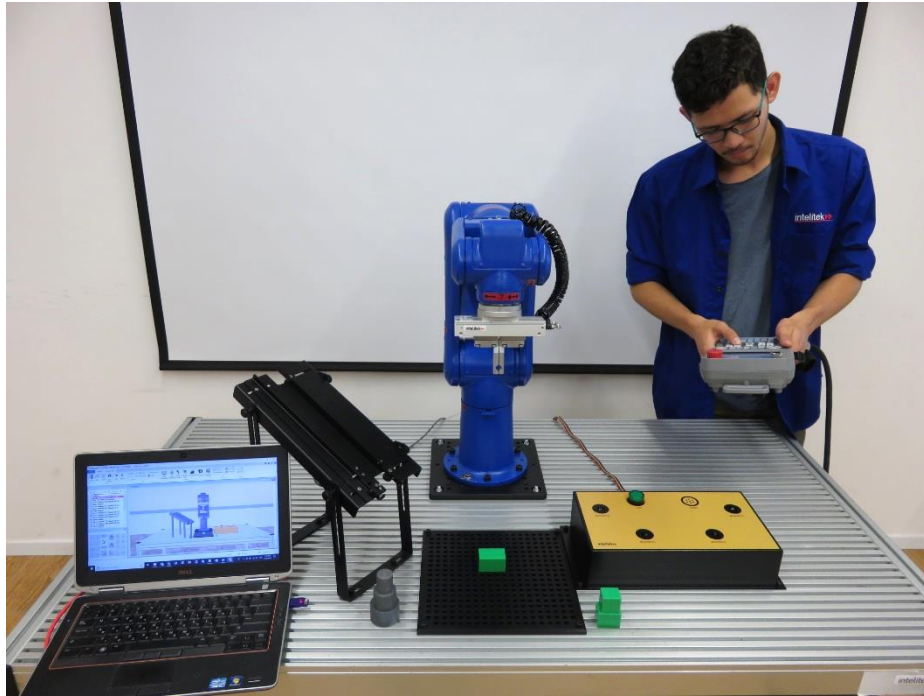


RoboCell

FOR MOTOMAN MHJF



USER MANUAL

Catalog #34-8000-0012 Rev. B

intelitek ®



Copyright © 2018 Intelitek Inc.

Tel: (603) 625-8600

RoboCell for Motoman MHJF User Manual

Fax: (603) 437-2137

Cat. # 34-8000-0012 Rev. B

July 2018

website: <http://www.intelitek.com>

email: info@intelitek.com

Intelitek software and documentation are available at <http://intelitekdownloads.com>.

All rights reserved. No part of this publication may be stored in a retrieval system, or reproduced in any way, including but not limited to photocopy, photography, magnetic, or other recording, without the prior agreement and written permission of the publisher. Program listings may be entered, stored, and executed in a computer system, but not reproduced for publication.

Every effort has been made to make this book as complete and accurate as possible. However, no warranty of suitability, purpose, or fitness is made or implied. Intelitek is not liable or responsible to any person or entity for loss or damage in connection with or stemming from the use of the software, hardware and/or the information contained in this publication.

Intelitek bears no responsibility for errors that may appear in this publication and retains the right to make changes to the software, hardware and manual without prior notice.

Table of Contents

1.	Starting RoboCell	3
1.1.	System Requirements.....	3
1.2.	Software Licensing.....	3
1.3.	Installing the Software.....	3
1.4.	RoboCell Utilities	6
1.4.1.	3D Simulation Software Demos.....	6
1.4.2.	Uninstalling RoboCell	6
1.5.	Quitting the Software.....	6
2.	Connecting RoboCell to the Physical Robot.....	7
2.1.	Overview	7
2.2.	Required Materials	7
2.3.	Connection Steps.....	8
2.3.1.	Loading Essential Parameters to the Robot Controller.....	8
2.3.2.	Connecting RoboCell and the Controller	11
2.3.3.	Confirming the Position Parameter	19
2.3.4.	Checking the Home Position	20
2.4.	Initial Connection and Testing.....	22
3.	CellSetup	25
3.1.	Opening CellSetup	25
3.2.	CellSetup Window	26
3.3.	The File Tab	27
3.4.	Compose Tab.....	28
3.5.	The CellSetup Help Menu	30
3.6.	Objects	30
3.6.1.	Object Placement	30
3.6.2.	Object Configuration	31
3.6.3.	Object Properties Menu	32
3.6.4.	Defining Inputs and Outputs	33
3.7.	New Objects Menu	34
3.7.1.	Robots	34
3.7.2.	I/O Devices	37
3.7.3.	Storage Devices	40
3.7.4.	Machines	44
3.7.5.	Welding Tools.....	45
3.7.6.	Welding Materials	46

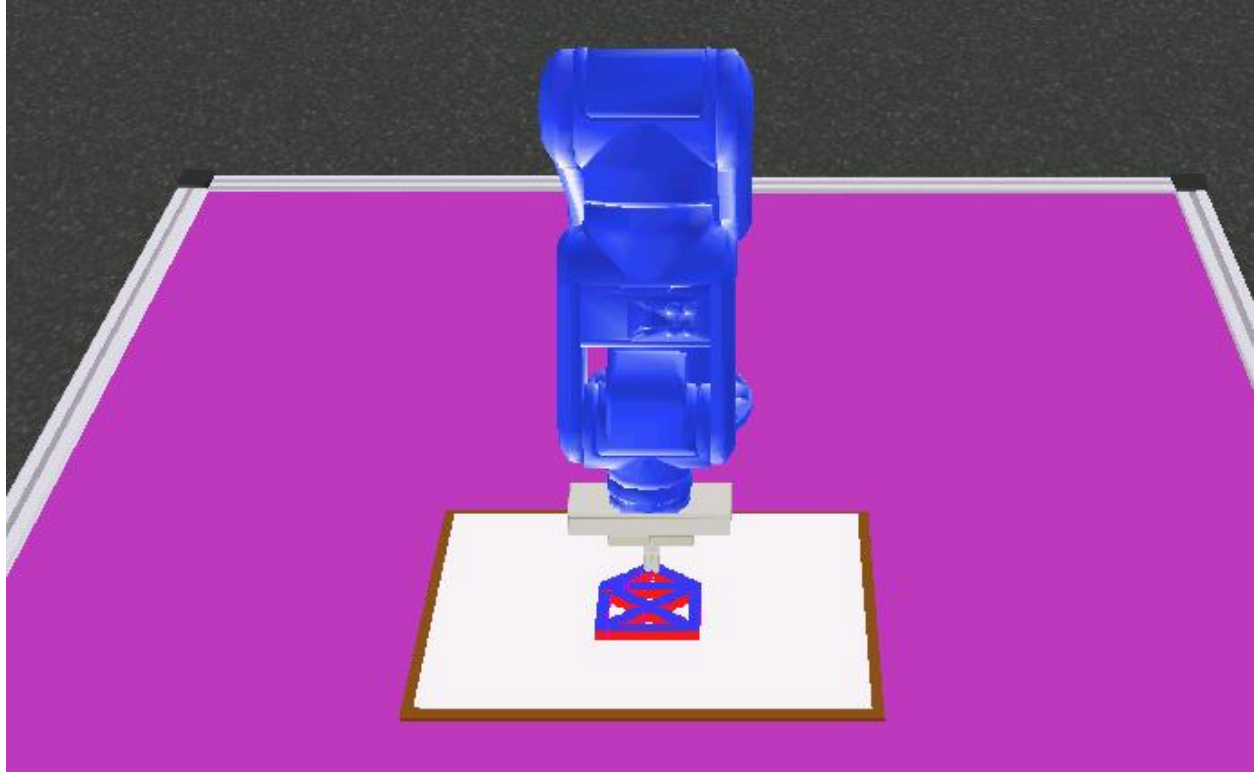
3.7.7.	General.....	47
3.7.8.	Materials	49
3.7.9.	User Objects	50
3.7.10.	Designing User Parts and Objects	51
4.	The Elements of RoboCell	53
4.1.	Overview	53
4.2.	The Command Ribbon	53
4.2.1.	The File Tab	54
4.2.2.	The Home Tab	56
4.2.3.	The Virtual Cell Tab	58
4.3.	The RoboCell Help Menu	60
4.4.	The Virtual Cell Window	60
4.4.1.	Graphic Interface.....	61
4.5.	Other RoboCell Windows	63
5.	Cell Operation	65
5.1.	The Jog Control Window	65
5.1.1.	Overview	65
5.1.2.	Coordinate Modes.....	65
5.1.3.	Manual Axis Movement	66
5.1.4.	Manual Gripper Operation	68
5.2.	Send Robot Commands	68
5.2.1.	Send Robot Functions.....	68
5.2.2.	Send Options	69
5.3.	Gripper	69
5.3.1.	Grasping Objects	69
5.3.2.	Releasing Objects	70
5.4.	Impact Detection	70
6.	Positions.....	71
6.1.	Overview	71
6.2.	Teach Panel	71
6.3.	Defining Positions (Joint Coordinates).....	73
6.3.1.	Recording Positions	73
6.3.2.	Teaching Positions.....	74
6.4.	Position List	76
6.4.1.	Overview	76
6.4.2.	Position Popup Window	76
6.4.3.	The Watch Area.....	77
6.5.	Data Dashboard.....	78
7.	Controlling the Physical Robot.....	80

7.1.	Using the Programming Pendant	80
7.2.	On-Line Mode / Off-Line Mode	80
7.3.	Servo On / Servo Off.....	81
7.4.	Downloading Positions	82
8.	Program Editing	83
8.1.	Opening and Closing a Program	83
8.2.	Program Editing Tools	85
8.3.	Adding and Editing Commands	86
8.4.	Axis Control Commands	87
8.5.	Program Flow Commands	90
8.6.	Input/Output Commands	98
8.7.	FS100 Controller-Specific Commands.....	103
9.	Variable Programming	106
9.1.	Using a Variable Instead of a Numeric Value:	106
9.2.	Monitoring Variable Value	108
10.	Program Execution.....	109
10.1.	Running a Program.....	109
10.2.	Halting Program Execution	110
10.3.	Input and Output Status	110
10.4.	RoboCell Log File	111
11.	Replicating Real and Simulated Cells	112
11.1.1.	From Simulation to Practice.....	112
11.1.2.	From Practice to Simulation.....	112
12.	Sample RoboCell Projects	113
12.1.	Sample Project Descriptions.....	113
13.	Troubleshooting.....	114
13.1.	General Troubleshooting.....	114
13.2.	Resolving the Common Movement Alarm (Alarm 4107).....	115

Introducing RoboCell

RoboCell is a full-featured robotics control software package, which provides a user-friendly tool for robot programming, operation, and simulation.

This manual covers all features, menus, and commands of RoboCell Version 10 with the Yaskawa Motoman MHJF manipulator and FS100 robotic controller.



1. Starting RoboCell

The following sections describe the steps involved in starting RoboCell.

1.1. SYSTEM REQUIREMENTS

For best performance, the following system is recommended:

- Intel Core i5 2400 GHz 2400 GHz or higher
- At least 4 GB RAM
- A hard drive with at least 200 MB of free disk space
- A mouse
- Windows 7/8/8.1/10

 **Note:** Your operating system might have additional hardware requirements.

1.2. SOFTWARE LICENSING

The RoboCell software is protected by a licensing agreement. Full details on the Intelitek software licensing are provided in the Intelitek Software Licensing Guide.

1.3. INSTALLING THE SOFTWARE

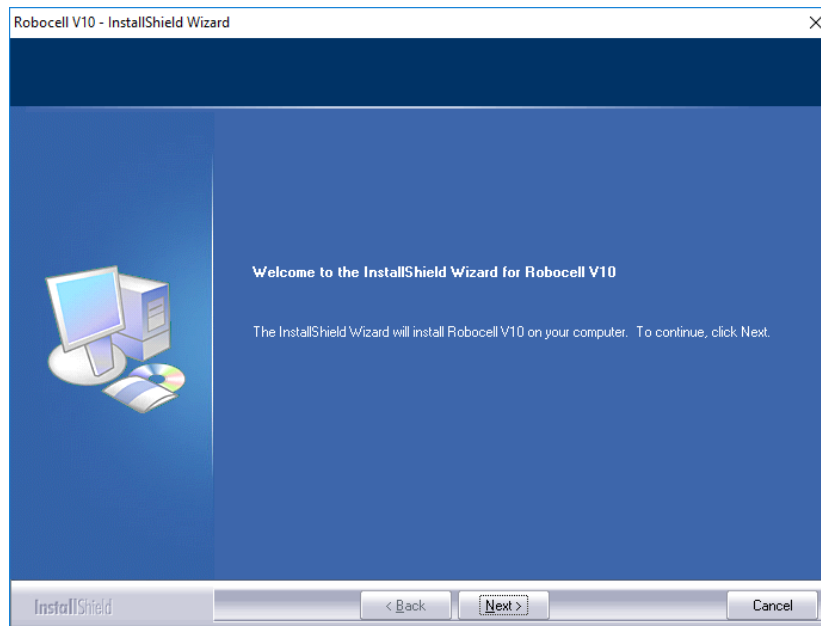
Close any applications that are open before proceeding with the installation procedures. If you are about to reinstall the software or install a newer version to an existing RoboCell directory, it is recommended that you back up any existing user-created files before you begin the installation. It is also recommended that you remove the previous RoboCell version for Windows installation using the software's Uninstall utility (For more information on uninstalling RoboCell, see 1.4.2 Uninstalling RoboCell on page 6).

To install RoboCell:

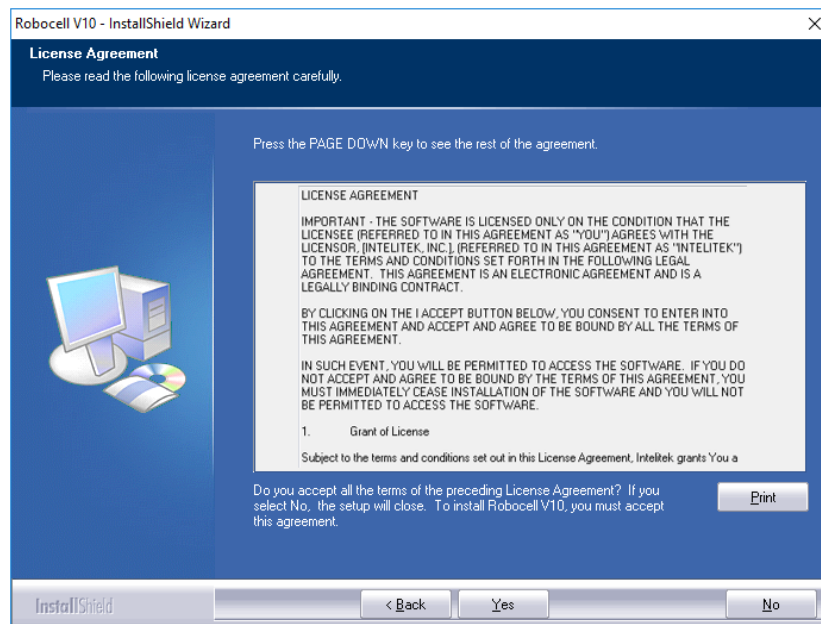
1. Double-click the RobocellV10Setup.exe file to start the installation procedure.



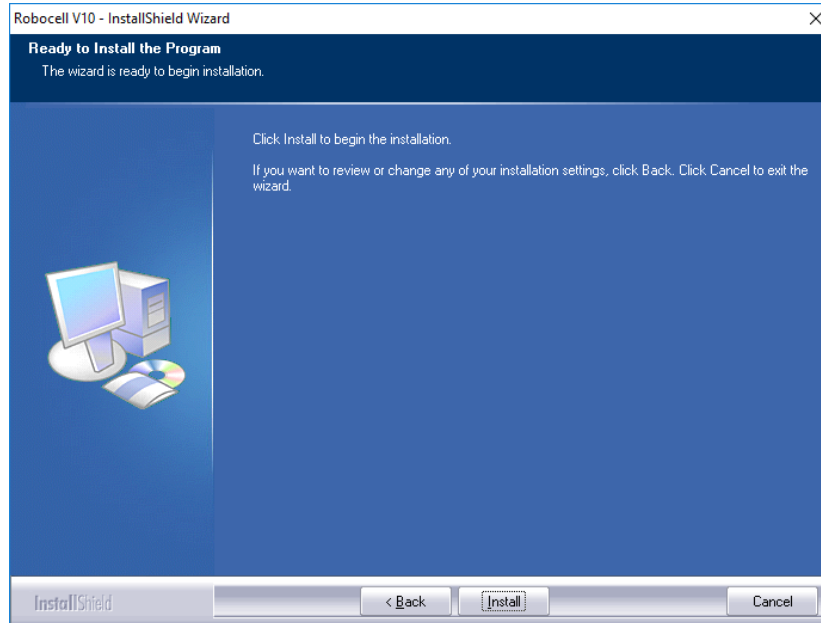
2. Wait until the Welcome window is displayed.



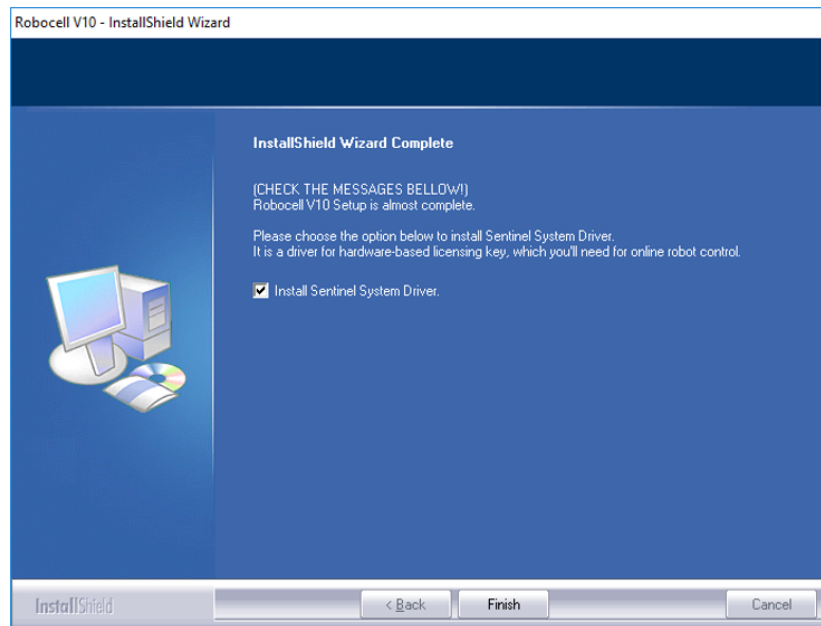
3. Click **Next**. The License Agreement window is displayed.



4. Review the Intelitek software license agreement. You must accept the terms of this agreement in order to proceed with the installation. To accept, select **Yes**. The Destination Location Window is displayed.
5. Click **Next**. The InstallShield Wizard is ready to Install the program.



6. Click **Next**. The software is installed. You are prompted to install a driver.



7. Keep the checkbox checked and click **Finish**. RoboCell and its accompanied driver are installed.

1.4. ROBOCELL UTILITIES

These utilities are available from the RoboCell group:

- CellSetup
- 3D Simulation Software Demos
- Uninstall

1.4.1. 3D Simulation Software Demos

Demonstration files included with the software allow you to observe the capabilities of RoboCell. To run these demo files, perform the following:

1. Navigate to **File | Open**.
2. Select the desired *.WS file (RoboCell project file) from those listed in the window.
3. Select **Home | Cycle** to start the demo.
4. To stop the demo, select **|Home | Stop** or **Virtual Cell | Stop**.
5. Use the menu options to navigate in the virtual cell. See section 4.2.3 The Virtual Cell Tab on page 58 for full details.
6. To exit the Demos utility, select **File | Exit**.

1.4.2. Uninstalling RoboCell

To uninstall RoboCell:

1. From the RoboCell program group, select **Uninstall**.
2. Follow the instructions that are displayed on the screen.

The Intelitek Software Licensing Guide provides full details on the procedures for protecting your license, transferring the license from one PC to another, and returning the license to Intelitek for retrieval at a later time.

1.5. QUITTING THE SOFTWARE

To close RoboCell (or its components), perform any of the following:

- Select **File | Exit**.
- Click the **X** in the RoboCell title bar.
- Press **[Alt]+F4**.

2. Connecting RoboCell to the Physical Robot

2.1. OVERVIEW

You can control your Yaskawa Motoman MHJF manipulator (robotic arm) using the RoboCell V10 control and simulation software. This section provides details on how to connect the RoboCell software with the Motoman hardware.

However, this guide does not detail robot controller operation procedures. For more information about using the controller, including the operation of the programming pendant, see the FS100 Operator's Manual.

- For more information about RoboCell, see the RoboCell for MHJF User Manual.

2.2. REQUIRED MATERIALS

The following items are required for initial connection of RoboCell with the FS100 controller and robotic arm:

- Computer with RoboCell V10 installed and free USB and Ethernet ports
- Motoman MHJF robotic arm
- Motoman FS100 robotic controller with attached programming pendant
- MOTCOM32 dongle
- Red Ethernet cable
- Flash drive with essential controller parameters
- Flat-head screwdriver

2.3. CONNECTION STEPS

Connecting RoboCell with the robot controller involves multiple procedures. These procedures are documented in the sections below.

- ① **Important Note:** *The initial connection of RoboCell and the introduction of new parameters may delete data that it is already stored in the controller memory, such as jobs and variables. Before commencing with the procedures below, create a backup of the data on the controller by transferring the current data to an external memory device. See the FS100 Operator's Manual, section 7, for details on how to accomplish this.*

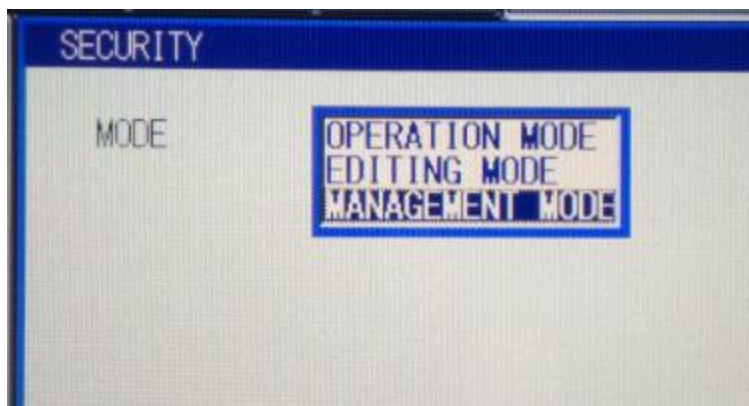
2.3.1. Loading Essential Parameters to the Robot Controller

Perform these steps:

1. Power on the controller.
2. On the initial screen, press the **Connect to FS100** button.



3. Follow the on-screen instructions to connect the programming pendant.
4. Turn the MODE key to **TEACH** mode.
5. In the main menu on the left side of the screen, navigate to **SYSTEM INFO | SECURITY**.
6. Set the security mode to **MANAGEMENT** mode.



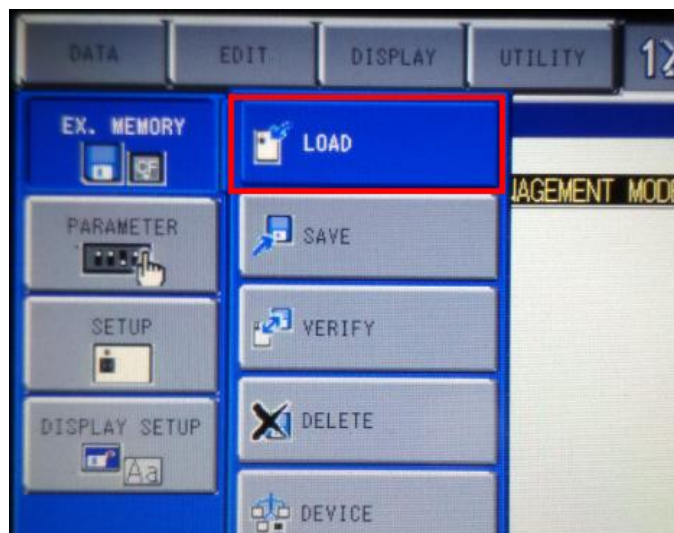
7. Enter the password **99999999** and press the programming pendant's ENTER key.



- Note:** Management mode is required for all connection procedures. The three-key icon in the menu at the top of the programming pendant's screen indicates that management mode is the current security mode.

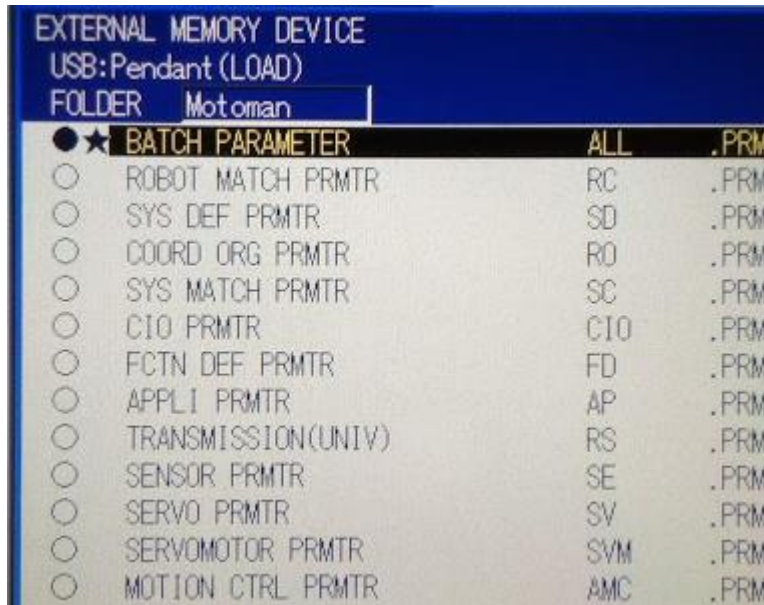


8. Insert the flash drive containing the essential parameters into the USB slot on the back of the programming pendant.
9. In the main menu, navigate to **EX. MEMORY | DEVICE**.
10. In the drop-down menu, select **USB PENDANT**.
11. In the main menu, navigate to **EX. MEMORY | LOAD**.



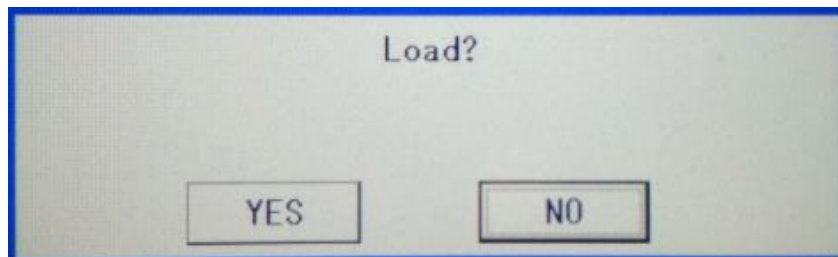
12. Select **PARAMETER**.

13. Select **BATCH PARAMETER**. A star is displayed to the left of the line.



14. Press the ENTER key.

15. In the **Load?** pop-up window, select **YES**. The required parameters for operating the robot with RoboCell are loaded to the controller.



16. Allow the parameters to load. This may take several seconds.

17. Turn the controller off.

2.3.2. Connecting RoboCell and the Controller

Follow these procedures to allow data transfer between your computer and the robot controller.

2.3.2.1. Making the Physical Connection

1. With a flat-head screwdriver or other tool, open the left window of the front panel of the controller.



2. Connect your computer to the controller using the red Ethernet cable. For purposes of clarity, the image below shows the entire front panel removed.

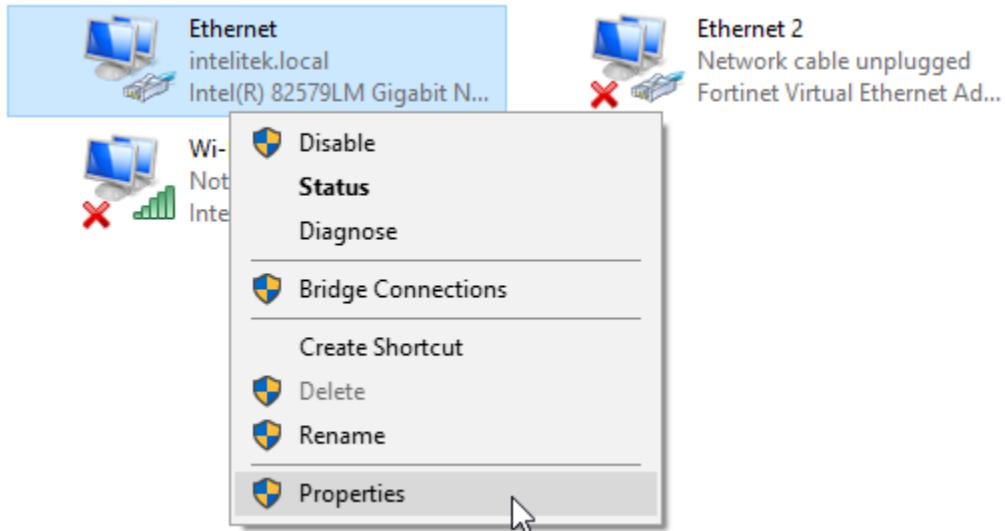


2.3.2.2. Enabling Ethernet Communication - Computer

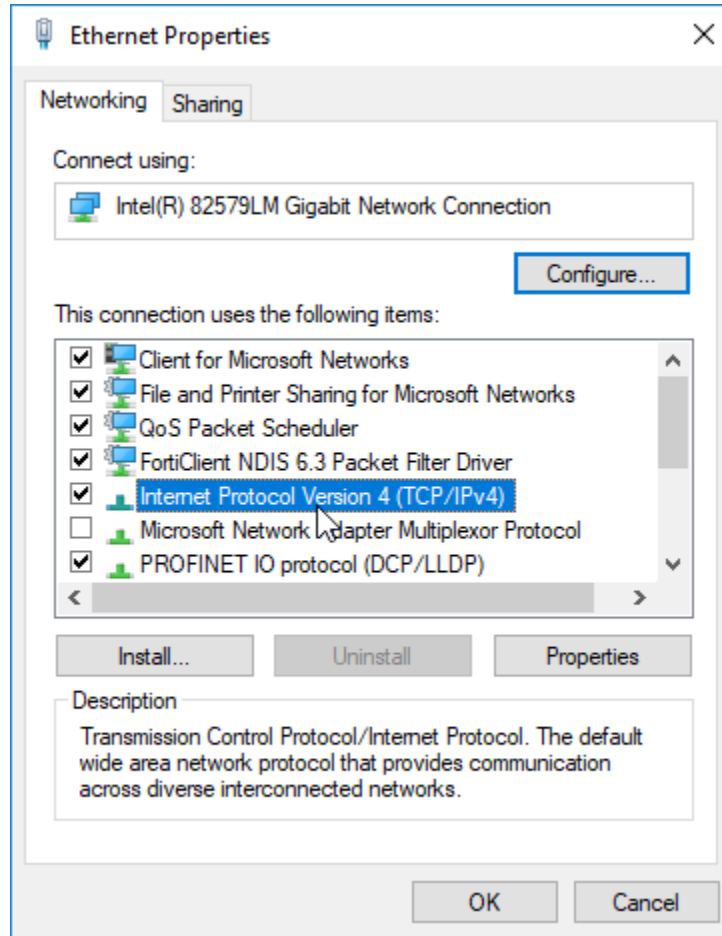
RoboCell requires a specific Ethernet IP address to communicate with the controller.

To change the required Ethernet adapter settings on the Windows 10 operating system:

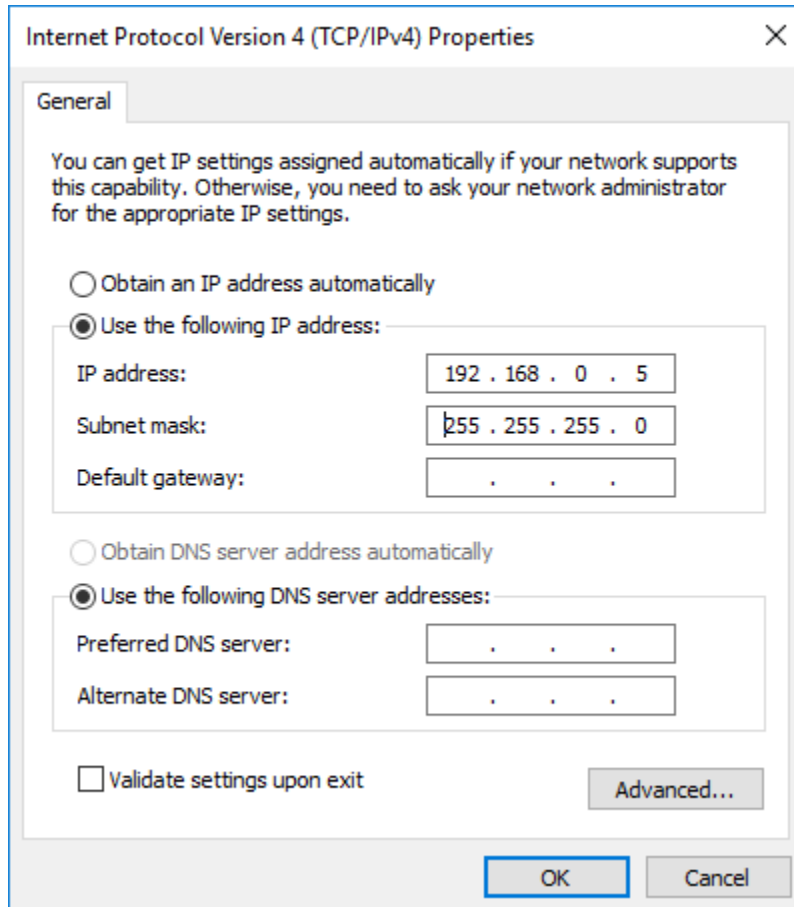
1. Turn the robot controller on.
2. On your computer, navigate to **Control Panel | Network and Ethernet | Network Connections**.
3. Double-click the active Ethernet connection and select **Properties**.



4. The Ethernet Properties window is displayed. Double-click **Internet Protocol Version 4 (TCP/IPv4)**.



- The Internet Protocol Version 4 (TCP/IPv4) Properties window is displayed. Select the **Use the following IP address** radio button and enter the following:
 - For IP address: **192.168.0.5**
 - For Subnet mask: **255.255.255.0**



- Note:** If you plan to switch back and forth between networks, use the Internet Protocol Version 4 (TCP/IPv4) Properties window's **Alternate Configuration** tab (if present) to configure the required IP address and subnet mask.
- Click **OK**.

2.3.2.3. Enabling Ethernet Communication - Controller

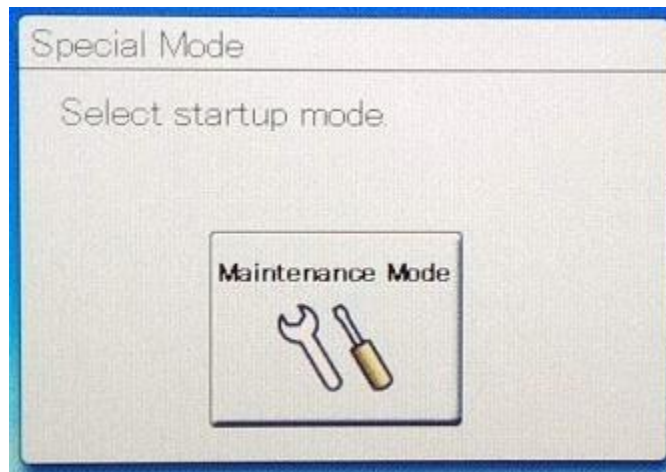
Your controller's Ethernet IP address must be changed to match your computer's Ethernet IP address.

❶ **Note:** *If the programming pendant screen is off (asleep), press any key to turn it on again.*

1. At the top of the initial programming pendant screen, select **SPECIAL MODE**.

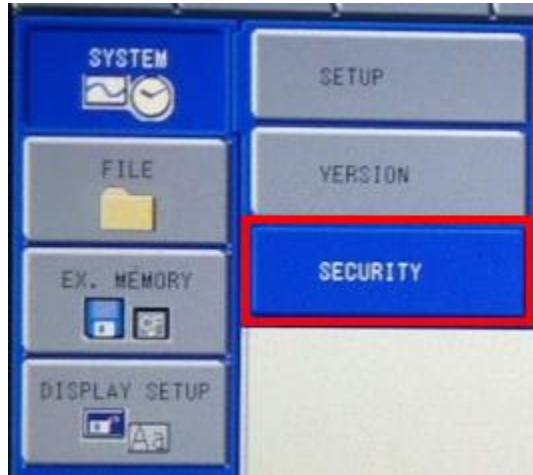


2. The Special Mode window is displayed. Press the **Maintenance Mode** button.

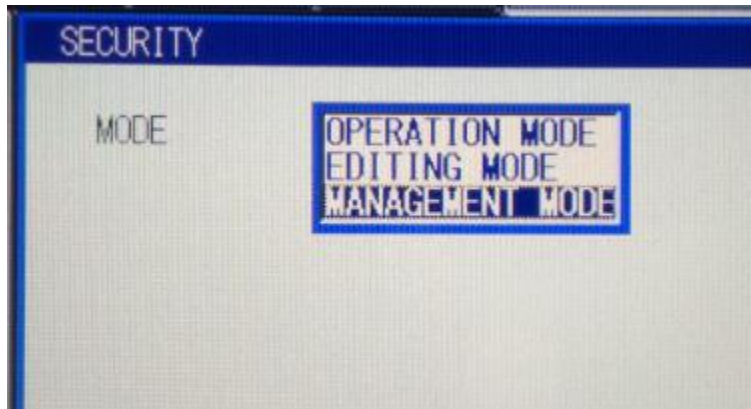


3. Follow the on-screen instructions to connect the programming pendant.

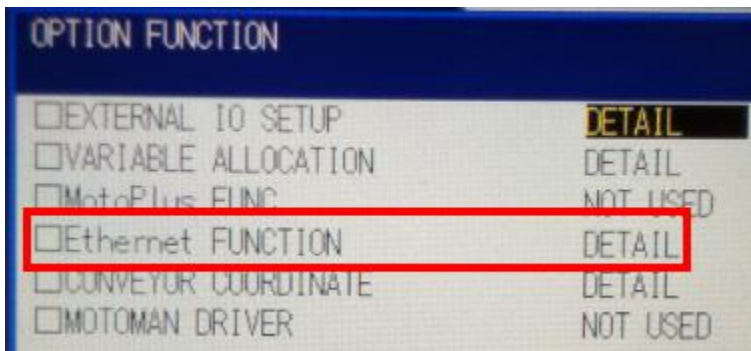
4. In the main menu on the left, navigate to **SYSTEM | SECURITY**.



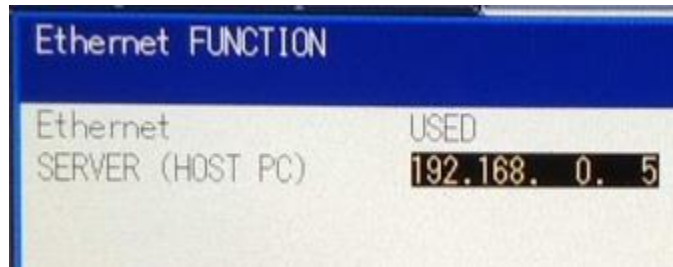
5. Set the security mode to **MANAGEMENT** mode.



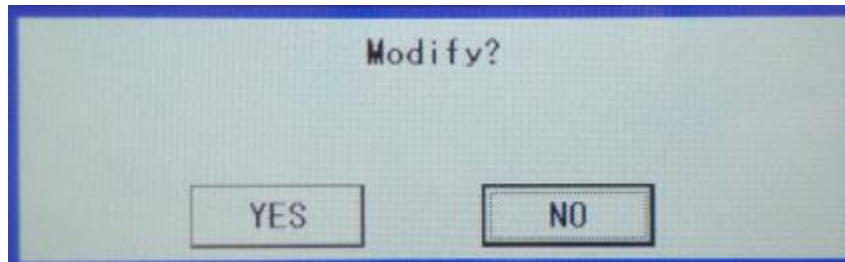
6. Enter the password **99999999** and press the ENTER key.
7. In the main menu, navigate to **SYSTEM | SETUP | OPTION FUNCTION**.
8. Select **ETHERNET FUNCTION**.



9. Change **SERVER (HOST PC)** to **192.168.0.5**.



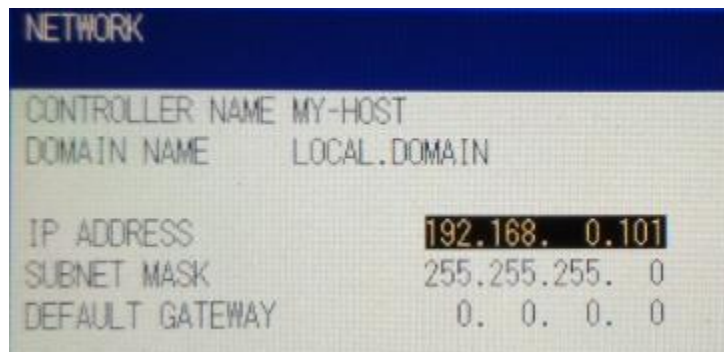
10. Press the ENTER key twice to confirm.
11. The **Modify?** Pop-up window is displayed. Select **YES**.



2.3.2.4. Modifying the Controller's IP Address

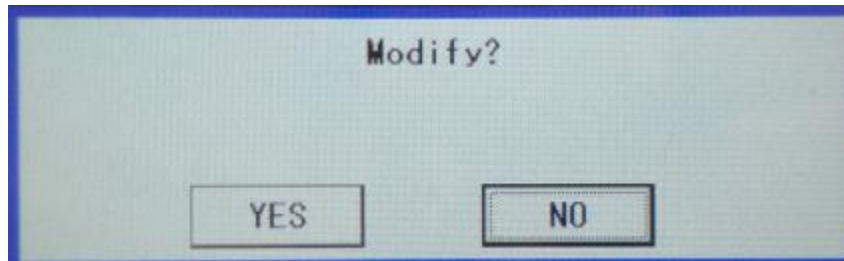
The controller's IP address must be configured in order to communicate with RoboCell.

1. In the programming pendant's main menu, navigate to **SYSTEM | SETUP | NETWORK**.
2. Change the I.P. Address to **192.168.0.101**.



3. Press the ENTER key twice.

4. In the **Modify?** pop-up window, select **YES**.



5. Turn the controller off.

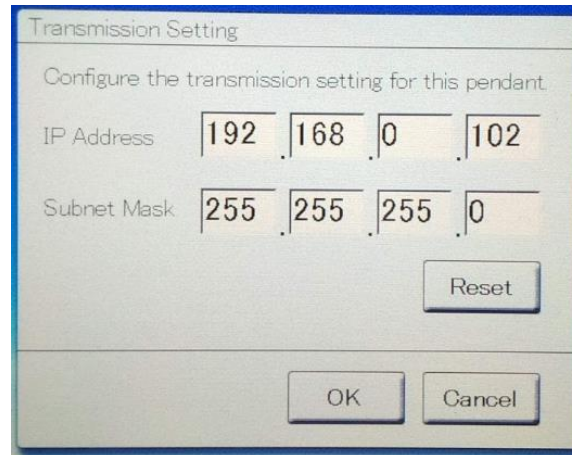
2.3.2.5. Modifying the Programming Pendant's IP Address

The *programming pendant's* IP address must be modified for it to communicate with the controller, whose IP address has been changed.

1. Turn the controller on.
2. On the initial screen, select **TRANSMISSION SETTING**.



3. Change the IP address to **192.168.0.102** and press **OK** to confirm.



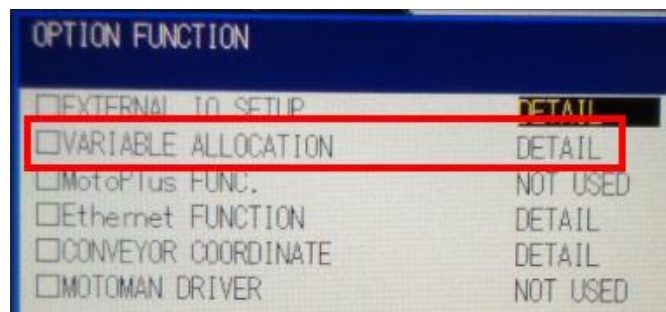
4. Turn the controller off.

2.3.3. Confirming the Position Parameter

RoboCell requires that at least 1000 positions be installed on the controller. The following procedure is performed to confirm that the correct number of positions are indeed installed.

To confirm the installation of the required position parameter:

1. Turn the controller on.
2. At the top of the initial programming pendant screen, select **SPECIAL MODE**.
3. Press the **Maintenance Mode** button.
4. Follow the on-screen instructions to connect the programming pendant.
5. In the main menu, navigate to **SYSTEM | SECURITY**.
6. Set the security mode to **MANAGEMENT** mode.
7. Enter the password **99999999** and then press the ENTER key.
8. In the main menu, navigate to **SYSTEM | SETUP | OPTION FUNCTION**.
9. Select **VARIABLE ALLOCATION**.



- 10. POSITION(ROBOT)** must have a value of 1280. If necessary, adjust the value by selecting ++ or -- until **1280** is displayed.

VARIABLE ALLOCATION			
VARIABLE NAME	ADJUST	NUMBER	RANGE
BYTE	++ / --	100	(0 - 2000)
INTEGER	++ / --	100	(0 - 2000)
DOUBLE	++ / --	100	(0 - 2000)
REAL	++ / --	100	(0 - 2000)
STRING	++ / --	100	(0 - 1931)
POSITION(ROBOT)	++ / --	1280	(0 - 10000)
POSITION(BASE)	++ / --	128	(0 - 10000)
POSITION(STATION)	++ / --	128	(0 - 10000)

- Press ENTER.
- For all pop-up windows, select **YES**.
- Turn the controller off.

2.3.4. Checking the Home Position

Absolute encoder data is different for each FS100/MHJF robotic system's home position. Therefore, this data must be checked before operating the MHJF once new parameter data has been uploaded.

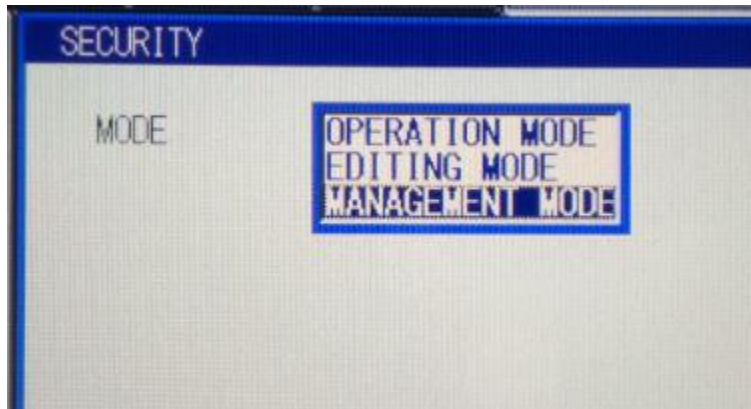
To check and, if necessary, modify the absolute data of the home position:

- Power on the controller.
- On the initial screen, press the **Connect to FS100** button.



- Follow the on-screen instructions to connect the programming pendant.
- Turn the MODE key to **TEACH** mode.
- In the main menu on the left side of the screen, navigate to **SYSTEM INFO | SECURITY**.

- Set the security mode to **MANAGEMENT** mode.



- Enter the password **99999999** and press the programming pendant's ENTER key.
- In the main menu on the left side of the screen, navigate to **ROBOT | HOME POSITION**.

The home position absolute data is displayed.

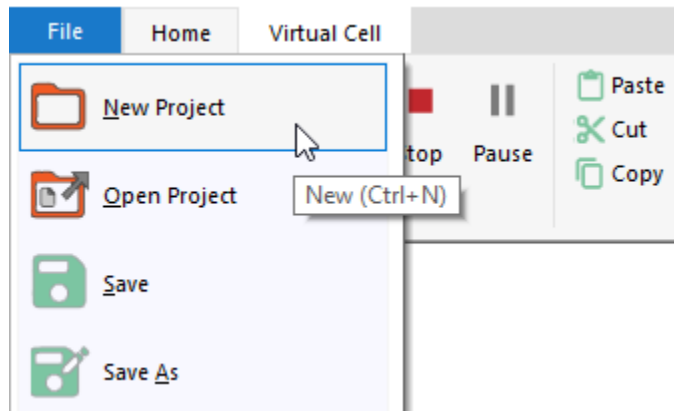
HOME POSITIONING		
	SELECT	ABSOLUTE DATA
R1 :S	<input checked="" type="radio"/>	4439
L	<input type="radio"/>	-19529
U	<input type="radio"/>	10422
R	<input type="radio"/>	2984
B	<input type="radio"/>	-8919
T	<input type="radio"/>	-21775

- Check the sticker on the top of your FS100 controller and confirm that the absolute data values match the absolute data values on the programming pendant screen.
- If the values match, continue to the next step. If the values do not match, modify them using the cursor, SELECT, and numerical keys.
- Turn the controller off.

2.4. INITIAL CONNECTION AND TESTING

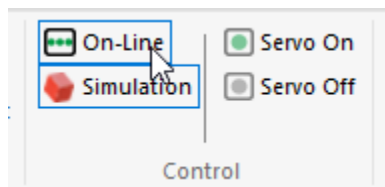
To test the system and control the manipulator using RoboCell:

1. Insert the MOTOCOM32 dongle into a USB slot on your computer.
2. Turn the programming pendant key to **REMOTE** mode.
3. Turn the controller on.
4. Launch RoboCell V10.
5. Open a new RoboCell project by navigating to **File | New Project** or by pressing Ctrl + N on your keyboard.

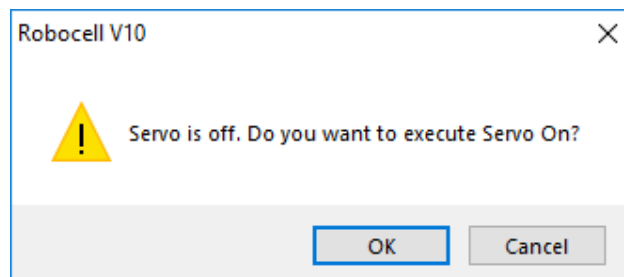


A new RoboCell project is opened.

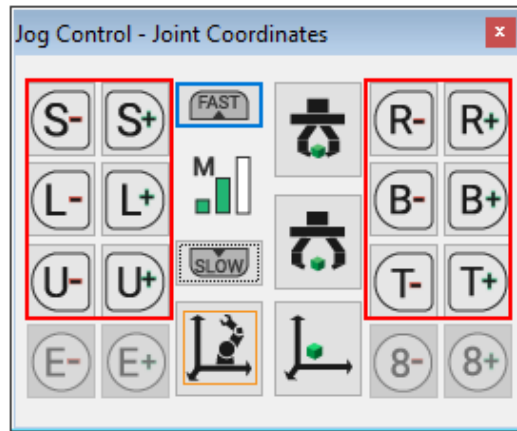
6. In the Control area of the Home tab, select **On-Line**.



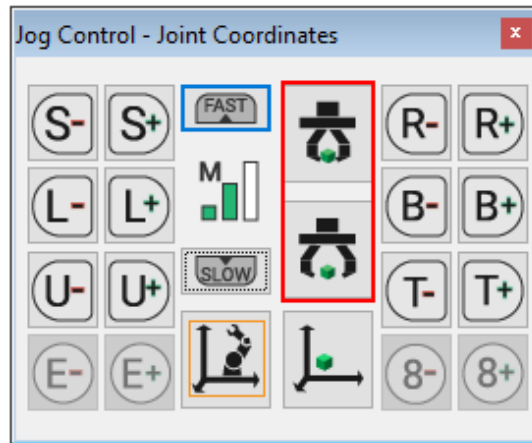
7. You are prompted to turn servo power on. Click **OK**.



- Manipulate the robotic arm by clicking the axis keys in the Jog Control window. See section 5.1, The Jog Control Window on page 65 for more information.



- If the gripper is already connected, click the open and close gripper buttons to test it.



- If any input or output devices are connected, test them using the buttons in the Digital I/O area of the Data Dashboard. See section 6.5 Data Dashboard on page 78 for more information.

Data Dashboard																
Robot Actual Position																
Cartesian Coordinates	X (mm)	Y (mm)	Z (mm)	Rx (Deg.)	Ry (Deg.)	Rz (Deg.)										
	439.00	0.00	275.00	180.00	-90.00	0.00										
Joint Angles (Deg.)	Swivel Base	Lower Arm	Upper Arm	Arm Roll	Wrist Bend	Tool Flange										
	0.00	0.00	0.00	0.00	0.00	0.00										
Encoder Pulses																
S:	0	L:	0	U:	0	R:	0	B:	0	T:	0	7:	0	8:	0	
Digital I/O																
Inputs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Outputs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

3. CellSetup

CellSetup allows you to create a new virtual cell or modify an existing one. A *virtual cell* (also known as a *workcell* or simply *cell*) created using CellSetup can be theoretical or a replica of actual robotic installations.

After creating a virtual cell, you can open it in RoboCell, where you can define positions, write programs, and run them. If a controller and a robot are connected to your computer, you can switch RoboCell to On-Line, and run the virtual robot in the virtual cell, together with the real robot in the real cell.

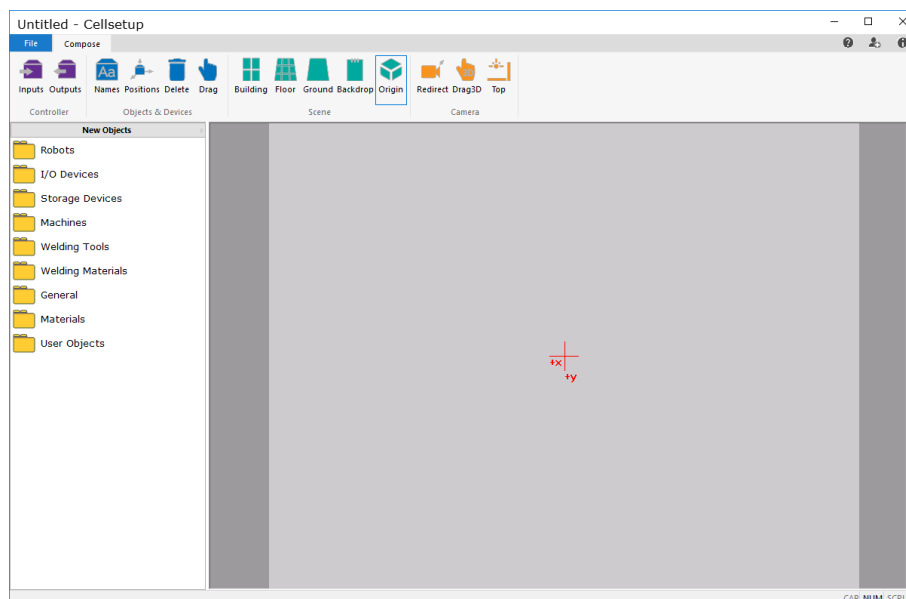
3.1. OPENING CELLSETUP

To open CellSetup:

1. Select **Start | Programs | RoboCell V10 | CellSetup V10**. After initialization, this screen is displayed:



2. Select **File | New** to create a new virtual cell. A blank cell is displayed.

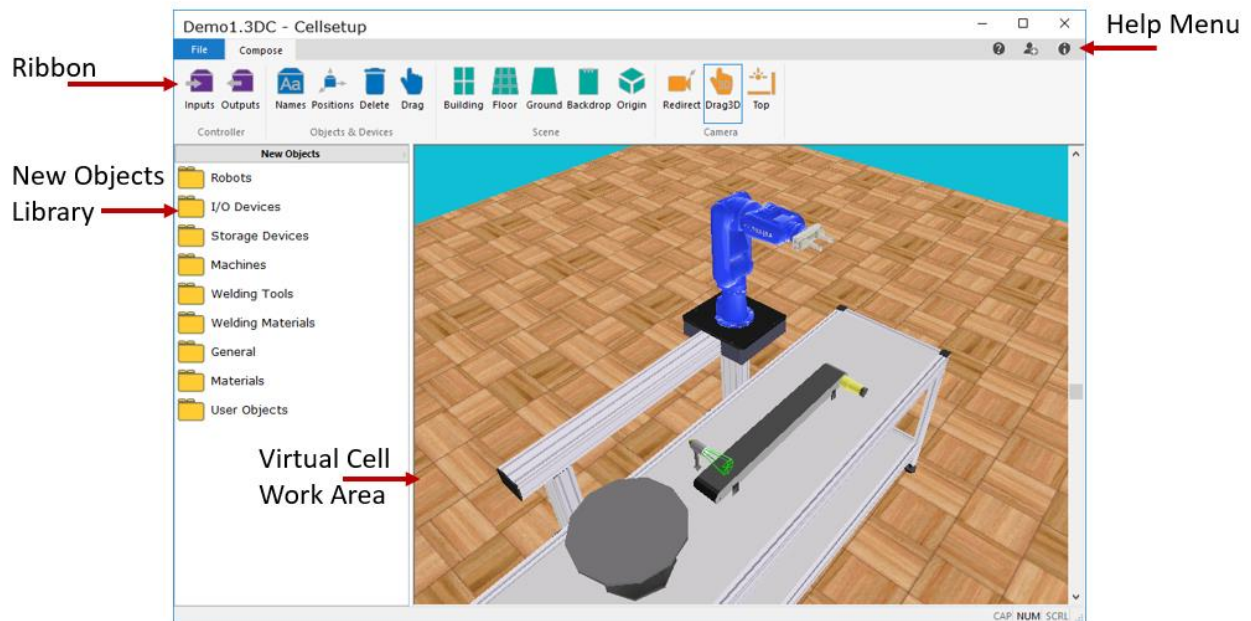


If you want to open an existing virtual cell, select **File | Open** from the Opening Screen. For further details, see section 3.3 The File Tab on page 27.

If you are working in RoboCell and you want to modify the current virtual cell you are using, select **Virtual Cell | Edit**. CellSetup opens with your current virtual cell. At this point, you cannot open another cell. Therefore, the New and Open options are not available. When you exit CellSetup after having completed your modifications, you are returned to RoboCell where the newly edited cell is displayed.

3.2. CELLSETUP WINDOW

This image shows an existing cell in the CellSetup window.

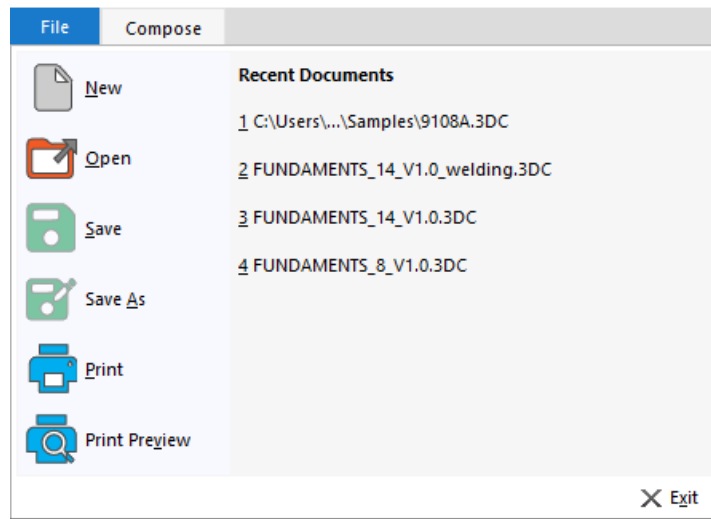


The window comprises:

- A ribbon containing all commands and options. The ribbon has two tabs: **File** and **Compose**.
- A working area containing the virtual cell and any added objects.
- A library from which new objects can be added to the virtual cell.

3.3. THE FILE TAB

The file tab is shown here:



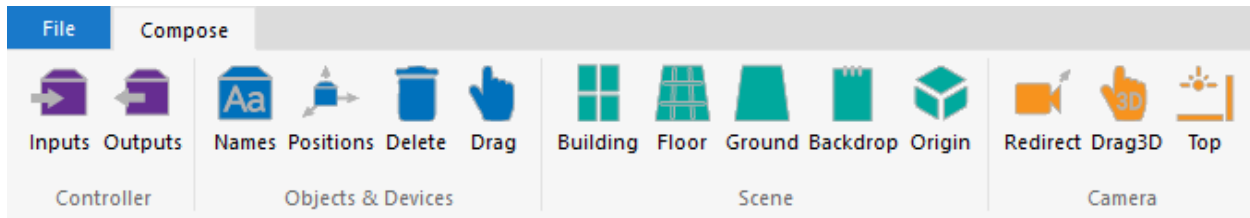
Cell data is stored in a *.3DC (CellSetup) file. The File tab contains the usual Windows functions that allow you to load and save 3DC files, and to exit the software. In addition, you can open the most-recently opened virtual cell files from this menu.

This table summarizes the options in the File tab.

CellSetup File Tab		
Icon	Option	Summary
	New	This command opens a new, untitled CellSetup file, and opens the New Object menu.
	Open	The Open dialog box is displayed. You can open an existing CellSetup file (e.g., robo1.3DC). The cell defined in the file is displayed, and the New Objects menu opens.
	Save	Saves the current placement of all objects in the cell to a CellSetup file. The default file extension is 3DC.
	Save As...	Saves the currently active CellSetup file under a new file name.
	Print	Prints the currently displayed 3D image.
	Print Preview	Opens a dialog window that shows how the printed cell appears on paper.
	Exit	Closes CellSetup.

3.4. COMPOSE TAB

The Compose tab is shown here:



The tab contains functions that allow you to add, delete, and relocate objects. Through this menu you can also set cell graphics and adjust the camera angle.




This table summarizes the options of the Compose tab.

Compose Tab		
Icon	Option	Summary
	Inputs	Opens the Inputs tab of the Controller Setup dialogue box. This tab allows you to define input connections for relevant objects, such as the experiment table. See section 3.6.4 Defining Inputs and Outputs on page 33 for more information.
	Outputs	Opens the Outputs tab of the Controller Setup dialogue box. This tab allows you to define output connections for relevant objects, such as the feeder. See section 3.6.4 Defining Inputs and Outputs on page 33 for more information.
	Names	Shows/hides the object name labels. When you select this option, a label on each object appears. Names are assigned by the software but can be renamed by the user in the object's properties menu.
	Position	Shows/hides the Object Position labels (see the image below). When you select this option, a label on each object shows its position. The coordinates that are displayed on the object's label indicate the object's position (X, Y coordinates) relative to the cell's point of origin.
	Delete Object	Activates delete mode. Once delete mode is active, click the object you want to delete. Click Yes to confirm the deletion. If you click No , the delete mode remains in effect, allowing you to select and delete another object. Press [Esc] to exit delete mode.
	Drag (Object)	Activates drag mode in order to relocate an object in the cell. After selecting this option, click the object you want to relocate and drag it to its new position. The drag mode remains in effect until you select another option or press [Esc] .

Compose Tab		
Icon	Option	Summary
	Building	Opens the Building tab of the cell's Texture Settings dialogue. Select Picture and click Edit to browse for a desired .bmp file or select Color and click Edit to define a color for the specified texture.
	Floor	Opens the Floor tab of the cell's Texture Settings dialogue. Select Picture and click Edit to browse for a desired .bmp file or select Color and click Edit to define a color for the specified texture.
	Ground	Opens the Ground tab of the cell's Texture Settings dialogue. Select Picture and click Edit to browse for a desired .bmp file or select Color and click Edit to define a color for the specified texture.
	Backdrop	Opens the Backdrop tab of the cell's Texture Settings dialogue. Select Picture and click Edit to browse for a desired .bmp file or select Color and click Edit to define a color for the specified texture.
	Origin	Shows the origin of the cell at floor level (under the center of the robot's base) as a red cross marked with X and Y axes. The positions of all objects are defined as relative to this point of origin. When manipulating the graphic display, the cross may disappear momentarily. Rotating the view does not change the X and Y dimensions of the cell.
	Redirect	Allows you to select a different focal point in the graphic display of the cell. To change the center point of the graphic display window, click the icon or menu option. Use the cursor to point and click any spot in the scene. This now becomes the new center point of the display. All zooming of the view focuses on this point; all rotations revolve around this point. Press [Esc] to cancel the Redirect Camera mode.
	Drag3D	Activates the drag mode to relocate the entire cell in a new position. Click anywhere in the image and drag the entire cell to a new position. The drag mode remains in effect until you select another option or press [Esc] .
	Top	Resets camera position to display an overhead view of the cell. You can rotate or zoom in/out of this view. Use the vertical scroll bar or the mouse scroll wheel to alter the angle of the view.

3.5. THE CELLSETUP HELP MENU

The options of the Help menu are summarized in this table.

Help Menu		
Icon	Option	Summary
	Help	Opens the RoboCell for MHJF User Manual
	About	Shows the RoboCell software version
	Registration	Opens the Software License Window for registration, and for transferring RoboCell license to another computer. (See the Intelitek Software Licensing Guide for further information.)

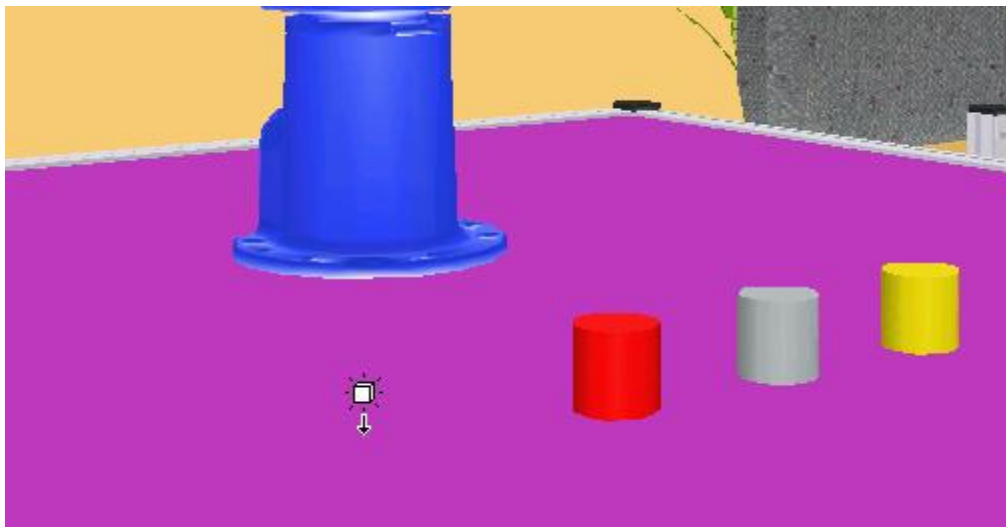
3.6. OBJECTS

You can place a number of objects into the workspace.

3.6.1. Object Placement

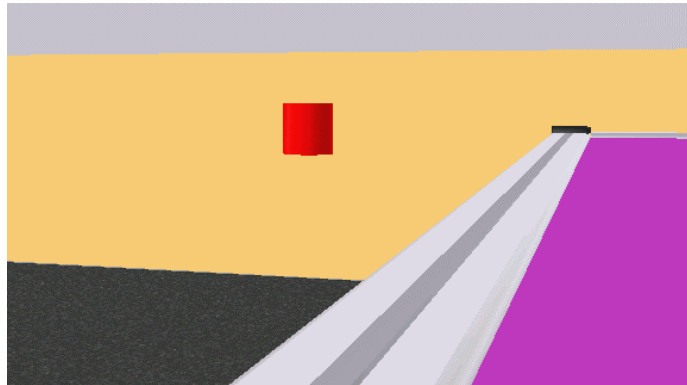
To place a new object in the cell, do the following:

1. Double-click the object's name in the New Objects menu.
2. Move the cursor into the virtual cell work area.
3. Point and click the location where you want to place the object. You may need to wait a moment for it to be displayed. *Do not double-click.*



If you have trouble placing an object in the exact spot you want, zoom in and out, and adjust the view from top to bottom until you can more accurately place the object.

Most objects are placed in the cell automatically at table level. These objects look as if they float in space unless a table has also been placed in the cell.



The height of some objects (template, jig, cylinder, and cube) is determined at the time they are placed in the cell.

If, for example, you click the conveyor when placing a cube in the scene, the cube is placed on the conveyor at the proper height.

It is recommended that a table be placed in the cell before any of these four objects is added to the cell. Otherwise, these objects may be set on the cell floor and are hidden when a table is added.

Once an object's height is set, it cannot be changed in CellSetup. For example, you cannot move the cube from the table onto the conveyor. Moving an object whose height has been set causes it to float in space or penetrate another object. If the file is saved with a floating object, the object falls to the surface below it when the file is loaded in RoboCell.

- To change an object's height in CellSetup, you must delete the object and create a new one.
- To move an object, simply click it and drag it to another position. Or use the object's properties menu to set precise position coordinates.

The robot cannot be dragged to another position. Its position can be changed only by means of its properties menu.

3.6.2. Object Configuration

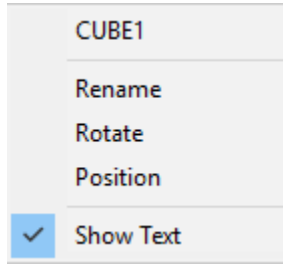
Selecting some of the objects from the New Objects menu opens a configuration dialog box for defining the object's permanent attributes (such as size and color). Attributes must be set before an object can be placed in a cell.

An object's configuration cannot be changed once the object has been placed in the cell. You must delete the original object and select and configure the object again from the New Objects menu.

Objects that have configuration options are described below.

3.6.3. Object Properties Menu

Double-clicking an object that has already been placed in the cell opens the Object Properties menu.



Object properties can be changed at any time during cell setup.

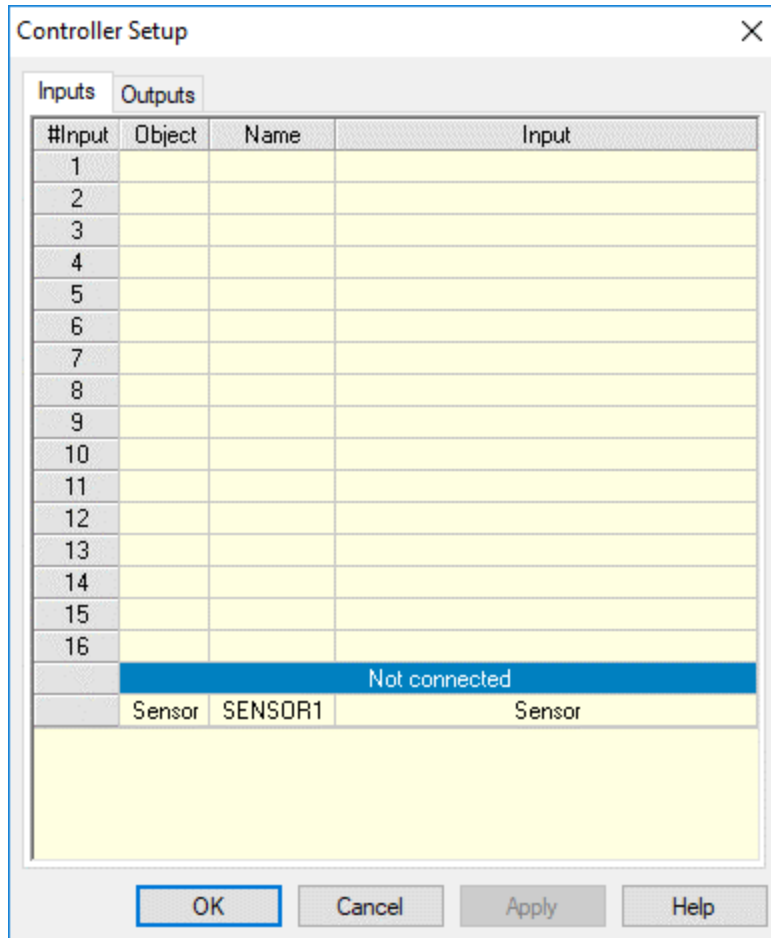
All object property menus contain the object’s name and the options listed in the table below. Some objects have additional properties which are described in later sections of this chapter.

Object Properties Menu	
Option	Summary
Rename	<p>Every object is automatically assigned a name and a number when it is placed in the cell. This name is displayed in the object’s properties menu. The number indicates the order in which objects of the same type were added to the cell (e.g., FEEDER1, GFEEDER2).</p> <p>Clicking Rename opens a dialog box that allows you to change the name of the object. The number can be changed or deleted.</p>
Rotate	<p>Objects can be rotated to any degree, in the minus (clockwise) or plus (counterclockwise) direction.</p> <p>Clicking Rotate opens a dialog box that allows you to change the object’s orientation.</p> <p>The degree of rotation is always defined relative to the object's default (predefined) orientation.</p>
Position	<p>The cell has a point of origin at table level defined by XY coordinates (0,0). All object positions are defined in XY coordinates relative to this point of origin.</p> <p>For most objects, these XY coordinates indicate the center of the object’s base.</p> <p>By default, the robot is placed in the cell with the center of its base (i.e., the robot's point of origin) at the cell’s point of origin.</p> <p>Clicking Set Position opens a dialog box that allows you to change an object’s X and/or Y coordinates.</p>
Show Text	<p>Shows/hides the label for the selected object.</p> <p>This option can only be activated when View Object Names or View Object Positions is selected.</p>

3.6.4. Defining Inputs and Outputs

The robot controller has sixteen digital input and sixteen digital output connections. Certain objects that are placed in the cell have a predefined number of input and/or output positions at which they can be connected to the controller.

Through the Controller Setup dialog box (below) you can view the defined objects and modify their controller input and output connections under the appropriate tab.



Devices that have not yet been connected are listed under **Not connected**.

To connect a non-connected device, click the device and drag it to a free input/output. A plus (+) sign is displayed next to the cursor.

To modify an existing connection, drag the input/output to a new setting. A swap sign is displayed next to the cursor. If you drag the device to a setting that already has a connected device, the new device replaces the previous device which is disconnected and relocated under **Not connected**.

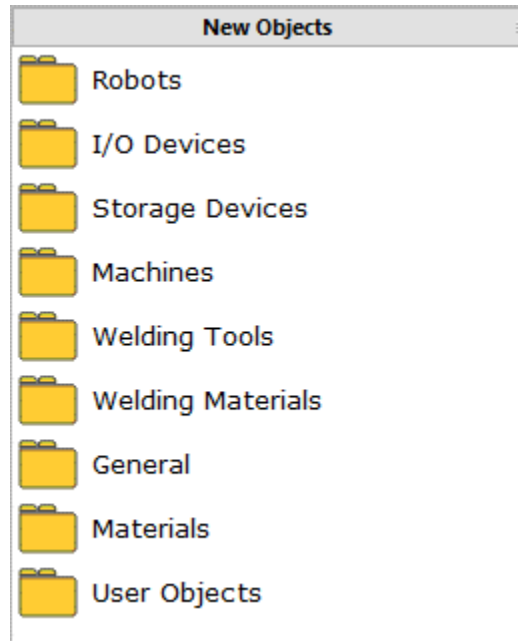
To disconnect input/output, drag the object to the **Not connected** list. A minus (-) sign is displayed next to the cursor.

Note: Outputs 1-4 are relay outputs. Rotary tables and conveyors can only be connected to these outputs.

3.7. NEW OBJECTS MENU

New objects are added to the cell using the New Objects menu. The objects are arranged in different folders.

To add a new object to the cell, click the desired object group (folder) and then select the specific object. The groups are detailed below.



3.7.1. Robots

The robot, which is automatically placed at the cell's point of origin, should be the first object you place in the cell.

The robot cannot be dragged to a new location. Its position can be changed only by means of the robot's Properties menu (see 3.6.3 Object Properties Menu, page 32).

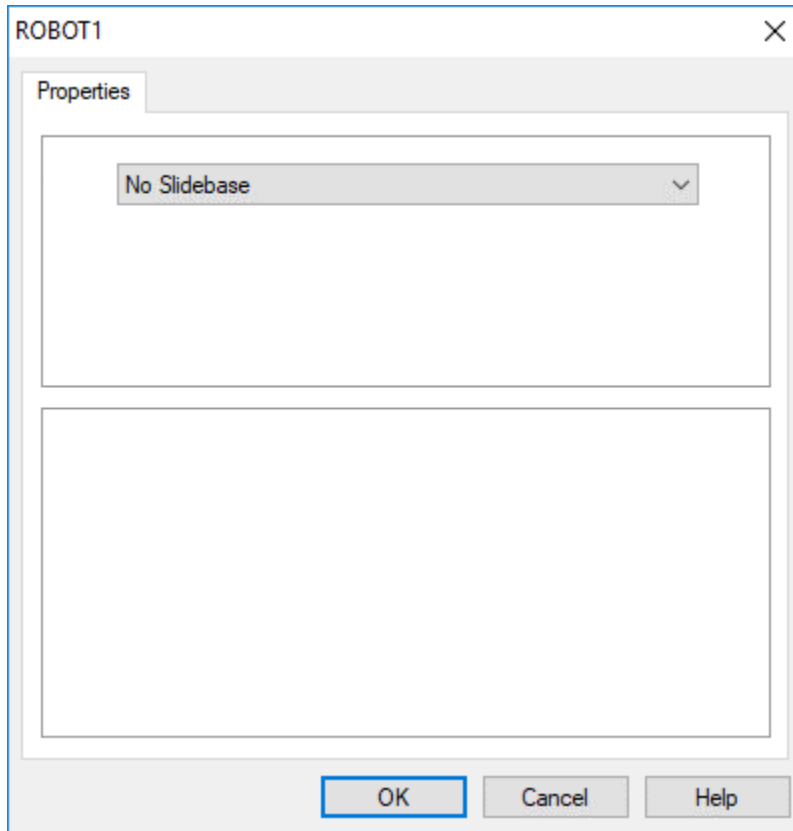
Only one robot can be placed in the cell.

To place a robot into the cell:

- Click **Robots**. Select **MHJF**.

3.7.1.1. Robot Configuration

A dialog box titled ROBOT1 opens allowing you to define whether or not the robot is mounted on a linear slidebase (LSB), a peripheral device that enables linear movements of the robot unit.

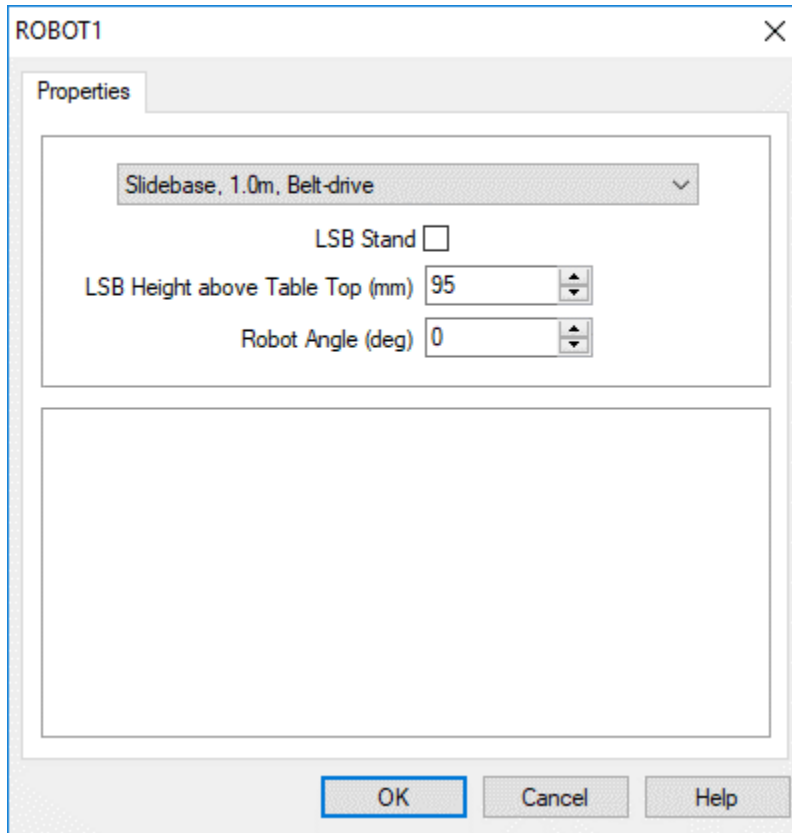


By default, the robot is set to have no slidebase. When you click **OK** to confirm this option, the robot is immediately placed in the cell.

Alternatively, these slidebase options are available:

- Slidebase 1.0m, Belt drive
- Slidebase 1.8m, Belt drive

If you select a slidebase, the Robot Configuration dialog box appears as follows:



Robot Configuration Dialog Box Properties Tab	
LSB Height	By default, the LSB Height is 100 mm.
Robot Angle	By default, the robot is mounted perpendicular to the slidebase at 90°.

If you select a slidebase, the robot and the slidebase are treated as one object in the cell.

When the MHJF is added to the cell with a linear slidebase (LSB), the LSB is automatically connected to the external axis E. The LSB can then be controlled in RoboCell using the E+/E- axis buttons of the Jog Control (Joint Coordinates) window.

3.7.1.2. Additional Robot Properties

In addition to the standard object properties (see 3.6.3 Object Properties Menu on page 32), this option is listed as a Robot property:

Additional Robot Properties	
Show Envelope	Displays the span of the robot's working range. This option facilitates the placing of objects within the reach of the robot.

3.7.2. I/O Devices

RoboCell I/O (input/output) devices can be connected to any free input/output terminal. The robot controller has sixteen digital input and sixteen digital output connections.

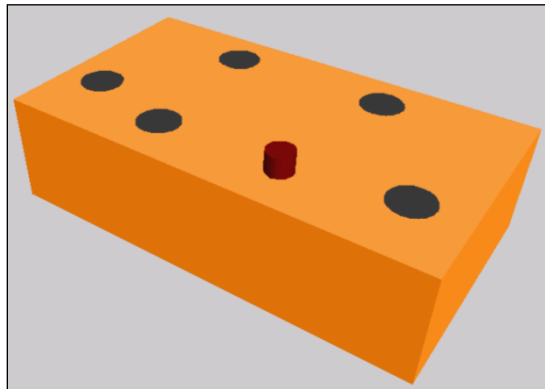
To place an I/O device in the cell, click the I/O Devices group and select the required device.

3.7.2.1. Experiment Table

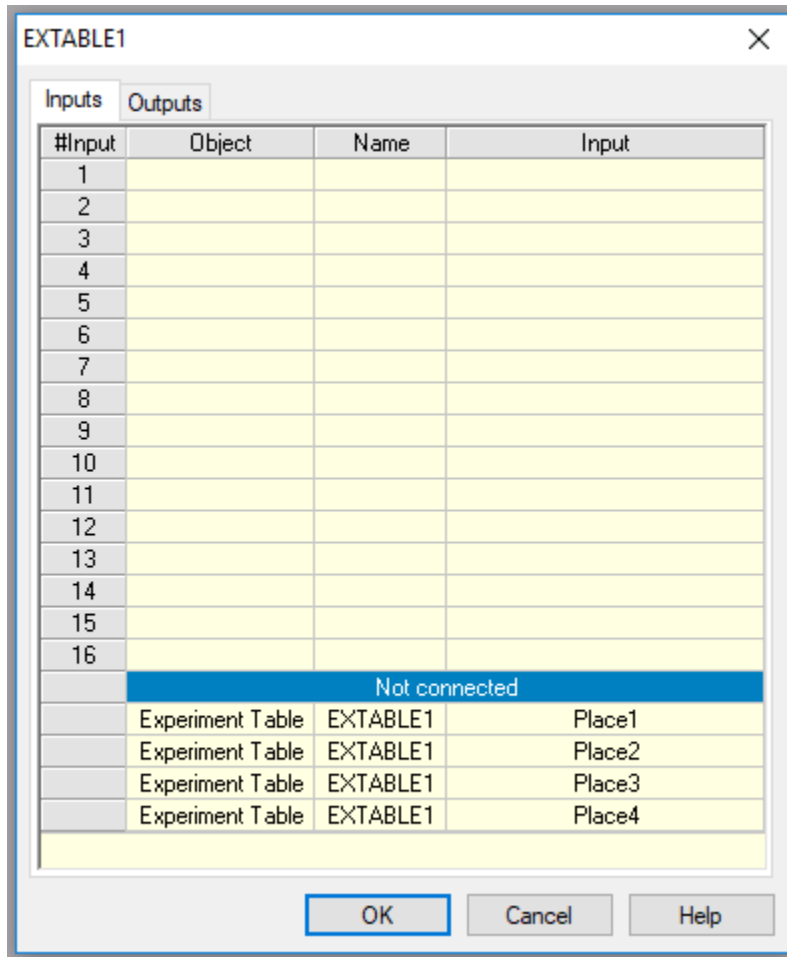
An experiment table includes the following input and output devices:

- Four (4) touch sensors – (input device). These are activated when there is a part on the switch, or something activates the switch.
- Lamp (output device)
- Buzzer (output device)

An experiment table with input and output devices is shown here.



When you select **Experiment Table**, this dialog box is displayed, enabling you to define the input and output connections:



Although one of the experiment table inputs may be disabled if the input is later defined for another device, all other switches remain connected to their defined inputs.

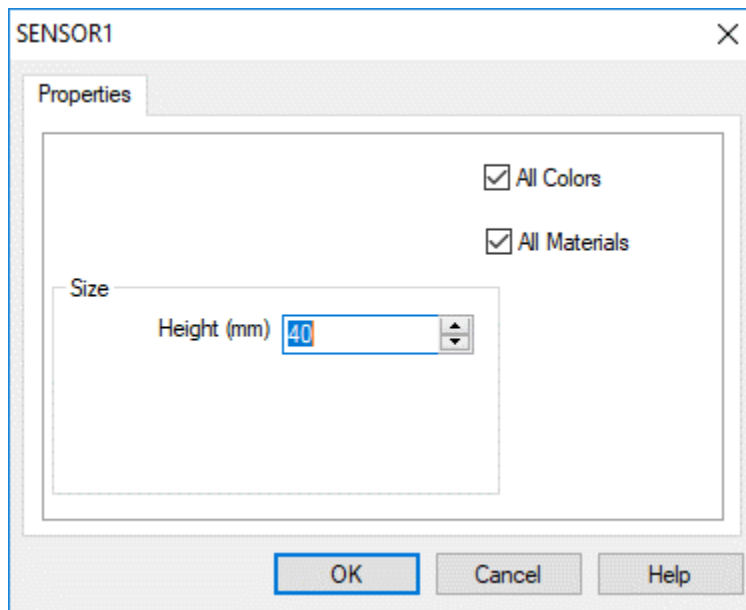
In RoboCell, a sound is emitted when the outputs that control the buzzer and lamp are turned on and off.

In addition, the lamp turns bright red when its associated output is turned on.

3.7.2.2. Sensor

RoboCell includes digital sensors that can be configured to detect specific objects based on their color and/or material (metal, plastic, wood, etc.). The objects are listed in the Materials group and their properties are defined when they are placed in the cell (For more information, see 3.7.9 User Objects on page 50.).

When you select **Sensor**, the dialog box below is displayed, enabling you to define the color/material to which the sensor is sensitive, as well as the sensor’s height.



Sensor Configuration	
Option	Description
Color Detect	By default, the sensor is set to detect all colors. To set it to a specific color, uncheck All Colors . Click Color Detect to open a color palette from which you can select the required color. If the sensor is configured to be sensitive to All Colors , the sensor in the cell is colored different shades of gray with its upper part in white. This distinguishes it from a sensor configured to be sensitive to gray, which is entirely gray.
Material Detect	By default, the sensor is set to detect all materials. To set it to a specific material, uncheck All Materials . Select a material (Metal, Glass, Plastics, or Wood) from the drop-down list.
Height (mm)	Defines the height (above table level) at which the sensor is placed in the cell. The default height is 40 mm. If you increase the height, the stand on which the sensor is mounted appears taller.

3.7.3. Storage Devices

The storage devices in RoboCell enable you to store, place, and retrieve objects listed in the Materials group.

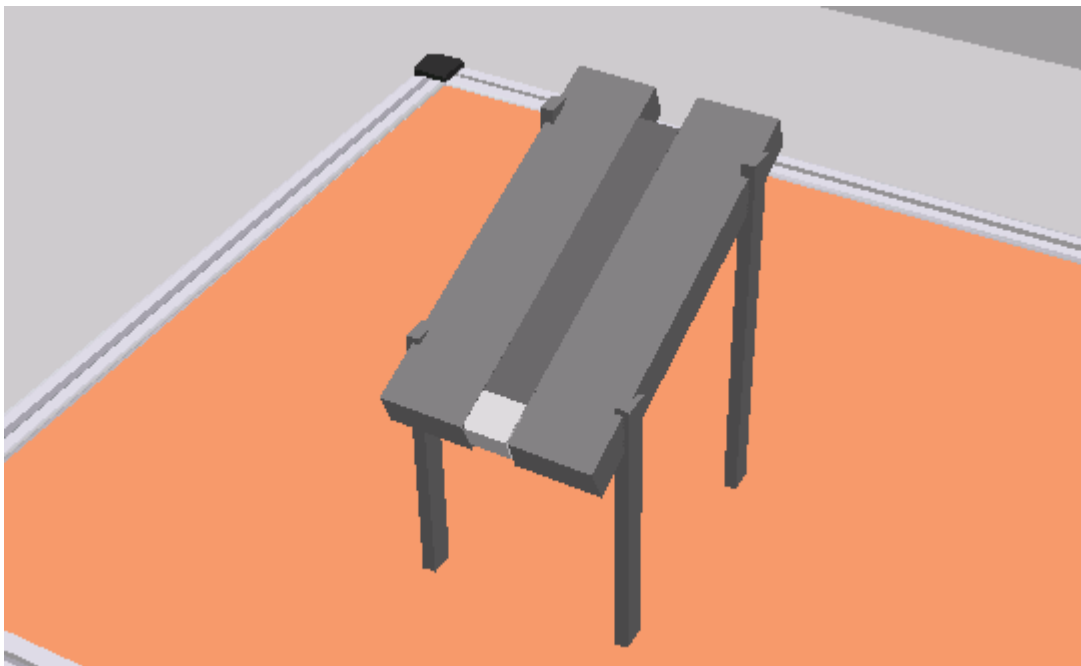
3.7.3.1. Gravity Feeder

The gravity feeder is an inclined plane on which objects are placed, one above the other.



Unlike most objects, the feeder's position is defined as the point at which the robot takes an object from the lower part of the feeder. All the remaining objects slide down the feeder to be taken in turn.

When placing objects on the gravity feeder, you must position them in the light gray section of the feeder's mouth.



The feeder is fitted with a touch sensor (named NotEmpty) located at the lower part of the feeder. The sensor indicates when there are parts in the feeder, i.e., it is on when the feeder is not empty.

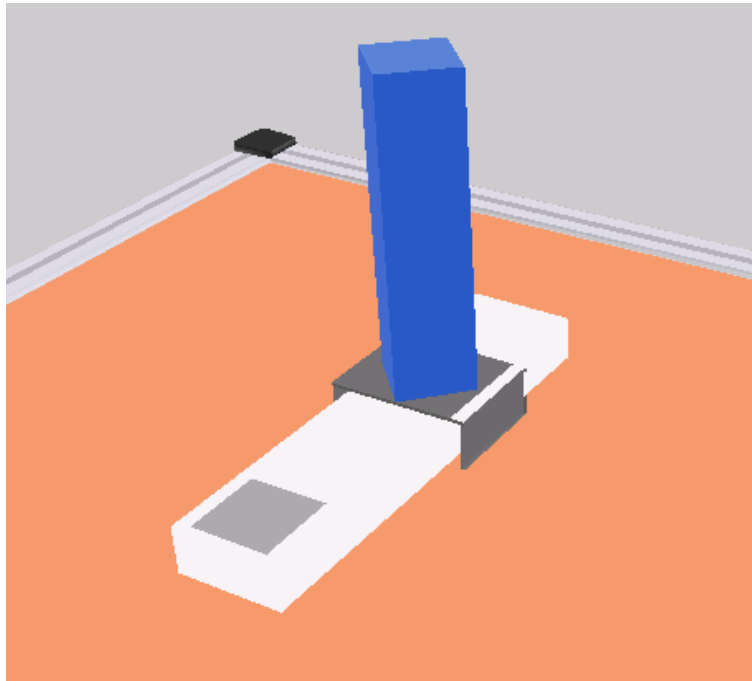
3.7.3.2. Feeder (Pneumatic Feeder)

The (pneumatic) feeder is constructed from a magazine of objects and a pneumatic cylinder fitted at the lower section.

The feeder is also fitted with a touch sensor that is pressed (the input is on) when there are parts in the feeder (NotEmpty sensor)

The pneumatic cylinder is controlled via a controller output. When the output is turned on, the piston extends, pushing out the next object.

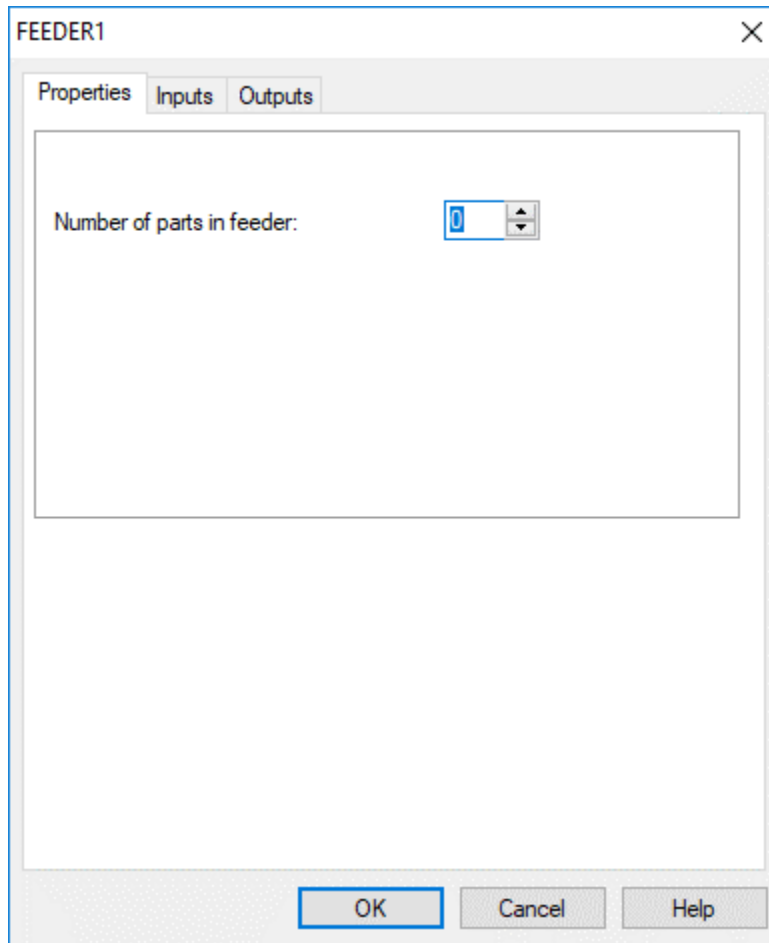
The robot picks the objects from the feeder output tray, and the feeder position is defined as the point at which the robot takes an object from the feeder.



3.7.3.3. Feeder Configuration

Configuration for the gravity feeder and the pneumatic feeder is described below. The differences between the two methods are indicated.

When you select a feeder, this configuration dialog box is displayed:



Feeder Configuration Properties tab:	
Number of Parts in Feeder	Defines the number of parts that are loaded in the feeder at the start of operation (when the cell is opened or reset). Only one material object (e.g., cylinder, cube) should be placed in a parts feeder. The software multiplies the object by the number of parts specified. Default: 0
Height at Front of Feeder (Gravity Feeder only)	Defines the height of the lower part of the feeder above the table level. Default: 200
Height at Back of Feeder (Gravity Feeder only)	Defines the height of the higher part of the feeder above the table level. Default: 280

Inputs tab:

To define the robot controller input to which the feeder's NotEmpty sensor is connected, click the Inputs tab.

Outputs tab: (Pneumatic feeder only)

To define the robot controller output to which the feeder is connected, click the Outputs tab. A new part appears in the feeder when the output is turned on.

Feeder Properties

In addition to the standard object properties (see 3.6.3 Object Properties Menu on page 32), this option is also listed in the Feeder Properties menu:

Additional Feeder Properties	
Capacity	Enables you to alter the number of parts loaded in the feeder after the feeder has been configured.

Jig, Rack, Template, Buffers, ASRS, and Bin

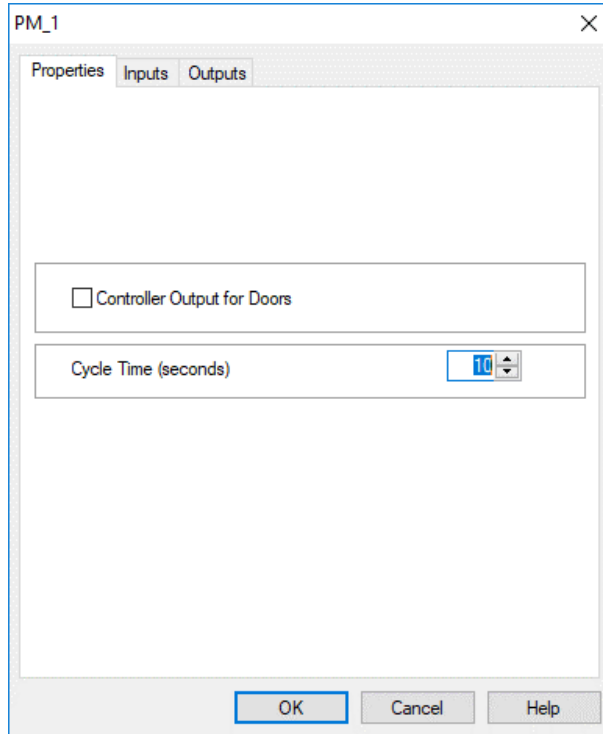
No settings are required for these storage devices. Double-click the desired device and place them in the cell at the desired location.

3.7.4. Machines

The Machines group contains six CNC milling machines (spectraLIGHT Mill 0200, proLIGHT Mill 1000, eXpertMill VMC-0600, Super proLIGHT Mill 1000, BenchMill 6000, and ProMill 8000) and four CNC lathes (spectraLIGHT Turn 0400, proLIGHT Turn 3000, BenchTurn 7000 and ProTurn 9000). A CNC controller controls these machines. The robot controller and the machine controller communicate through inputs and outputs. The machine output is the controller input, and vice versa.

All of these machines are configured in the same way.

When you select any of these machines, this configuration dialog box is displayed:



Machine configuration properties tab:	
Controller Output for Doors	Specifies whether or not the machine doors are controlled by the robot controller output. If the doors operate independently of the robot controller, they open and close at the beginning and end of the machine cycle.
Cycle time (seconds)	Sets the amount of time (in seconds) of the machine process. The default cycle time is 10 seconds.

Machine configuration inputs tab:	
Controller input (machine output)	The machine informs the controller that a process is being carried out, i.e., when the input is on, the machine is busy.

Machine configuration outputs tab:	
Controller output (machine input)	Closes and opens the machine door. The machine cannot start if the door is open.
Controller output (machine input)	Closes and opens the machine chuck or vise. The chuck or vise holds the object during the process.
Controller output (machine input)	Sets the robot controller output to which the machine is connected. The machine cycle starts when the output is turned on.

To modify the configuration after the machine has been placed in the cell, double-click the machine to display its Object Properties menu (see 3.6.3 Object Properties Menu on page 32). Select **Machine Setting** to display the Configuration dialog box.

3.7.5. Welding Tools

The Welding Tools group includes:

- Welding Table
- Welding Machine
- Gun Stand
- Welding Jig A
- Welding Jig B

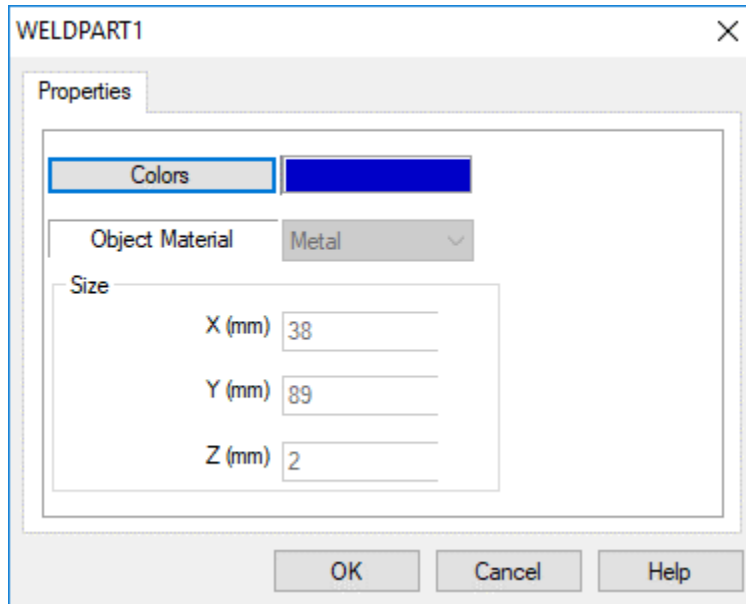
Welding Tools	
Welding Table	The welding table holds the welding object and it should be placed in the cell just before or after the robot (the first object) has been placed. Most welding objects are automatically placed at the correct default welding table height, even if they are not placed directly on the welding table.
Welding Machine	The welding machine supplies the electrical power required to weld the materials. It is connected to the welding gun by a cable. You cannot save a cell containing only a welding gun or only a welding machine. Neither or both must be present in order to save the cell.
Gun Stand	The gun stand holds the welding gun between welding sessions. The robot picks up the gun from the stand before starting the weld, and returns it to the stand when the welding is finished.
Welding Jig A	Welding Jig A is a device used to securely hold two materials being welded together in a T-joint.
Welding Jig B	Welding Jig B can be used for various types of welds.

3.7.6. Welding Materials

The Welding Materials group includes the plate and the metal cube.

3.7.6.1. Plate

Plate refers to the welding metal plates which are used with Welding Jig A. When you select **Plate**, this dialog box is displayed:

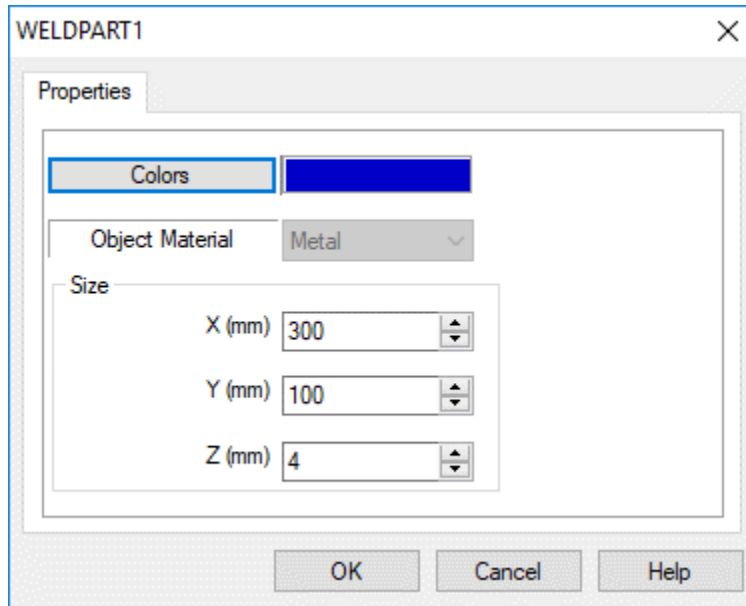


This table summarizes the options of the Plate Configuration dialog box.

Plate Configuration Dialog Box	
Option	Description
Colors	You can define the color of the plate to which the sensor is sensitive. The default color is blue. To change the color, click Colors . This opens a color palette from which you can select the required color. Make sure the color you select is compatible with the sensor's detection definition.
Size	The default dimensions of the plates are 38 x 2 x 89 mm. These are for display purposes only and cannot be changed.

3.7.6.2. Metal Cube

When you select **Metal Cube**, this dialog box is displayed:



The Metal Cube dialog box is described in this table:

Metal Cube Configuration Dialog Box	
Option	Description
Colors	Define the color of the cube in the same way as described for the plate.
Size	The default dimensions of the cube are: 300 x 4 x 100 mm. You can modify these dimensions.

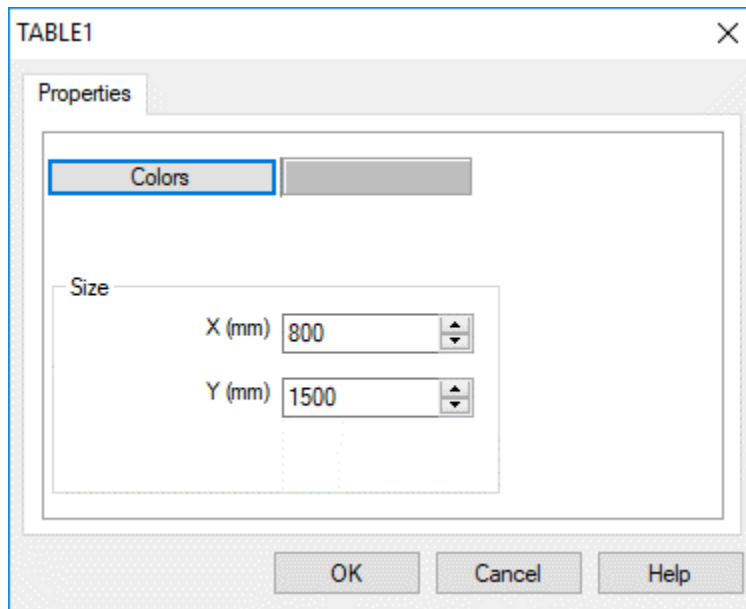
3.7.7. General

The General group includes a table and a computer. The computer is for display purposes only.

3.7.7.1. Table

It is a good idea to place the table in the cell just before or after the robot has been placed. It is also recommended that you place one or more tables in the cell so that objects do not float in space or lie on the floor. Most objects are displayed at the correct height, even if they are not sitting on tables.

When you select **Table**, this dialog box is displayed:



The dialog box is described in this table:

Table Configuration	
Option	Description
Colors	To change the color, click Colors . This opens a color palette from which you can select the required color.
Size	Defines the width and length of the table. X and Y dimensions are always in accordance with the X and Y dimensions of the cell. (Use the View Scene Origin option to display the X and Y dimensions of the cell.)

3.7.7.2. Table Properties

The Table Properties menu does not have a Rotate option (see 3.6.3 Object Properties Menu 32). Instead it has a Resize option for scaling the table's dimensions.

Table Properties	
Scale X; Scale Y	<p>Enter a ratio value (e.g., 2 or 0.5) in the X and Y fields. Both fields must contain a value. If you want to change only one dimension, be sure to enter 1 in the other field (do not enter 0).</p> <p>Resizing is progressive. The scale reverts to 1 each time the table is resized.</p>

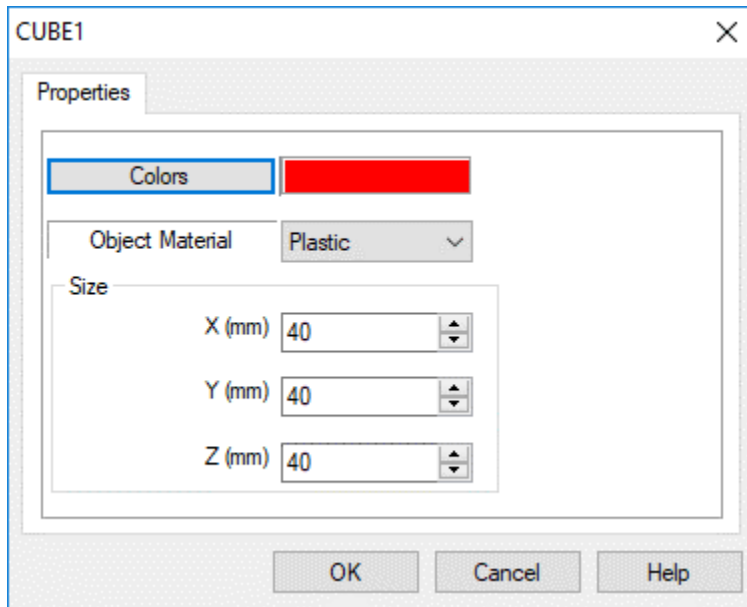
3.7.8. Materials

The Materials group includes objects that can be manipulated by the robot:

- Cube
- Cylinder
- Ball
- User defined parts (see 3.7.9 User Objects 50 below)

You can place as many objects as needed.

When you select **Cube**, **Cylinder**, or **Ball**, the Object Configuration dialog box is displayed:



The options of the Object Configuration dialog box are summarized here:

Object Configuration Dialog Box	
Option	Description
Colors	You can define the color of the plate to which the sensor is sensitive. The default color is red. To change the color, click Colors . This opens a color palette from which you can select the required color. Make sure the color you select is compatible with the sensor's detection definition.
Object Material	The cell sensor can differentiate between different materials. The default material is plastic. Other available materials are glass, metal, and wood. When metal is selected, you can perform welding using the features described in Chapter 7, Welding Features.

Object Configuration Dialog Box	
Option	Description
Size (mm) Cube: X, Y, Z Cylinder: Diameter, Height Ball: Diameter	The default dimensions of the objects can be modified if desired.
Horizontal/ Vertical (for Cylinder only)	Defines whether the cylinder is placed in a horizontal or vertical position. Cylinders that are placed in the lathe should be defined as horizontal.

3.7.9. User Objects

To insert an object or a part, select the desired part/object from the Materials/User Objects group.

Point and click the location where you want to place the part/object. You may need to wait a moment for the cursor to be displayed. *Do not double-click.*

3.7.9.1. Parts

Parts are basic shapes that can be used to simulate any type of material the user chooses. Simply click Part 1, 2, or 3 to add a new part to the cell.

In RoboCell, parts are changed to show what they look like at the end of the mill or lathe machining process. You may design your own finished parts.

User parts are stored as files in RWX format. The robot and machines can manipulate these objects. User part files are saved as PART*_0.RWX (where * represents 1, 2, or 3), and finished parts are named PART*_1.RWX. The files are located in the SBWSIMn RESOURCE folder.

3.7.9.2. Objects

The objects included in the User Objects group are for illustration purposes only. User Object files are named USER*.RWX. The files are located in the SBWSIMn\RESOURCE folder.

Advanced users who know how to create 3D objects in RWX format can design and use these objects in the robotic cell. No more than three user-defined parts and eight user objects may be used at one time. For further directions on designing user parts/objects, see 3.7.10 Designing User Parts and Objects on page 51.

3.7.10. Designing User Parts and Objects

RoboCell invites you to design your own parts and objects. Part design is covered in the sections below.

3.7.10.1. Modify an Existing User Part File

You may change the existing user part to create your own user part. To modify an existing user part:

1. Open the SBWSIMx folder.
2. Open the RESOURCE folder.
3. Using a text editor that saves files in plain ASCII format, (e.g., Notepad), open one of the existing user part files (PART1_0.RWX, PART2_0.RWX, or PART3_0.RWX). The file looks like this:

```
ModelBegin
TransformBegin
    Color 0.0 1.0 1.0
    Surface 0.4 0.3 0.2
    Opacity 1.000000
    LightSampling Facet
    GeometrySampling Solid
    TextureModes Lit
    Texture NULL
    ClumpBegin
Translate 0.0 0.0251 0.0
    Tag 1
    Block 0.05 0.05 0.05
ClumpBegin
Tag 2001
Color 1 0 0
Opacity 1
Block 0.03 0.02 0.07
ClumpEnd
    TransformEnd
ModelEnd
```

4. Edit the file to create your own part.
5. Save the file. Since you can only have three user part files at a time, you must replace the original file. *Do not change the file name.*
6. To create your own finished part, edit the Part*_1 file.

If you do not want to create a finished part, delete the Part*_1 file.

3.7.10.2. Create a New Part/Object File

If you can create your own user part and user object files, perform the following:

1. Open a text editor that saves files in plain ASCII text, such as Notepad or DOS Edit.
 2. Write the program to create your own part.
 3. Save as file type .RWX. (You must name the file PART1_0.RWX, PART2_0.RWX, or PART3_0.RWX.)
 4. To create a finished part, name the file PART*_1.RWX. There must also be a PART*_0.RWX file to go with it (i.e., in order for there to be a PART1_1 file there must be a PART1_0.RWX file).
 5. Copy the files to the SBWSIMn\RESOURCE folder.
- ⓘ Note:** *It is recommended that you do not discard the original files supplied with RoboCell. Save the original files under different file names.*
6. To create a User Object, follow the above steps and name the file USER*.RWX. USER*.RWX files cannot be manipulated by the robot or machines. You must replace the original user object files.

3.7.10.3. Import a CAD File

You can create a 3D object in a CAD program, and import it to RoboCell. To import a CAD file do the following:

1. Create a file in any CAD program. Save as *.DXF or *.3DC.
2. Use the *DXFtoRWX*, or *3DCto RWX* converter to convert the file to RWX format.
3. Name the files PART1_0.RWX, PART2_0.RWX, or Part3_0.RWX.
4. To create a finished part, name the file PART*_1.RWX. There must also be a PART*_0.RWX file to go with it (i.e., in order for there to be a PART1_1.RWX file there must be a PART1_0.RWX file).
5. Copy the file to the SBWSIMn\RESOURCE folder.

To import a *user object*, follow the above steps and name the file USER*.RWX.

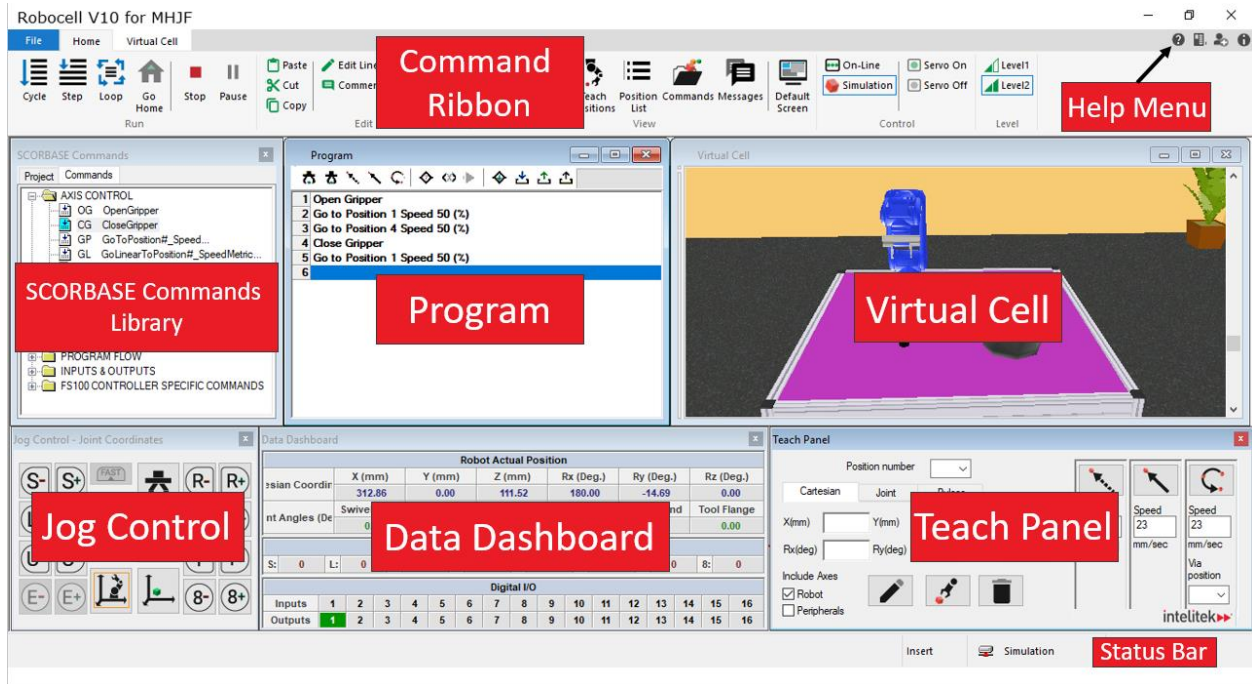
For more information on RWX files, consult a manual on RWX programming.

4. The Elements of RoboCell

4.1. OVERVIEW

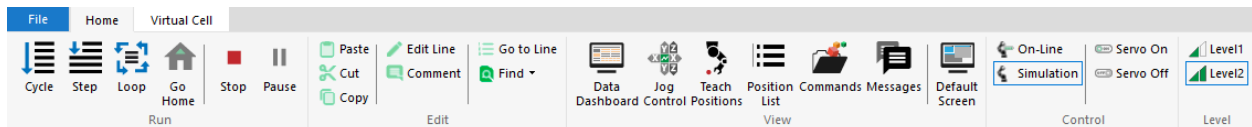
RoboCell integrates SCORBASE robotic programming with a graphic display module, which allows you to define robot positions and execute robot programs in a virtual robotic workcell.

An overview of the RoboCell application window is shown below. The various elements are discussed in the sections that follow.



4.2. THE COMMAND RIBBON

The top of the RoboCell window has a ribbon containing all commands and options. The ribbon has three tabs: **File**, **Home**, and **Virtual Cell**.

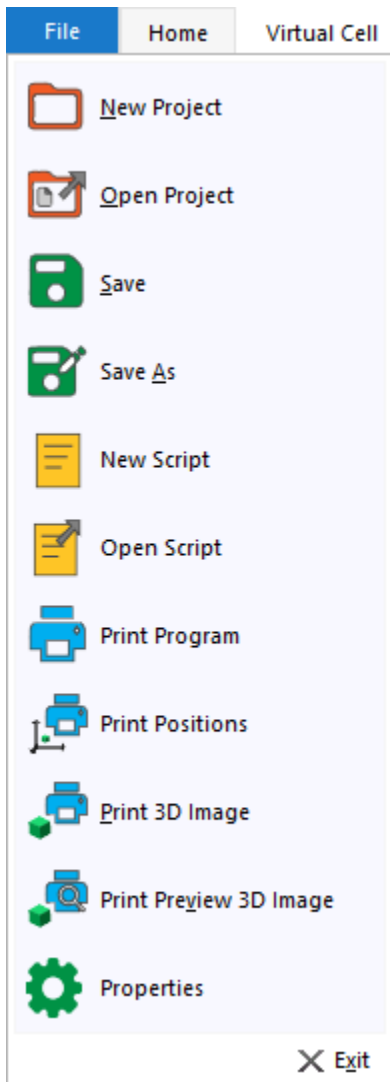


4.2.1. The File Tab












A RoboCell project includes the following files:

- SCORBASE program (file extension *.SBP)
- Recorded Position files (file extension *.PNT)
- Project data (file extension *.WS)
- Graphic image file (file extension *.3DC). In new RoboCell projects, this type of file must be imported from CellSetup.

The following options are available in the File tab:

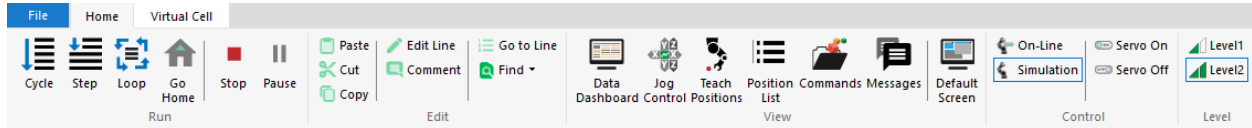


This table summarizes the options in the File Tab.

File Tab		
Icon	Option	Description
	New Project	This option opens a new untitled project. The new untitled project does not include a virtual cell (3D window). To add a virtual cell select Virtual Cell Import and select an existing *.3DC file.
	Open Project	Opens the Load Project window which lists SCORBASE files (without a virtual cell), and RoboCell files (that include a virtual cell)
	Save Project	Saves the project (program, positions, and graphics)
	Save Project as...	Saves the project (program, positions, and graphics) under a different name
	New Script	Opens Notepad for writing a new Visual Basic script. Save the file with a .VBS extension in the same directory as the SCORBASE project file. The best practice for naming the file is: <project name>.VBS.
	Open Script	Opens the File Selector for opening an existing Visual Basic script file.
	Print Program	Prints the program.
	Print Positions	Prints the user-defined positions.
	Print 3D Image	Print a snapshot of the current view angle of the virtual cell.
	Print Preview 3D Image	Displays a print preview of the snapshot of the current angle of the virtual cell.
	Properties	Displays the RoboCell Properties menu, where you can change preferences such as whether or not to load the last open project when RoboCell is started.
X	Exit	Closes RoboCell.














4.2.2. The Home Tab

The Home tab, shown here, contains most of the main RoboCell operations.



The Home tab commands are detailed in this table.

Home Tab		
Icon	Option (Shortcut)	Description
	Run single cycle (F7)	Executes the program from the selected (highlighted) program line to the end of the program.
	Run single line (F6)	Executes the selected (highlighted) program line. See 10 Program Execution on page 109 for more information.
	Loop (F8)	Executes the program from the selected (highlighted) program line. When the last program line is reached, the program starts again from the first line.
	Home	Sends the robot to its home position.
	Stop (F9)	Immediately stops program execution and movement of all axes.
	Pause (F10)	Stops program execution after the current line is executed.
<p>Note: <i>Pause and Stop are software methods for halting program execution. In an actual emergency situation, you should use the EMERGENCY button on the programming pendant.</i></p>		
	Cut (Ctrl+X)	Deletes selected text or lines from the program lines, and places it on the Windows and SCORBASE clipboards.
	Copy (Ctrl+C)	Places a copy of selected text or lines from the program lines on the Windows and SCORBASE clipboards.
	Paste (Ctrl+V)	Inserts the contents of the SCORBASE clipboard into the program lines.
	Find (Ctrl+F)	Opens a dialog box that allows you to search for a particular text string, SCORBASE command, or command argument.
	Find Next (F3)	Repeats the last Find operation for the next occurrence.
	Comment	Inserts/deletes asterisk at beginning of a SCORBASE program command line. This action toggles the command line between a remark and an executable command.
	Go to Line...	Opens a dialog box that displays the total number of lines in the program and prompts you for a line number. The program editor jumps to the line you specify.

Home Tab		
Icon	Option (Shortcut)	Description
	Data Dashboard	Shows/hides the Data Dashboard. The Data Dashboard displays the current position of the robot and shows which inputs and outputs are activated.
	Jog Control	Shows/hides the Jog Control window. The jog control window enables manual control over the movements of the robot, the gripper, and peripheral axes.
	Teach Positions	Shows/hides the Teach Panel. This area enables recording, teaching, and deleting positions. It also enables the user to send the robot and/or peripherals to a previously defined position.
	Position List	Shows/hides a list of position in the project file. The list includes position coordinates in degrees and/or axis encoder data.
	Commands	Shows/hides the SCORBASE Commands Library, the library from which you can add commands to your program.
	Messages	Opens the Messages Window. The data displayed in the Messages window is printed using the PS (Print to Screen & Log) command.
	Default Screen	Displays the SCORBASE Commands Library, Jog Control window, Teach Panel, Program window, Data Dashboard, and Virtual Cell.
	On-Line	Initiates On-Line mode, in which RoboCell takes control over the physical MHJF controller. For On-Line mode to initialize successfully, the programming pendant must be set to REMOTE mode, the USB dongle (key) is inserted into the USB slot on your computer, and the dongle's driver is installed.
	Simulation	Initializes simulation mode, which is the normal RoboCell operating mode when your computer is not connected to the actual robot.
	Servo On	In On-Line mode, this button allows RoboCell to control the physical MHJF.
	Servo Off	In On-Line mode, this button turns off RoboCell's control of the physical MHJF.
	Level 1	Displays only basic commands and options and disables advanced commands and options.
	Level 2	Displays and enables all commands and options.











4.2.3. The Virtual Cell Tab

All graphic display tools are accessible from the Virtual Cell tab.



These tools help you adjust the camera view and quickly and easily change the position of the robot. The options in the tab are described in this table.

Virtual Cell Tab		
Icon	Option	Description
	Reset	This option returns the virtual cell image to the initial setting and position (saved in CellSetup). This tool is useful when you want to initialize the virtual cell before re-running a program.
	Origin	This option shows the origin of the cell at floor level (under the center of the robot's base) as a red cross marked with X- and Y-axes. The positions of all objects are defined as relative to this point of origin. When manipulating the graphic display, the cross may disappear momentarily.
	Import	This option opens the Import 3D Model window which lists graphic module files (*.3DC files). Select one of the files to open the graphic display module window. When you save the project, the current workcell is included as part of the project.
	Edit	This option opens CellSetup. If the project already includes a virtual cell, CellSetup opens that workcell for editing. Otherwise, CellSetup opens with a new untitled empty cell.
	Stop (F9)	Immediately stops program execution and movement of all axes.
	Send to Object	This option sends the robot to the selected object. It is useful for defining a robot pick position. Make sure the gripper is open before sending the robot to the object.
	Send to Point	This option sends the robot to the selected point. It is useful for defining a robot place position.
	Send to Above	Sends the robot to a point (by default, 100 mm) above the selected point on the table.
	Options for Send Robot	Opens the Send Robot Options dialog box enabling you to define the vertical offset values and the gripper's orientation when Send Robot commands are executed. For more information, see section 5.2.1 Send Robot on page 68.
	Show Path	When selected, a line showing the gripper path is drawn on the screen as the robot moves. The distance between the points is proportional to the TCP speed.
	Clear Path	Clears the robot path that was drawn using the Show Path option.
	Show Envelope	Displays the span of the robot's working range, allowing you to see whether objects are within the reach of the robot.

Virtual Cell Tab		
Icon	Option	Description
	Names	Shows/hides object name labels.
	Positions	Shows/hides object position labels. The label displays the X and Y coordinates of the object position that is relative to the coordinates of the 3D model.
	Relative to Robot	Shows/hides Object Position Robot Coordinates. The object position displayed is relative to the robot coordinates. The robot coordinates may be different from cell coordinates if: <ul style="list-style-type: none"> • The robot was moved after initial setup (in CellSetup). • The robot is mounted on a linear slidebase. When the robot moves, the position of objects relative to the robot changes.
	Number Parts in Feeder	Shows/hides a label that displays the number of remaining objects in a feeder.
	Save (Camera Position)	This option saves the current camera position. Only one camera position can be saved. Saving a new position overwrites the previously saved position.
	Restore	Restores the position recorded using the Save (Camera Position) option.
	Redirect (Camera)	Selects a new center of interest in the virtual cell. Redirect mode remains in effect until you select another option or press [Esc] .
	Follow Me	This option enables you to follow the movement of any object (robot gripper, certain object, etc.) when the cell is running. To select the point, choose a point on a moving object (part of the robot or an object). Press [Esc] to exit this mode.
	Top View	Resets camera position to the top center point of the cell.
	Drag Image	This option activates the drag mode in order to drag the entire image to a new position. Using the cursor, point and click anywhere on the image and drag it to its new position. The drag mode remains in effect until you select another option or press [Esc] .

4.3. THE ROBOCELL HELP MENU

This table describes the Help menu options.

Help Menu		
Icon	Option	Summary
	Help	Opens the RoboCell for MHJF User Guide
	Scripting Help	Opens the scripting help
	Registration	Opens the Software License Window for registration, and for transferring RoboCell license to another computer. (See the Intelitek Software Licensing Guide for further information.)
	About	Shows the RoboCell software version

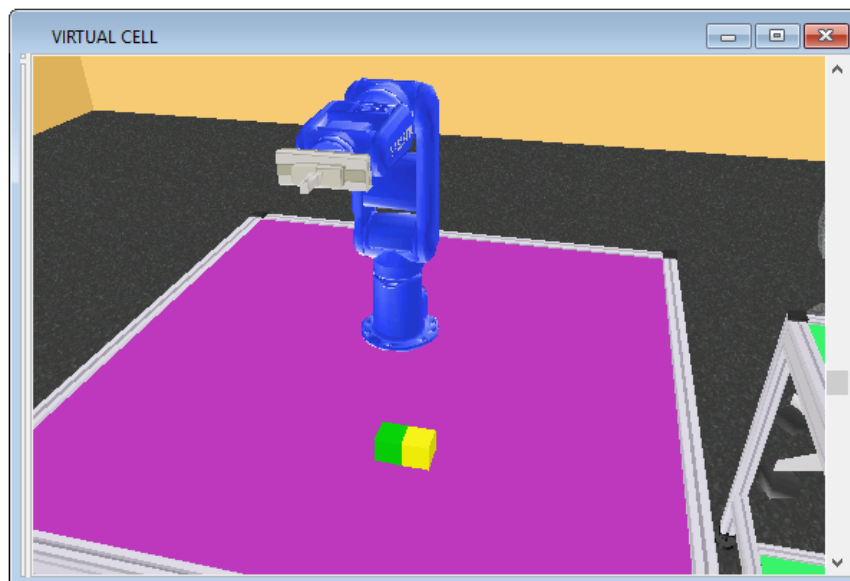
4.4. THE VIRTUAL CELL WINDOW

The virtual cell displays the graphic module file, which is an integral part of a RoboCell project. Opening or saving a project opens or saves the RoboCell program, robot positions, and graphic module data as one entity.

You can open a new RoboCell project and then import a virtual cell from an existing project or a cell created using CellSetup. **(Navigate to Virtual Cell | Import.)**

Only one CellSetup file/window can be opened at a time.

When you open a project which already contains a virtual cell, this window is displayed:



The window displays the robotic workcell and all its components. Using the graphic display tools described in section 4.2.3 The Virtual Cell Tab, you can manipulate the image and view it from different angles and positions. Other tools enable you to see other data related to the cell (such as object name, position or the number of part in a feeder).

4.4.1. Graphic Interface

The 3D image you see on your screen is the output of a virtual video camera fitted in the virtual cell. You can use the mouse to manipulate the image.

The graphic display in both CellSetup and RoboCell have the same graphic interface.

4.4.1.1. Rotate

To rotate the scene (move the camera around the center of the image):

1. Place the cursor anywhere in the graphic window and click the right mouse button. The cursor changes to a magnifying glass.
2. Keep the right mouse button pressed and drag to the right to rotate the scene counterclockwise.
3. Similarly, keep the right mouse button pressed and drag to the left to rotate the scene clockwise.



The image rotates around its center point. To change the center point, use the Redirect Camera tool.

4.4.1.2. Zoom

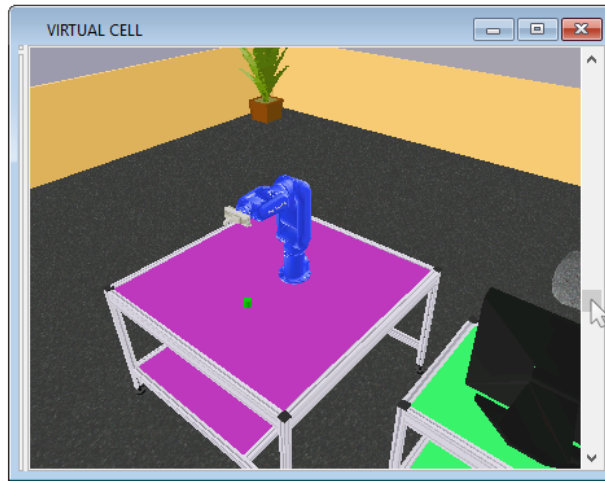
To zoom in or out:

1. Place the cursor anywhere in the graphic window and click the right mouse button. The cursor changes to a magnifying glass.
2. Drag up to zoom in.
3. Drag down to zoom out.

The image zooms in and out while its center point remains fixed. To change the center point, use the Redirect Camera tool.

4.4.1.3. Angle

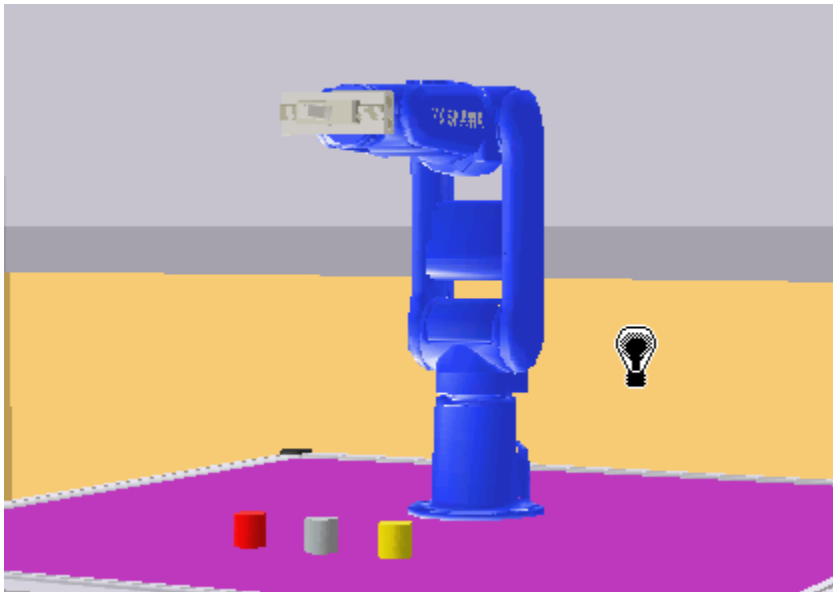
To change the angle of the overhead scene (to move the camera up or down), either click on the vertical scroll bar and drag it up or down or use the mouse scroll wheel.



The camera moves up or down while the center point of the image remains fixed. To change the center point, use the Redirect Camera tool.

4.4.1.4. Brightness

To change cell illumination (bright or dark), click the right mouse button and drag in any direction while simultaneously pressing the Ctrl key. The cursor changes to a light bulb.



4.5. OTHER ROBOCELL WINDOWS

RoboCell contains several windows that help you manipulate, teach, and program the robot. These windows include the Data Dashboard, Teach Panel, Jog Control window, SCORBASE Commands Library, Program window, Messages window, and the Virtual Cell window. The functions of these windows are summarized below, and the windows are discussed in greater detail in the later sections of this guide. The Virtual Cell Window is also detailed above in section 4.4.

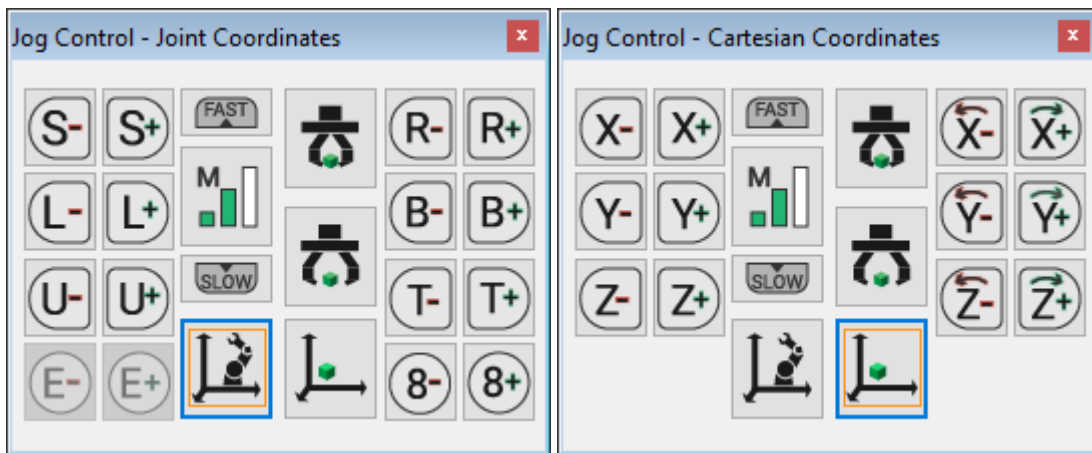
- **SCORBASE Commands library:** Displays the SCORBASE commands. You can add a command to the Program window by double-clicking it.
- **Program window:** Displays the program. The next command to be run is highlighted.
- **Jog Control window:** Allows you to manually manipulate the robot and to open and close the gripper. See section 5.1 The Jog Control Window for more details.
- **Data Dashboard:** Displays the robot position in 3D space, joint data, and status of inputs and outputs. See section 6.5 Data Dashboard for more details.
- **Virtual Cell:** Displays the robot in its virtual environment.
- **Teach Panel:** Allows you to record and teach positions. You can also send the robot to a position using this window. See section 6.2 Teach Panel for more details.
- **Position List:** Displays coordinate and axis information for all positions in the project. See section 6.4 Position List for more details.
- **Messages window:** Displays printed messages.

5. Cell Operation

5.1. THE JOG CONTROL WINDOW

5.1.1. Overview

The Jog Control window allows you to manipulate the robot and peripheral axes in the simulated cell in the same way that it provides control over an actual robotic system. Jog control has two modes, as shown in the images here.



5.1.2. Coordinate Modes

Two coordinate systems, or modes, can be used in RoboCell:

- **Joint coordinates mode**, in which each joint of the robot is moved independently. This mode is shown above left, and is the default state of the Jog Control window.
- **Cartesian (XYZ) coordinates mode**, in which joints are controlled synchronously to provide movement along and about the X-, Y-, and Z- axes. This mode is shown above right.

You can toggle between joints coordinate mode and Cartesian coordinate mode by using joints mode and Cartesian mode buttons shown here:



Joints mode button

Cartesian mode button



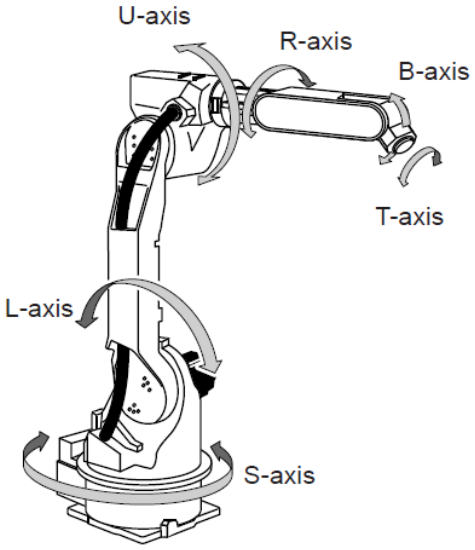














5.1.3. Manual Axis Movement

Rotating the virtual robot axes is performed by clicking the buttons in the Jog Control window. Rotation of an axis continues for as long as a button is pressed or until a joint limit is reached or a collision occurs.

The tables below explain how clicking the buttons in the Jog Control window controls the robot and peripheral movements.

5.1.3.1. Movement in Joints Mode

When joints mode is selected, clicking the buttons (or pressing the corresponding keys on the keyboard) rotates one robot axis at a time, as described below:

Joints Mode Axis Movement		
Buttons	Joint Motion	Legend
 	Rotate the S-axis (swivel base joint) left and right.	
 	Rotate the L-axis (lower arm joint) up and down.	
 	Rotate the U-axis (upper arm joint) up and down.	
 	Rotate the R-axis (arm roll joint) left and right.	
 	Rotate the B-axis (wrist bend joint) up and down.	
 	Rotate the T-axis (tool flange joint) left and right.	
 	Move peripheral axis #7 (LSB, if connected).	
 	Move peripheral axis #8 (if connected).	

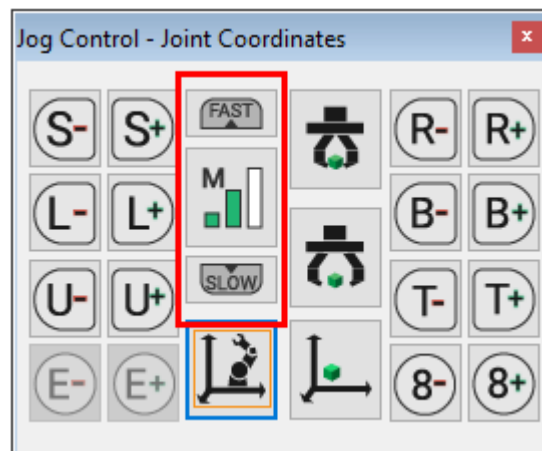
5.1.3.2. Movement in Cartesian Mode

When Cartesian (XYZ) mode is selected, clicking the buttons moves the TCP as described in the table below. Movements in Cartesian mode are typically a combination of simultaneous movements of a few joint axes.

Cartesian Mode Axis Movement		
Buttons	XYZ Motion	Legend
	Tool center point (TCP) moves parallel to the X-axis (back and forth).	
	TCP moves parallel to the Y-axis (right and left).	
	TCP moves parallel to the Z-axis (up and down).	
	Wrist axes rotate about the X-axis with the TCP fixed.	
	Wrist axes rotate about the Y-axis with the TCP fixed.	
	Wrist axes rotate about the Z-axis with the TCP fixed.	

5.1.3.3. Axis Speed

The Jog Control window's speed option controls the speed of manual movement.



There are four manual speeds. From slowest to fastest, these speeds are inching (I), low (L), medium (M), and high (H).

5.1.4. Manual Gripper Operation

The gripper control options of the Jog Control window are described in this table:

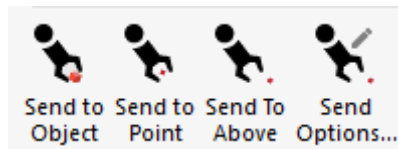
Gripper Options		
Icon	Option	Description
	Open Gripper	Completely opens the gripper
	Close Gripper	Completely closes the gripper

Press the respective buttons once to open or close the gripper.

5.2. SEND ROBOT COMMANDS

5.2.1. Send Robot Functions

Three Send Robot buttons in the Virtual Cell tab allow you to quickly manipulate the robot to a desired location. While these options do not exist in the MHJF’s programming pendant, the functions are available in RoboCell.

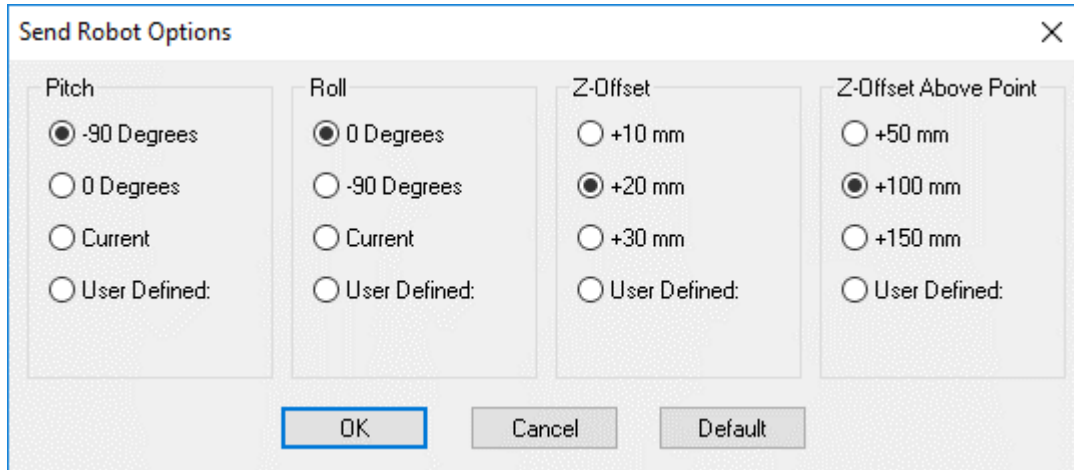


- **Send to Object:** This option, useful for defining a pick position, sends the robot’s TCP (tool point center) to the selected object.
- **Send to Point:** This option sends the robot’s TCP to the selected point. It is useful for defining a place position.
- **Send to Above:** Sends the robot’s TCP to a point (by default, 100 mm) above the selected point on the table.

5.2.2. Send Options

The **Send Options...** button, which opens the Send Robot Options dialog box, allows you to define some of the parameters of the three Send Robot buttons.

The Send Robot Options dialog box is shown here:



The settings remain in effect for all subsequent Send Robot commands.

By default, the robot moves to an object or point with the gripper perpendicular (-90 degrees) to the table and with no (0) rotation.

You can also use Jog Control window (see section 5.1.3 Manual Axis Movement) to adjust the orientation (pitch [Rx], roll [Ry], and yaw [Rz]) of the wrist and gripper. You can then select the option **Use Current** to maintain the gripper's orientation during subsequent Send Robot commands.

The Z-offset value is used by the Send Robot to Object and Send Robot to Point commands. The Z-offset Above Point value is used by the Send Robot Above Point command.

5.3. GRIPPER

The Gripper is a dynamic and important part of your robot. The features of the gripper are covered in the sections below. To learn about manual operation of the gripper, see section 5.1.4 Manual Gripper Operation.

5.3.1. Grasping Objects

The robot gripper in RoboCell grasps objects only as a result of Close Gripper commands.

When the gripper closes on an object, the object is pushed into the center of gripper.

In addition, cubes and horizontal cylinders are rotated into alignment with the gripper jaws, provided the rotation offset does not exceed 35 degrees. When the offset is about 45 degrees, the gripper simply grasps the object at its corners.

Since objects are rotated around their center, a long object that is grasped far from its center must be aligned with the gripper as much as possible before the gripper is closed.

5.3.2. Releasing Objects

RoboCell operates in a gravitational field. Objects that are released from the gripper fall to the highest surface below them.

If more than half of an object's base rests on a surface, the object remains on the surface. Otherwise it falls onto the surface below.

Objects can be picked up and released at any rotation.

5.4. IMPACT DETECTION

RoboCell checks for the following impact conditions:

- The tip of the gripper hits an object or the robot itself.
- The edge of the gripper motor (mounted on top of the gripper) hits an object or the robot itself.
- An object held by the gripper hits another object or the robot.

RoboCell's impact error message is sent when the robot cannot reach a target position.

In response to the impact error message:

1. Click **OK** to resume.
2. Move the robot away from the impact condition.

If a Send Robot command caused the impact, the robot resumes movement from the position that preceded the Send Robot command.

When RoboCell is operating on-line with an actual robotic cell, the Yaskawa controller controls collision detection and response, and the simulation's collision detection function is disabled.

6. Positions

6.1. OVERVIEW

RoboCell provides the methods described below for defining (referred to as *recording* or *teaching*) robot positions. A position is identified by its assigned number.

6.2. TEACH PANEL

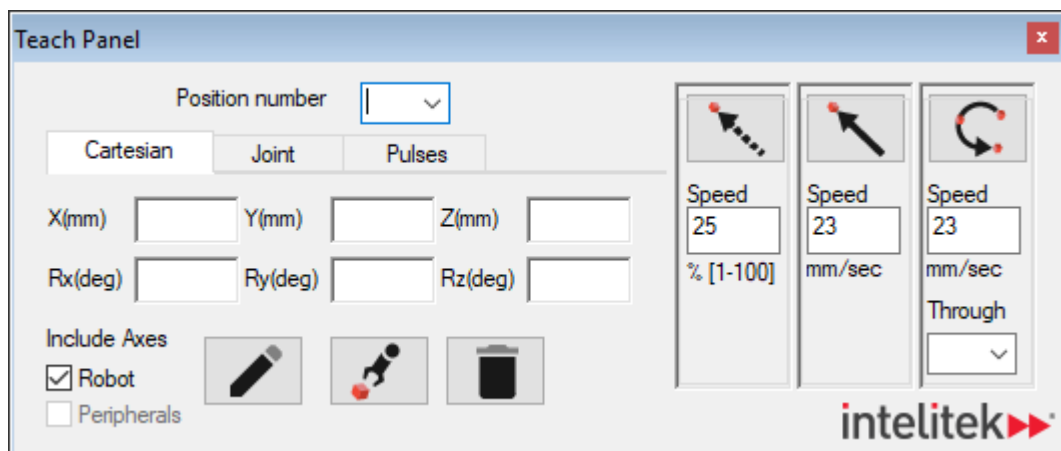
The Teach Panel enables the following:

- Teaching positions (by inputting Cartesian coordinates).
- Recording the current robot position.
- Sending the manipulator to a defined position using the following methods of interpolation:
 - Joint (Go to Position)
 - Linear (Go Linear)
 - Arc (Go Circular)

To display the description of the function of each icon in the Teach Panel, simply hover the mouse over the desired icon without clicking a mouse button.

You can define up to 1,000 positions.





The Teach Panel (RoboCell Level 2) is shown here:





Positional data can be displayed in Cartesian coordinates, joint angles, or encoder pulses. The three tabs in the teach panel allow to toggle between the display modes.

Teach Panel Positional Data Display Tabs						
Display Mode	Tab					
Cartesian	Cartesian		Joint		Pulses	
	X(mm)	434.36	Y(mm)	17.33	Z(mm)	273.70
	Rx(deg)	90.00	Ry(deg)	-90.00	Rz(deg)	90.00
Joint Angle	Cartesian		Joint		Pulses	
	S	3.75	L	-91.06	U	91.32
	R	94.06	B	3.76	T	-94.06
Encoder Pulses	Cartesian		Joint		Pulses	
	S	5121	U	-3302	L	-2005
	R	128419	B	-3850	T	-53512

The Teach Panel offers the following options:

Teach Panel	
Option	Description
Position Number	A numerical name for the position. All of the positions in the project can be selected from the Position Number dropdown menu.
 Teach	Teaches position using the Cartesian Coordinate System.
 Record	Records the current robot position (in joint coordinates) to the position displayed in the position number field.
 Delete	Deletes from memory the position in the position number field.
X(mm), Y(mm) Z(mm), Rx(deg), Ry(deg), Rz(deg)	Fields for displaying or changing the Cartesian coordinates of the selected position.
 Go to Position	Executes the Go to Position command, which sends the robot's TCP (Tool Center Point) from its current position to the selected position

Teach Panel	
Option	Description
 Go Linear (to Position)	Executes the Go Linear to Position command. Sends the axes in a straight line to the selected position
 Go Circular (to Position)	Executes the Go Circular to Position command. This command sends the robot in a circular path to the target position (in the position number field), via the position specified in the Via position field. The circular motion applies only to the robot.
Via position	Selects the intermediate position through which the Go Circular movement passes
Speed	Selects the speed for all movement commands in percent (for Go to Position only) and mm/sec (for Go Linear and Go Circular)

6.3. DEFINING POSITIONS (JOINT COORDINATES)

This section covers the procedures involved in defining positions.

Although the terms *teach* and *record* are often used interchangeably, RoboCell makes the following distinction:

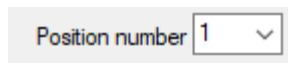
Record position: defines a robot position in a Joint coordinates system.

Teach position: defines a robot position in a Cartesian coordinates system.

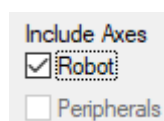
6.3.1. Recording Positions

To record a position:

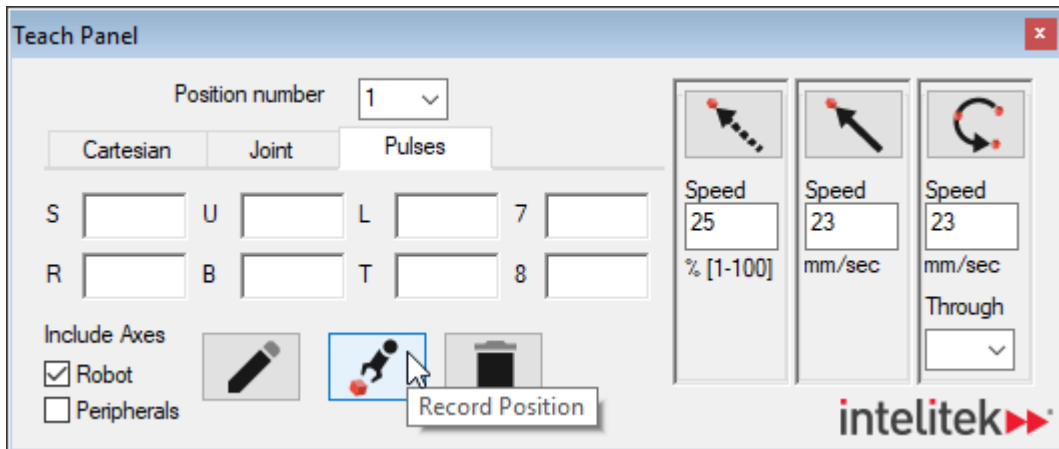
1. Use either the Jog Control window or the Send Robot to Object/Position/Above Position tools to send the robot to the desired position.
2. In the Teach Panel's **Position number** drop-down menu, enter a new position number or select an existing position number (if you want to overwrite a previously defined position).



3. Select **Include Axes (Robot and/or Peripherals)**:
 - a. **Robot**: to record a position for the robot axes.
 - b. **Peripherals**: to record a position for the peripheral axes (only available if there are peripheral devices in the cell, such as a linear slidebase).



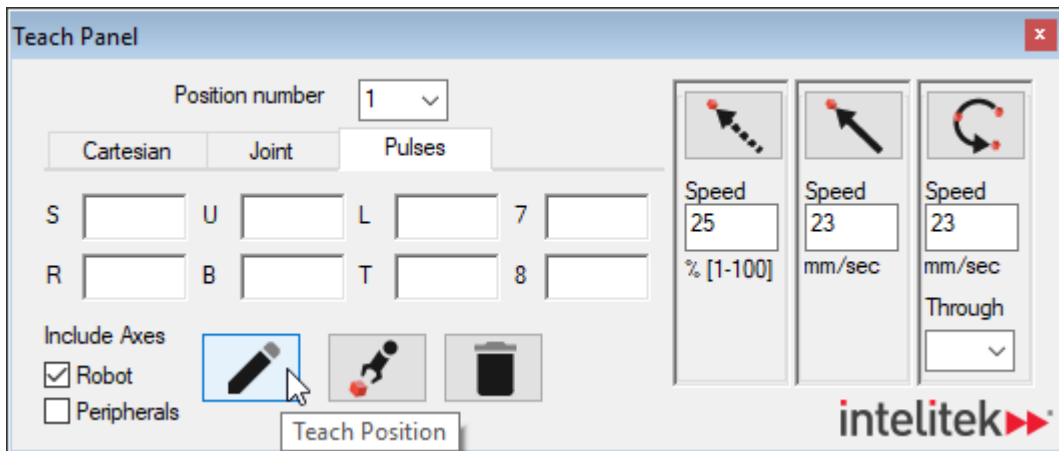
4. To record the current position, click the Record Position button.



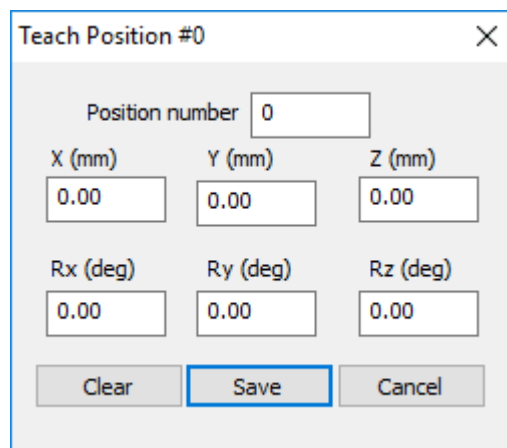
6.3.2. Teaching Positions

Teaching positions can be performed in RoboCell Level 2. To teach a position:

1. In the Teach Panel, click the Teach Position button.



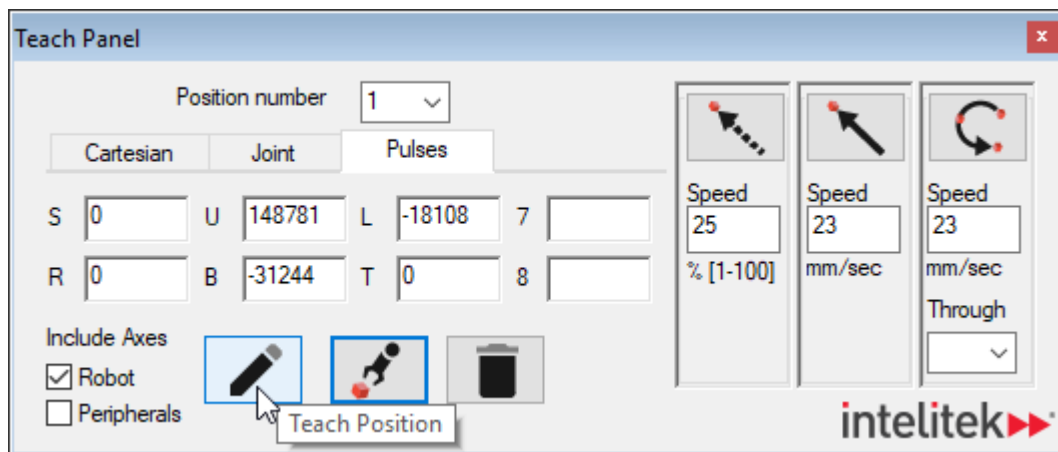
The Teach Position pop-up menu is displayed.



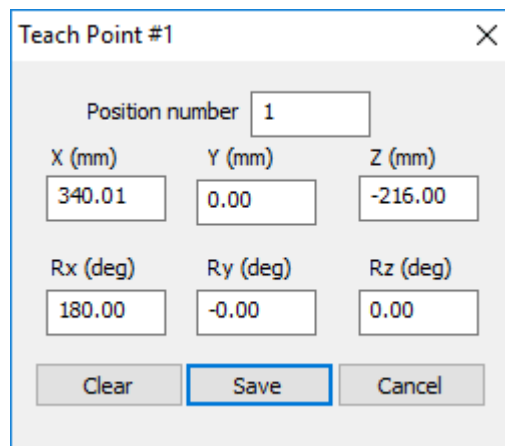
2. In the **Position number** field, enter a new position number.
3. Enter desired values into the **X (mm)**, **Y (mm)**, **Z (mm)**, **Rx (deg)**, **Ry (deg)**, and **Rz (deg)** fields.
4. Click **Save**. The position is taught.

To use existing position coordinates to define a new position (or modify that position's coordinates), perform this procedure:

1. In the **Position Number** field in the Teach Positions dialog box, select a position number.
2. In the Teach Panel's **Position number** drop-down menu, select an existing position number.
3. Click the Teach Position button.



The Teach Position pop-up menu is displayed.



4. Modify any or all coordinates as desired.
5. Click **Save**.

6.4. POSITION LIST

6.4.1. Overview

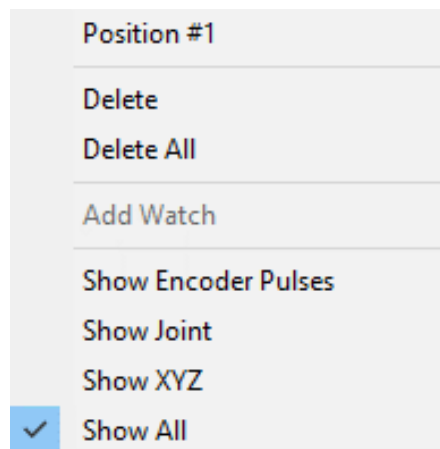
The positions window, shown below, enables you to view encoder counts, joint angles, and Cartesian coordinates values of the positions in the project. Each row in the top portion of the table represents a single position.

#	Coord.	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5	Axis 6	Axis 7	Axis 8	Type
		X (mm)	Y (mm)	Z (mm)	Rx (deg)	Ry (deg)	Rz (deg)	Pulses	Pulses	
1	Joint(deg)	0.00	0.00	0.00	0.00	0.00	0.00			Pulses
	XYZ	439.00	0.00	275.00	180.00	-90.00	0.00			
2	Encoder Pulses	38616	-10580	-533	0	-95263	977			
	Joint(deg)	28.28	3.38	-0.35	0.00	-93.03	-1.72			Pulses
	XYZ	223.15	120.07	119.79	180.00	0.00	30.00			
3	Encoder Pulses	-37394	60498	-19918	3302	-26574	0			
	Joint(deg)	-27.39	-19.34	-13.13	-2.42	-25.95	0.00			Pulses
	XYZ	360.30	-190.18	-29.35	177.60	-31.54	-24.84			
4	Encoder Pulses	-18934	89223	-6813	3302	-26574	0			
	Joint(deg)	-13.87	-28.52	-4.49	-2.42	-25.95	0.00			Pulses
	XYZ	431.18	-109.66	-50.18	177.63	-31.00	-11.33			

The **Type** column displays the method used to reach the position. There are three types, corresponding to the method that the position was taught/recorded using the teach panel. These types are Cartesian, Joint, and Pulses.

6.4.2. Position Popup Window

Right-click a position to open its popup menu. The popup menu is shown here.



The menu's options are summarized in this table:

Position Popup Menu Options	
Option	Description
Position No.	Displays the number of the position.
Delete	Deletes the selected position.
Delete All	Deletes all listed positions from the Position Inventory Panel and the SCORBASE memory.
Add Watch	Copies the position data to the watch area.
Show Joint	Shows only Joint values of all positions.
Show XYZ	Shows only the X, Y, Z, Rx, Ry, and Rz values of all positions.
Show Both	Shows both Joint and X, Y, Z, Rx, Ry, and Rz values of all positions.

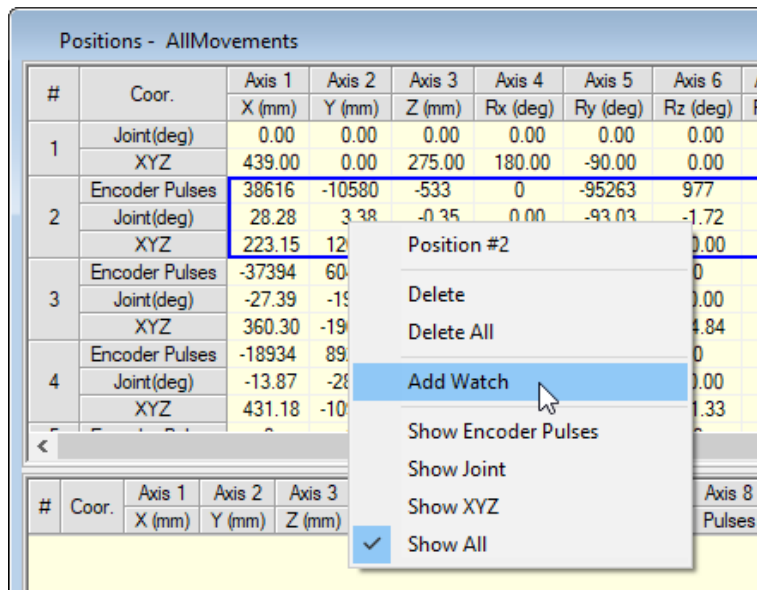
Note: The peripheral positions are always displayed using encoder counts.

6.4.3. The Watch Area

The section at the bottom of the Position List window is a watch area designed to help you compare a selected position to other positions in the list. The watch section is especially useful when there are many positions in the position list.

To add a position to the watch area:

1. Right-click the desired position.
2. Select **Add Watch**.



The position is added to the watch area.

#	Coord.	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5	Axis 6	Axis 7	Axis 8	Type
		X (mm)	Y (mm)	Z (mm)	Rx (deg)	Ry (deg)	Rz (deg)	Pulses	Pulses	
1	Joint(deg)	0.00	0.00	0.00	0.00	0.00	0.00			Pulses
	XYZ	439.00	0.00	275.00	180.00	-90.00	0.00			
2	Encoder Pulses	38616	-10580	-533	0	-95263	977			
	Joint(deg)	28.28	3.38	-0.35	0.00	-93.03	-1.72			Pulses
3	XYZ	223.15	120.07	119.79	180.00	0.00	30.00			
	Encoder Pulses	-37394	60498	-19918	3302	-26574	0			
4	Joint(deg)	-27.39	-19.34	-13.13	-2.42	-25.95	0.00			Pulses
	XYZ	360.30	-190.18	-29.35	177.60	-31.54	-24.84			
4	Encoder Pulses	-18934	89223	-6813	3302	-26574	0			
	Joint(deg)	-13.87	-28.52	-4.49	-2.42	-25.95	0.00			Pulses
4	XYZ	431.18	-109.66	-50.18	177.63	-31.00	-11.33			

6.5. DATA DASHBOARD

The Data Dashboard displays the current position of the robot and shows which inputs and outputs are activated. By default, the table presents position information in both the Joint Coordinate System and the Cartesian Coordinate System.

Data Dashboard																
Robot Actual Position																
Cartesian Coordinates	X (mm)		Y (mm)		Z (mm)		Rx (Deg.)		Ry (Deg.)		Rz (Deg.)					
	439.00		0.00		275.00		180.00		-90.00		0.00					
Joint Angles (Deg.)	Swivel Base		Lower Arm		Upper Arm		Arm Roll		Wrist Bend		Tool Flange					
	0.00		0.00		0.00		0.00		0.00		0.00					
Encoder Pulses																
S:	0	L:	0	U:	0	R:	0	B:	0	T:	0	7:	0	8:	0	
Digital I/O																
Inputs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Outputs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

The **Cartesian Coordinates** row displays the current Cartesian Coordinate System (XYZ) distance values of the tool center point (TCP) from the origin, in meters, as well as the rotational angle values (Rx, Ry, Rz) of the TCP, in degrees.

X (mm)	Y (mm)	Z (mm)	Rx (Deg.)	Ry (Deg.)	Rz (Deg.)
360.36	-190.10	-29.38	177.61	-31.55	-24.83

The **Joint Angles (Deg.)** row displays the angles between the two links of each joint, in degrees.

Swivel Base	Lower Arm	Upper Arm	Arm Roll	Wrist Bend	Tool Flange
-27.37	-19.33	-13.13	-2.42	-25.92	0.02

The **Encoder Pulses** row, shown below, displays the current values of the encoders for each of the robot's six axes and for two peripheral device axes.

Encoder Pulses															
S:	-37373	L:	60502	U:	-19933	R:	3316	B:	-26548	T:	-10	7:	0	8:	0

The encoder values change whenever the axes are moved.

The **Digital I/O** rows display which inputs and outputs (I/Os) are off or on. An I/O that is on is highlighted, while an I/O that is off is greyed.

Digital I/O																
Inputs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Outputs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

7. Controlling the Physical Robot

The following sections describe how to control the actual MHJF manipulator with RoboCell. To connect your computer and RoboCell project to the robot's controller, see section 2. Connecting RoboCell to the Physical Robot on page 7.

⚠ Warning: *Follow all safety precautions when operating the MHJF. For details on safety precautions, see the FS100 Operator's Manual.*

7.1. USING THE PROGRAMMING PENDANT

The programming pendant (PP) is a hand-held terminal that gives the operator direct control over the robot. In addition to controlling movement of the axes, the PP may be used for recording positions, sending the axes to recorded positions, writing programs (jobs), running programs and other functions.

To control the axes using the PP and maintain a connection between the actual and simulated robot, RoboCell must be operating in On-Line mode, and the MODE key on the PP must be switched to **Teach**. This disables control of the axes from RoboCell windows.

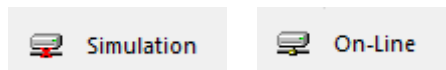
Most PP operations are reflected in the RoboCell dialog boxes. For example, encoder and XYZ values change in the Data Dashboard when the robot is jogged.

Programming Pendant operation is described fully in the FS100 Operator's Manual.

7.2. ON-LINE MODE / OFF-LINE MODE

RoboCell can run either in On-Line or in Simulation mode. In On-Line mode, RoboCell communicates with the controller over wired connection. If the Servo On state is selected, RoboCell controls the robot, peripherals, and I/O box. To control the axes of the actual robot using RoboCell, RoboCell must be operating in On-Line mode, and the MODE key on the PP must be switched to **Remote**.

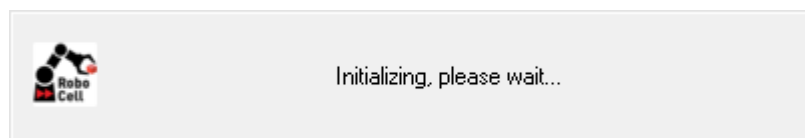
In Simulation mode, only the virtual robot can be controlled. The active mode is displayed in the status bar.



To change to On-Line mode, select **Home | On-Line**.

To change to Simulation mode, select **Home | Simulation**.

If RoboCell is opened in On-Line mode, or On-Line is selected from the command ribbon, this message is displayed as RoboCell searches for the controller:



If the controller is detected, On-Line mode is activated.

If the controller is not detected, Simulation mode remains the active mode.

7.3. SERVO ON / SERVO OFF

The Servo On state enables servo control of the axes. This state is available only in On-Line mode. In the Servo Off state, axis movement commands cannot be executed. The control state (Servo On or Servo Off) is displayed in the lower right corner of the status bar.



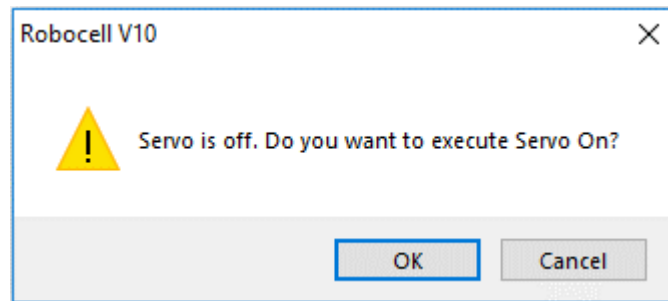
To enable control of the axes, perform one of the following:

- Select **Home | Servo On**.
- Press **F5**.

To disable control, perform one of the following:

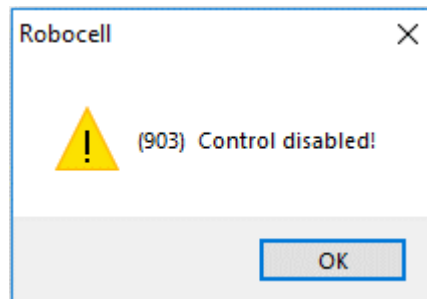
- Select **Home | Servo Off**.
- Press **F2**.

If you have disabled control and RoboCell opens in On-Line mode, or On-Line mode is selected after detection of the controller, this prompt is displayed:



Press **OK** to enable control, or press **Cancel** to remain in the Servo Off state.

The controller automatically disables control if an impact condition, trajectory error, or thermic overload error occurs during execution of a movement command. If you attempt to move the axes when control is disabled, this error message is displayed:

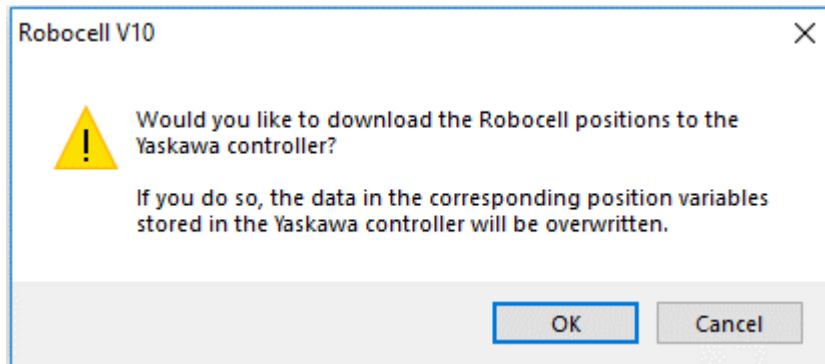


When RoboCell is in Simulation mode, the Control state cannot be altered.

7.4. DOWNLOADING POSITIONS

In On-line mode, positions that you teach and record in RoboCell are transferred to the Yaskawa controller. The actual robot passes through these positions when running a motion program (job).

When running programs and teaching, recording, or editing positions in On-line mode, or when switching from Simulation mode to On-line mode, RoboCell prompts you with the following confirmation message:



- Selecting **OK** overwrites the relevant positions on the controller.
- Selecting **Cancel** leaves the controller positions as is.

⚠ Warning: *Even if you select **Cancel** and then run the program in On-line mode, the robot passes through the program's positions as they are defined on the controller. These previously-defined positions may be different than those in the active RoboCell program, and therefore running this program could result in a potentially dangerous situation and/or collision.*

8. Program Editing

A SCORBASE program is a set of instructions written by the user to control the robot, peripheral equipment, and to communicate with external I/O devices. This chapter explains how to create and edit a program.

The following tools are used for program editing:

- The Program Window
- The SCORBASE Commands Library

8.1. OPENING AND CLOSING A PROGRAM

Every program is part of a RoboCell project. A project also includes the user-defined positions, project data and, if imported, a virtual cell (.3dc file). Only one project can be opened at a time.

To open a saved program, open the project containing the desired program by performing one of the following:

- Select **File | Open Project**.
- Press **Ctrl + O**.

The *Load Project* window opens, prompting you to select the project that contains the program you want to edit.

The program is displayed in the Program Window.

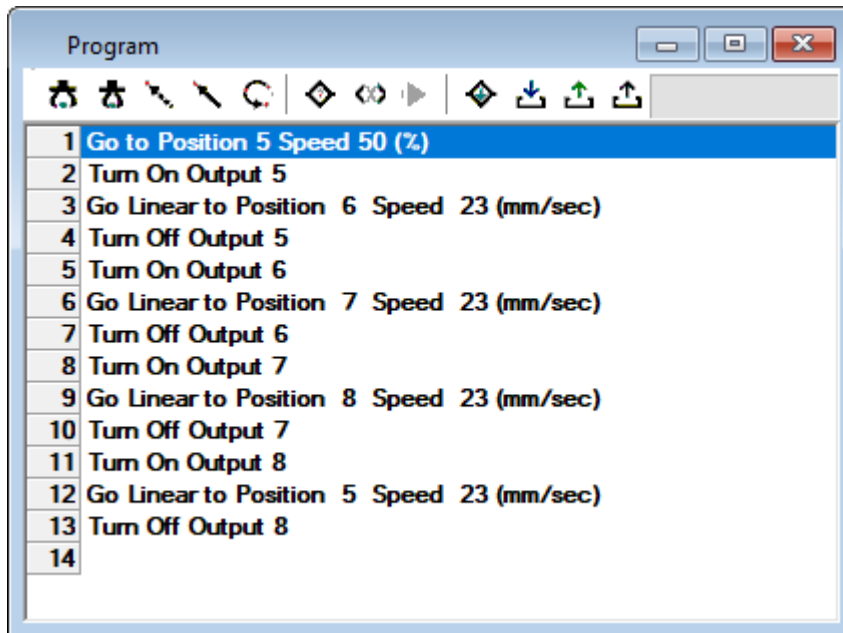
To create a new project, perform one of the following:

- Select **File | New Project**.
- Press **Ctrl + N**.

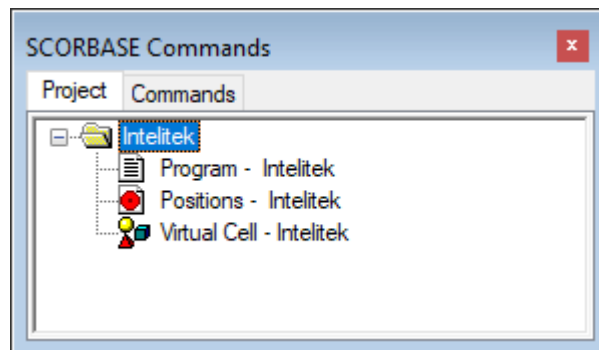
By default, RoboCell projects are opened with the Compose Screen layout. This layout displays five or six windows:

- SCORBASE Commands Library
- Program window
- Jog Control window
- Teach Panel
- Data Dashboard
- Virtual Cell (if already imported. To add a virtual cell, select **Virtual Cell | Import** and select an existing .3DC file.)

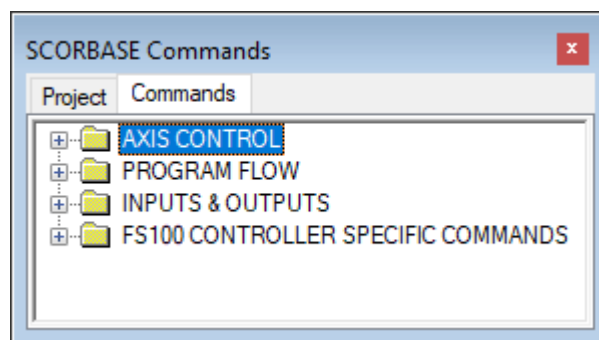
The Program Window, shown below, contains the text of the program currently loaded. Its title bar displays the name of the project. Also shown by default is the programming toolbar. The toolbar contains a drop-down list of all subroutines available, for quick navigation.



The SCORBASE Commands Library contains two tabs to switch between the project files and the commands. The **Project** tab is shown here:



The **Commands** tab displays the Command Tree, which contains all of the available commands.



8.2. PROGRAM EDITING TOOLS

SCORBASE is a text-based programming language in which every command is a single text line. SCORBASE programs are edited using standard Windows text editing options, which can be accessed:

- Via the Edit functions of the Home tab.
- By pressing the designated keyboard shortcut.
- By right-clicking the mouse in the Program Window to open a pop-up menu.

Program Editing Tools		
Icon	Option - Shortcut	Description
	Cut - Ctrl+X	Deletes selected text or lines from the program, and places it on the Windows and SCORBASE clipboards.
	Copy - Ctrl+C	Places a copy of selected text or lines from the program on the Windows and SCORBASE clipboard.
	Paste - Ctrl+V	Inserts the contents of the SCORBASE clipboard into the program.
	Find - Ctrl+F	Opens a dialog box (shown here) that allows you to search for any string, such as a command or text. <div data-bbox="609 926 1242 1140" style="border: 1px solid gray; padding: 5px; margin: 5px 0;"> <p style="text-align: right;">Find in PROGRAM X</p> <p>Find what: <input type="text"/></p> <p> <input type="checkbox"/> Match case Direction <input type="radio"/> Up <input checked="" type="radio"/> Down <input type="button" value="Find Next"/> <input type="button" value="Cancel"/> </p> </div>
	Find Next - (F3)	Repeats the last Find operation for the next occurrence. (Accessible also from the Find dialog window).
	Comment	Inserts/deletes an asterisk at beginning of a command line. SCORBASE ignores command lines that start with an asterisk. This feature is useful for debugging.
	Go to Line	Opens a dialog box (shown below) that displays the total number of lines in the program and prompts you for a line number. Type in the number. The program editor jumps to the line you specify. <div data-bbox="597 1472 1252 1770" style="border: 1px solid gray; padding: 5px; margin: 5px 0;"> <p style="text-align: right;">Go To Line ... X</p> <p style="text-align: center;">Total number of lines: 6.</p> <p style="text-align: center;"><input type="text"/></p> <p style="text-align: center;"> <input type="button" value="OK"/> <input type="button" value="Cancel"/> </p> </div>

In addition, use the keyboard for the following functions:

Additional Program Editing Functions	
Keyboard Shortcut	Function
[Ins]	Toggles between Insert Mode and Overwrite Mode. The currently active mode is shown in the Status bar at the very bottom of the RoboCell window. In Insert Mode, a new command is inserted into the program above the line currently marked by the cursor.
[Del]	Deletes the line or lines currently marked by the cursor.
[Ctrl+Home]	Brings the cursor to the first line of the program.
[Ctrl+End]	Brings the cursor to the last line of the program.
[PgUp]	Displays the previous page of program lines.
[PgDn]	Displays the next page of program lines.

8.3. ADDING AND EDITING COMMANDS

SCORBASE commands are organized in a Command Tree, which is displayed in the Commands Library window. To access the Command Tree, click the Command tab in the window.

SCORBASE commands are grouped into these categories:

- Axis Control
- Program Flow
- Inputs & Outputs
- FS100 Controller Specific Commands

Clicking on a category opens/closes the list of commands.

To add commands to a program, perform one of the following:

- Double-click the desired command in the Command Tree.
- Type the two letters written next to the command.
- Click the command icons in the Program Window (applicable only for selected commands).

If you are working in Insert Mode, the new command line is added above the currently selected (highlighted) line. If you are working in Overwrite Mode, the new command replaces the selected line. Toggle between the two modes using the [Ins] key. Many commands open dialog boxes for completing the command line parameters. Your current mode is shown in the status bar at the bottom of the RoboCell screen.



To change a command parameter, click the command to re-open the command dialog box. Change the required parameter and then click OK to close the dialog box.

8.4. AXIS CONTROL COMMANDS

The Axis Control options, commands, and functions are described in the tables below.

Basic Axis Control Commands (Available in Level 1)		
Icon	Command	Description
	OG Open Gripper	Fully opens the gripper
	CG Close Gripper	Fully closes the gripper (on itself, or on a grasped object)
	GP Go to Position	<p>Opens the Go to Position dialog box, shown here. The Go to Position (Joint) command sends the robot to a recorded position, in the shortest time, using Point to Point (P to P) control. In Point to Point control, all axes move independently and there is no control over the tool center point (TCP) trajectory.</p> <div data-bbox="657 777 1201 1396" style="border: 1px solid black; padding: 5px;"> <p>Go to Position ✕</p> <p>Target Position: <input type="text" value="7"/> ▼</p> <p> <input checked="" type="radio"/> Joint <input type="radio"/> Linear <input type="radio"/> Circular </p> <p> <input type="radio"/> Fast (100%) <input checked="" type="radio"/> Speed (1...99%) <input type="text" value="50"/> <input type="radio"/> Speed (mm/sec) </p> <p style="text-align: center;"> <input type="button" value="OK"/> <input type="button" value="Cancel"/> </p> </div>

Movement Control	
Option	Description
Target Position	The destination of the movement. Enter a number or a variable in this field.

Basic Speed/Duration Options	
Option	Description
Fast	Executes the movement at the fastest speed possible

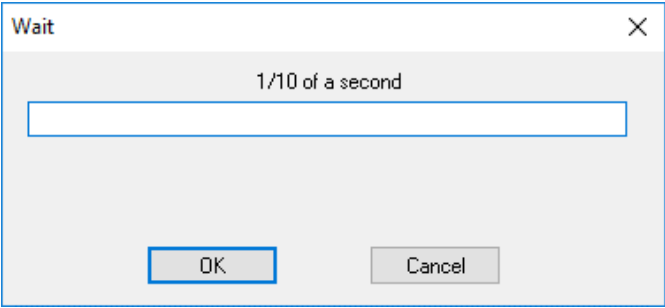
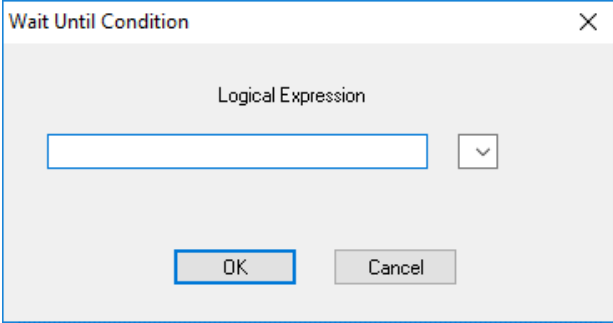

Basic Speed/Duration Options	
Option	Description
Speed	Executes the movement at a slower speed. Enter a number from 1 through 99(%), or a variable, in the Speed field. Default: 25.

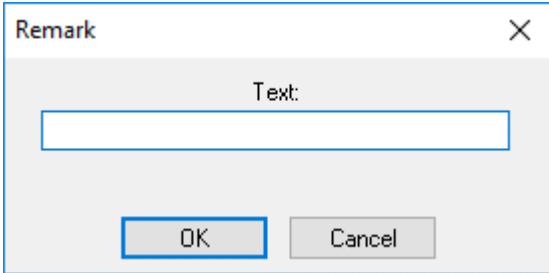

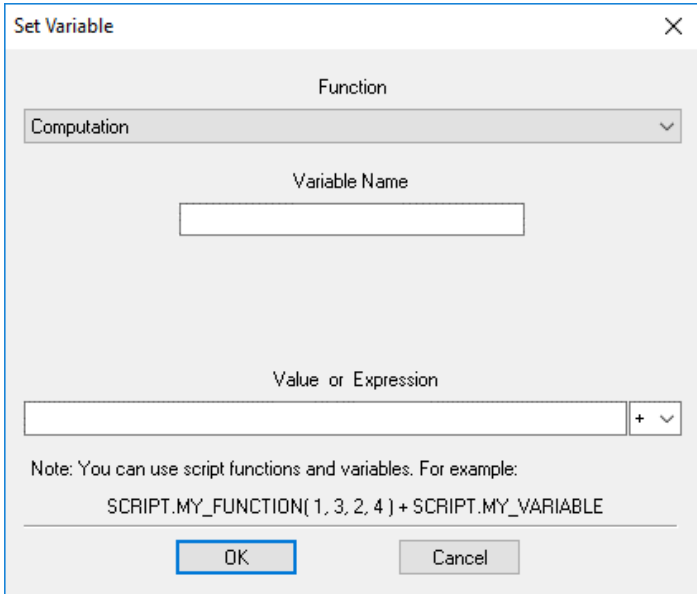
Additional Movement Control Options		
Icon	Option	Description
	GL Go Linear to Position #_Speed ...	<p>Sends the robot's TCP (tool center point) from its current position to the target position, along a linear path (straight line). The linear motion applies only to the robot axes. For linear motion, speed can also be defined in mm/s.</p>

Additional Movement Control Options		
Icon	Option	Description
	GC Go Circular to Position #_ Speed...	<p>Sends the robot's TCP in a circular path to the target position, via the position specified in the Via Position field. The circular motion applies only to the robot.</p> <p>Note: The Go Circular command follows the circle defined by the three positions (current TCP position, target position, and the intermediate position specified in the Via position field).</p> <div data-bbox="672 495 1216 1117" style="border: 1px solid #ccc; padding: 10px; margin: 10px auto; width: fit-content;"> <p>Go to Position ✕</p> <p>Target Position: <input type="text" value="4"/> <input type="radio"/> Joint</p> <p>Via Position: <input type="text" value="5"/> <input type="radio"/> Linear</p> <p style="text-align: right;"><input checked="" type="radio"/> Circular</p> <hr/> <p><input checked="" type="radio"/> Speed (mm/sec) <input type="text" value="23"/></p> <p style="text-align: center;"><input type="button" value="OK"/> <input type="button" value="Cancel"/></p> </div>
N/A	Via Position	The position via which the destination of the movement is reached. Enter a number or a variable in this field.
N/A	SO ServoOn	Enables axis control by the programming pendant.
N/A	SF ServoOff	Disables axis control by the programming pendant.

8.5. PROGRAM FLOW COMMANDS

The Program Flow commands are summarized in the tables below.

Program Flow Commands		
Icon	Command	Description
N/A	WT Wait (10ths of second)	<p>Halts program execution for a time specified in tenths of a second. The command opens this dialog box:</p> 
N/A	WC Wait Until Condition	<p>Halts program execution until the defined condition is fulfilled. The command opens this dialog box:</p> 
N/A	JU Jump to	<p>This unconditional jump command causes the program pointer to jump to the line that contains the specified Label. The command opens this dialog box:</p>  <p>When the Jump command is used, Jump is selected in the dialog box. You can modify the Jump command to an IF jump command (conditional jumping).</p>

Program Flow Commands		
Icon	Command	Description
N/A	RE Remark	<p>Allows insertion of a comment line for explanation and documentation into the program. The command opens this dialog box:</p> 
N/A	RB Ring Bell	When executed, this command produces a beep.
N/A	RT Reset Timer	<p>RoboCell uses a timer that measures time in units of tenths of a second. The timer starts operating when RoboCell is opened.</p> <p>The Reset Timer command resets the value of the timer to 0.</p> <p>To use the timer, the timer value must be assigned to a variable, using the Set Variable command.</p>
	SV Set Variable to Computation	<p>Allows you to assign a value, or an expression (result of a specific computation), to a variable. The command opens the Set Variable dialog box (Computation is selected by default).</p> 

Program Flow Commands		
Icon	Command	Description
	SV Set Variable to Computation (Continued)	<p>Variable Name</p> <p>The first character of the variable name must be a letter.</p> <p>Value or Expression</p> <p>Enter a value or a formula.</p> <p>To set the variable to a fixed value, enter a number (in the range of +/-1000000).</p> <p>To set the variable to the result of a computation, enter a string that consists of two arguments and an operator. An argument can be either an integer or a variable (e.g., fun*2).</p> <p>Click the arrow to see a list of operators, or use the list of operators, which is shown below.</p>

The Operators drop-down List is shown here:



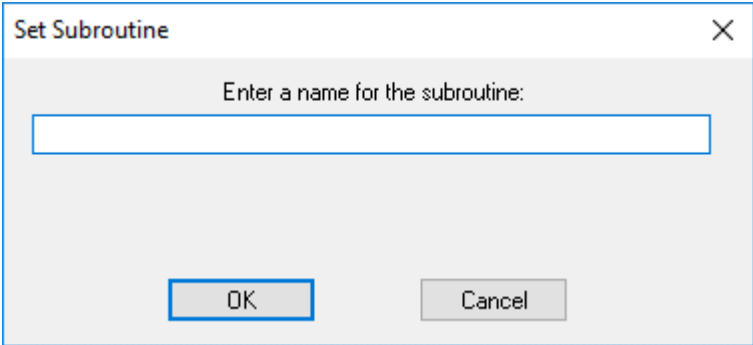
The operators in the list are summarized in this table.

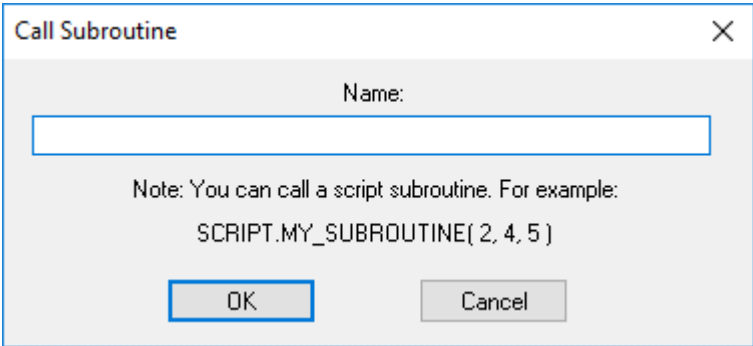

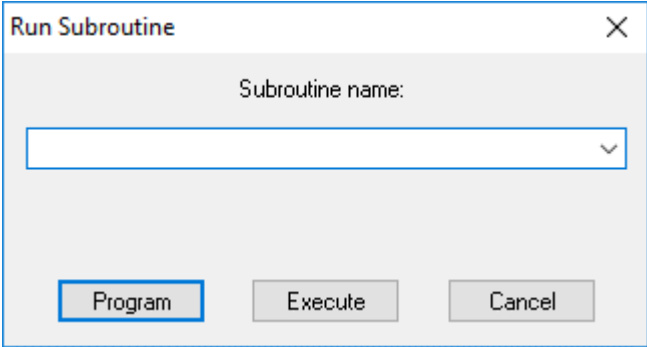
Operators	
Arithmetic Operators	
+	Addition
*	Multiplication
-	Subtraction
/	Division
\	Floating Point Division
Algebraic Operators	
^	Power (raises the first argument to the power of the second argument).
MOD	Modulus (returns the remainder of the first argument divided by the second).

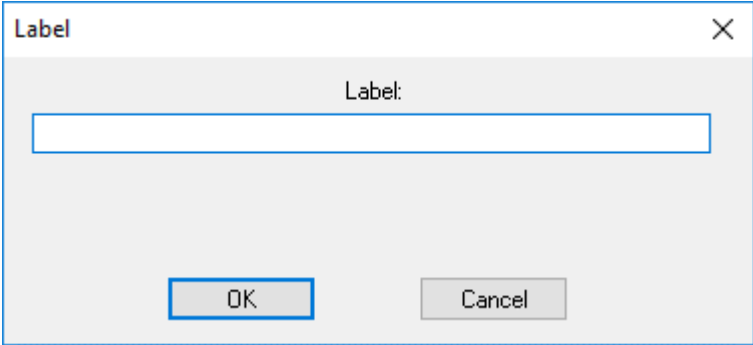
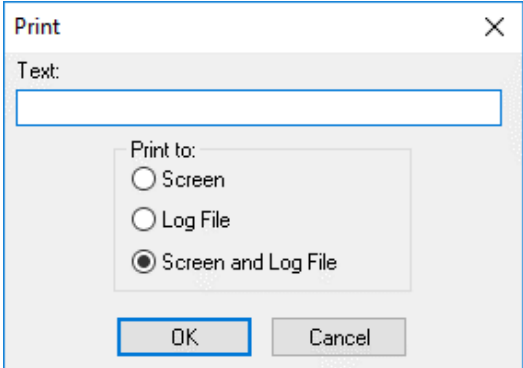
Logical (Boolean) Operators	
NOT	Not
AND	And
OR	Or
XOR	Exclusive or
EQV	Equivalent (Null or Boolean values only)
IMP	Implication (Null or Boolean values only)

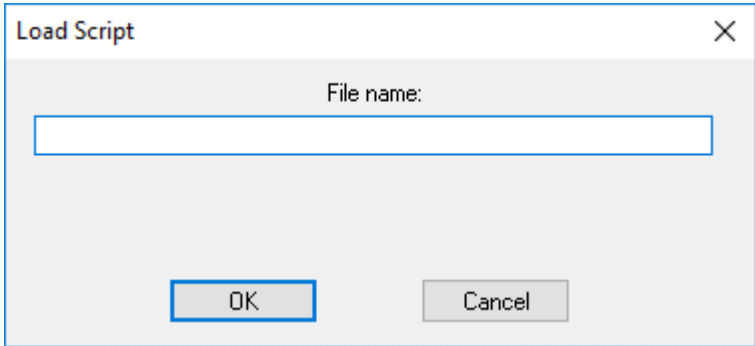
The result of a logical operation is 1 (True), or 0 (False). Any operand with a non-zero value is considered true, while a zero value is considered false.

Set Variable Dialog Box		
Icon	Option	Description
	SV Set Variable to Computation (Continued)	<p>A value assigned using the SV command can be used for conditional jumping using the IF <Condition> Jump command. Actions can then be generated according to the variable value. The following example jumps the program cursor to a label if more than three seconds elapse after timer reset:</p> <pre>Reset Timer ... Set Variable TIME to timer IF TIME > 30 jump to PICKUP</pre> <p>For more information on variables, See section 9 Variable Programming, on page 106.</p>
N/A	VT Set Variable to Timer	<p>Allows you to assign the current value of the RoboCell timer to a variable.</p> <p>The timer starts operating when RoboCell is opened.</p> <div data-bbox="620 730 1338 1341" data-label="Image"> </div> <p>To initialize the timer in a program, use the Reset Timer (RT) command.</p>
	IF If Jump to	<p>A conditional branch command, which is used to determine the program flow in relation to the value of the variables. The command opens this dialog box:</p> <div data-bbox="711 1486 1247 1803" data-label="Image"> </div>

Set Variable Dialog Box		
Icon	Option	Description
		<p>If the condition in the IF field is true, program execution jumps to the line specified by the label in the Jump to field.</p> <p>If the condition in the IF field is false, program execution skips to the following line.</p> <p>IF</p> <p>Enter the condition. The condition includes a variable name, a comparison operator and another variable name, or a number.</p> <p>Jump to</p> <p>Enter the name of a Label. (Be sure to include a line with this Label in your program.)</p> <p>Example:</p> <pre>If COUNTER > 0 jump to START_LOOP Go to Position 1 speed 5 ... START_LOOP: Go to Position 2 speed 5</pre> <p>If the value of Counter is greater than zero, the robot goes to Position #1.</p> <p>If the value of Counter is equal to or less than zero, the robot goes to Position #2.</p> <p>Use two equal signs (=) for equal operators. For example:</p> <pre>If COUNTER == 0 jump to END</pre> <p>The IF jump command can be converted to an unconditional jump command by selecting Jump (instead of If).</p>
N/A	SS Set Subroutine	<p>Creates a subroutine. You can program up to 64 subroutines in one program. The command opens this dialog box:</p>  <p>Note: Create subroutines only at the end of the main program. Every subroutine must end with a Return from Subroutine command.</p>
N/A	RS Return from Subroutine	<p>Marks the end of a subroutine. At run time, this command terminates the execution of the subroutine, and the program resumes execution at the line that follows the Call Subroutine command.</p> <p>Every subroutine must end with a Return from Subroutine command.</p>


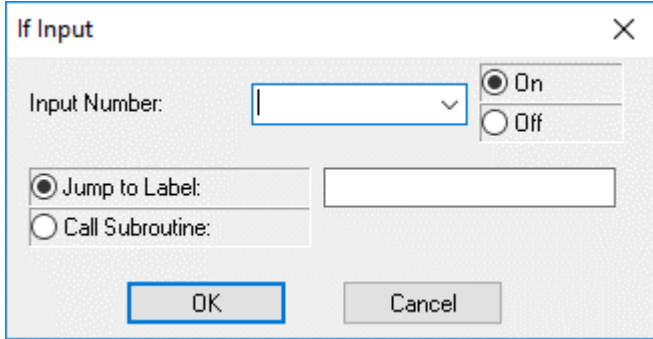
Set Variable Dialog Box		
Icon	Option	Description
N/A	CS Call Subroutine	<p>Activates the specified subroutine. The command opens this dialog box:</p>  <p>Notes:</p> <ul style="list-style-type: none"> • Use the Call Subroutine command either from the main program or from another subroutine command. • You can call the same subroutine repeatedly in the same program. • You can call a subroutine from a Visual Basic script that is loaded. Append the prefix "SCRIPT." to the beginning of the Visual Basic subroutine name. If more than one VB script is loaded, it is the responsibility of the programmer to ensure that there is no conflict of names. • After the subroutine is executed, the program resumes execution from the line that follows the Call Subroutine command.
	RU Run Subroutine	<p>Activates the selected subroutine. The command opens this dialog box:</p>  <p>Either write a subroutine name, or select one from the drop-down menu. To include the subroutine in the program, press the Program button. To run the subroutine, press the Execute button.</p> <p>When the command is initiated, execution of the main program continues, and the subroutine runs concurrently. RoboCell facilitates the operation of up to 100 concurrent programs. (The Call subroutine command suspends the main program until the completion of the subroutine.)</p>


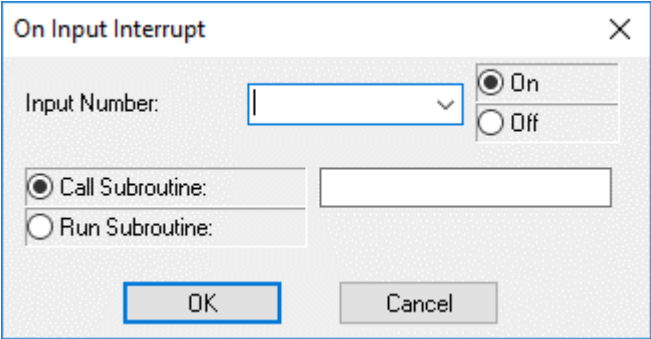
Set Variable Dialog Box		
Icon	Option	Description
N/A	LA Label	<p>Marks a line in the program that is referenced by a Jump command.</p> <p>The command opens this dialog box:</p>  <p>In the Label field, enter a name.</p> <p>Do not include blank spaces – use an underscore.</p> <p>Do not use the same label name more than once.</p>
N/A	PS Print to Screen & Log	<p>Prints data containing strings, messages and variable values to a log file, or to the message window, or to both. The command opens this dialog box:</p>  <p>To print a value of a variable, place the variable name in single quote marks. For example: VARX='X' prints as VARX=50 (when the value of X is 50).</p> <p>To print a value of a Visual Basic Script variable, include "SCRIPT." followed by the variable name in single quote marks. For example: VARY='SCRIPT.Y' prints as VARY=100 (when the value of SCRIPT.Y is 100). Select the desired print destination.</p>

Set Variable Dialog Box		
Icon	Option	Description
N/A	LS Load Script	<p>Loads a Visual Basic script by means of the following dialog box:</p>  <p>The script must be located in the same subdirectory as the RoboCell project file. You must be sure that the LS command is executed before you attempt to call any of the affected subroutines or variables.</p> <p>You may load more than one script into a single RoboCell project. However, it is the programmer's responsibility to ensure that there are no name conflicts.</p>
N/A	EN End Program	<p>This command enters the command End on the selected line of the program. It signifies the end of the program. It may be used in more than one subroutine, such as subroutines which respond to various error conditions.</p>

8.6. INPUT/OUTPUT COMMANDS

The input/output options, commands, and functions are described in the following tables.

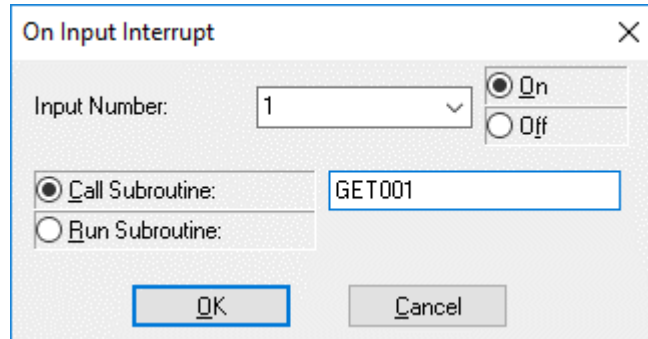
Input and Output Command		
Icon	Option	Description
	II If Input # On/Off Jump	<p>Causes the program to jump to a label or call a subroutine, if the state of the tested digital input matches the status specified (On or Off). The command opens this dialog box:</p>  <p>In the Input Number field, enter the number of digital input (1-8), or a variable.</p>

	OI On Input Interrupt # On/Off	<p>Sets the condition for an input interrupt service. The service (Call Subroutine or Run Subroutine) is performed whenever the condition (input status) is satisfied, regardless of the current program pointer position. The command opens this dialog box:</p> <div data-bbox="656 352 1302 688" style="border: 1px solid #ccc; padding: 10px; margin: 10px auto; width: fit-content;">  <p>The dialog box 'On Input Interrupt' contains the following fields and controls:</p> <ul style="list-style-type: none"> Input Number: A dropdown menu. On/Off: Radio buttons for 'On' (selected) and 'Off'. Call Subroutine: Radio button and an adjacent text input field. Run Subroutine: Radio button and an adjacent text input field. Buttons: 'OK' and 'Cancel' at the bottom. </div> <p>Input Number Enter the number of a digital input, a variable or the word ANY. Use of the word ANY causes any input (1-8) to evoke the interrupt state.</p> <p>On/Off Select the state of the input.</p> <p>Call Subroutine Enter the name of a subroutine that is executed in case of interrupt. Execution of the main program is suspended until completion of the subroutine.</p> <p>Run Subroutine Enter the name of a subroutine that is executed in case of interrupt. Execution of the main program continues, and the subroutine is spawned as a concurrent process.</p>
---	--------------------------------------	--

An interrupt command causes the program to halt the command it is currently executing (which can also be a movement or a delay), and to immediately execute the command specified for this interrupt. If the specified command is a Call Subroutine, the program resumes from the point where it was suspended, as soon as the subroutine completes its execution. If the specified command is a Run Subroutine, the main program resumes immediately after the concurrent process is started.

An interrupt command can be disabled and enabled by means of the EI (Enable Interrupt) and DI (Disable Interrupt) commands described below.

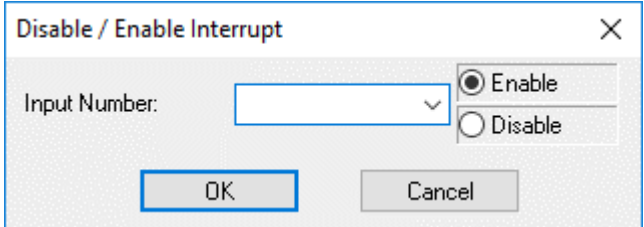
For Example:

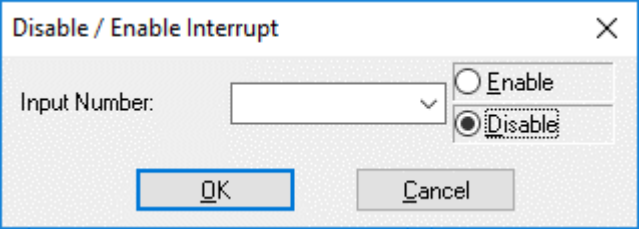

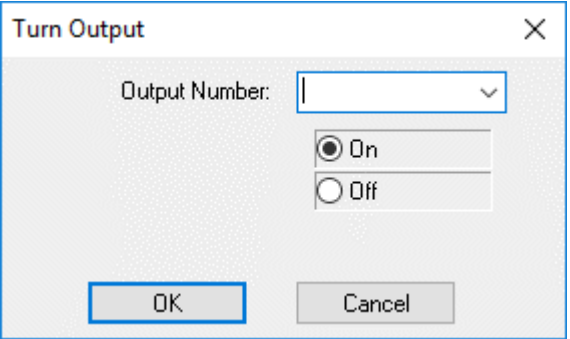


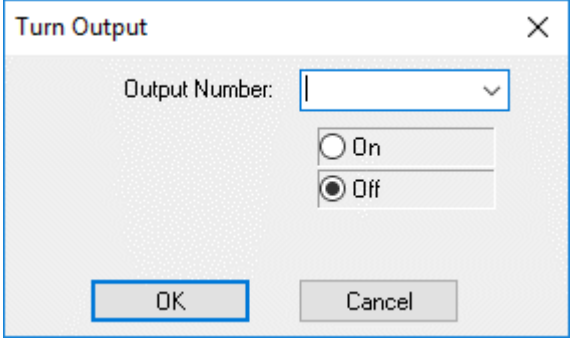
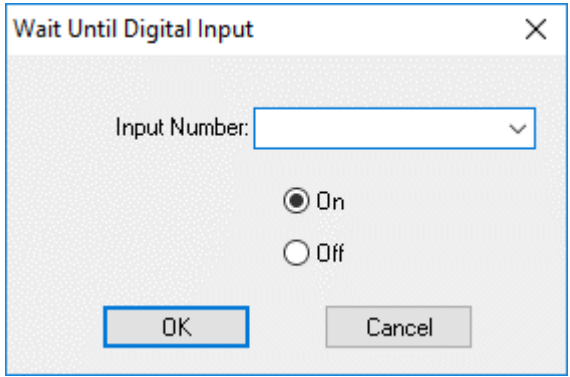
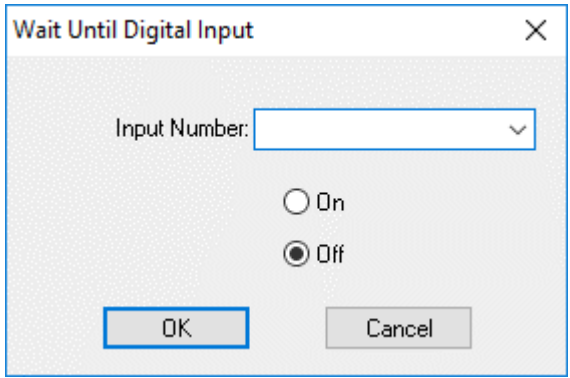
This inserts the following line of code:

```
On input interrupt 1 on call subroutine GET001
```

When Input 1 is turned on, the program immediately calls subroutine GET001. If any axes are moving when the interrupt occurs, they immediately stop. When the subroutine is completed (Return from Subroutine command is reached), the axes reassume the position and status that were interrupted, and the program continues from that point.

Interrupt Commands		
Icon	Option	Description
N/A	EI Enable Interrupt #	<p>Causes the specified input interrupt to become active. The command opens this dialog box:</p>  <p>When an interrupt is active, it is waiting for an interrupt status (defined using the OI [On Interrupt] command described above).</p> <p>Input Number Enter an input number, a variable, or the word ALL.</p> <p>Enable/Disable Select Disable to modify the command.</p>

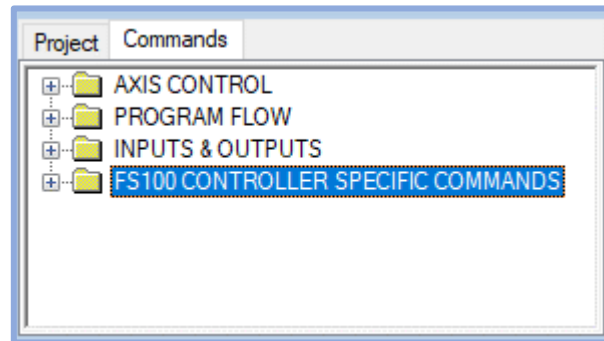
Interrupt Commands		
Icon	Option	Description
N/A	DI Disable Interrupt #	<p>Causes the specified input interrupt to become inactive. When an interrupt is inactive, it is disregarded until the EI (Enable Interrupt) command reactivates it. The command opens this dialog box:</p>  <p>Input Number Enter an input number, a variable, or the word ALL.</p> <p>Enable/Disable Select Enable to modify the command.</p>
	ON Turn On Output #	<p>Sets the state of the specified digital output On. The command opens this dialog box:</p>  <p>Output Number Select a number or type a variable name. The default settings of Digital and On can be changed.</p>

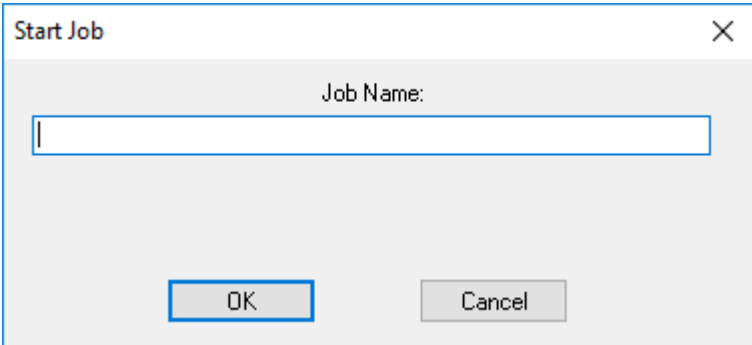
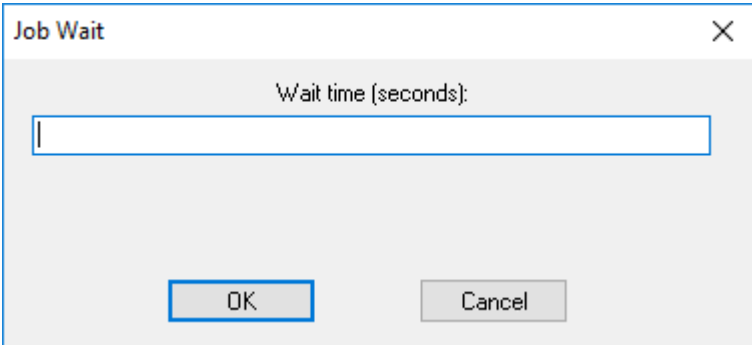
Interrupt Commands		
Icon	Option	Description
	OF Turn Off Output #	<p>Sets the state of the specified digital output Off. The command opens this dialog box:</p>  <p>Output Number Select a number or type a variable name. The default settings of Digital and Off can be changed.</p>
N/A	WN Wait Until Digital Input # On	<p>Halts program execution until the selected input is switched on. The command opens this dialog box:</p> 
N/A	WN Wait Until Digital Input # Off	<p>Halts program execution until the selected input is switched off. The command opens this dialog box:</p> 

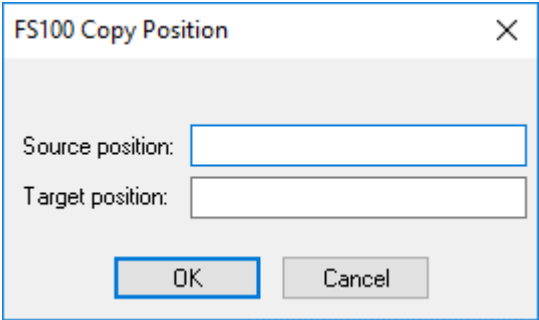
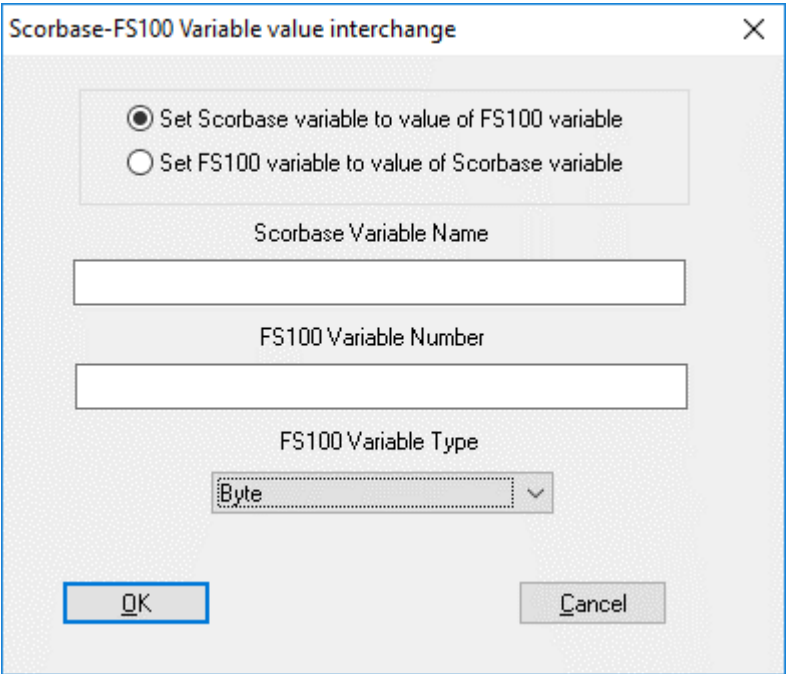
8.7. FS100 CONTROLLER-SPECIFIC COMMANDS

The FS100 controller-specific commands are the final branch of the SCORBASE Commands Library tree. The commands are used to execute instructions that are stored or saved on the MHJF's FS100 controller memory.

The command submenu is shown here. Each command is detailed in the table on the next several pages.



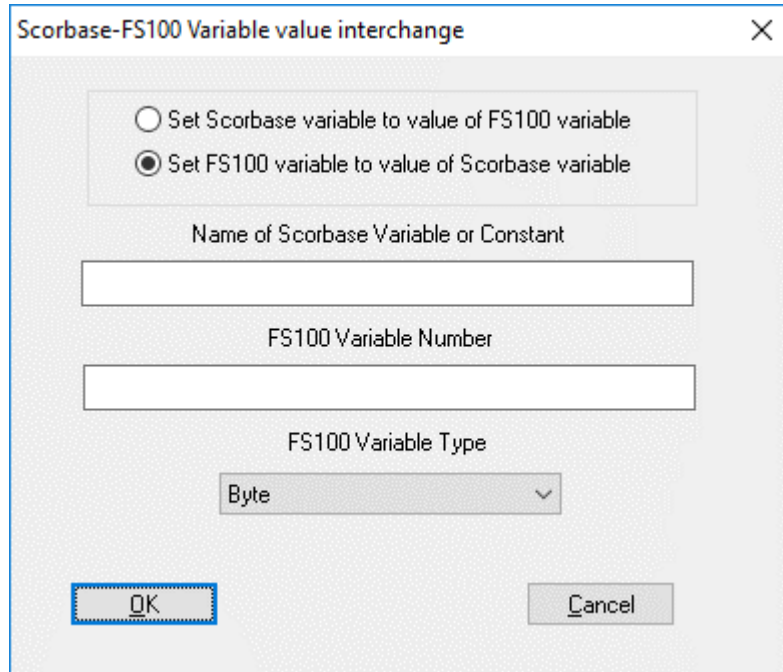
Advanced Commands	
Command	Description
AJ FS100StartJob	<p>Starts a job (FS100/MHJF motion program). Enter the name of the job and click OK.</p> 
JW FS100JobWait	<p>Pauses a FS100 job for a designated amount of time. Enter the wait time in seconds, and then click OK.</p> 

<p>CP FS100CopyPosition</p>	<p>Copies position data from a position variable on the FS100 to a recorded position in a RoboCell project. To use this command:</p> <ol style="list-style-type: none"> 1. In the Source position field, enter the number of the FS100 position variable. 2. In the Target position field, enter the number or variable name of position in your RoboCell project. 3. Click OK. 
<p>EV SetScorbaseVariableTo FS100Variable</p>	<p>Sets the value of a variable in your RoboCell project to the value of a variable saved on the FS100 controller. To use this command:</p> <ol style="list-style-type: none"> 1. In the Scorbase Variable Name field, enter the desired variable from your Robocell project. 2. In the FS100 Variable Number field, enter the number of the desired variable on the FS100 controller. 3. In the FS100 Variable Type drop-down menu, select the relevant FS100 variable type. 4. Click OK. 

EX SetFS100VariableToScorbaseVariable

Sets the value of a variable saved on the FS100 controller to the value of a variable in your RoboCell project. To use this command:

1. In the **FS100 Variable Number field**, enter the number of the desired variable on the FS100 controller.
2. In the **Scorbase Variable Name field**, enter the desired variable from your Robocell project.
3. In the **FS100 Variable Type** drop-down menu, select the relevant FS100 variable type.
4. Click **OK**.



9. Variable Programming

The SCORBASE language allows variable programming. Variables allow you to write commands that change as the state of the robot or its environment changes during program execution. Therefore, they are useful for creating loops and subroutines in robot programs.

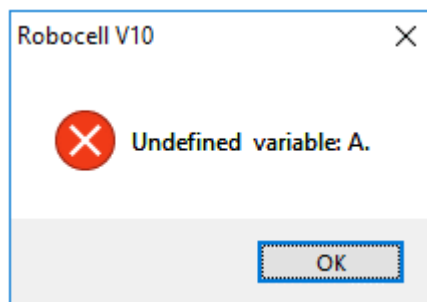
To use a variable, it must first be defined using the **Set Variable to Computation** command. Variable names can be up to 22 characters long. It is recommended, however, that you use meaningful names that are as short as possible. The first character of the name must be alphabetic.

SCORBASE has full access to variables in Visual Basic scripts that are loaded in memory. The variable name must begin with **SCRIPT.**, for example: **SCRIPT.A**. Values can be transferred both to and from variables in external Visual Basic scripts.

In most editing commands, a variable can be specified instead of a numeric value.

A variable cannot be used to specify a Label or a Subroutine.

If at run time the program encounters a variable whose value is not defined or is out of range, an error message is displayed.



9.1. USING A VARIABLE INSTEAD OF A NUMERIC VALUE:

Example #1

```
Set Variable station1=5
Set Variable lamp=1
Go to Position station1 fast
Turn on output lamp
```

In this example, one of the robot stations named “*station1*” is recorded as position #5 and a lamp is connected to output #1. When the program initiates, the value (5) is assigned to the variable named *station1* and the value (1) is assigned to the variable *lamp*. The following program lines send the robot to position “*station1*” and turn on a “*lamp*”. Using a meaningful name for the variable makes the programming, debugging and maintenance easier.

Example #2

```
Set variable pos = 0
Start:
Set variable pos = pos +1
Go to position pos fast
Wait 50 (10ths of seconds)
If pos < 5 jump to start
```

In this example, the robot moves to Positions #1, #2, #3, #4, and #5. At each position, the robot waits 50 seconds. After waiting at position #5, the program terminates.

Example #3

```
Start:
Set variable tested = 0
Loop:
Set variable tested = tested +1
If input tested off call sub off
If input tested on call sub on
If tested < 8 jump to loop
Jump to start

Set Subroutine off
Turn off output tested
Return from subroutine.

Set Subroutine on
Turn on output tested
Return from subroutine
```

In this example, the program sequentially scans digital inputs 1 through 8 in an endless loop.

If the tested input is *On*, the program turns **on** the corresponding output.

If the tested input is *Off*, the program turns the corresponding output **off**.

9.2. MONITORING VARIABLE VALUE

To monitor a variable value, RoboCell offers the following tools:

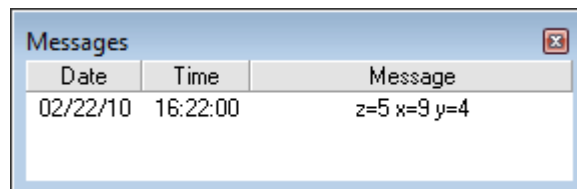
- When RoboCell is running and a **Set Variable** command is executed, the current value of the variable is displayed in the status line at the bottom of the screen.
- When the program is *not* running, click the **Set Variable** command in which the value is assigned to the variable, and then execute this line using the Run Single Line command. The variable value is displayed in the status line at the bottom of the screen.
- The **Print to Screen & Log (PS)** command can also be used to print the actual value of a variable, by placing the variable name within single quote marks in the text to be printed.

For example:

```
Set Variable z = 5
Set Variable x = 9
Set Variable y = x - z
Print to screen z='z' x='x' y='y'
```

When the program is running, the following messages are displayed in the status bar:

- When the first line is executed, the message is z=5.
- When the second line is executed, the message is x=9.
- When the third line is executed, the message is y=4.
- When the fourth line is executed, this window is displayed:






Messages		
Date	Time	Message
02/22/10	16:22:00	z=5 x=9 y=4

10. Program Execution

10.1. RUNNING A PROGRAM

RoboCell offers three modes of running a program. To select the running mode, click the appropriate Run icon in the Home tab or the mode's keyboard shortcut.

Running a Program Options		
Icon	Option (Shortcut)	Description
	Step (F6)	Runs the currently selected (highlighted) line
	Cycle (F7)	Runs the program from the currently selected (highlighted) line. Running stops after the last line is executed
	Loop (F8)	Runs the program from the currently selected (highlighted) line. After the last line is executed, program execution continues from the first line

Notes:

- The PLAY key on the programming pendant cannot be used to start execution of RoboCell programs.
- Always restart execution of a program from the program's top line after you have changed program data (e.g., recorded new coordinates for a position, edited a program line, etc.).
- The Home tab is locked when a program is running. This occurs in order to keep the Stop and Pause buttons at the front of the screen.

10.2. HALTING PROGRAM EXECUTION

Stop and Pause are the two methods of halting program execution in RoboCell.

To stop or pause programs, perform one of the following:

- Click the appropriate icon in Home tab.
- Press **F9 / F10**. Make sure the RoboCell application is the currently active window before you press **F9 (STOP)** or **F10 (PAUSE)**.
- In On-Line mode, you can also press the red EMERGENCY STOP key on the Programming Pendant.

Halting the Program		
Icon	Option (Shortcut)	Description
	Stop (F9)	Program execution is stopped immediately. Use this command only in emergencies.
	Pause (F10)	Stops program execution only after the current command has been executed. Thus, axes may continue moving (to complete their motion) after the Pause command is issued.

10.3. INPUT AND OUTPUT STATUS

The Data Dashboard shows the status of the controller digital inputs and outputs.

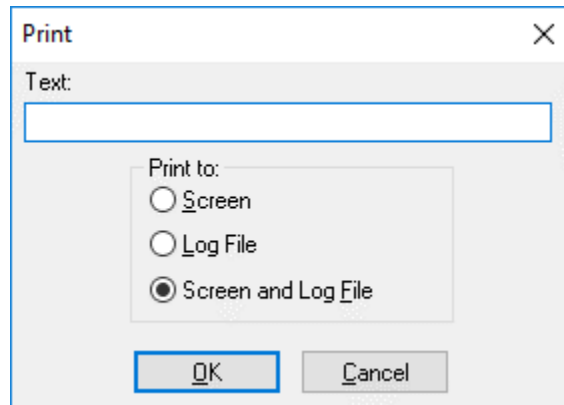
Data Dashboard																	
Robot Actual Position																	
Cartesian Coordinates	X (mm)	Y (mm)	Z (mm)	Rx (Deg.)	Ry (Deg.)	Rz (Deg.)											
		439.00	0.00	275.00	180.00	-90.00	0.00										
Joint Angles (Deg.)	Swivel Base	Lower Arm	Upper Arm	Arm Roll	Wrist Bend	Tool Flange											
	0.00	0.00	0.00	0.00	0.00	0.00											
Encoder Pulses																	
S:	0	L:	0	U:	0	R:	0	B:	0	T:	0	7:	0	8:	0		
Digital I/O																	
Inputs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
Outputs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	

For both inputs and outputs, when the I/O status is *Off* (false), the matching I/O number is off (grey). When the I/O status is *On* (true), the matching I/O number is on (green).

If control is On (Servo On is selected), clicking on an output number toggles the status of the I/O box's digital outputs.

10.4. ROBOCELL LOG FILE

The RoboCell log file records the messages printed using the **PS (Print to Screen Log)** command. To print to a message to the log file, click either **Log File** or **Screen and Log File**.



The Log file is initialized (cleared) each time RoboCell is loaded.

To see the file content, open the file SCBS.LOG using a text editor (such as WordPad). The file is located in the RoboCell subdirectory named BIN.

RoboCell generates a SCBS.BAK backup file of the SCBS.LOG file each time the program is closed.

11. Replicating Real and Simulated Cells

RoboCell's representation of the MHJF is based on actual dimensions and functions of the actual robot. Thus, programming performed in RoboCell can be used with an actual robotic installation.

11.1.1. From Simulation to Practice

Using CellSetup and Cell Simulation, create a cell, record all positions to be used by the program, write the program, and verify program execution.

Then, to construct an actual robotic installation that duplicates the simulated cell:

1. Using CellSetup as a guide, place all objects and devices in their approximate locations.
2. To print a 3D Image, select **File | Print 3D Image**.
3. If necessary, hide labels and/or change the angle and zoom of the camera, and print several cell maps.
4. Consider your program and determine which of the recorded positions must be precise; for example, the point at which the robot takes an object from a feeder, or the point at which the robot places a part in a machine.
5. In On-Line mode, send the robot to these key positions. Adjust the location and orientation of the actual objects and devices (e.g., feeder, machine) according to the location of the gripper.

If you are unable to relocate the object or device, you can record the position coordinates additional times.

11.1.2. From Practice to Simulation

To create a simulated cell based on an actual installation, you need to know the exact location and orientation of every element in the cell.

If the cell layout is simple, this information can be obtained through a coordinate grid or a ruler. For a more complex cell layout, you may need a technical drawing (e.g., an AutoCAD) dwg file, which shows the center point and orientation of all objects.

All measurements should be made with both the robot and the cell at the same point of origin and with the same orientation.

12. Sample RoboCell Projects

Your RoboCell package contains a number of sample projects to help familiarize you with the capabilities of RoboCell. These are accessible by means of the open file command. Navigate to **File | Open** or press **Ctrl + O** on your keyboard. You are then shown a listing of all the *.WS files in the following default installation subdirectory:

C:\Intelitek\RoboCell\Projects\Samples

12.1. SAMPLE PROJECT DESCRIPTIONS

This section provides a description of the projects that are installed by default together with RoboCell.

AllMovements

Includes the robot, a table, and two different objects. The program includes a series of movements that run on a permanent loop.

MHJF Cell1

Includes the robot, a table, and nine different objects in a non-descript room. No program is included.

MHJF Cell2

Includes the robot on a slidebase, a table, and three different objects. No program is included.

Pick and Place

Includes the robot, a table, and eight different objects. The program includes two pick and place sequences that are run on a permanent loop.

Tower of Hanoi

Includes the robot, a table, three targets, and three pneumatic feeders. The robot takes parts from the feeders and builds towers of various sizes and shapes.

13. Troubleshooting

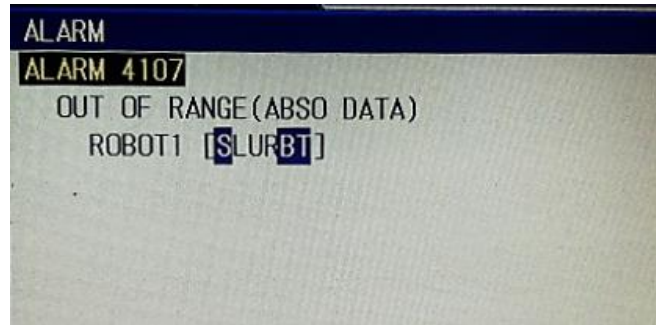
13.1. GENERAL TROUBLESHOOTING

This table covers problems that you may encounter when working in RoboCell.

Troubleshooting in RoboCell	
Problem	Potential Solution
The installation is unsuccessful.	Make sure all applications, including anti-virus monitors and network drivers, are closed. Then try installing the software again.
The program does not respond correctly.	There may be a message box hidden behind the application window. Minimize the application windows to check for messages or prompts.
The robot cannot be dragged to another position.	By default, the robot is placed at the cell's (XY plane) point of origin to simplify the teaching of positions. The robot's position can be changed only by means of its Properties menu in CellSetup.
The screen layout is jumbled or crowded.	Select Home Compose Screen to reset the screen layout. You can also select Home User Screen to reset a screen layout that you set and saved previously.
The peripheral axes do not move in RoboCell.	When RoboCell is operating on-line, the peripheral settings defined in the CellSetup file are not loaded; the peripheral setup defined in RoboCell is retained. Simulated peripheral axes that do not match the RoboCell definitions do not respond to RoboCell commands in the graphic display.
On-Line mode is inoperable.	Ensure that your computer is connected to the FS100 controller and the programming pendant's MODE key is set to REMOTE . Also ensure that the USB dongle (key) is inserted into the USB slot on your computer and that the dongle's driver is installed.
An error message (Alarm 4107) is displayed when the FS100 controller or RoboCell is turned on or turned on-line.	Use the FS100's programming pendant to resolve the error. For more information, see section 13.2 Resolving the Common Movement Alarm (Alarm 4107) on page 115.

13.2. RESOLVING THE COMMON MOVEMENT ALARM (ALARM 4107)

The common movement alarm typically occurs when one or more of the robotic arm's axes are moved while the controller is off. Because the weight of the pneumatic gripper may cause the MHJF's T-axis (wrist bend axis) to rotate slightly, the activation of **Alarm 4107** (shown below on the programming pendant screen) may occasionally occur.



Note that the alarm can be *dismissed* by simply pressing **RESET** on the programming pendant's screen. However, before the alarm is resolved, you cannot play, edit, or create any jobs (motion programs). Instead, the controller will return a **Position not checked** error message.

To resolve alarm 4107, perform the following procedure using the programming pendant:

1. Turn the MODE key to **Teach** mode.
 2. Press the SERVO ON READY key to initiate servo power.
 3. On the programming pendant's screen, press **RESET**.
 4. In the main menu, navigate to **ROBOT | SECOND HOME POS.**
 5. Lightly grip the enable switch and press FWD. The robot is manipulated to the second home position.
 6. Ensure that all values in the **Current** column are equal to their corresponding values in the **Specified** column.
 7. In the Menu Area at the top-left corner of the screen, select **DATA | CONFIRM POSITION**. The position of the robot is confirmed. You can once again create, edit, and play jobs.
- ① **Note:** To avoid activating alarm 4107, it is recommended that the controller be switched off with the gripper pointed directly downwards. This way, the weight of the gripper will not cause the arm's B-axis to rotate when the controller is off.