

**FANUC America Corporation
SYSTEM R-30*i*A and R-30*i*B
Controller KAREL and TPP
Program Diagnostic Option
Manual**

MARACTPPD03081E REV C

Applies to software version 7.40 and higher.

© 2014 FANUC America Corporation
All Rights Reserved.

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

The information illustrated or contained herein is not to be reproduced, copied, downloaded, translated into another language, distributed, or published in any physical or electronic format, including Internet, or transmitted in whole or in part in any way without the prior written consent of FANUC America America, Inc.

AccuStat®, ArcTool®, iRVision®, KAREL®, PaintTool®, PalletTool®, SOCKETS®, SpotTool®, SpotWorks®, and TorchMate® are Registered Trademarks of FANUC America Corporation.

FANUC America reserves all proprietary rights, including but not limited to trademark and trade name rights, in the following names:

AccuAir™, AccuCal™, AccuChop™, AccuFlow™, AccuPath™, AccuSeal™, ARC Mate™, ARC Mate Sr.™, ARC Mate System 1™, ARC Mate System 2™, ARC Mate System 3™, ARC Mate System 4™, ARC Mate System 5™, ARCWorks Pro™, AssistTool™, AutoNormal™, AutoTCP™, BellTool™, BODYWorks™, Cal Mate™, Cell Finder™, Center Finder™, Clean Wall™, DualARM™, LR Tool™, MIG Eye™, MotionParts™, MultiARM™, NoBots™, Paint Stick™, PaintPro™, PaintTool 100™, PAINTWorks™, PAINTWorks II™, PAINTWorks III™, PalletMate™, PalletMate PC™, PalletTool PC™, PayloadID™, RecipTool™, RemovalTool™, Robo Chop™, Robo Spray™, S-420i™, S-430i™, ShapeGen™, SoftFloat™, SOFT PARTS™, SpotTool+™, SR Mate™, SR ShotTool™, SureWeld™, SYSTEM R-J2 Controller™, SYSTEM R-J3 Controller™, SYSTEM R-J3iB Controller™, SYSTEM R-J3iC Controller™, SYSTEM R-30iA Controller™, SYSTEM R-30iA Mate Controller™, SYSTEM R-30iB Controller™, SYSTEM R-30iB Mate Controller™, TCP Mate™, TorchMate™, TripleARM™, TurboMove™, visLOC™, visPRO-3D™, visTRAC™, WebServer™, WebTP™, and YagTool™.

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

About This Manual	i
Safety	ix
Chapter 1 OVERVIEW	1-1
Chapter 2 HARDWARE AND SOFTWARE	2-1
2.1 HARDWARE AND SOFTWARE REQUIREMENTS	2-2
2.1.1 Hardware	2-2
2.1.2 Software	2-2
2.2 PERFORMANCE	2-2
Chapter 3 SETUP AND OPERATIONS	3-1
3.1 SETTING UP THE PROGRAM DIAGNOSTIC OPTION	3-2
3.2 DUMP SELECTIONS SCREEN	3-3
3.3 TASK SELECTION SCREEN	3-5
3.4 STOP LOGGING TASKS SCREEN	3-7
3.5 LIST SELECTED TASKS SCREEN	3-9
3.6 EVENT CLASS SELECTION SCREEN	3-10
3.7 EVENT DETAIL SELECTION SCREEN	3-12
3.8 ENABLE OR DISABLE ALL EVENT LOGGING	3-13
Chapter 4 LOGGING EVENTS	4-1
4.1 SETTING UP EVENTS	4-2
4.2 LOGGING EVENTS TO AN ASCII FILE	4-3
4.2.1 ASCII File General Event Information	4-3
4.2.2 ASCII File Specific Event Information	4-4
Appendix A EXAMPLES	A-1
A.1 OVERVIEW	A-2
A.2 KAREL PROGRAM EXAMPLE	A-2
A.3 TEACH PENDANT PROGRAM EXAMPLE	A-3
A.4 ASCII FILE EXAMPLE	A-3
Glossary	GL-6
Index	Index-1

List of Tables

Table 4-1. Event Logging Information [4-4](#)

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 personnel 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.

OVERVIEW

Contents

Chapter 1 OVERVIEW 1–1

The *KAREL and TPP (Teach Pendant Programming) Program Diagnostic Option* is a tool used to diagnose user application or controller system software problems. This option generates logs that show details about events, including the sequence in which they occur.

This option allows you to log certain *events* that occur during KAREL and TPP program execution. It also allows you to generate ASCII files that contain all or part of the logged data. The ASCII files contain detailed information about each event that is logged.

For example, you can log the following types of events:

- Start of execution of KAREL statements or TPP lines.
- Initiation and completion of robot motion.
- Triggering of condition handlers.
- Internal system events for diagnosis of system software problems.

Refer to the following for additional information:

- For information about related **hardware and software**, refer to [Chapter 2 *HARDWARE AND SOFTWARE*](#).
- For information about **installing and setting up** the debugging option, refer to [Chapter 3 *SETUP AND OPERATIONS*](#).
- For information about **events**, refer to [Chapter 4 *LOGGING EVENTS*](#).
- For **examples** of this option as related to KAREL programming, teach pendant programming, and the ASCII file format, refer to [Appendix A](#).

HARDWARE AND SOFTWARE

Contents

Chapter 2	HARDWARE AND SOFTWARE	2-1
2.1	HARDWARE AND SOFTWARE REQUIREMENTS	2-2
2.1.1	Hardware	2-2
2.1.2	Software	2-2
2.2	PERFORMANCE	2-2

2.1 HARDWARE AND SOFTWARE REQUIREMENTS

There are certain hardware and software requirements that must be met before you can run the Program Diagnostic Option.

Note The hardware requirements in this section apply to all robot models.

2.1.1 Hardware

The Program Diagnostic Option requires the following hardware for all robot models:

- A FANUC Robot Controller
- A teach pendant
- 42,000 bytes of Temporary memory
- 18,000 bytes of CMOS
- 30,000 bytes of FROM file system space

2.1.2 Software

The Program Diagnostic Option can be installed in controllers with KAREL software version 7.20 or later.

A series of teach pendant screens is used to control the logging of diagnostic information. The teach pendant screens allow you to:

- Specify the kinds of events you want to log
- Specify the tasks for which events are to be logged
- Generate ASCII files containing the logged data

Note For information about how to install the Program Diagnostic Option, refer to the *Software Installation Manual*.

2.2 PERFORMANCE

The Program Diagnostic Option can affect the performance of your system.

- This option reduces execution speed of KAREL and teach pendant program logic by approximately 1 to 2%.

Note You can select "Disable all logging" on the Diagnostic Logging main menu to eliminate this reduction.

Refer to [Section 3.8](#) for more information about system performance.

- Actual logging of events adds about .08 ms per event logged.
- Motion execution is not affected.

SETUP AND OPERATIONS

Contents

Chapter 3	SETUP AND OPERATIONS	3-1
3.1	SETTING UP THE PROGRAM DIAGNOSTIC OPTION	3-2
3.2	DUMP SELECTIONS SCREEN	3-3
3.3	TASK SELECTION SCREEN	3-5
3.4	STOP LOGGING TASKS SCREEN	3-7
3.5	LIST SELECTED TASKS SCREEN	3-9
3.6	EVENT CLASS SELECTION SCREEN	3-10
3.7	EVENT DETAIL SELECTION SCREEN	3-12
3.8	ENABLE OR DISABLE ALL EVENT LOGGING	3-13

3.1 SETTING UP THE PROGRAM DIAGNOSTIC OPTION

Before you can log information and write data log files, you must set up the Program Diagnostic Option. This chapter contains information about how to set up the Program Diagnostic Option, including a detailed procedure for each setup screen.

The following screens are used to control diagnostic logging:

- *Clear event log* deletes all data from the log.
- *Dump Selections screen* allows you to write log data to an ASCII file, [Section 3.2](#) .
- *Task Selection screen* allows you to add tasks for which events are logged, [Section 3.3](#) .
- *Stop Logging a task Selection screen* allows you to select tasks to be removed from the list of currently logged tasks, [Section 3.4](#) .
- *Currently Selected tasks screen* lists tasks currently selected for logging, [Section 3.5](#) .
- *Event Class Selections screen* allows you to select or de-select classes or groups of related event types for logging, [Section 3.6](#) .
- *Event Detail Selections screen* allows you to select or de-select individual event types for logging, [Section 3.7](#) .

For example, you can log the following types of events:

- Start of execution of KAREL statements or TPP lines.
- Initiation and completion of robot motion.
- Triggering of condition handlers.
- Internal system events for diagnosis of system software problems.

Variable Watch Enable For R-30iB Controllers, V8.30P/05 and later

This function does not require the diagnostic option.

The variable watch function allows the KAREL programmer to find where in KAREL code any variable is set. This can be useful in determining the cause of some unexpected variable setting.

Typically this will be used with an atomic field within a structure but it can also be used for moderately sized user defined types. For a user defined type the specific field is not noted in the output only the name of the watched variable.

To enable the variable watch function:

- Set \$KARELMON.\$PROG_NAME to the name of the program that the variable you want to watch is defined in.

- Set \$KARELMON.\$FIELD_NAME to the name of the variable or variable field that you want to watch.
- Set \$KARELMON.\$WATCH_VAR = TRUE to enable watching.
- Cycle power on the controller for the new program name and variable/field name to take effect
- Run the program which causes the variable to be set to some unexpected value.
- Refer to MD:conslog.dg for the monitoring result. This will provide the routine name and line number where the variable was set.
- Any time that you change anything in \$KARELMON you must cycle power on the controller for it to take effect.

If the variable is changes a message like to following can be found in D:CONSLOG.DG

- [pgrunstp] Watch trigger (97) : task = PGWATCH, rout = SETIT line=22

If the specified variable is more than 1024 bytes long then only the first 1024 bytes will be monitored.

If the variable was set from KCL> or TP this will be noted in MD:CONSLOG.DG

- [pgrunstp] External watch trigger

Variable Cross Reference File For R-30iB Controllers, V8.30P/05 and later

This function does not require the diagnostic option.

The program cross reference listing is generated based on setting of a system variable \$KARELMON.\$XREF_ONCE. If this variable is TRUE at the NEXT startup a cross reference file will be generated FR:\VARXREF.LS. The variable is then set back to FALSE.

The VARXREF.LS file is a comma delimited file which can be imported into some other tool such as Excel. It provides any external variable reference that is found in any PC file loaded on the system.

This file contains one line of data for each cross reference consisting of the referencing program, the referenced program and the referenced variable. it does not contain any information about internal references.

3.2 DUMP SELECTIONS SCREEN

The Dump Selection Screen allows you to select a range of records to be dumped or logged. If the log is currently empty, an error message is displayed. Use [Procedure 3-1](#) to dump records to the debug log.

Procedure 3-1 Dumping Records to the Debug Log**Steps**

1. Press MENU.
2. Select TEST CYCLE.
3. Press F1, [TYPE].
4. Select Debug Ctl. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Main Menu
  Clear event log
  Dump log data
  Add a task to log
  Stop logging a task
  List selected tasks
  Set classes to log
  Change events to log
  Enable all logging
  Disable all logging
```

5. Select Dump log data. The number of records in each range is displayed on the screen. See the following screen for an example.

```
DIAGNOSTIC LOGGING
Dump Selections
  All log data
    (Will write 500 records)
  Since last power-up
    (Will write 1 records)
  Before last power-up
    (Will write 300 records)
  Reset file number
  Select RD: for files
  Select FL PY: for files
  Select MC: for files
```

6. Select the range of data to be dumped or logged.
 - To request a dump of **all event information currently in the log**, select All log data.

- To request a dump of **all event information logged since the most recent power-up**, select Since last power-up.
- To request a dump of **all event information between the last two power-ups**, select Before last power-up. This will log all events prior to the last power-down.

The information will be dumped to an ASCII file.

Note By default, successive log files are written to MC:PGDBG201.DT, MC:PGDBG202.DT, and so forth.

7. **To reset the file number in the file naming convention** , select Reset file number. The starting point is MC:PGDBG201.DT.
8. **To change the default saving device to RAM disk** , select Select RD: for files. All files will be automatically saved to RAM disk.
9. **To change the default saving device to floppy disk**, select Select FLPY: for files. All files will automatically be saved to floppy disk.
10. **To change the default saving device to memory card**, select Select MC: for files. All files will automatically be saved to a memory card.

3.3 TASK SELECTION SCREEN

The Task Selection screen allows you to select additional tasks for which you want to log information. Use [Procedure 3-2](#) to select tasks to log.

Procedure 3-2 Selecting Tasks to Log

Steps

1. Press MENU.
2. Select TEST CYCLE.
3. Press F1, [TYPE].
4. Select Debug Ctl. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
  Main Menu
Clear event log
Dump log data

Stop logging a task
List selected tasks
Set classes to log
Change events to log
Enable all logging
Disable all logging
```

5. Select Add a task to log. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Task Selection                      1/1
Use CHOICE to select task to add, or EXIT
```

6. To display a list of tasks from which you can select, press F4, [CHOICE]. You will see a screen similar to the following.

Note Selecting a program means that events will be logged for a task that has the selected program as its MAIN program.

```
1 PRG001                      5 SUB0012
2 PRG002                      6 SUB0013
3 PRG003                      7 SUB0021
4 SUB0011
DIAGNOSTIC LOGGING
Task selection
Use CHOICE to select task to add, or EXIT
```

7. Use the arrow keys to select the task for which you want events logged.
8. Press ENTER when you have chosen a task. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Task selection                      1/1
Use CHOICE to select task to add, or EXIT
                                PROG003
```

9. To request logging for the displayed task, press F3, OK.

**Caution**

Do not exit this screen by pressing PREV, F1, [TYPE], or any other key before you press F3, OK; otherwise, the selected task will not be logged.

3.4 STOP LOGGING TASKS SCREEN

The Stop Logging Tasks screen allows you to select tasks to be removed from the list of currently logged tasks. Use [Procedure 3-3](#) to stop logging tasks.

Procedure 3-3 Stop Logging Tasks

Steps

1. Press MENU.
2. Select TEST CYCLE.
3. Press F1, [TYPE].
4. Select Debug Ctl. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Main Menu
Clear event log
Dump log data
Add a task to log
Stop logging a task
List selected tasks
Set classes to log
Change events to log
Enable all logging
Disable all logging
```

5. Select Stop logging a task. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Stop Logging
Use CHOICE to select task not to log
or EXIT if none
```

6. To display a list of tasks, from which you can select, press F4, [CHOICE]. You will see a screen similar to the following.

Note When you select a program, the events will no longer be logged for any tasks that have the selected program as their MAIN program.

```
1 PRG001          5 SUB0012
2 PRG002          6 SUB0013
3 PRG003          7 SUB0021
4 SUB0011
DIAGNOSTIC LOGGING
Stop logging
```

7. Use the arrow keys to select the task for which you want to stop logging events.
8. Press ENTER when you have chosen a task. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Stop Logging                      1/1
Use CHOICE to select task not to log
or EXIT if none
PROG003
```

9. To stop logging for the task that is current displayed, press F3, OK.

**Caution**

Do not exit this screen by pressing PREV, F1, [TYPE] , or any other key before you press F3, OK; otherwise, logging for the selected task will not stop.

3.5 LIST SELECTED TASKS SCREEN

The List Selected Tasks Screen allows you to displays the tasks that are currently selected and being logged. Use [Procedure 3-4](#) to list selected tasks.

Procedure 3-4 List Selected Tasks

Steps

1. Press MENU.
2. Select TEST CYCLE.
3. Press F1, [TYPE].
4. Select Debug Ctl. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Main Menu
Clear event log
Dump log data
Add a task to log
Stop logging a task
List selected tasks
Set classes to log
Change events to log
Enable all logging
Disable all logging
```

5. Select List selected tasks. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Currently Selected Tasks
PRG003
PRG001
```

This screen displays the tasks that are currently selected and being logged.

6. Press PREV on the teach pendant to return to the main menu.

3.6 EVENT CLASS SELECTION SCREEN

The Event Class Selection Screen allows you to determine which classes of events should be logged ([Procedure 3-5](#)). Each class of events enables or disables one or more detailed event types. The following is a list of event classes and the associated event types. Refer to [Section 4.2](#) for more detailed information on the types of events that can be logged.

Note An asterisk (*) indicates logging is for internal use only.

- Execute KAREL statement
 - Execute KAREL statement
- Execute TPP line
 - Execute TPP line
- Call/return
 - KAREL or TPP routine called
 - KAREL or TPP routine returned
- Motion
 - Motion started
 - Motion planned (teach pendant motion only)
 - Motion cancel issued
 - Motion stop issued
 - Resume move
 - Motion done received
 - Motion completed normally
 - MMR received*
- Condition Handler
 - Condition handler triggered
 - Condition handler enabled
 - Condition handler disabled
- Interrupt rtn
 - Before transferring to an ISR (Interrupt Sub-Routine)
 - After transferring to an ISR (Interrupt Sub-Routine)
 - Return from ISR (Interrupt Sub-Routine)
- Task start/end

- KAREL or TPP task starts execution
- KAREL or TPP task aborts
- Packet_rcd *
 - Interpreter receives packet
- Pcode exec *
 - Starting execution of p-code
- AMR activity *
 - Normal AMR rcvd by AMGR
 - Start AMR rcvd by AMGR
 - Stop AMR rcvd by AMGR
 - AMR sent to AX
 - AMR rcvd from AX

Use [Procedure 3-5](#) to enable or disable event classes for logging.

Procedure 3-5 Enable or Disable Event Classes for Logging

Steps

1. Press MENU.
2. Select TEST CYCLE.
3. Press F1, [TYPE].
4. Select Debug Ctl. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Main Menu
Clear event log
Dump log data
Add a task to log
Stop logging a task
List selected tasks
Set classes to log
Change events to log
Enable all logging
Disable all logging
```

5. Select Set classes to log. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Event Class Selections          1/10
NO Execute KAREL line
NO Execute TPP line
NO Call/return
NO Motion
NO Condition Handler
NO Interrupt rtn
NO Task start/end
NO Packet_rcd
NO Pcode exec
```

6. Enable or disable classes of events to be logged.

- **To turn on logging for a class of events** , press F4, YES.
- **To turn off logging for a class of events** , press F5, NO.

The changes take effect immediately. The current status of whether an event is being logged is displayed to the left of the event.

Note If you specify an entire class of events, you will enable or disable logging for all event types in that class.

3.7 EVENT DETAIL SELECTION SCREEN

The Event Detail Selections Screen allows you to enable and disable logging of specific events. Use [Procedure 3-6](#) to enable or disable a specific event type for logging.

Procedure 3-6 Enable or Disable a Specific Event Type for Logging

Steps

1. Press MENU.
2. Select TEST CYCLE.
3. Press F1, [TYPE].
4. Select Debug Ctl. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Main Menu
Clear event log
Dump log data
Add a task to log
Stop logging a task
List selected tasks
Set classes to log
Change events to log
Enable all logging
Disable all logging
```

5. Select Change events to log. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Event Detail Selections          1/30
YES  Controller power-up
YES  Logging enabled
YES  Logging disabled
NO   Execute KAREL line
NO   KAREL or TPP routine called
NO   KAREL or TPP routine returned
YES  Motion started
YES  Motion planned
YES  Motion cancel issued
```

6. Enable or disable a specific event type to be logged.

- To turn on logging for an event type , press F4, YES.
- To turn off logging for an event type , press F5, NO.

The changes take effect immediately.

3.8 ENABLE OR DISABLE ALL EVENT LOGGING

The Enable/Disable All Logging Screen allows you to start or stop the automatic logging of all selected events. When the Program Diagnostic Option is installed, logging of selected events in

selected tasks is enabled. The *Disable all logging* menu selection turns off all logging. It also eliminates the overhead added when logging is enabled (refer to [Section 2.2](#), "Performance.") The *Enable all logging* menu selection can be used to start logging again.

Note Event and task selections are not changed by *Disable all logging* or *Enable all logging*.

Use [Procedure 3-7](#) to enable and disable all event logging.

Procedure 3-7 Enabling and Disabling All Event Logging

Steps

1. Press MENU.
2. Select TEST CYCLE.
3. Press F1, [TYPE].
4. Select Debug Ctl. You will see a screen similar to the following.

```
DIAGNOSTIC LOGGING
Main Menu
Clear event log
Dump log data
Add a task to log
Stop logging a task
List selected tasks
Set classes to log
Change events to log
Enable all logging
Disable all logging
```

5. Select either Enable or Disable all logging.



Caution

If you Enable or Disable all logging, you must perform a Cold Start for the change to take effect.

- If you select Enable all logging, selected events in selected tasks will automatically start being logged after you perform a Cold Start.
 - If you select Disable all logging, selected events in selected tasks will automatically stop being logged after you perform a Cold Start.
6. Perform a Cold Start.

Conditions

- All personnel and unnecessary equipment are out of the workcell.



Warning

DO NOT turn on the robot if you discover any problems or potential hazards. Report them immediately. Turning on a robot that does not pass inspection could result in serious injury.

- Visually inspect the robot, controller, workcell, and the surrounding area. During the inspection make sure all safeguards are in place and the work envelope is clear of personnel.
 - Turn the power disconnect circuit breaker to ON.
 - When the BMON screen is displayed on the teach pendant**, press and hold the SHIFT and RESET keys. Or, **on the operator panel**, press and hold RESET.
 - Release all of the keys.
- **On the operator panel** the ON LED will be illuminated, indicating robot power is on.
 - **On the teach pendant screen**, you will see a screen similar to the following.

```
UTILITIES Hints          ^
      ApplicationTool
Vx.xxx/xx              XXXX/XX

Copyright xxxx, All Rights Reserved
      FANUC CORPORATION
      FANUC America Corporation
Licensed Software: Your use constitutes
your acceptance. This product protected
by several U.S. patents.
```


LOGGING EVENTS

Contents

Chapter 4	LOGGING EVENTS	4-1
4.1	SETTING UP EVENTS	4-2
4.2	LOGGING EVENTS TO AN ASCII FILE	4-3
4.2.1	ASCII File General Event Information	4-3
4.2.2	ASCII File Specific Event Information	4-4

Events are conditions or situations that occur in a KAREL or teach pendant program while the program is running. The Program Diagnostic Option allows you to record certain events, as they take place, to help you in debugging programs. The logged events can be dumped to an ASCII file.

4.1 SETTING UP EVENTS

An event will be recorded in the log when all of the following are true:

- You have selected Enable all logging from the main menu.
- You have enabled event logging for the task.
- You have selected the event type to log either,
 - Specifically, using the Change Events To Log screen to select a specific event.
 - Generally, using the Set Classes To Log screen to select a class of events.

Types of Events that can be Logged

With the Program Diagnostic Option, the following kinds of events can be recorded.

- Controller power-up
- Logging enabled
- Logging disabled
- Execute KAREL line
- KAREL or TPP routine called
- KAREL or TPP routine returned
- Motion started
- Motion planned
- Motion cancel issued
- Motion stop issued
- Resume move
- Motion done received
- Motion completed normally
- Condition handler triggered
- Condition handler enabled
- Condition handler disabled
- Before processing ISR

- Return from KAREL ISR
- KAREL or TPP task starts execution
- KAREL or TPP task aborts
- Interpreter receives packet
- Starting execution of p-code
- Start of execution of TPP line
- MMR_RCVD (MMR received)
- MMR received
- Normal AMR received by AMGR
- Start AMR received by AMGR
- Stop AMR received by AMGR
- AMR sent to AX
- AMR received from AX

4.2 LOGGING EVENTS TO AN ASCII FILE

The last 500 events that are recorded can be written to an ASCII file. This can be done by requesting to write the logged data to a file on the Dump Selections screen (refer to [Section 3.2](#)).

4.2.1 ASCII File General Event Information

Specific information about each event is logged to the ASCII file. [Table 4–1](#) describes the following information for each event:

- **Event name:** Event description displayed in the Event Detail Selections screen.
- **When recorded:** Conditions under which the event is recorded. For example, the STRT_K_LINE event is recorded when a KAREL statement is about to be executed.
- **Information recorded:** List of values logged with each event. The following are reported for all events (except as noted).
 - Event time (seconds) since power-up. This is always a multiple of 4 milliseconds (.004 second).
 - Number of the task associated with the event.
 - Name of the KAREL routine or TPP program, in which the statement was triggered.
 - Program line number at which statement event was triggered.

- **Enabled default:** YES if logging of the specified event type is Enabled by default.
- **Comments:** Additional information, if appropriate.

4.2.2 ASCII File Specific Event Information

Table 4–1 contains each event and the corresponding information that is logged.

Table 4–1. Event Logging Information

Event Name	When Recorded	Information Recorded	Enabled Default	Comments
Controller power-up	At every controller COLD start	Routine name,line number		
		Clock time: at which power-up occurred, in "DD-MMM-YY HH:MM" format.		
		Task number will always be 1.		
Logging enabled	When logging is started. Typically, this is when the task starts.	Standard information only	YES	
Logging disabled	When logging is stopped. Typically, this is when the task ends.	Standard information only	YES	
Execute KAREL line	At beginning of execution of KAREL statement.	Number of words allocated in a task's ROUTINE stack	NO	
		Number of words allocated in a task's DATA stack		
		Number of words available for combined ROUTINE and DATA stacks		
KAREL or TPP routine called	When a KAREL routine or KAREL or TPP program is called. This is either directly or as a result of a condition handler action.	Standard information only	NO	Routine name is name of called routine; line number is line number from which call was made or interrupt occurred.

Table 4-1. Event Logging Information (Cont'd)

KAREL or TPP routine returned	When a KAREL PROGRAM or ROUTINE or TP program returns to the calling or interrupted program	Standard information only	NO	The ROUTINE name is the name of the returning program or routine; the line number is the line in the calling or interrupted program to which the task is returning, generally the same as the line number shown for in the CALL event.
Motion started	When a request to initiate a motion is issued.	Group mask	NO	For TPP motion commands, this will typically follow a PLAN-MOVE event.
		MMR address, in hexadecimal		
		MMR status, in hexadecimal; normally FFFFFFFF		
Motion planned	For TPP motion commands, this occurs typically before the motion is actually started.	Group mask	NO	Typically followed by a START-MOVE event
		MMR address, in hexadecimal		
		MMR status, in hexadecimal; normally FFFFFFFF		
Motion cancel issued	When a CANCEL statement or condition handler action is executed, a MOVE... UNTIL condition is satisfied, a CANCEL severity error is posted, or a program is aborted with a motion planned or in progress.	Group mask	NO	
		MMR address, in hexadecimal		
		MMR status, in hexadecimal; normally FFFFFFFF		

Table 4-1. Event Logging Information (Cont'd)

Motion stop issued	When a STOP statement or condition handler action is executed or a STOP severity error is posted.	Group mask	NO	
		MMR address, in hexadecimal		
		MMR status, in hexadecimal; normally FFFFFFFF		
Resume move	When a stopped motion is resumed; typically when a RESUME statement or condition handler action is executed; also when a program is CONTINUED following a STOP error condition.	Group mask	NO	
		MMR address, in hexadecimal		
		MMR status, in hexadecimal; normally FFFFFFFF		
Motion done received	When the termination type is satisfied for a KAREL or TPP motion statement is executed. For KAREL MOVE ... NOWAIT statements, this event is recorded when the motion starts.	Group mask	NO	
		MMR address, in hexadecimal		
		MMR status, in hexadecimal; normally FFFFFFFF		
Motion completed normally	When a motion is completed or cancelled. This is generally after MTN_DONE is recorded.	Group mask		
		MMR address, in hexadecimal		
		MMR status, in hexadecimal; normally FFFFFFFF		

Table 4-1. Event Logging Information (Cont'd)

Condition handler triggered	When a global condition handler triggers	Condition handler number	NO	The routine name and line number indicate the routine and line number being executed (in the task that defined the condition handler) at the time the condition handler triggered. In some cases, this might be the line following the line being executed when the condition handler is triggered.
Condition handler enabled	When a global condition handler is enabled by a ENABLE statement or condition handler action.	Condition handler number	NO	
Condition handler disabled	When a global condition handler is disabled by a DISABLE statement or condition handler action. It is not recorded when a condition handler triggers.	Condition handler number	NO	
Before processing ISR	Before calling a KAREL PROGRAM from a condition handler.	Un-pause flag; if non-zero, UNPAUSE was specified for this interrupt	NO	The routine name and line number indicate the code that was executing when the interrupt routine request was received.
		Task priority: task priority at which this routine is to execute		
		Interrupt routine priority (\$PRIORITY value)		
After processing ISR	When the interrupt routine is ready to run.	Standard information only	NO	Routine name is the name of the interrupt routine. Line number will always be 1.

Table 4-1. Event Logging Information (Cont'd)

Return from KAREL ISR	When exiting from an interrupt routine.	Standard information only	NO	Routine name and line number are the interrupt routine name and the line number from which it returned.
KAREL or TPP task starts execution	When a task selected for logging starts executing.	Standard information only	YES	Routine name is the main program name; line number is always 1.
KAREL or TPP task aborts	When a task ends.	Standard information only	YES	Routine and line number indicate the last statement executed by the task. This might be an END statement of a KAREL program or the last line of a TPP program.
Interpreter receives packet	When an interpreter task receives a packet.	Packet address	NO	Intended for system level analysis.
		Packet status		
		Request code (including sub-system code)		
		Requestor id		
		ITR-level		
Starting execution of p-code	At start of execution of KAREL p-code instruction.	Number of words on the routine stack	NO	Intended for system level analysis.
		Number of words on the data stack		
		P-code mnemonic		
Start of execution of TPP line	At start of execution of a line of a TPP program.	Number of words on the data stack	NO	
		Number of words on the routine stack		
		Number of words available for combined ROUTINE and DATA stacks		

Table 4–1. Event Logging Information (Cont'd)

MMR received	When an MMR is received back from the the motion sub-system. This occurs when the motion completes, is cancelled, or is stopped.	Group mask	NO	
		MMR address, in hexadecimal		
		MMR status, in hexadecimal; normally FFFFFFFF		
Normal AMR recvd by AMGR	An AMR is received by AMGR.	Address of AMR	NO	Intended for system level analysis.
		AMR number		
		AMR AMGR_wk		
		AMR ax_phase		
Start AMR recvd by AMGR	A start AMR request is processed by AMGR.	Address of AMR	NO	Intended for system level analysis.
		AMR number		
		AMR AMGR_wk		
		AMR ax_phase		
Stop AMR recvd by AMGR	A stop AMR request is processed by AMGR.	Address of AMR	NO	Intended for system level analysis.
		AMR number		
		AMR AMGR_wk		
		AMR ax_phase		
AMR sent to AX	When an AMR is sent from AMGR to an AX task.	Address of AMR	NO	Intended for system level analysis.
		AMR number		
		AMR AMGR_wk		
		AMR ax_phase		
AMR rcvd from AX	When an AMR is receivedack from an AX task.	Address of AMR	NO	Intended for system level analysis.
		AMR number		
		AMR AMGR_wk		
		AMR ax_phase		

EXAMPLES

Contents

Appendix A EXAMPLES A-1

 A.1 OVERVIEW A-2

 A.2 KAREL PROGRAM EXAMPLE A-2

 A.3 TEACH PENDANT PROGRAM EXAMPLE A-3

 A.4 ASCII FILE EXAMPLE A-3

A.1 OVERVIEW

This appendix contains example programs and the resulting log data, that show how the Program Diagnostic Option operates.

- [Section A.2](#) contains a KAREL program (T) which calls a teach pendant program (TPP).
- [Section A.3](#) contains the teach pendant program that is called (TTT).
- [Section A.4](#) contains the log file that is generated after running the KAREL program (T) with the following Event class selections:
 - YES Execute KAREL line
 - YES Execute TPP line
 - YES Call/return
 - YES Motion
 - YES Condition Handler
 - YES Interrupt rtn
 - YES Task start/end
 - NO Packet_rcd
 - NO Pcode exec

A.2 KAREL PROGRAM EXAMPLE

The KAREL program in [KAREL Program Example \(T.KL\)](#) is the MAIN program from which the TPP program is called and executed.

KAREL Program Example (T.KL)

```
1 program t
2 var
3   p1,p2: xyzwpr
4   i,j,k: INTEGER
5
6 routine ttt from ttt
7 routine tt
8 begin
9   i = i + 1
10 end tt
11
12 begin
13 if uninit(p1) THEN
14   p1 = curpos(0,0)
15   p2 = p1
```

```

16   p2.x = p2.x + 100
17 endif
18 condition[1]:
19   when k = 500 DO
20     k = 0
21     tt
22     enable condition[1]
23   endcondition
24 k = 0
25 i = 0
26 connect timer to k
27 enable condition[1]
28 wait for i=2
29 disconnect timer k
30 disable condition[1]
31 move to p1
32 WITH $SPEED=50 move to p2
33 ttt
34 delay 1000
35 end t

```

A.3 TEACH PENDANT PROGRAM EXAMPLE

The TPP program in [Teach Pendant Program Example \(TTT.TP\)](#) is called by the KAREL program (T) in [KAREL Program Example \(T.KL\)](#) .

Teach Pendant Program Example (TTT.TP)

```

1:J P[1] 100% FINE
2:  WAIT    0.00(sec)
3:  R[1]=0
4:  LBL[1]
5:  R[1]=R[1]+1
6:  IF R[1] <= 3,JMP LBL[1]
7:  WAIT    .50(sec)

```

A.4 ASCII FILE EXAMPLE

With logging task T enable and event classes selected as indicated in [KAREL Program Example \(T.KL\)](#) , if T is executed and the log dumped, the file show in [ASCII File Example](#) is generated.

Note For detailed information about the fields on the ASCII report shown in [ASCII File Example](#) , refer to [Section 4.2.2](#) .

ASCII File Example

Event	Time	TID	Routine	Line					
POWER-UP	10.588	1	***	***	02-JUN-95	09:31			
TASK-START	159.668	2	T	1					
LOG-ENABLE	159.668	2	T	1					
STRT-K-LINE	159.668	2	T	13	0	0	300		
STRT-K-LINE	159.680	2	T	14	0	0	300		
STRT-K-LINE	159.684	2	T	15	0	0	300		
STRT-K-LINE	159.684	2	T	16	0	0	300		
STRT-K-LINE	159.684	2	T	17	0	0	300		
STRT-K-LINE	159.684	2	T	18	0	0	300		
STRT-K-LINE	159.688	2	T	24	0	0	300		
STRT-K-LINE	159.688	2	T	25	0	0	300		
STRT-K-LINE	159.688	2	T	26	0	0	300		
STRT-K-LINE	159.688	2	T	27	0	0	300		
STRT-K-LINE	159.692	2	T	28	0	0	300		
CH-ENABLE	159.688	2	T	28	1				
CH-TRIGGER	160.176	2	T	28	1				
PRE-ISR	160.176	2	T	28	0	128	0		
CALL	160.176	2	TT	28					
IN-ISR	160.176	2	TT	1					
STRT-K-LINE	160.176	2	TT	9	0	50	250		
STRT-K-LINE	160.184	2	TT	10	0	50	250		
RETURN	160.184	2	TT	28					
RTN-INT-RTN	160.184	2	TT	28					
CH-TRIGGER	160.676	2	T	28	1				
PRE-ISR	160.676	2	T	28	0	128	0		
CALL	160.676	2	TT	28					
IN-ISR	160.680	2	TT	1					
STRT-K-LINE	160.680	2	TT	9	0	50	250		
STRT-K-LINE	160.688	2	TT	10	0	50	250		
RETURN	160.688	2	TT	28					
RTN-INT-RTN	160.688	2	TT	28					
CH-TRIGGER	161.176	2	T	28	1				
PRE-ISR	161.176	2	T	28	0	128	0		
CALL	161.176	2	TT	28					
IN-ISR	161.176	2	TT	1					
STRT-K-LINE	161.176	2	TT	9	0	50	250		
STRT-K-LINE	161.184	2	TT	10	0	50	250		
RETURN	161.184	2	TT	28					
RTN-INT-RTN	161.184	2	TT	28					
STRT-K-LINE	161.204	2	T	29	0	0	300		
STRT-K-LINE	161.204	2	T	30	0	0	300		
STRT-K-LINE	161.208	2	T	31	0	0	300		
CH-DISABLE	161.208	2	T	31	1				
START-MOVE	161.208	2	T	31	018C5F1C	FFFFFFFF	0		

MTN-DONE	161.260	2	T	31	018C5F1C	FFFFFFFF	0
STRT-K-LINE	161.260	2	T	32	0	0 300	
START-MOVE	161.260	2	T	32	018C5F70	FFFFFFFF	
248							
MMR_RCVD	161.284	2	T	31	018C5F1C	00000000	0
MTN_ENDED	161.284	2	T	31	018C5F1C	00000000	0
MTN-DONE	162.196	2	T	32	018C5F70	FFFFFFFF	0
STRT-K-LINE	162.196	2	T	33	0	0 300	
CALL	162.204	2	TTT	33			
STRT-T-LINE	162.204	2	TTT	1	0	10 290	
PLAN-MOVE	162.204	2	TTT	1	018C5FC4	FFFFFFFF	0
START-MOVE	162.204	2	TTT	1	018C5FC4	FFFFFFFF	0
MMR_RCVD	162.220	2	T	32	018C5F70	00000000	0
MTN_ENDED	162.224	2	T	32	018C5F70	00000000	0
MTN-DONE	162.536	2	TTT	1	018C5FC4	FFFFFFFF	0
STRT-T-LINE	162.536	2	TTT	2	0	10 290	
MMR_RCVD	162.564	2	TTT	1	018C5FC4	00000000	0
MTN_ENDED	162.564	2	TTT	1	018C5FC4	00000000	0
STRT-T-LINE	162.564	2	TTT	3	0	10 290	
STRT-T-LINE	162.564	2	TTT	4	0	10 290	
STRT-T-LINE	162.568	2	TTT	5	0	10 290	
STRT-T-LINE	162.568	2	TTT	6	0	10 290	
STRT-T-LINE	162.572	2	TTT	4	0	10 290	
STRT-T-LINE	162.572	2	TTT	5	0	10 290	
STRT-T-LINE	162.576	2	TTT	6	0	10 290	
STRT-T-LINE	162.580	2	TTT	4	0	10 290	
STRT-T-LINE	162.580	2	TTT	5	0	10 290	
STRT-T-LINE	162.580	2	TTT	6	0	10 290	
STRT-T-LINE	162.580	2	TTT	7	0	10 290	
RETURN	163.080	2	TTT	33			
STRT-K-LINE	163.080	2	T	34	0	0 300	
STRT-K-LINE	164.084	2	T	35	0	0 300	
TASK-END	164.108	2	T	35			
LOG-DISABLE	164.108	2	T	35			

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, and 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 milliamperere.

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 an 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 set. 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

A

ASCII file
 event log, 4–3
 file specific event information, 4–4
 general event information, 4–3
 program example, A–3

D

dump log, 3–3
dump selection screen, 3–3

E

event class selection, 3–10
event detail selection, 3–12
event log
 ASCII file, 4–3
 setting up events, 4–2
event logging
 enable/disable, 3–13
example programs, A–2

F

file specific event information
 ASCII file, 4–4

G

general event information
 ASCII file, 4–3

H

hardware
 requirements, 2–2

K

KAREL program example, A–2

L

list selected tasks, 3–9

O

overview
 example programs, A–2
 KAREL and TPP option, 1–1

P

performance diagnostic option, 2–2
program diagnostic option
 setup, 3–2
program examples, A–2
 ASCII file, A–3
 KAREL, A–2
 teach pendant program, A–3

R

R566
 order number, 1–2
requirements
 hardware, 2–2
 software, 2–2

S

screens
 dump selection, 3–3
 enable/disable all event logging, 3–13
 event class selection, 3–10
 event detail selection, 3–12
 list selected tasks, 3–9

stop logging tasks, 3–7
task selection, 3–5
setup
 events, 4–2
 program diagnostic option, 3–2
software
 requirements, 2–2

stop logging tasks, 3–7

T

task selection screen, 3–5
teach pendant program
 example, A–3