from the robot. It also allows Remote Operation with no limitations, such as running a program using the FWD/BWD keys, foreground editing a program, and clearing DCS alarms.
- **Remote iPendant on HMI** This option allows Monitor iPendant (JECHO), Navigate iPendant (JCGTP), and Jogging iPendant (JITP) from any HMI device that has a modern browser. iPendant Controls are not required to be installed.

18.2 *i*PENDANT CONTROLS INSTALLATION

Some of the *i*Pendant advanced functions require you to have the *i*Pendant Controls loaded on the PC that you will be using. Procedure 18-1 describes how and where to obtain these controls and install them on your PC.

Procedure 18-1 Installing FANUC America Corporation *i*Pendant Controls

Conditions

- You are using a PC that is connected to the Internet
- You have obtained a FANUC America Corporation Customer Resource Center (cRc) login username and password.

Steps

1. Use Microsoft® Internet Explorer on your PC to connect to the FANUC America Corporation Customer Resource Center (cRc).
2. Type your username and password at the prompt. Contact FANUC America Corporation if you need to request login information.
3. Click Downloads.
4. Click Download PC Software.
5. Scroll to iPendant Controls, click the appropriate version, and follow the directions to begin downloading a self-extracting zip file that contains all the necessary setup files.
6. Log off of the FANUC America Corporation cRc website.
7. On your PC, navigate to the temporary directory where you saved the file iPendant_controls_Vx.xx.exe (where x.xx is the appropriate version number).
8. After the files are unzipped, double-click setup.exe in the setup directory.

   The FANUC America Corporation *i*Pendant Controls are now installed on your PC and can be used to create custom screens or for remote *i*Pendant operation or monitoring.

18.3 REMOTE MONITORING

18.3.1 Overview

Remote monitoring provides you with the capability to display and monitor the current *i*Pendant screens and operations on a PC using Microsoft® Internet Explorer. It is meant as a DISPLAY ONLY mode and therefore, the remote connection normally cannot interact with the screens or affect the operation of the *i*Pendant or robot controller. See Section 18.3.4 for limitations.

The following can be displayed remotely:

- All *i*Pendant screens available from the MENUS and [TYPE] keys.
- All popup menus, and windows.
- Multiple window configurations (Double and Triple modes on the *i*Pendant for example).
- Any input from the *i*Pendant numeric keypad, Function keys or cursor movement.
- Any custom screens that are accessible from the [TYPE] menu in the BROWSER Screen.
- Any top level HELP or DIAGNOSTIC Screens.

18.3.2 Setup

Setup consists of:

- Identifying requirements
- Configuring Internet Explorer ™
- Testing the Network Connection

18.3.2.1 Requirements

The following are the requirements for remote display of the *i* Pendant screens.

- The PC must have Microsoft® Internet Explorer 5.5 or greater installed.
- The PC must have the *i*Pendant Controls installed. Refer to Procedure 18-1 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 *i*Pendant connected.
- The robot controller must be connected to a network and be properly configured for Network access to the above PC.

18.3.2.2 Configuring Internet Explorer ™

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

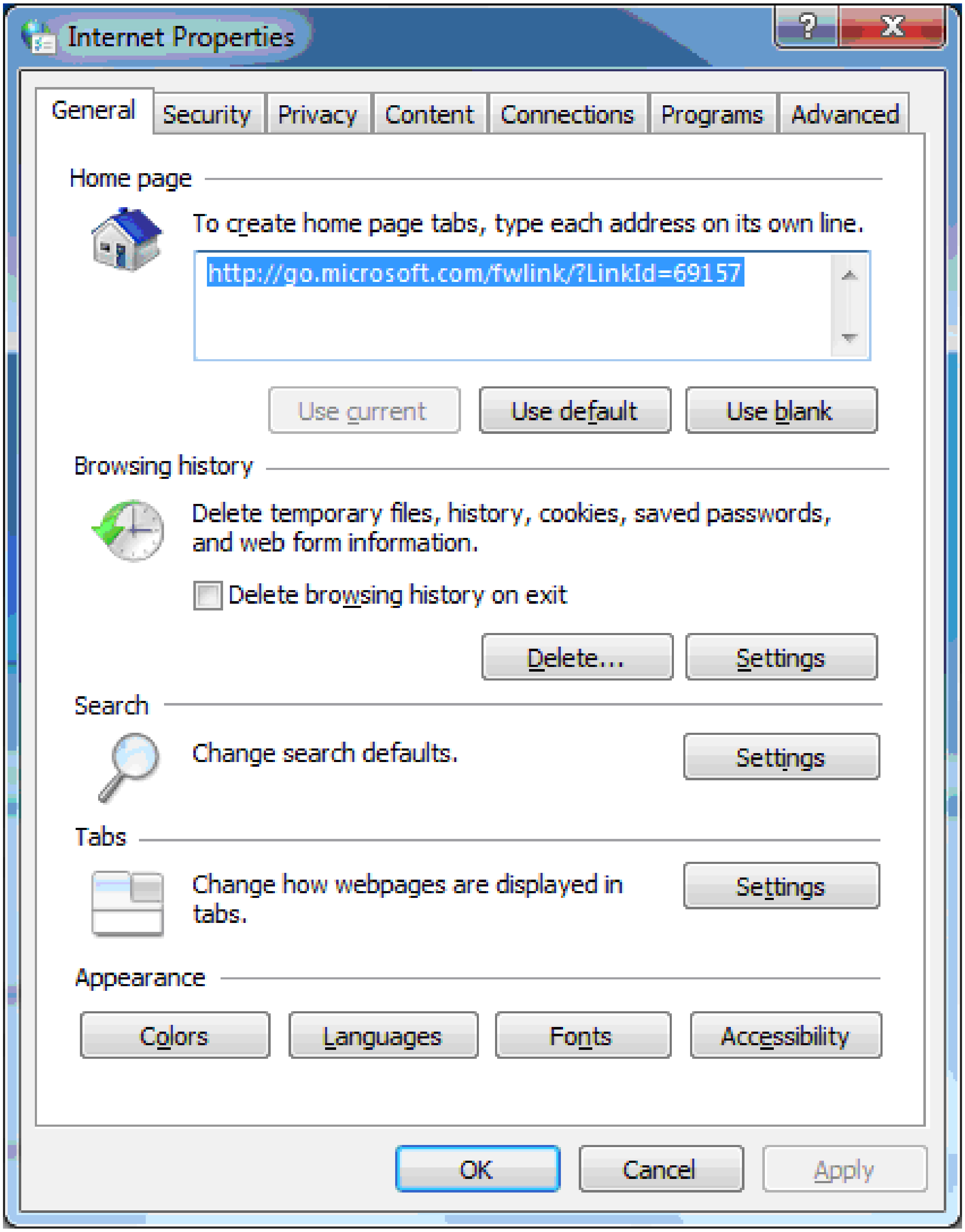
Procedure 18-2 Configuring Internet Explorer

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 ™.

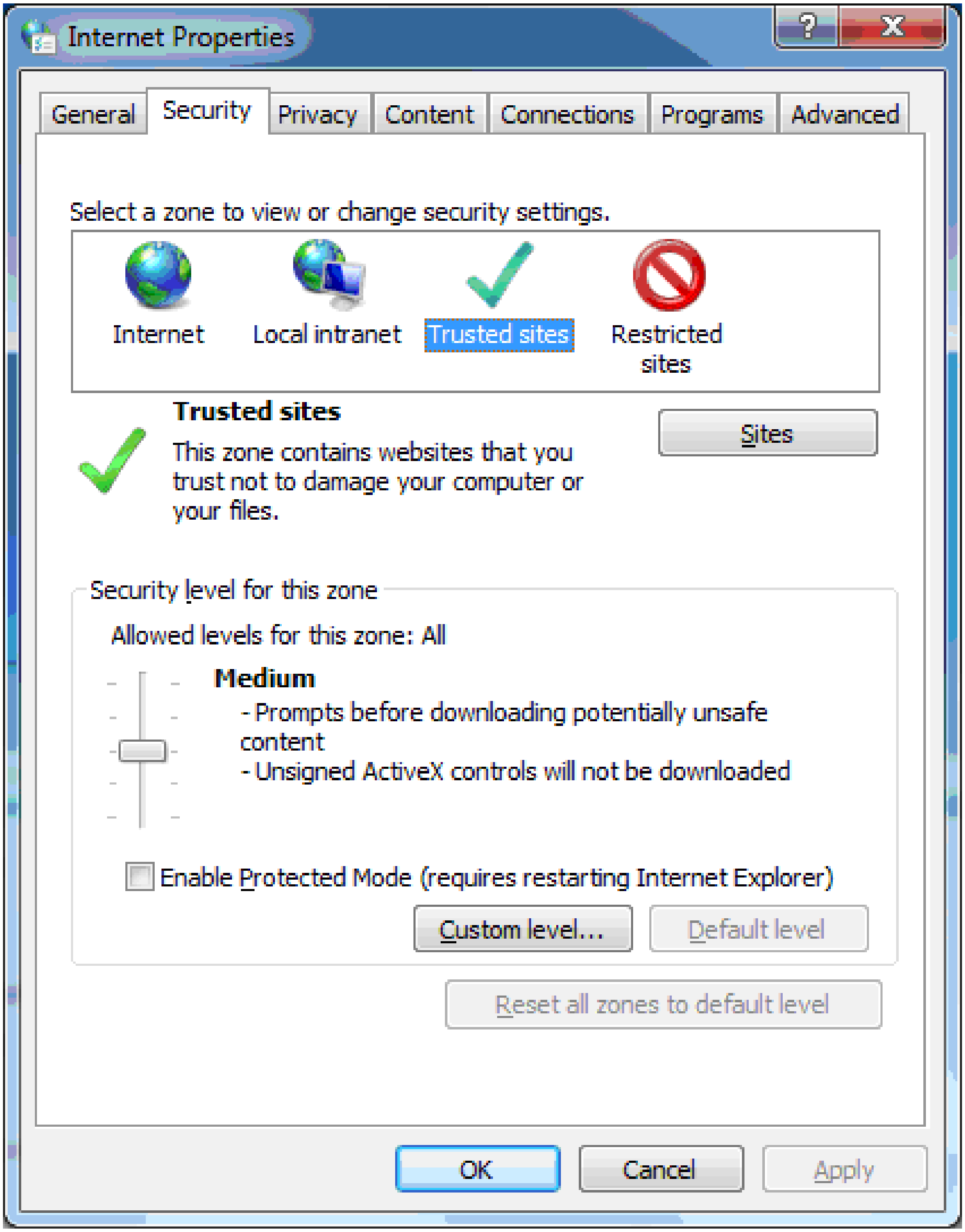
Figure 18–1. Internet Properties



2. Trusted Sites:

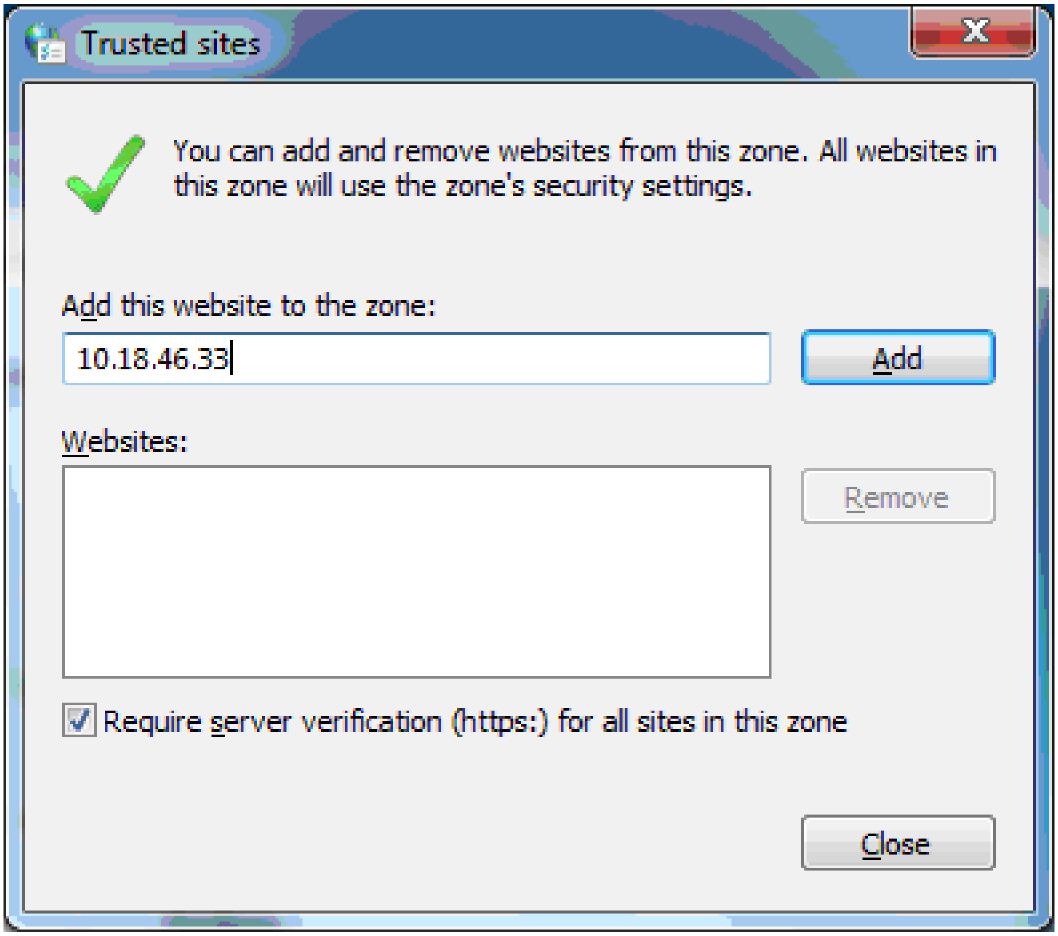
a. Select the [Security] tab.

Figure 18–2. Security



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

Figure 18–3. Trusted Sites



c. Uncheck the [Require server verification (https:) for all the sites in this zone] box.

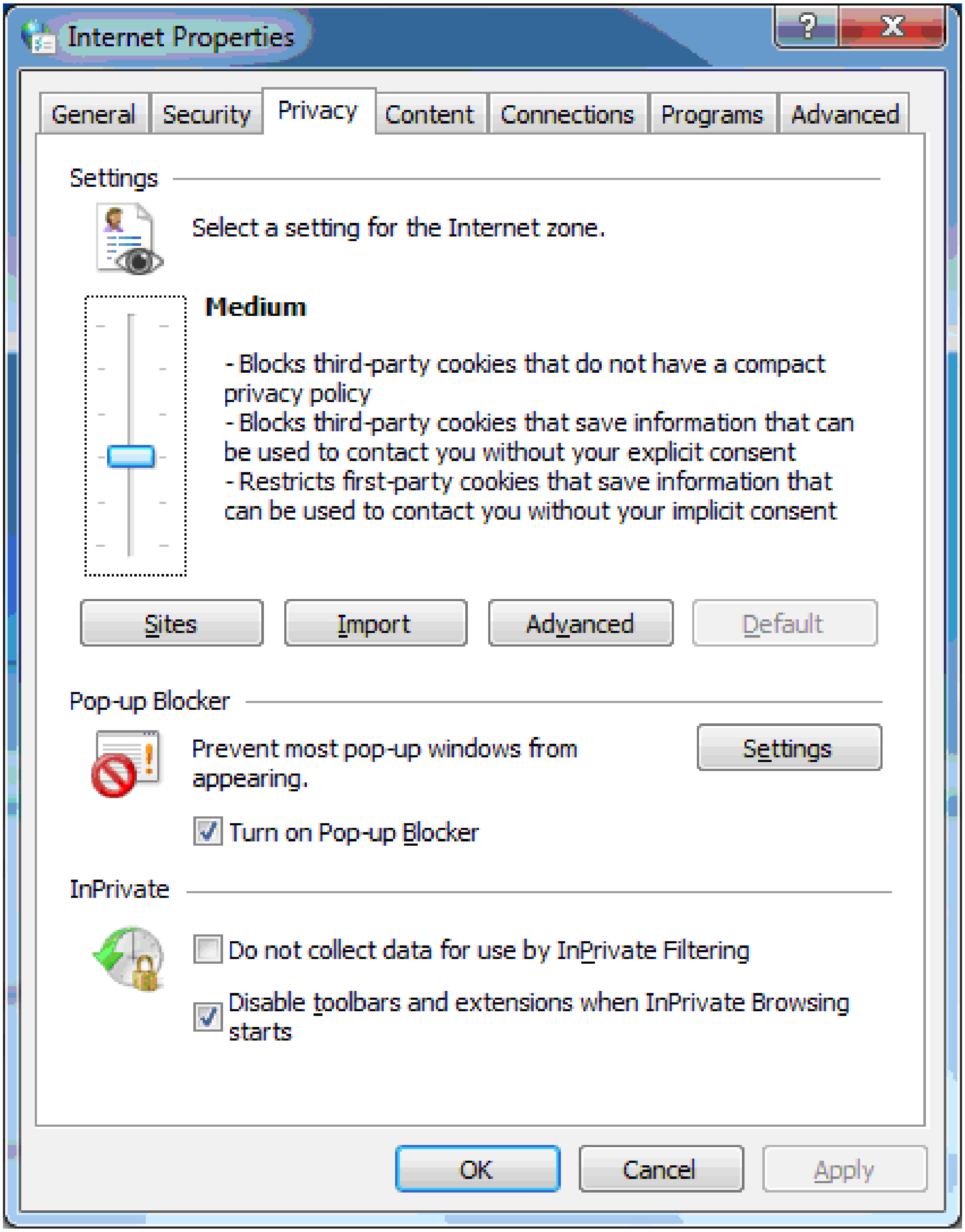
d. In the [Add this Web site to the zone] textbox, enter the IP address of the robot controller (or the last digit of the IP address can be replaced by *). Then, click the [Add] button.

e. Click the [Close] button to close the dialog box.

3. Popup Blockers

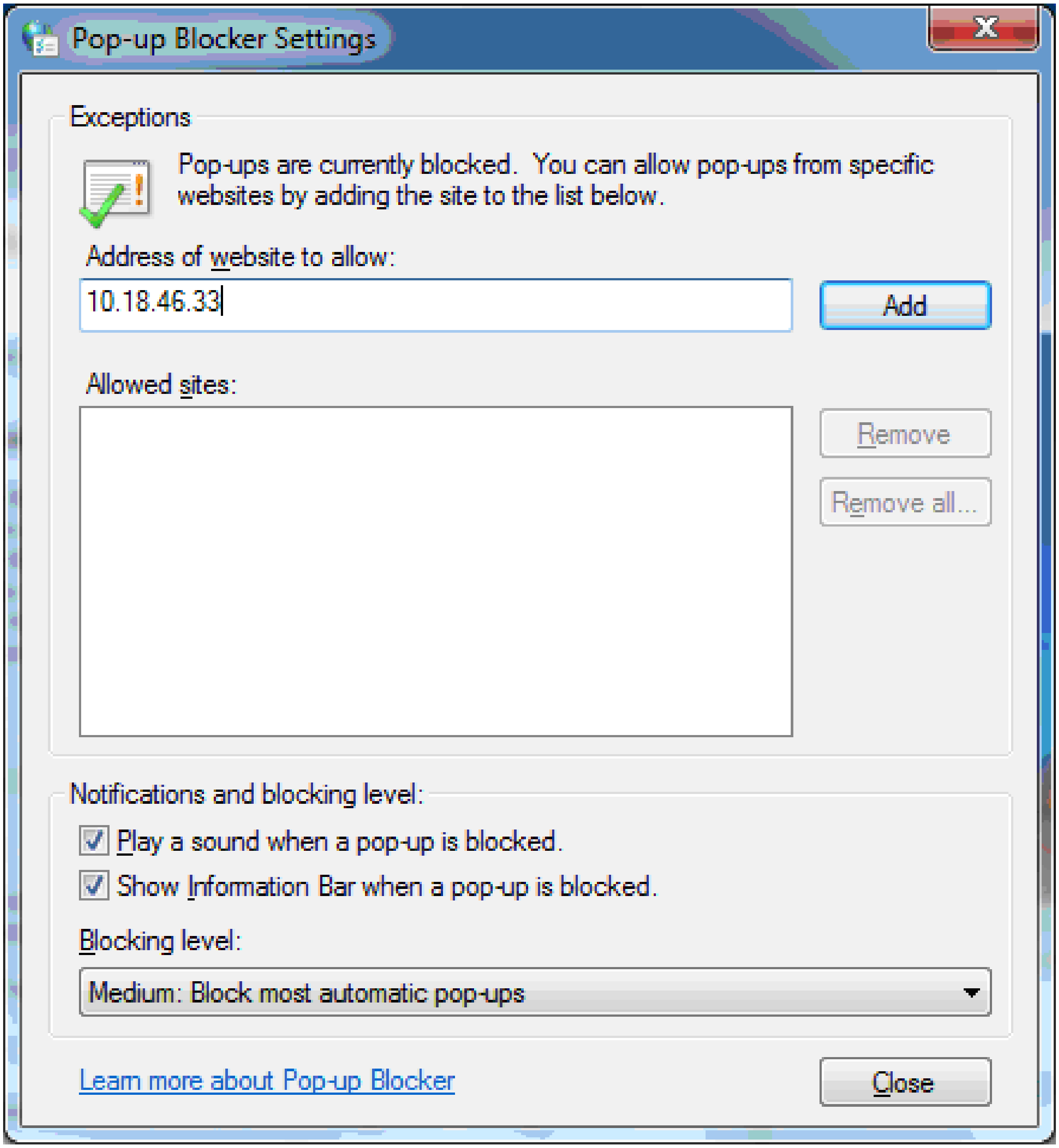
a. Select the [Privacy] tab.

Figure 18–4. Privacy



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

Figure 18–5. Pop-up Blocker Settings



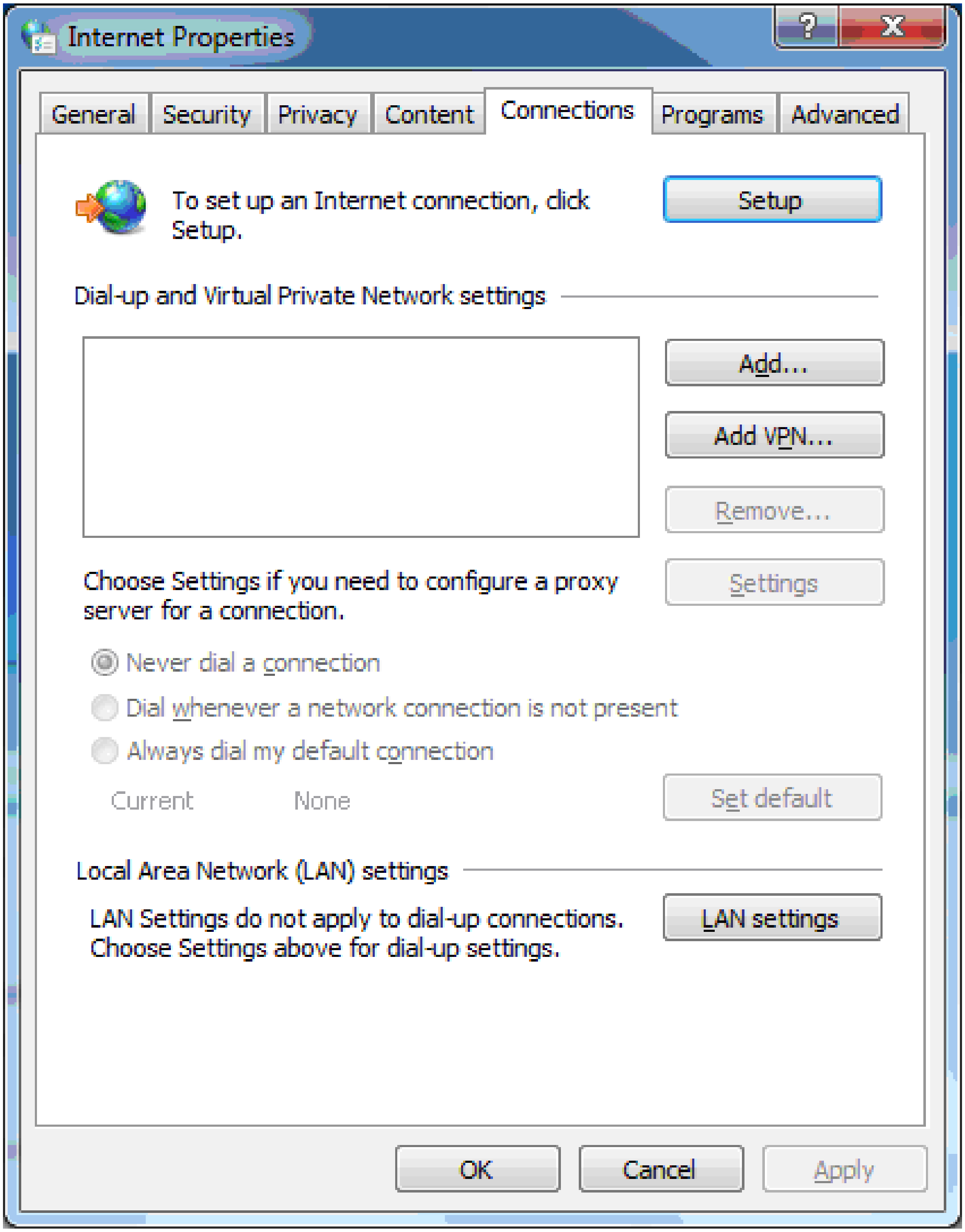
c. Enter the IP address of the robot controller in the [Address of Web site to allow] textbox, and click the [Add] button.

d. Click the [Close] button to close the dialog box.

4. Proxy Setting

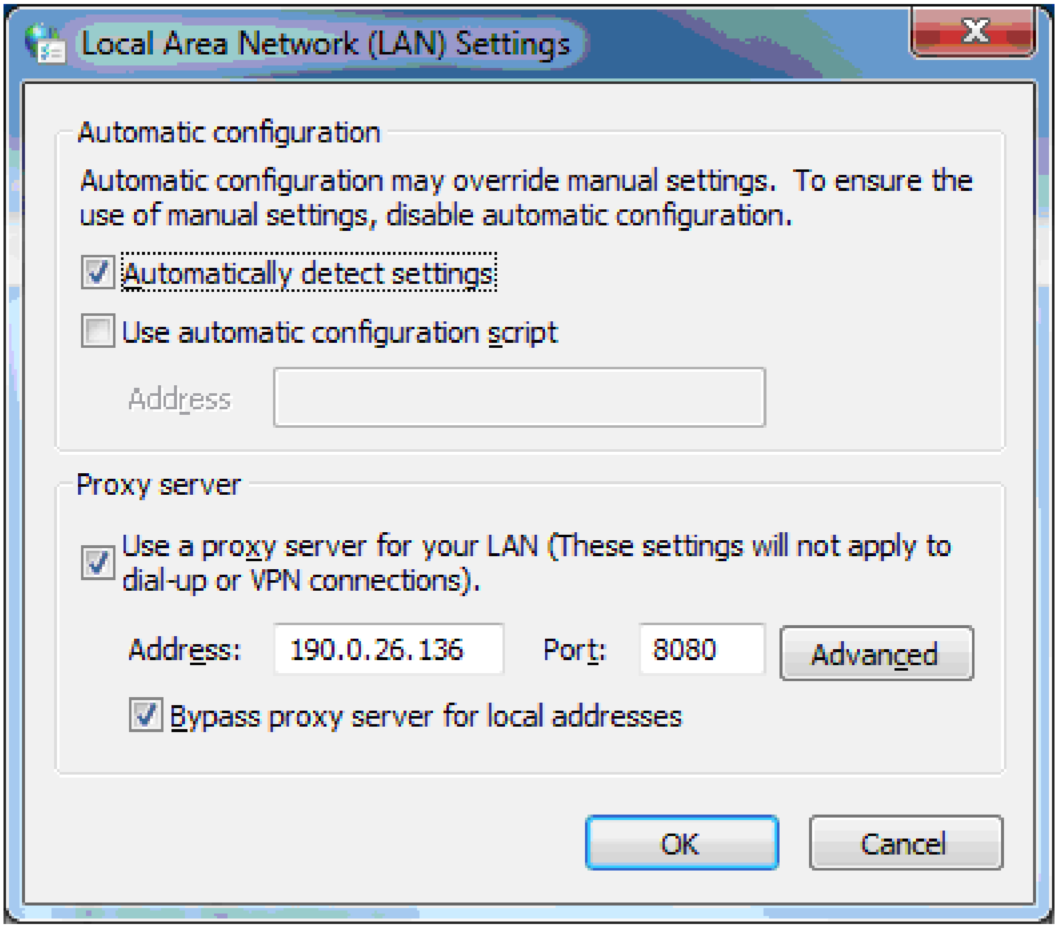
a. Select the [Connections] tab.

Figure 18–6. Internet Properties



b. Click the [LAN Settings] button.

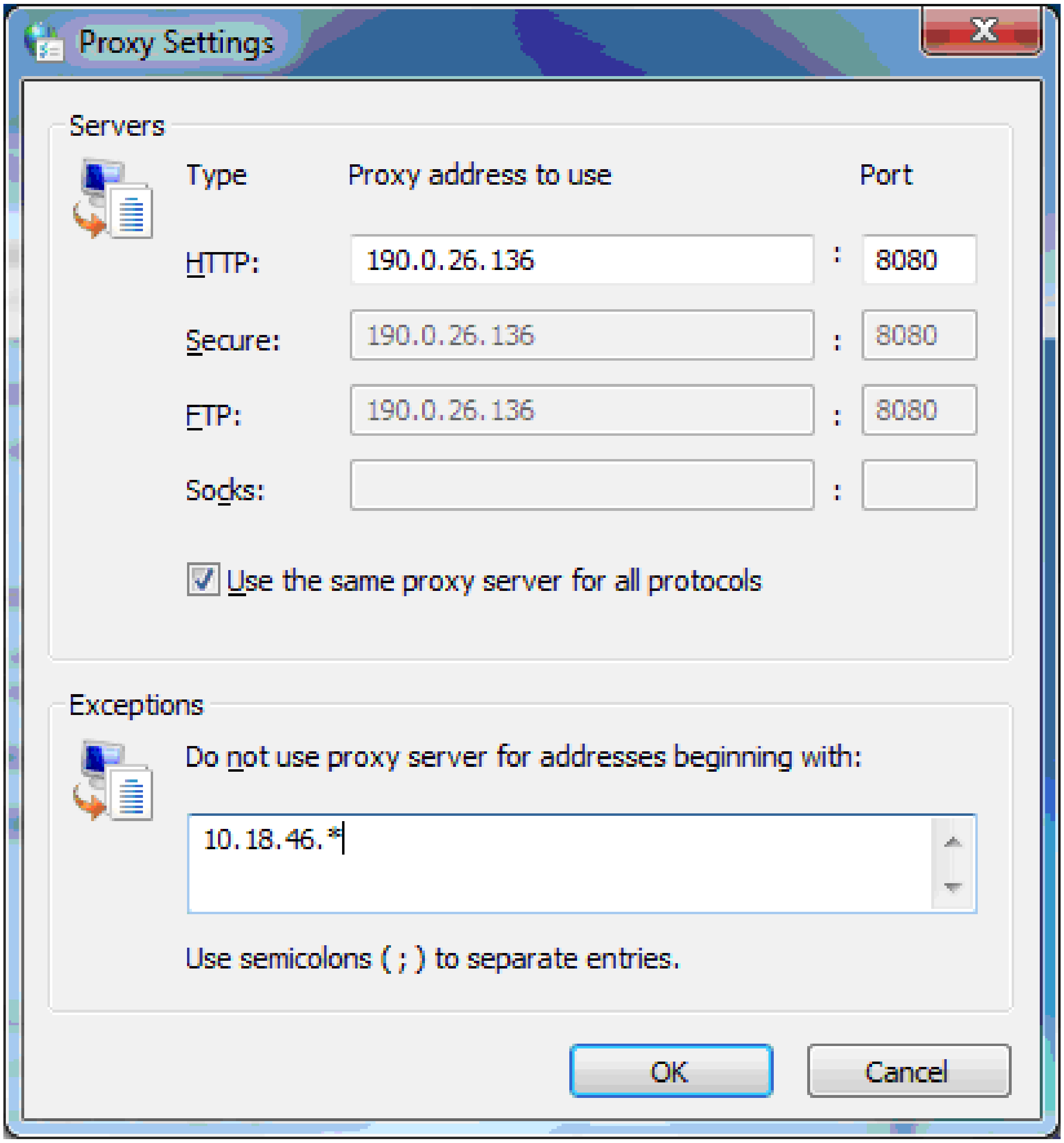
Figure 18–7. Local Area Network (LAN) Settings



c. When the [Use a proxy server for your LAN] check box is not checked, proceed to Step 4g . When it is checked, perform Step 4d through Step 4f .

d. Click the [Advanced…] button of [Proxy server].

Figure 18–8. 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.

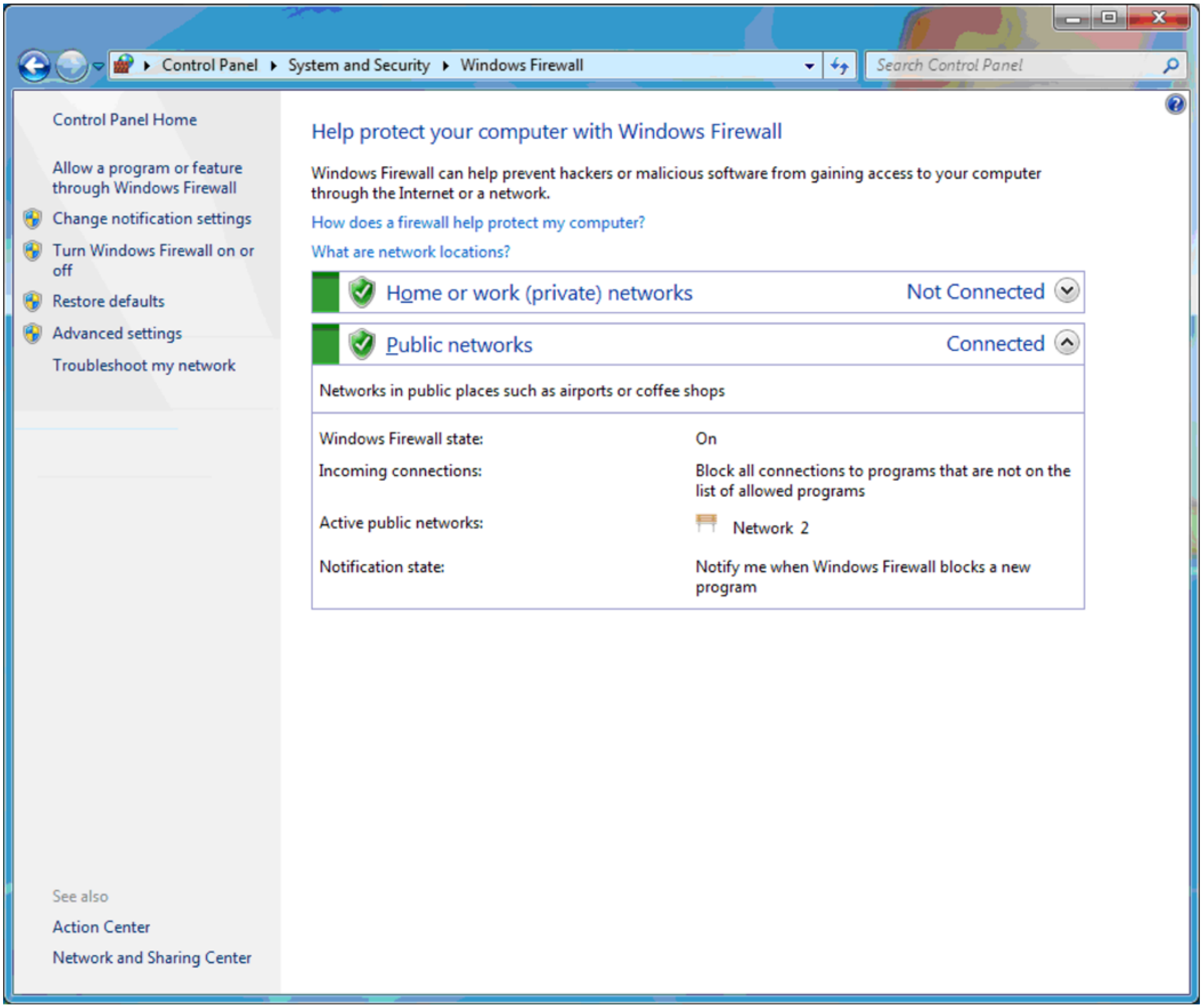
g. Click the [OK] button to close the Internet property page.

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

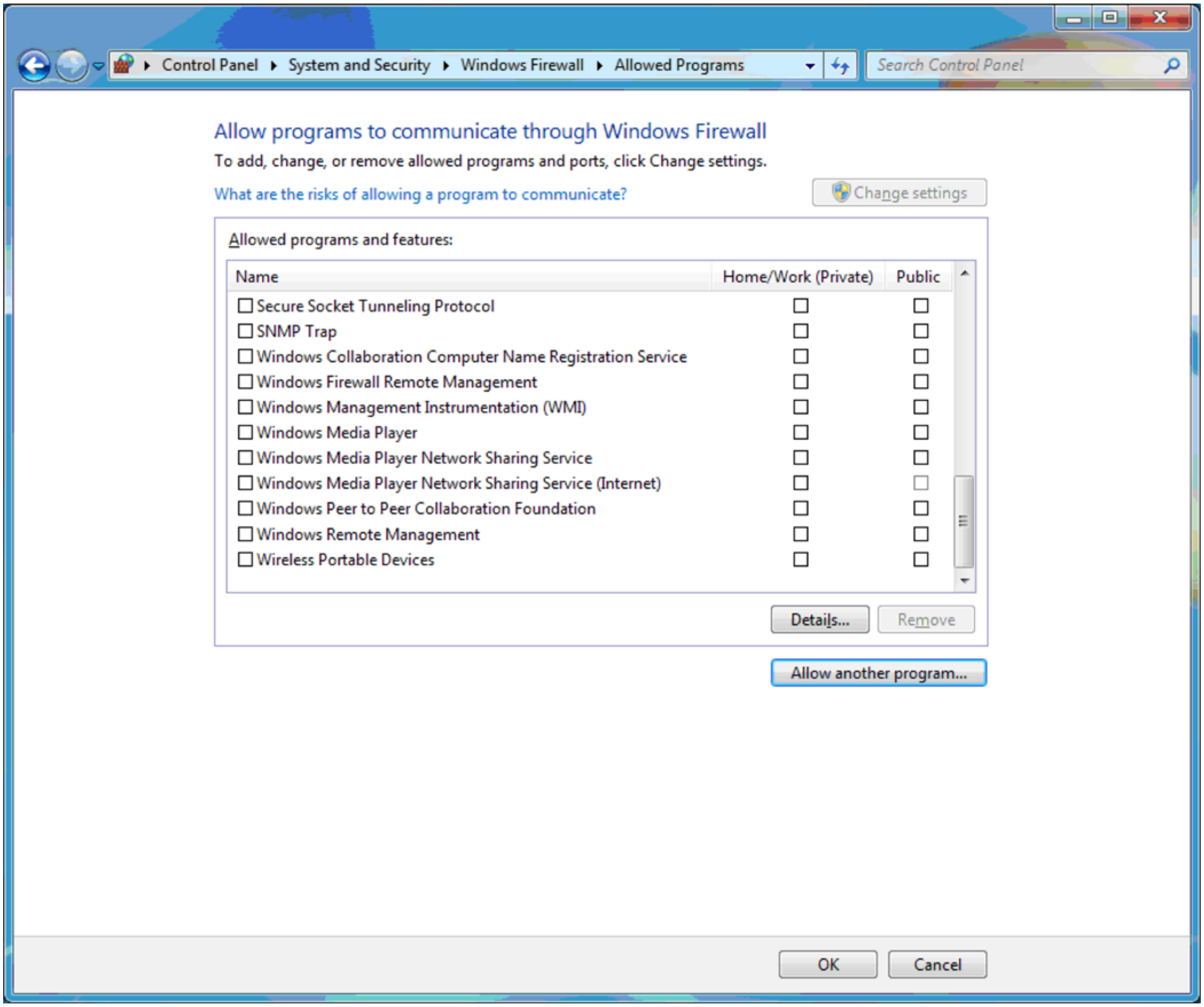
      a. In the Control Panel window, open [Windows Firewall].

Figure 18–9. Windows Firewall



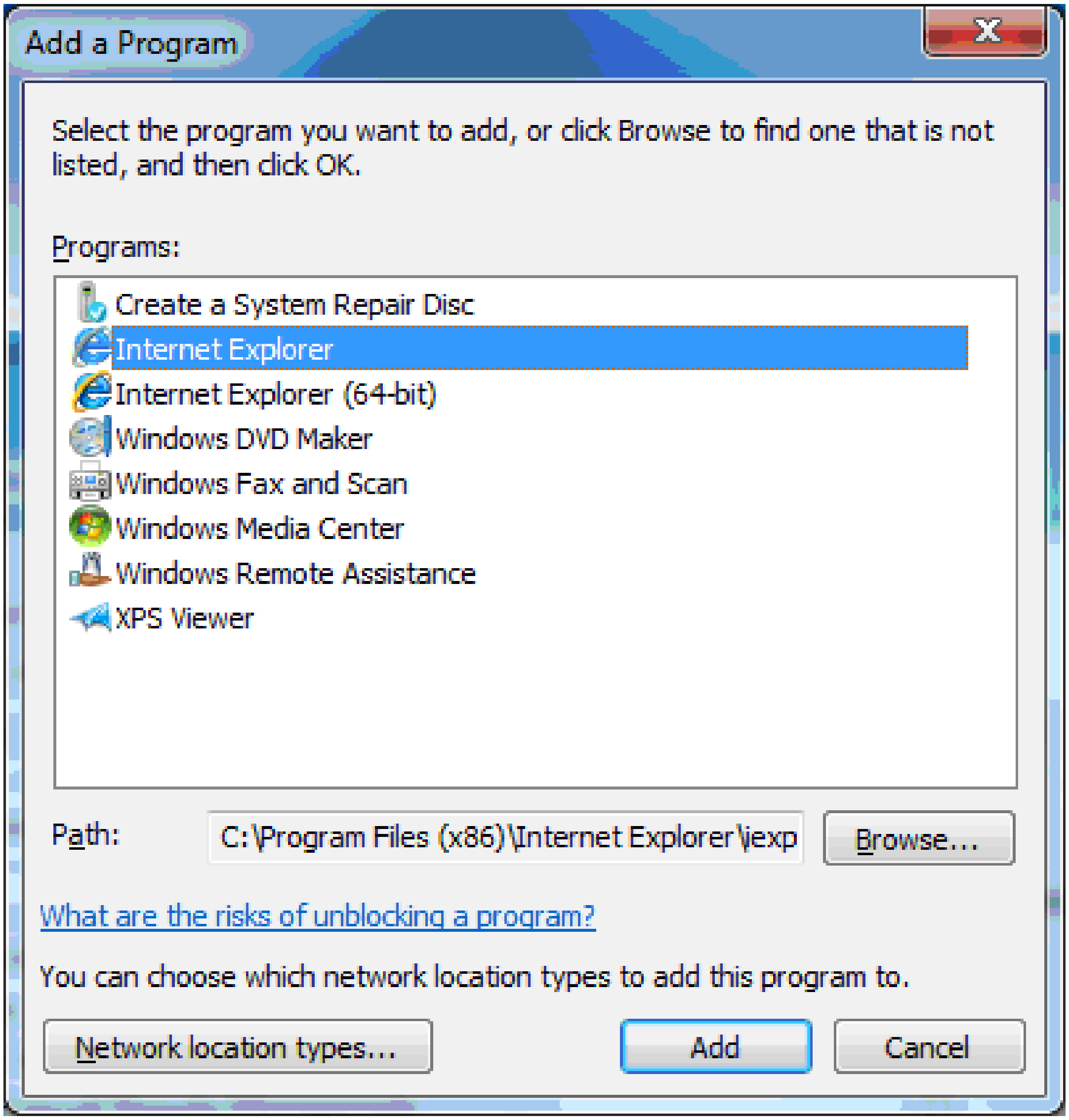
b. Click [Allow a program or feature through Windows Firewall].

Figure 18–10. Allowed Programs



c. Click the [Change settings] button.

Figure 18–11. Add a Program



d. Select [Internet Explorer] in the list, and click the [Add] button.

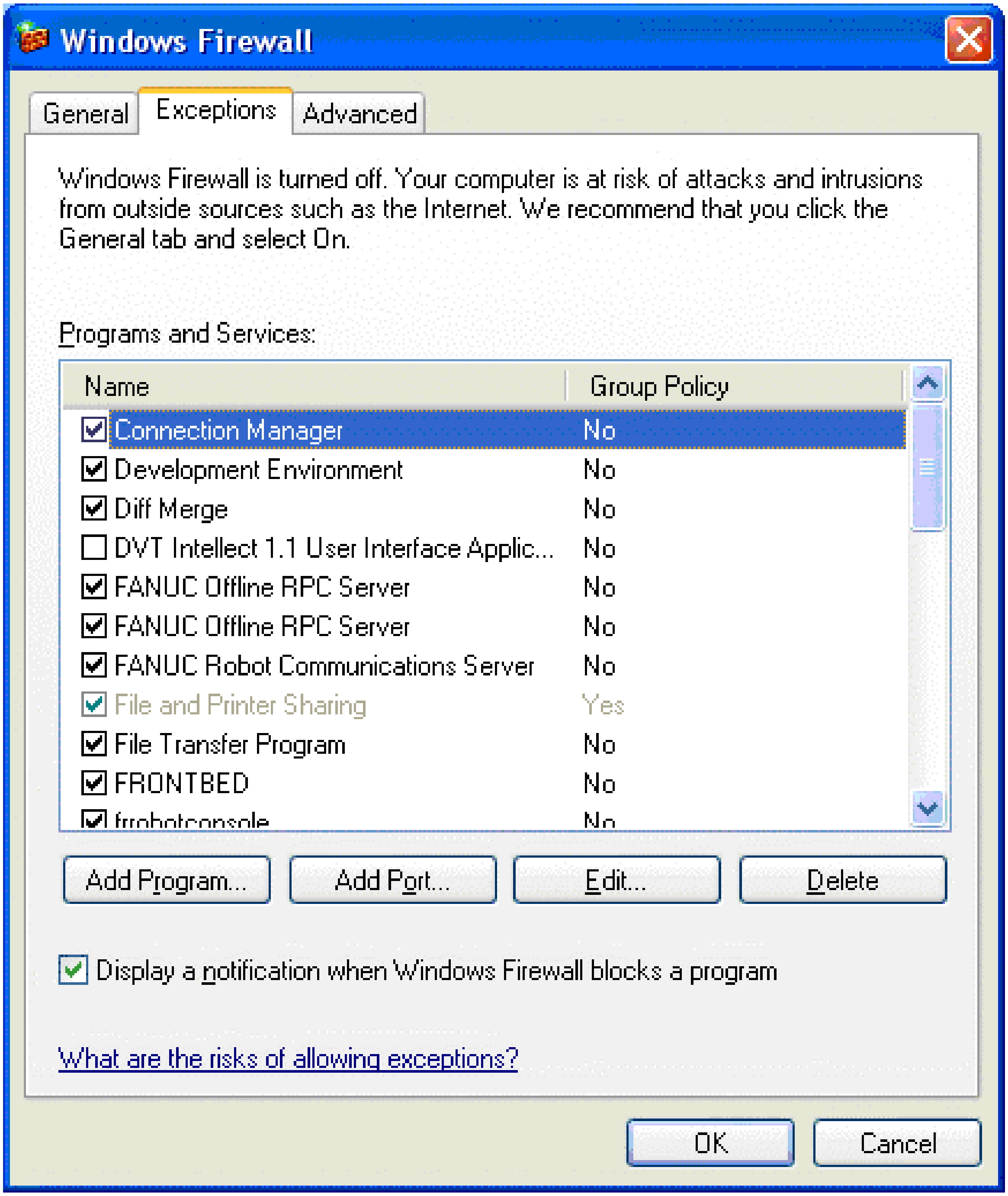
e. Click the [OK] button to close the window.

a. In Windows XP or Windows Vista,

a. In the Control Panel window, open [Windows Firewall].

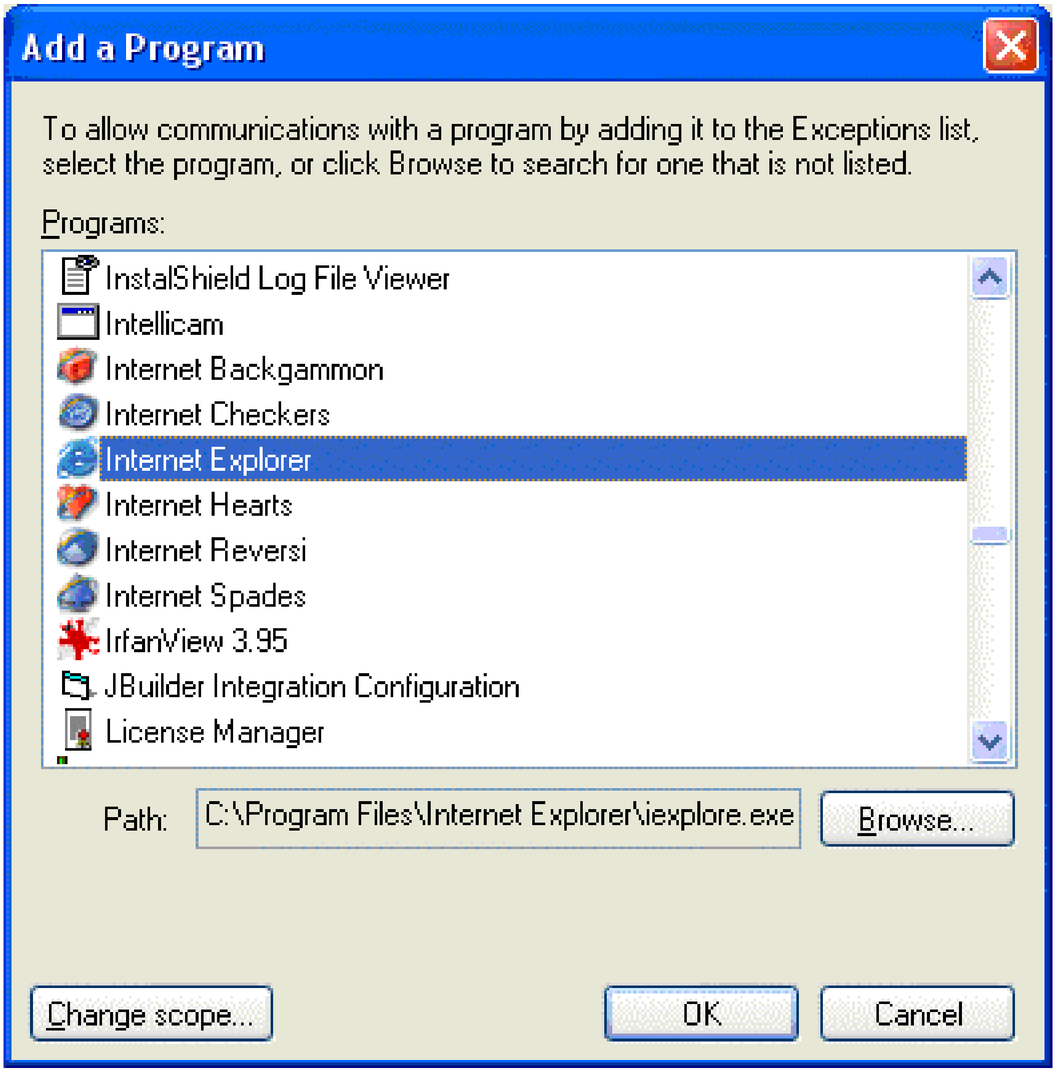
b. Click the [Exceptions] tab.

Figure 18–12. Exceptions



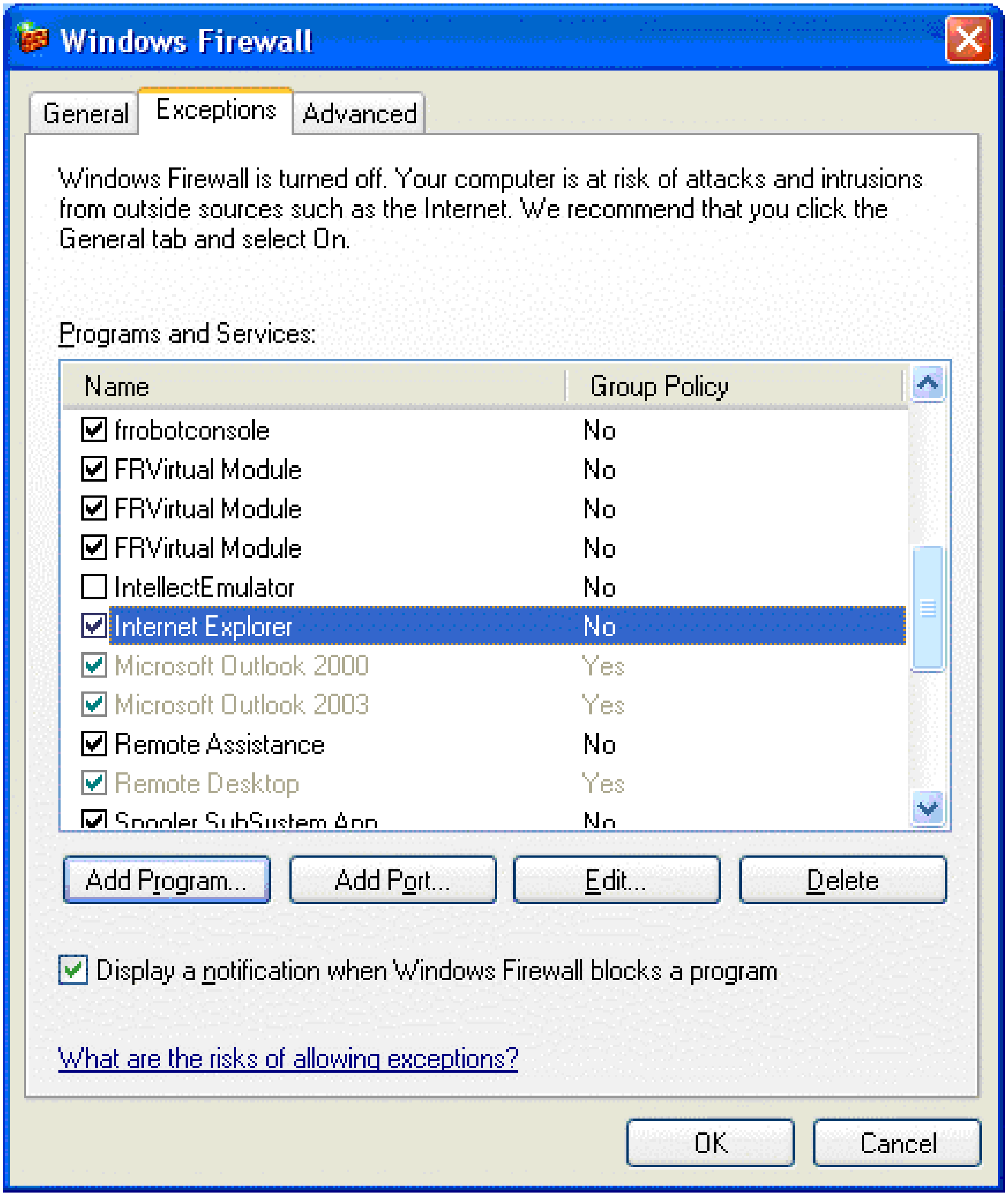
c. Click the [Add Program] button.

Figure 18–13. Add a Program



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

Figure 18–14. Exceptions



e. Click the [OK] button.

18.3.2.3 Testing the Network Connection

This section describes a method to verify that the network connection from the PC to the robot is configured correctly and operational.

Procedure 18-3 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 18.3.2.2 above.
- The robot is turned on and is connected to a network that is accessible by the above PC.

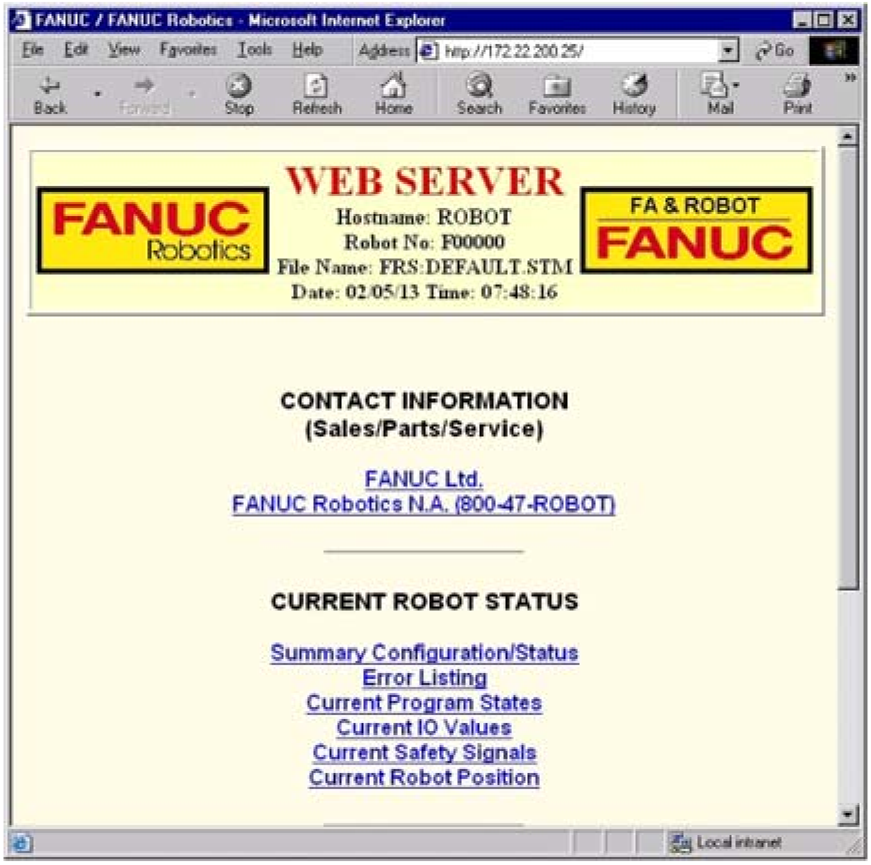
Steps

1. Bring up Internet Explorer on the PC.
2. In the Internet Explorer Address field, Enter “http://<myrobot_name_ or_address>”.

   Where <myrobot_name_ or_address> is either the DNS name of your robot i.e. pderob111.frc.com) or the IP address of your robot. (for example 192.168.1.100)

If the connection is successfully made you will see the HOME page of the robot displayed in Internet Explorer. It will be similar to that shown in Figure 18–15 .

Figure 18–15. Robot HOME page



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

18.3.3 Operation

After you have properly configured Microsoft® Internet Explorer and verified that you can connect to the robot as detailed above you can now access the Remote *i*Pendant screen. The following sections detail the procedure required for this and the limitations of this feature.

18.3.3.1 Remotely Monitoring the *i*Pendant

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

Procedure 18-4 Remotely Monitoring the *i* Pendant

Conditions

- The PC is connected to a network that can be used to access the robot.
- The PC has Microsoft® Internet Explorer 5.5 or greater loaded and is properly configured as detailed in Section 18.3.2.2 above.
- The PC has the *i*Pendant Controls installed as detailed in Procedure 18-1 .
- The robot is turned on and connected to a network that is accessible by the above PC.
- The robot has a functional *i*Pendant connected and operational.

Steps

1. Bring up Internet Explorer on the PC.
2. In the Internet Explorer Address field, Enter “http://<myrobot_name_ or_address>” to access the robot HOME page.

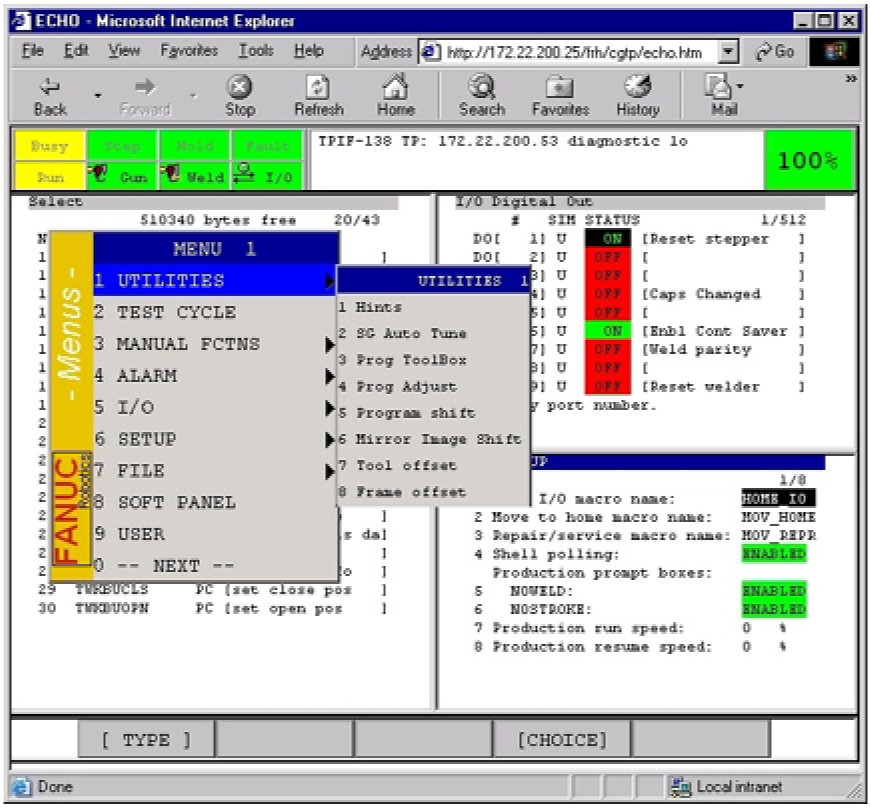
   Where <myrobot_name_ or_address> is either the DNS name of your robot (i.e. pderob111.frc.com) or the IP address of your robot. (i.e. 192.168.1.100)
3. From the HOME page, select Monitor iPendant (ECHO).

If the connection is successfully made you will briefly see the LOGIN screen, similar to that shown in Figure 18–16 , displayed in Internet Explorer.

Figure 18–16. Remote *i*Pendant LOGON Screen

After the LOGON Screen, you will see a remote display of the current *i* Pendant screen. The display will show the current *i*Pendant screen and any activity that might be occurring on it (popup menus, dynamic data, screen reconfigurations, for example). It might be similar to that shown in Figure 18–17 .

Figure 18–17. Remote *i* Pendant Monitoring Screen



18.3.3.2 Troubleshooting Remote Connection

Tips for troubleshooting when the remote *i* Pendant connection does not work.

- Try adding your robot controller IP address as a trusted site in Internet Explorer. Then connect using the IP address instead of the hostname.
- If there is a firewall between the robot controller and the PC, the robot controller must be able to open a TCP connection to port 60005 on the PC. The PC must also be able to open a TCP connection to port 3002 and a UDP connection to port 60004 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.

18.3.4 Limitations

Remote monitoring provides the user with the capability to display and monitor the current *i*Pendant screens and operations on a PC using Microsoft® Internet Explorer.

The following limitations apply to the remotely displayed *i*Pendant screen during the remote monitoring operation:

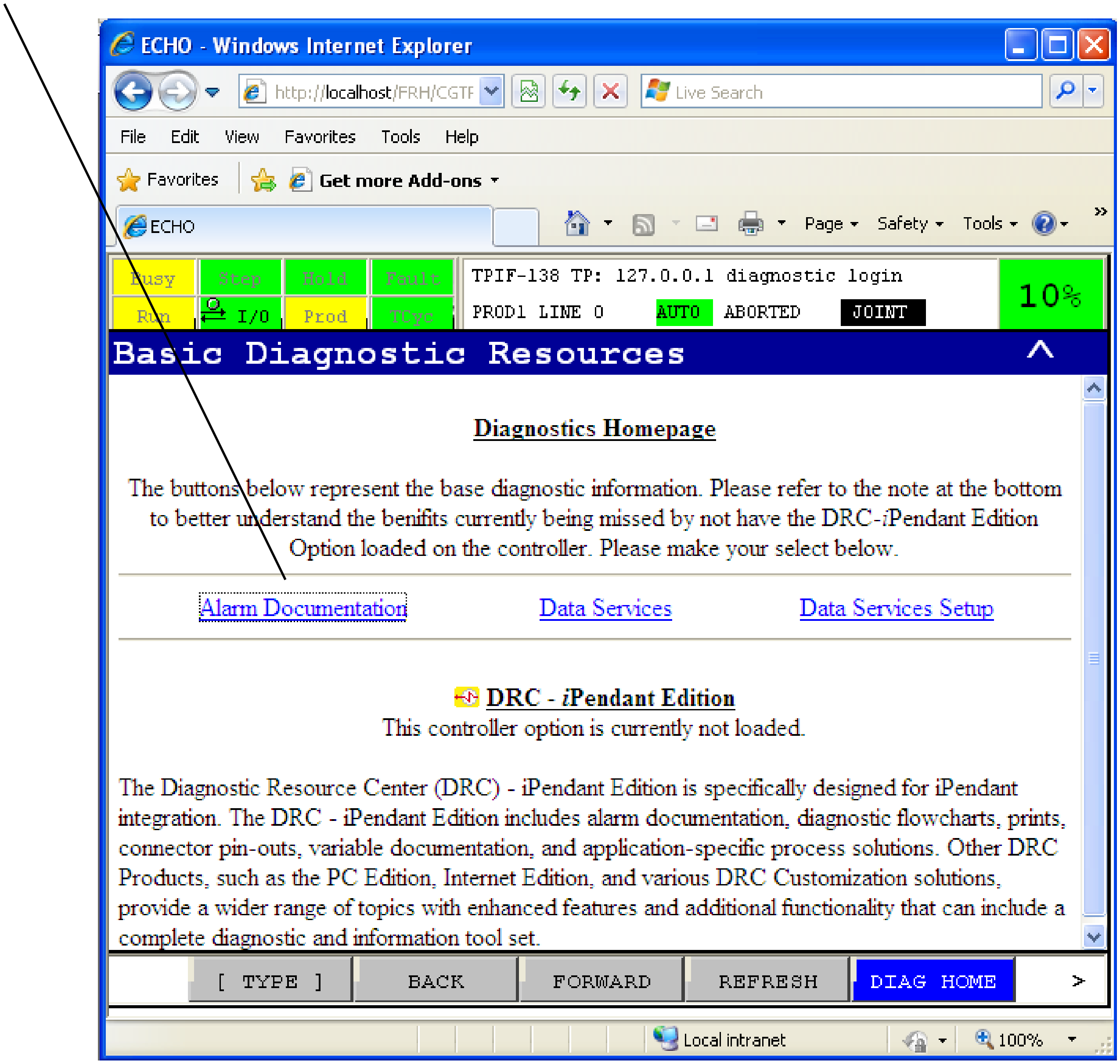
- It is meant as a DISPLAY ONLY mode, as such the remote connection normally cannot interact with the screens or affect the operation of the *i*Pendant or robot controller except in the following cases:

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

- The remote display will look similar to the *i*Pendant but will not be an exact duplicate. Fonts, Character sizes, colors, images, and overall format might be different.
- If a popup menu is being displayed on the *i*Pendant when the remote monitoring connection is made to the robot controller, that popup menu will not appear on the initial remote display. Subsequent pop-ups will be displayed on the remote screen as they occur on the *i*Pendant.
- Highlighted“Links” on a web page, a custom screen, a Help or Diagnostics screen, being displayed on the *i*Pendant, are not highlighted on the remote screen as shown in Figure 18–18 .

Figure 18–18. Highlighted Link

Remote *i*Pendant with Highlighted Link



18.4 REMOTE OPERATION

18.4.1 Overview

Remote Operation allows you to display an *i*Pendant-like screen on a PC using Microsoft ® Internet Explorer. The remote user can configure the remote display, navigate the controller menus and screens, and enter data remotely. Remote Operations are only available if the Internet Connectivity and Connection option is loaded.

Warning

Remote Operation is completely independent from the operation of the actual *i*Pendant therefore it can affect the operation the robot controller. Extreme caution must be exercised when using this feature.

Note The remote operator cannot see what is currently on the actual *i*Pendant or what the operator is doing nor can the *i*Pendant operator see what the remote operator is doing or what screens are being displayed on the remote PC.

For information on the limitations of this feature, refer to Section 18.4.4 .

The following functions can be performed on the remote *i*Pendant operation display:

- All *i* Pendant screens available from the MENUS and/or [TYPE] keys can be displayed as well as any custom screens and HELP or DIAGNOSTIC Screens.
- All popup menus, and windows.
- Multiple window configurations (i.e. Double and Triple modes on the *i* Pendant).
- Any input from the *i*Pendant numeric keypad, Function keys or cursor movement.

18.4.2 Setup

18.4.2.1 Requirements

The following are the requirements for remote operation of the *i*Pendant screens.

- PC must have Microsoft® Internet Explorer 5.5 or greater installed.
- PC must have the iPendant Controls installed. Refer to Procedure 18-1 for installation instructions.
- PC must be connected to a network , and be properly configured to allow a TCP/IP connection to the Robot Controller with the *i*Pendant connected.

- The robot controller must be connected to a network and be properly configured for Network access to the above PC.
- The robot controller must have the Internet Protocol Connectivity and Customization (IPCC) Option installed.
- The HTTP Authentication must be properly configured on the robot to allow iPendant access. Refer to the HTTP Authentication section in this manual for configuration information.

18.4.2.2 Configuring Microsoft® Internet Explorer

See Section 18.3.2.2 for information on configuring Microsoft® Internet Explorer.

18.4.3 Remote *i*Pendant Operation

After you have properly configured Internet Explorer ™ and verified that you can connect to the robot, you can now display the remote *i*Pendant Operation screen. The following sections detail the operation and the limitations of this feature.

18.4.3.1 Connecting to the Controller

This section will describe the method to connect to the robot controller and display the remote *i*Pendant Operation screen.

Procedure 18-5 Remote *i*Pendant Connection

Steps

1. Bring up Internet Explorer on the PC.
2. In the Internet Explorer Address field, type the following: `http://<myrobot_name_ or_address>` to view the robot HOME page.

   Where <myrobot_name_ or_address> is either the DNS name of your robot (i.e. pderob111.frc.com) or the IP address of your robot (i.e. 192.168.1.100).
3. From the HOME page, select Navigate iPendant (CGTP). If it greyed out, then you do not have the Internet Protocol Connectivity and Customization (IPCC) option installed.
4. If you have the HTTP Authentication for the *i*Pendant set to AUTHORIZE, the prompt shown in Figure 18–19 will be displayed on the PC.

Figure 18–19. Remote *i* Pendant Operation Password Screen

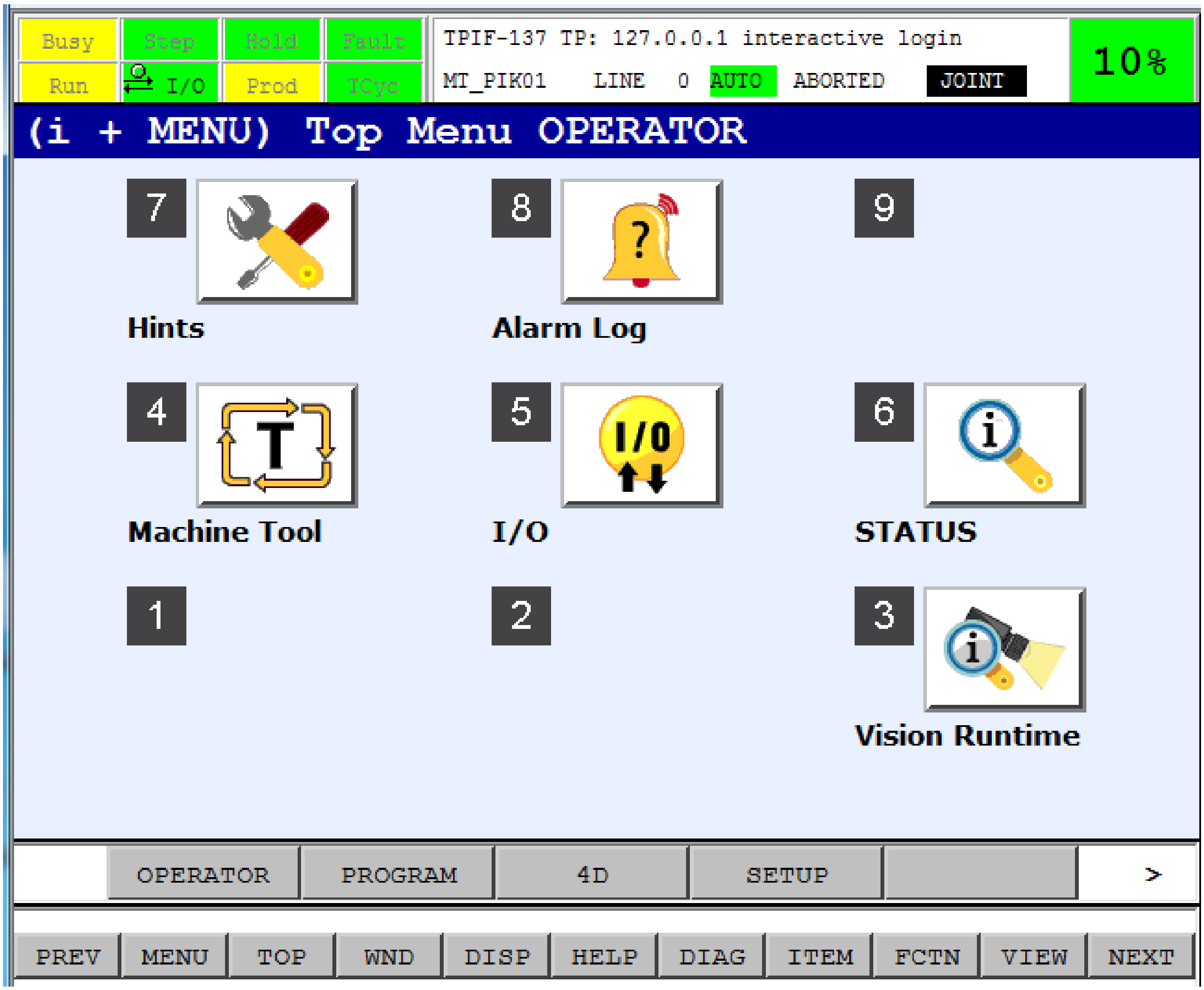
Type the USERNAME and PASSWORD that you set on the Robot Controller in the HOSTCOMM Setup>HTTP Authentication for the *i*Pendant.

If the connection is successfully made you will briefly see the LOGIN screen, similar to that shown in Figure 18–20 , displayed in Internet Explorer.

Figure 18–20. Remote *i* Pendant LOGON Screen

If this is the first time you are logging in, you will see an *i*Pendant screen similar to the one shown in Figure 18–21 . If you have logged in before you might see a different screen configuration, depending on how your controller was set up.

Figure 18–21. Remote iPendant Operation Screen



If the remote *i*Pendant connection failed, refer to Section 18.3.3.2 for troubleshooting tips.

18.4.3.2 Keys

While in the Remote Operation Mode you can use your mouse to select any of the available function keys or the auxiliary keys available below the *i*Pendant screen.

These auxiliary keys allow you access to certain functions available as Hard Keys on the actual *i*Pendant. Some of these keys are also mapped to keys on the PC keyboard as shown in Table 18–1 below:

Table 18–1. Remote Operation Key Mapping

***i*Pendant Function Key**	**Description**	**PC Key Mapping**
PREV	This key moves to Previous Screen or Cancels a pending operation	ESC
MENU	This key activates the MENU popup	F8
TOP	This key activates the TOP menu	
WND	This key changes the active window	
DISP	This key activates the DISPLAY Menu	
HELP	This key activates HELP for the current screen	
DIAG	This key activates DIAGNOSTICS for the current Alarm	
ITEM	Allows you to select an item on the screen by entering an Item Number	
FCTN	Activates the FUNCTION Popup Menu	F12
VIEW	Activates the Related View menu if *i* is displayed on the Focus Bar	
NEXT	Moves to the next page of softkeys if they are available.	F6
	Enters the SELECT menu	F9
	Enters the EDIT menu	F10
	Enters the DATA menu	F11
	Move the cursor in the appropriate direction	UP, DOWN, LEFT and RIGHT ARROWS
	Page Up and Page Down	SHIFT+UP Arrow, SHIFT+DOWN Arrow, PAGE UP and PAGE DOWN

In addition mouse events are supported in most screens as shown in Table 18–2 below:

Table 18–2. 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.
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 Cause & Remedy for an alarm.	Double mouse click on an alarm in the Alarm Log screen.
To display DETAIL for an alarm.	Right mouse click on an alarm in the Alarm Log screen.
Change the focus to another window.	Mouse click on the window. If a graphical menu is displayed, then mouse click on the Focus Bar for that window.
MENU	Mouse click on the left side of the Focus Bar
DISPLAY Menu	Mouse click on the center of the Focus Bar
FCTN	Mouse click on the right side of the Focus Bar
Related Views	Mouse click on the *i* shown in the Focus Bar
Maximize/Restore	Mouse click on the Maximize/Restore icon shown in the icon menu
Zoom	Mouse click on the ^ shown in the Focus Bar

For further information on navigating *i*Pendant screens and operating the *i*Pendant, refer to the *application-specific Setup and Operations Manual*.

18.4.3.3 Editing Guidelines

- LOOK/MONITOR mode is available in any window.
- Each window can have a unique default program.
- Selecting a teach pendant program from the SELECT screen in the window will cause that program to be the default program for that window.
- The window will use $UI_DEFPROG[3] to $UI_DEFPROG[8] based on the order of connection.
- The Status line will always show the default program for the left-hand window even if it does not have focus.
- The Editor title line shows the program that is being displayed.
- The program selected will be retained during cycle power. The current line number will not be retained during cycle power.
- The same program can be displayed in multiple windows. The cursor is independent.
- The Editor will not allow editing in the foreground. If you try to make a change to the program, then the warning “TPIF-069 Use background edit” is posted
- The program is open while in the EDIT screen; otherwise the program will be closed.
- Closing a window will cause the selected program to be closed.
- Disconnecting the Internet Explorer session will cause all the programs in all windows to be closed.
- FWD/BWD is not available from Internet Explorer

18.4.3.4 Background Editing Guidelines

- Background edit is supported in each window. There will be a background edit program unique for each window so multiple edit sessions will not affect each other.
- The background edit programs will be –BCKED4- through –BCKED9- based on the order of connection.
- On Internet Explorer, you have no control over what connection you get. However, if you time out and reconnect, the connection should be the same. If you edit at your desk and stay logged in, then go to the lab you will get a different connection and you will not be able to continue with the same background edit program while in the lab.
- When a program is selected for background edits, then the comment for the background edit program will show the selected program.
- The same program cannot be background edited in multiple windows. If you select the same program already being background edited in another window, then the warning “ TPIF-167

Program already in background edit” is posted. You need to End_edit the background session in the other window first.

- Background edit programs are retained during cycle power.
- The three windows within the same connection will share the same copy and paste buffer but they will not share with other Internet Explorer connections. This provides the ability to copy and paste from one teach pendant program to another.
- Each window has its own undo and redo buffers so multiple edit sessions will not affect each other.

18.4.4 Limitations

Remote operation allows you to display and monitor any *i*Pendant screen and perform many *i*Pendant operations on a PC using Internet Explorer ™ .

The following limitations apply to the Remote Operation of the *i*Pendant screen:

- You cannot make any changes through this connection if read only is enabled. To allow changes, set $UI_CONFIG.$READONLY[2]=FALSE. Refer to the application-specific *Setup and Operations Manual* for information on setting system variables.
- Many operations that are available on the actual *i*Pendant are not allowed through the remote monitoring function. These include:
  - Jogging the Robot
  - Running a Program using the FWD/BWD Keys
  - Changing COORD
  - Changing Speed Override
  - Foreground editing a program (Only background editing is allowed)
  - Functions associated with the application-specific keys on the *i*Pendant such as WIRE+/-, BACKUP, and so forth. Refer to the *Teach Pendant Keys* Section in the Preface of this manual for more information.
- Several of the *i*Pendant screens can only be displayed by a single device at any given time. The actual *i*Pendant always has precedence. If the *i*Pendant is currently displaying a screen that the Remote Operator has requested, and that screen can only be displayed in one place, the error “TPIF 110– Screen used by other device”, will be posted and the screen on the remote device will not be changed. Also, if the remote device is currently displaying a screen that is requested by the actual *i*Pendant operator, the remote display will be forced to the UTILITIES>HINTS screen. However, an error will not be posted.
- The remote display will look similar to the *i*Pendant but will not be an exact duplicate. Fonts, Character sizes, colors, images, and overall format might be different.

There is an inactivity timeout (in seconds) which will automatically logout the session. To turn off, set $UI_CONFIG.$TIMEOUT=0.

18.5 REMOTE FILE VIEWER

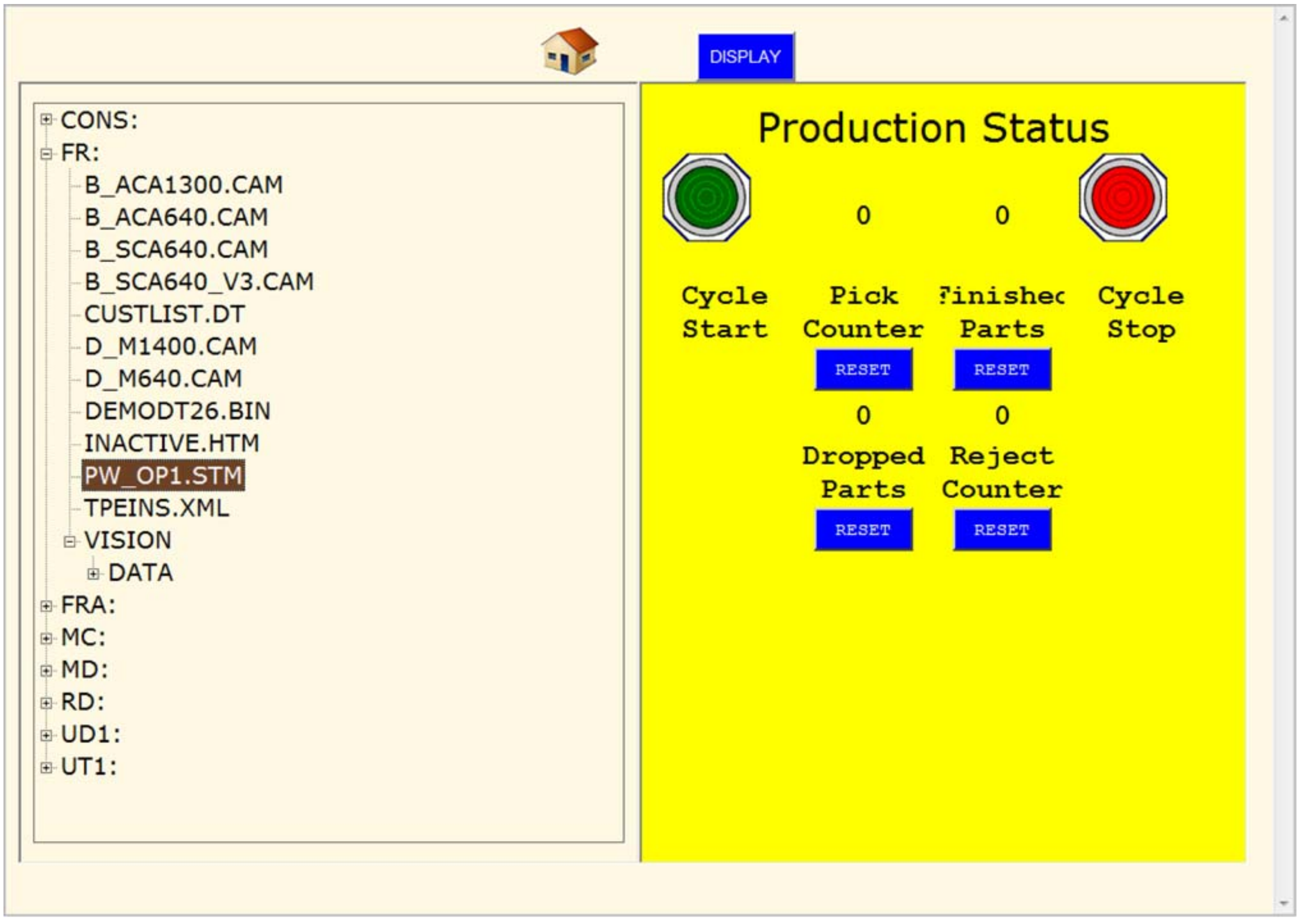
18.5.1 Overview

Remote File Viewer is used to navigate and display the devices, folders, and files on the robot controller. A classic tree view of folders and directories makes it easy to view files. However, you cannot perform any operations on the folders or files, such as copy or delete. This is a remote viewer only.

18.5.2 Operation

The Remote File Viewer requires the iPendant Controls be installed properly. Refer to REMOTE MONITORING Section 18.3.2 . After you have properly configured Microsoft® Internet Explorer and verified that you can connect to the robot you can now access the Remote File Viewer screen. From the HOME page, select TreeView/File link. You will see a screen similar to the one shown in Figure 18–22 .

Figure 18–22. TreeView/File



To view a file, cursor to a file on the left side and press the DISPLAY button at the top of the screen. If the file is readable text, an image, or a web page, it will be displayed on the right side. If the file is binary, you will be prompted to open or save the file from the robot. If the cursor is on a selection that is not accessible, such as a device or directory, then the right side will be blank after you press the DISPLAY button.

To return to the HOME page, press the HOME icon at the top of the screen.

18.6 JOG IN AUTO MODE

18.6.1 Overview

The Jog in Auto Mode option allows you to jog the robot remotely using a PC and Microsoft® Internet Explorer. No other software options on the robot controller are required. Jogging is only allowed when the teach pendant is disconnected from the robot and the robot is in Auto Mode.

Warning

E-Stop button is necessary near the PC that operates the robot. The operator using the PC should be in a place where it is possible to see the robot. Otherwise, you could injure personnel or damage equipment.

18.6.2 PC Jog IP Address Setup

The robot is pre-configured to allow a PC to connect with the IP Address of "192.168.0.100". If your PC has a different IP address, then you must change it on the robot using the teach pendant. If your teach pendant is already disconnected, then use the Remote *i*Pendant Operation described in Section 18.4 . Use Procedure 18-6 to set up the PC Jog IP Address.

Procedure 18-6 Setting up the PC Jog IP Address

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

```
SETUP Protocols
Protocol          Description
1  TCP/IP             TCP/IP Detailed Setup
2  FTP                File Transfer Protocol
3  NONE               Connects tag to port
```

5. Make sure TCP/IP is selected.
6. Press F3, DETAIL. You will see a screen similar to the following.

```
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
PC Jog IP addr:             192.168.0.100
Host Name (LOCAL)       Internet Address
     1  *********       ******************
     2  *********       ******************
     3  **********      ******************
```

7. Move the cursor to PC Jog IP addr and enter the IP address of the PC. If PC Jog IP addr is not shown, then the Jog in Auto Mode option is not loaded. If this is the case, contact your FANUC Representative to purchase the Jog in Auto Mode option.

18.6.3 Setting Up *i*Pendant Controls

The following are the requirements for Jog in Auto Mode. The Jog in Auto Mode option requires you to have the FANUC America Corporation *i*Pendant Controls loaded on the PC that you will be using. Refer to Section 18.2 .

18.6.3.1 Requirements

Refer to Section 18.3.2.1 for the requirements for Jog in Auto Mode option

18.6.3.2 Configuring Microsoft® Internet Explorer

Refer to Section 18.3.2.2 for information on configuring Microsoft® Internet Explorer.

18.6.3.3 Testing the Network Connection

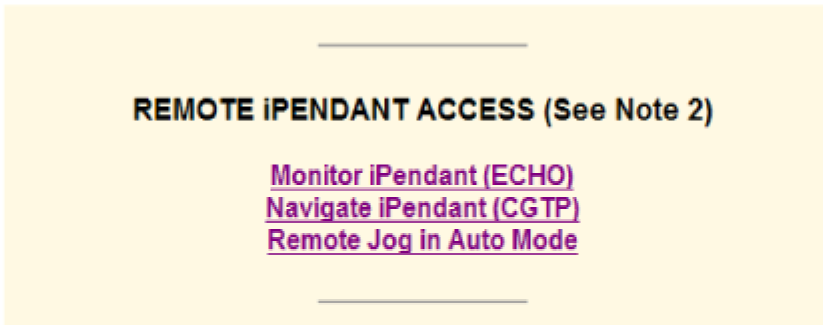
Refer to Section 18.3.2.3 for information on testing the network connection.

18.6.4 Remote Jog in Auto Mode

After you have properly configured Internet Explorer and have verified that you can connect to the robot, you can now display the Remote Jog in Auto Mode screen. The following sections detail the operation of this feature.

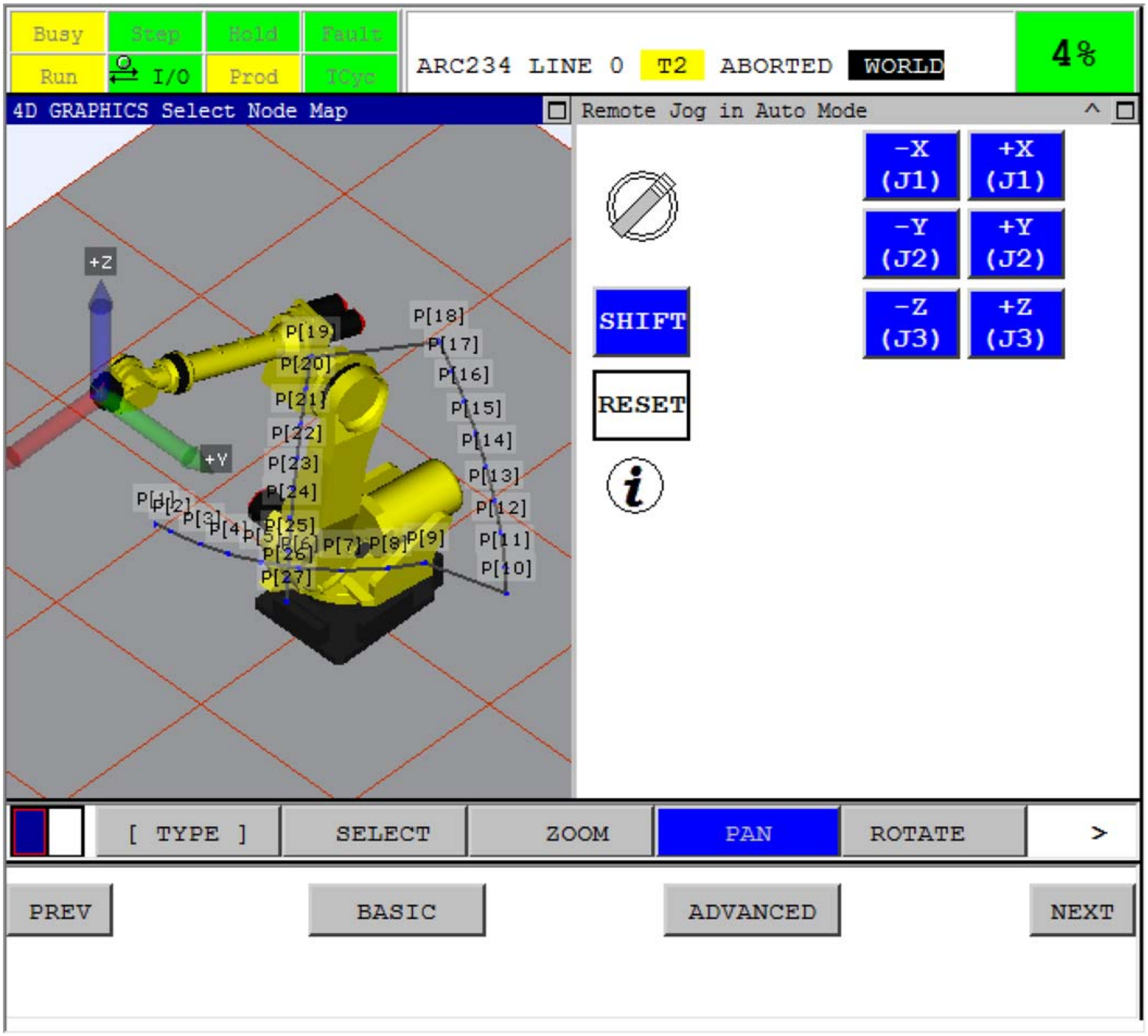
From the robot HOME page, select Remote Jog in Auto Mode. See Figure 18–23 for an example.

Figure 18–23. Remote iPendant Access



If the connection is successfully made you will briefly see the LOGIN screen displayed in a new Internet Explorer window. Then you will see a screen similar to the one shown in Figure 18–24 .

Figure 18–24. Remote Jog in Auto Mode Screen



The 4D GRAPHICS Display of the robot will be shown in the left pane. 4D requires the PC have an advanced graphics card capable of displaying OpenGL 2.0. If the 4D Graphics option is loaded, then the [TYPE] key can be used to select additional 4D items to display such as 4D Select Node Map.

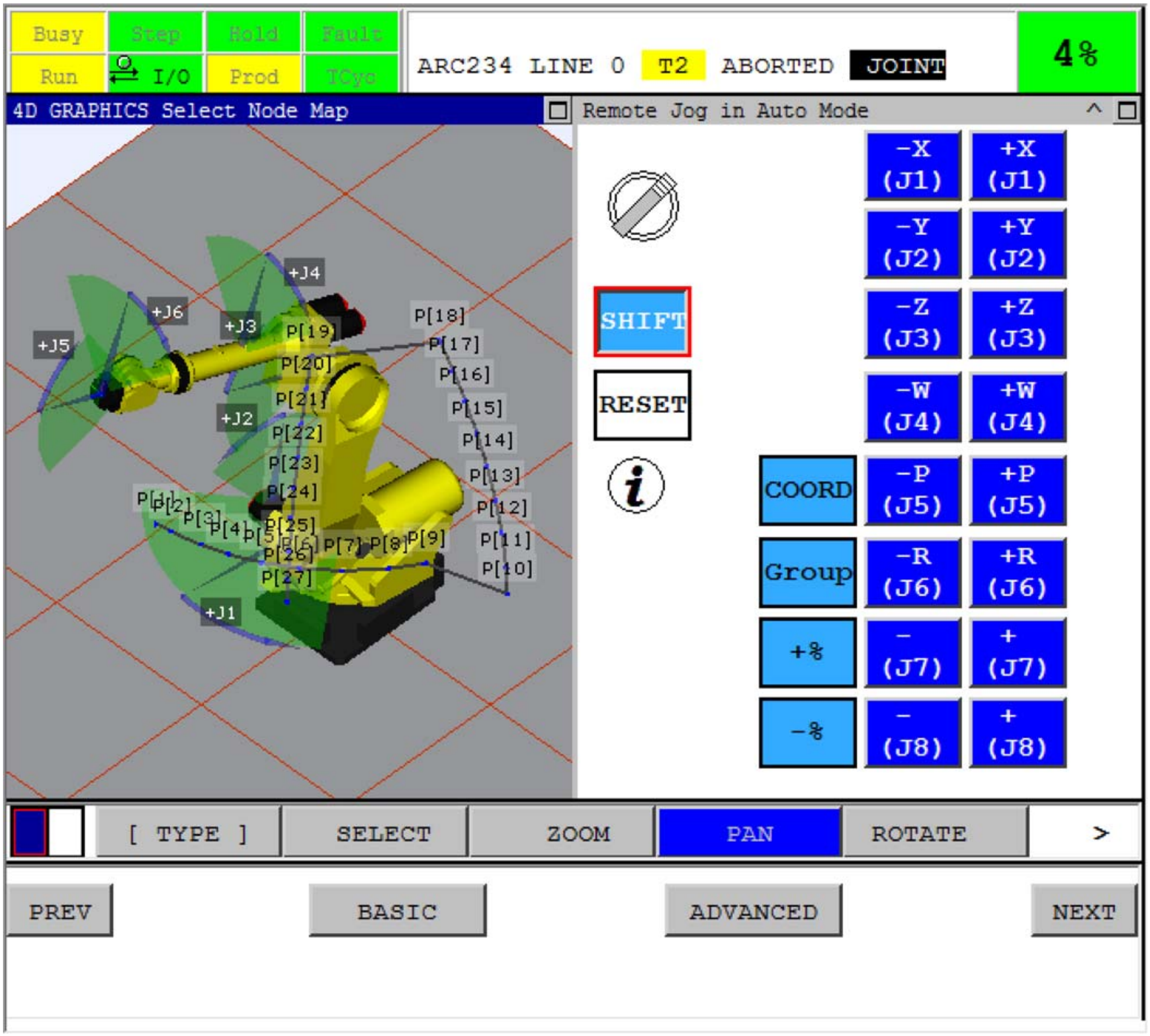
The Remote Jog in Auto Mode screen will be displayed in the right pane. In this screen, you can use your mouse to select any of the buttons or function keys. These buttons allow you access to certain functions available as Hard Keys on the actual *i*Pendant as shown in Table 18–3 .

Table 18–3. Remote Jog Key Mapping

Button	Description
Switch	Remote Jog Enable switch. The switch must be ON in order to jog.
X,Y,Z	Jog buttons. When button is pressed jogging will start. When button is released jogging will stop.
SHIFT	This button must be pressed in order to jog. It is also used for SHIFT-RESET to reset after a DCS or collision guard alarm.
RESET	This button resets the active alarm.
ⓘ + Jog Buttons	Shows you the direction that the robot will move when the jog button is pressed. Only works if SHIFT is released.
ⓘ + COORD	Toggles jog indicators on and off. Only works if SHIFT is released and Enable switch is ON.
ADVANCED	This button displays the advanced buttons.

When the ADVANCED button is pressed, you will see a screen similar to the one shown in Figure 18–25 .

Figure 18–25. Advanced Remote Jog in Auto Mode



While in the Advanced Remote Jog in Auto Mode screen you have additional buttons as shown in Table 18–4 .

Table 18–4. Advanced Remote Jog Key Mapping

Button	Description
W,P,R	Jog buttons. When button is pressed jogging will start. When button is released jogging will stop.

Table 18–4. Advanced Remote Jog Key Mapping (Cont'd)

Button	Description
COORD	Each press changes the jog coordinate system.
SHIFT-COORD	This button brings up the jog menu where you can check and change the frame number of each frame (TOOL, JOG, USER), the jog group number, and the sub-group type (ROBOT,EXT).
GROUP	The GROUP key is used to switch groups.
+%, -%	Jog speed buttons used to increment or decrement the jog speed. Use with SHIFT to increment and decrement in larger increments.
BASIC	This button removes the advanced buttons. The jog speed is reset to 10%. The jog group is reset to 1. The jog coordinate system is reset to WORLD.

Procedure 18-7 Jogging the Robot in AUTO Mode

Conditions

- The Mode Select switch on the front panel of the robot controller must be in AUTO mode.
- The fence circuit must be closed for the robot to move in AUTO mode.
- All personnel and unnecessary equipment are out of the workcell.
- All EMERGENCY STOP faults have been cleared.
- All other faults have been cleared and the fault light is not illuminated.

> **Warning**
>
> **Make certain that all safety requirements for your workplace have been followed; otherwise, you could injure personnel or damage equipment.**

- The PC must remain connected to the robot without interference from network traffic. An internal watchdog is implemented so if the connection is broken between the PC and the robot, the robot will stop motion. The watchdog is automatically sent every 100 ms while a jog button is pressed.

1. Press the Remote Jog enable switch to the ON position and press RESET.

> **Warning**
>
> **In the next step, the robot will move. To stop the robot immediately any time during jogging, release the mouse button.**

2. To jog, continuously press the left mouse button over the jog button that corresponds to the direction in which you want to move the robot. To stop jogging, release the left mouse button.
3. When you are finished jogging, press the Remote Jog ON/OFF switch to OFF.

18.7 PC REMOTE iPENDANT

18.7.1 Overview

The PC Remote iPendant option allows you to connect as the iPendant and jog the robot remotely using a PC and Microsoft® Internet Explorer. The connection is only allowed when the robot is in Auto Mode. The teach pendant, if disabled, is logically disconnected from the robot. It also allows Remote Operation with no limitations. Some of the operations available by PC Remote iPendant function are listed below:

- View all menus and change settings.
- Foreground editing a program.
- Change I/O status.
- Change register, position register, string register.
- Jogging the Robot.
- Running a robot using the FWD/BWD keys.
- Changing COORD.
- Changing Speed Override.
- Clearing DCS alarm.
- Applying DCS parameter.

The following limitations apply to the PC Remote iPendant function.

- PC Remote iPendant connection is refused when the TP enable key is enabled on the Real iPendant.
- PC Remote iPendant connection is refused when the operation mode is not AUTO mode.
- Mode switch must be AUTO mode while PC Remote iPendant is connected. If the mode switch is changed from AUTO to T1 or T2, the PC is logged out immediately.
- Real iPendant has mastership, therefore PC Remote iPendant connection is closed when Real iPendant connects.
- Multiple PC Remote iPendant connections are not allowed.

Warning

E-Stop button is necessary near the PC that operates the robot. The operator using the PC should be in a place where it is possible to see the robot. Otherwise, you could injure personnel or damage equipment.

Warning

The robot operates same as T2 (100%): Test Mode 2. During program test run, full program speed is allowed, and the override can be changed from low to 100%.

18.7.2 Setup

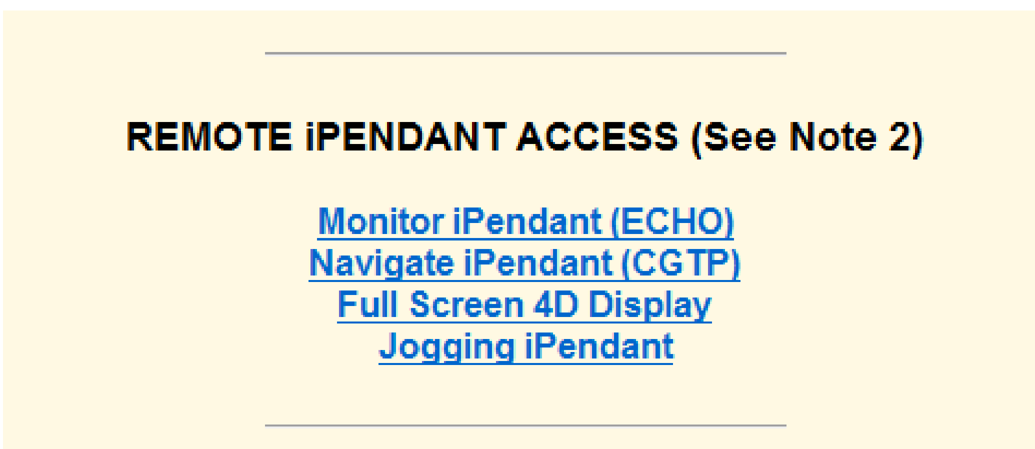
Refer to Section 18.6.2 and Section 18.6.3 to set up the PC Remote *i*Pendant option.

18.7.3 PC Remote iPendant

After you have properly configured Internet Explorer and have verified that you can connect to the robot, you can now display the Remote iPendant screen. The following sections detail the operation of this feature.

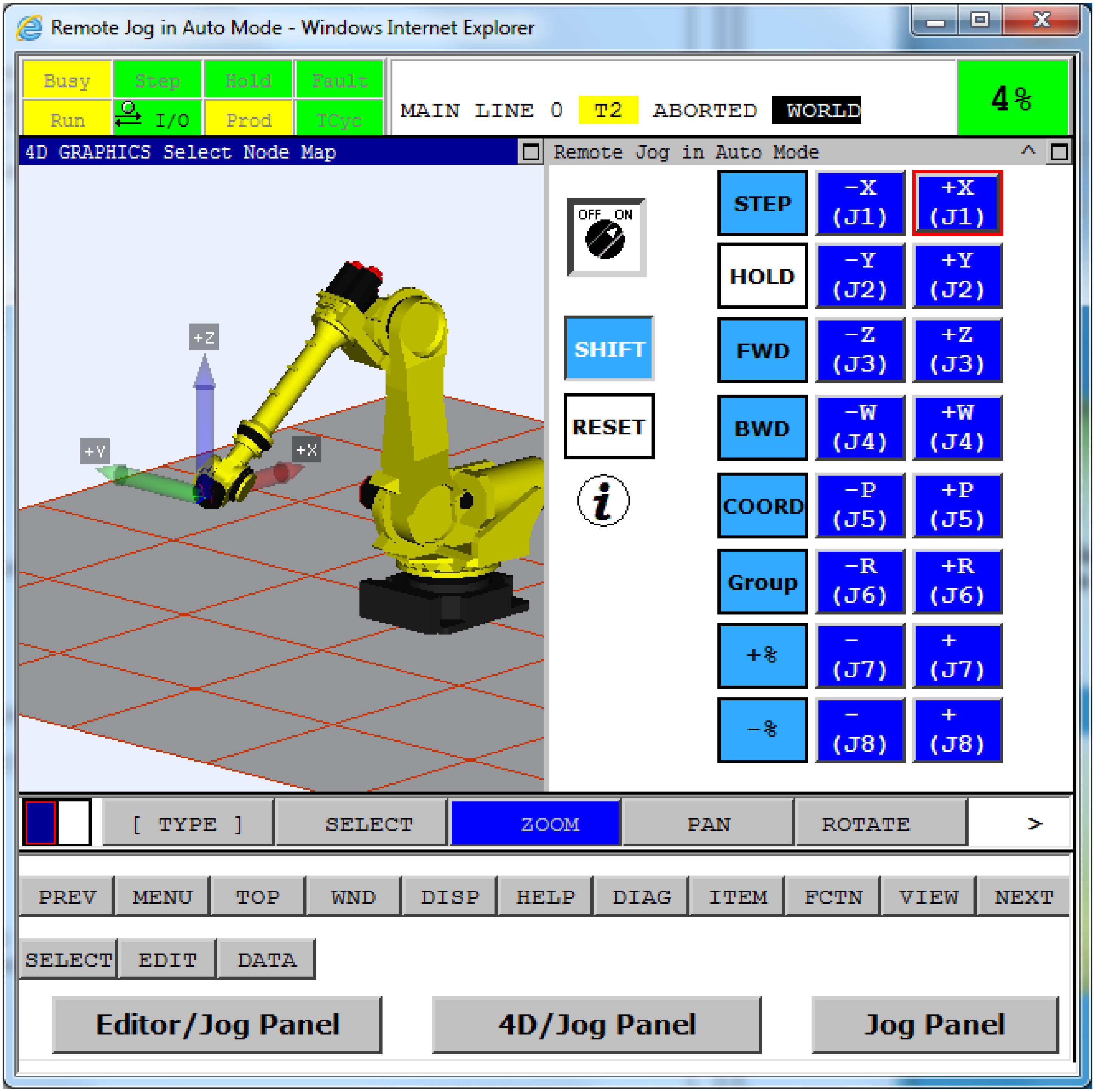
From the robot HOME page, select Jogging iPendant. See Figure 18–26 for an example.

Figure 18–26. Remote iPendant Access



If the connection is successfully made you will briefly see the LOGIN screen displayed in a new Internet Explorer window. Then you will see a screen similar to the one shown in Figure 18–27 .

Figure 18–27. PC Remote iPendant Screen



The 4D GRAPHICS Display of the robot will be shown in the left pane. 4D requires the PC have an advanced graphics card capable of displaying OpenGL 2.0. If the 4D Graphics option is loaded, then the [TYPE] key can be used to select additional 4D items to display such as 4D Select Node Map.

The Jog Panel will be displayed in the right pane. In this screen, you can use your mouse to select any of the buttons. These buttons allow you access to functions available as Hard Keys on the actual *i*Pendant. You have full control over the Remote iPendant using the buttons at the bottom of the screen.

If you have Remote iPendant option loaded, but now you want to restrict to only Remote Jog in Auto Mode functionality, you can set $JOG_IN_AUTO = 0 and reselect the "Jogging iPendant" link from the HOME page. You will be shown 4D with the BASIC and ADVANCED jog screens. You cannot change the screen.

When you want to allow all iPendant functionality, you can set $JOG_IN_AUTO = 1 and reselect the "Jogging iPendant" link from the HOME page.

18.8 REMOTE iPENDANT

18.8.1 Overview

The Remote iPendant option (R843) provides the following 3 features for HMI devices:

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

These links are available from the HOME page of the robot. They are similar to other options listed in this appendix, namely "REMOTE MONITORING (ECHO)", "REMOTE OPERATION (CGTP)", and "PC REMOTE iPENDANT (ITP)". The difference is that the HMI links are implemented using JavaScript so they do not require iPendant Controls. Any modern browser on any HMI device should be able to display and connect to the robot using the HMI links.

The following limitations apply to all Remote iPendant links for HMI devices:

- User custom screens that incorporate iPendant Controls cannot be displayed. This will exclude any graphical screens such as 4D, Vision Setup, Icon Editor, Panel Wizard, Interface Panel, PMC display, etc. If your HMI is a Windows PC with Internet Explorer and you have the iPendant Controls installed, then this limitation is removed.
- Popup menus and dialogs are not supported. HMI menus are used for MENU and [TYPE]. Display and Function menus have been rewritten in JavaScript. Other popup menus will use the legacy Teach Pendant style.

The following limitations apply to HMI Jogging iPendant.

- Jogging iPendant is refused when the TP enable key is enabled on the Real iPendant.
- Jogging iPendant is refused when the operation mode is not AUTO mode.
- Mode switch must be AUTO mode while Jogging iPendant is connected. If the mode switch is changed from AUTO to T1 or T2, the HMI is logged out immediately.
- Real iPendant has mastership, therefore Jogging iPendant connection is closed when Real iPendant connects.
- Multiple Jogging iPendant connections are not allowed.

Warning

E-Stop button is necessary near the HMI that operates the robot. The operator using the HMI should be in a place where it is possible to see the robot. Otherwise, you could injure personnel or damage equipment.

Warning

The robot operates same as T2 (100%): Test Mode 2. During program test run, full program speed is allowed, and the override can be changed from low to 100%.

18.8.2 Setup

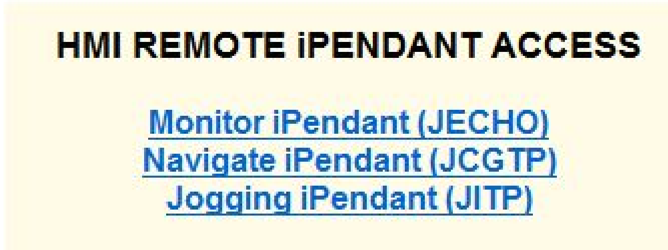
Your HMI device requires a browser. For Microsoft devices, Internet Explorer is recommended. For all other devices, Google Chrome is recommended. You do not need to install iPendant Controls.

The browser will connect to the robot's web server using standard port 80 for Monitor iPendant (JECHO) and Navigate iPendant (JCGTP). Typically browser security is not an issue when using port 80.

Jogging iPendant (JITP) will use non-standard ports 3080 and 4080. Many browsers and firewalls will prevent data on these ports. For Internet Explorer, you may have to follow the setup in Section 18.3.2.2 for a trusted site. A virtual private network (VPN) may be required to allow security on a private network.

Once the browser on the HMI is connected to the HOME page of the robot you will see the links as shown in Figure 18–28 .

Figure 18–28. HMI REMOTE iPENDANT ACCESS



The links will open a new window with default size of 1024x800. However, the pages are designed to fit into any size window. If you are using a WebBrowser Control, you may need to directly enter the links as shown in the table below:

Table 18–5. Link Address

FEATURE	LINK
JECHO	http://robot_ip_address/frh/jcgtp/echo.stm
JCGTP	http://robot_ip_address/frh/jcgtp/cgtp.stm
JITP	http://robot_ip_address:3080/frh/jcgtp/itp.stm

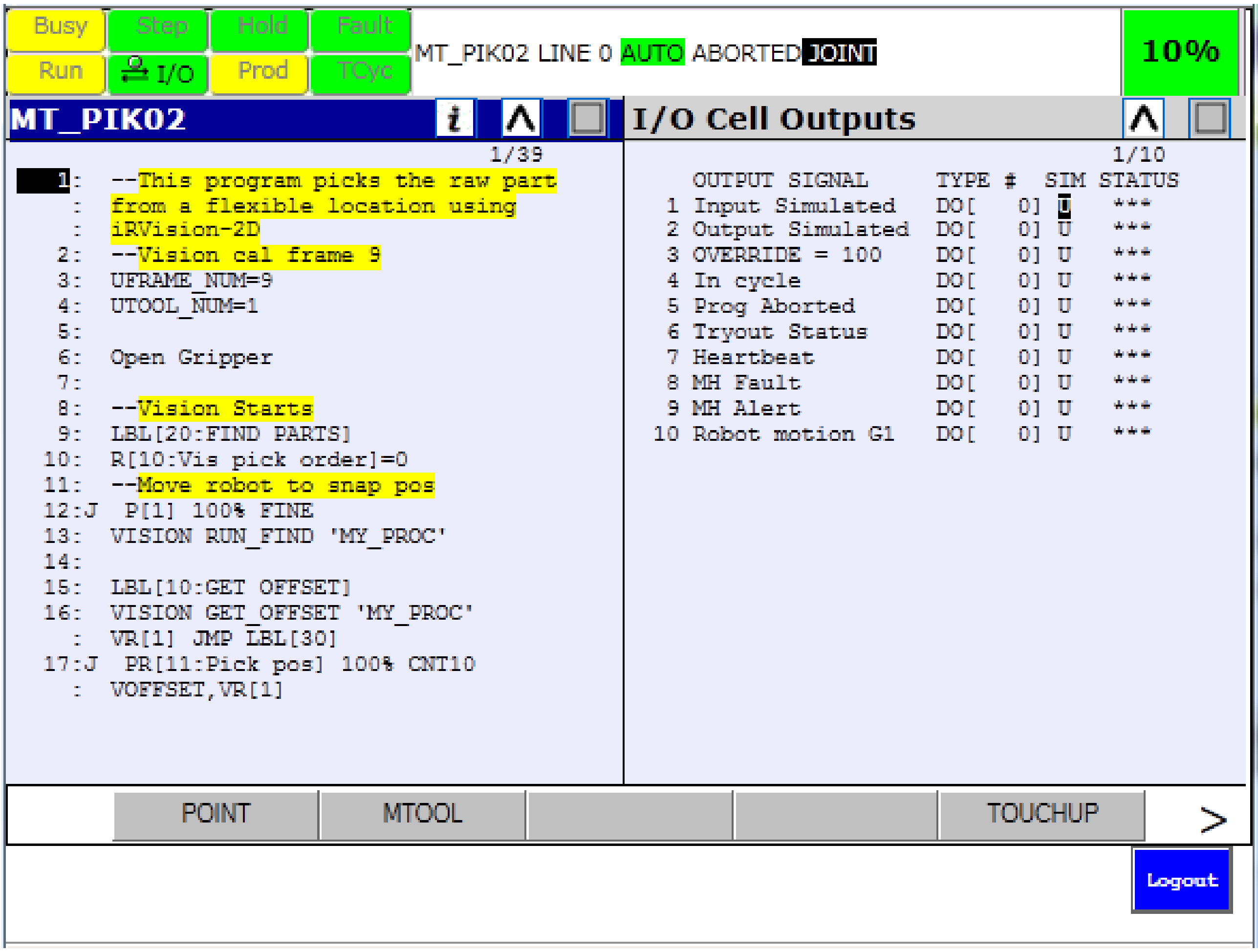
18.8.3 Operation

Press the Monitor iPendant (JECHO) to display and monitor the current iPendant screens and operations on your HMI device. It is meant as a DISPLAY ONLY mode and therefore, the remote connection cannot interact with the screens or affect the operation of the iPendant or robot controller. Occasionally the iPendant will use a screen that requires the iPendant Controls and your HMI device will be blank.

The Logout button is used to correctly logout your connection. Some devices do not close their network connections properly if you close the window without logging out.

Figure 18–29 .

Figure 18–29. Monitor iPendant (JECHO)

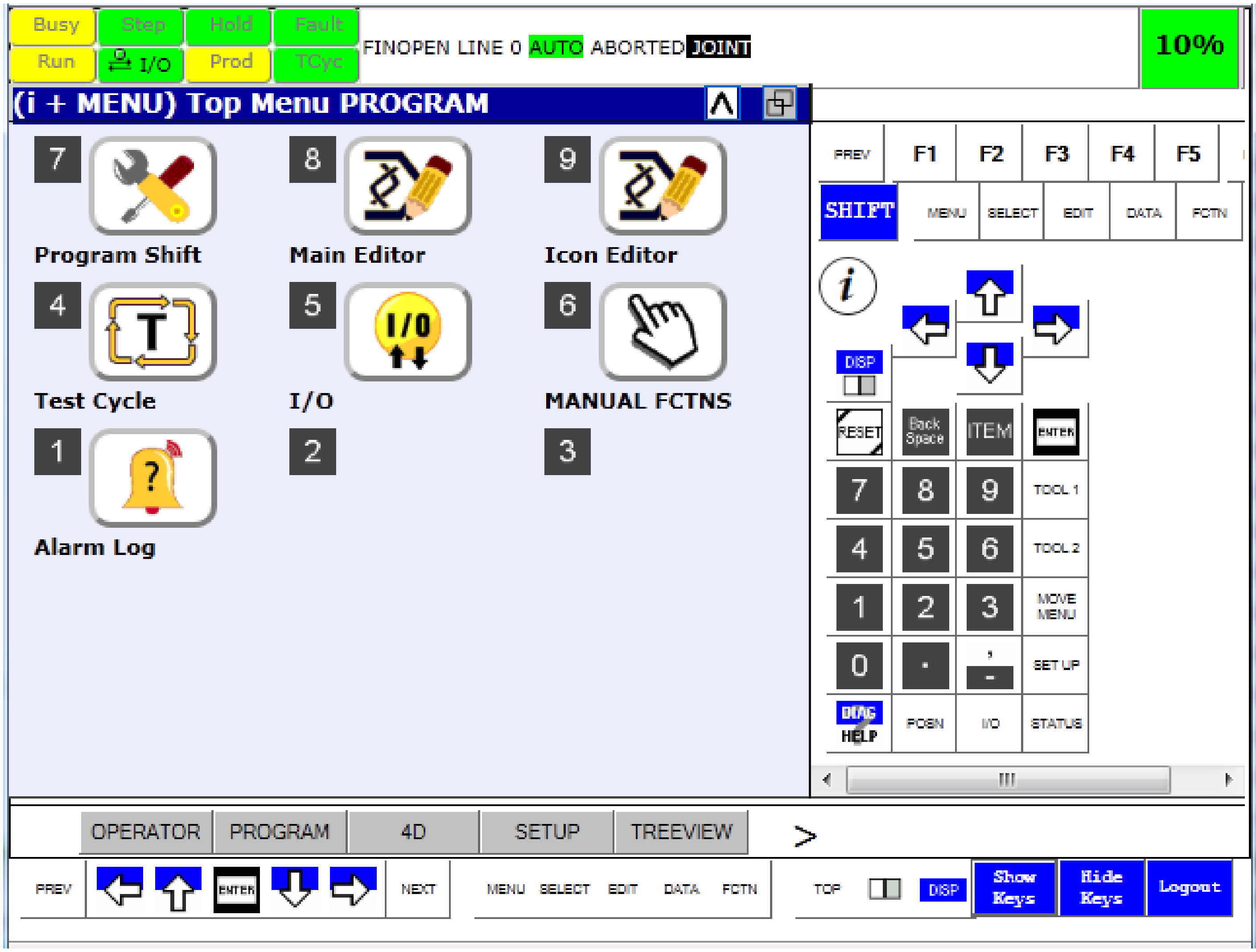


Press the Navigate iPendant (JCGTP) to display an iPendant-like screen on your HMI device. The remote user can configure the remote display, navigate the controller menus and screens, and enter data remotely. Menus that require the iPendant Controls cannot be displayed. Refer to Section 18.4.4 for futher limitations.

The Show Keys button on the lower right will bring up the iPendant softkeys excluding the motion keys. The Hide Keys button will hide the softkeys. The Logout button is used to correctly logout your connection.

Figure 18–30 .

Figure 18–30. Navigate iPendant (JCGTP)



The following conditions are required before using Jogging iPendant:

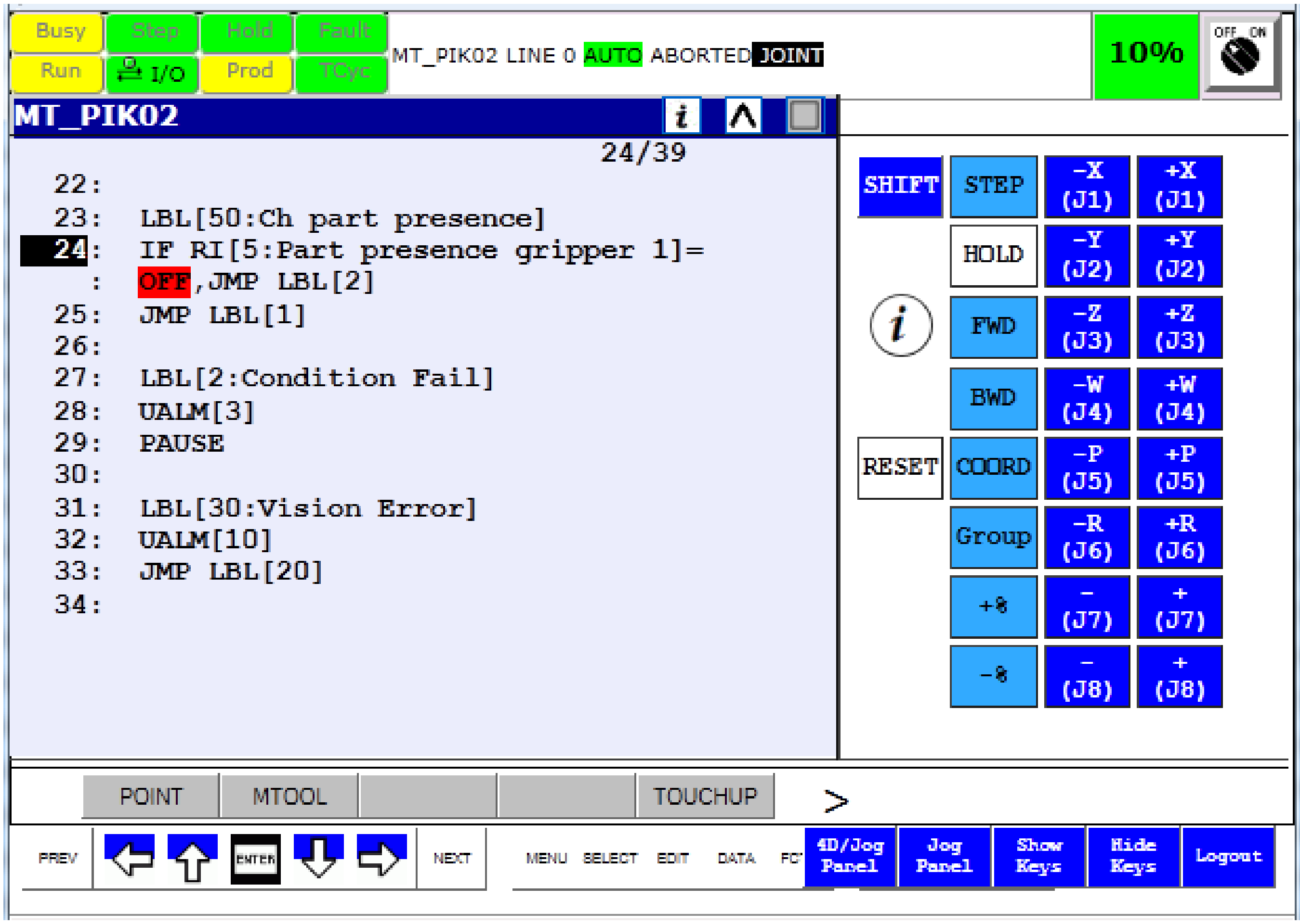
- On the Teach Pendant, set up the PC Jog IP address. Your HMI device will be considered the PC in this case. Refer to Section 18.6.2 .
- Robot must be in AUTO mode.
- Teach Pendant must be DISABLED. It can also be disconnected if that is desired.

Press the Jogging iPendant (JITP) to connect as the iPendant using your HMI device. The Teach Pendant, if still connected, will be logically disconnected from the robot. Jogging iPendant can jog the robot remotely. It also allows Remote Operation, such as running a program using the FWD/BWD softkeys, foreground editing a program, and clearing DCS alarms. Menus that require the iPendant Controls cannot be displayed.

The Show Keys button on the lower right will bring up the iPendant softkeys excluding the motion keys. The Jog Panel button will bring up the iPendant softkeys that control motion. The Hide Keys button will hide the softkeys. The Logout button is used to correctly logout your connection.

Figure 18–31 .

Figure 18–31. Jogging iPendant (JITP)



Note **While any of the Jog Panel keys are being pressed, the HMI device will send a watchdog to the controller. Depending on your communication speed, you may experience "SYST-066 Teach Pendant communication error". To avoid this error you may increase $UI_CONFIG.$JWDOG_TIMER. This timer in ms notifies the controller to wait additional time only when the remote device is sending the keys. It does not affect the real Teach Pendant. If your communication does go down, the controller will not shut down the keys until $JWDOG_TIMER has elapsed. Collision Guard and/or Dual Check Safety is recommended to protect your equipment.**

Appendix A

DIAGNOSTIC INFORMATION

Contents

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 . Details on auto-negotiating and manually setting speed and duplex level can be found in the “Setting up TCP/IP” chapter in the *Internet Options Setup and Operations Manual*. The robot will auto-negotiate by default and should not be changed in most cases. The speed and duplex used on either interface is shown under the STATUS screen for interface #1 or interface #2. To do this, Press MENU, 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 MENU.
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.

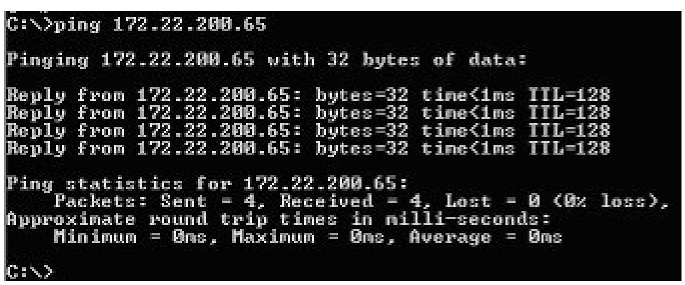
The prompt line on the teach pendant will indicate if the PING was successful, or if the PING request timed out.

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.

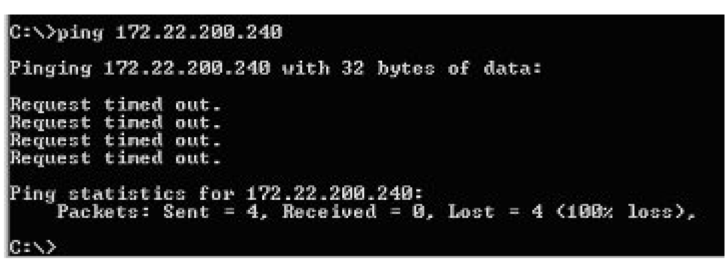
```
PING 192.168.0.10
```

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.



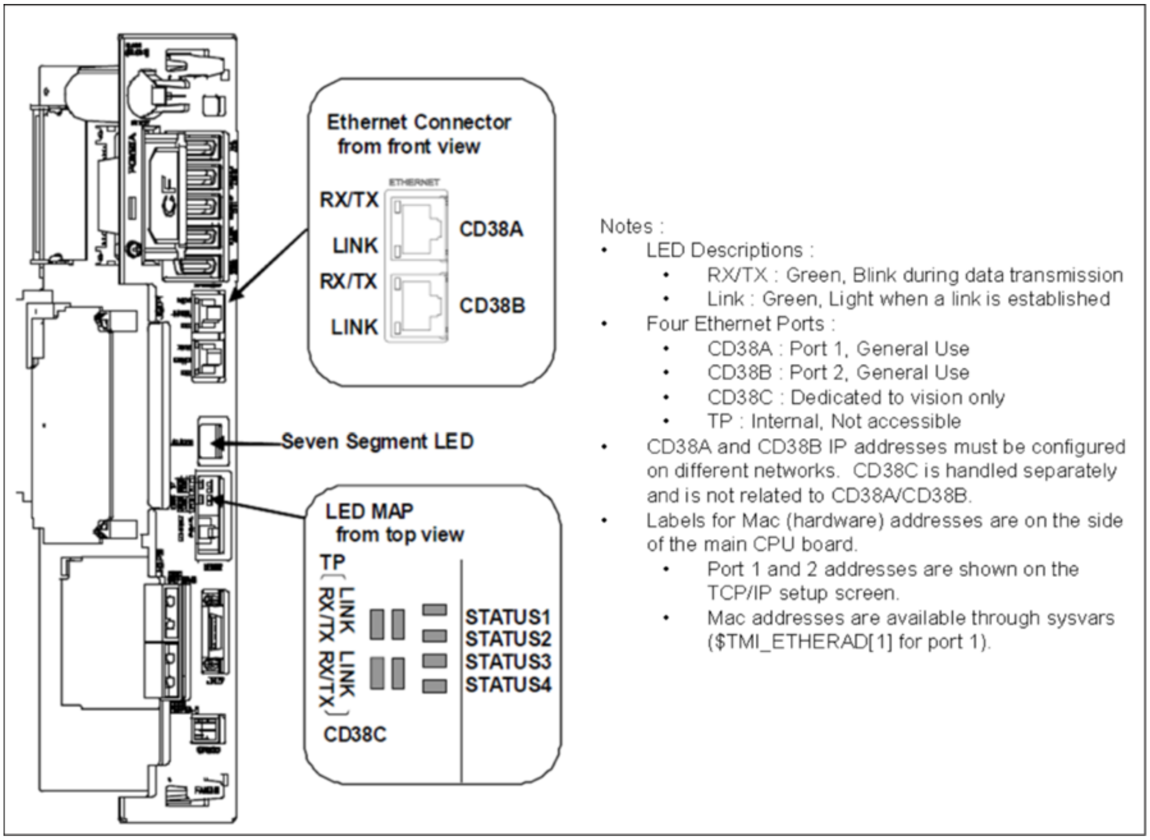
```
C:\>ping 172.22.200.240

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 in the *Internet Options Setup and Operations Manual* for details on configuring the IP address and subnet mask for the robot.

A.2 ETHERNET LEDS

The Ethernet LEDs are located on the Main board. See Figure A–1 .

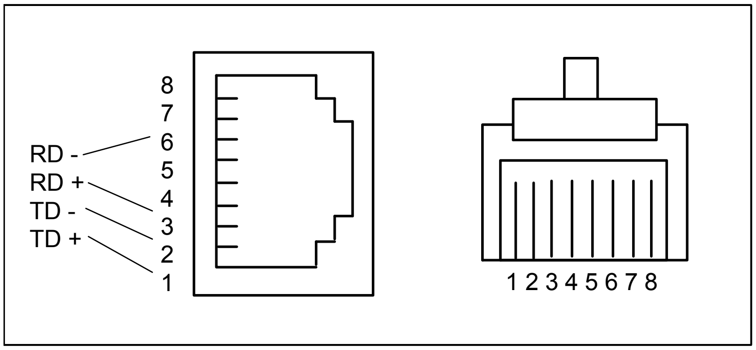
Figure A–1. Ethernet LEDs



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–2 for 10 Base-T/100 Base-TX connector assignment.

Figure A–2. 10 Base-T/100 Base-TX Connector Pin Assignments



Appendix B

CONFIGURE FTP WITH A KAREL COMMAND FILE

Contents

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’	

Table B–1. Example NETSETUP.CF File (for interface #1) (Cont'd)

INSTRUCTION	DESCRIPTION
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.
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	

Table B–1. Example NETSETUP.CF File (for interface #1) (Cont'd)

INSTRUCTION	DESCRIPTION
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 = "	
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'	

Table B–1. Example NETSETUP.CF File (for interface #1) (Cont'd)

INSTRUCTION	DESCRIPTION
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/'	
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'	

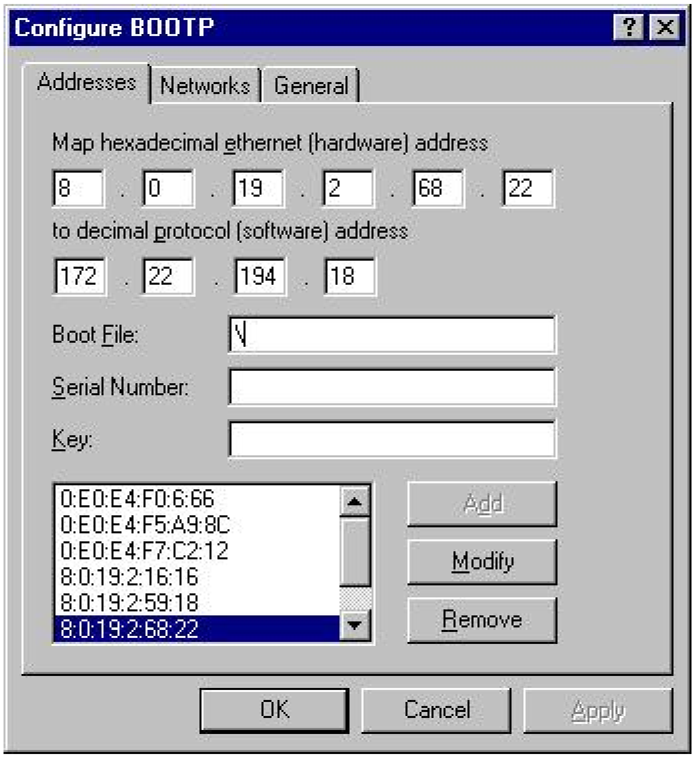
Appendix C

SETTING UP DISTINCT BOOTP/TFTP SERVER

Contents

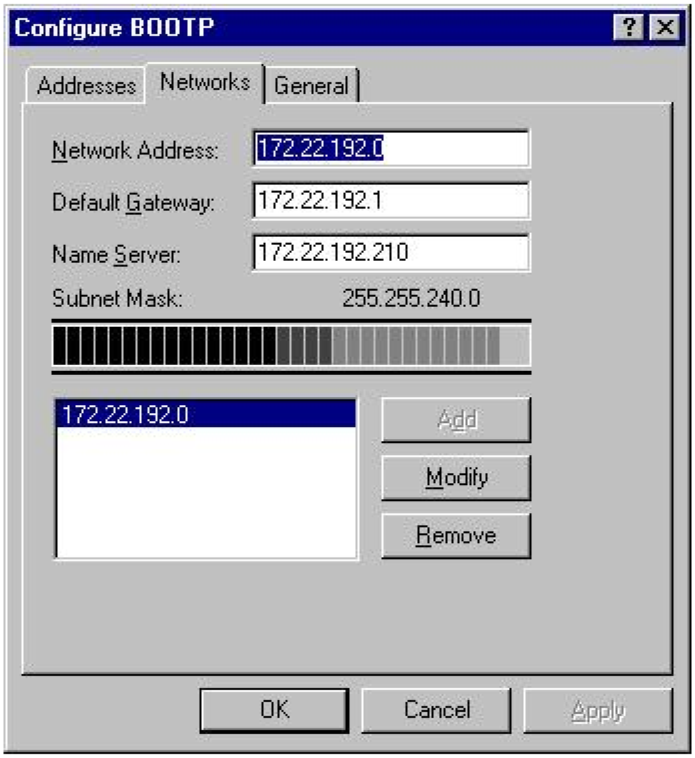
C.1 OVERVIEW

The BOOTP server has a database of robot-Ethernet-address-to-IP-address mappings. Every robot that is loaded over Ethernet needs to have an entry in this database. An entry is added using the following screen:



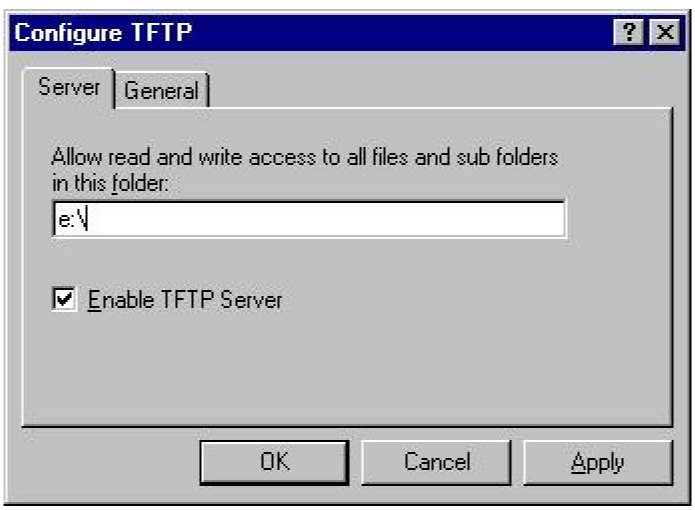
The Boot File field lists the directory where the media files are located on the PC. The directory information in this field is offset from the TFTP “root” path described below.

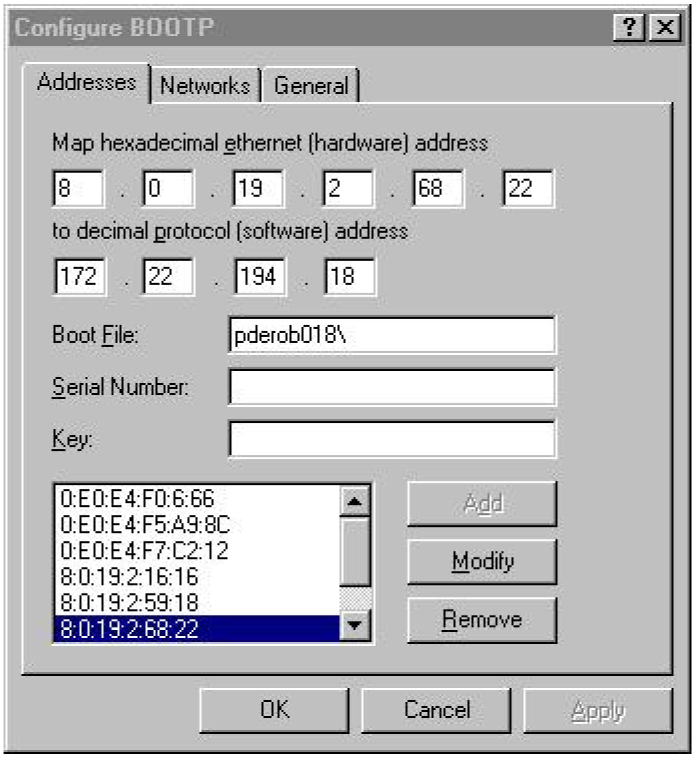
General network parameters are set up to be returned to every robot that needs to be loaded over Ethernet.



TFTP configuration

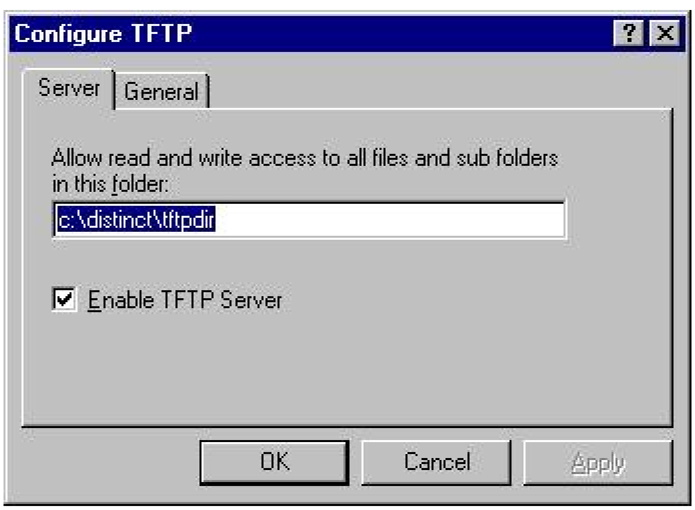
The TFTP “root” path needs to be set up correctly. It describes the base drive/directory where the media files are located on the PC.

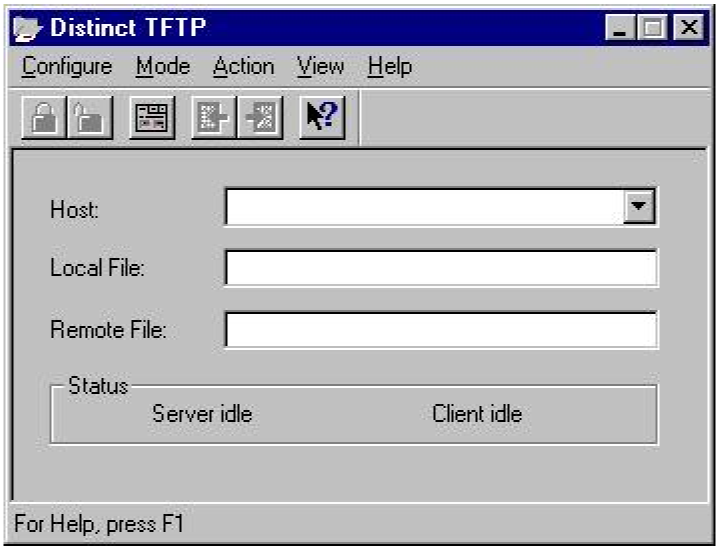




C.2 CONFIGURING THE BOOTP/TFTP SERVER TO PERFORM IMAGE BACKUP/RESTORE

You can set up the Boot File field in the BOOTP server configuration and the TFTP “root” path in the TFTP server configuration to allow for unique directories for each robot to hold the backups.





In this case , C:\distinct\tftpdir is the root directory of the backups/restore and each robot has an individual directory under this root directory to hold the backed up images.

Note You can use the same server setup to allow for both full loads and image backup/restores. You would need to copy the media files in a unique directory under C:\distinct\tftpdir using the above setup.

Note If you are using a UNIX host to run the BOOTP/TFTP servers, the boot file field must end in a forward slash (/) character (for example, pderob018/) , while if you are running the BOOTP/TFTP servers on a PC, the boot file field must end in a back slash (\ ) character (for example, pderob018\).

Glossary

A

abort

Abnormal termination of a computer program caused by hardware or software malfunction or operator cancellation.

absolute pulse code system

A positional information system for servomotors that relies on battery-backed RAM to store encoder pulse counts when the robot is turned off. This system is calibrated when it is turned on.

A/D value

An analog to digital-value. Converts a multilevel analog electrical system pattern into a digital bit.

AI

Analog input.

AO

Analog output.

alarm

The difference in value between actual response and desired response in the performance of a controlled machine, system or process. Alarm=Error.

algorithm

A fixed step-by-step procedure for accomplishing a given result.

alphanumeric

Data that are both alphabetical and numeric.

AMPS

Amperage amount.

analog

The representation of numerical quantities by measurable quantities such as length, voltage or resistance. Also refers to analog type I/O blocks and distinguishes them from discrete I/O blocks. Numerical data that can vary continuously, for example, voltage levels that can vary within the range of -10 to +10 volts.

AND

An operation that places two contacts or groups of contacts in series. All contacts in series control the resulting status and also mathematical operator.

ANSI

American National Standard Institute, the U.S. government organization with responsibility for the development and announcement of technical data standards.

APC

See absolute pulse code system.

APC motor

See servomotor.

application program

The set of instructions that defines the specific intended tasks of robots and robot systems to make them reprogrammable and multifunctional. You can initiate and change these programs.

arm

A robot component consisting of an interconnecting set of links and powered joints that move and support the wrist socket and end effector.

articulated arm

A robot arm constructed to simulate the human arm, consisting of a series of rotary motions and joints, each powered by a motor.

ASCII

Abbreviation for American Standard Code for Information Interchange. An 8-level code (7 bits plus 1 parity bit) commonly used for the exchange of data.

automatic mode

The robot state in which automatic operation can be initiated.

automatic operation

The time during which robots are performing programmed tasks through unattended program execution.

axis

1. A straight line about which a robot joint rotates or moves. 2. One of the reference lines or a coordinate system. 3. A single joint on the robot arm.

B

backplane

A group of connectors mounted at the back of a controller rack to which printed circuit boards are mated.

BAR

A unit of pressure equal to 100,000 pascals.

barrier

A means of physically separating persons from the restricted work envelope; any physical boundary to a hazard or electrical device/component.

battery low alarm

A programmable value (in engineering units) against which the analog input signal automatically is compared on Genius I/O blocks. A fault is indicated if the input value is equal to or less than the low alarm value.

baud

A unit of transmission speed equal to the number of code elements (bits) per second.

big-endian

The adjectives big-endian and little-endian refer to which bytes are most significant in multi-byte data types and describe the order in which a sequence of bytes is stored in a computer's memory. In a big-endian system, the most significant value in the sequence is stored at the lowest storage address (i.e., first). In a little-endian system, the least significant value in the sequence is stored first.

binary

A numbering system that uses only 0 and 1.

bit

Contraction of binary digit. 1. The smallest unit of information in the binary numbering system, represented by a 0 or 1. 2. The smallest division of a programmable controller word.

bps

Bits per second.

buffer

A storage area in the computer where data is held temporarily until the computer can process it.

bus

A channel along which data can be sent.

bus controller

A Genius bus interface board for a programmable controller.

bus scan

One complete communications cycle on the serial bus.

Bus Switching Module

A device that switches a block cluster to one bus or the other of a dual bus.

byte

A sequence of binary digits that can be used to store a value from 0 to 255 and usually operated upon as a unit. Consists of eight bits used to store two numeric or one alpha character.

C

calibration

The process whereby the joint angle of each axis is calculated from a known reference point.

Cartesian coordinate system

A coordinate system whose axes (x, y, and z) are three intersecting perpendicular straight lines. The origin is the intersection of the axes.

Cartesian coordinates

A set of three numbers that defines the location of a point within a rectilinear coordinate system and consisting of three perpendicular axes (x, y, z).

cathode ray tube

A device, like a television set, for displaying information.

central processing unit

The main computer component that is made up of a control section and an arithmetic-logic section. The other basic units of a computer system are input/output units and primary storage.

channel

The device along which data flow between the input/output units of a computer and primary storage.

character

One of a set of elements that can be arranged in ordered groups to express information. Each character has two forms: 1. a man-intelligible form, the graphic, including the decimal digits 0-9, the letters A-Z, punctuation marks, and other formatting and control symbols; 2. a computer intelligible form, the code, consisting of a group of binary digits (bits).

circular

A MOTYPE option in which the robot tool center point moves in an arc defined by three points. These points can be positions or path nodes.

clear

To replace information in a storage unit by zero (or blank, in some machines).

closed loop

A control system that uses feedback. An open loop control system does not use feedback.

C-MOS RAM

Complementary metal-oxide semiconductor random-access memory. A read/write memory in which the basic memory cell is a pair of MOS (metal-oxide semiconductor) transistors. It is an implementation of S-RAM that has very low power consumption, but might be less dense than other S-RAM implementations.

coaxial cable

A transmission line in which one conductor is centered inside and insulated from an outer metal tube that serves as the second conductor. Also known as coax, coaxial line, coaxial transmission line, concentric cable, concentric line, concentric transmission line.

component

An inclusive term used to identify a raw material, ingredient, part or subassembly that goes into a higher level of assembly, compound or other item.

computer

A device capable of accepting information, applying prescribed processes to the information, and supplying the results of these processes.

configuration

The joint positions of a robot and turn number of wrist that describe the robot at a specified position. Configuration is designated by a STRING value and is included in positional data.

continuous path

A trajectory control system that enables the robot arm to move at a constant tip velocity through a series of predefined locations. A rounding effect of the path is required as the tip tries to pass through these locations.

continuous process control

The use of transducers (sensors) to monitor a process and make automatic changes in operations through the design of appropriate feedback control loops. While such devices historically have been mechanical or electromechanical, microcomputers and centralized control is now used, as well.

continuous production

A production system in which the productive equipment is organized and sequenced according to the steps involved to produce the product. Denotes that material flow is continuous during the production process. The routing of the jobs is fixed and set-ups are seldom changed.

controlled stop

A controlled stop controls robot deceleration until it stops. When a safety stop input such as a safety fence signal is opened, the robot decelerates in a controlled manner and then stops. After the robot stops, the Motor Control Contactor opens and drive power is removed.

controller

A hardware unit that contains the power supply, operator controls, control circuitry, and memory that directs the operation and motion of the robot and communications with external devices. See control unit.

controller memory

A medium in which data are retained. Primary storage refers to the internal area where the data and program instructions are stored for active use, as opposed to auxiliary or external storage (magnetic tape, disk, diskette, and so forth.)

control, open-loop

An operation where the computer applies control directly to the process without manual intervention.

control unit

The portion of a computer that directs the automatic operation of the computer, interprets computer instructions, and initiates the proper signals to the other computer circuits to execute instructions.

coordinate system

See Cartesian coordinate system.

CPU

See central processing unit.

CRT

See cathode ray tube.

cps (viscosity)

Centipoises per second.

CRT/KB

Cathode ray tube/keyboard. An optional interface device for the robot system. The CRT/KB is used for some robot operations and for entering programs. It can be a remote device that attaches to the robot via a cable.

cycle

1. A sequence of operations that is repeated regularly. The time it takes for one such sequence to occur. 2. The interval of time during which a system or process, such as seasonal demand or a manufacturing operation, periodically returns to similar initial conditions. 3. The interval of time during which an event or set of events is completed. In production control, a cycle is the length of time between the release of a manufacturing order and shipment to the customer or inventory.

cycle time

1. In industrial engineering, the time between completion of two discrete units of production. 2. In materials management, the length of time from when material enters a production facility until it exits. See throughput.

cursor

An indicator on a teach pendant or CRT display screen at which command entry or editing occurs. The indicator can be a highlighted field or an arrow (> or ^).

cylindrical

Type of work envelope that has two linear major axes and one rotational major axis. Robotic device that has a predominantly cylindrical work envelope due to its design. Typically has fewer than 6 joints and typically has only 1 linear axis.

D

D/A converter

A digital-to-analog converter. A device that transforms digital data into analog data.

D/A value

A digital-to-analog value. Converts a digital bit pattern into a multilevel analog electrical system.

daisy chain

A means of connecting devices (readers, printers, etc.) to a central processor by party-line input/output buses that join these devices by male and female connectors. The last female connector is shorted by a suitable line termination.

daisy chain configuration

A communications link formed by daisy chain connection of twisted pair wire.

data

A collection of facts, numeric and alphabetical characters, or any representation of information that is suitable for communication and processing.

data base

A data file philosophy designed to establish the independence of computer program from data files. Redundancy is minimized and data elements can be added to, or deleted from, the file designs without changing the existing computer programs.

DC

Abbreviation for direct current.

DEADMAN switch

A control switch on the teach pendant that is used to enable servo power. Pressing the DEADMAN switch while the teach pendant is on activates servo power and releases the robot brakes; releasing the switch deactivates servo power and applies the robot brakes.

debugging

The process of detecting, locating and removing mistakes from a computer program, or manufacturing control system. See diagnostic routine.

deceleration tolerance

The specification of the percentage of deceleration that must be completed before a motion is considered finished and another motion can begin.

default

The value, display, function or program automatically selected if you have not specified a choice.

deviation

Usually, the absolute difference between a number and the mean of a set of numbers, or between a forecast value and the actual data.

device

Any type of control hardware, such as an emergency-stop button, selector switch, control pendant, relay, solenoid valve, or sensor.

diagnostic routine

A test program used to detect and identify hardware/software malfunctions in the controller and its associated I/O equipment. See debugging.

diagnostics

Information that permits the identification and evaluation of robot and peripheral device conditions.

digital

A description of any data that is expressed in numerical format. Also, having the states On and Off only.

digital control

The use of a digital computer to perform processing and control tasks in a manner that is more accurate and less expensive than an analog control system.

digital signal

A single point control signal sent to or from the controller. The signal represents one of two states: ON (TRUE, 1. or OFF (FALSE, 0).

directory

A listing of the files stored on a device.

discrete

Consisting of individual, distinct entities such as bits, characters, circuits, or circuit components. Also refers to ON/OFF type I/O blocks.

disk

A secondary memory device in which information is stored on a magnetically sensitive, rotating disk.

disk memory

A non-programmable, bulk-storage, random-access memory consisting of a magnetized coating on one or both sides of a rotating thin circular plate.

drive power

The energy source or sources for the robot servomotors that produce motion.

DRAM

Dynamic Random Access Memory. A read/write memory in which the basic memory cell is a capacitor. DRAM (or D-RAM) tends to have a higher density than SRAM (or S-RAM). Due to the support circuitry required, and power consumption needs, it is generally impractical to use. A battery can be used to retain the content upon loss of power.

E

edit

1. A software mode that allows creation or alteration of a program. 2. To modify the form or format of data, for example, to insert or delete characters.

emergency stop

The operation of a circuit using hardware-based components that overrides all other robot controls, removes drive power from the actuators, and causes all moving parts of to stop. The operator panel and teach pendant are each equipped with EMERGENCY STOP buttons.

enabling device

A manually operated device that, when continuously activated, permits motion. Releasing the device stops the motion of the robot and associated equipment that might present a hazard.

encoder

1. A device within the robot that sends the controller information about where the robot is. 2. A transducer used to convert position data into electrical signals. The robot system uses an incremental optical encoder to provide position feedback for each joint. Velocity data is computed from the encoder signals and used as an additional feedback signal to assure servo stability.

end effector

An accessory device or tool specifically designed for attachment to the robot wrist or tool mounting plate to enable the robot to perform its intended tasks. Examples include gripper, spot weld gun, arc weld gun, spray paint gun, etc.

end-of-arm tooling

Any of a number of tools, such as welding guns, torches, bells, paint spraying devices, attached to the faceplate of the robot wrist. Also called end effector or EOAT.

engineering units

Units of measure as applied to a process variable, for example, psi, Degrees F., etc.

envelope, maximum

The volume of space encompassing the maximum designed movements of all robot parts including the end effector, workpiece, and attachments.

EOAT

See end of arm tooling, tool.

EPROM

Erasable Programmable Read Only Memory. Semiconductor memory that can be erased and reprogrammed. A non-volatile storage memory.

error

The difference in value between actual response and desired response in the performance of a controlled machine, system or process. Alarm=Error.

error message

A numbered message, displayed on the CRT/KB and teach pendant, that indicates a system problem or warns of a potential problem.

Ethernet

A Local Area Network (LAN) bus-oriented, hardware technology that is used to connect computers, printers, terminal concentrators (servers), and many other devices together. It consists of a master cable and connection devices at each machine on the cable that allow the various devices to "talk" to each other. Software that can access the Ethernet and cooperate with machines connected to the cable is necessary. Ethernets come in varieties such as baseband and broadband and can run on different media, such as coax, twisted pair and fiber. Ethernet is a trademark of Xerox Corporation.

execute

To perform a specific operation, such as one that would be accomplished through processing one statement or command, a series of statements or commands, or a complete program or command procedure.

extended axis

An optional, servo-controlled axis that provides extended reach capability for a robot, including in-booth rail, single- or double-link arm, also used to control motion of positioning devices.

F

faceplate

The tool mounting plate of the robot.

feedback

1. The signal or data fed back to a commanding unit from a controlled machine or process to denote its response to the command signal. The signal representing the difference between actual response and desired response that is used by the commanding unit to improve performance of the controlled machine or process. 2. The flow of information back into the control system so that actual performance can be compared with planned performance, for instance in a servo system.

field

A specified area of a record used for a particular category of data. 2. A group of related items that occupy the same space on a CRT/KB screen or teach pendant LCD screen. Field name is the name of the field; field items are the members of the group.

field devices

User-supplied devices that provide information to the PLC (inputs: push buttons, limit switches, relay contacts, an so forth) or perform PLC tasks (outputs: motor starters, solenoids, indicator lights, and so forth.)

file

1. An organized collection of records that can be stored or retrieved by name. 2. The storage device on which these records are kept, such as bubble memory or disk.

filter

A device to suppress interference that would appear as noise.

Flash File Storage

A portion of FROM memory that functions as a separate storage device. Any file can be stored on the FROM disk.

Flash ROM

Flash Read Only Memory. Flash ROM is not battery-backed memory but it is non-volatile. All data in Flash ROM is saved even after you turn off and turn on the robot.

flow chart

A systems analysis tool to graphically show a procedure in which symbols are used to represent operations, data, flow, and equipment. See block diagram, process chart.

flow control

A specific production control system that is based primarily on setting production rates and feeding work into production to meet the planned rates, then following it through production to make sure that it is moving. This concept is most successful in repetitive production.

format

To set up or prepare a memory card or floppy disk (not supported with version 7.20 and later) so it can be used to store data in a specific system.

FR

See Flash ROM.

F-ROM

See Flash ROM.

FROM disk

See Flash ROM.

G

general override stat

A percentage value that governs the maximum robot jog speed and program run speed.

Genius I/O bus

The serial bus that provides communications between blocks, controllers, and other devices in the system especially with respect to GE FANUC Genius I/O.

gripper

The "hand" of a robot that picks up, holds and releases the part or object being handled. Sometimes referred to as a manipulator. See EOAT, tool.

group signal

An input/output signal that has a variable number of digital signals, recognized and taken as a group.

gun

See applicator.

H

Hand Model.

Used in Interference Checking, the Hand Model is the set of virtual model elements (spheres and cylinders) that are used to represent the location and shape of the end of arm tooling with respect to the robot’s faceplate.

hardware

1. In data processing, the mechanical, magnetic, electrical and electronic devices of which a computer, controller, robot, or panel is built. 2. In manufacturing, relatively standard items such as nuts, bolts, washers, clips, and so forth.

hard-wire

To connect electric components with solid metallic wires.

hard-wired

1. Having a fixed wired program or control system built in by the manufacturer and not subject to change by programming. 2. Interconnection of electrical and electronic devices directly through physical wiring.

hazardous motion

Unintended or unexpected robot motion that can cause injury.

hexadecimal

A numbering system having 16 as the base and represented by the digits 0 through 9, and A through F.

hold
A smoothly decelerated stopping of all robot movement and a pause of program execution. Power is maintained on the robot and program execution generally can be resumed from a hold.

HTML.
Hypertext Markup Language. A markup language that is used to create hypertext and hypermedia documents incorporating text, graphics, sound, video, and hyperlinks.

http.
Hypertext transfer protocol. The protocol used to transfer HTML files between web servers.

I

impedance
A measure of the total opposition to current flow in an electrical circuit.

incremental encoder system
A positional information system for servomotors that requires calibrating the robot by moving it to a known reference position (indicated by limit switches) each time the robot is turned on or calibration is lost due to an error condition.

index
An integer used to specify the location of information within a table or program.

index register
A memory device containing an index.

industrial robot
A reprogrammable multifunctional manipulator designed to move material, parts, tools, or specialized devices through variable programmed motions in order to perform a variety of tasks.

industrial robot system
A system that includes industrial robots, end effectors, any equipment devices and sensors required for the robot to perform its tasks, as well as communication interfaces for interlocking, sequencing, or monitoring the robot.

information
The meaning derived from data that have been arranged and displayed in a way that they relate to that which is already known. See data.

initialize
1. Setting all variable areas of a computer program or routine to their desired initial status, generally done the first time the code is executed during each run. 2. A program or hardware circuit that returns a program a system, or hardware device to an original state. See startup, initial.

input

The data supplied from an external device to a computer for processing. The device used to accomplish this transfer of data.

input device

A device such as a terminal keyboard that, through mechanical or electrical action, converts data from the form in which it has been received into electronic signals that can be interpreted by the CPU or programmable controller. Examples are limit switches, push buttons, pressure switches, digital encoders, and analog devices.

input processing time

The time required for input data to reach the microprocessor.

input/output

Information or signals transferred between devices, discreet electrical signals for external control.

input/output control

A technique for controlling capacity where the actual output from a work center is compared with the planned output developed by CRP. The input is also monitored to see if it corresponds with plans so that work centers will not be expected to generate output when jobs are not available to work on.

integrated circuit

A solid-state micro-circuit contained entirely within a chip of semiconductor material, generally silicon. Also called chip.

interactive

Refers to applications where you communicate with a computer program via a terminal by entering data and receiving responses from the computer.

interface

1. A concept that involves the specifications of the inter-connection between two equipments having different functions. 2. To connects a PLC with the application device, communications channel, and peripherals through various modules and cables. 3. The method or equipment used to communicate between devices.

interference zone

An area that falls within the work envelope of a robot, in which there is the potential for the robot motion to coincide with the motion of another robot or machine, and for a collision to occur.

interlock

An arrangement whereby the operation of one control or mechanism brings about, or prevents, the operations of another.

interrupt

A break in the normal flow of a system or program that occurs in a way that the flow can be resumed from that point at a later time. Interrupts are initiated by two types of signals: 1. signals originating within the computer system to synchronize the operation of the computer system with the outside world; 2. signals originating exterior to the computer system to synchronize the operation of the computer system with the outside world.

I/O

Abbreviation for input/output or input/output control.

I/O block

A microprocessor-based, configurable, rugged solid state device to which field I/O devices are attached.

I/O electrical isolation

A method of separating field wiring from logic level circuitry. This is typically done through optical isolation devices.

I/O module

A printed circuit assembly that is the interface between user devices and the Series Six PLC.

I/O scan

A method by which the CPU monitors all inputs and controls all outputs within a prescribed time. A period during which each device on the bus is given a turn to send information and listen to all of the broadcast data on the bus.

ISO

The International Standards Organization that establishes the ISO interface standards.

isolation

1. The ability of a logic circuit having more than one inputs to ensure that each input signal is not affected by any of the others. 2. A method of separating field wiring circuitry from logic level circuitry, typically done optically.

item

1. A category displayed on the teach pendant on a menu. 2. A set of adjacent digits, bits, or characters that is treated as a unit and conveys a single unit of information. 3. Any unique manufactured or purchased part or assembly: end product, assembly, subassembly, component, or raw material.

J

jog coordinate systems

Coordinate systems that help you to move the robot more effectively for a specific application. These systems include JOINT, WORLD, TOOL, and USER.

JOG FRAME

A jog coordinate system you define to make the robot jog the best way possible for a specific application. This can be different from world coordinate frame.

jogging

Pressing special keys on the teach pendant to move the robot.

jog speed

Is a percentage of the maximum speed at which you can jog the robot.

joint

1. A single axis of rotation. There are up to six joints in a robot arm (P-155 swing arm has 8). 2. A jog coordinate system in which one axis is moved at a time.

JOINT

A motion type in which the robot moves the appropriate combination of axes independently to reach a point most efficiently. (Point to point, non-linear motion).

joint interpolated motion

A method of coordinating the movement of the joints so all joints arrive at the desired location at the same time. This method of servo control produces a predictable path regardless of speed and results in the fastest cycle time for a particular move. Also called joint motion.

K

K

Abbreviation for kilo, or exactly 1024 in computer jargon. Related to 1024 words of memory.

KAREL

The programming language developed for robots by the FANUC America Corporation.

L

label

An ordered set of characters used to symbolically identify an instruction, a program, a quantity, or a data area.

LCD

See liquid crystal display.

lead time

The span of time needed to perform an activity. In the production and inventory control context, this activity is normally the procurement of materials and/or products either from an outside supplier or from one's own manufacturing facility. Components of lead time can include order preparation time, queue time, move or transportation time, receiving and inspection time.

LED

See Light Emitting Diode.

LED display

An alphanumeric display that consists of an array of LEDs.

Light Emitting Diode

A solid-state device that lights to indicate a signal on electronic equipment.

limiting device

A device that restricts the work envelope by stopping or causing to stop all robot motion and that is independent of the control program and the application programs.

limit switch

A switch that is actuated by some part or motion of a machine or equipment to alter the electrical circuit associated with it. It can be used for position detection.

linear

A motion type in which the appropriate combination of axes move in order to move the robot TCP in a straight line while maintaining tool center point orientation.

liquid crystal display

A digital display on the teach pendant that consists of two sheets of glass separated by a sealed-in, normally transparent, liquid crystal material. Abbreviated LCD.

little-endian

The adjectives big-endian and little-endian refer to which bytes are most significant in multi-byte data types and describe the order in which a sequence of bytes is stored in a computer's memory. In a big-endian system, the most significant value in the sequence is stored at the lowest storage address (i.e., first). In a little-endian system, the least significant value in the sequence is stored first.

load

1. The weight (force) applied to the end of the robot arm. 2. A device intentionally placed in a circuit or connected to a machine or apparatus to absorb power and convert it into the desired useful form. 3. To copy programs or data into memory storage.

location

1. A storage position in memory uniquely specified by an address. 2. The coordinates of an object used in describing its x, y, and z position in a Cartesian coordinate system.

lockout/tagout

The placement of a lock and/or tag on the energy isolating device (power disconnecting device) in the off or open position. This indicates that the energy isolating device or the equipment being controlled will not be operated until the lock/tag is removed.

log

A record of values and/or action for a given function.

logic

A fixed set of responses (outputs) to various external conditions (inputs). Also referred to as the program.

loop

The repeated execution of a series of instructions for a fixed number of times, or until interrupted by the operator.

M

mA

See milliampere.

machine language

A language written in a series of bits that are understandable by, and therefore instruct, a computer. This is a "first level" computer language, as compared to a "second level" assembly language, or a "third level" compiler language.

machine lock

A test run option that allows the operator to run a program without having the robot move.

macro

A source language instruction from which many machine-language instructions can be generated.

magnetic disk

A metal or plastic floppy disk (not supported on version 7.10 and later) that looks like a phonograph record whose surface can store data in the form of magnetized spots.

magnetic disk storage

A storage device or system consisting of magnetically coated metal disks.

magnetic tape

Plastic tape, like that used in tape recorder, on which data is stored in the form of magnetized spots.

maintenance

Keeping the robots and system in their proper operating condition.

MC

See memory card.

mechanical unit

The robot arm, including auxiliary axis, and hood/deck and door openers.

medium

plural **media** . The physical substance upon which data is recorded, such as a memory card (or floppy disk which is not supported on version 7.10 and later).

memory

A device or media used to store information in a form that can be retrieved and is understood by the computer or controller hardware. Memory on the controller includes C-MOS RAM, Flash ROM and D-RAM.

memory card

A C-MOS RAM memory card or a flash disk-based PC card.

menu

A list of options displayed on the teach pendant screen.

message

A group of words, variable in length, transporting an item of information.

microprocessor

A single integrated circuit that contains the arithmetic, logic, register, control and memory elements of a computer.

microsecond

One millionth (0.000001) of a second

milliampere

One one-thousandth of an ampere. Abbreviated mA.

millisecond

One thousandth of a second. Abbreviated msec.

module

A distinct and identifiable unit of computer program for such purposes as compiling, loading, and linkage editing. It is eventually combined with other units to form a complete program.

motion type

A feature that allows you to select how you want the robot to move from one point to the next. MOTYPES include joint, linear, and circular.

mode

1. One of several alternative conditions or methods of operation of a device. 2. The most common or frequent value in a group of values.

N

network

1. The interconnection of a number of devices by data communication facilities. "Local networking" is the communications network internal to a robot."Global networking" is the ability to provide communications connections outside of the robot's internal system. 2. Connection of geographically separated computers and/or terminals over communications lines. The control of transmission is managed by a standard protocol.

non-volatile memory

Memory capable of retaining its stored information when power is turned off.

O

Obstacle Model.

Used in Interference Checking, the Obstacle Model is the set of virtual model elements (spheres, cylinders, and planes) that are used to represent the shape and the location of a given obstacle in space.

off-line

Equipment or devices that are not directly connected to a communications line.

off-line operations

Data processing operations that are handled outside of the regular computer program. For example, the computer might generate a report off-line while the computer was doing another job.

off-line programming

The development of programs on a computer system that is independent of the "on-board" control of the robot. The resulting programs can be copied into the robot controller memory.

offset

The count value output from a A/D converter resulting from a zero input analog voltage. Used to correct subsequent non-zero measurements also incremental position or frame adjustment value.

on-line

A term to describe equipment or devices that are connected to the communications line.

on-line processing

A data processing approach where transactions are entered into the computer directly, as they occur.

operating system

Lowest level system monitor program.

operating work envelope

The portion of the restricted work envelope that is actually used by the robot while it is performing its programmed motion. This includes the maximum the end-effector, the workpiece, and the robot itself.

operator

A person designated to start, monitor, and stop the intended productive operation of a robot or robot system.

operator box

A control panel that is separate from the robot and is designed as part of the robot system. It consists of the buttons, switches, and indicator lights needed to operate the system.

operator panel

A control panel designed as part of the robot system and consisting of the buttons, switches, and indicator lights needed to operate the system.

optional features

Additional capabilities available at a cost above the base price.

OR

An operation that places two contacts or groups of contacts in parallel. Any of the contacts can control the resultant status, also a mathematical operation.

orientation

The attitude of an object in space. Commonly described by three angles: rotation about x (w), rotation about y (p), and rotation about z (r).

origin

The point in a Cartesian coordinate system where axes intersect; the reference point that defines the location of a frame.

OT

See overtravel.

output

Information that is transferred from the CPU for control of external devices or processes.

output device

A device, such as starter motors, solenoids, that receive data from the programmable controller.

output module

An I/O module that converts logic levels within the CPU to a usable output signal for controlling a machine or process .

outputs

Signals, typically on or off, that controls external devices based upon commands from the CPU.

override

See general override.

overtravel

A condition that occurs when the motion of a robot axis exceeds its prescribed limits.

overwrite

To replace the contents of one file with the contents of another file when copying.

P

parity

The anticipated state, odd or even, of a set of binary digits.

parity bit

A binary digit added to an array of bits to make the sum of all bits always odd or always even.

parity check

A check that tests whether the number of ones (or zeros) in an array of binary digits is odd or even.

parity error

A condition that occurs when a computed parity check does not agree with the parity bit.

part

A material item that is used as a component and is not an assembly or subassembly.

pascal

A unit of pressure in the meter-kilogram-second system equivalent to one newton per square meter.

path

1. A variable type available in the KAREL system that consists of a list of positions. Each node includes positional information and associated data. 2. The trajectory followed by the TCP in a move.

PCB

See printed circuit board.

PC Interface

The PC Interface option provides the RPC functions and PC send macros required by applications created using PC Developer’s Kit.

pendant

See teach pendant.

PLC

See programmable logic controller or cell controller.

PMC

The programmable machine controller (PMC) functions provide a ladder logic programming environment to create PMC functions. This provides the capability to use the robot I/O system to run PLC programs in the background of normal robot operations. This function can be used to control bulk supply systems, fixed automation that is part of the robot workcell, or other devices that would normally require basic PLC controls.

printed circuit board

A flat board whose front contains slots for integrated circuit chips and connections for a variety of electronic components, and whose back is printed with electrically conductive pathways between the components.

production mode

See automatic mode.

program

1. A plan for the solution of a problem. A complete program includes plans for the transcription of data, coding for the computer, and plans for the absorption of the results into the system. 2. A sequence of instructions to be executed by the computer or controller to control a robot/robot system. 3. To furnish a computer with a code of instructions. 4. To teach a robot system a specific set of movements and instructions to do a task.

programmable controller

See programmable logic controller or cell controller.

programmable logic controller

A solid-state industrial control device that receives inputs from user-supplied control devices, such as switches and sensors, implements them in a precise pattern determined by ladder diagram-based programs stored in the user memory, and provides outputs for control of processes or user-supplied devices such as relays and motor starters.

Program ToolBox

The Program ToolBox software provides programming utilities such as mirror image and flip wrist editing capabilities.

protocol

A set of hardware and software interfaces in a terminal or computer that allows it to transmit over a communications network, and that collectively forms a communications language.

psi

Pounds per square inch.

Q

queue.

1. Waiting lines resulting from temporary delays in providing service. 2. The amount of time a job waits at a work center before set-up or work is performed on the job. See also job queue.

R

RAM

See Random Access Memory.

random access

A term that describes files that do not have to be searched sequentially to find a particular record but can be addressed directly.

Random Access Memory

1. Volatile, solid-state memory used for storage of programs and locations; battery backup is required. 2. The working memory of the controller. Programs and variable data must be loaded into RAM before the program can execute or the data can be accessed by the program.

range

1. A characterization of a variable or function. All the values that a function can possess. 2. In statistics, the spread in a series of observations. 3. A programmable voltage or current spectrum of values to which input or output analog signals can be limited.

RI

Robot input.

RO

Robot output.

read

To copy, usually from one form of storage to another, particularly from external or secondary storage to internal storage. To sense the meaning of arrangements of hardware. To sense the presence of information on a recording medium.

Read Only Memory

A digital memory containing a fixed pattern of bits that you cannot alter.

record

To store the current set or sets of information on a storage device.

recovery

The restoration of normal processing after a hardware or software malfunction through detailed procedures for file backup, file restoration, and transaction logging.

register

1. A special section of primary storage in a computer where data is held while it is being worked on.
2. A memory device capable of containing one or more computer bits or words.

remote/local

A device connection to a given computer, with remote devices being attached over communications lines and local devices attached directly to a computer channel; in a network, the computer can be a remote device to the CPU controlling the network.

repair

To restore robots and robot systems to operating condition after damage, malfunction, or wear.

repeatability

The closeness of agreement among the number of consecutive movements made by the robot arm to a specific point.

reset

To return a register or storage location to zero or to a specified initial condition.

restricted work envelope

That portion of the work envelope to which a robot is restricted by limiting devices that establish limits that will not be exceeded in the event of any reasonably foreseeable failure of the robot or its controls. The maximum distance the robot can travel after the limited device is actuated defines the restricted work envelope of the robot.

RIA

Robotic Industries Association Subcommittee of the American National Standards Institute, Inc.

robot

A reprogrammable multifunctional manipulator designed to move material, parts, tools, or specialized devices, through variable programmed motions for the performance of a variety of tasks.

Robot Model.

Used in Interference Checking, the Robot Model is the set of virtual model elements (sphere and cylinders) that are used to represent the location and shape of the robot arm with respect to the robot's base. Generally, the structure of a six axes robot can be accurately modeled as a series of cylinders and spheres. Each model element represents a link or part of the robot arm.

ROM

See Read Only Memory.

routine

1. A list of coded instructions in a program. 2. A series of computer instructions that performs a specific task and can be executed as often as needed during program execution.

S

saving data.

Storing program data in Flash ROM, to a floppy disk (not supported on version 7.10 and later), or memory card.

scfm

Standard cubic feet per minute.

scratch start

Allows you to enable and disable the automatic recovery function.

sensor

A device that responds to physical stimuli, such as heat, light, sound pressure, magnetism, or motion, and transmits the resulting signal or data for providing a measurement, operating a control or both. Also a device that is used to measure or adjust differences in voltage in order to control sophisticated machinery dynamically.

serial communication

A method of data transfer within a PLC whereby the bits are handled sequentially rather than simultaneously as in parallel transmission.

serial interface

A method of data transmission that permits transmitting a single bit at a time through a single line. Used where high speed input is not necessary.

Server Side Include (SSI)

A method of calling or "including" code into a web page.

servomotor

An electric motor that is controlled to produce precision motion. Also called a "smart" motor.

SI

System input.

signal

The event, phenomenon, or electrical quantity that conveys information from one point to another.

significant bit

A bit that contributes to the precision of a number. These are counted starting with the bit that contributes the most value, of "most significant bit", and ending with the bit that contributes the least value, or "least significant bit".

singulating

Separating parts into a single layer.

slip sheet

A sheet of material placed between certain layers of a unit load. Also known as tier sheet.

SO

System output.

specific gravity

The ratio of a mass of solid or liquid to the mass of an equal volume of water at 45C. You must know the specific gravity of the dispensing material to perform volume signal calibration. The specific gravity of a dispensing material is listed on the MSDS for that material.

SRAM

A read/write memory in which the basic memory cell is a transistor. SRAM (or S-RAM) tends to have a lower density than DRAM. A battery can be used to retain the content upon loss of power.

slpm

Standard liters per minute.

Standard Operator Panel (SOP).

A panel that is made up of buttons, keyswitches, and connector ports.

state

The on or off condition of current to and from and input or output device.

statement

See instruction.

storage device

Any device that can accept, retain, and read back one or more times. The available storage devices are SRAM, Flash ROM (FROM or F-ROM), floppy disks (not available on version 7.10 and later), memory cards, or a USB memory stick.

system variable

An element that stores data used by the controller to indicate such things as robot specifications, application requirements, and the current status of the system.

T

Tare

The difference between the gross weight of an object and its contents, and the object itself. The weight of an object without its contents.

TCP

See tool center point.

teaching

Generating and storing a series of positional data points effected by moving the robot arm through a path of intended motions.

teach mode

1. The mode of operation in which a robot is instructed in its motions, usually by guiding it through these motions using a teach pendant. 2. The generation and storage of positional data. Positional data can be taught using the teach pendant to move the robot through a series of positions and recording those positions for use by an application program.

teach pendant

1. A hand-held device used to instruct a robot, specifying the character and types of motions it is to undertake. Also known as teach box, teach gun. 2. A portable device, consisting of an LCD display and a keypad, that serves as a user interface to the KAREL system and attaches to the operator box or operator panel via a cable. The teach pendant is used for robot operations such as jogging the robot, teaching and recording positions, and testing and debugging programs.

telemetry

The method of transmission of measurements made by an instrument or a sensor to a remote location.

termination type

Feature that controls the blending of robot motion between segments.

tool

A term used loosely to define something mounted on the end of the robot arm, for example, a hand, gripper, or an arc welding torch.

tool center point

1. The location on the end-effector or tool of a robot hand whose position and orientation define the coordinates of the controlled object. 2. Reference point for position control, that is, the point on the tool that is used to teach positions. Abbreviated TCP.

TOOL Frame

The Cartesian coordinate system that has the position of the TCP as its origin to stet. The z-axis of the tool frame indicates the approach vector for the tool.

TP.

See teach pendant.

transducer

A device for converting energy from one form to another.

U

UOP

See user operator panel.

URL

Universal Resource Locator. A standard addressing scheme used to locate or reference files on web servers.

USB memory stick

The controller USB memory stick interface supports a USB 1.1 interface. The USB Organization specifies standards for USB 1.1 and 2.0. Most memory stick devices conform to the USB 2.0 specification for operation and electrical standards. USB 2.0 devices as defined by the USB Specification must be backward compatible with USB 1.1 devices. However, FANUC America Corporation does not support any security or encryption features on USB memory sticks. The controller supports most widely-available USB Flash memory sticks from 32MB up to 1GB in size.

USER Frame

The Cartesian coordinate system that you can define for a specific application. The default value of the User Frame is the World Frame. All positional data is recorded relative to User Frame.

User Operator Panel

User-supplied control device used in place of or in parallel with the operator panel or operator box supplied with the controller. Abbreviated UOP .

V

variable

A quantity that can assume any of a given set of values.

variance

The difference between the expected (or planned) and the actual, also statistics definitions.

vision system

A device that collects data and forms an image that can be interpreted by a robot computer to determine the position or to “see” an object.

volatile memory

Memory that will lose the information stored in it if power is removed from the memory circuit device.

W

web server

An application that allows you to access files on the robot using a standard web browser.

warning device

An audible or visible device used to alert personnel to potential safety hazards.

work envelope

The volume of space that encloses the maximum designed reach of the robot manipulator including the end effector, the workpiece, and the robot itself. The work envelope can be reduced or restricted by limiting devices. The maximum distance the robot can travel after the limit device is actuated is considered the basis for defining the restricted work envelope.

write

To deliver data to a medium such as storage.

Index

R

S

T

U

W