

# TwidoSuite V2.2

## Getting Started

05/2009



---

Schneider Electric assumes no responsibility for any errors that may appear in this document. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Schneider Electric.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

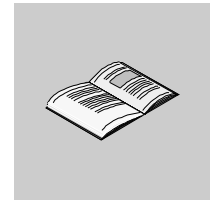
Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

© 2009 Schneider Electric. All rights reserved.

---

# Table of Contents



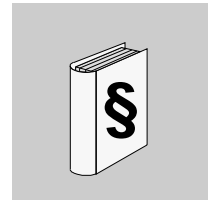
---

	<b>Safety Information</b> .....	<b>5</b>
	<b>About the Book</b> .....	<b>7</b>
<b>Chapter 1</b>	<b>Installation and Program Start</b> .....	<b>9</b>
	System Requirements .....	10
	Installing and Uninstalling the Software .....	11
	Registering TwidoSuite .....	12
<b>Chapter 2</b>	<b>Launching TwidoSuite</b> .....	<b>13</b>
	Program Start .....	13
<b>Chapter 3</b>	<b>Upgrading the Twido Firmware</b> .....	<b>15</b>
	Twido PLC Firmware Update .....	15
<b>Chapter 4</b>	<b>The TwidoSuite Interface</b> .....	<b>17</b>
	Main Window .....	18
	Setting Preferences .....	22
<b>Chapter 5</b>	<b>Creating and Editing a Program</b> .....	<b>23</b>
	Step 1: Create a New Project .....	25
	Step 2: Describe Application .....	27
	Step 3: Define Application I/O Wiring Specifications .....	30
	Step 4: Configure Hardware and Software Resources .....	32
	Step 5: Write the Program .....	37
	Step 6: Test the Program .....	46
	Step 7: Transfer and Run the Program .....	54
	Step 8: Debug the Program .....	57
	Step 9: Manage the Program Report .....	61
<b>Chapter 6</b>	<b>Monitoring an Application</b> .....	<b>63</b>
	Monitoring an Application .....	63
<b>Index</b>	.....	<b>65</b>



---

## Safety Information



---

### Important Information

#### NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

### **DANGER**

**DANGER** indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

### **WARNING**

**WARNING** indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

---

 **CAUTION**

**CAUTION** indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

**CAUTION**

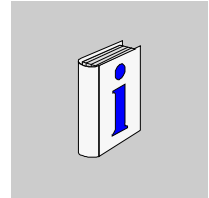
**CAUTION**, used without the safety alert symbol, indicates a potentially hazardous situation which, if not avoided, **can result in** equipment damage.

**PLEASE NOTE**

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

---

## About the Book



---

### At a Glance

#### Document Scope

This document provides basic information and instructions for getting the TwidoSuite programming software set up and operating.

#### Validity Note

This documentation is valid for TwidoSuite Version 2.2.

#### User Comments

We welcome your comments about this document. You can reach us by e-mail at [techcomm@schneider-electric.com](mailto:techcomm@schneider-electric.com).





---

# Installation and Program Start



---

## Overview

The TwidoSuite programming software is designed to run on various Windows 2000/XP/Vista operating systems. This chapter describes your computer system requirements to install and run this software. It also provides instructions for installing/uninstalling and starting the software.

## What's in this Chapter?

This chapter contains the following topics:

Topic	Page
System Requirements	10
Installing and Uninstalling the Software	11
Registering TwidoSuite	12

## System Requirements

### Hardware Requirements

Your PC needs to meet the following minimum hardware requirements to run the TwidoSuite programming software:

Requirement	Minimum
computer	processor Pentium 466 MHz
RAM	128 MB
free hard drive space	100 MB

A CD-ROM drive is the required installation medium.

### Software Requirements

The TwidoSuite programming software requires one of the following software operating systems:

Operating System	Edition/Service Pack	Special Considerations
Windows 2000	service pack 2 or above	For Windows 2000, Windows XP or Windows Vista, you need administration privileges to install the TwidoSuite configuration software.
Windows XP	service pack 2 or above	
Windows Vista	service pack 1 or above	

**NOTE:** Avoid patch 834707-SP1 (corrected by patch 890175) and patch 896358 which cause display problems with the Online Help.

The browser needs to be Microsoft Internet Explorer 4.01 or higher.

### Connection to a Twido PLC

The TwidoSuite programming software runs on a PC that connects to the RS-485 programming port on the Twido PLC. You need to make the connection with a special TSXPCX1031 programming serial cable, which ships with the software.

For more information about the required hardware connections, refer to the *Twido controllers hardware guide for compact and modular bases*.

## Installing and Uninstalling the Software


### Before You Start

Before you install the TwidoSuite programming software:

- uninstall any previous version of the TwidoSuite programming software (note that if you own a version of the TwidoSuite programming software, uninstall it as well for TwidoSuite is compatible with automation projects created with any version of TwidoSuite),
- close all Windows applications and virus-protection software.

### Installing the Software

To install the TwidoSuite programming software:

Step	Action	Result
1	Insert the TwidoSuite CD in your PC's CD-ROM drive.	If the AUTORUN function is activated, the installation will start automatically, then go to step 4. Otherwise go to step 2.
2	If the installation does not start automatically, click <b>Start</b> → <b>Run</b> .	The <b>Run</b> dialog box will appear.
3	Enter [Drive:]\setup.exe, then click <b>OK</b> .	The language selection screen appears.
4	Choose a language and click <b>OK</b> .	A message appears asking you if you want to register your copy of the software.
5	To register the software, click <b>Yes</b> . To run the software without registering it, click <b>No</b> .	An unregistered software package will run for 30 days, then stop running. You will need to uninstall the expired software before installing a new version. The new installation must be registered before it will work.
6	Choose the installation language among the list of available languages.	The software will be installed in both the selected local language and in the English version. You will later be able to execute the software in either of these languages by selecting the execution language from the TwidoSuite Application Launcher.
7	Follow the remaining on-screen installation instructions.	Once the software has been installed, an icon appears on your desktop, which you can double-click to start the software:  TwidoSuite

### Uninstalling the Software

To uninstall the TwidoSuite programming software from your computer, choose the following path: **Start** → **Control Panel** → **Add/Remove Programs**

## Registering TwidoSuite


### Introduction

Take a few minutes to register this software and become a member of the Schneider automation community.

Registering is free and it will keep you updated with the latest product news, software and firmware upgrades for your Twido controller.

**NOTE:** There is a 30-day trial period for this software. After expiry of the trial period, you **must** register to be able to continue using this software.

### How to Register TwidoSuite

1. Go to **Miscellaneous** .
2. Click **About** on the task bar.
3. Complete the **License** form with "Company", "User" and "Serial Number" information, and press **Now**.  
**Result:** This starts the **TwidoSuite Registration Wizard**.
4. Follow the onscreen instructions to register your copy of the TwidoSuite software.  
**Note:** There are four possible ways to register: Web, Phone, Fax, and E-mail.
5. Read the Terms of Service, acceptance is required.
6. Submit the registration.

---

# Launching TwidoSuite

# 2

---

## Program Start

### How to Start the Program

You can either start the program via the **TwidoSuite Application Launcher**:

- from the **Start → Programs → Schneider Electric → TwidoSuite → Application Launcher** path
- or from the desktop Application Launcher shortcut that has been installed with the program:



### Application Launcher Options

The Application Launcher constitutes the launch center for the TwidoSuite software suite.

The TwidoSuite application comes in several language pairs available from the Schneider Electric download site:

- English-French
- English-German
- English-Spanish
- English-Italian
- English-Chinese(simplified)



Depending on which language pair you installed, it lets you run TwidoSuite in English (default language) or the other installation language.

Use the Application Launcher to start TwidoSuite in:

- **Programming Mode**
- **Monitoring Mode**
- **PLC Firmware Update**

## Additional Features

Use the additional features to:

- see **What's new?**  ,
- view the **Video Demo**  .

**NOTE:** By default the Video Demo is shown at TwidoSuite startup. To disable this feature uncheck the box "Show video at startup" at the bottom of the video screen.

## TwidoSuite Application Launcher

The following figure shows an example of the TwidoSuite Application Launcher for the English-French installed language pair:



---

# Upgrading the Twido Firmware

# 3

---

## Twido PLC Firmware Update

### Overview

**NOTE:** The Schneider Electric Twido PLC is delivered with firmware installed. The procedures described for loading new firmware are only required if an update is being made. Before replacing the firmware, you should carefully study the corresponding instructions and contact the Schneider Electric support center.

When the TwidoSuite software is connected to the PLC, the firmware version that you are currently using for your Twido PLC can be found via the **Miscellaneous** button:



The **Twido Windows Executive Loader Wizard** is a Windows-based program that guides you through the steps needed to update the **Firmware Executive** in your Schneider Electric Twido programmable controller. The Firmware Executive is the operating system that runs your applications and manages controller operation.

The wizard provides an intuitive, point-and-click user interface that guides you through the steps of the update procedure. It provides directions to quickly download the latest version of the executive for your controller.

See the *Executive Loader Wizard User Guide* for detailed instructions. This guide can be accessed using the Exec Loader utility online help.

### Opening the Exec Loader Wizard

You can open the Executive Loader Wizard from the TwidoSuite Application Launcher by selecting the **PLC Firmware Update** option (see Application Launcher Options ).

**New Firmware (Basic Steps)**

There are four steps required to upgrade your executive. The wizard provides a window for each step and online help is available for each step. The following table summarizes each step:

Step	Window	Function
1	Welcome	Introduces the Executive Loader Wizard.
2	Parameters	Selects the correct executive file to transfer to your controller, and selects the PC serial port to communicate with the controller.
3	File and Device Properties	Compares the hardware IDs and the executive version information for the executive file and the controller.
4	Transfer Progress	Monitors the transfer of the executive file to the controller.

**NOTE:** Loading the firmware may not be interrupted under any circumstances as it means the module can no longer be accessed, including via the Executive Loader Wizard. Ensure that neither the communication connection nor the main power input is disconnected or deactivated in any way and do not deactivate the procedure with **Abort**.



---

# The TwidoSuite Interface

# 4

---

## Overview

This section describes the TwidoSuite:

- main window workspace
- preferences settings

This section also provides a work flow for creating a TwidoSuite project (see *Application Steps/Substeps Bar, page 20*).

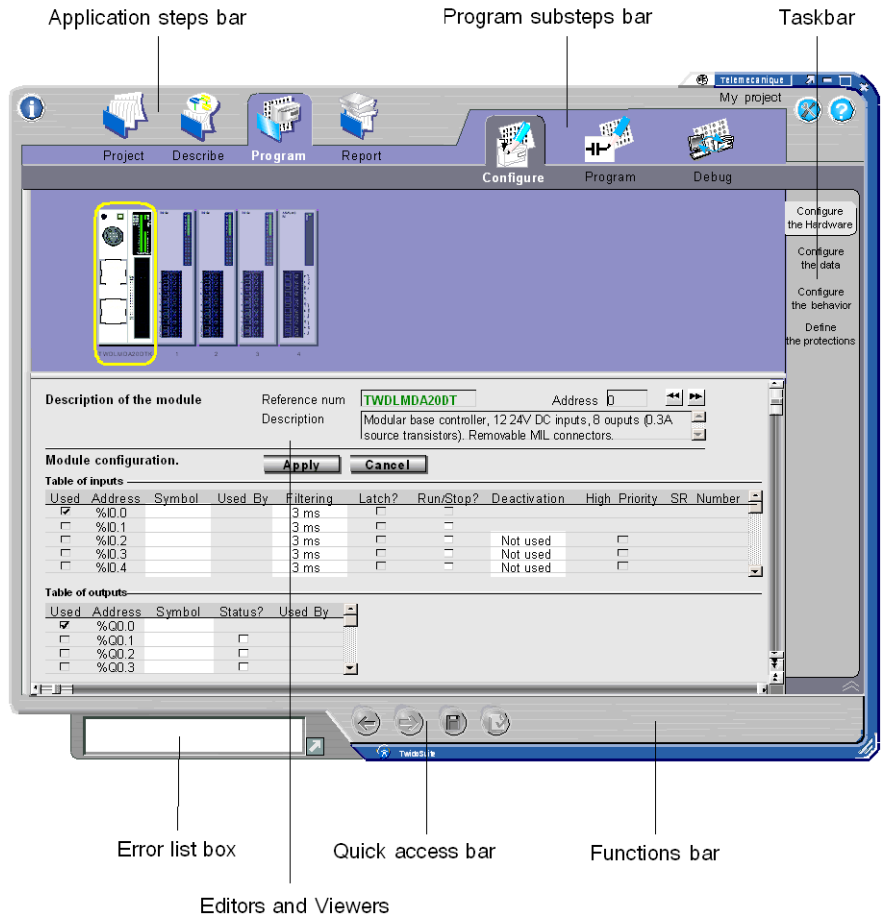
## What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Main Window	18
Setting Preferences	22






## Main Window

### TwidoSuite Workspace










<b>Application steps bar:</b>	Displays TwidoSuite's 4 Application Steps (icons and titles), Project Name, Preference, Miscellaneous and Help Pages buttons.
<b>Program substeps bar:</b>	Displays the 3 Program substeps (Configure, Program, Debug). It is showing only when the Program step is selected.
<b>Taskbar:</b>	Provides access to all the tasks that you can perform under the selected application step or substep.
<b>Functions bar:</b>	Provides access to special functions associated to the selected task.
<b>Quick access bar:</b>	Displays at all times Previous/Next commands, Save and Analyze Program shortcuts.
<b>Editors and Viewers:</b>	Editors and viewers are TwidoSuite windows that organize programming and configuration controls for efficient development of applications.
<b>Error list box bar:</b>	Displays information about the application, the controller, and TwidoSuite.






## Display Options

	Maximizes the main window
	Minimizes the main window
	Closes the TwidoSuite application
	Docking button for the Application Steps bar
	Resizes the main window

## Application Steps/Substeps Bar

	<p><b>Project</b> This is the first step in creating/opening an application. The Application Steps/Substeps Bar displays the name of the open application in the top-right corner.</p>
	<p><b>Describe</b> This is the second step in building the Twido automation application. The Describe step lets you select hardware resources from the catalog and describe how equipment interconnect with one-another via networks.</p>
	<p><b>Program</b> This is the third step in setting up your Twido application. It provides access to the following 3 substeps.</p> <p><b>1.Configure</b> This substep lets you view/set:</p> <ul style="list-style-type: none"> <li>● <b>Hardware Configuration</b> for the base controller, expansion modules, communication modules and hardware options</li> <li>● <b>Data Objects Configuration</b> for Simple Objects, Function Blocks, I/O Objects, and Professional Objects</li> <li>● <b>Controller Behavior</b></li> <li>● <b>Application Program / Password Protections</b></li> </ul> <p><b>2.Program</b> Opens the default <b>Program Editor</b> (Ladder or List) and provides access to the Symbol Editor.</p> <p><b>3.Debug</b> Lets you manage the <b>TwidoSuite-to-PLC connection</b> and provides access to <b>Animation Tables Editor</b> and <b>Monitoring</b> tasks.</p>
	<p><b>Report</b> This is the final step in setting up your application. It opens the <b>Report Manager</b> window to select report components, configure printing options, and export the application report to HTML format.</p>
	<p><b>Miscellaneous</b> Provides access <b>General Help</b> and information <b>About TwidoSuite</b>.</p>
	<p><b>Preference</b> Opens the <b>User Preferences</b> window.</p>
	<p>Shows the <b>Task-level Quick Help</b> pages</p>

## Quick Access Bar

	Return to <b>previous</b> task, as you left it
	Move back to <b>next</b> task, as you left it
	<b>Save</b> current project
	<b>Analyze</b> Program
	Launches and stops the <b>Simulator</b> .

## Setting Preferences

### Displaying Preferences

Click the **Preference** icon  on the Steps bar to display the Preferences window as shown below.

### Preferences Window

Preferences Apply Restore

**Default project directory** Directoly:  ...

**Default projects**

None

Schneider default

Customized  ...

**Default program editor**

Ladder

List

**Autosave project**

Yes every  minutes

No

**Default background color**

Clear Dark

**Default project image**

Default image

Personalized image  ...

**Default functional levels of applications**

Automatic **Automatic management**

The very highest

The very lowest

Manual **Manual management**

**Connection management**

Name	Connection type	IP / Phone	Punt / Address	Baud rate	Parity	Stop bits	Timeout	Break timeout
COM1	Serial	COM1	Punt				5000	20
COM2	Serial	COM1	Punt				5000	20
My connection 1	Serial	192.168.1.1	Direct				5000	20
Modem connection	Ethernet	xxxxxxxxxxx		19200	None	1	5000	20

Add Modify Delete

### Customize Project Preferences

- Set the **default project directory** path
- Set the **default/customize projects**
- Set the **default program editor** (Ladder or List)
- Set the **autosave** options
- Set the **background color**
- Set the **default/custom project image**
- Select the **automatic/manual functional level** management method to implement at project start
- Add/update/remove connections by using the **connections management** table

---


# Creating and Editing a Program

# 5

---

## Overview

In this section, you will learn how to create and edit a program with TwidoSuite.

To start TwidoSuite you need open the Application Launcher. Click the  icon located on your PC desktop.

The Application Launcher dialog box appears as shown below:



From TwidoSuite Application Launcher, you have access to the following modes:

- TwidoSuite Programming Mode
- TwidoSuite Monitoring Mode
- PLC Firmware Update utility

### What's in this Chapter?


This chapter contains the following topics:

<b>Topic</b>	<b>Page</b>
Step 1: Create a New Project	25
Step 2: Describe Application	27
Step 3: Define Application I/O Wiring Specifications	30
Step 4: Configure Hardware and Software Resources	32
Step 5: Write the Program	37
Step 6: Test the Program	46
Step 7: Transfer and Run the Program	54
Step 8: Debug the Program	57
Step 9: Manage the Program Report	61

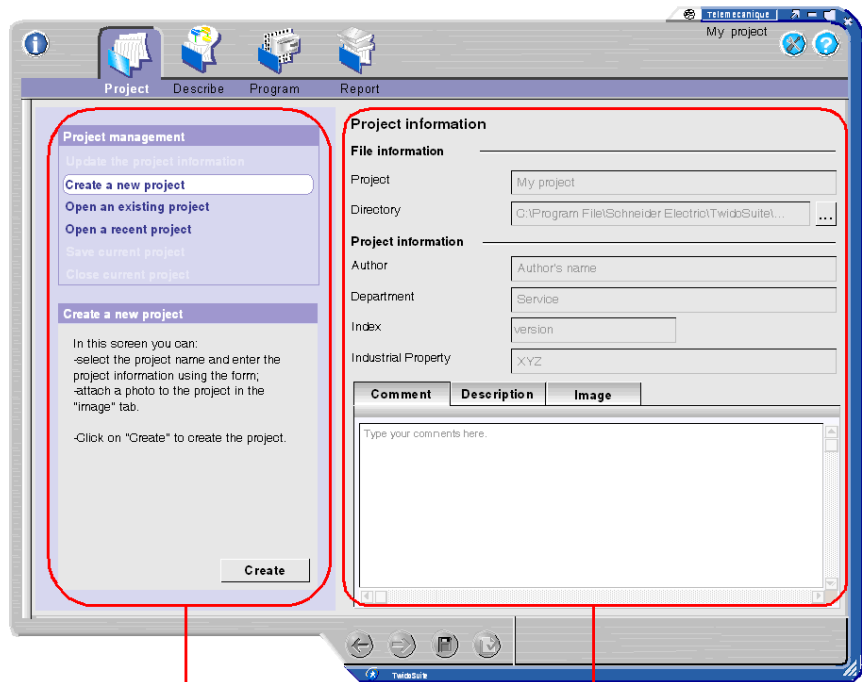


## Step 1: Create a New Project

### How to Start the Project Manager

Click the **Project** icon  on the TwidoSuite workspace Steps bar to display the Project window and create/open/edit a TwidoSuite project.

### Project Window



Actions frame


Project information main pane

## How to Create a New Project

1. Click **Create a new project** in the "Project Management Actions Pane."
2. Enter a descriptive name for the project in the **File Information** frame. A valid project name can contain up to 36 alphanumeric characters (spaces allowed).
3. Specify the new path for the **project directory** or use the default path. In the **Directory** text box, either enter directly the storage directory path or click the Browse button located immediately to the right of the Directory text box to locate the folder where the project files will be stored.
4. Enter project authoring information in the **Project Information** frame. This includes providing **Author's** name, **Department** name, project version number in **Index** field and **Industrial Property**.
5. In the **Comments** tab, include the name of the project and write a brief description. Use the right scrollbar to read the complete text.
6. If you are updating an existing project, click the **Description** tab to view a thumbnail representation of the system architecture (see *Step 2: Describe Application, page 27*).
7. Select the **Image** tab and click **Modify** to customize the project picture, or use the default picture. The project picture is printed on the front cover of the project report (see *Step 9: Manage the Program Report, page 61*). A valid picture file must be in the .BMP format.
8. Click **Create** to save your changes to the new project file.  
**Note:** If you are updating information of an existing project, click **Apply** to save your changes.

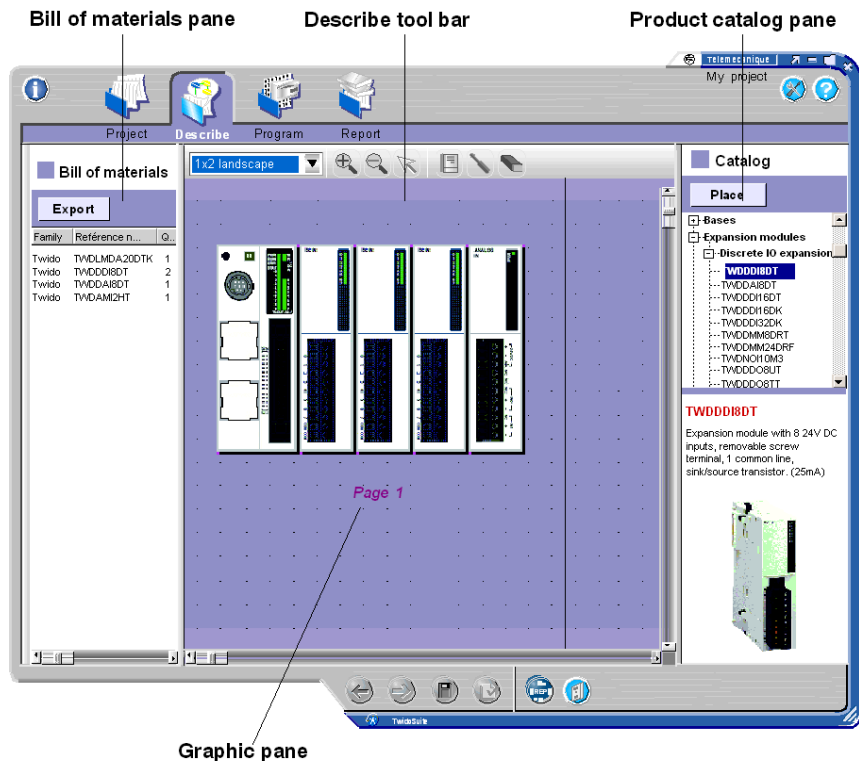
## Step 2: Describe Application

### How to Start the Describe Feature

Click the **Describe** icon  on the TwidoSuite workspace Steps bar to display the Describe window.

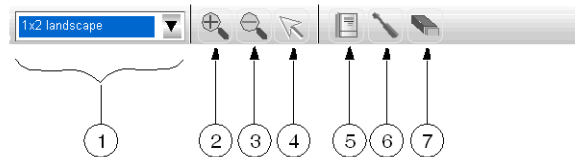
### Describe Window

In the Describe window, you declare the Twido automation system elements selecting hardware resources from the catalog and describing how equipment interconnect with one-another via networks.



Elements of the Describe window:

- **Graphic Pane:** This is where you build the graphical representation of your application by adding hardware elements. Drag selected hardware elements from the catalog pane to the graphic pane to gradually construct your automation system. You use the graphic pane to declare networks and how equipment interconnect with one another, as well.
- **Describe tool bar:** Use the Describe tool bar to manage the graphic pane and elements.



- Select page format
  - Zoom in
  - Zoom out
  - Select graphic element
  - Add a note
  - Configure element
  - Erase element
- **Product catalog pane:** This dockable pane displays the Twido hardware elements catalog, including compact base and modular controllers, expansion modules, options, network elements in a tree-like structure. The bottom part of the pane provides a short description of the selected hardware element.
  - **Bill of materials pane:** This dockable pane shows a bill of materials of the hardware elements that currently make up the open project. You can conveniently click **Export** to save the bill of materials to a file in .CSV format.

### How to Describe Your Application

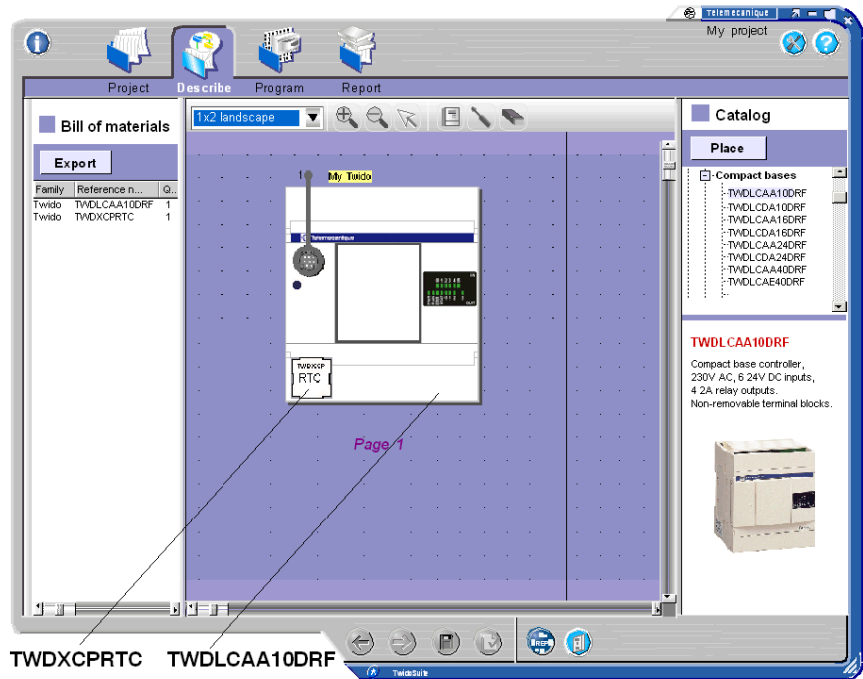
1. Use drag-and-drop to place catalog elements in the graphic pane, starting with the base controller (compact or modular), then adding expansion modules and options as supported by the controller and as required by your application.
2. Configure communication ports by double-clicking the ports graphical representation in the graphic pane.
3. Use drag-and-drop to place network elements available from the catalog pane in the graphic pane.
4. Click to select a Twido communication port and drag the cursor to draw the network links between the Twido base controller and network elements. Connect network elements with one another in the same manner to build network links.

**NOTE:** To find out more about describing applications, declaring network elements and configuring networks, please refer to the *TwidoSuite Operation Guide - Online Help*.

## Automatic-Watering System Example

In this example, you will need to:

1. place a TWDLC•A10DRF 10 I/O compact base controller in the Describe window and
2. add the RTC (real-time clock) option to the TWDLC•A10DRF controller in the Describe window, as shown in the figure below.



## Step 3: Define Application I/O Wiring Specifications

### Introduction

Before you proceed with programming of your application, you must set your application I/O wiring specifications.

### I/O Wiring Specifications

It is good practice to draw up a small sketch of your application I/O wiring.

The following are I/O specifications for the automatic-watering system:

#### Input specifications

- **%I0.0** input from water **high** level sensor in water tank
- **%I0.1** input from water **low** level sensor in water tank

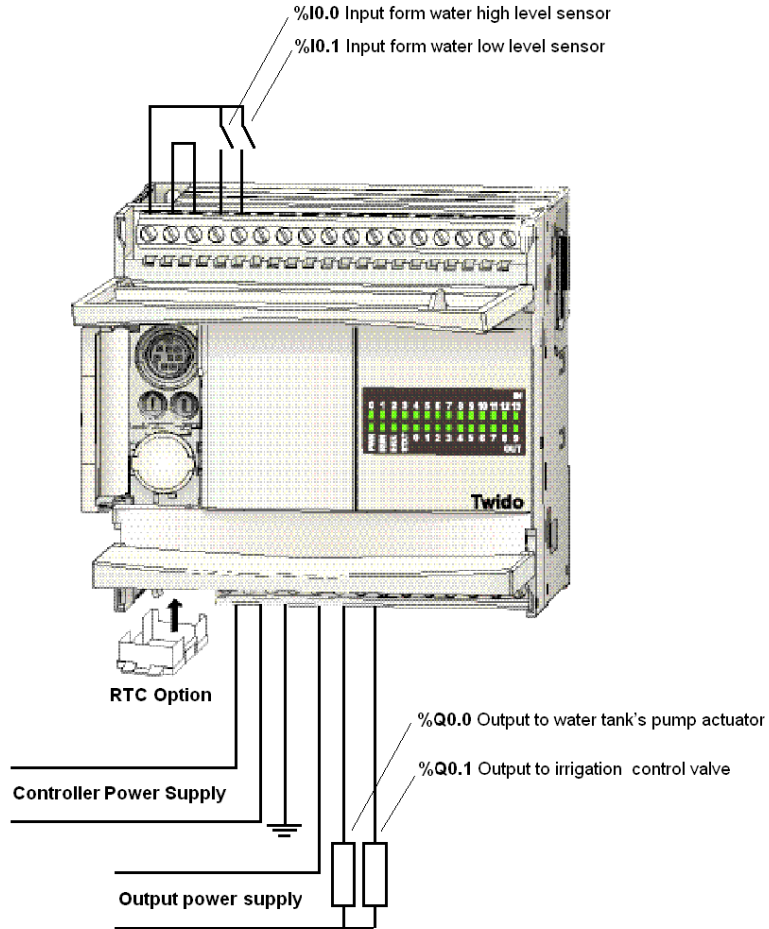
#### Output specifications

- **%Q0.0** output to water pump actuator
- **%Q0.1** output to irrigation control valve

#### RTC Specifications

- **RTC** option must be installed on Twido 10 I/O base controller.

I/O wiring specifications are outlined in the following wiring sketch:





## Step 4: Configure Hardware and Software Resources

### Introduction

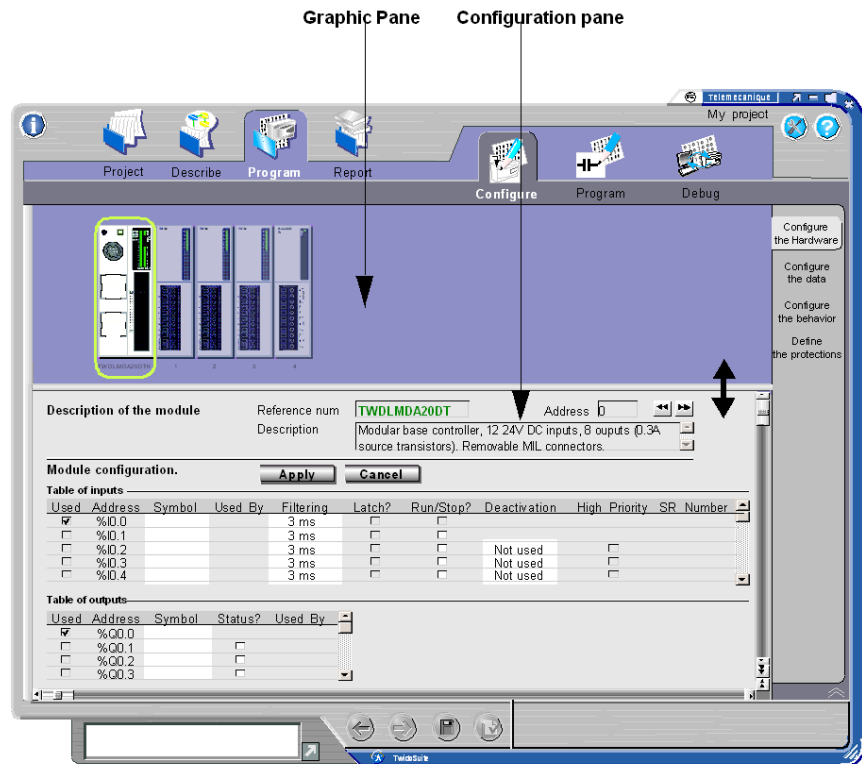
Now that you have declared the hardware elements in the Describe step, you will need to configure hardware and software resources.

To have access to hardware and software configuration options, select

**Program**  → **Configure**  on the TwidoSuite Steps bar.

### Configure Hardware Resources

Select the **Program** → **Configure** → **Configure Hardware** task to open the hardware configuration window.



**Graphic Pane:** Picture of the controller and its expansions

**Configuration Pane:** Description (with configuration) of the currently selected hardware elements



Configurable hardware resources include:

- Base and remote controllers discrete inputs and outputs
- Expansion I/O modules parameters, such as analog modules
- AS-Interface module and slave devices parameters
- CANopen master module and slave devices parameters
- Options parameters

### **Automatic-Watering System Example**

In this application example, you will find it useful to symbolize inputs and outputs (see *Step 3: Define Application I/O Wiring Specifications, page 30*).

To configure symbols for I/Os, follow these steps:

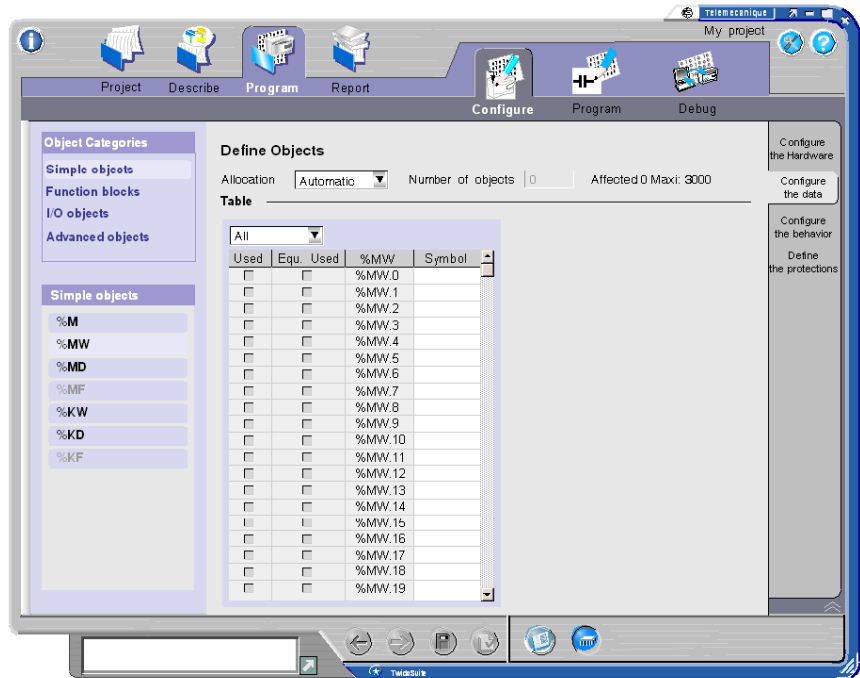
1. Select the **Program** → **Configure** → **Configure Hardware** task to open the hardware configuration window.
2. Click once on the controller picture to select the TWDLCAA10DRF controller in the graphic pane.
3. In the configuration pane, enter symbolic names for the I/Os you want to symbolize, for example:
  - WATER\_HIGH\_LEVEL, in %I0.0 symbol field
  - WATER\_LOW\_LEVEL, in %I0.1 symbol field
  - WATER\_PUMP, in %Q0.0 symbol field
  - IRRIGATION\_VALVE, in %Q0.1 symbol field

**Note:** Symbols are in upper-case format by default. A valid symbol can contain any combination of 32 characters: (A-Z),(0-9), and (\_). No spaces allowed.

4. Changes are effective as soon as the cursor leaves the current input or output line.

## Configure Software Resources

Select the **Program** → **Configure** → **Configure Data** task to open the software configuration window.



Configurable software resources include:

- Simple objects
  - Memory bits (%M)
  - Memory words (%MW, %MD, %MF)
  - Constants (%KW, %KD, %KF)
- Function blocks
  - Timers
  - Counters
  - Drum controllers
  - LIFO/FIFO registers
  - Shift bit registers
  - Step counters

- I/O objects
  - Fast counters
  - PLS/PWM pulse generators
  - Very fast counters
- Advanced objects
  - Schedule blocks
  - PID
  - Macro functions

**NOTE:** Internal memory objects such as system bits and words, memory bits and words, and network exchange words are **non-configurable** objects.

### Automatic-Watering System Example

In this application example, you will need to configure two software objects:

- One **Schedule Block (Real Time Clock 0)** that will manage the weekly watering schedule.
- One **Counter (%C0)** that will serve as a maintenance cycle counter.

### NOTE: Configuring schedule blocks

- To support schedule blocks, make sure the RTC option is installed on your TWDLCAA10DRF controller.
- Schedule blocks are non-programmable objects that do not need any declaration in the Ladder or List program. You simply need to enable them and configure their respective parameters.

To configure the **Schedule Block (Real Time Clock 0)**, follow these steps:

1. Select the **Program** → **Configure** → **Configure Data** task to open the software objects configuration window.
2. From the "Object Categories" frame, select **Advanced objects**.  
**Result:** The list of available advanced function blocks for that particular controller is displayed in the dedicated "Advanced objects" frame below.
3. From the "Advanced objects" frame, select **%SCH - Schedule blocks**.  
**Result:** The schedule blocks configuration window appears in the center pane. Available schedule blocks (numbered 0 to 15) are listed in a table.
4. Select the schedule block you want to configure in the table: **Real Time Clock 0**.  
**Result:** By default, the schedule block is disabled and all fields appear as non-configurable in the configuration window.
5. Click the **Configured** checkbox to activate the configuration fields and follow these steps to configure the schedule block (the following settings are given as an example only):
  - a. Enter the **Output Bit** address. In this application, this is **the address of the output terminal connected to the control valve of the irrigation system: %Q0.1**
  - b. Select **Start Month** and **End Month** for starting and ending the Schedule Block.  
**Range:** January to December.

- c. Select **Start Date** and **End Date** for the day of the month for starting and ending the Schedule Block.  
**Range:** 1 to 31 (depending on month selection).
  - d. Select the **Start Time** and **End Time** for setting the time of day for starting and ending the Schedule Block.  
**Range:** 00:00 to 23:59.
  - e. Select the **Days of the Week** to activate the schedule block output. Checked boxes indicate the days that the output device is activated.
6. Use the Up/Down keys to scroll through the list of schedule blocks in the table or use the mouse pointer to select a particular schedule block.
  7. Changes are effective as soon as the cursor leaves the current schedule block line.

Now that you have configured the schedule block, follow these steps to configure the **maintenance cycle counter (%C0)**:

1. Select the **Program** → **Configure** → **Configure Data** task to open the software objects configuration window.
2. From the "Object Categories" frame, select **Function blocks**.  
**Result:** The list of available function blocks for that particular controller is displayed in the dedicated "Function blocks" frame below.
3. From the "Function blocks" frame, select **Counters**.  
**Result:** The counters configuration table appears in the center pane. Available counters range from #0 to #127.
4. Select the up/down counter you want to configure in the table: **%C0**.
5. For the selected up/down counter set the corresponding fields in the table, as follows (the following settings are given as an example only):
  - a. Click the **Symbol** text box and enter a symbolic name for this counter: MAINTENANCE\_REQUIRED (for example.)  
**Note:** Symbols are in upper-case format by default. A valid symbol can contain any combination of 32 characters: (A-Z),(0-9), and (\_). No spaces allowed.
  - b. Select a **Preset** value: **100** (watering cycles, for example.)  
**Range:** 0 to 9999.  
**Default:** 9999.
  - c. Select **Adjustable** option to allow the Preset value to be changed from within Animation Tables.
6. Use the Up/Down keys to scroll through the list of counters in the table or use the mouse pointer to select a particular counter.
7. Changes are effective as soon as the cursor leaves the current counter line.




**NOTE:** Counters are programmable objects that need to be declared in the Ladder or List program. It is at that stage that you will specify whether the counter used is a up or down counter.

## Step 5: Write the Program

### Introduction

Use the **Program Editor** to develop the main program which is the core of your Twido automation application. Major programming and configuration features include reversible ladder and list language programming.

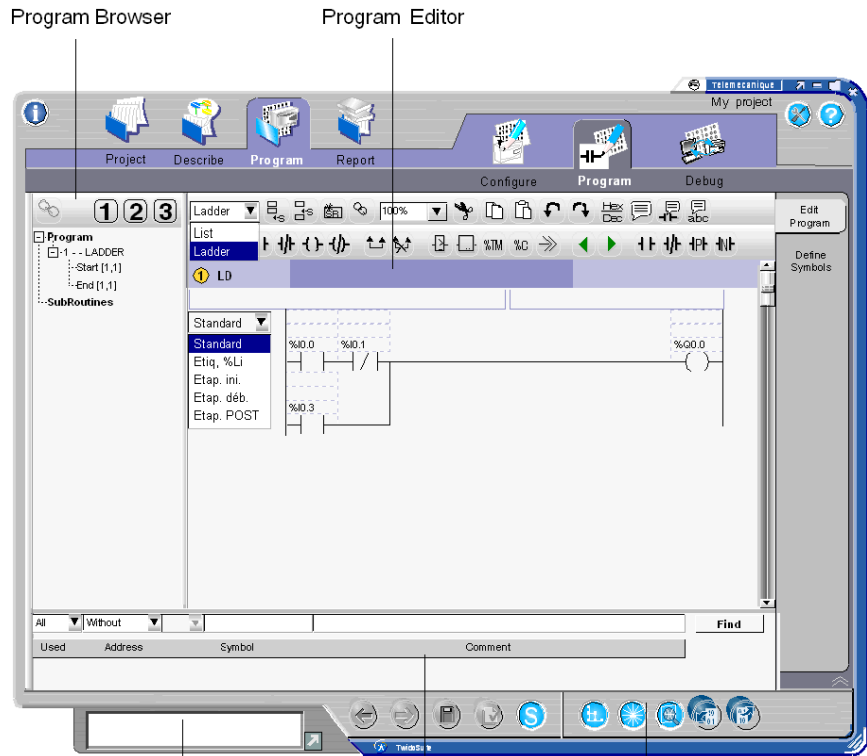
This section will focus on providing you the basics of Ladder language programming in TwidoSuite program editor. For details on List language programming and Ladder/List reversibility features, please refer to the *TwidoSuite Programming Guide* on the CD-ROM.

To have access to the program editor, select **Program**  → **Program**   on the TwidoSuite Steps bar and click **Edit Program** on the task bar.

**NOTE:** In program mode, the task bar provides access to the **Symbol Editor** as well.

## Program Edition Window

The **Program Edition** window features the **Program Editor** and **five browsers and tools** available on the functions bar, as described in the following figure:



Error List Box

Data Browser

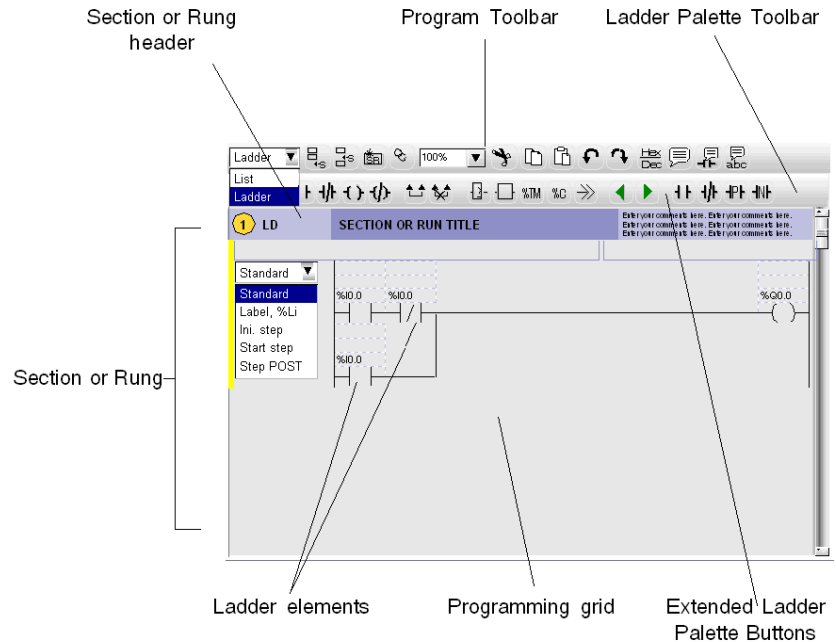
Functions Bar

- Import Program
- Export Program
- Show/hide ProgramBrowser
- Show/hide Cross-Ref Browser
- Show/hide Data Browser

## Ladder Editor Elements

The **Ladder Editor** is a graphics-based, program editor used to view, create and edit ladder programs.

When you select **Ladder** in the program tool bar drop-down list box, the program editor switches to Ladder mode. The following figure describes the various elements that make up the ladder editor:

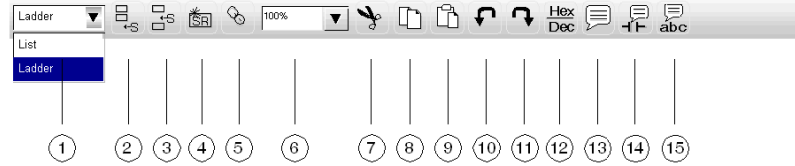


## Program Edition Tool Bars

3 ladder programming tool bars (see figure above) are available that allow you to easily edit programs and graphically enter ladder instructions:

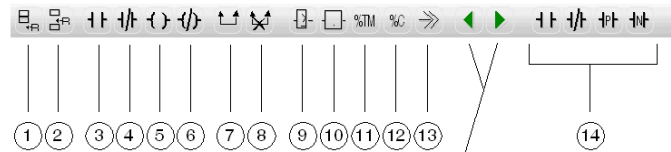
- Program Tool Bar
- Ladder Palette Tool Bar
- Extended Ladder Palette

### Program Tool Bar



- 1 Toggle Ladder/List mode
- 2 Add section
- 3 Insert section
- 4 Add subroutine
- 5 Cut in 2 sections
- 6 Zoom ratio (default is 100%)
- 7 Cut
- 8 Copy
- 9 Paste
- 10 Undo
- 11 Redo
- 12 Display in hexadecimal or decimal
- 13 Display/Hide rung comments
- 14 Display/Hide ladder elements comments
- 15 Display/Hide symbols

### Ladder Palette Tool Bar

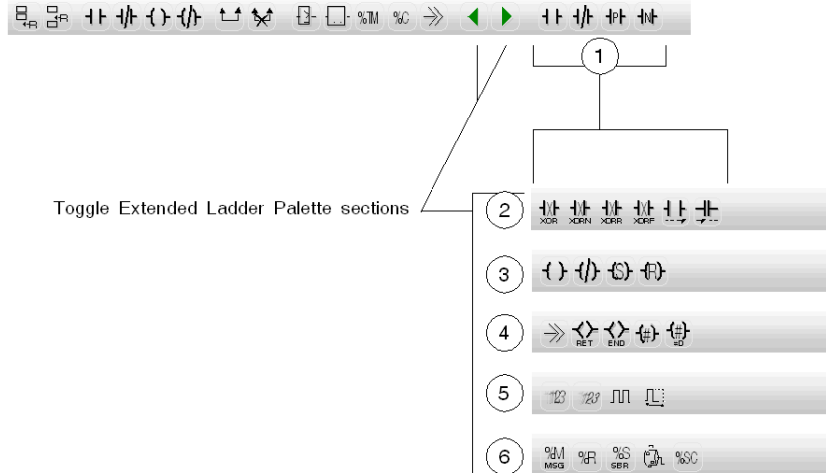


Toggle Extended Ladder Palette sections

- 1 Add rung
- 2 Insert rung
- 3 Add normally open contact
- 4 Add normally closed contact
- 5 Add coil
- 6 Add inverse coil
- 7 Add link
- 8 Remove link
- 9 Add Comparison block
- 10 Add Operation block
- 11 Add Timer block
- 12 Add Counter block
- 13 Add Jump or Subroutine Call
- 14 Extended Ladder Palette buttons (see below)



## Extended Ladder Palette



- 1 Contacts: Normally Open, Normally Closed, Rising Edge, Falling Edge
- 2 Special Contacts: XOR, XORN, XORR, XORF, OPN, SHORT
- 3 Coils: Normal Coil, Inverse Coil, Set Coil, Reset Coil
- 4 Special Coils: JMP/SR, RET, END, Grafcet Coil 1, Grafcet Coil 2
- 5 Basic Function Blocks: %FC, %VFC, %PLS, %PWM
- 6 Advanced Function Blocks: %MSG, %R, %SBR, %DR, %SC

## Overview of Ladder Programming

You write a ladder program in the logical order required to control a machine or process.

### Sections/Subroutines

A TwidoSuite ladder program is made up of a sequence of **sections** and **subroutines** containing ladder **rungs**. Sections and subroutines are delimited by section/subroutine headers. They are numbered automatically by the TwidoSuite software.

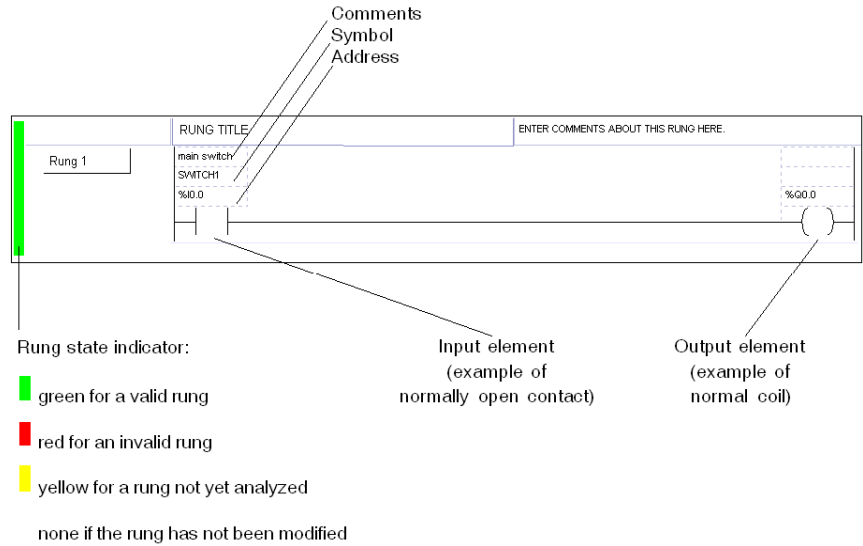
The following figure describes a typical **section header** (subroutine headers are similar to this).



### Rungs

A ladder **rung** is similar to a relay logic diagram. Each rung consists of one network of linked graphical elements organized in a programming grid starting with a potential bar on the left and ending with a potential bar on the right.


The following figure describes a typical ladder **rung**.

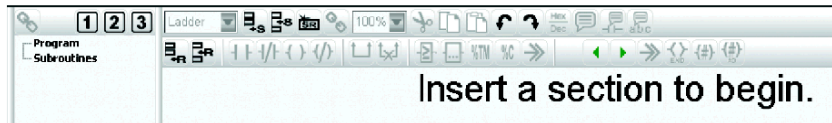


**NOTE:** Programs are limited to a total of 128 sections and subroutines with a maximum of 64 subroutines (SR0 to SR63).

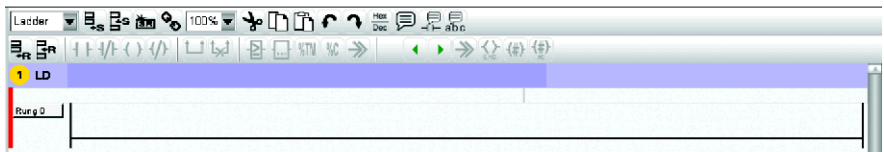
### How to Edit a Program

The following procedure gives you the basics to start editing a ladder program.

1. When you start writing a new program, press the **Add Section** icon  on the program tool bar at the following prompt.

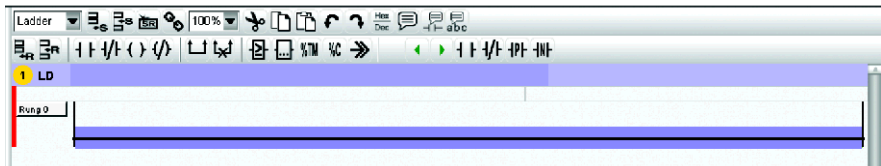



**Result:** The first section is inserted in the ladder editor, displaying the first empty rung, as shown below. (Note that a program **must** contain at least one section.)



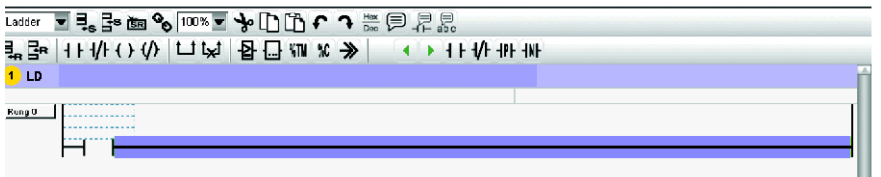
2. Double-click the section header to enter the following:
  - Section title
  - Comments
3. Double-click the rung header to enter the following:
  - Rung title
  - Comments
  - Declare rung type: Standard, Label (%Li), or Grafcet step (Ini. step, Start step, or Step post)
4. Select the empty horizontal link that runs between the rung left and right power lines.
 

**Result:** The selected link is highlighted in blue..




5. Click the **Normal contact** icon  (example) on the Ladder Palette to insert the input graphical instruction.
 

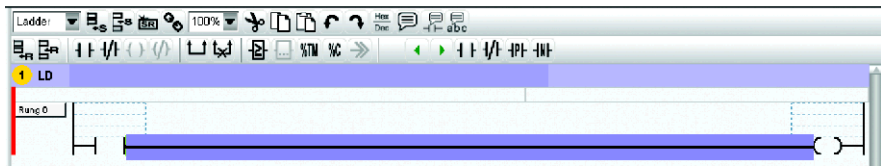
**Result:** The normal contact is inserted in the leftmost part of the rung.



6. Repeat steps 4 to 5 to insert other graphical instructions, as needed.
7. A valid rung must always feature an output element (coil or operation block). Click

the **Coil** icon  (example) on the Ladder Palette to insert the output graphical instruction.

**Result:** The coil element is inserted in the rightmost part of the rung.




**Note:** At this stage, you will note that the red color bar shows that the rung is still invalid.

- You must specify an address to each element in the rung. Click the address field above each element and enter the desired address. Enter **%I0.0** for the input contact and **%Q0.0** for output coil (example). Note that you may also optionally specify a symbol and comment for each element.



**Result:** The yellow color bar indicates that this rung is valid, but has not been analyzed (compiled with the program) yet.

- Click the **Analyze** icon  to start the program validation process and check for possible program errors in the **Error list box**.

**NOTE:** To find out more about program edition rules and procedures, please refer to the *TwidoSuite Operation Guide - Online Help*.

### Example Program for the Automatic-Watering System

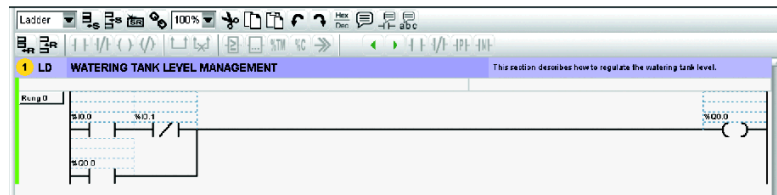
This section gives you the ladder program that applies to the automatic-watering system example. This application consists of 2 independent processes:

- Water tank level management
- Watering Count

This will require to create 2 sections in the program, one for each process.

#### Water tank level management section

Insert the first section of the program, and create the following program.

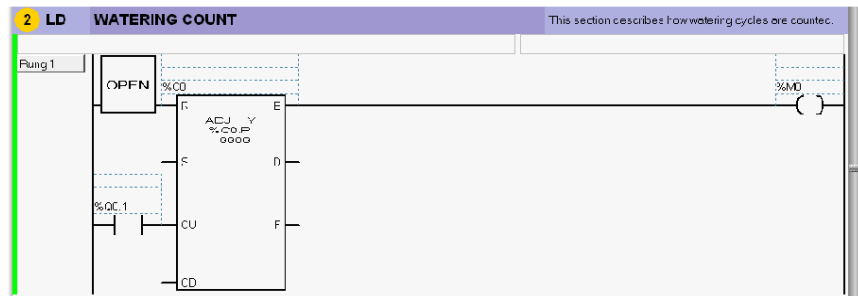


#### Description of the water tank level regulation process:

- Whenever the water low-level is reached (%I0.1 is closed), the water-pump is started (%Q0.0 is closed). Note that %I0.0 remains normally closed.
- Whenever the water high-level is reached (%I0.0 is open), the water-pump is stopped (%Q0.0 coil is activated and %Q0.0 contact is closed). Note that the %Q0.0 contact allows to keep the water-pump running until the water high-level is reached.

## Watering count

Add a section to the program, create the following program that will count the watering cycles and save your project. (**Note:** If you have not done it yet at the Data Objects Configuration stage, double-click the %C0 element and set the **Preset** value to **100** cycles in the Counters table.)



### Description of the watering-count process:

- 1 Whenever the watering control valve is activated (%Q0.1 is closed) by the RTC function (Real-Time Clock 0), the counter %C0 is incremented.
- 2 When the counter current value reaches 100, the memory bit %M0 is set to 1. Memory bit %M0 can be used to control an LED that will indicate that maintenance is required after completion of the 100 watering cycles. Note that you may connect a push button to the R input to manually reset the maintenance cycle counter.

## Save the Program

Once you have finished writing your program, you can save your program in any of 2 ways:

- Provided that you have already given a name to your application and specified the project directory (see *Step 1: Create a New Project, page 25*), click the **Save**

**current project** shortcut  on the **Quick Access Bar**.

- Select  **Project** → **Save current project** task and click the **Save** button.

**Result:** The project files containing the application program are saved to the disk in the project directory which location has been specified in the project file information window.


**NOTE:** It is good practice to frequently save your project to the PC disk as you are writing larger ladder/list instructions programs to prevent any significant loss of work due to power shut-down or other computer problems.

## Step 6: Test the Program

### Introduction

TwidoSuite offers a simulator function that enables you to test a newly written program without loading it on a controller. The Twido Simulator lets you run the program and test different parameters in order to see if the expected responses are occurring.

### Starting Twido Simulator

To start the Twido Simulator select **Program** → **Program** and click the Twido Simulator button  at the bottom right of the TwidoSuite screen.

Result:

- TwidoSuite switches automatically to **Program** → **Debug** → **Animate the program**.
- The Twido Simulator overview window appears.
- The Dashboard pops up.
- If an RTC is configured or embedded, the Time Management Panel is also displayed.

The Simulator loads the configuration you have created in the Describe screen.

**NOTE:** Some functions of your application cannot be tested using the Simulator.

### Using Twido Simulator

Once the Simulator is launched you can use the functions that are available in debug mode. These are:

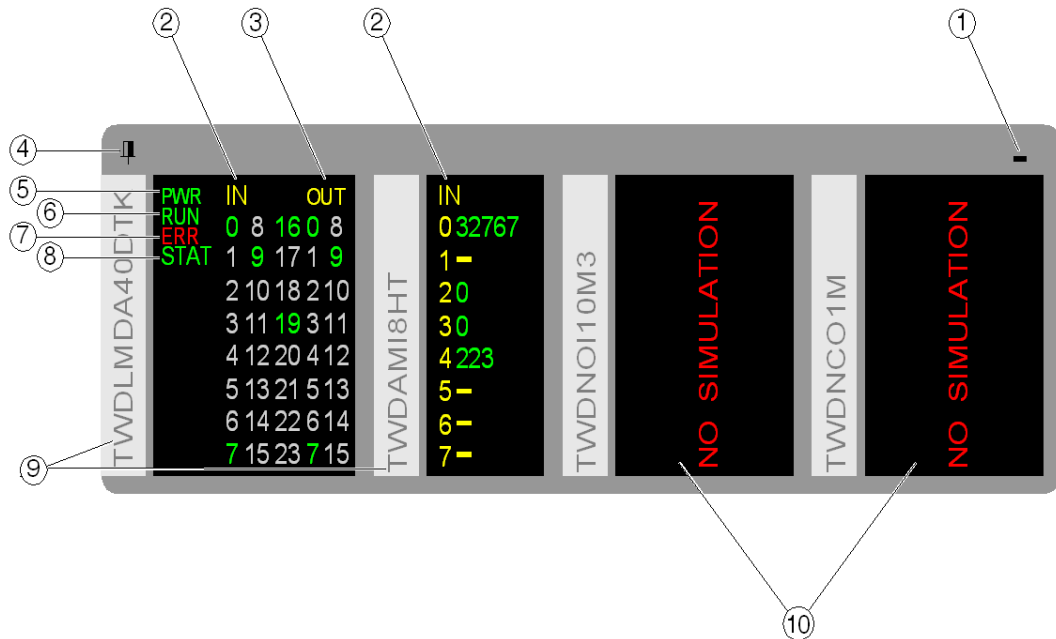
- Animate the program
- Manage animation tables
- Check PLC
- Monitor hardware configuration
- Monitor software configuration
- Monitor described configuration
- Monitor the behavior

You can find more information on animating tables and managing animation tables in Debug the Program (*see page 58*).


## Twido Simulator Overview Window

The Twido simulator overview window shows the state of inputs/outputs of all the described base controller and expansion modules.

The overview window is shown below:



1. Click to minimize to tray.

When the overview window is minimized, the Twido Simulator icon  appears in the status area (on the bottom right of your screen). Double click on this icon to restore the overview window.

2. Inputs states:

- Click to set/reset the corresponding input
- Double click to open the discrete input settings window.

3. Outputs states: double click to open the output tracing window.

4. Click to always toggle the overview window on top

5. **PWR** LED: Indicates if the simulated PLC is powered up.

6. **RUN** LED: Indicates the state of the simulated PLC.

7. **ERR** LED: Indicates the state of the simulated PLC.

- 8. **STAT LED**: The operation of the **STAT LED** is defined by user logic
- 9. Base controller and expansion modules references
- 10. Expansion modules that cannot be simulated.

Right click on the Overview Window to expand the following contextual menu:




- Time Management: Opens the Time Management panel.
- Output Tracing: Opens the Output Tracing window.

To access analog input settings, double click on the analog input LED.

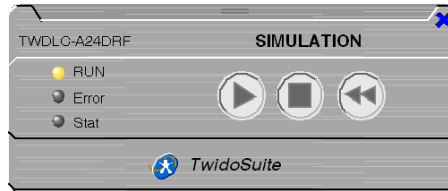
To access discrete input settings, double click on the discrete input LED.

### Twido Simulator Dashboard

You can use the Twido Simulator dashboard to issue the following commands to the controller:

- **Run (in simulate mode)** 
- **Stop** 
- **Initialize** 

When the TwidoSuite application is in simulate mode, these commands can be selected from the Twido Simulator dashboard shown below:

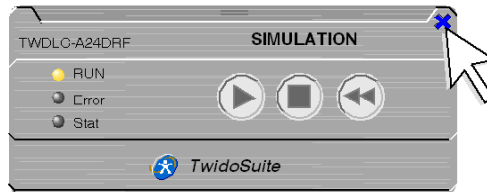


**RUN**, **ERR**, and **STAT** LEDs are simulated in the Twido Simulator dashboard as they would appear on a connected base controller (for details about these LEDs, see the Twido Hardware Reference Guide).



## Using Twido Simulator Dashboard

To close the control panel dashboard:

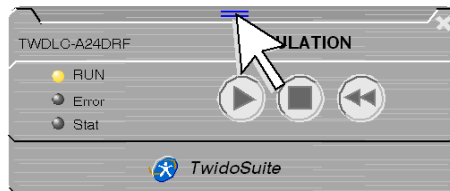


To re-open the dashboard, click the button as shown below:

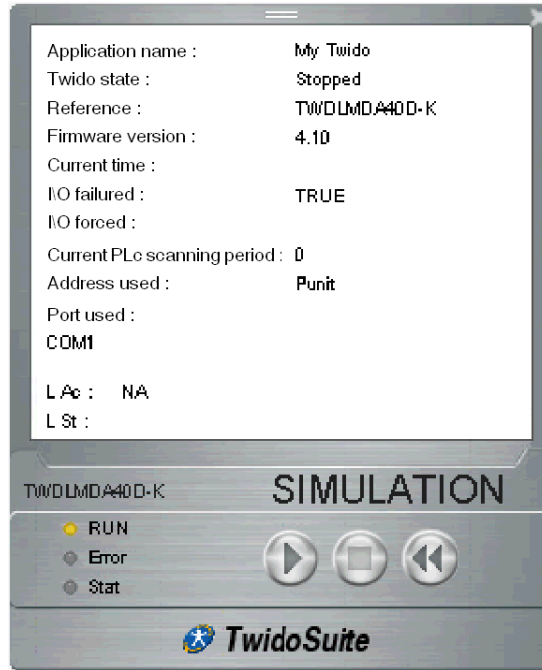


## Expanding Twido Simulator Dashboard

To expand/reduce the control panel dashboard, click as shown below



The dashboard is then expanded as shown below. It may be reduced again by clicking the same expansion/reduction button.



### Time Management Panel

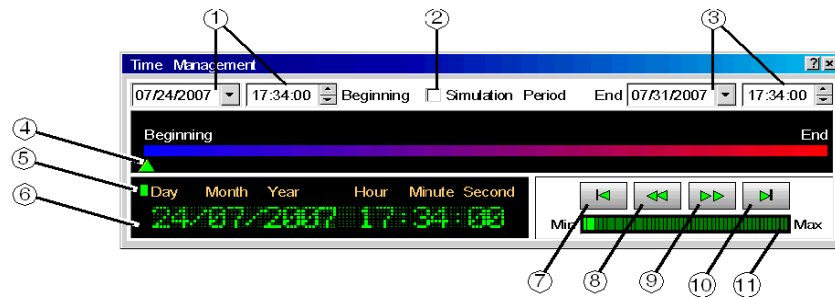
The Time Management Panel allows to control the timing of the simulation (Action on the time management panel modifies the RTC time accordingly and immediately), you can choose:

- The beginning and the end of the simulation
- To accelerate the simulation.

**NOTE:** The default initial time of the simulation is the time of your PC.

The default duration of the simulation is 7 days.

The Time Management Panel is shown below:



1. Beginning time: Time at the beginning of the simulation
2. Simulation Period:
  - If this box is checked, then when end time is reached the simulator stops and a window pops up with the following text: End Time Reached! PLC switches to STOP even if you don't click OK.
  - If this box is not checked the simulation stays in RUN mode.
3. End time: Time at the end of the simulation
4. A cursor indicates the current time progression.  
You can move manually the cursor to set current time.
  - If the current time is former to the beginning time then the cursor will go on "Beginning"
  - If the current time is posterior to the end time then the cursor will go on "End".
5. Simulator status:
  - ■ : Simulator in Stop mode
  - ▶ : Simulator in Run mode
  - ▶▶ : Simulator in fast forward
  - ◀◀ : Simulator in fast rewind
6. Indicates the current time of the simulation
7. Back to beginning time
8. Fast rewind in time
9. Fast forward in time
10. Go to the end time.
11. Set the level of fast forward/rewind in time

## Setting Simulation Parameters

In order to run tests on your program, the Twido Simulator lets you set the input values and the configure which outputs are traced.

You can set both the discrete inputs and analog inputs.

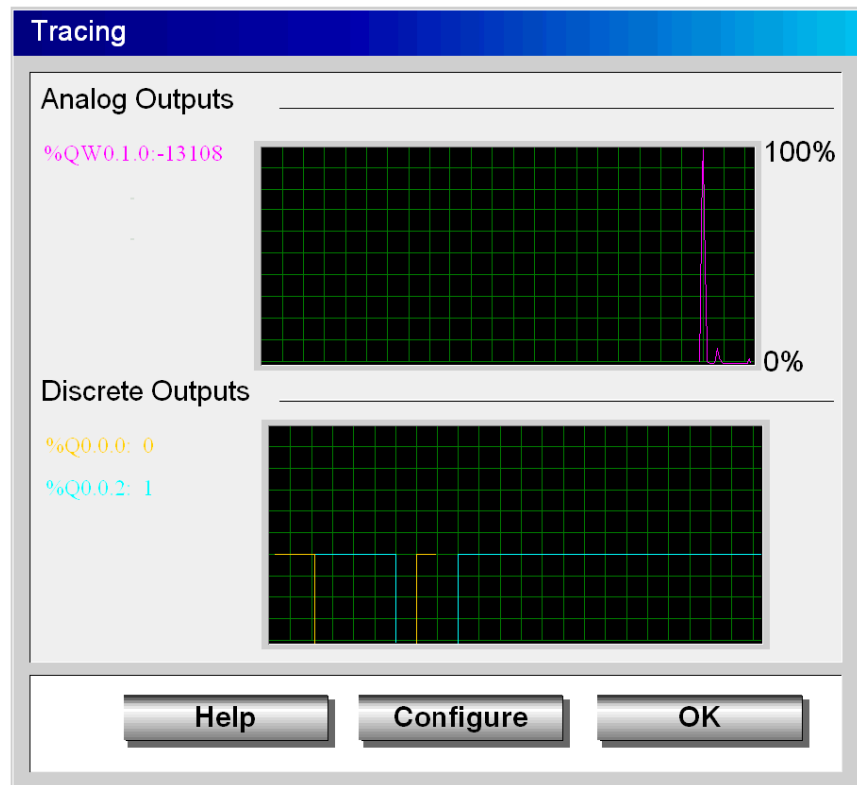
- To open the discrete input settings window, double-click on the discrete input LED.
- To open the analog input settings window, double-click on the analog input LED.

You can also configure up to 8 outputs to be traced. You open the tracing configuration window by clicking the Configure button of the output tracing window.


## Output Traces with the Simulator

The outputs tracing window shows the variation of analog and discrete outputs over time.

The outputs tracing window is shown below:



## Stopping Twido Simulator

To stop the Twido Simulator, click the button  at the bottom right of the TwidoSuite screen.

After stopping the simulation TwidoSuite returns to the screen displayed before starting Twido Simulator.

**NOTE:** It is not possible to backup the Twido Simulator program before stopping.

## Step 7: Transfer and Run the Program

### Introduction

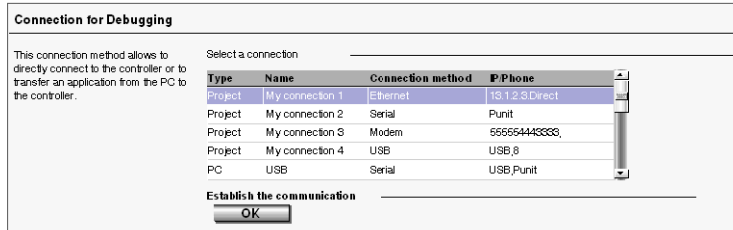
In this section, you will find out how to transfer the program to the Twido PLC and start running your program on the controller. Note that a program **must** be saved before it can be transferred to the PLC.

### Transfer the Program

In order to run your application for debugging, you must first transfer it to the controller RAM.

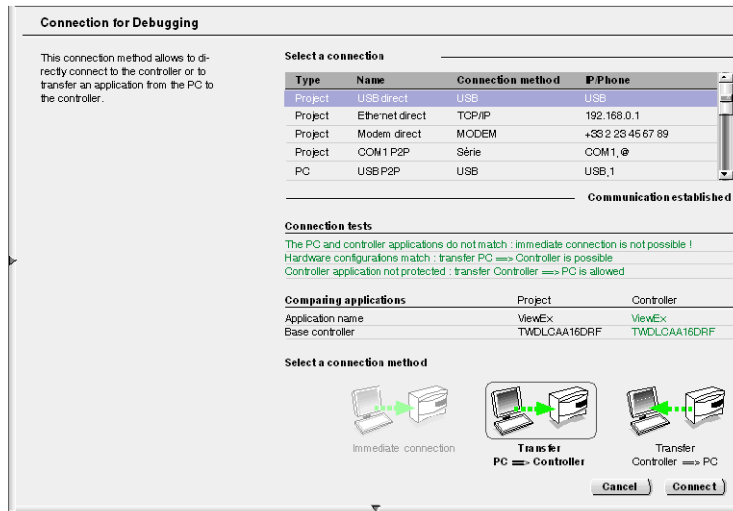
Follow these steps to download your application program to the PLC:

1. If the PC is not yet connected to the controller, select a connection from the connections table in **Program** → **Debug** → **Connect** task and click **Run**.



**Results:** TwidoSuite tries to establish a connection with the controller and performs synchronization checks between the PC and the PLC applications.

- The **Connection Test Table** displays the results of 3 synchronization checks.



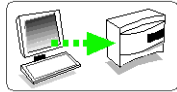
Line 1: Comparison between the PC and the PLC applications

Line 2: Hardware compatibility check

Line 3: Security check (if the PLC application is valid), otherwise indicates if the PLC application is invalid.

- The **Application Comparison Table** is displayed only if both PC and PLC application are not identical.

2. If the PC and PLC applications are different, select **Transfer PC => Controller**.



3. If the controller application is password protected, you are prompted for a password. Enter the password and select **OK** to continue the transfer.
4. If the controller is running, a warning box is displayed stating that to continue the transfer, the controller must be stopped.

If the controller is stopped, the transfer begins. If the controller is running, select **OK** to stop the controller and continue the transfer. Select **Cancel** to close the box and cancel the transfer.

A warning dialog box is displayed indicating that the contents of the controller application will be overwritten.

5. Select **OK** to continue the transfer.

An option can be set to protect the application (see Security Features in the TwidoSuite Operation Guide - Online Help.)

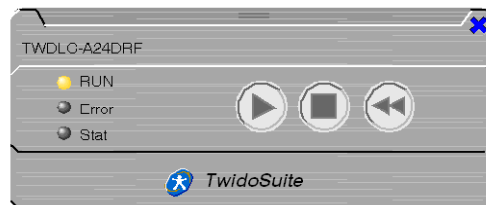
**NOTE:** Symbols and program comments are not transferred to the controller.

## Run the Program

Use the PLC **Soft Control Panel** to Run/Stop/Initialize a program that has been downloaded to the PLC (see *Transfer the Program*, page 54.)

To run the program that you have just transferred to the PLC, follow this procedure:

1. If the connection between TwidoSuite and the PLC is not active, select a connection from the connections table in **Program** → **Debug** → **Connect** task and click **OK** to establish a connection.
2. When a successful connection is established between TwidoSuite and the controller, the PLC **Soft Control Panel** dialog box appears in the form of a dashboard, always on top of your TwidoSuite workspace.



It features the following command buttons:

-  **RUN**
-  **STOP**
-  **INITIALIZE**

3. Click **RUN** to execute the TwidoSuite application program currently in controller's RAM.

**Note:** Each command requires confirmation before execution.

**Result:** 3 soft panel LEDs (RUN, Error, Stat) provide live diagnostics. (See *TwidoSuite Operation Guide -Online Help* for details.)

### Using the Control Panel Dashboard

The Control Panel Dashboard is used in the same way as the Dashboard used for the Simulator, refer to Twido Simulator Dashboard (*see page 48*).



## Step 8: Debug the Program

### Overview

TwidoSuite provides many options to help you debug and adjust your program:

- Cross-Reference browser
- Animation tables
- Forcing I/Os
- Program animation



### Cross-Referencing your Program


#### Introduction

Cross referencing makes it easier to locate several occurrences of the same program element through the application, for debugging purposes.

It consists of selecting items in a program such as an address that you want to track in an application. Once the items are selected, then a list is generated displaying where the items are used in the application. Cross referencing provides a useful list of the operands, symbols, lines and operators used in all or part of an application.

#### Cross-Reference Browser

Select **Program**  → **Program**  on the TwidoSuite Steps bar .


Click **Cross-Ref browser**  on the functions bar to display the cross-reference browser. The cross-reference browser displays to the right of the program window.

Operand: %I0.0.0	<a href="#">Configure</a>	
Symbol: TEST		
Section	Location	Operator
Section 1	Rung 1	— —
Section 1	Rung 1	— /—

#### How to Cross-Reference your Application

This is a simple four-step procedure:

1. Click the specific program item you want to cross-reference.  
**Result:** The cross-reference table is updated.
2. If you want to change the configuration settings of the cross-referenced object, click the **Configure** shortcut in the Cross-Reference browser.  
**Result:** This brings up the hardware or software object configuration screen with the cursor pointing on the cross-referenced object.

3. Configure the desired object in the hardware or software object configuration table.
4. Click the  button to return to the Cross-Reference browser.

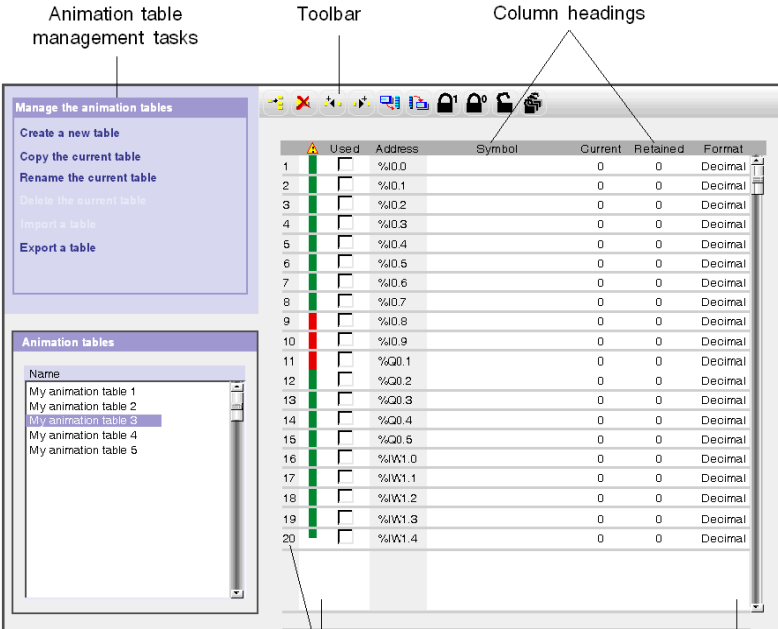
### Using Animation Tables

Use an animation table to view and adjust program variables **online**, when the PLC is either in Run or Stop mode.

With an animation table, you can

- as the program is running, monitor the actual value of any variable in the controller and compare against expected value,
- change the value of variables independent of calculated values, until they are updated again,
- backup the current values of all variables in an animation table,
- restore to the controller the retained values of all variables in an animation table,
- force inputs and outputs.

### Elements of an Animation Table



The screenshot shows the 'Animation table' window. On the left, there are two panels: 'Manage the animation tables' with options like 'Create a new table', 'Copy the current table', 'Rename the current table', 'Delete the current table', 'Import a table', and 'Export a table'; and 'Animation tables' with a list of tables named 'My animation table 1' through '5'. The main area is a table with columns: 'Used', 'Address', 'Symbol', 'Current', 'Retained', and 'Format'. The 'Address' column is highlighted with a label 'Address' pointing to the '1' in row 1. The 'Symbol' column is highlighted with a label 'Column headings' pointing to the header. The 'Current' and 'Retained' columns are highlighted with a label 'Animation table' pointing to the '0' in row 1. A toolbar with various icons is located above the table.

Used	Address	Symbol	Current	Retained	Format
<input checked="" type="checkbox"/>	%I0.0		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.1		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.2		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.3		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.4		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.5		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.6		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.7		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.8		0	0	Decimal
<input checked="" type="checkbox"/>	%I0.9		0	0	Decimal
<input checked="" type="checkbox"/>	%Q0.1		0	0	Decimal
<input checked="" type="checkbox"/>	%Q0.2		0	0	Decimal
<input checked="" type="checkbox"/>	%Q0.3		0	0	Decimal
<input checked="" type="checkbox"/>	%Q0.4		0	0	Decimal
<input checked="" type="checkbox"/>	%Q0.5		0	0	Decimal
<input checked="" type="checkbox"/>	%IW1.0		0	0	Decimal
<input checked="" type="checkbox"/>	%IW1.1		0	0	Decimal
<input checked="" type="checkbox"/>	%IW1.2		0	0	Decimal
<input checked="" type="checkbox"/>	%IW1.3		0	0	Decimal
<input checked="" type="checkbox"/>	%IW1.4		0	0	Decimal

For more details, please refer to *Debugging Online via the Program Animation in TwidoSuite Operation Guide - Online Help*.

## Forcing I/Os

Forcing I/Os means that you forcibly set the values of one or more input or output bits to either 0 or 1.

**NOTE:** You can force I/Os when TwidoSuite is in **online** mode, with the controller either in Run or Stop mode.

### How to Force I/Os in an Animation Table

The animation table tool bar provides **Force Commands**:

- Force to 1
- Force to 0
- Clear Force
- Clear All Force

To force a value to 0 or 1, or clear a forced value:

1. Highlight a variable in the animation table.
2. Click the desired **Force to 0**, **Force to 1** or **Clear Force** command on the animation table tool bar.

**Result:** In the column labeled **Current** in the animation table, the forced variable displays with a "F" prefix paired with the forced state "0" or "1": "F 0" or "F 1".

## Animating a Program

You can animate both Ladder and List programs when online. This section describes how to animate a Ladder language program. For more details, please refer to *Animating a Program* in *TwidoSuite Operation Guide - Online Help*.

### How to Animate a Ladder Program

1. Select the **Program** → **Debug** → **Animate program** task to open the program editor in animated mode.

**Result:** The program editor displays animated ladder elements.


2. Watch the animated ladder elements during program execution.

### How Ladder Animation Works


- You can watch the backlight of logical objects such as contacts, coils and special objects turn on as their logical value becomes 1.
- You can keep track of data variables of fonction blocks, comparison blocks and operation blocks, including current and preset values.

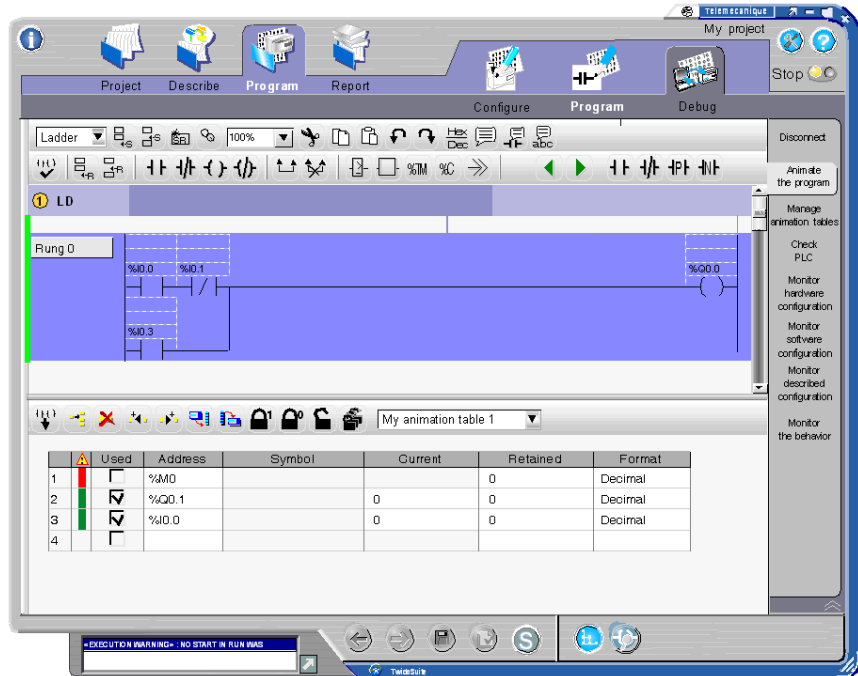
**NOTE:** Binary operands are displayed as 0 or 1, while all other values are displayed in hexadecimal or decimal, depending on the format selected in Preferences (*see page 22*).

### Display/hide the Animation tables

Click the icon  in the functions bar to display/hide the Animation tables in the lower part of the window.

### Copying Rung Objects to an Animation Table

To copy rung objects to the selected animation table, press  (leftmost button on the animation table's tool bar) as shown in the figure below.




## Step 9: Manage the Program Report

### Introduction

With TwidoSuite report, you create custom printouts and html exports (to .htm file) by selecting report items from the report manager browser.

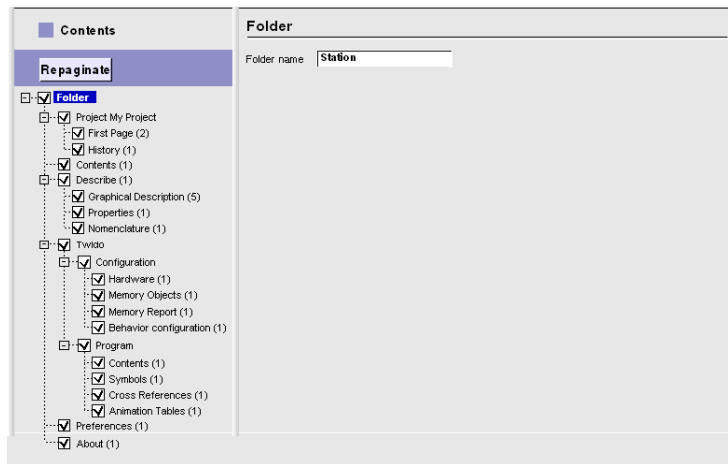
### Report Manager

Use the report manager to select report components, configure printing options, and export the application report to HTML format.

To access the Report Manager window, select the **Report** icon  on the TwidoSuite Steps bar.

### How to Configure the Report

The report manager provides a tree-like structure view of the report items and subitems that can be included in the report by checking the appropriate boxes.



To configure the report, select an item from the report browser and set the options that appear in the configuration window. For details on report options, please refer to the *TwidoSuite Operation Guide - Online Help*.

## Page Setup

Select the **Report** → **Page Setup** task to set the information you want to display in page headers and footers, and the page margins as well.

You can configure your report pages to show up to nine different types of information in three separate fields per header and footer line, including:

You can configure in your report page to show up to three different type of information in three separate fields per header and footer, including:

- Print date
- Creation date
- Modification date
- Project name
- Any text (Free)
- No text (None)
- Page number/Total number of pages (Page i/n)
- Page number (Page i)
- TwidoSuite version
- File name

## Print the Report

Select the **Report** → **Print** task to prepare the report for sending to the printer and/or to an HTML export file.

The print window allows you to:

- preview the report with several page format and zooming options,
- choose the destination printer and access the printer properties,
- repaginate your report according to selected printing options
- send your report for printing on the selected print device,
- export your report to a HTML file.

---

# Monitoring an Application

# 6

---

## Monitoring an Application

### How to Start the Monitoring Utility

To start the TwidoSuite Monitoring Utility:

1. Open the Application Launcher by clicking the icon located on your PC desktop.



**Result:** The Application Launcher dialog box appears as shown below.



2. From TwidoSuite Application Launcher, click Monitoring Mode

## Monitoring Features

Select Check PLC:

The screenshot displays the Monitoring Utility interface. On the left, a 'Select a task' panel lists several options, with 'Modify modem Configuration' selected. Below this, the 'Modify modem configuration' dialog box is open, showing a modem selection dropdown set to 'TD-33/V90', a 'Reset Hayes Command' field containing 'ATEOO1&B1&DO&HO&IO&R1&SOSO', and 'Apply' and 'Cancel' buttons. The main interface on the right is divided into several sections: 'Controller Status' with checkboxes for 'RAM Executable', 'RAM Protected', and 'I/O Forced', and a 'Date and time of the last top' field; 'Potentiometers' with two adjustment point input fields; 'Scan Time' with three input fields for 'Maximum (ms)', 'Current (ms)', and 'Minimum (ms)', and a 'Watchdog time out' checkbox; 'Controller Real Time Clock' with fields for 'Date (dd/mm/yy)', 'Time (hh:mm:ss)', and 'RTC Correction'; and 'Modem connected' with fields for 'Name' and 'Command'.

The Monitoring Utility allows you to perform the following tasks:

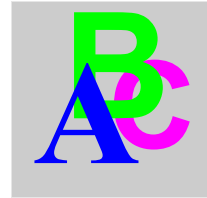
1. See memory information
2. Set RTC time
3. Configure RTC correction factor
4. See Ethernet statistics
5. Modify modem configuration

For details on how to use the various monitoring features, please refer to *Hardware Monitoring* chapter in *TwidoSuite Operation Guide - Online help*.



---

# Index



---

## E

Executive Loader Wizard, *15, 15*

## F

Firmware Executive  
Update, *15*

## H

hardware installation requirements, *10*

## I

installation requirements, *10*  
installing the software, *11*

## P

Preferences, *22*

## R

removing the software, *11*

## S

software installation requirements, *10*  
start program, *13*

## U

uninstalling the software, *11*

