

# SCORBASE

For

***ER 4u***

## User Manual

Catalog #100342 Rev. A

intelitek▶▶



Copyright 2001 Intelitek Inc.

Catalog # 100342 Rev. A

July 2001

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.

**INTELITEK INC.**  
444 East Industrial Park Drive  
Manchester NH 03109-537  
Tel: (603) 625-8600  
Fax: (603) 625-2137

Web site [www.intelitek.com](http://www.intelitek.com)



# Table of Contents

---

---

<b>CHAPTER 1</b>	<b>Features of SCORBASE for SCORBOT-ER 4u</b>	<b>1</b>
<b>CHAPTER 2</b>	<b>Starting SCORBASE</b>	<b>3</b>
	System Requirements .....	
	Software Installation .....	
	Uninstalling the Software .....	
	Activating the Software .....	
	Quitting the Software .....	
<b>CHAPTER 3</b>	<b>Menus Overview</b>	<b>7</b>
	The SCORBASE Window .....	7
	File Menu.....	8
	Edit Menu .....	9
	Run Menu.....	10
	Options Menu .....	11
	View Menu.....	12
	Window Menu.....	14
	Help Menu .....	14
	Tool Bar.....	15
	Charts.....	16
<b>CHAPTER 4</b>	<b>Homing and Control</b>	<b>19</b>
	On Line / <i>Off-line</i> modes.....	21
	Control On / Control Off – CON/COFF.....	22
<b>CHAPTER 5</b>	<b>Position Definition</b>	<b>23</b>
	Cartesian and Joint Coordinate Systems .....	24
	Absolute and Relative position.....	26
	Manual Movement Dialog Box .....	27
	Robot Movement Dialog Box .....	29
	Teach Positions Dialog Box .....	29
	Record Positions (Joints Coordinates).....	32
	Record Relative Position 2 & Pro.....	32
	Teaching Positions (XYZ Coordinates) .....	33

	List Positions window.....	35
	Dialog Bars.....	36
	Encoder Counts Dialog Bar .....	36
	XYZ Dialog Bar.....	36
	Joints Dialog Bar .....	36
	Using a Teach Pendant with SCORBASE .....	37
<b>CHAPTER 6</b>	<b>Program Editing</b>	<b>39</b>
	Opening and Closing a Program.....	39
	Program Editing Tools .....	40
	Adding and Editing Commands .....	42
	Axis Control Commands .....	43
	Program Flow Commands .....	50
	Input/Output Commands.....	60
<b>CHAPTER 7</b>	<b>Variable Programming</b>	<b>65</b>
<b>CHAPTER 8</b>	<b>Program Execution</b>	<b>69</b>
	Running a Program .....	70
	Halting Program Execution .....	70
	Digital Inputs & Outputs Dialog Bar.....	71
	Analog Inputs & Outputs Dialog Bars.....	72
	Log File.....	72
<b>CHAPTER 9</b>	<b>Project Files Management</b>	<b>75</b>
	Project Management.....	75
<b>CHAPTER 10</b>	<b>System Setup</b>	<b>77</b>
	Display Options .....	77
	Scorbase Level.....	79
	Peripheral Setup .....	80

# CHAPTER 1.

---

---

## Features of SCORBASE for *SCORBOT-ER 4u*

This manual describes all the features and operations for all levels of SCORBASE. When necessary, illustrations show the differences in the levels, and descriptions note the availability of options and commands.

SCORBASE for *SCORBOT-ER 4u* is a robotics control software package, for robot programming and operation. SCORBASE for *SCORBOT-ER 4u* provides numerous capabilities:

- \* Communication with the robot controller over USB channel.
- \* Control and real-time status display of five robot axes, gripper and two peripheral axes.
- \* Full support and real-time status display of 8 digital inputs, 8 digital outputs, 2 analog outputs and 4 analog inputs.
- \* Position definition and display as well as manual robot movement in joint coordinates (encoder units).
- \* Cartesian coordinates (X,Y,Z, Pitch and Roll) are also available.
- \* Robot movement definition as Joint, Linear, or Circular, with 10 active speed settings. (Availability depends on level.)
- \* Default setting of 1000 positions and 1000 active program lines.
- \* Interrupt programming for handling responses to changes in digital input status.
- \* Variable programming, in 3 levels of complexity, to moderate the learning curve. This makes it possible for beginners to start at a lower level, and advance through the levels as they become more skilled in robotics programming.
- \* Saving and loading projects.
- \* SCORBASE can be installed as part of RoboCell, an interactive graphic software package, which provides simulation of the robot and other devices in the workcell.



## CHAPTER 2.

---

---

# Starting SCORBASE

The instructions in this chapter are for SCORBASE only.

If you will be installing SCORBASE as part of the RoboCell software package, follow the instructions in the chapter, Starting RoboCell, in the RoboCell User's Manual.

## System Requirements

SCORBASE for ER 4u computer requirement are:

- \* Pentium III with 350 MHz processor, or higher.
- \* At least 128 MB of RAM.
- \* A hard drive with at least 20 -Megabyte of free disk space.
- \* Windows 98/2000.
- \* A Super VGA or better graphics display, minimum 256 colors.
- \* A Mouse or other pointing device.
- \* USB port.

## Software Installation

The SCORBASE software is supplied on a CD.

To install the software, do the following:

1. Insert CD into the CD-ROM drive.
2. Choose *Run* from the Start menu.
3. Type D:\Setup and click OK. If necessary replace D:\ with the letter of your CD-ROM drive.
4. Follow the instructions for installation as they appear on the screen.

## Uninstalling the Software

To uninstall SCORBASE -

1. Choose *Programs* from the Start menu
2. Select *SCORBASE for ER 4u*
3. Activate *Uninstall* command

## Activating the Software

To start SCORBASE, do the following:

1. Make sure that all components that will be used are installed and connected according to the installation procedures detailed in the User manuals supplied with the robot and controller.
2. Turn on the computer and the controller.
3. Choose *Programs* from the Start menu
4. Select *SCORBASE for ER 4u*.
5. Select *SCORBASE for ER 4u* command

The SCORBASE application window will appear.

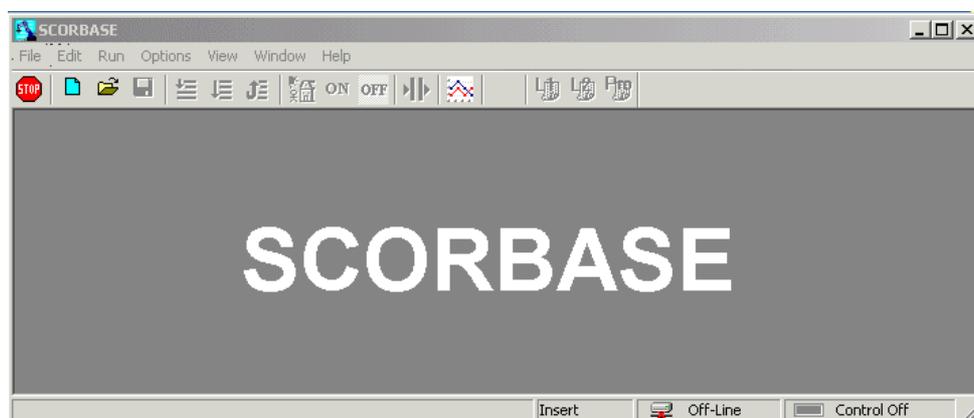
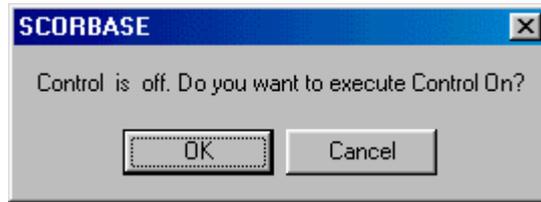


Figure 2- 1: SCORBASE Opening Window

**Notes:**

If SCORBASE opens in On-line mode the controller is detected and the following message appears



Select OK to select Control On (CON) state or cancel to remain in Control Off state.

If SCORBASE opens in *Off-line* mode or controller is not detected it will load and operate in *Off-line* mode.

Only one application of SCORBASE can be active at a time.

## Quitting the Software

Stop SCORBASE program (if running) and save project data.

Use any of the following standard Windows methods:

1. In SCORBASE, select File | Exit.
2. Press Alt + F4
3. From the Title-bar click on application icon and select Close

If changes in project are not saved, SCORBASE prompts you to save the changes before closing.

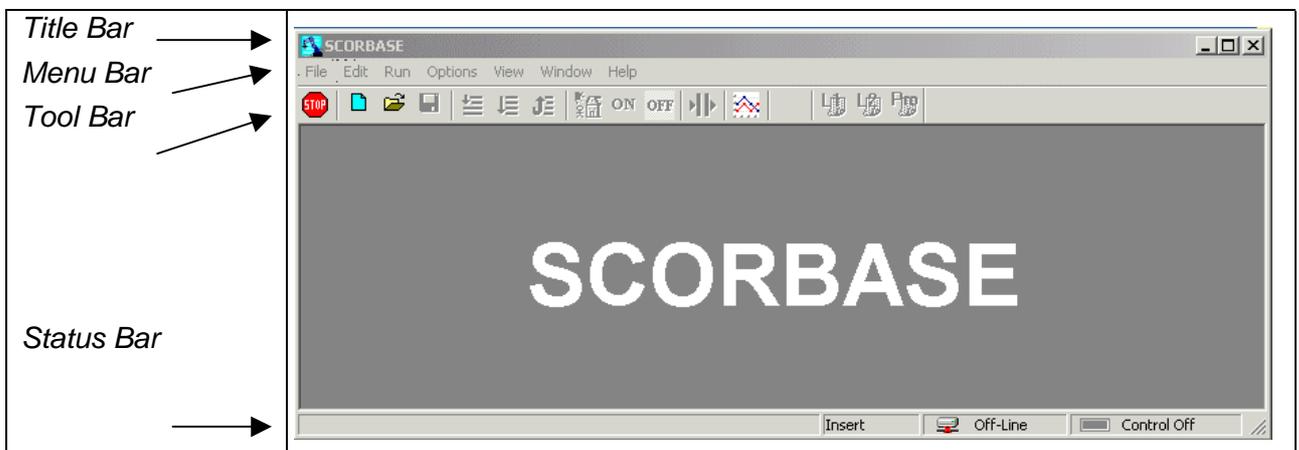


# CHAPTER 3.

## Menu Overview

### The SCORBASE Window

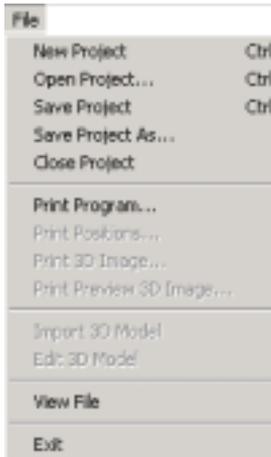
The figure below shows the main bars of SCORBASE screen:



- Title Bar* Contains the usual Windows controls for sizing and closing the application screen.
- Menu Bar* Contains menus with SCORBASE commands
- Tool Bar* Contains icon menu with frequently used SCORBASE commands.
- Status Bar* Displays, at the bottom of the screen, information regarding the SCORBASE software, modes of operation, current activity, etc. When you position the mouse over an icon, a description of the icon appears in the status bar.

# File Menu

A SCORBASE project contains a program (SBP file), user-defined positions (PNT file), and a project data file (WS file). If RoboCell is used, the project will include the cell image (3DC file). Throughout this manual, the term “project” will refer to the programs positions (and image) files saved by the user as one entity.



The File menu contains the usual Windows functions that allow you to open new or previously saved projects, save and close projects. You can print files containing robot programs and positions, and exit the software.

The File menu options are:



*New Project*  
**CTRL+N**

\* Opens a new, untitled, project.



*Open Project*  
**CTRL+O**

Opens an existing project.



*Save Project*  
**CTRL+S**

Saves the currently active project.

*Save Project As...*

Saves the currently active project under a new project name.

*Close Project*

Closes the currently open project.

*Print Program*

Prints the program. (Program window must be active).

*Print Positions*

Prints the recorded positions. (Positions window must be active)

*Print Preview 3D Image*

Reserved for RoboCell program option

*Import 3D Model*

Reserved for RoboCell program option

*Edit 3D Model*

Reserved for RoboCell program option

*View File* Opens View File window to display the Program or Position files.

*Exit* Quits SCORBASE

If changes to a program or position file have been made but not yet saved, a warning message will be displayed before you exit Scorbaser.

For more information on SCORBASE file management, see Chapter 9.

## Edit Menu



The Edit menu contains the usual Windows functions that allow you to edit program files.

*Cut*  
**CTRL+X** Deletes selected text or lines from the program file, and places it on the Windows and SCORBASE clipboard.

*Copy*  
**CTRL+C** Places a copy of selected text or lines from the program file on the Windows and SCORBASE clipboard.

*Paste*  
**CTRL+V** Inserts the contents of the SCORBASE clipboard into the program file.

*Find*  
**CTRL+F** Opens a dialog box which allows you to search for a particular text string, SCORBASE command, or command argument:

*Find Next* Repeats the last Find operation for the next occurrence.

*Command/  
Remark (\*...)*  
**CTRL+R** 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 will jump to the line you specify.

*Edit Line* Edit a selected line. Useful for long programs.

For more information on these and additional SCORBASE editing functions, see Chapter 6.

# Run Menu

Run	
Search home - all axes	
Search home - robot	
Search home - peripherals	
Run single line	F6
Run single cycle	F7
Run continuously	F8
Go home - all axes	
Go home - robot	
Go home - peripherals	
Pause	F10
Stop	F9

The Run menu contains SCORBASE commands for homing the robot and peripheral axes, and executing programs.

*If the software is operating -line, only the Run program options will be available in this menu.*



**Search Home - all axes**

Homes both the robot and any configured peripheral axes.

**Search Home - robot**

Homes the robot.

This command is available only if system was homed once, after opening SCORBASE.

**Search Home - peripheral**

Homes the peripheral.

This command is available only if system was homed once, after opening SCORBASE.



**Run Single Line (F6)**

Executes the selected (highlighted) program line.



**Run Single Cycle (F7)**

Executes the program from the selected (highlighted) program line to the end of the program.



**Run Continuously (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.

**Go Home – All axes**

Sends the robot and peripherals to their home positions.

This command does not home the system.

**Go Home - Robot**

Sends the robot to its home position.

This command does not home the robot.

**Go Home – Peripheral**

Sends the peripherals to their home position.

This command does not home the peripherals.



*Pause (F10)*

Stops program execution after the current line is executed.



*Stop (F9)*

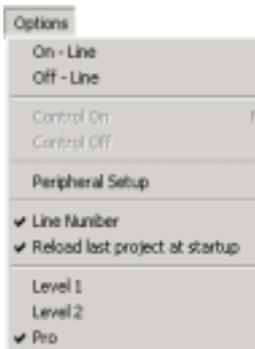
Immediately stops program execution and movement of all axes.

*Pause and Stop are software methods for halting program execution. In an actual emergency situation, you should use the EMERGENCY button on the controller or the ABORT key on the Teach Pendant*

For more information on SCORBASE setup options, see Chapter 10.

For more information on program execution, see Chapter 8.

## Options Menu



The Options menu allows you to define your preferences for operating the software.

*On-Line*

Establish communication with the controller.

*Off-line*

SCORBASE does not communicate with the controller, even though it may be connected. *Off-line* mode is useful for checking and debugging programs.



*Control On (F5)*

Enables servo control of the axes and I/O device.



*Control Off*

Disables servo control of the axes and communication with I/O device.

*Peripheral Setup*

SCORBASE allows you to define the peripheral devices, which are connected and operated by the controller as axes 7 and 8.

*Line Number*

Toggles the display of program line numbers in the program window (On and Off).

*Reload last project at startup*

When checked opens the last project saved when SCORBASE is started.



*Level 1*

Displays list of commands and options at introductory level. Commands related to higher level are disabled.



*Level 2*

Displays list of commands and options at advanced level. Commands related to higher level are disabled.



*Pro*

Displays list of all commands and options.

For more information on SCORBASE setup options, see Chapter 10.

## View Menu



The options in the View menu allow you to toggle the display of SCORBASE dialog bars and menus.

### *Workspace*

Toggles the display of the workspace window. Through this window, the user can access the project files and the SCORBASE commands tree.

### *Manual Movement*

Toggles the display of the manual movement dialog box. This box enables manual control over the movements of the robot, the gripper and peripheral axes.

### *Teach Positions*

Toggles the display of the Teach Position dialog box. This box enables recording, teaching and deleting positions. It also enables the user to send the robot and/or peripherals to a previously defined position.

### *Robot Movements*

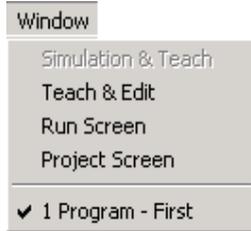
Opens a dialog box that displays the robot axes. Pressing on a selected axis will move it in the selected direction.

### *Charts*

Opens Charts window. See end of this chapter.

<i>Dialog Bars</i>	<p>Toggles the display of seven dialog bars that enable the following:</p> <ul style="list-style-type: none"> <li>* <i>Joints</i> – Displays the robot’s joints angle (five joints)</li> <li>* <i>XYZ</i> - Displays the robot tool center point (TCP) position and orientation. The coordinate system origin is at the center of the robot base at table level.</li> <li>* <i>Digital outputs</i> - Displays the status of digital outputs 1-8 (dark green - off; light green - on). You may click on an output to toggle its status</li> <li>* <i>Digital inputs</i> - Displays the status of digital inputs 1-8 (dark green - off; light green - on). In <i>Off-line</i> clicking on input toggles its status.</li> <li>* <i>Analog outputs</i> - Displays the value of analog output 1&amp;2 (0-255)-(0-10 volt).</li> <li>* <i>Analog inputs</i> - Displays the value of analog input 1-4 (0-255). In <i>Off-line</i> you may set the analog input value.</li> <li>* <i>Encoders</i> – Displays the encoder counts of axes 1-8. (Encoders are zeroed after homing the system.)</li> </ul>
<i>Show All Dialog Bars</i>	Displays all seven dialog bars.
<i>Close All Dialog Bars</i>	Close all seven dialog bars
<i>Movement Information</i>	<p>Displays the following data:</p> <ul style="list-style-type: none"> <li>* Position error of eight axes in encoder counts</li> <li>* Home switch status of all eight axes</li> <li>* PWM for one selected axe. The PWM (Pulse Width Modulation) is the controller output for the selected axe motor.</li> </ul>
<i>Messages</i>	<p>Opens the Messages window.</p> <p>The data printed in the Messages window is printed using the PS (<i>Print to Screen &amp; Log</i>) command.</p>
<i>Toolbar</i>	Toggles the display of programming tools bar (default on).
<i>Status Bar</i>	Toggles the display of lower status bar (default on)

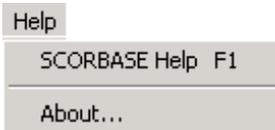
## Window Menu



The Windows menu enables selecting one out of four SCORBASE window setups.

<i>Simulation &amp; Teach</i>	Sets the screen to display the RoboCell window and dialog bars required to define positions.
<i>Teach &amp; Edit</i>	Use this option to open Teach position, Manual movement, Program and Workspace windows.
<i>Run Screen</i>	Displays the dialog boxes and menus used for program execution.
<i>Project Screen</i>	Displays the Program and Positions of the Project.

## Help Menu



The Help menu features:

<i>SCORBASE Help (F1)</i>	Displays SCORBASE help files
<i>About</i>	SCORBASE Software Version number.

## Tool Bar

The tool bar contains buttons to execute commonly used tasks.



Icon	Design	Function
	<i>Stop</i>	Immediately stops program execution and movement of all axes.
	<i>New</i>	Opens a new, untitled project.
	<i>Open</i>	Opens an existing project.
	<i>Save</i>	Saves the currently active project.
	<i>Run single line</i>	Executes the selected (highlighted) program line.
	<i>Run single circle</i>	Executes the program from the selected (highlighted) program line, to the end of the program..
	<i>Run continuously</i>	Executes the program from the selected (highlighted) program line. When the last program line is reached, the program starts again from the first line.
	<i>Search Home</i>	Search Home for all axes
	<i>Control On</i>	Enables servo control of the axes.
	<i>Control Off</i>	Disables servo control of the axes. When control is off, axes cannot be moved.
	<i>Pause</i>	Stops program execution after the current line is executed.
	<i>Charts</i>	Opens the Charts window
	<i>Level 1</i>	Displays list of commands and options at introductory level. Commands related to higher level are disabled.
	<i>Level 2</i>	Displays list of commands and options at advanced level. Commands related to higher level are disabled.
	<i>Pro</i>	Displays list of commands and options at professional level. At this level, all options and commands can be activated.

# Charts

SCORBASE charts can be configured to display the following data (on Y-axis) vs. time (on X-axis).

1. Encoder counts (axis position), represent the actual axis position. The encoder counts are shown in a blue line.
2. Position error is the difference between the required axis position and the actual axis position. The error is expressed in encoder counts and shown in a red line.
3. The PWM value represents the controller output. The error and the axis control parameters determine the PWM value shown in a green line.

Each axis data (1-8) is shown in a different chart. A sample chart for axes 1 and 2 is shown in Figure 3- 1.

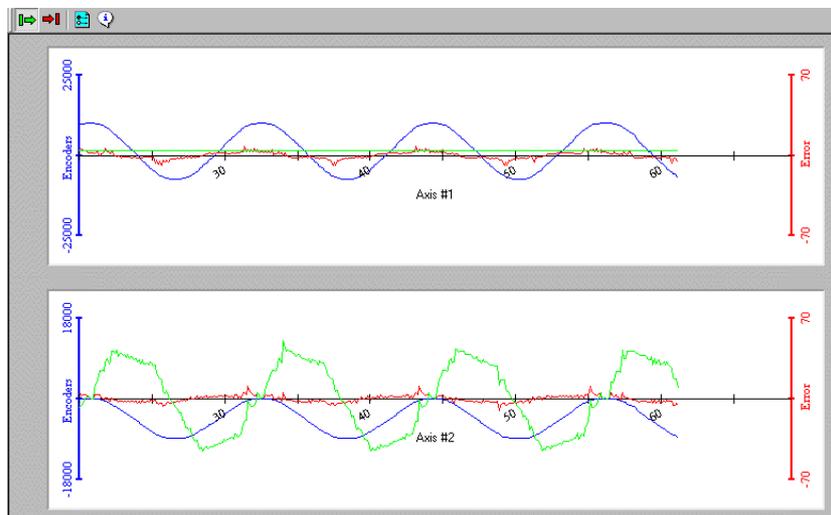


Figure 3- 1: Chart for 2 Axes

To open the Charts window select one of the following options:

- \* Press the *Charts* tool in the Tool bar 
- \* Select *View | Charts* menu command

Use the Chart window tool bar to select one of the following options:

Icon	Design	Function
	<i>Start chart</i>	Starts drawing the chart
	<i>Stop chart</i>	Stops drawing the chart
	<i>Options</i>	Opens a dialog box to select the data range that will be displayed in the chart(s). The following table is opened.



Click and check to select the data that will be displayed in the charts. Every axe is shown in a separate chart.

In the example shown above the encoder, error and PWM of axes 1 & 2 will be displayed. Since every axis is shown in a different chart two charts will be opened (as shown in Figure 3- 1).



*Legend*

Displays the colors used in the charts for encoder, error and PWM -

*Encoders* – Blue

*Error* – Red

*PWM* – Green



## Homing and Control

### General

The robot and peripheral axes location is monitored and controlled using encoders. To initialize the encoders and to obtain repeatable performance the axes must first reach a predefined position known as *hard home*. All recorded positions and movements refer to the Hard Home position. The homing procedure finds the hard home for the selected axes.

SCORBASE offers two commands related to the home position.

1. The first one known as *Search Home* starts the homing procedure. During *Search Home* each axis is homed separately. The controller activates the currently homed motor axis, until its micro-switch is pressed. Then the controller initializes the axis encoder counter and turn to home the next axis. After all configured axes are homed, homing procedure ends.
2. The second command related to home position is named *Go Home*. This command sends the selected axes to a position where the encoders value is zero. *This command does not home the axes!!*

The two commands are available in three levels:

- \* *Search / Go home all* (applies for all active axes)
- \* *Search / Go home robot* (applies for the robot)
- \* *Search / Go home peripherals*. (applies for axes 7 & 8).

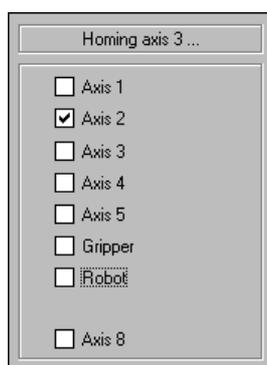
## Search Home - All Axes

To start the homing procedure, do one of the following:

- \* Select Run | Search home - all axes

- \* Click the Search Home icon. 

A window is opened displaying the number of the currently homed axis. Each time an axis is successfully homed, a checkmark appears next to the axis number. After the five axes and the gripper are homed, a checkmark appears next to robot.



To abort homing while the procedure is in progress, do one of the following:

- \* Press F9 (Stop command)
- \* Press the red EMERGENCY button on the controller.
- \* Press the EMERGENCY key on the Teach Pendant.

### Notes:

If homing procedure fails a message will appear.

The *Search Home - All Axes* command executes the robot's homing procedure as well as any peripheral devices that have been configured in the *Options / Peripherals Setup* menu

If the system was homed and SCORBASE mode was changed to *Off-line* mode there is no need to home the system again when returning to *On-line* mode.

When SCORBASE is *On-line* for the first time only, the *Search Home All Axes* command is available.

When SCORBASE is *Off-line*, or when RoboCell is installed, the homing procedure is not required, although it can be executed. The homing procedure will initialize Joint and XYZ values according to a software definition. All encoders will be set to 0, while the robot Cartesian coordinates will be set according to a software model.

## Search Home - Robot

Runs homing procedure for the robot.

Homing of the robot is enabled only after the system was homed once.

## Search Home - Peripherals

Runs homing procedure for the configured peripherals.

Homing of the robot is enabled only after the system was homed once.

## Go Home All Axes – Robot - Peripherals

After axes have been homed, selecting *Run / Go Home - All Axes* will send them back to their home position at any time. This command sends the robot and peripherals to a position where the axes encoders value equals zero. The *Go Home command* does not run homing procedure

Selecting the *Go home - robot* or *Go home - peripherals* will send the selected axis to its home position.

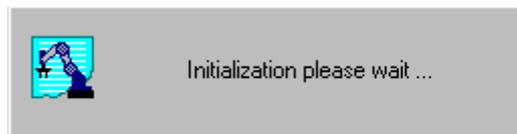
## On Line / Off-line Modes

Scorbase can run either in *On-line* or in *Off-line* modes. In *On-line* mode, SCORBASE communicates with controller over the USB channel. If *Control on* state is selected, SCORBASE controls the robot, peripherals and I/O device. In *Off-line*, SCORBASE can be used only in *Control off* state (useful for programming and debugging). The active mode is displayed in the status bar.

To change to On-line, select *Options / On-line*.

To change to Off-line, select *Options / Off-line*.

If SCORBASE is opened in *On-line*, or *On-line* is selected from the options menu, the following window appears as SCORBASE searches for the controller.



If controller is detected, the On-line mode is started

If controller is not detected, Off-line mode remains

**Note:** SCORBASE's opening mode is similar to its closing mode.

## Control On / Control Off – CON/COFF

The Control On state, enables servo control of the axes and I/O device. It is available only in On-line mode. In the Control Off state, axis movement commands cannot be executed. The Control state (On or Off) is displayed on the lower right corner of the status bar.

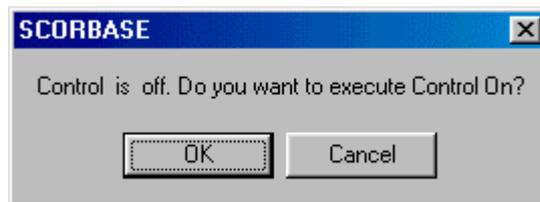
To enable control over the axes, do one of the following:

- \* Select Option | Control On
- \* Click the Control On icon. 
- \* Press the F5 key.

To disable control, do either of the following:

- \* Select Option | Control Off
- \* Click the Control Off icon. 

If SCORBASE opens in On-line, or On-line mode is selected after detecting the controller, the following dialog window will appear:



Press *OK* to select *Control On* or *Cancel* to remain in *Control Off*.

The controller will automatically disable control if an impact condition, trajectory error, or thermic overload error occurs during execution of a movement command. If attempting to move the axes when control is disabled, a *Control Off* message will be displayed.



When SCORBASE is in Off-line mode, the *Control On/Off* state cannot be altered.

# CHAPTER 5.

---

---

## Position Definition

Every SCORBASE project includes a set of pre-defined positions and a program that sends the robot from one position to the other. Prior to running a program all the positions used in that program should be defined. Scorbaser offers few sets of tools to define and store positions that will be used in the programs.

The following SCORBASE tools are used in the process of defining positions.

- \* Manual Movement dialog box
- \* Teach Positions (simple or expand).
- \* Robot movement dialog box
- \* Position data dialog bars
- \* Position window.

To activate the dialog boxes most useful for position definition mode, select *Window / Teach & Edit*.

Any of the following four methods can be used for Position definition:

No.	Method	Level
1	Absolute position Joint coordinates	1, 2, Pro
2	Relative position Joint coordinates	2, Pro
3	Absolute position Cartesian coordinates	2, Pro
4	Relative position Cartesian coordinates	2, Pro

**Note:**

Defining peripheral position can be done only in PRO level.

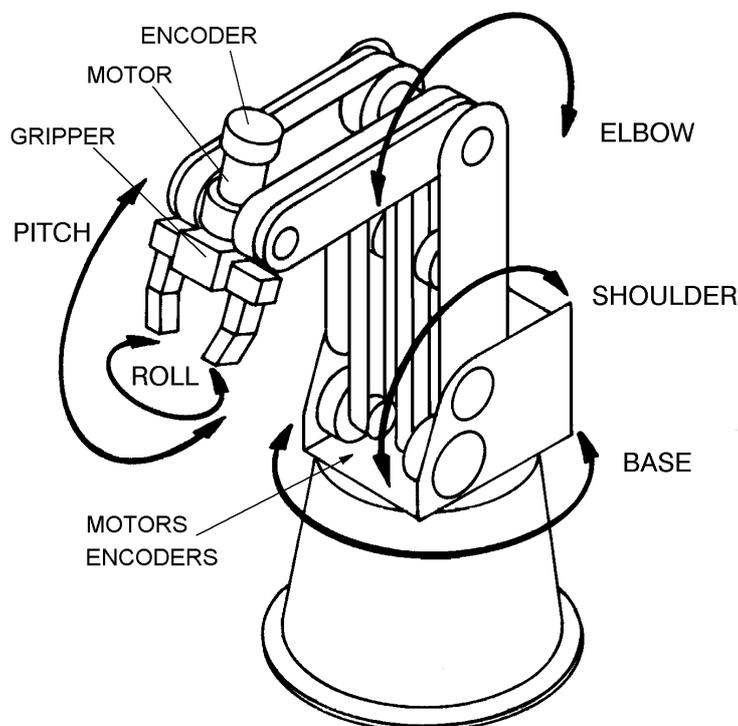
## Cartesian and Joint Coordinate Systems

Defining a position in SCORBASE can be done by using either the Joints or Cartesian coordinate systems. In both systems, a robot position is defined, using five parameters derived from the data supplied by the five axes encoders (encoder is an angular movement sensor attached to the axes motor).

A Peripheral position is always defined using one variable that stores the sensor output (encoder value) of that position.

### Joint Coordinate System

A robot position in Joint coordinates is defined by five angle values, representing each angle of the joints. The joint names are Base, Shoulder, Elbow, Pitch, and Roll.



*Figure 5- 1: Robot Joints*

For example after homing the robot position in Joints coordinates is:

Axe #1 - Base = (0°)

Axe #2 - Shoulder = (-120°)

Axe #3 - Elbow = (~95°)

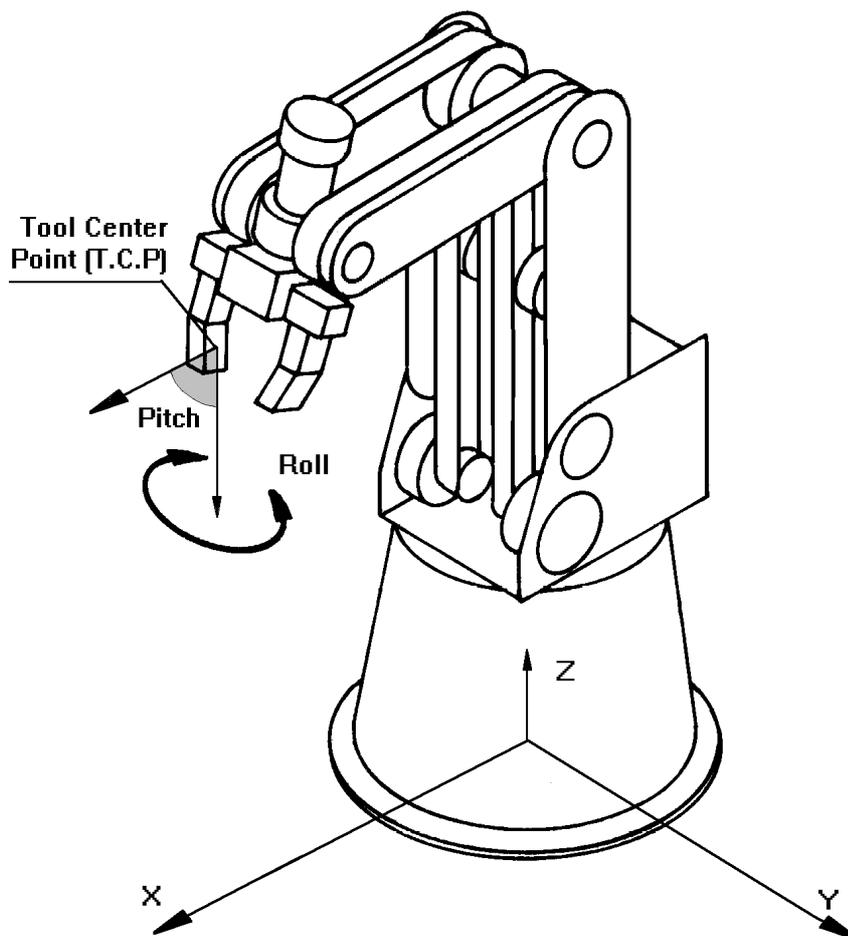
Axes #4 - Pitch = (~88°)

Axe #5 - Roll = (0°)

## Cartesian (XYZ) Coordinate System

A robot position in Cartesian (or XYZ) coordinates, is defined by five parameters. The first three parameters are:

- 1) the distance of the robot's Tool Center Point (TCP) from the point of origin (the center bottom of the robot base), along the three axes that describe three-dimensional space (X,Y,Z);
- 2 and 3) are the Pitch and Roll angles of the gripper specified in angular units.



*Figure 5- 2: XYZ*

For example: After homing, the robot position in Cartesian coordinates is:

X = (~169) [mm]

Y = (0) [mm]

Z = (~503) [mm]

Pitch = (~63°)

Roll = (0°)

Scorbase can transform and display a position recorded in one coordinates system, to another system.

## Absolute and Relative position.

SCORBASE offers two methods to define a robot or peripheral position. They are known as Absolute position and Relative position. The two methods are applicable in Cartesian or Joints coordinates.

An *absolute* position is defined using all five robot position parameters. If joint coordinate system is used, robot position is defined using the Base, Shoulder, Elbow, Pitch, Roll angles. In the XYZ coordinate system, a position is defined using the X, Y, Z values in millimeters, and the Pitch, Roll in degrees. An absolute position is usually a fixed position in world space.

A *relative* position is a position whose coordinates are defined as an offset from a *reference position* coordinates. If the coordinates of the reference position change, the relative position moves along with it, maintaining the same offset. A relative position can be defined in either in Cartesian or Joint coordinate values.

A position can also be defined as *relative to current*. In this case, the relative position is calculated as an offset from the coordinates of current robot position

## Record and Teach

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

*Record position*: defines a robot position in a Joints coordinates system.

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

## Manual Movement Dialog Box

Recording a robot position (in Joint coordinates) is done by manipulating the robot to the required position and then recording it. The Manual Movement dialog box allows direct control and manipulation of the robot and peripheral axes.

The Manual Movement dialog box is automatically opened when a project is opened, or when *Window / Teach & Edit* display setting is selected.

To open the Manual Movement dialog box when there is no open project, select *View / Manual Movement*.



Figure 5- 3: Manual Movement Dialog Box

The following chart explains how clicking the buttons in the Manual Movements dialog box (or pressing the matching keys on the keyboard) controls the robot and peripheral movements.

**Joints** When Joints is selected, clicking on the buttons (or pressing the matching keys on the keyboard) moves one robot axe at a time, as shown in the following chart:

Keys	Joint Motion
1 / Q	Rotates the BASE right and left
2 / W	Moves the SHOULDER up and down.
3 / E	Moves the ELBOW up and down.
4 / R	Moves the wrist (PITCH) up and down.
5 / T	Rotates the wrist (ROLL) right and left.
6 / Y	Opens and closes gripper via servo control.
7 / U	Moves peripheral axis #7 (if connected)
8 / I	Moves peripheral axis #8 (if connected)

## XYZ

When XYZ is selected, clicking the buttons (or pressing the matching keys on the keyboard) moves the TCP, as shown in the following chart.

Movements in XYZ mode are sometimes combined from simultaneous movements of few axes.

Keys	XYZ Motion
1 / Q	TCP moves along X-axis (back and forth) . .
2 / W	TCP moves along Y-axis (right and left) .
3 / E	TCP moves along Z-axis (up and down) .
4 / R	Axes move in order to change the gripper's pitch angle, while the TCP position does not change.
5 / T	The gripper rolls, and the TCP position does not change.



*Open Gripper /* Completely opens the gripper.



*Close Gripper* Completely closes the gripper.



*Speed* Selects the speed of manual movement speed: 10 fastest; 1 slowest, default 5.

The robot can be manipulated from the Manual Movement dialog box before it has been homed, in Joint mode only. In fact, it is often necessary to bring the robot into a more suitable position before initiating the homing routine. However, an axis limit error message may appear during manipulation of a robot that has not been homed.

Movement of an axis continues as long as the button or key is pressed, or until a software or hardware limit is reached.

## Robot Movement Dialog Box

The Manual Movement dialog box enables control over the robot in XYZ and Joints modes.

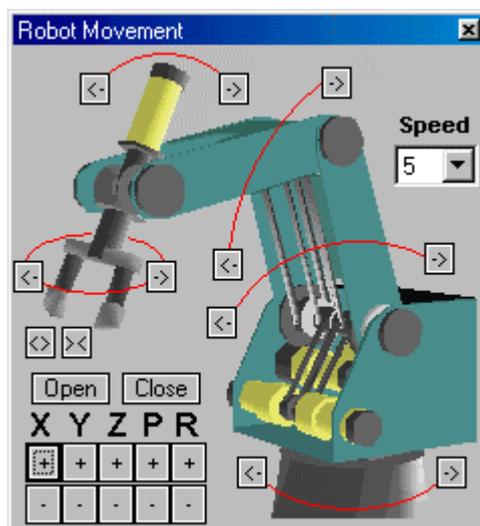


Figure 5- 4: Robot Movement Dialog Box

Clicking on an axis image or the XYZPR buttons, will move the robot as described in the Manual Movement dialog box.

To open the Robot movement dialog box select *View / Robot Movement*.

## Teach Positions Dialog Box

The Simple/Expanded Teach Positions dialog box enables the following:

1. Teaching positions (in joint coordinates)
2. Recording positions (in Cartesian coordinates).
3. Sending the axes to the recorded positions (when program is not running)

To get the description of the functions of each icon in the Teach Position Dialog Box, place the mouse on icon. *Do not activate mouse button.*

The user can define 1000 positions. A higher computer (CPU and memory) can hold more positions.

The Teach Position (Simple) Dialog box contains the following options:

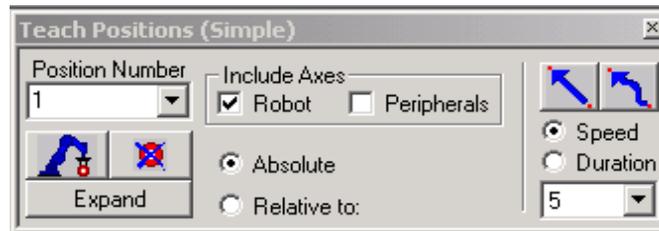


Figure 5- 5: Teach Positions Dialog Box (Simple)

Description	Level	Function
<i>Position Number</i>	1, 2 & Pro	A numerical name for position.
 <i>Record</i>	1, 2 & Pro	Records the current robot position, (in joint coordinates) to the position displayed in the position number field.
 <i>Delete</i>	1, 2 & Pro	Deletes the position in the position number field from memory.
<i>Expand</i>	2 & Pro	Opens the Teach Positions (Expanded dialog box), by toggling Simple/Expand.
<i>Include Axes</i>	Pro	Instructs the controller to define coordinates for Robot, Peripheral axes or both. Peripheral position can be recorded only in Pro level.
<i>Absolute / Relative to</i>	2 & Pro	Select whether the positions are defined as absolute or relative to another position. When <i>Relative to</i> is selected, a <i>Relative To</i> field appears. Select either <i>an existing position</i> or <i>current</i> . A position that is relative to current means the reference position will be the robot position at the time it is sent to that position.
<i>Go Position</i>		Executes the <i>Go Position</i> command. Sends the axes to the selected position (see note).
<i>Go Linear</i>	2 & Pro	Executes the <i>Go Linear to Position</i> command. This command sends the robot's TCP (Tool Center Point) from its current position to the selected position along a linear path (straight line). The linear motion applies only to the robot axes. (see note)
<i>Speed</i>	1, 2 & Pro	Selects the speed for all movement commands. 10 fastest, 1 slowest, 5 default.

<i>Duration</i>	Pro	Defines the time it will take to complete a movement command, Time is Defined in 10ths of a second. (see note)
-----------------	-----	--

**Note:** Read Chapter 7 to find more on these commands.

## Expand (2 & Pro)

Clicking the Expand button opens the Teach Positions (Expanded) dialog box that contains the following options:

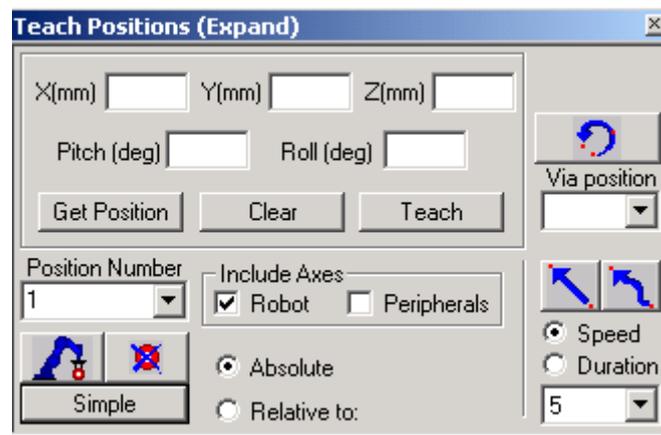


Figure 5- 6: Teach Positions Dialog Box (Expand)

<b>Description</b>	<b>Function</b>
<i>X(mm), Y(mm) Z(mm), Pitch (deg.) Roll (deg.)</i>	Fields for displaying or changing the Cartesian coordinates of the selected position.
<i>Get Position</i>	Displays the Cartesian coordinates of the selected position.
<i>Clear</i>	Clears all position coordinate fields. The position data is not changed.
<i>Teach</i>	Teaches position using Cartesian coordinates system.
<i>Go Circular</i>	Executes the <i>Go Circular to Position</i> 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 <i>Via position</i> field.  The circular motion applies only to the robot
<i>Via position</i>	Selects the intermediate position through which the <i>Go Circular</i> movement passes.
<i>Simple</i>	Returns to Simple teach position dialog box.

# Record Positions (Joints Coordinates)

## Record Absolute Position

1, 2 & Pro

To record an absolute position, do the following:

4. Using either the Manual Movement dialog box or the Robot Movement dialog box move the robot to the position you want to record.
5. Click on the Teach Position dialog box.
6. Type a Position Number in the *position number* field or select existing positions number if you want to modify (overwrite) a previously defined position.
7. Select *Absolute*.
8. Select *Include Axes* option (robot or Peripheral):
  - a) *Robot*: to record a position for the robot axes.
  - b) *Peripherals*: to record a position for the peripheral axes– pro level only.
9. Click on the *Record Position* icon. The current position will be recorded.

## Record Relative Position

To record a relative position, do the following:

1. Make sure you have first defined a reference position.
2. Move the robot to the position whose coordinates you want to record as relative to another position.
3. In the Position Number field in the Teach Positions (Simple) dialog box, enter a new position number.
4. Select Relative to, and enter a number (or select current) for the reference position in the Relative to field.
5. Click Record. You have now recorded a relative position.

A position that *is Relative to current* means the specified offset will be computed from the location wherever the robot is positioned at the time it is sent to the relative position.

If the reference position changes, the relative position also moves.

## Teaching Positions (XYZ Coordinates)

Teaching position in XYZ coordinates can be done only if the teach position window is expanded.

### Teach Absolute Position

### 2 & Pro

To teach an absolute XYZ position

1. Click *Expand*.
2. Enter XYZ Pitch and Roll values.
3. Click *Teach*

To use existing position coordinates, define a new position (or modify that position's coordinates), do the following:

1. In the Position Number field in the Teach Positions dialog box, select an absolute position number.
2. Click *Get Position*. The XYZPR values of the position now appear in the XYZ Pitch Roll fields. *If the position is relative only the offset values will be displayed.*
3. In the Position Number field, enter a different number (or leave the position number if you want to modify that position).
4. Select *Robot / Peripheral* (or both):
  - a) *Robot*: to record a position for the robot axes.
  - b) *Peripherals*: to record a position for the peripheral axes.
5. In one or more of the coordinate fields, enter a new value (in millimeters or degrees).
6. Click *Teach*

**Warning:** If you click *Record*, the current TCP coordinates will be written to the selected position.

## Teach Relative XYZ Position

## 2 & Pro

To record a relative XYZ position, do the following:

Make sure you have first recorded the reference position.

1. In the *Position Number* field, in the Teach Positions dialog box, enter the new position number.
2. Select *Relative to*, and enter the number of the reference position in the *Relative to* field. All XYZ coordinate fields are blank or show 0.
3. In one of more of the coordinate fields, enter a new value (in millimeters or degrees).
4. Click *Teach*.

A position that is relative to current means the specified offset will be computed from wherever the robot is located at the time it is sent to the relative position.

If the reference position changes, the relative position also moves.

# List Positions Window

## List Positions

The List Positions window displays the positions of the currently open project. The list is presented in a tabular format. As default, the table presents position information in Joints and Cartesian coordinates systems.

Every row in the table represents one position. To manipulate the list, select a position (row), and right-click to display the popup window.

#	Coor	Axis 1 X (mm)	Axis 2 Y (mm)	Axis 3 Z (mm)	Axis 4 Pitch (deg)	Axis 5 Roll (deg)	Axis 7 (mm/deg)	Axis 8 (mm/deg)	Type
1	Joint	0.00	-37.28	108.96	13.09	-0.11			Abs. (Joint)
	XYZ	276.99	0.00	129.26	-84.37	0.11			
2	Joint	0.00	-13.00	94.70	2.48	0.00			Abs. (Joint)
	XYZ	277.42	0.00	35.72	-84.27	0.00			
3	Joint	-39.72	-12.99	94.73	2.53	0.00			Abs. (Joint)
	XYZ	213.51	-177.40	35.71	-84.27	0.00			
4	Joint	-39.72	-34.33	107.32	11.24	0.00			Abs. (Joint)
	XYZ	213.62	-177.49	118.04	-84.23	0.00			
5	Joint	0.00	-120.26	95.03	88.78	0.00			Abs. (Joint)
	XYZ	169.13	0.00	994.27	-63.95	0.00			

Figure 5- 7: List Positions with Add Watch bar on the bottom

## Position Popup Window

Positions2	<i>Position No.</i>	The number/row of the position in the list
Delete	<i>Delete</i>	Deletes the selected position.
Delete All	<i>Delete All</i>	Deletes all listed position from SCORBASE memory
Add Watch	<i>Add watch</i>	Moves the position data to the watch list.
Show Joint	<i>Show joint</i>	Shows only joint values of <b>all</b> positions
Show XYZ	<i>Show XYZ</i>	Shows only the XYZPR of <b>all</b> positions
<input checked="" type="checkbox"/> Show Both	<i>Show both</i>	Shows XYZPR and Joint values of <b>all</b> positions

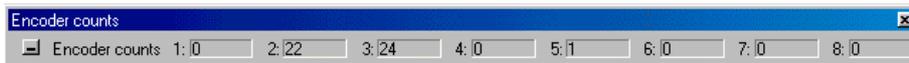
**Note:** The peripheral positions are always displayed using encoder counts.

## Dialog Bars

Scorbase offers three dialog bars that display the robot position in Cartesian (XYZ) coordinates, Joints Coordinates, and all axes encoder counts. All dialog bars are accessible at all levels, via the view menu.

### Encoder Counts Dialog Bar

The Encoders dialog bar displays the current values of the encoders for each of the eight axes.



*Figure 5- 8: Encoder dialog bar*

To display the Encoders dialog bar, select *View / Dialog Bar / Encoders*.

The encoder values change whenever the axes are moved.

These values are set to 0 (or close to zero) after the *Search Home* command is executed.

### XYZ Dialog Bar

The XYZ dialog bar displays the current Cartesian (XYZ PR) values of the TCP.



*Figure 5- 9: XYZ Dialog Bar.*

To display the XYZ dialog bar, select *View / Dialog Bar / XYZ*

The values shown in the above example are the XYZ values after the *Search Home* command is executed.

### Joints dialog bar.

The Joints dialog bar displays the angles between the two links of the joint, in degrees.



*Figure 5- 10: Joints Dialog Bar*

To display the Dialog bar, select *View / Dialog Bar / Joints*.

The values shown in the above example, are the XYZ values after the *Search Home* command is executed.

## Using a Teach Pendant with SCORBASE

The Teach Pendant is a hand-help terminal that permits the operator direct control of the robot and peripheral axes. In addition to controlling movement of the axes, the Teach Pendant may be used for recording positions, sending the axes to recorded positions, and other functions.

To control the axes from the Teach Pendant, SCORBASE must be operating in the *On-line* mode, and the Teach/Manual switch on the Teach Pendant must be switched to *Teach*. This disables control of the axes from SCORBASE dialog boxes.

All Teach Pendant operations will be reflected in the SCORBASE dialog boxes. For example, positions recorded by the Teach Pendant will appear in the Position Number list in the Teach Positions dialog box; encoder and XYZ values will change in the Encoder Counts and XYZ dialog boxes.

The Teach pendant operation is described fully in the Teach Pendant for Controller User's Manual.



## CHAPTER 6.

---

---

# 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 SCORBASE program.

The following tools are used for program editing:

- \* Program commands editor.
- \* Command tree that holds all SCORBASE commands.

## Opening and Closing a Program

Every SCORBASE program is part of a SCORBASE project. A project also includes the user-defined positions, project data and Cell Setup (if RoboCell is installed).

To open a saved program, open the project containing this program by doing one of the following:

- \* Select File | Open Project...
- \* Click on the Open an Existing Project icon .
- \* Press the CTRL + O keys.

In all cases, the *Load Project* window will open, prompting you to select the project that contains the program you want to edit.

To open a new program, open a new project by doing one of the following:

- \* Select File | New Project...
- \* Click on the Create a new project icon .
- \* Press the CTRL + N keys.

**Note:** Only one project can be opened at a time.

To activate the dialog boxes that are most useful for program editing, select *Window / Teach & Edit*. The screen layout that will appear depends on the currently selected SCORBASE level.

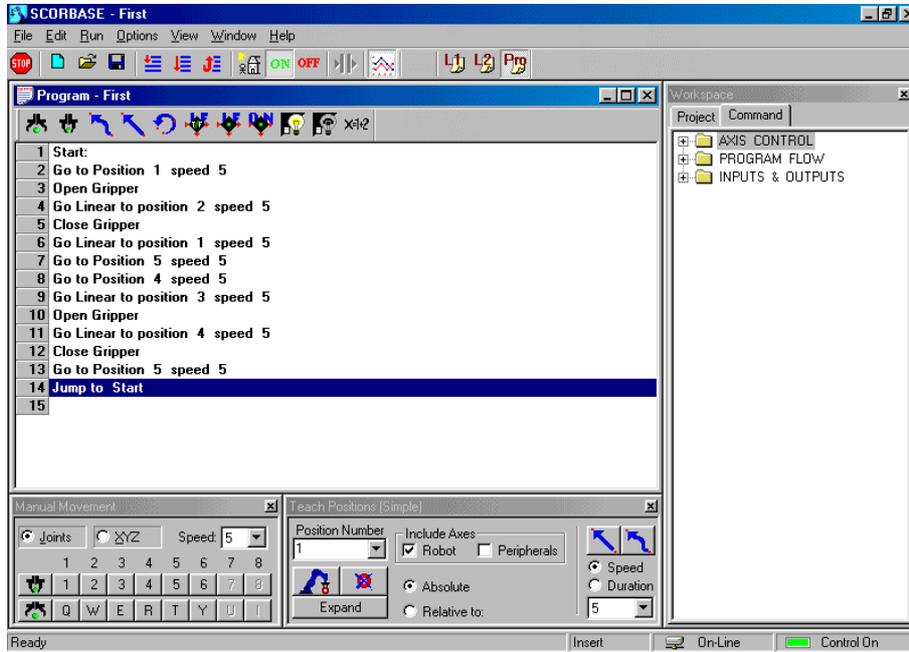
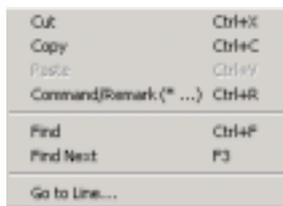


Figure 6- 1: Project Window - Pro Level

## Program Editing Tools

SCORBASE is a text based programming language, in which every command is a single text line. SCORBASE programs are edited by means of the usual Windows text editing tools, which can be accessed either by the Edit menu, pressing the designated keys, or opening the Popup menu with a right click when the mouse is in the program window.



*Cut - Ctrl+X*

Deletes selected text or lines from the program, and places it on the Windows and SCORBASE clipboards.

*Copy - Ctrl+C*

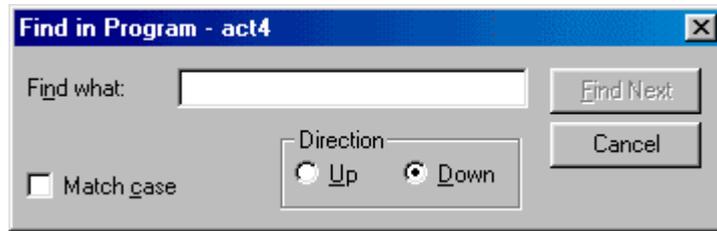
Places a copy of selected text or lines from the program on the Windows and SCORBASE clipboard.

*Paste - Ctrl+V*

Inserts the contents of the SCORBASE clipboard into the program.

*Find - Ctrl+F*

Opens a dialog box that allows you to search for any string, such as a command or text.



*Find Dialog Box*

Type the text you are searching for in the *Find what* field.

Select the searching direction (up/down).

Select Match case for a case-sensitive search.

*Find Next - (F3)*

Repeats the last Find operation for the next occurrence. (Accessible also from the Find dialog window)

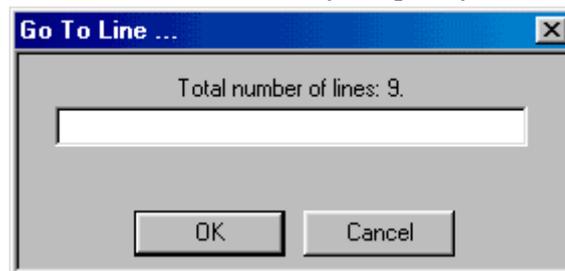
*Command/Remark (\*...)*

Inserts/deletes 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 that displays the total number of lines in the program and prompts you for a line number. The program editor will jump to the line you specify.



In addition, use the keyboard for the following functions:

[Ins]

Toggles between *Insert* and *Overwrite* modes. The active mode is shown in the Status bar.

When *Insert* is active, 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 following page of program lines.

## Adding and Editing Commands

SCORBASE commands are organized in a Command Tree. At the Introductory level (Level 1), only basic commands appear in the Command Tree. At the Advanced Level (Level 2), the number of commands is increased. At the Professional level (Pro level), all commands are accessible.

To see the Command Tree, click on the Command tab in the Workspace window.

SCORBASE commands are grouped into three categories (folders):

1. Axis & Control group
2. Program Flow group
3. Inputs & Outputs group

Clicking on a group (folder) opens and closes a list of the group's commands.

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

- \* Click on *Command* in the Command Tree.
- \* Type the two letters written next to the command.
- \* Click on the command icons in the Program window (applicable only for selected commands).

### **Notes:**

If *Insert* mode is selected, the new command line will be added above the currently selected (highlighted) line. If *Overwrite* mode is selected, the new command will replace the selected line

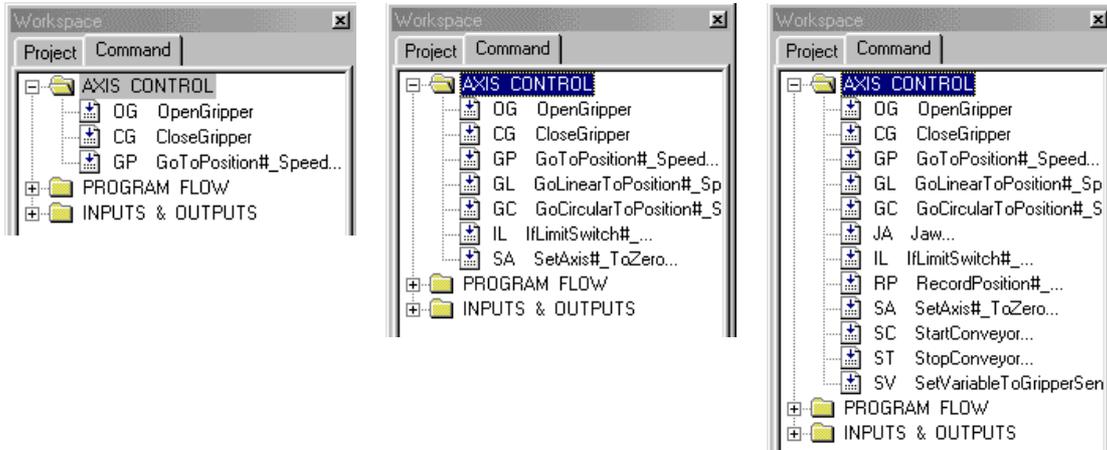
Many commands open dialog boxes for completing the command line parameters.

### ***Editing Commands:***

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

To delete, cut, copy and paste a line, use usual Windows tools.

# Axis Control Commands



**Level 1 – Introductory**

**Level 2 – Advanced**

**Pro - Professional**

*Figure 6- 2: Command tree - Axis Control*



**OG Open Gripper**

**Level 1, Level 2, Pro**

This commands fully opens the gripper.



**CG Close Gripper**

**Level 1, Level 2, Pro**

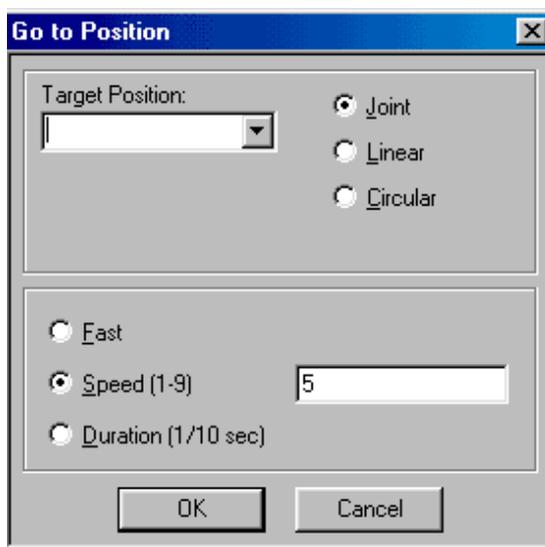
This command fully closes the gripper (on itself, or on a grasped object).



## GP Go to Position

Level 1, Level 2, Pro

Selecting the *Go to Position* command open the go to position dialog box.



The *Go to Position* 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 TCP trajectory. Required parameters are: Target Position, and Speed. The Movement control mode can be changed through this window.

*Target Position*      The destination of the movement.

Enter a number or a variable in this field.

### Select Movement Control mode.

- Joint*      Checked if window is opened from the *Go to Position* command.
- When *Joint* is selected, the robot moves to the target position in Point to Point control (no control over TCP trajectory).
- Linear*      Check to modify the *Go to Position* command, to *Go Linear to* (available only in Level 2 and pro).
- See the *Go Linear to* command for more details.
- Circular*      Check to modify the *Go to Position* command to *Go Circular to* (available only in Level 2 and Pro).
- See the *Go Circular to* command for more details.

### Select *Speed / Duration*

<i>Fast</i>	Executes the movement at the fastest speed possible.
<i>Speed</i>	Executes the movement at a slower speed. Enter a number from 1 through 9, or a variable, in the <i>Speed</i> field.  Default setting is 5 (average speed).
<i>Duration</i>	Executes the movement in a specific amount of time. Enter the time in tenths of a second, or a variable.  Available only in Pro Level.



#### GL Go Linear to Position #\_Speed ...

Level 2, Pro

The *Go Linear to Position* command 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.



#### GC Go Circular to Position #\_ Speed...

Level 2, Pro

The *Go Circular to Position* command 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.

Go to Position

Target Position:   Joint

Via Position:   Linear

Circular

Fast

Speed (1-9)

Duration (1/10 sec)

OK Cancel

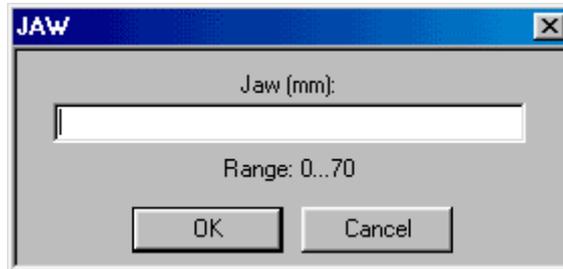
#### Note:

The *Circular* command follows the circle, that blocks the triangle defined by the three positions (Current TCP position and target position, and the intermediate position specified in the *Via position* field).

## JA Jaw

## Level 2, Pro

The *Jaw* command moves the gripper's jaw to the specified span. The command opens a dialog box.



Enter a number or a variable in the *Jaw* field.

*Jaw* activates Servo Control for the gripper motor, whereas *Open Gripper* and *Close Gripper* commands do not use the gripper axis Servo Control.

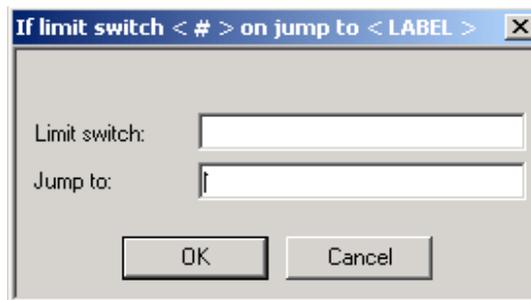
*Accuracy cannot be guaranteed if the width is less than 5 mm or greater than 65 mm.*

Unless you need the *Jaw* command for a specific application, the *Open Gripper* and *Close Gripper* commands are recommended.

## IL If Limit Switch < # > on jump to <Label>

## Level 2, Pro

The *IL* is a conditional jump command. It causes program execution to jump to the line that contains the specified Label, if the selected axis micro switch is pressed (On). The command opens a dialog box.



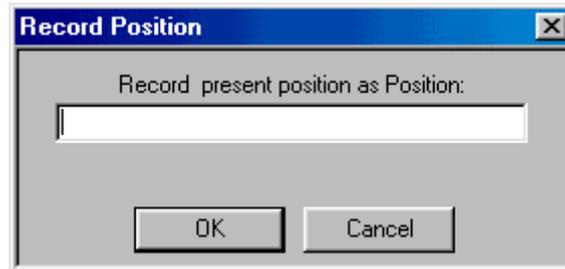
Enter the Axe number, or a variable in the *Limit Switch* field.

Enter the name of a Label in the *Jump to* field.

## RP Record Position #\_

Pro

When the *Record Position* command is executed (during program execution), the controller records the current position data to the specified position. The command opens a dialog box.



Enter a position number or a variable in the *Record Present Position as Position* field.

The *Record Position* command is useful when a position (and all relative positions that refer to that position) must be re-located, during program execution. This command updates the position data.

## SA Set Axis #\_ ... (to Zero)

Level 2, Pro

This command initializes (sets to 0) the encoder count of the selected axis. The command opens a dialog box.



Enter a number or a variable in the *Axis* field.

This command is useful for performing a cyclical motion of peripheral equipment such as a conveyor or a rotary table.

## SC Start Conveyor

Pro

The *Start Conveyor* command starts the conveyor, as a speed-controlled conveyor. Movement of the conveyor will continue until a *Stop Conveyor* (ST) command is encountered.

The command opens a dialog box.



Enter the Axis number in the *Axis* field

Enter a number or a variable in the *Speed* field.

Select a movement direction (Plus or Minus).

*When operating a speed controlled conveyor by means of the Start/Stop Conveyor commands, do not record positions and/or use the Set Axis (to Zero) command for the conveyor, in the same program.*

## ST Stop Conveyor

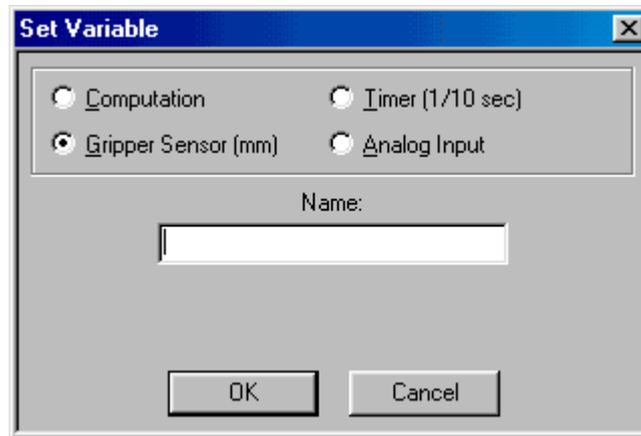
Pro

The *Stop Conveyor* command stops the continuous motion of a conveyor that was initiated by a *Start Conveyor* command.

## SV Set Variable to Sensor

Pro

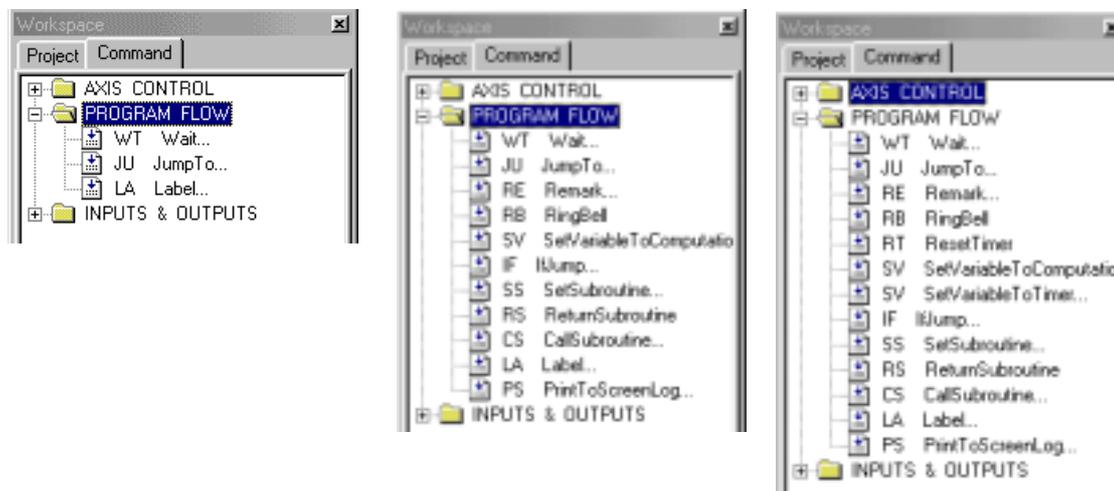
The *Set Variable to Sensor* command assigns the value of the *gripper* opening to a variable, in millimeters.



In the *Name* field, enter the name of the variable.

For more information on variables, refer to Chapter 7 - "Variable Programming."

## Program Flow Commands



**Level 1 – Introductory**

**Level 2 – Advanced**

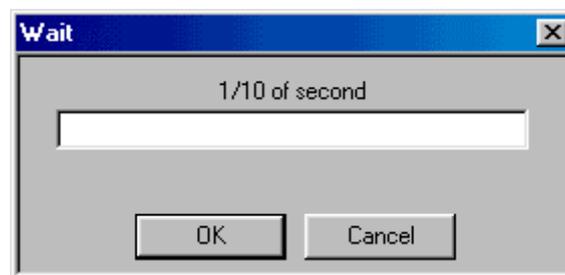
**Pro - Professional**

*Figure 6- 3: Command Tree - Program Flow branch*

**WT Wait (10ths of second)**

**Level 1, Level 2, Pro**

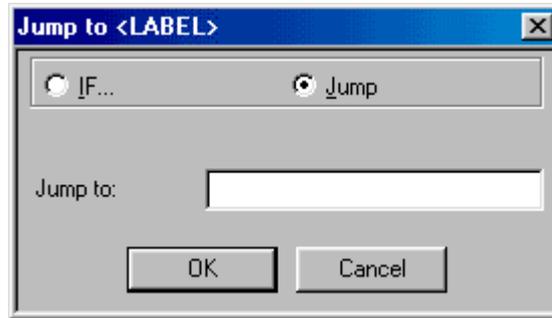
The *Wait* command halts program execution for a specified time. The command opens a dialog box.



Enter a number or a variable in the *10ths of Second* field.

**JU Jump to****Level 1, Level 2, Pro**

This is an unconditional jump command that causes the program pointer to jump to the line that contains the specified Label. The command opens a dialog box.

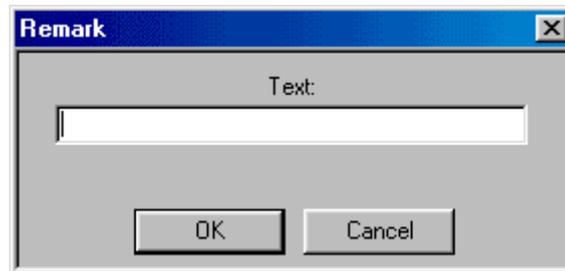


In the *Jump* to field, enter the Label name. (Be sure to include a line with this Label in your program.)

When *Jump* command is used, *Jump* is checked in the dialog box. You can modify the *Jump* command to *IF jump* (conditional jumping) only in Levels 2 and Pro.

**RE Remark****Level 2, Pro**

Allows inserting a comment line into the program, for explanation and documentation. The command opens a dialog box.



Enter up to 47 characters of text, including spaces.

**RB Ring Bell****Level 2, Pro**

When executed, this command produces a beep, using the computer's internal loudspeaker.

## RT Reset Timer

Pro

SCORBASE uses a timer that measures time in units of tenths of a second. Timer starts when SCORBASE is opened.

The *Reset Time* command resets the value of the SCORBASE timer to 0.

To use the timer, the timer value must be assigned to a variable, using the *Set Variable* command. (See *Set Variable* command for more details).

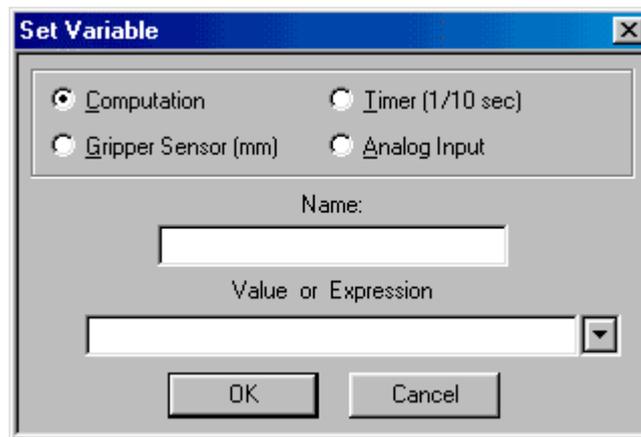


## SV Set Variable (to Computation)

Level 2, Pro

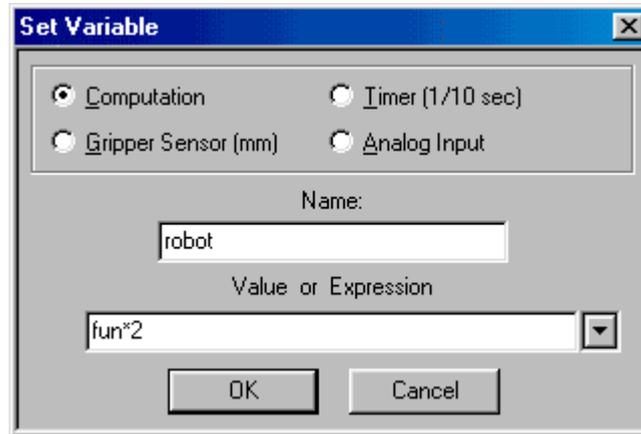
The *Set Variable to Computation* command allows you to assign a value, or an expression (result of a specific computation), to a variable.

Selecting *Set Variable to Computation* opens the *Set Variable* dialog box, with *Computation* checked.



In Level 2, the *Set Variable* value can be only a result of computation. In Pro Level, variable values can be derived from other sources, depending on the command selected. See description in the next command.

The *Set Variable* command allows you to assign the value of an integer, or the result of a computation, to a variable. The following example sets the value of a variable named “robot” to the value of the variable “fun”, multiplied by two.



In the *Name* field, enter the name of the variable. The first letter of the name must be a letter (such as "robot").

In the *Value or Expression* field, enter a value or a computation.

To set the variable to a fixed value, enter a number (in the range of +/-1000000), or enter a variable name.

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 (such as fun \* 2).

In the *Value or Expression* field, click on the arrow to see a list of operators, or use the list below.

#### ***Arithmetic Operators***

- + Addition
- \* Multiplication
- Subtraction
- / Division

#### ***Algebraic Operators***

- % Modulus (returns the remainder of the first argument divided by the second).
- \*\* Power (raises the first argument to the power of the second argument).

### ***Logical (Boolean) Operators***

&	And
	Or
^	Exclusive or

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.

### ***Comparison Operators***

<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to
<>	Not equal

By default, = is assumed to be the operator. You do not have to include it in the Value or Expression field.

### ***Examples:***

Set Variable COUNT = COUNT - 1

Set Variable C = A \* C

Set Variable DD = DD + 1

Set Variable POS = P >= 1

Set Variable V1 = V <> 1

Set Variable VAR\_A = A % 3

Set Variable M = M ^ 1

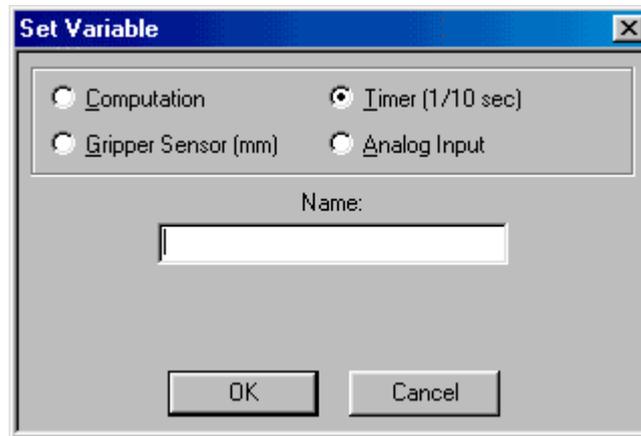
Set Variable R = 3 \*\* 2

For more information on variables, refer to the section Variable Programming at the end of this chapter.

## SV Set Variable (to Timer, Gripper Sensor or Analog Input)

Pro

At Pro level, there are four ways to set a variable value. Selecting *Set Variable to Timer* opens the following dialog box.



In the *Name* field, enter the name of the variable (*such as "robot"*).

To determine the source of the variable value, select one of the following:

*Computation* Variable value equals a constant, a variable, or a result of algebraic or Boolean operations

*Gripper sensor* Variable value equals the distance between jaws (in millimeters).

This command is useful when there is a need to measure the object in gripper, or to check gripper status (open closed or with an object).

*Timer* Variable value equals the current value of SCORBASE timer.

SCORBASE timer starts when SCORBASE is opened.

To initialize SCORBASE timer in a program, use the Reset Timer (RT) command

*Analog input* \* Variable value equals analog input value.

\* Enter the analog input number (1-4).

Analog input value is an integer in the range of 0 to 255, corresponding to 0 to 10 volt at the Controller input.

**Hint:**

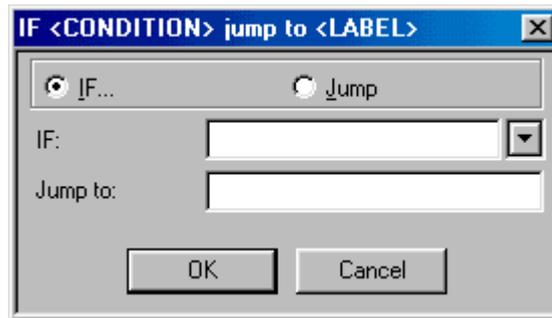
Value assigned using the *SV* command can be used for conditional jumping using the *IF* *<Condition>* *Jump* command. Actions can then be generated according to the variable value. The following example jumps the program cursor to a label if more than three seconds elapsed since timer reset.

- \* Reset Timer
- \* ...
- \* Set Variable TIME to timer
- \* IF TIME > 30 jump to PICKUP

For more information on variables, see Chapter 7, "*Variable Programming*".

**IF If Jump to****Level 2, Pro**

This is a conditional branch command, which is used to determine the program flow in relation to the value of the variables. The command opens a dialog box.



If the condition in the IF field is *true*, program execution will jump to the line specified by the label in the *Jump to* field.

If the condition in the IF field is *false*, program execution will continue to the following line.

In the IF field, enter the condition. The condition includes a variable name, a comparison operator and another variable name, or a number.

In the *Jump to* field, enter the name of a Label. (Be sure to include a line with this Label in your program.)

Example:

- \* If COUNTER > 0 jump to START\_LOOP
- \* Go to Position 1 speed 5
- \* ...
- \* START\_LOOP:
- \* Go to Position 2 speed 5

If the value of Counter is greater than zero, the robot will go to Position #1.

If the value of Counter equals or is less than zero, the robot will go to Position #2.

**Notes:**

Use two equal signs (= =) for equal operators. For example:

- \* If COUNTER = = 0 jump to END
- The *IF jump* command can be converted to unconditional jump command by selecting *Jump* (instead of *If*).

**SS Set Subroutine**

**Level 2, Pro**

The *Set Subroutine* command creates a subroutine. You can program up to 64 subroutines in one program. The command opens a dialog box.



In the *Name* field, enter a name or number for the subroutine.

**Note:** Create Subroutines only at the end of the main program.

**RS Return from Subroutine**

**Level 2, Pro**

The *Return from Subroutine* command marks the end of a subroutine. At run time, it terminates the execution of the subroutine, and the program resumes execution at the line that follows the *Call Subroutine* command.

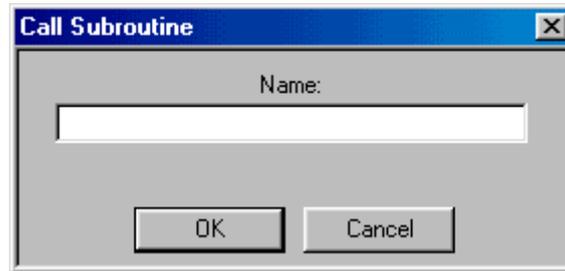
**Note:**

Every subroutine must end with *must* end with a return from subroutine command.

## CS Call Subroutine

Level 2, Pro

The *Call Subroutine* command activates the specified subroutine. The command opens the following dialog box.



Enter the name of the subroutine in the name field.

**Notes:** Use *Call subroutine* from the main program.

You can call the same subroutines several times from a few places in the program.

After the subroutine is executed, the program returns to the line that follows the *Call subroutine* command.

## LA Label

Level 1, Level 2, Pro

The *Label* command marks a line in the program that will be referenced by a *Jump* command. The command opens a dialog box.



In the *Label* field, enter a name.

**Notes:**

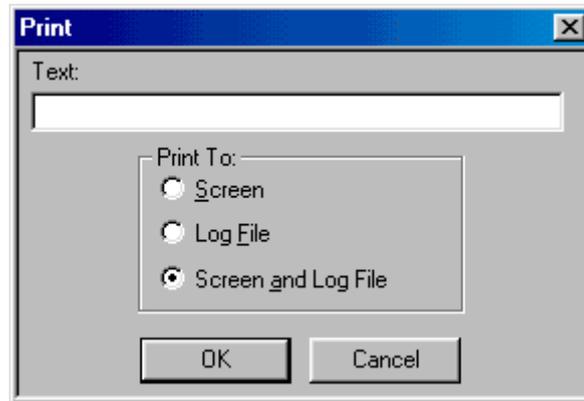
Do not include blank spaces; use an underscore.

Don't use the same label name more than once.

## PS Print to Screen & Log

## Level 2, Pro

The *Print* command instructs SCORBASE to print data containing strings, messages and variables value, to a log file, or to the message window (or both). The command opens a dialog box.

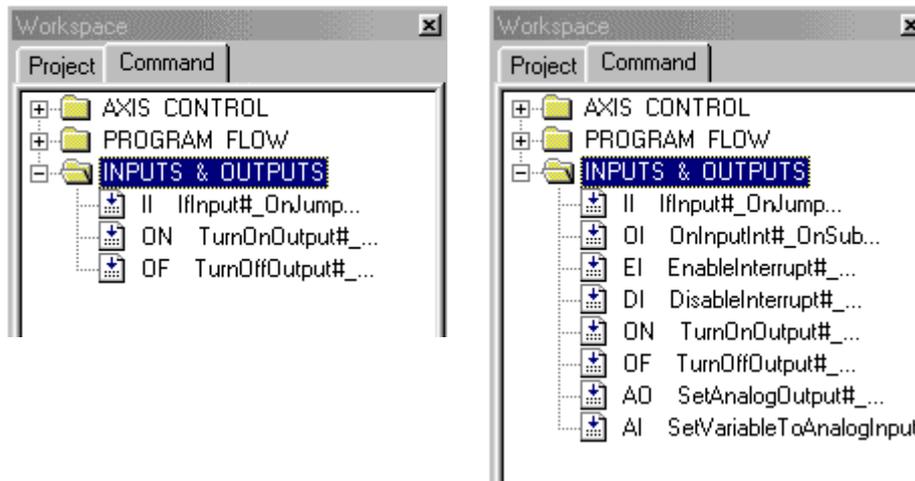


Enter text and spaces of up to 47 characters.

To print a value of a variable, place the variable name in single quote marks. For example: VARX='X' will print as VARX=50 (when the value of X is 50)..

Select: *Screen* / *Log File*, or *Screen & Log File*.

# Input/Output Commands



**Level 1 & 2 - Introductory & Advanced**

**Pro - Professional**

*Figure 6- 4: Command Tree -Inputs & Outputs branch*

## II If Input # On/Off Jump

**Level 1, Level 2, Pro**

The *If Input* command 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 a dialog box.



In the *Input Number* field, enter the number of digital input (1-8), or a variable.

Select either *On* or *Off* for the state of the input.

Select either *Jump* or *Call Subroutine*; then complete the active field.

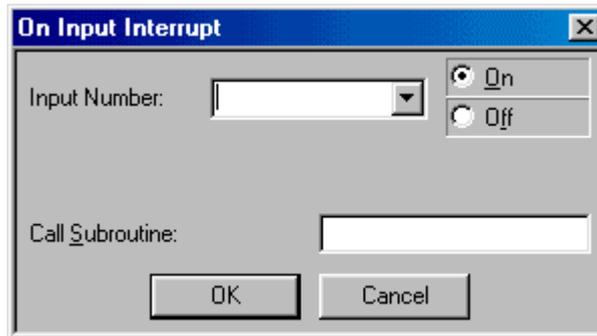
In the *Jump to* field, enter the name of a Label.

In the *Call Subroutine* field, enter the name of a subroutine, or a variable.

## OI On Input Interrupt # On/Off

Pro

This command sets the condition for an input interrupt service. The service (Call Subroutine) will be performed whenever the condition (input status) is satisfied, regardless of the current program pointer position. The command opens a dialog box.



In the *Input Number* field, enter number of a digital input, a variable or the word ANY.

Select the state of the input: *On/Off*

In the *Call Subroutine* field enter the name of a subroutine that will be executed in case of interrupt.

### Notes:

Using the word ANY in the *Input number* field will cause any input (1-8) to evoke interrupt state.

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 will resume from the point where it was suspended, as soon as the subroutine completes its execution.

An interrupt command can be disabled and enabled by means of the *Enable Interrupt* and *Disable Interrupt* command (see below).

Examples:

On Input Interrupt 5, on call sub. SUB5

When input 5 is turned on, the program immediately calls subroutine SUB5. If any axes are moving when the interrupt occurs, they will immediately stop. When the subroutine is completed (*Return from Subroutine* command is reached), the axes will reassume the position and status that were interrupted, and the program will continue from that point.

## DI Disable Interrupt #

Pro

The *Disable Interrupt* command causes the specified input interrupt to become inactive. When an interrupt is inactive, it is disregarded until the *Enable Interrupt* command reactivates it. The command opens a dialog box.



In the *Input Number* field, enter an input number, a variable, or the word ALL

Select *Enable* to modify the command.

## EI Enable Interrupt #

Pro

The *Enable Interrupt* command causes the specified input interrupt to become active. The command opens a dialog box:



When an interrupt is active, it is waiting for an interrupt status (defined using the on interrupt command).

In the *Input Number* field, enter an input number, a variable, or the word ALL

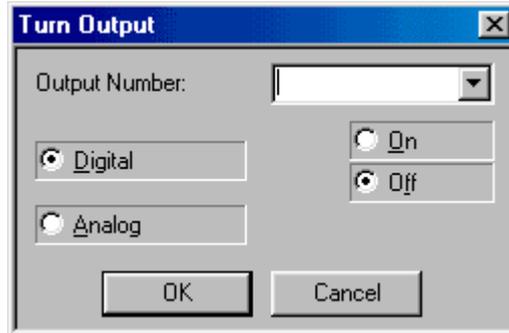
Select *Disable* to modify the command



### On Turn On Output #

Level 1, Level 2, Pro

This command sets the state of the specified digital output on.



In the *Output Number* field select a number or type a variable name.

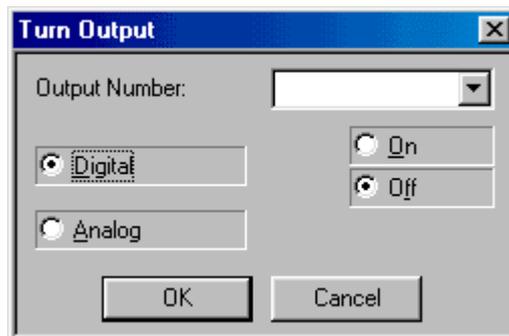
The dialog box opens with the *On* and the *Digital* options selected. The initial setting can be modified.



### Off Turn Off Output #

Level 1, Level 2, Pro

This command sets the state of the specified digital output to Off.



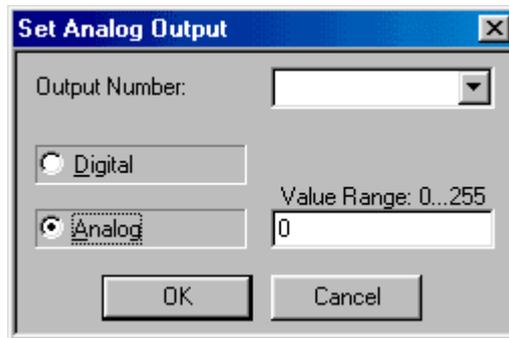
In the *Output Number* field, select a number or type a variable name.

The dialog box opens with the *Off* and the *Digital* options selected. The initial settings can be modified.

## AO Set Analog Output #

Pro

This command sets the state of the specified analog output. The command opens a dialog box



In the *Output Number* field, enter a number (1 or 2), or a variable.

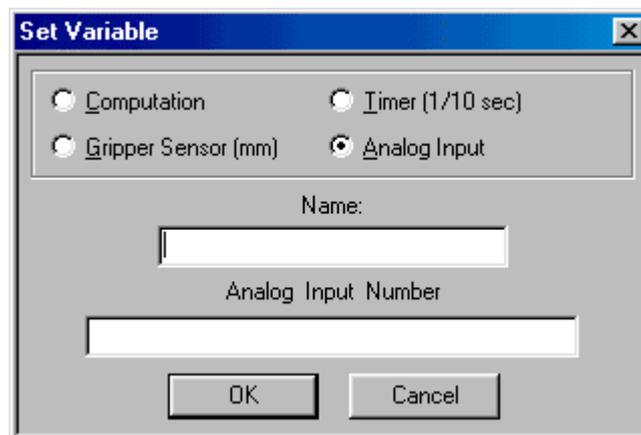
Type a number 0-255 in the value range field. The number corresponds to the output voltage range 0-10 volt.

The dialog box opens with the *Analog* option selected. The initial settings can be modified.

## AI Set Variable to Analog Input #

Pro

This command sets the value of the specified analog input to a variable. The command opens a dialog box



In the *Name* field type the variable name.

In the *Analog Input Number* type input number (1-4), or a variable.

The variable value will be an integer in the range of 0-255 corresponding to an input voltage of 0-10 volt.

Refer to the description of the *Set Variable* commands in the preceding section.

---

---

## Variable Programming

### Variable Programming

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

**Notes:**

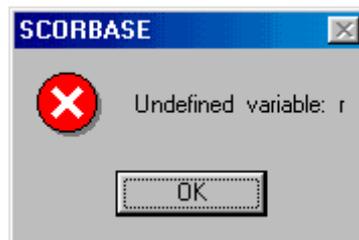
To use a variable, it must be defined by means of the *Set Variable* command. See the SV command for more details.

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 letter of the name must be an alphabetic letter.

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.



Undefined variable error message.

*Example #1* - Using a variable instead of a numeric value:

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

In Example #1, 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* - Using a variable instead of a numeric value:

```
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 Example #2, the robot moves to Pos. #1, #2, #3, #4 and #5. In every position the robot waits 50 seconds. After waiting at position #5, the program terminates.

*Example #3* - Using a variable instead of a numeric value:

```
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 Example #3, 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 turn the corresponding output **off**.

## Monitoring Variable Value

To monitor a variable value SCORBASE offers the following tools:

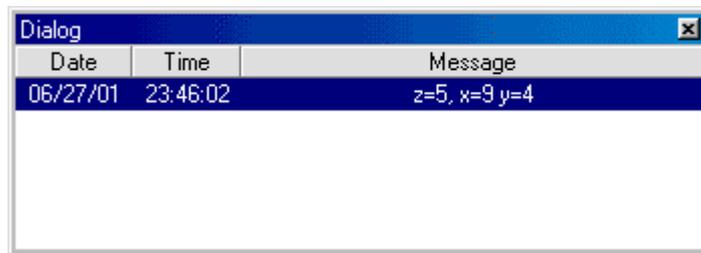
1. When SCORBASE is running and a *Set Variable* command is executed, the current value of the variable is displayed in the status line at the bottom of the screen.
2. When program is *not* running, click on the *Set Variable* command in which the value is assigned to the variable, and then execute this line using the *Run Single Line* command. The variable value is displayed in the status line at the bottom of the screen.
3. 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.

*Examples:*

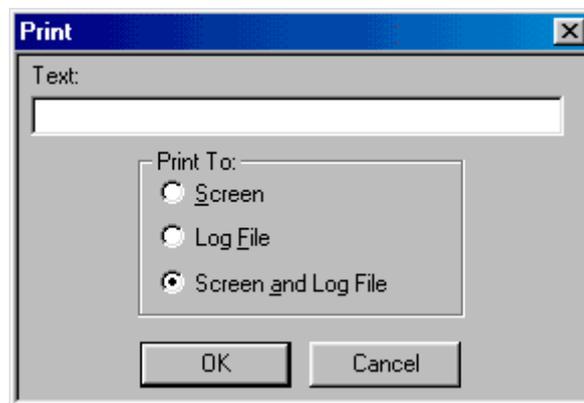
```
Set Variable z = 5
Set Variable x = 9
Set Variable y = x - z
Print to screen z='z' x='x' y='y'
```

When program is running the following messages will appear in the status bar:

1. When the first line is execute the message is z=5
2. When the second line is execute the message is x=9
3. When the third line is execute the message is y=4
4. When the forth line is executed the following window is opened:



The message window is opened when the *Print to Screen* command is executed either the *Print to screen* **or** *print to Screen and log file* radio button is selected.



To open the message window select *View | Messages*.

## CHAPTER 8.

# Program Execution

The following SCORBASE windows are used for activating and monitoring program execution.

Program window that shows section of the program currently executed.

- \* *Status bar* - shows the currently executed line or current value of a variable.
- \* *Dialog bars* - show the Analog Input, Analog Output, Digital Input, Digital Output values.
- \* *Log file* - records data during program execution.

To activate the windows which are most useful for program execution, select *Window / Run Screen*. The screen layout that will appear depends on the currently loaded level of SCORBASE, as shown in the following examples.

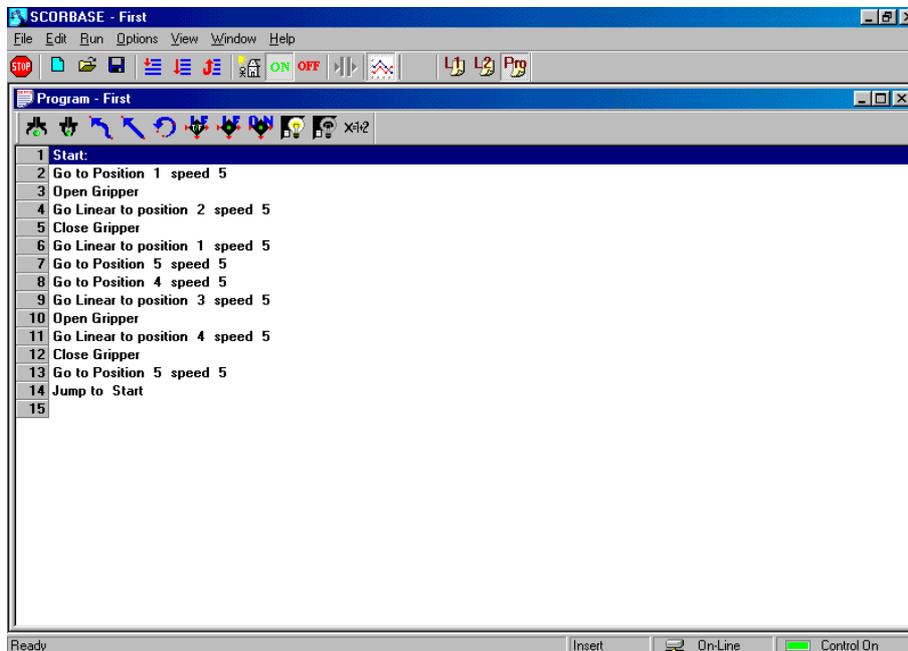


Figure 8- 1: Run Screen Pro level.

To activate the dialog bars that are most useful for program execution, select *View / Dialog bars / (select the dialog bar name)*.

# Running a Program

Scorbase offers three modes to run a program.

<i>Single line</i>		<b>F6</b>	Runs the currently selected (highlighted) line
<i>Single cycle</i>		<b>F7</b>	Runs the program from the currently selected (highlighted) line. Running stops after the last line is executed.
<i>Continuous</i>		<b>F8</b>	Runs the program from the currently selected (highlighted) line. After the last line is executed, program execution continues from the first line.

To select the running mode do one of the following:

- \* Click on one of the Run icons in the tool bar.
- \* Select one of the Run options in the Run Menu.
- \* Press the function key.

The [*Run*] key on the Teach pendant *cannot be used* to start execution of SCORBASE programs.

## Notes:

Always restart execution of a program from the first line after you have changed program data (e.g., recorded new coordinates for a position, edited a program line, etc.).

# Halting Program Execution

There are two ways to halt program execution from SCORBASE

<i>Stop</i>		<b>F9</b>	Program execution is stopped <b>immediately</b> . Use this command in case of emergency.
<i>Pause</i>		<b>F10</b>	Stops program execution only after the current command has been executed. Thus, axes may continue moving (to complete their motion) after the <i>Pause</i> command is issued.

To stop or pause programs from SCORBASE do one of the following:

Click on the *Stop / Pause* icon in the toolbar.

Select *Run / Stop*, or *Run / Pause*.

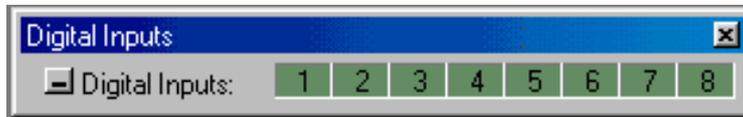
Press F9 / F10 - Make sure the SCORBASE application is the currently active window before you press F9./ F10

To STOP the program you can also press the red EMERGENCY button on the controller or -

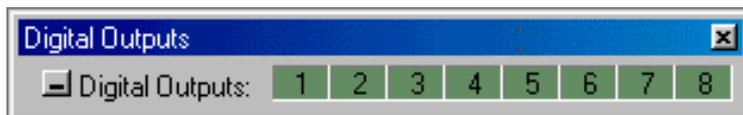
Press the ABORT key on the Teach Pendant (if installed).

## Digital Inputs & Outputs Dialog Bar

The Digital Inputs & Outputs dialog bars show the status of the controller digital inputs and outputs. The display is available in all modes of operation.



*Digital input dialog bar*



*Digital output dialog bar*

In the two dialog bars when the I/O status is *Off* (false), the matching I/O number is dark green and when the I/O status is *On* (true), the matching I/O number is light green

### Notes:

If a control is *On*, the sixteen Output / Input LEDS on the front panel reflect the I/O status.

If the control is *On*, clicking on an output numbers in the Digital Output dialog bar, toggles the status of the controller's digital outputs.

In *Off-line* the Digital Output dialog bar can also be used to simulate the status of controller's digital inputs. This option is useful for checking the command *If Input#\_onJump*.

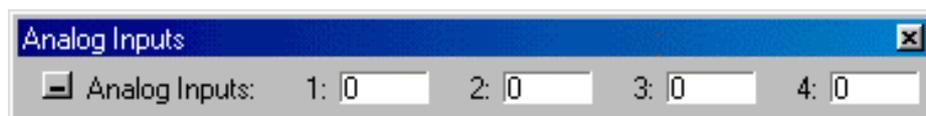
To test the program response for the *If Input#\_onJump* command, in *On-line* mode, short the designated input terminal to the digital input ground.

## Analog Inputs & Outputs Dialog Bars

SCORBASE can monitor and control four (4) analog inputs and two (2) analog outputs. The Analog Inputs & Outputs dialog bars show the value of the controller Analog Inputs and Outputs.



*Analog Output*



*Analog Input*

### Notes:

The Analog Inputs and Output resolution is 8-bit. The Analog Input / Output range is from 0 (minimum) to 255 (maximum).

Output values can only be manipulated when SCORBASE is operating On-line. To change the value of Analog Output, do one of the following:

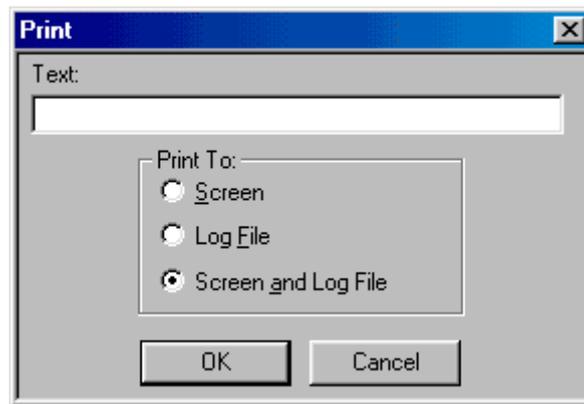
1. Click on *Output* (the color turn from light yellow to white), and type a number.
2. Use the AO (*Set Analog Output #*) command, and run this line.

When the controller receives an Analog Input signal from an external device, the value (0-255) of the signal will be reflected in the Input Value field.

## Log File

### SCORBASE Log

The SCORBASE 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 on the *Log file* or on the *Screen and log file* radio buttons.



*Print to screen command.*

**Notes:**

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

To see the file content, open the file SCBS.LOG using a text editor (such as word pad). The file is located in one of SCORBASE's subdirectories, named BIN.

SCORBASE generates a backup file each time the program is closed.



## Project Files Management

A SCORBASE project includes the following files:

- \* SCORBASE program (file extension \*.SBP)
- \* Recorded Position files (file extension \*.PNT)
- \* Project data (file extension \*.WS)
- \* Graphic image (if RoboCell is installed – file extension 3DC).

Opening or saving a project from the file menu opens or saves all project files (three or four files).

As default all files are located in the *Projects* directory (folder) in the ER 4U directory.

### Project Management

SCORBASE project files are managed by means of the usual Windows file tools, which can be accessed by icons or the File menu.

Description	Icon	Function
<i>New (Ctrl+N)</i>		Opens a new, untitled, project named <i>Untitled</i> . All project-related files are created.
<i>Open (Ctrl+O)</i>		Opens a Load Project File dialog box for opening a previously saved project. All project-related files are opened.
<i>Save (Ctrl+S)</i>		Saves the currently active project. If the project is untitled, a dialog box is opened for defining the project name.  All project-related files are saved.
<i>Save As...</i>		Opens the save project dialog box for saving the currently active project, under a new name.  All project-related files are saved under that new name.  As default all projects are saved in a projects folder.

<b>Description</b>	<b>Icon</b>	<b>Function</b>
<i>Print Program</i>		Prints the program. The program window must be active to select this option
<i>Print Positions</i>		Prints the position table. The position window must be open and active to select this option. You can open the position window by selecting Window   Project screen or from the workspace window.
<i>Print 3D image</i>		Prints the 3D image (if RoboCell program is installed).
<i>View File</i>		Opens the <i>View File</i> window to display the Program or Position of any selected project.
<i>Exit</i>		Quits SCORBASE. If changes to a program or position file have been made, but not yet saved, a warning message will be displayed.

# CHAPTER 10.

---

---

## System Setup

SCORBASE offers the following setup options to enable system configuration:

Display options.

Operation mode and Programming level.

Peripheral setup.

### Display Options

The display options enable optimal usage of the screen area. SCORBASE offers three basic display options and a wide range of dialog bars and windows through which the user can see and change system data.

### Simulation & Teach

Available only if RoboCell is installed. Refer to RoboCell user guide manual.

### Teach & Edit

When a SCORBASE project is opened, the screen is set for the Teach & Edit display mode. In this mode, the following windows and dialog boxes are displayed:

Program window that holds the SCORBASE program.

- \* Manual movement dialog box.
- \* Teach position dialog box
- \* Workspace window that shows:
  1. Project data (positions, user program and graphical display (if installed)).
  2. SCORBASE commands tree.

To open the Teach & Edit layout, select *Window | Teach & Edit*

## Run Screen

The *Run Screen* display option opens only the program window. When the program is running, the currently executed line is highlighted and information on the currently executed command is displayed in the status bar.

## Project Screen

The *Project Screen* display option holds the Program window and the Position window.

When the program is running, the currently executed line is highlighted, and information on the currently executed command is displayed in the status bar.

## Line Number

SCORBASE allows you to toggle the line numbers displayed in the Program window, On and Off. Select *Options / Line Numbers*.

By default, program line numbers are displayed.

## Reload Last Project at Start-up

When SCORBASE is initiated, the program can be set up to automatically open the last project. To toggle this option, select *Options / Reload Last Project at Startup*.

If a checkmark appears next to the *Reload Last Project at Startup* option, it indicates the option is on.

## Additional Display Options

If a checkmark appears next to the *Reload Last Project at Startup* option,

<i>Joint</i>	Shows angle of the robot joints.
<i>XYZ</i>	Shows position coordinates of the TCP
<i>Digital output</i>	Shows digital output status and enables toggling an output status.
<i>Digital input</i>	Shows the digital input status. Enables toggling an input status in <i>Off-line</i> mode, for program debugging.
<i>Analog output</i>	Displays the value (0-255) of Analog output 1 and 2. These values can also be modified through this dialog bar.

<i>Analog input</i>	Displays the value (0-255) of Analog inputs 1-4. Enables setting a value for an input in <i>Off-line</i> mode, for program debugging.
<i>Encoders</i>	Shows the eight encoders value.
All Dialog bars are accessible from the View menu.	
The screen is adjusted to prevent window and dialog bars overlapping.	
<i>Movement information</i>	Displays the following: <ul style="list-style-type: none"> <li>* Position error for all eight axes</li> <li>* Home switch status (for all eight axes). The number 1 indicates the switch is on (pressed) while 0 indicates the switch is off (released).</li> <li>* Selected axis (1 – 8) PWM value. The PWM value indicates the power sent to the axis motor.</li> </ul>
<i>Messages</i>	Displays the content of the PS ( <i>Print To Screen</i> ) commands.

## Scorbase Level

Scorbase offers three programming levels. They are:

1. Introductory (Level 1)
2. Advanced (Level 2) and
3. Professional (Pro).

A higher level offers more commands and tools. Selecting the level is done through the buttons in the Tool bar (the selected level is down), or from the Options menu (the selected level is checked).

## Peripheral Setup

SCORBASE allows you to define the devices that are connected and operated by the controller, as Axes 7 and 8. The following peripherals can be connected to the Controller USB .

- \* Rotary Table, 24V \_\_\_\_\_ Catalog #1009
- \* Conveyor Belt (gray), 24V \_\_\_\_\_ Catalog #1010
- \* 1.0m Linear Slidebase, belt-drive, 24V \_\_\_\_\_ Catalog #1020
- \* 1.0m Linear Slidebase, screw-drive, 24V \_\_\_\_\_ Catalog #1008
- \* 1.8m Linear Slidebase, belt-drive, 24V \_\_\_\_\_ Catalog #1021
- \* Linear Table 0.3m, 24V \_\_\_\_\_ Catalog #1013
- \* XY-Table, 24V \_\_\_\_\_ Catalog #1014
- \* Motor Kit 24V \_\_\_\_\_ Catalog #1234
- \* 48" Linear Slidebase, 24V \_\_\_\_\_ Catalog #1001
- \* 1.5m Linear Slidebase, screw-drive, 24V \_\_\_\_\_ Catalog #1007

### Note:

The following peripherals are not supported. To update, contact your local distributor:

- \* 1.0m Linear Slidebase, belt-drive, 24V \_\_\_\_\_ Catalog #1018
- \* 1.8m Linear Slidebase, belt-drive, 24V \_\_\_\_\_ Catalog #1019

Do not change the peripheral setup unless you are authorized to do so.

To define the devices, select *Options | Peripheral*.

A dialog box will open. Click on the arrow to open the list of available devices, and then click on the desired device.

