COMPLETE OUR ONLINE SURVEY

Motoman is committed to total customer satisfaction! Please give us your feedback on the technical manuals you received with your Motoman robotic solution.

To participate, go to the following website:

http://www.motoman.com/forms/techpubs.asp
Introduction

Chapter 1:

About This Document .............................................................. 1-1
Overview ................................................................................... 1-1
Visual DCI 4 Components .......................................................... 1-1
System Requirements ................................................................. 1-2
Reference to Other Documentation ............................................ 1-2
Customer Service ....................................................................... 1-2

Safety

Chapter 2:

Standard Conventions .............................................................. 2-1
General Safeguarding Tips ......................................................... 2-2
Mechanical Safety Devices ......................................................... 2-2
Installation Safety ....................................................................... 2-2
Programming Safety .................................................................... 2-3
Operation Safety .......................................................................... 2-4
Maintenance Safety ..................................................................... 2-4

Installation

Chapter 3:

Installing the Software ............................................................... 3-1
Installing the Hardware Key ....................................................... 3-2
Controller Setup ......................................................................... 3-2
XRC Controller ........................................................................... 3-2
XRC Cable Connections .............................................................. 3-2
XRC Parameters ........................................................................ 3-3
MRC Controller ........................................................................... 3-4
MRC Cable Connections .............................................................. 3-4
MRC Parameters ........................................................................ 3-5
ERC Controller ........................................................................... 3-6
ERC Cable Connections .............................................................. 3-6
ERC Parameters ........................................................................ 3-6
Building a Data Cable ................................................................. 3-7
Null Modem Pin Connections ..................................................... 3-8
Removing Visual DCI 4 .............................................................. 3-8
## Contents

### Basic Operation

Chapter 4: Basic Operation

- Main Window ................................................................. 4-1
- Specifying Robot Program Path ........................................... 4-1
- Variable Editor ............................................................... 4-1
- Assigning Variable Labels .................................................. 4-2
- Saving Configuration Files .................................................. 4-2
- Opening a New Configuration File ......................................... 4-2
- Opening an Existing Configuration File .................................. 4-2
- Setting Communications Parameters ...................................... 4-2
- Communications Setup Options .......................................... 4-3

### Variable Editor

Chapter 5: Variable Editor

- Index Variable ................................................................... 5-1
- Creating/Modifying the Robot Program Structure ..................... 5-2
- Saving Variables to the PC .................................................. 5-2
- Loading Variables from the PC ............................................. 5-3
- Loading and Saving Job Files ............................................... 5-4
- Using the Variable Editor ................................................... 5-4
- Closing a Visual DCI Session .............................................. 5-5

### Techniques

Chapter 6: Techniques

- Optimizing for Cycle Time .................................................. 6-1
- Optimizing for Variable Latency ............................................ 6-1
- Optimizing Memory Usage ................................................... 6-1

### OLE Commands

Chapter 7: OLE Commands

- Properties ........................................................................... 7-1
- Methods ............................................................................. 7-2
- Events ............................................................................... 7-2
- Examples ............................................................................ 7-2
- Starting a Copy of Visual DCI .............................................. 7-2
- Enable/Disable Communication ............................................ 7-3
- Change the Configuration File .............................................. 7-3
- Retrieving Variable Data from Visual DCI ............................ 7-3
- Saving Variable Data to Visual DCI ...................................... 7-3
- Hiding or Showing the Session of Visual DCI ........................ 7-3
- Determining if DCI is Enabled to Communicate ...................... 7-3
- Determining the Present File Path ........................................ 7-4
- Changing the Present File Path ............................................ 7-4
Troubleshooting

Chapter 8:
DCI Stops Functioning Normally ..................................................... 8-1
Visual DCI 4 Does Not Work After Installation ............................... 8-2
Applications ...................................................................................... 8-2

Index
Chapter 1: Introduction

Visual Data Communication by Instruction (DCI) is an on-line control system developed for the Motoman family of robots. Visual DCI 4 includes a built-in variable editor which lets the user modify any robot variable including byte, integer, double, real, and position variables. These modifications are performed while the robot is in cycle. Visual DCI 4 also allows the user to upload and download complete robot programs which can be simultaneously modified using a basic ASCII editor like NotePad.

About This Document

Please read this manual for safety precautions, installation instructions, and other important information before using Visual DCI 4.

This manual is intended as an introduction and overview for personnel who are familiar with the operation of their Motoman robot model, and Microsoft® Windows®/PC usage. This manual contains the following chapters:

Chapter 1: Introduction
Provides general information about Visual DCI 4, a list of reference documents, and customer service information.

Chapter 2: Safety
Provides information regarding the safe use and operation of Visual DCI 4.

Chapter 3: Installation
Provides instructions for Visual DCI 4 software installation, hardware key installation, and first time use.

Chapter 4: Operation
Provides step by step instructions and examples for using Visual DCI 4.

Appendices
Provides detailed descriptions of various Visual DCI features including: bone and more.

Glossary
Provides descriptions of various Visual DCI terms.

Overview
Visual DCI includes the following features:
- Variable Editor
- OLE

Visual DCI 4 Components
Before you start the Visual DCI installation, make sure your Visual DCI package includes all of the items listed below:
- One RS-232 serial interface cable, 9 pin to 9 pin Null Modem (Motoman Part Number 147235-1)
- Motoman CD Browser® CD-ROM
- Visual DCI 4 User’s manual
- Hardware Key (Part number differs depending on software order)
1-2 Introduction

System Requirements
Computer - IBM PC or compatible
Processor - Intel Pentium 200
Memory - 8 MB
Hard Disk - 6 MB minimum available
CD ROM Drive
Monitor - VGA Graphics
COM Port - One RS-232 port capable of operating at 9600 baud for each copy of Visual DCI 4.
Interface Cable - One RS-232 serial interface cable for each copy of Visual DCI 4 (Supplied with Visual DCI 4)
Operating System - MicroSoft Windows 98, NT or Windows XP
Controller Software - Controller software must have the DCI Communications option installed.
Visual DCI 4's on-line help requires Microsoft Internet Explorer 3.02 or later. Internet 4.0 or later is recommended and provided on the Motoman CD Browser.

Reference to Other Documentation
For additional information, refer to the following:
• Operator's Manual for your application
• Manipulator Manual for your robot model

Customer Service
If you need technical assistance, contact the Motoman service staff at (937) 847-3200. Please have the following information ready before you call:
• Visual DCI version
• Operating system (Microsoft Windows 98/NT/2000/XP)
• System configuration (hard disk capacity, memory, software, etc.)
• List of all software installed after shipment from Motoman
• Description of difficulty (make note of any error messages)
• Application specific information including robot jobs, process details etc...
Chapter 2: Safety

It is the purchaser’s responsibility to ensure that all local, county, state, and national codes, regulations, rules, or laws relating to safety and safe operating conditions for each installation are met and followed.

We suggest that you obtain and review a copy of the ANSI/RIA National Safety Standard for Industrial Robots and Robot Systems. This information can be obtained from the Robotic Industries Association by requesting ANSI/RIA R15.06. The address is as follows:

Robotic Industries Association
900 Victors Way
P.O. Box 3724
Ann Arbor, Michigan 48106
TEL: (734) 994-6088
FAX: (734) 994-3338

Ultimately, the best safeguard is trained personnel. The user is responsible for providing personnel who are adequately trained to operate, program, and maintain the robot cell. The robot must not be operated by personnel who have not been trained!

We recommend that all personnel who intend to operate, program, repair, or use the robot system be trained in an approved Motoman training course and become familiar with the proper operation of the system.

This safety section addresses the following:
• Standard Conventions
• General Safeguarding Tips
• Mechanical Safety Devices
• Installation Safety
• Programming Safety
• Operation Safety
• Maintenance Safety

Standard Conventions

This manual includes information essential to the safety of personnel and equipment. As you read through this manual, be alert to the four signal words:
• DANGER
• WARNING
• CAUTION
• NOTE

Pay particular attention to the information provided under these headings which are defined below (in descending order of severity).

⚠️ **DANGER!**

Information appearing under the DANGER! caption concerns the protection of personnel from the immediate and imminent hazards that, if not avoided, will result in immediate, serious personal injury or loss of life in addition to equipment damage.

⚠️ **WARNING!**

Information appearing under the WARNING! caption concerns the protection of personnel and equipment from potential hazards that can result in personal injury or loss of life in addition to equipment damage.
CAUTION!

Information appearing under the CAUTION! caption concerns the protection of personnel, equipment, software, and data from hazards that can result in minor personal injury or equipment damage.

NOTE: Information appearing in a NOTE caption provides additional information which is helpful in understanding the item being explained.

General Safeguarding Tips

All operators, programmers, plant and tooling engineers, maintenance personnel, supervisors, and anyone working near the robot must become familiar with the operation of this equipment. All personnel involved with the operation of the equipment must understand potential dangers of operation. General safeguarding tips are as follows:

- Improper operation can result in personal injury and/or damage to the equipment. Only trained personnel familiar with the operation of this robot, the operator’s manuals, the system equipment, and options and accessories should be permitted to operate this robot system.
- Do not enter the robot cell while it is in automatic operation. Programmers must have the teach pendant when they enter the robot cell.
- Improper connections can damage the robot. All connections must be made within the standard voltage and current ratings of the robot I/O (Inputs and Outputs).
- The robot must be placed in Emergency Stop (E-STOP) mode whenever it is not in use.
- In accordance with ANSI/RIA R15.06, section 6.13.4 and 6.13.5, use lockout/tagout procedures during equipment maintenance. Refer also to Section 1910.147 (29CFR, Part 1910), Occupational Safety and Health Standards for General Industry (OSHA).

Mechanical Safety Devices

The safe operation of the robot, positioner, auxiliary equipment, and system is ultimately the user’s responsibility. The conditions under which the equipment will be operated safely should be reviewed by the user. The user must be aware of the various national codes, ANSI/RIA R15.06 safety standards, and other local codes that may pertain to the installation and use of industrial equipment. Additional safety measures for personnel and equipment may be required depending on system installation, operation, and/or location. The following safety measures are available:

- Safety fences and barriers
- Light curtains
- Door interlocks
- Safety mats
- Floor markings
- Warning lights

Check all safety equipment frequently for proper operation. Repair or replace any non-functioning safety equipment immediately.

Installation Safety

Safe installation is essential for protection of people and equipment. The following suggestions are intended to supplement, but not replace, existing federal, local, and state laws and regulations. Additional safety measures for personnel and equipment may be required depending on system installation, operation, and/or location. Installation tips are as follows:

- Be sure that only qualified personnel familiar with national codes, local codes, and ANSI/RIA R15.06 safety standards are permitted to install the equipment.
- Identify the work envelope of each robot with floor markings, signs, and barriers.
• Position all controllers outside the robot work envelope.
• Whenever possible, install safety fences to protect against unauthorized entry into the work envelope.
• Eliminate areas where personnel might get trapped between a moving robot and other equipment (pinch points).
• Provide sufficient room inside the workcell to permit safe teaching and maintenance procedures.

Programming Safety
All operators, programmers, plant and tooling engineers, maintenance personnel, supervisors, and anyone working near the robot must become familiar with the operation of this equipment. All personnel involved with the operation of the equipment must understand potential dangers of operation. Programming tips are as follows:
• Any modifications to PART 1 of the controller PLC can cause severe personal injury or death, as well as damage to the robot! Do not make any modifications to PART 1. Making any changes without the written permission of Motoman will VOID YOUR WARRANTY!
• Some operations require standard passwords and some require special passwords. Special passwords are for Motoman use only. YOUR WARRANTY WILL BE VOID if you use these special passwords.
• Back up all programs and jobs onto a floppy disk whenever program changes are made. To avoid loss of information, programs, or jobs, a backup must always be made before any service procedures are done and before any changes are made to options, accessories, or equipment.
• The concurrent I/O (Input and Output) function allows the customer to modify the internal ladder inputs and outputs for maximum robot performance. Great care must be taken when making these modifications. Double-check all modifications under every mode of robot operation to ensure that you have not created hazards or dangerous situations that may damage the robot or other parts of the system.
• Improper operation can result in personal injury and/or damage to the equipment. Only trained personnel familiar with the operation, manuals, electrical design, and equipment interconnections of this robot should be permitted to operate the system.
• Inspect the robot and work envelope to be sure no potentially hazardous conditions exist. Be sure the area is clean and free of water, oil, debris, etc.
• Be sure that all safeguards are in place.
• Check the E-STOP button on the teach pendant for proper operation before programming.
• Carry the teach pendant with you when you enter the workcell.
• Be sure that only the person holding the teach pendant enters the workcell.
• Test any new or modified program at low speed for at least one full cycle.
• Incorrect use of MotoCal for robot calibration may cause incorrect robot calibration resulting in robot crash and/or personal injury or death.
• Incorrect use of MotoCal for job Filtering or Downloading, may cause robot crash and/or personal injury or death.
Operation Safety

All operators, programmers, plant and tooling engineers, maintenance personnel, supervisors, and anyone working near the robot must become familiar with the operation of this equipment. All personnel involved with the operation of the equipment must understand potential dangers of operation. Operation tips are as follows:

- Be sure that only trained personnel familiar with the operation of this robot, the operator’s manuals, the system equipment, and options and accessories are permitted to operate this robot system.
- Check all safety equipment for proper operation. Repair or replace any non-functioning safety equipment immediately.
- Inspect the robot and work envelope to ensure no potentially hazardous conditions exist. Be sure the area is clean and free of water, oil, debris, etc.
- Ensure that all safeguards are in place.
- Improper operation can result in personal injury and/or damage to the equipment. Only trained personnel familiar with the operation, manuals, electrical design, and equipment interconnections of this robot should be permitted to operate the system.
- Do not enter the robot cell while it is in automatic operation. Programmers must have the teach pendant when they enter the cell.
- The robot must be placed in Emergency Stop (E-STOP) mode whenever it is not in use.
- This equipment has multiple sources of electrical supply. Electrical interconnections are made between the controller, external servo box, and other equipment. Disconnect and lockout/tagout all electrical circuits before making any modifications or connections.
- All modifications made to the controller will change the way the robot operates and can cause severe personal injury or death, as well as damage the robot. This includes controller parameters, ladder parts 1 and 2, and I/O (Input and Output) modifications. Check and test all changes at slow speed.

Maintenance Safety

All operators, programmers, plant and tooling engineers, maintenance personnel, supervisors, and anyone working near the robot must become familiar with the operation of this equipment. All personnel involved with the operation of the equipment must understand potential dangers of operation. Maintenance tips are as follows:

- Do not perform any maintenance procedures before reading and understanding the proper procedures in the appropriate manual.
- Check all safety equipment for proper operation. Repair or replace any non-functioning safety equipment immediately.
- Improper operation can result in personal injury and/or damage to the equipment. Only trained personnel familiar with the operation, manuals, electrical design, and equipment interconnections of this robot should be permitted to operate the system.
- Back up all your programs and jobs onto a floppy disk whenever program changes are made. A backup must always be made before any servicing or changes are made to options, accessories, or equipment to avoid loss of information, programs, or jobs.
- Do not enter the robot cell while it is in automatic operation. Programmers must have the teach pendant when they enter the cell.
- The robot must be placed in Emergency Stop (E-STOP) mode whenever it is not in use.
- Be sure all safeguards are in place.
- Use proper replacement parts.
- This equipment has multiple sources of electrical supply. Electrical interconnections are made between the controller, external servo box, and other equipment. Disconnect and lockout/tagout all electrical circuits before making any modifications or connections.
• All modifications made to the controller will change the way the robot operates and can cause severe personal injury or death, as well as damage the robot. This includes controller parameters, ladder parts 1 and 2, and I/O (Input and Output) modifications. Check and test all changes at slow speed.

• Improper connections can damage the robot. All connections must be made within the standard voltage and current ratings of the robot I/O (Inputs and Outputs).
Chapter 3: Installation

This chapter provides instructions for Visual DCI software installation, hardware key installation, and first time use.

Installing the Software

Visual DCI 4 is part of the MotoSoft family of software. MotoSoft software is provided on a single CD-ROM with a browser/installer utility. (Motoman CD-ROM Browser, P/N 141720-1). Please refer to the CD-Browser for detailed installation instructions.

Visual DCI 4 is installed in c:\Program Files\Motoman\VisualDCI4. To install Visual DCI, proceed as follows:

1. Insert the CD-ROM browser into your CD-ROM drive.
2. The setup program executes automatically. 
   Note: Setup executes automatically when the CD is inserted into the CD-ROM drive unless autoexecute has been disabled on your computer. If the setup program does not autoexecute, select demo32.exe from the CD-ROM Open Properties dialog. Demoshield Player appears and prompts you for a .dbd file. Double-click *.dbd to begin the install process.
3. The introduction screen appears. Click Next to continue.
4. Click the Communication Software button. The Motoman Software License Agreement page appears.
5. Click the Accept button to agree to the terms outlined in the scrolling window. The software selection screen appears.

Note: Although you may install any of the software titles on the CD-ROM, you must install a hardware security key for each software bundle purchased. If the hardware key is not installed, you will be unable to run the software.
3-2 | Installation

**Note:** Although you may install any of the software titles on the CD-ROM, you must install a hardware security key for each software bundle purchased. If the hardware key is not installed, you will be unable to run the software.

6. Click the **Visual DCI 4** button to begin the installation process. The Visual DCI 4 install wizard begins.

7. Follow the Visual DCI 4 install wizard instructions as it guides you through the installation process.

### Installing the Hardware Key

The hardware key supplied with Visual DCI 4 must be installed on your computer or Visual DCI 4 will not function properly. The hardware key attaches to the computer's parallel port. This port is commonly used to connect printers and other peripheral devices to your computer. To attach the hardware key, proceed as follows:

1. Disconnect any device currently connected to your computer’s parallel port.
2. Carefully insert the hardware key into the parallel port. If the key does not fit, do not force it. The key should fit snugly but does not require significant force to insert.
3. Connect your peripheral cable to the free end of the hardware key. The key will not interfere with the operation of your printer or other peripheral devices.

If you are using two or more MotoSoft products that require the use of different hardware keys, you can “stack” the keys (connect in series).

### Controller Setup

Before you can begin using Visual DCI 4, you must establish a physical connection between the robot controller and remote PC, and set transmission parameters for your system. To complete setup, complete the appropriate steps for your controller.

#### XRC Controller

**XRC Cable Connections**

1. Turn the XRC main power off. Remove one cable access plate from the side of the XRC cabinet and route the female end of the data cable into the cabinet.

   **Note:** Use a sealed strain relief in the plate to secure the data cable. A sealed strain relief will prevent debris from entering the cabinet and reduce the possibility of damaging components inside the XRC cabinet if the data cable is accidentally pulled.

2. Open the XRC cabinet door and locate the XCP01 board.

3. Connect the data cable to the 9-pin serial interface connector (CN03) on the XCP01 board.
4. Connect the other end of the data cable to the COM port of your computer.

5. Run the cable where it will not present a tripping hazard.

**XRC Parameters**

Several XRC transmission parameters must be setup before Visual DCI 4 will function. For more information on changing parameter values, refer to the XRC Operator's manual for your application.

<table>
<thead>
<tr>
<th>Table 3-1  XRC Parameters</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Parameter</strong></th>
<th><strong>Contents</strong></th>
<th><strong>Initial Value</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>RS000</td>
<td>2 BSC Like (data transmission function)</td>
<td>2</td>
</tr>
<tr>
<td>RS030</td>
<td>Number of data bits 7:7 (bit)</td>
<td>8</td>
</tr>
<tr>
<td>RS031</td>
<td>Number of stop bits 0:1 (bit)</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1:1.5</td>
<td>2:2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Parameter</strong></th>
<th><strong>Contents</strong></th>
<th><strong>Initial Value</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>RS032</td>
<td>Parity specification</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0:No specification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1:Odd parity</td>
<td>2:Even parity</td>
</tr>
<tr>
<td>RS033</td>
<td>Transmission Rate</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>1:150 (baud rate)</td>
<td>2:300</td>
</tr>
<tr>
<td></td>
<td>3:600</td>
<td>4:1200</td>
</tr>
<tr>
<td></td>
<td>5:2400</td>
<td>6:4800</td>
</tr>
<tr>
<td></td>
<td>7:9600</td>
<td>8:19200</td>
</tr>
<tr>
<td>RS034</td>
<td>Timer A</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Sequence monitoring timer. Serves as protection against invalid or no response. Unit:0.1 sec. (setting range 0 to 100)</td>
<td></td>
</tr>
<tr>
<td>RS035</td>
<td>Timer B</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>Text reception monitoring timer. Serves as protection against no response of text end character. Unit:0.1 sec. (setting range 0 to 255)</td>
<td></td>
</tr>
<tr>
<td>RS036</td>
<td>Retry 1</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Number of resendings of a sequence character at an invalid or no response (setting range: 0 to 30)</td>
<td></td>
</tr>
</tbody>
</table>
MRC Controller

MRC Cable Connections

If you are using an MRC controller, there are two ways to connect the controller to your computer.

- External connection - recommended if using a portable computer to run Visual DCI 4 between different controllers
- Internal connection - recommended if the computer running Visual DCI is permanently attached to the controller.

Depending on which connection you use, it is very important to correctly set the parameters for the RS-232 or MCP01 connection. Refer to Table 3-2 for correct parameter values.

### External Connection

1. Lift the subpanel cover on the playback box, and connect the 25-pin data cable connector to the RS-232 serial interface connector.

   **Note:** A 25-to-9 pin adapter may be required to make this connection. This adapter can be obtained from any computer store.

2. Connect the other end of the data cable to the COM port of your computer.

### Internal Connection

1. Turn the MRC main power off. Remove one of the cable access plates from the side of the MRC cabinet and route the female end of the data cable into the cabinet.

   **Note:** Use a sealed strain relief in the plate to secure the data cable. A sealed strain relief will prevent debris from entering the cabinet and reduce the possibility of damaging components inside the MRC cabinet if the data cable is accidentally pulled.

2. Open the MRC cabinet door and locate the MCP01 board.

3. Connect the data cable to the 9-pin serial interface connector on the MCP01 board.

---

### Table 3-1 XRC Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Contents</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS037</td>
<td>Retry 2  Number of resendings of a text at a block check error (reception of NAK). Setting range: 0 to 10</td>
<td>3</td>
</tr>
<tr>
<td>RS038</td>
<td>Block check method 0: Check sum</td>
<td>0</td>
</tr>
</tbody>
</table>
4. Connect the other end of the data cable to the COM port of your computer.
5. Run the cable where it will not present a tripping hazard.

MRC Parameters
Several MRC transmission parameters must be setup before Visual DCI 4 will function. For more information on changing parameter values, refer to the MRC Operator’s manual for your application.

External RS-232 connection
If using the external RS-232 connection, set the RS000 and RS001 parameters as follows:
RS000 : 2
RS001 : 3

Internal MCP01 connection
If using the internal MCP01 connection, set the RS000 and RS001 parameters as follows:
RS000 : 3
RS001 : 2

Table 3-2  MRC Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Contents</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS030</td>
<td>Number of data bits 7:7 (bit)</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RS031</td>
<td>Number of stop bits 0:1 (bit)</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1:1.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2:2</td>
<td></td>
</tr>
<tr>
<td>RS032</td>
<td>Parity specification</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0:No specification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1:Odd parity</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2:Even parity</td>
<td></td>
</tr>
<tr>
<td>RS033</td>
<td>Transmission Rate</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>1:150 (baud rate)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2:300</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3:600</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4:1200</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5:2400</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6:4800</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7:9600</td>
<td></td>
</tr>
<tr>
<td>RS034</td>
<td>Timer A</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Sequence monitoring timer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Serves as protection against invalid or no</td>
<td></td>
</tr>
<tr>
<td></td>
<td>response.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unit:0.1 sec. (setting range 0 to 100)</td>
<td></td>
</tr>
</tbody>
</table>
ERC Controller

ERC Cable Connections

1. Connect the 25-pin data cable connector to the RS-232 serial interface connector on the front control subpanel of the ERC.

   **Note:** A 25-to-9 pin adapter may be required to make this connection. This adapter can be obtained from any computer store.

2. Connect the other end of the data cable to the COM port of your computer.

ERC Parameters

Several ERC transmission parameters must be setup before Visual DCI 4 will function. For more information on changing parameter values, refer to the ERC Operator’s manual for your application.

**Table 3-3 ERC Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Contents</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS00</td>
<td>Number of data bits 7:7 (bit) 8:8 (bit)</td>
<td>8</td>
</tr>
</tbody>
</table>
Building a Data Cable

You can use the serial interface cable supplied with your Visual DCI package, or you can make a cable to suit your specific needs. Figure 3-3 illustrates the required cable connections.

Note: You do not use all of the pins on the serial interface cable connectors. Any unused pins can be left open.
3-8 Installation

Null Modem Pin Connections

<table>
<thead>
<tr>
<th>25 Pin</th>
<th>9 Pin</th>
<th>9 Pin</th>
<th>25 Pin</th>
</tr>
</thead>
<tbody>
<tr>
<td>FG (Frame Ground)</td>
<td>1</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>TD (Transmit Data)</td>
<td>2</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>RD (Receive Data)</td>
<td>3</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>RTS (Request To Send)</td>
<td>4</td>
<td>7</td>
<td>-</td>
</tr>
<tr>
<td>CTS (Clear To Send)</td>
<td>5</td>
<td>8</td>
<td>-</td>
</tr>
<tr>
<td>SG (Signal Ground)</td>
<td>7</td>
<td>5</td>
<td>-</td>
</tr>
<tr>
<td>DSR (Data Set Ready)</td>
<td>6</td>
<td>6</td>
<td>-</td>
</tr>
<tr>
<td>DTR (Data Terminal Ready)</td>
<td>20</td>
<td>4</td>
<td>-</td>
</tr>
</tbody>
</table>

Removing Visual DCI 4

To uninstall Visual DCI, proceed as follows:

1. Double-click Add/Remove Programs in the Windows Control Panel.
2. Select "Visual DCI" from the list and click Add/Remove.
3. Click Yes to confirm deletion of the Visual DCI program files.
Chapter 4: Basic Operation

This chapter provides detailed operation information for Visual DCI 4, including detailed step-by-step instructions. To launch Visual DCI 4, select Visual DCI 4 from the Windows Start menu. Visual DCI 4 is located in the Motoman program group unless you specified another group during installation.

Main Window

When you first start Visual DCI 4, the main window appears.

**Robot Program Path** - displays the directory path where files are loaded from and saved.

**Variable Editor** - opens the Variable Editor window. From this window you can assign variable labels to any or all variables.

**Communication** - manually Starts or Stops communication between robot controller and PC.

**Communication Status Windows** - displays information about the COM port. COM port status and settings are displayed. The left hand window displays COM port status; open or closed. The right hand window displays COM port settings; Baud Rate, Parity, Data Length, and Stop Bits.

Specifying Robot Program Path

The Visual DCI 4 communications interface with the robot controller allows the exchange of robot program files. Through Visual DCI 4, the controller can access remote and networked devices for off-line storage and retrieval of job files. It also makes the program files available to other applications (for example, an ASCII text editor) while the robot remains in cycle. To specify a program file path to use when loading and saving robot job files, proceed as follows:

1. In the main window, select Settings > Robot Program Path.
2. Navigate to desired folder to store robot files.
3. Click OK.

Variable Editor

Visual DCI 4 allows you to assign variable labels to any or all variables. Using variable labels does not affect the robot program, Visual DCI 4 only uses the labels internally. This makes variables more easily recognizable.

For example, if the value of variable B1 determines which job the robot performs, we might rename this variable "Robot Seq." Visual DCI 4 now recognizes any reference to variable B1 and substitutes the label "Robot Seq."

Visual DCI 4 also lets you save variable label assignments in the configuration .dci files. You can then retrieve the appropriate file for use as required. By doing this, you can maintain a configuration file for each robot program.
Assigning Variable Labels
To assign variable labels, proceed as follows:
1. Click the Variable Editor button on the main window. The Variable Editor window appears.
2. Select the variable type tab you would like to assign labels.
3. Click in the Name field of the variable number you would like to label and simply type in the label you would like to use.

Changes are automatically saved to the default configuration file. However, if you want to keep the assigned labels for use in future sessions, you should save and rename them to a configuration file. Otherwise, label assignments may accidently be overwritten during future sessions.

Saving Configuration Files
To save variable label assignments to a configuration file, proceed as follows:
1. In the main window, select File > Save as, or use the keyboard shortcut Ctrl+S.
2. Specify the new file name and location and click Save.

Opening a New Configuration File
To create a new configuration file, proceed as follows:
1. In the main window, select File > New, or use the keyboard shortcut Ctrl+N.
2. Specify the file name and location and click Save.

Opening an Existing Configuration File
To open an existing configuration file, proceed as follows:
1. In the main window, select File > Open, or use the keyboard shortcut Ctrl+O.
2. Navigate to the desired dci file and click Open.

Setting Communications Parameters
In order for the controller and PC to work together, the communications parameters set up on each device must match. Typically, you need to configure the Visual DCI 4 communications parameters only if:
- Controller parameters are changed.
- Visual DCI 4 is connected to a different controller.

Note: Visual DCI 4 uses the same default communications parameters as the controller. If these parameters have been changed on the controller or if you want to use a communications port on the PC other than COM 1, follow the instructions below.
Note: Visual DCI 4 uses the communications settings chosen within DCI. The program ignores any communications settings selected through the Windows Control Panel.

To configure the communication parameters, proceed as follows:

1. In the main window, select Settings > Communication. The Communications Setup window appears.

2. Select the desired values from the pull down menus for each setting.

   Note: To restore the port setup to the factory default settings, click the Restore Defaults button.

3. Click OK.
Chapter 5: Variable Editor

This section introduces you to the variable editor. It contains:
- A brief overview of the mechanics of data transfer between controller and Visual DCI 4
- Information on creation and/or modification of required robot program structure
- Detailed instructions for use of variable editor.

Index Variable

The variable names used to identify different variables have two components. A prefix letter identifies the type of variable; B for byte, I for integer, and R for real. The second component of the variable name is a variable index with a value ranging from 00 to 99.

The controller commands LOADV and SAVEV do not send the variable index to the PC. The PC receives only the type of variable (B, I, and R) and its value. As a result, the PC has no way of knowing which of a particular type of variable is being loaded or saved. Before the PC can properly store or retrieve a variable, the controller must upload the variable index to the PC.

In the case of P-variables, all P-variables being used in an application must be uploaded before they can be downloaded. You must use an Index Variable to identify the variable index. Any byte variable can be used as the Index Variable.

In the sample programs listed in this section, B(00) is used as the Index Variable. Every time the controller loads or saves a variable, it must first issue a SET B00 command that specifies the appropriate variable index, followed by a SAVEV B00. This gives the PC the variable index of the variable to be saved or loaded. When the controller next gives a SAVEV or LOADV command for a particular type of variable, the PC recognizes the specific variable identified by the Index Variable, B(00).

Note: • When stepping through or developing a program, make sure that for each variable you want the program to load or save you first set and save the Index Variable. Otherwise, Visual DCI 4 and the controller will run out of synchronization and cause various errors.

• You do not need to specify an Index Variable to load or save jobs. This requirement applies only to variables.

• For additional information on variables and variable types, refer to your Programming or Operator’s Manual.
Creating/Modifying the Robot Program Structure

Visual DCI lets you modify program variables while a robot is in cycle. However, it is the controller that directs the operation of the robot. The controller must issue requests to the PC to download variables to the controller.

Saving Variables to the PC

Variables can be saved from the controller to the PC as often as required. The following example shows how to save the desired variables from the controller to the PC. Comments regarding specific program lines appear to the right of the program line. These comments do not appear in the actual program listing.

### Table 5-1 MAIN.JBI

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOP</td>
<td></td>
</tr>
<tr>
<td>CALL JOB: SAVEV</td>
<td>Call job to download variables from PC</td>
</tr>
<tr>
<td>*mainloop</td>
<td>Identifies start of subroutine</td>
</tr>
<tr>
<td>SET B00 1</td>
<td>Index = 1</td>
</tr>
<tr>
<td>SAVEV B00</td>
<td>B(01) is the variable being saved or uploaded to the PC.</td>
</tr>
<tr>
<td>CALL JOB: DC1 IF B01 = 1</td>
<td>These three lines call job based on value of variable B(01).</td>
</tr>
<tr>
<td>CALL JOB: DC2 IF B01 = 2</td>
<td></td>
</tr>
<tr>
<td>CALL JOB: DC3 IF B01 = 3</td>
<td></td>
</tr>
<tr>
<td>JUMP *mainloop</td>
<td>Jumps back to beginning of subroutine, can include conditional statement.</td>
</tr>
<tr>
<td>END</td>
<td></td>
</tr>
</tbody>
</table>

### Table 5-2 SAVEV.JBI

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOP</td>
<td></td>
</tr>
<tr>
<td>SET B00 1</td>
<td>Index = 1</td>
</tr>
<tr>
<td>SAVEV B00</td>
<td>B(00) is an index counter whose value is the index of the variable being saved - tells the PC which variable is to be loaded.</td>
</tr>
<tr>
<td>SET B00 2</td>
<td>Index = 2</td>
</tr>
<tr>
<td>SAVEV B00</td>
<td>Upload index value</td>
</tr>
<tr>
<td>SET B02 0</td>
<td>Set B(02) = 0</td>
</tr>
<tr>
<td>SAVEV B02</td>
<td>Upload B(02) to PC</td>
</tr>
<tr>
<td>SET B00 3</td>
<td>Index = 3</td>
</tr>
<tr>
<td>SAVEV B00</td>
<td>Upload index value</td>
</tr>
<tr>
<td>SET B03 0</td>
<td>Set B(03) = 0</td>
</tr>
<tr>
<td>SAVEV B03</td>
<td>Upload B(03) to PC</td>
</tr>
<tr>
<td>SET B00 1</td>
<td>Index = 1</td>
</tr>
<tr>
<td>SAVEV B00</td>
<td>Upload index value</td>
</tr>
</tbody>
</table>
Loading Variables from the PC

Variables can be downloaded from the PC to the controller as often as required. The following example shows how to download the desired variables from the PC to the controller.

**Table 5-2  SAVEV JBI**

| SAVEV P001  | Upload P(001) to PC |
| SET B00 0  | Index=0 |
| SAVEV B00  | Upload index value |
| SAVEV P000  | Upload P(001) to PC |
| SET B00 1  | Index=1 |
| SAVEV B00  | Upload index value |
| SET I01 1000  | Set I(01)=1000 |
| SAVEV I01  | Upload I(01) |
| SET B00 2  |
| SET D02 2000000  |
| SAVEV B00  |
| SAVEV D02  |
| SET B00 3  |
| SAVEV B00  |
| SET R03 300  |
| SAVEV R03  |
| SAVEJ JOB: DC3 JBI  |
| LOADJ JOB: DC3 JBI  |
| RETURN  |
| END  |

**Table 5-3  LOADV JBI**

| NOP  |
| SET B00 1  | Set index variable to 1 |
| SAVEV B00  | Upload index variable to PC |
| LOADV B01  | Download B(01) from PC to the controller |
| SET B00 1  | Set index variable to 1 |
| SAVEV B00  | Upload index variable to PC |
| LOADV P001  | Download P(01) from PC to the controller |
| SET B00 2  | Set index variable to 2 |
| SAVEV B00  | Upload index variable to PC |
| LOADV B02  | Download B(02) from PC to the controller |
| SET B00 3  |
| SAVEV B00  |
| LOADV B03  |
| SET B00 0  |
| SAVEV B00  |
| LOADV P000  |
| JUMP *11 IF B03 <>3  |
Loading and Saving Job Files

The controller can use Visual DCI 4 to access the remote PC for off-line storage and retrieval of job files. This capacity helps to conserve online storage. It also allows the user to edit jobs in an ASCII text editor (such as NotePad) while Visual DCI 4 runs in background. With an ASCII editor and Visual DCI 4 the user can access and edit job files without taking the robot out of cycle.

You can save jobs as either independent or related files. The JBI and JBR file extensions identify each of these file types respectively. A job saved as an independent file contains only the instructions for the named job. In a job saved as a related file, the program instructions are searched for internal job calls. Any jobs, frames, and tools called in the program instructions are appended to the end of the file. The resulting file then contains the program instructions for all related jobs.

For example, saving the MAIN job we used above as an independent file would store only the MAIN instructions. However, when we save the same job as a related file, the instructions for SAVEV, DC1, DC2, and DC3 are appended to the end of the MAIN instructions and stored in a single file.

Using the Variable Editor

The Visual DCI 4 Variable Editor lets the user save and modify program variables while the robot is in cycle. Variables can be modified and then downloaded back to the controller. To begin using the Variable Editor, proceed as follows:

1. From the Communication portion of the main window, click the Start button. A communications link between the controller and the PC is established.

   Note: Visual DCI communication must be ON before putting the robot in cycle. Visual DCI must remain ON until after the robot is taken out of cycle.

2. At the robot controller, select the master job and place the robot in cycle as follows:
   a) Press TEACH mode.
   b) Select the master job.
   c) Place the controller in AUTO mode.
   d) Press CYCLE START.

3. Click the Variable Editor button on the main window. The Variable Editor window appears.
4. Select the variable type tab you wish to modify.
5. Click in the Name field of the variable number you wish to change and type the new name.
6. Click in the Present Value field and type in the new value.

*Note:* You may need to use the horizontal scroll bar to view all the Axes, and Var Type values for the P and EX variables.

7. The controller uploads and downloads the new variables as directed by the Master job.

**Closing a Visual DCI Session**

Visual DCI must remain ON until after the robot is taken out of cycle. To end a Visual DCI 4 session, proceed as follows:

1. From the robot controller, press HOLD or change mode to TEACH or REMOTE. This takes the robot out of cycle.
2. Click the Stop button on the Communication portion of the main window.

*Note:* If you assigned variable labels during the current session and have not already saved them, you must save these labels in a configuration file before you close the session. Otherwise, the current label assignments will only apply to the default configuration file.

Visual DCI saves any configuration changes you have made during the current session to the configuration file. These can include the communication parameters, default variable file, user name assignments, and file path selections.
NOTES
Chapter 6: Techniques

In this section we offer some techniques and hints to optimize the operation of Visual DCI 4 for your particular applications.

Optimizing for Cycle Time

The techniques listed below can be incorporated into your processes, either singly or in combination, to reduce total cycle time.

- Set up a byte variable that when true indicates that the values of other variables have not changed. Write an instruction sequence in the job program that tests the value of the change variable. In this sequence, jump to a variable download sequence when the change variable tests false. By waiting until changes occur to download variables, you reduce the average number of downloads per cycle to a minimum.

Table 6-1  Example Conditional Load Sequence

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SET B00 6</td>
<td>Set B(00)=6</td>
</tr>
<tr>
<td>SAVEV B00</td>
<td>Upload index value</td>
</tr>
<tr>
<td>LOADV B06</td>
<td>B(06) is the change variable</td>
</tr>
<tr>
<td>JUMP IF B06</td>
<td>insert the variable download instructions here, these will only</td>
</tr>
<tr>
<td>&lt;&gt; 0</td>
<td>execute when the change variable B(06) equals 0</td>
</tr>
<tr>
<td>*skipseq</td>
<td></td>
</tr>
<tr>
<td>RET</td>
<td></td>
</tr>
<tr>
<td>END</td>
<td></td>
</tr>
</tbody>
</table>

Optimizing for Variable Latency

For some operations the robot needs to update variable values frequently. The techniques listed below help satisfy the requirement to reduce variable latency.

- You can include instructions in your job programs to get variables as often as required. If necessary, you could update each variable as it is used in the program. Doing so ensures that the robot always uses the most current value for any given variable.
- You can write several different instruction sequences to update specific variables or groups of variables. By attaching each instruction sequence to a different conditional statement, you can update variables only under specific circumstances.
- You can call a different sequence to update selected variables as the robot starts each new task. This method updates groups of variables as needed in a specific instruction sequence, rather than downloading all at once.

Optimizing Memory Usage

In some cases, minimizing controller memory overhead may be a concern. The following techniques help to conserve memory.

- Write program instructions tying controller load sequences to conditional statements.
- Save jobs as individual rather than related files. While the use of related files can simplify some operations, the resulting files can be very large. This approach is most effective when the master job contains references to several jobs, but is not likely to use all of them during a specific session.
NOTES
Chapter 7: OLE Commands

Visual DCI 4 supports OLE (Object Linking and Embedding) Automation. OLE Automation allows other PC-based application packages to control Visual DCI. Users can, for example, set serial communications port settings, turn on DCI, or reset the communications buffer from other software applications.

OLE, or ActiveX as it has become known, is a vast standard that has had literally hundreds of books dedicated to explaining it. Fortunately, creating and using an OLE client in Visual Basic is quite simple.

Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Read/Write Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommStatus</td>
<td>Read</td>
</tr>
<tr>
<td>Element1</td>
<td>Read/Write Used to transfer data when using GetVarData and SaveVarData</td>
</tr>
<tr>
<td>Element2</td>
<td>Read/Write Used to transfer data when using GetVarData and SaveVarData</td>
</tr>
<tr>
<td>Element3</td>
<td>Read/Write Used to transfer data when using GetVarData and SaveVarData</td>
</tr>
<tr>
<td>Element4</td>
<td>Read/Write Used to transfer data when using GetVarData and SaveVarData</td>
</tr>
<tr>
<td>Element5</td>
<td>Read/Write Used to transfer data when using GetVarData and SaveVarData</td>
</tr>
<tr>
<td>Element6</td>
<td>Read/Write Used to transfer data when using GetVarData and SaveVarData</td>
</tr>
<tr>
<td>FilePath</td>
<td>Read/Write Reads or writes the PC directory to which the robot jobs will be transferred to or from the robot controller.</td>
</tr>
<tr>
<td>Visible</td>
<td>Read/Write Allows the user to display the session of DCI being used. &quot;By default, the DCI session is invisible.&quot;</td>
</tr>
</tbody>
</table>
7-2 OLE Commands

Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCIStart</td>
<td>Enables communication between the robot and Visual DCI</td>
</tr>
<tr>
<td>DCIStop</td>
<td>Disables communication between the robot and Visual DCI</td>
</tr>
<tr>
<td>GetVarData</td>
<td>Retrieves the present contents of variable and item passed in.</td>
</tr>
<tr>
<td></td>
<td>The contents of the variable are stored in the Element Properties.</td>
</tr>
<tr>
<td>SaveVarData</td>
<td>Saves the present contents of variable and item passed in.</td>
</tr>
<tr>
<td></td>
<td>The contents of the variable are stored in the Element Properties.</td>
</tr>
<tr>
<td>SetConfigFile</td>
<td>Change the configuration this session of DCI is running.</td>
</tr>
</tbody>
</table>

Events

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommError</td>
<td>Occurs when a communication error has occurred</td>
</tr>
<tr>
<td>DCILoadJ</td>
<td>Occurs when the robot has sucessfully loaded a job</td>
</tr>
<tr>
<td>DCILoadV</td>
<td>Occurs when the robot has sucessfully loaded a variable</td>
</tr>
<tr>
<td>DCISaveJ</td>
<td>Occurs when the robot has sucessfully saved a job</td>
</tr>
<tr>
<td>DCILoadV</td>
<td>Occurs when the robot has sucessfully saved a variable</td>
</tr>
</tbody>
</table>

Examples

Starting a Copy of Visual DCI

1. Add a reference of Visual DCI to your project by selecting Project > References > Motoman Visual DCI.
2. Dim RobotServer as new DCIserver.
Enable/Disable Communication

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RobotServer.StartDCI()</td>
<td>To Open the Comm Port and enable communication with the robot</td>
</tr>
<tr>
<td>RobotServer.StopDCI()</td>
<td>To Close the Comm Port and disable communication with the robot</td>
</tr>
</tbody>
</table>

Change the Configuration File

```
"RobotServer.SetConfigFile(""C:\Test.dci"")" Sets the Config file to C:\Test.dci.
```

Retrieving Variable Data from Visual DCI

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dim RetVal as Variant</td>
<td></td>
</tr>
<tr>
<td>&quot;RobotServer.GetVarData(BVar, 3)&quot;</td>
<td>Retreive the present value of B3</td>
</tr>
<tr>
<td>RetVal = RobotServer.Element1</td>
<td>RetVal is now equal to the value of B3</td>
</tr>
</tbody>
</table>

Saving Variable Data to Visual DCI

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RobotServer.Element1 = 50</td>
<td>Set Element1 of the VisualDCI interface to 50</td>
</tr>
<tr>
<td>&quot;RobotServer.SaveVarData(Bvar,3)&quot;</td>
<td>Save the present value of Element1 to B3</td>
</tr>
</tbody>
</table>

Hiding or Showing the Session of Visual DCI

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RobotServer.Visible = true</td>
<td>Shows the session of VisualDCI that your application is controlling</td>
</tr>
<tr>
<td>RobotServer.Visible = false</td>
<td>Hides the session of VisualDCI that your application is controlling</td>
</tr>
</tbody>
</table>

Determining if DCI is Enabled to Communicate

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dim RetVal as Boolean</td>
<td></td>
</tr>
</tbody>
</table>
### OLE Commands

| RetVal = RobotServer.CommStatus | "Returns true if the comm port is open, otherwise returns false"

#### Determining the Present File Path

<table>
<thead>
<tr>
<th>Dim Path as String</th>
<th>Path = RobotServer.FilePath</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Returns the path that all robot jobs being uploaded to or downloaded from</td>
</tr>
</tbody>
</table>

#### Changing the Present File Path

<table>
<thead>
<tr>
<th>Dim Path as String</th>
<th>&quot;Path = &quot;&quot;C:\Temp&quot;&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>RobotServer.FilePath = Path</td>
<td>Changes the present path to C:\Temp</td>
</tr>
</tbody>
</table>
DCI Stops Functioning Normally

Did you turn off the computer and controller?
If not, turn off both and restart.

Is at least one portion of the program functioning? (Loading or saving certain jobs?)
If Yes, what kind of jobs are working or causing problems?

Are jobs from the same controller/robot?
If jobs are from a different system, the jobs may be incompatible.

Are any of the jobs dependent upon a job not there or currently running?
This may affect the saving of a job from controller to the PC.

Do any of the jobs have position variables?
If a job using any type of position variable is loaded into the controller, the position variable must be defined in the controller. This is true for FDE and VDE as well.

Were any of the jobs created with MotoSim?
If jobs were created for a different robot, CMOS file or all.prm file may affect compatibility.

Are any of these jobs extremely long or for dual robots?
They may be out of memory. Visual DCI may also need more time to gather the information. Change parameter RS034 to a value between 30 -100.

Create small sample jobs using different functions. Run the jobs separately to isolate any problem caused by a specific function. For example, create jobs that employ:

- Motion only
- Position variables (if you are using them)
- Welding commands only
- Incorrect syntax, etc.

Did you make any changes to the controller or computer?
Try reverting to the original settings to see if that affected Visual DCI (other users may have changed settings).

Check cables. Make certain all cables are connected. Check for intermittent connections by disconnecting and reconnecting cables. Replace cables if necessary. Long cables can degrade signal quality. Also check for added equipment such as welding power sources, positioners, etc. Electrical noise from nearby devices can cause malfunction.
Check to see if a cable is damaged by swapping the cable in question with a good cable. If you do not have an extra cable, visually check cable for damage.

Close any other applications you have running. Certain software can interfere with Visual DCI.

Did you encounter an Alarm 5920-1 (Insufficient memory) message?
You may have too many jobs open or several jobs may be too long. Eliminate some jobs, break jobs down into smaller jobs, or download jobs to reduce memory requirements.

Did you encounter an Alarm 5930-22 (Time out)?
The PC is not responding. This typically occur on systems running Windows 3.11.

- Cables are not connected or they do not have a good connection.
- Operator error
- Visual DCI turned off
Try setting RS034 to a value between 30 and 100. The default setting is “30” (30 seconds). Increase the setting to give the controller more time to respond to the PC.

**Visual DCI 4 Does Not Work After Installation**

*Are system parameters set properly?*
Refer to Chapter 3 Installation and verify parameter settings for your configuration.

*RS034 Parameter*
Response Waiting Timer. May be set between 0 and 100 (normally set to 30). If you are running under Win 3.1, you may want to set this higher.

*Are the cables connected?*
Verify data cable connections between robot controller and PC.

*Is the Hardware Key correctly installed on the computer’s parallel port?*
The hardware key attaches to the computer’s parallel port. This port is commonly used to connect printers and other peripheral devices to your computer.

**Applications**

*Download jobs*
Optional features must be activated on the controller before Visual DCI will load. This is also true for FDE (Floppy Disk Emulator) and VDE (Visual Data Exchange). Jobs with syntax errors will not be loaded.

*Download variables*
Position variables must be defined before jobs using those position variables can be loaded. This is also true for jobs loaded with FDE or VDE. This has been used to shift to corners of boxes of various sizes.

*Memory*
When running out of memory, DCI can save the jobs to the computer, delete those jobs from the controller, and load when needed. Another option is to purchase expanded memory.

*Modified jobs*
The SAVEJ and SAVEV functions will save the current version of the job or variable to the computer. This may be useful if a vision system is used to shift a robot’s position. It may be used to backup all jobs on the computer on a weekly or routine schedule. To backup the entire system setting, it is highly recommended to use FDE.

*Saving variable*
May be useful if a database on the computer is used to track schedules.

*Activation*
Other computer program such as Visual Basic or C can open and begin Visual DCI. Some have used this with the computer’s clock to save information on a regular schedule.

*Computer Programming*
Motoman does not support interface screens created by customers unless Motoman created them as a purchased additional option. If you want a custom interface, make sure a computer programmer creates it. An example interface for Excel and Visual Basic are provided. These may be modified.

*Number of running copies*
This is limited to the number of available COM ports on the computer.
Chapter 1: Index

A
About This Document 1-1
ANSI/RIA 2-1
Assigning Variable Labels 4-2

B
Basic Operation 4-1
Building a Data Cable 3-7

C
Change the Configuration File 7-3
Changing the Present File Path 7-4
Closing a Visual DCI Session 5-5
Communication 4-1
Communication Status Windows 4-1
Communications Parameters 4-2
Communications Setup 4-3
Configuration Files 4-2
Controller Setup 3-2
Customer Service 1-2
Cycle Time 6-1

D
Determining the Present File Path 7-4
Determining if DCI is Enabled to Communicate 7-3

E
Enable/Disable Communication 7-3
ERC Cable Connections 3-6
ERC Controller 3-6
ERC Parameters 3-6
Events 7-2
Examples 7-2
External Connection 3-4
External RS-232 connection 3-5

G
General Safeguarding Tips 2-2

H
Hiding or Showing the Session of Visual DCI 7-3

I
Index Variable 5-1
Installation 3-1
Installation Safety 2-2
Installing the Hardware Key 3-2
Installing the Software 3-1
Internal Connection 3-4
Internal MCP01 connection 3-5
Introduction 1-1

J
Job Files 5-4

L
Loading and Saving Job Files 5-4
Loading Variables from the PC 5-3

M
Main Window 4-1
Maintenance Safety 2-4
Mechanical Safety Devices 2-2
Memory Usage 6-1
Methods 7-2
MRC Cable Connections 3-4
MRC Controller 3-4
MRC Parameters 3-5

N
Null Modem Pin Connections 3-8

O
OLE Commands 7-1, 8-1
Opening a New Configuration File 4-2
Opening an Existing Configuration File 4-2
Operation Safety 2-4
Optimizing for Cycle Time 6-1
Optimizing for Variable Latency 6-1
Optimizing Memory Usage 6-1
Overview 1-1
### Index

<table>
<thead>
<tr>
<th>Category</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>P</strong></td>
<td></td>
</tr>
<tr>
<td>Programming Safety</td>
<td>2-3</td>
</tr>
<tr>
<td>Properties</td>
<td>7-1</td>
</tr>
<tr>
<td><strong>R</strong></td>
<td></td>
</tr>
<tr>
<td>Reference to Other</td>
<td>1-2</td>
</tr>
<tr>
<td>Documentation</td>
<td></td>
</tr>
<tr>
<td>Removing Visual DCI 4</td>
<td>3-8</td>
</tr>
<tr>
<td>Retrieving Variable Data from Visual DCI</td>
<td>7-3</td>
</tr>
<tr>
<td>Robot Program Path</td>
<td>4-1</td>
</tr>
<tr>
<td>Robot Program Structure</td>
<td>5-2</td>
</tr>
<tr>
<td>Robotic Industries Association</td>
<td>2-1</td>
</tr>
<tr>
<td><strong>S</strong></td>
<td></td>
</tr>
<tr>
<td>Saving Configuration Files</td>
<td>4-2</td>
</tr>
<tr>
<td>Saving Variables to the PC</td>
<td>5-2</td>
</tr>
<tr>
<td>Saving Variable Data to Visual DCI</td>
<td>7-3</td>
</tr>
<tr>
<td>Setting Communications</td>
<td>4-2</td>
</tr>
<tr>
<td>Parameters</td>
<td></td>
</tr>
<tr>
<td>Standard Conventions</td>
<td>2-1</td>
</tr>
<tr>
<td>Starting a Copy of Visual DCI</td>
<td>7-2</td>
</tr>
<tr>
<td>System Requirements</td>
<td>1-2</td>
</tr>
<tr>
<td><strong>T</strong></td>
<td></td>
</tr>
<tr>
<td>Techniques</td>
<td>6-1</td>
</tr>
<tr>
<td>Troubleshooting</td>
<td>8-1</td>
</tr>
<tr>
<td><strong>U</strong></td>
<td></td>
</tr>
<tr>
<td>Using the Variable Editor</td>
<td>5-4</td>
</tr>
<tr>
<td><strong>V</strong></td>
<td></td>
</tr>
<tr>
<td>Variable Editor</td>
<td>4-1, 5-1, 5-4</td>
</tr>
<tr>
<td>Variable Labels</td>
<td>4-2</td>
</tr>
<tr>
<td>Variable Latency</td>
<td>6-1</td>
</tr>
<tr>
<td>Visual DCI 4 Components</td>
<td>1-1</td>
</tr>
<tr>
<td><strong>X</strong></td>
<td></td>
</tr>
<tr>
<td>XRC Cable Connections</td>
<td>3-2</td>
</tr>
<tr>
<td>XRC Controller</td>
<td>3-2</td>
</tr>
<tr>
<td>XRC Parameters</td>
<td>3-3</td>
</tr>
</tbody>
</table>