

FATEK Communication Driver

Driver for Serial and TCP/IP
Communication with FATEK PLC

Contents

INTRODUCTION2

GENERAL INFORMATION.....3

 DEVICE SPECIFICATIONS.....3

 NETWORK SPECIFICATIONS.....3

 DRIVER CHARACTERISTICS3

 CONFORMANCE SERIAL TESTING4

 CONFORMANCE TCP/IP TESTING.....4

SELECTING THE DRIVER5

CONFIGURING THE DRIVER6

 CONFIGURING THE COMMUNICATION SETTINGS6

 CONFIGURING THE STANDARD DRIVER WORKSHEET8

EXECUTING THE DRIVER 12

TROUBLESHOOTING 13

REVISION HISTORY..... 14

Introduction

The FATEK driver enables communication between the Studio system and all communication ports of FATEK PLC that work under the standard mode, according to the specifications discussed in this document.

This document will help you to select, configure and execute the FATEK driver, and it is organized as follows:

- **Introduction:** This section, which provides an overview of the document.
- **General Information:** Identifies all of the hardware and software components required to implement communication between the Studio system and the target device.
- **Selecting the Driver:** Explains how to select the FATEK driver in the Studio system.
- **Configuring the Device:** Describes how the FATEK PLC must be configured to receive communication from the FATEK driver.
- **Configuring the Driver:** Explains how to configure the FATEK driver in the Studio system, including how to associate database tags with device registers.
- **Executing the Driver:** Explains how to execute the FATEK driver during application runtime.
- **Troubleshooting:** Lists the most common errors for this driver, their probable causes, and basic procedures to resolve them.
- **Revision History:** Provides a log of all changes made to the driver and this documentation.

Notes:

- This document assumes that you have read the “Development Environment” chapter in Studio’s *Technical Reference Manual*.
- This document also assumes that you are familiar with the Microsoft Windows NT/2000/XP environment. If you are not familiar with Windows, then we suggest using the **Help** feature (available from the Windows desktop **Start** menu) as you work through this guide.

General Information

This chapter identifies all of the hardware and software components required to implement serial and TCP/IP communication between the FATEK driver in Studio and a target device using the FACON-PLC protocol.

The information is organized into the following sections:

- Device Specifications
- Network Specifications
- Driver Characteristics
- Conformance Serial Testing
- Conformance TCP/IP Testing

Device Specifications

To establish communication, your target device must meet the following specifications:

- **Manufacturer:** FATEK Automation Corp.
- **Compatible Equipment:**
 - FATEK PLC and any other device that is compatible with the FACON-PLC protocol.
- **Programmer Software:** WinProLadder

For a description of the device(s) used to test driver conformance, see “Conformance Serial Testing” on page 3.

Network Specifications

To establish communication, your device network must meet the following specifications:

- **Physical Protocol:** RS232 / Ethernet
- **Logic Protocol:** FACON-PLC protocol
- **Device Runtime Software:** None
- **Specific PC Board:** None / Ethernet

Driver Characteristics

The FATEK driver package consists of the following files, which are automatically installed in the **/DRV** subdirectory of Studio:

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

 **Note:**

You must use Adobe Acrobat® Reader™ to view the **FATEK.PDF** document. You can install Acrobat Reader from the Studio installation CD, or you can download it from Adobe’s Web site.

The FATEK driver requests the **DrvStd.DLL (with version higher or equal 1.07)** into the **/BIN** folder.

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

- Windows NT/2000/XP

- Windows CE

For a description of the operating systems used to test driver conformance, see “Conformance Testing” below. The FATEK driver supports the following registers:

Register Type	Length	Write	Read	Bit	Word	Float	DWord
X	1 Byte	•	•	•	-	-	-
Y	1 Byte	•	•	•	-	-	-
M	1 Byte	•	•	•	-	-	-
S	1 Byte	•	•	•	-	-	-
T	1 Byte	•	•	•	-	-	-
C	1 Byte	•	•	•	-	-	-
TMR	4 Bytes	•	•	•	•	•	•
CTR	4 Bytes	•	•	•	•	•	•
HR	4 Bytes	•	•	•	•	•	•
DR	4 Bytes	•	•	•	•	•	•

Conformance Serial Testing

The following hardware/software was used for conformance testing:

- Configuration:
 - Baud Rate: 57600
 - Data Bits: 7
 - Stop Bits: 1
 - Parity: Even
 - COM Port: COM1

Driver Version	Studio Version	Operating System (development)	Operating System (target)	Equipment
1.00	6.1 + SP2	WinXP+SP2	WinXP+ SP2	FATEK PLC FB - 14MC

Conformance TCP/IP Testing

The following hardware/software was used for conformance testing:

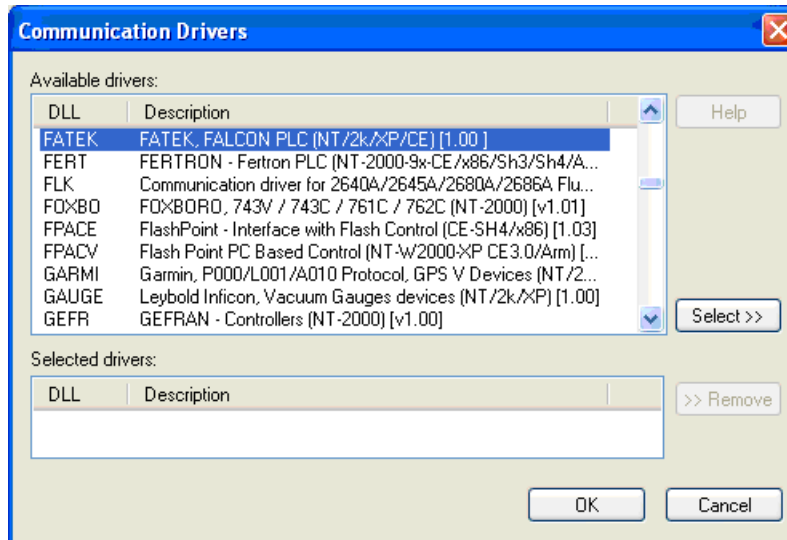
- Configuration:
 - IP Address: 10.168.23.33
 - Port Number: 500

Driver Version	Studio Version	Operating System (development)	Operating System (target)	Equipment
1.00	6.1 + SP2	WinXP+SP2	WinXP+ SP2	FATEK PLC FB - 14MC

Selecting the Driver

When you install Studio, all of the communication drivers are automatically installed in the `\DRV` subdirectory but they remain dormant until manually selected for specific applications. To select the FATEK driver for your Studio application:

1. From the main menu bar, select **Insert** → **Driver** to open the *Communication Drivers* dialog.
2. Select the **FATEK** driver from the *Available Drivers* list, and then click the **Select** button.



Communication Drivers Dialog

3. When the **FATEK** driver is displayed in the **Selected Drivers** list, click the **OK** button to close the dialog. The driver is added to the *Drivers* folder, in the *Comm* tab of the Workspace.

Note:

It is not necessary to install any other software on your computer to enable communication between Studio and your target device. However, this communication can only be used by the Studio application; it cannot be used to download control logic to the device. To download control logic to a FATEK PLC, you must also install the programming software (e.g., WinProLadder). For more information, please consult the documentation provided by the device manufacturer.

Attention:

For safety reasons, you must take special precautions when installing any physical hardware. Please consult the manufacturer's documentation for specific instructions.

Configuring the Driver

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

- Specifying communication parameters (only one configuration needed).
- Defining communication tags and controls in the Communication tables or *Standard Driver*.

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.

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

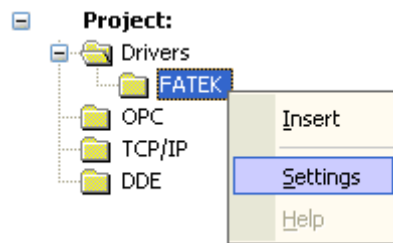
Once you have selected the FATEK driver in Studio, you must properly configure it to communicate with your target device. First, you must set the driver's communication settings to match the parameters set on the device. Then, you must build driver worksheets to associate database tags in your Studio application with the appropriate addresses (registers) on the device.

Configuring the Communication Settings

The communication settings are described in detail in the “Communication” chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

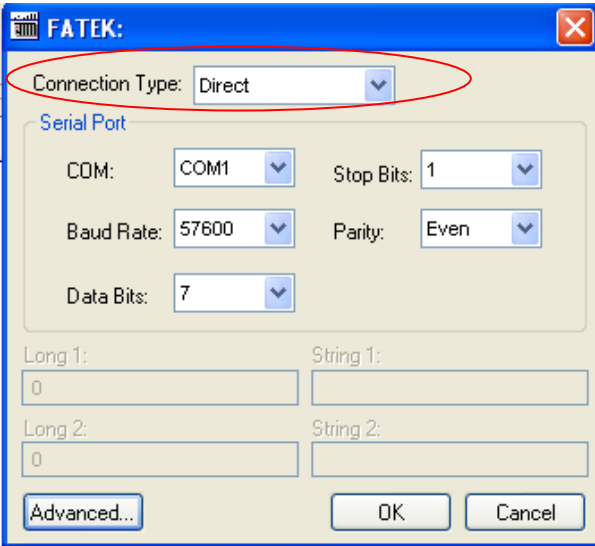
For the purposes of this document, only FATEK driver-specific settings and procedures will be discussed here. To configure the communication settings for the FATEK driver:

1. In the *Workspace* pane, select the *Comm* tab and then expand the *Drivers* folder. The FATEK driver is listed here as a subfolder.
2. Right-click on the *FATEK* subfolder and then select the **Settings** option from the pop-up menu:

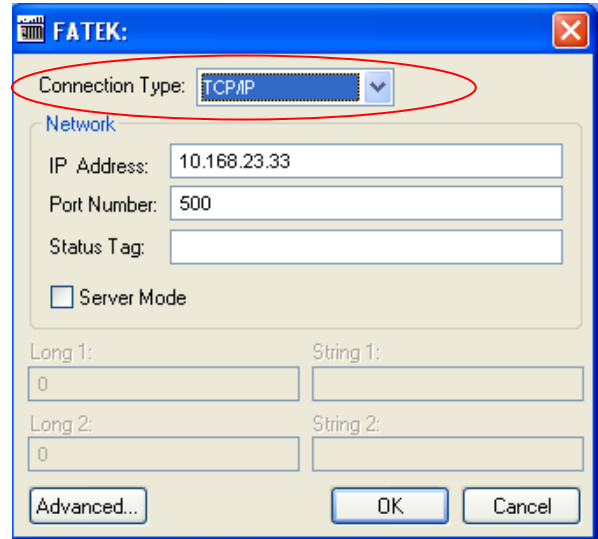


Select Settings from the Pop-Up Menu

The FATEK: Communication Settings dialog is displayed:



Settings for Direct (Serial) Connection Type



Settings for TCP/IP Connection Type

Different communication settings are displayed according to the connection type selected: **Direct** (Serial) or **TCP/IP**. To change the connection to match the FATEK PLC, click on the **Connection Type** combo box and select the correct type from the menu.

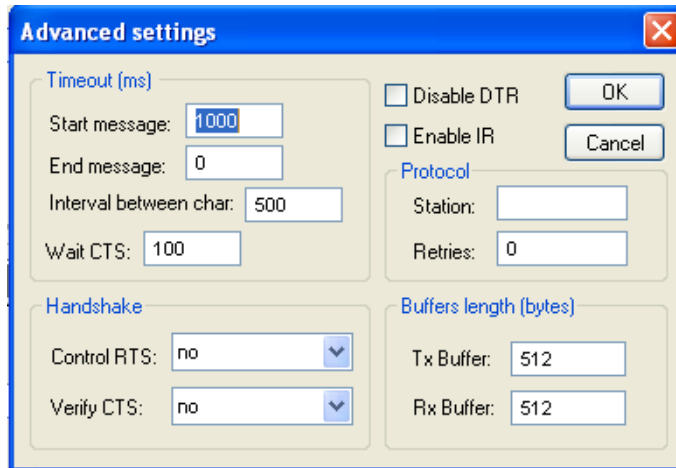
3. This driver supports communicating with multiple PLCs when using the Serial Encapsulation over TCP. On the Driver Settings Window, you can configure at least 1 PLC. Please refer to the **Station** configuration on the driver sheets to see how to configure it communicate with other PLCs that have different IP addresses and/or TCP Port numbers
4. Proceed to configure the remaining settings to enable communication with your target device. To ensure error-free communication, the driver settings must *exactly match* the corresponding settings on the device. Please consult the manufacturer's documentation for instructions how to configure the device and for complete descriptions of the settings.

Depending on your circumstances, you may need to configure the driver *before* you have configured your target device. If this is the case, then take note of the driver settings and have them ready when you later configure the device.

➤ **Attention:**

For safety reasons, you **must** take special precautions when connecting and configuring new equipment. Please consult the manufacturer's documentation for specific instructions.

- If you are using a Data Communication Equipment (DCE) converter (e.g., 232/485) between your PC and your target device, then you must also adjust the **Control RTS** (Request to Send) setting to account for the converter. In the *Communication Settings* dialog, click the **Advanced** button to open the *Advanced Settings* dialog:



Advanced Settings Dialog

When the dialog is displayed, configure the **Control RTS** setting using the following information:

Setting	Default	Values	Description
Control RTS	no	No	Do not set the RTS (Request to Send) handshake signal.
		Yes	Set the RTS (Request to Send) handshake signal before communication. IMPORTANT:
		yes+echo	Set the RTS (Request to Send) handshake signal before communication, and echo the signal received from the target device.

Attention:
 If you incorrectly configure the **Control RTS** setting, then runtime communication will fail and the driver will generate a -15 error. See “Troubleshooting” for more information.

You do not need to change any other advanced settings at this time. You can consult the *Studio Technical Reference Manual* later for more information about configuring these settings.

- Click **OK** to close the *Advanced Settings* dialog, and then click **OK** to close the *Communication Settings* dialog.

Configuring the Standard Driver Worksheet

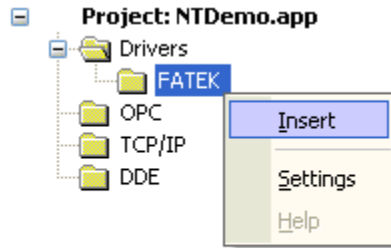
This section explains how to configure a *Standard Driver Worksheet* (or *Communication table*) to associate application tags with the PLC addresses. You can configure multiple worksheets that are each triggered by specific application behaviors. Doing this will optimize communication and improve system performance by ensuring that tags/registers are scanned only when necessary – that is, only when the application is performing an action that requires reading or writing to those specific tags/registers.

The configuration of these worksheets is described in detail in the “Communication” chapter of the *Studio Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

Note:
 We recommend configuring device registers in sequential blocks in order to maximize performance.

For the purposes of this document, only FATEK driver-specific parameters and procedures are discussed here. To insert a new Standard Driver Worksheet:

1. In the *Comm* tab, open the *Drivers* folder and locate the *FATEK* subfolder.
2. Right-click on the *FATEK* subfolder, and then select **Insert** from the pop-up menu:



Inserting a New Worksheet

A new FATEK driver worksheet is inserted into the *FATEK* subfolder, and the worksheet is opened for configuration:

FATEK001.DRV

Description:
 Fatek Test Increase priority

Read Trigger: Enable Read when Idle: Read Completed: Read Status:
 1

Write Trigger: Enable Write on Tag Change: Write Completed: Write Status:
 1

Station: Header: Min:
 1 X:0 Max:

	Tag Name	Address	Div	Add
1	TagA	12		
2	TagB	13		
3	TagC	14		
*				

FATEK Driver Worksheet

Most of the fields on this worksheet are standard for all drivers; see the “Communication” chapter of the *Technical Reference Manual* for more information on configuring these fields. However, the **Station**, **Header**, and **Address** fields use syntax that is specific to the FATEK driver.

3. Configure the **Station** and **Header** fields as follows:

- **Station** field: Use this field to specify the ID of the device. Valid values are 0–254 (no default).

For Ethernet communications using Serial Encapsulation, if you want to connect to a PLC that has an IP Address different than the one configured on the Driver Settings, please follow the below syntax:

<IP_address>:<tcp_port_number>|<device ID>

- **Header** field: Specify the address of the first register of a block of registers on the target device. The addresses declared in the *Body* of the worksheet are simply offsets of this **Header** address. When Read/Write operations are executed for the entire worksheet (see **Read Trigger** and **Write Trigger** above), it scans the entire block of registers from the first address to the last.

The **Header** field uses the following syntax:

<Type>:<AddressReference>

Example — **HR:0**

Where:

- **<Type>** is the register type. Type one of the following: **X, Y, M, S, T, C, TMR, CTR, HR** and **DR**.
- **<AddressReference>** is the initial address (reference) of the register type you configured.

After you edit the **Header** field, Studio checks the syntax to determine if it is valid. If the syntax is invalid, then Studio automatically inserts a default value of **x:0**.

You can also specify an tag (e.g. {**header**}), but the tag that is referenced must follow the same syntax and contain a valid value.

The following table lists all of the data types that are valid for the FATEK driver:

Data Types	Sample Syntax	Comments
X	X:0	Input Discrete
Y	Y:0	Output Relay
M	M:0	Internal Relay
S	S:0	Step Relay
T	T:0	Timer Discrete
C	C:0	Counter Discrete
TMR	TMR:0	Time Register
CTR	CTR:0	Counter Register
HR	HR:0	Data Register
DR	DR:0	Data Register

For each table row (i.e., each tag/register association), configure the **Address** field using the following syntax:

[Format]<AddressOffset>.[Bit]

Examples — **10.5**

Where:

- [**Format**] defines the format of the register. This is an *optional* parameter; if you leave this field blank, Studio provides the data in Word format. The [**Format**] options are as follows:
W: Word format
DW: DWord format
F: Float format
- **<AddressOffset>**: Parameter that is added to the **<AddressReference>** parameter of the **Header**, to compose the specific address of the register in the block.
- [**Bit**] (optional): The bit number (from 0 to 15) of the word address.

➤ Attention:

Keep in mind that when using the **Write Trigger** feature, the driver writes to the entire block of registers from the first address through the last. If there is a register that has not been declared in the worksheet, and its address is within the block, then the register will receive a zero (0) value. Check the worksheet for holes in the address range.

For examples of how device registers are specified using **Header** and **Address**, see the following table:

Device Register	Header Field	Address Field	Valid Range of Initial Addresses 16-bits
X1	X : 0	1	WX0000 ~ WX9984
Y2	Y : 1	1	WY0000 ~ WY9984
M3	M : 3	0	WM0000 ~ WM9984
S4	S : 0	4	WS0000 ~ WS9984
T5	T : 0	5	WT0000 ~ WT9984
C6	C : 0	6	WC0000 ~ WC9984
RT:10	TMR : 2	8	RT0000 ~ RT9999
RC:11	CTR : 0	11	RC0000 ~ RC9999
R:12	HR : 0	12	R00000 ~ R65535
D:13	DR : 0	13	D00000 ~ D65535

➤ Attention: You are not permitted to configure a range of addresses greater than the maximum block size (register's amount) supported by each PLC (as follows) in the same worksheet:

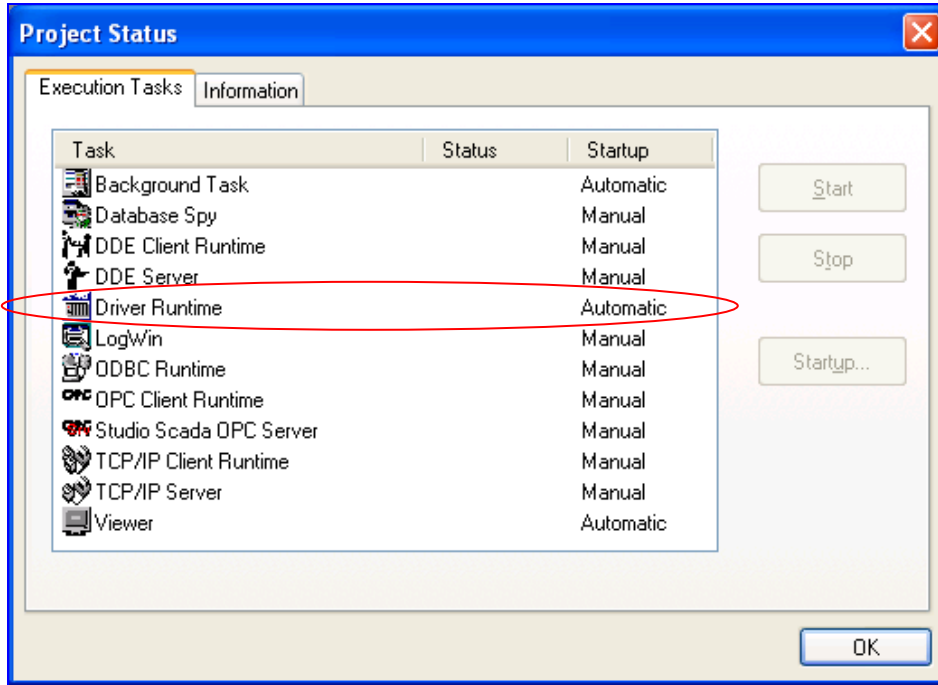
- For X, Y, M, S, T and C: Maximum amount is 255 registers.
- For RT, RC, R and D: Maximum amount is 64 registers.

Executing the Driver

By default, Studio will automatically execute your selected communication driver(s) during application runtime. However, you may verify your application's runtime execution settings by checking the *Project Status* dialog.

To verify that the the communication driver(s) will execute correctly:

1. From the main menu bar, select **Project** → **Status**. The *Project Status* dialog displays:



Project Status Dialog

2. Verify that the *Driver Runtime* task is set to **Automatic**.
 - If the setting is correct, then proceed to step 3 below.
 - If the **Driver Runtime** task is set to **Manual**, then select the task and click the **Startup** button to toggle the task's *Startup* mode to **Automatic**.
3. Click **OK** to close the *Project Status* dialog.
4. Start the application to run the driver.

Troubleshooting

If the FATEK driver fails to communicate with the target device, then the database tag(s) that you configured for the **Read Status** or **Write Status** fields of the Main Driver Sheet will receive an error code. Use this error code and the following table to identify what kind of failure occurred.

Error Code	Description	Possible Causes	Procedure to Solve
0	OK	Communication without problems	None required
1	Invalid Value	Specified invalid value (ex. Decimal format contains hexadecimal number)	Specify a valid value
2	Invalid Operation	Headers might have been inappropriately configured	Check the headers configuration
3	Invalid Block Size	The worksheet is too big and the message cannot be framed	Change offsets or create a new worksheet.
-17	Timeout between rx char.	<ul style="list-style-type: none"> - PLC in stop or error mode - Wrong station number - Wrong parity - Wrong RTS/CTS configuration settings 	<ul style="list-style-type: none"> - Check the cable wiring - Check the PLC state. It must be RUN - Check the station number. - Check the right configuration. See on the section 2.2 the different RTS/CTS valid configurations.
-15	Timeout Start Message	<ul style="list-style-type: none"> - Disconnected cables - PLC turned off, or in Stop or error mode - Wrong Station number - Wrong RTS/CTS control settings. 	<ul style="list-style-type: none"> - Check the cable wiring - Check the PLC state. It must be RUN - Check the station number. <p>Check the right configuration. See on the section 2.2 the different RTS/CTS valid configurations.</p>

⇒ **Tip:**

You can monitor communication status by establishing an event log in Studio's *Output* window (*LogWin* module). To establish a log for **Field Read Commands**, **Field Write Commands** and **Serial Communication**, right-click in the *Output* window and select the desired options from the pop-up menu.

You can also use the *LogWin* module (**Tools** → **LogWin**) to establish an event log on a remote unit that runs Windows CE. The log is saved on the unit in the `ce1log.txt` file, which can be downloaded later.

If you are unable to establish communication between Studio and the target device, then try instead to establish communication using the device's own programming software (e.g., WinProLadder). Quite often, communication is interrupted by a hardware or cable problem or by a device configuration error. If you can successfully communicate using the programming software, then recheck the driver's communication settings in Studio.

To test communication between Studio and the device, 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.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of Changes
A	1.0	Diego Barros	Jan/13/2006	<ul style="list-style-type: none">Initial versionDriver available for Windows NT/2000/XP/CE
B	1.0	Michael D. Hayden	Jun/15/2006	<ul style="list-style-type: none">Edited for language and usability.
C	1.0	Arthur de S. Allievi	Aug/18/2006	<ul style="list-style-type: none">Fixed Bit error on Item Writing.
D	2.0	Fellipe Petermella	Jun/19/2009	<ul style="list-style-type: none">Modified driver to support communication with more than one device simultaneously via TCP/IP
E	2.0	Andre Bastos	Sep/12/2013	<ul style="list-style-type: none">Updated to documentation to make it easier to understand how to configure the communication with multiple PLCs through TCP/IP encapsulation