

SYMAX Communication Driver

Driver for Serial Communication
Between Studio and SY/MAX Devices

Contents

INTRODUCTION	2
GENERAL INFORMATION.....	3
DEVICE CHARACTERISTICS	3
LINK CHARACTERISTICS	3
DRIVER CHARACTERISTICS	3
CONFORMANCE TESTING	4
INSTALLING THE DRIVER	5
CONFIGURING THE DRIVER	6
SETTING THE COMMUNICATION PARAMETERS	6
CONFIGURING THE DRIVER WORKSHEET	7
EXECUTING THE DRIVER	10
TROUBLESHOOTING	11
SAMPLE APPLICATION	13
REVISION HISTORY.....	14

Introduction

The SYMAX driver enables communication between Studio system and the SY/MAX devices, according to the specifications discussed in this publication.

This publication was designed to help you install, configure, and execute the SYMAX driver to enable communication with the SY/MAX devices. The information in this publication is organized as follows:

- **Introduction:** Provides an overview of the SYMAX driver documentation.
- **General Information:** Provides information needed to identify all the required components (hardware and software) used to implement communication between Studio and the SYMAX driver.
- **Installing the Driver:** Explains how to install the SYMAX driver.
- **Configuring the Driver:** Explains how to configure the communication driver.
- **Executing the Driver:** Explains how to execute the driver to verify that you installed and configured the driver correctly.
- **Troubleshooting:** Lists the most common error codes for this protocol and explains how to fix these errors.
- **Sample Application:** Explains how to use a sample application to test the driver configuration.
- **Revision History:** Provides a log of all modifications made to the driver and the documentation.



Notes:

- This document assumes that you have read the “Development Environment” chapter in the product’s *Technical Reference Manual*.
- This document also assumes that you are familiar with the Windows NT/2000/XP environment. If you are unfamiliar with Windows NT/2000/XP, we suggest using the **Help** feature (available from the Windows desktop **Start** menu) as you work through this guide.

General Information

This chapter explains how to identify all the hardware and software components used to implement communication between the SYMAX driver and SY/MAX devices.

The information is organized into the following sections:

- Device Characteristics
- Link Characteristics
- Driver Characteristics

Device Characteristics

This driver has been tested successfully with the following devices:

- **Manufacturer:** Squared D
- **Compatible Equipment:** SY/MAX

Link Characteristics

To establish communication, you must use links with the following specifications:

- **Device Communication Port:** Serial Port
- **Physical Protocol:** RS 232
- **Logic Protocol:** SY/MAX Protocol
- **Adapters/Converters:** None

Driver Characteristics

The SYMAX driver is composed of the following files:

- **SYMAX.INI:** Internal driver file. *You must not modify this file.*
- **SYMAX.MSG:** Internal driver file containing error messages for each error code. *You must not modify this file.*
- **SYMAX.PDF:** Document providing detailed information about the SYMAX driver.
- **SYMAX.DLL:** Compiled driver.



Notes:

- All of the preceding files are installed in the /DRV subdirectory of the Studio installation directory.
- You must use Adobe Acrobat® Reader™ (provided on the Studio installation CD-ROM) to view the SYMAX.PDF document.

You can use the SYMAX driver on the following operating systems:

- Windows 2000/XP
- Windows NT

For a list of the operating systems used for conformance testing, see “Conformance Testing” on page 4.

Conformance Testing

The following hardware/software was used for conformance testing:

- **Driver Configuration:**
 - **Protocol:** SY/MAX Protocol

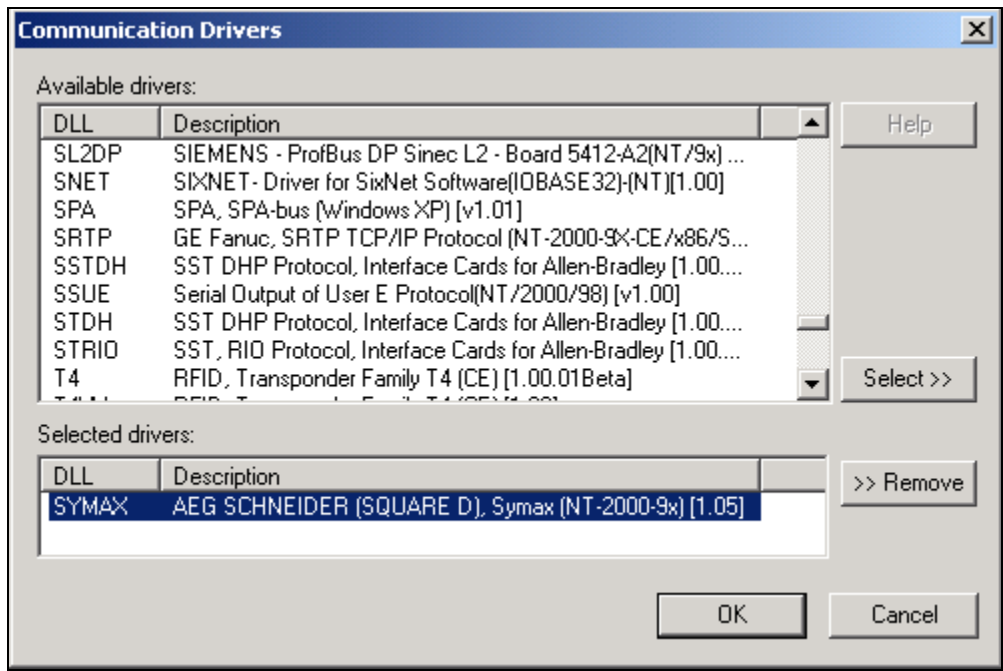
Driver Version	Studio Version	Operating System	Equipment
1.00	6.0	WinXP	SY/MAX

Installing the Driver

When you install Studio version 5.1 or higher, all of the communication drivers are installed automatically. You must select the driver that is appropriate for the application you are using.

Perform the following steps to select the driver from within the application:

1. Open Studio from the **Start** menu.
2. From the Studio main menu bar, select **File** → **Open Project** to open your application.
3. Select **Insert** → **Driver** from the main menu bar to open the *Communication Drivers* dialog.
4. Select the **SYMAX** driver from the *Available Drivers* list, and then click the **Select** button:



Communication Drivers Dialog

5. When the **SYMAX** driver displays in the *Selected Drivers* list, click the **OK** button to close the dialog.

Attention:
For safety reasons, you must take precautions when installing the physical hardware. Consult the hardware manufacturer's documentation for specific installation instructions.

Configuring the Driver

After opening Studio and selecting the SYMAX driver, you must configure the driver. Configuring the SYMAX driver is done in two parts:

- Specifying communication parameters
- Defining communication tags and controls in the Communication tables or *Driver* worksheet

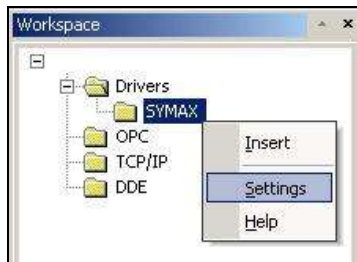
Worksheets are divided into two sections, a *Header* and a *Body*. The fields contained in these two sections are standard for all communications drivers — except the **Station**, **Header** and **Address** fields, which are driver-specific. This document explains how to configure the **Station**, **Header** and **Address** fields only.

Notes:
For a detailed description of the Studio *Standard Worksheets*, and information about configuring the standard fields, review the product's *Technical Reference Manual*.

Setting the Communication Parameters

Use the following steps to configure the communication parameters, which are valid for all driver worksheets configured in the system:

1. From the Studio development environment, select the **Comm** tab located below the *Workspace*.
2. Click on the *Drivers* folder in the *Workspace* to expand the folder.
3. Right-click on the *SYMAX* subfolder. When the pop-up menu displays, select the **Settings** option:



Select Settings from the Pop-Up Menu

The SYMAX: Communication Parameters dialog displays:



SYMAX: Communication Parameters Dialog

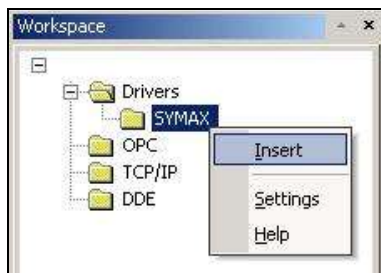
4. Click **OK** to close the dialog. This driver does not have custom parameters.

Configuring the Driver Worksheet

This section explains how to configure a *Standard Driver Worksheet* (or Communication table) to associate application tags with the PLC addresses. You can configure multiple *Driver* worksheets — each of which is divided into a *Header* section and *Body* section.

Use the following steps to create a new *Standard Driver* worksheet:

1. From the Studio development environment, select the **Comm** tab, located below the *Workspace* pane.
2. In the *Workspace* pane, expand the *Drivers* folder, and right-click the SYMAX subfolder.
3. When the pop-up menu displays, select the **Insert** option:

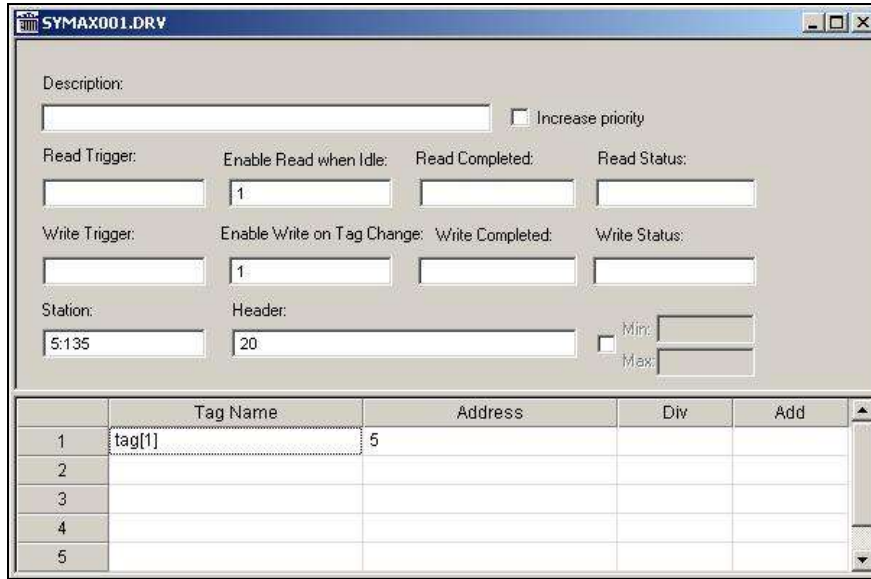


Inserting a New Worksheet

Note:

To optimize communication and ensure better system performance, you must tie the tags in different driver worksheets to the events that trigger communication between each tag group and the period in which each tag group must be read or written. Also, we recommend configuring the communication addresses in sequential blocks to improve performance.

The SYMAX001.drv dialog displays (similar to the following figure):



SYMAX Driver Worksheet

In general, all parameters on the *Driver* worksheet (except the **Station**, **Header** and **Address** fields) are standard for all communication drivers, and they will not be discussed in this publication. For detailed information about configuring the standard parameters, consult the *Studio Technical Reference Manual*.

4. Use the following information to complete the **Station**, **Header**, and **Address** fields on this worksheet:

- **Station** field: Use this field to specify the Drops.
 Syntax:

`<Drop #1>:<Drop #2>: ... :<Drop #n>`

- **Header** field: Use the information in the following table to define the type of registry that will be read from or written. **RegistryReference** is the initial registry (reference) of the configured group.
 Syntax:

`<RegistryReference>`

- **Address** field: Use the information provided in the following table to associate each tag to its respective device registry.
 Syntax:

`<Registry>.<bit>`

Where:

- **Registry**: is a parameter added to the **RegistryReference** parameter.
- **Bit**: is the bit number (from 0 – 7) from the **Registry** address. This parameter is optional.

Sample Configuration			
Station	Header Field	Address Field	Device Registry
5 : 123	20	0	20
		1	21
		2.2	22,bit 3

➔ Attention:
You must not configure a range of addresses in the same worksheet that is greater than the maximum block size (data buffer length) supported by the protocol. The maximum data buffer length for this driver is 255 lines in each *Standard Driver* worksheet.

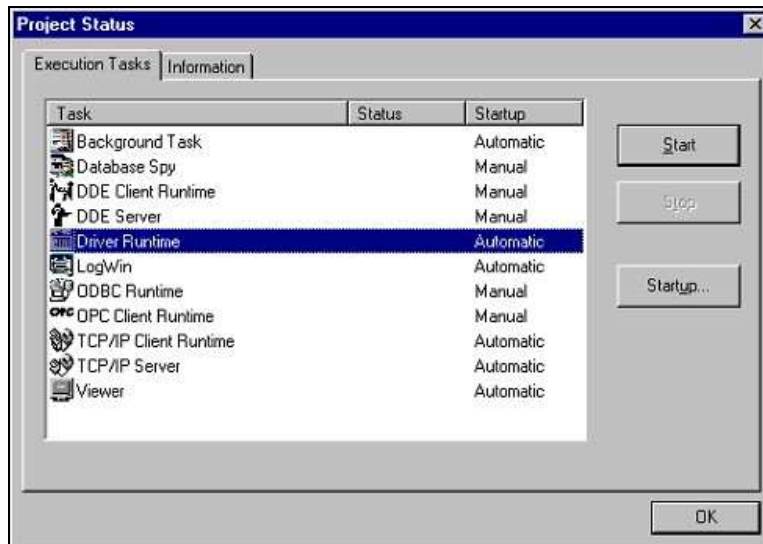
Executing the Driver

After adding the SYMAX driver to a project, Studio sets the project to execute the driver automatically when you start the run-time environment.

To verify that the driver run-time task is enabled and will start correctly, perform the following steps:

1. Select **Project** → **Status** from the main menu bar.

The *Project Status* dialog displays:



Project Status Dialog

2. Verify that the *Driver Runtime* task is set to **Automatic**.
 - If the setting is correct, click **OK** to close the dialog.
 - If the **Driver Runtime** task is set to **Manual**, select the **Driver Runtime** line. When the **Startup** button becomes active, click the button to toggle the *Startup* mode to **Automatic**.
3. Click **OK** to close the *Project Status* dialog.
4. Start the application to run the driver.

Troubleshooting

If the SYMAX driver fails to communicate with the device, the tag you configured for the **Read Status** or **Write Status** fields will receive an error code. Use this error code and the following table to identify the failure.

Error Code	Description	Possible Causes	Procedure to Solve
-15	Timeout Start Message	<ul style="list-style-type: none"> ▪ Disconnected cables ▪ PLC is turned off, in stop mode, or in error mode ▪ Wrong station number ▪ Wrong RTS/CTS control settings 	<ul style="list-style-type: none"> ▪ Check cable wiring. ▪ Check the PLC state – it must be RUN. ▪ Check the station number. ▪ Check the RTS/CTS configuration (see <i>Studio Technical Reference Manual</i> for valid configurations).
-17	Timeout between rx char.	<ul style="list-style-type: none"> ▪ PLC in stop mode or in error mode ▪ Wrong station number ▪ Wrong parity ▪ Wrong RTS/CTS configuration settings 	<ul style="list-style-type: none"> ▪ Check cable wiring. ▪ Check the PLC state – it must be RUN. ▪ Check the station number. ▪ Check the RTS/CTS configuration (see “Link Characteristics” for valid RTS/CTS configurations).
0	No error	Communication without problems	None required.
5	Invalid Protocol	The driver received an invalid message	Verify that specified address or header exists on the device.
12	Invalid block size	Worksheet range is greater than the maximum allowed.	<ul style="list-style-type: none"> ▪ Create a new worksheet. ▪ Typically, the maximum line amount for a worksheet is 255.

⇒ **Tip:**
 You can verify communication status using the Studio development environment *Output* window (*LogWin* module). To establish an event log for **Field Read Commands**, **Field Write Commands** and **Serial Communication**, right-click in the *Output* window. When the pop-up menu displays, select the option to set the log events. If you are testing a Windows CE target, you can enable the log at the unit (**Tools** → **LogWin**), and verify the `celog.txt` file created at the target unit.

If you are unable to establish communication with the PLC, try to establish communication between the PLC Programming Tool and the PLC. Quite frequently, communication is not possible because you have a hardware or cable problem, or a PLC configuration error. After successfully establishing communication between the device's Programming Tool and the PLC, you can retest the supervisory driver.

To test communication with Studio, we recommend using the sample application provided rather than your new application.

If you must contact us for technical support, please have the following information available:

- **Operating system** (type and version): To find this information, select **Tools** → **System Information**.
- **Project Information**: To find this information, select **Project** → **Status**.
- **Driver version and communication log**: Displays in the Studio *Output* window when the driver is running.
- **Device model and boards**: Consult the hardware manufacturer's documentation for this information.

Sample Application

A sample application is provided in the `/COMMUNICATION EXAMPLES/SYMAX` directory. We strongly recommend that you use this sample application to test the SYMAX driver before configuring your own, customized application, for the following reasons:

- To better understand the information provided in each section of this document.
- To verify that your configuration is working satisfactorily.
- To certify that the hardware used in the test (device, adapter, cable and PC) is working satisfactorily before you start configuring your customized applications.



Note:

This application sample is not available for all drivers.

Use the following procedure to perform the test:

1. Configure the device's communication parameters using the manufacturer's documentation.
2. Open and execute the sample application.
3. Execute the *Viewer* module in Studio to display information about the driver communication.



Tip:

You can use the sample application screen as the maintenance screen for your custom applications.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of Changes
A	1.04	Eric Vigiani	15 Jun 04	First version
B	1.05	Leandro Coeli	15 Dec 04	Fixed bug of 0xFE received values that had been discarded.
C	1.06	Leandro Coeli	25 Jan 05	Fixed bug of received values that had been shown correctly in the tags.