STRIO Communication Driver

Driver RIO Adapter for
SST 5136-SD-104 and 5136-SD-PCI Boards

Contents

Introduction

The STRIO driver enables communication between the Studio system and devices using the RIO protocol communicating through the SST 5136-SD-104 or 5136-SD-PCI boards, according to the specifications discussed in this document.

This document was designed to help you install, configure and execute the STRIO driver to enable communication with these devices. The information in this document is organized as follows:

- **Introduction**: Provides an overview of the STRIO driver documentation.
- **General Information**: Provides information needed to identify all the required components (hardware and software) used to implement communication between Studio and the STRIO driver.
- **Installing the Driver**: Explains how to install the STRIO driver.
- **Configuring the Driver**: Explains how to configure the STRIO 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 STRIO 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 Studio *Technical Reference Manual*.
- This document also assumes that you are familiar with the Windows XP environment. If you are unfamiliar with Windows 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 Studio STRIO driver and the PLC.

The information is organized into the following sections:

- Device Characteristics
- Link Characteristics
- Driver Characteristics

Device Characteristics

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

- **Manufacturer**: Allen Bradley
- **Compatible Equipment**:
  – PLC2, PLC3 and PLC5 family

For a list of the devices used for conformance testing, see “Conformance Testing.”

Link Characteristics

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

- **Device Communication Port**: Channel 1 A or B configured for Remote I/O Scanner
- **Physical Protocol**: Remote I/O
- **Logic Protocol**: RIO
- **Device Runtime Software**: SST 5136-SD-104 or 5136-SD-PCI Device Driver
- **Specific PC Board**: SST 5136-SD-104 or 5136-SD-PCI Board

Driver Characteristics

The STRIO driver is composed of the following files:

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

> **Notes:**
> • All of the preceding files are installed in the `/DRV` subdirectory of the Studio installation directory.

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

- Windows XP

Please contact SST if you intend to use this driver with Windows 7 or any 64 bits Windows OS.

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

The STRIO driver supports the following registers for the **Adapter** type:

Register Type	Length	Write	Read	Bit	Integer
O (Output)	1 Word	•	•	•	•
I (Input)	1 Word	–	•	•	•
BTR (Block Transfers Read)	1 Word	•	–	•	•
BTW (Block Transfers Write)	1 Word	–	•	•	•

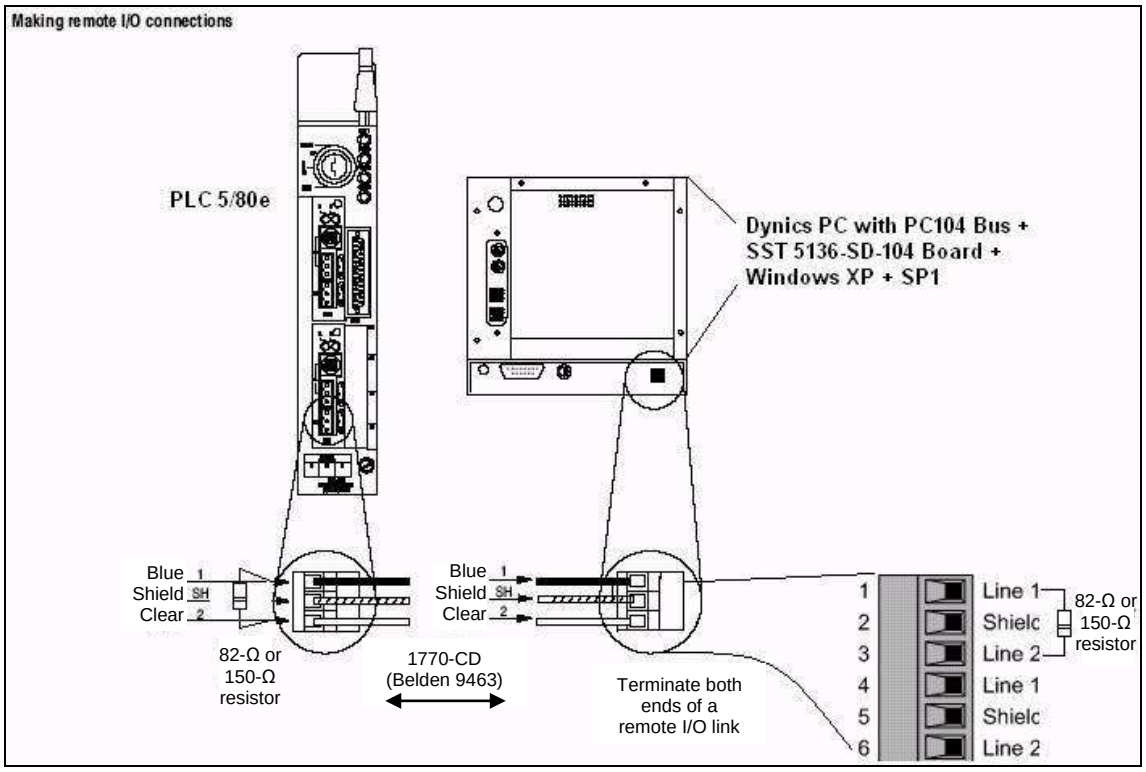
The STRIO driver supports the following registers for the **Scanner** type:

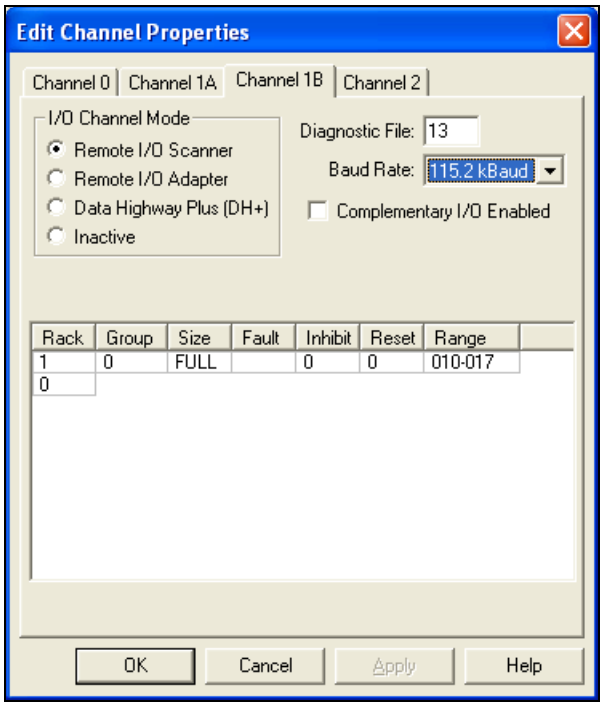
Register Type	Length	Write	Read	Bit	Integer
O (Output)	1 Word	–	•	•	•
I (Input)	1 Word	•	•	•	•
BTR (Block Transfers Read)	1 Word	–	•	•	•
BTW (Block Transfers Write)	1 Word	•	–	•	•

Conformance Testing

The following hardware/software was used for conformance testing:

- **Driver Configuration**:
  - **PLC Program**: FLOATTSTS_ALL.pc5
  - **PLC Port**: Channel 1B
  - **Baud Rate**: 57.6k, 115.2k, 230.4k
  - **Protocol Mode**: Scanner





- **Cable**:
  - **1770-CD (or Belden 9463) cable**. Connect a remote I/O network using a daisy chain or trunk line/drop line configuration.

    For trunk line/drop line configurations, use the 1770-SC station connectors and follow these guidelines:

    - The length of the trunk line cable depends on the communication rate.

- The length of the drop cable is 30.4 m (100 cable-ft).

The maximum cable length for remote I/O depends on the transmission rate. Configure all devices on a remote I/O link to communicate at the same transmission rate.

A remote I/O link using this communication rate…	…cannot exceed this cable length:
57.6 kbps	3,048 m (10,000 ft)
115.2 kbps	1,524 m (5,000 ft)
230.4 kbps	762 m (2,500 ft)

For proper operation, terminate both ends of a remote I/O link by using the external resistors shipped with the programmable controller. Use either a 150_ or 82_ terminator.

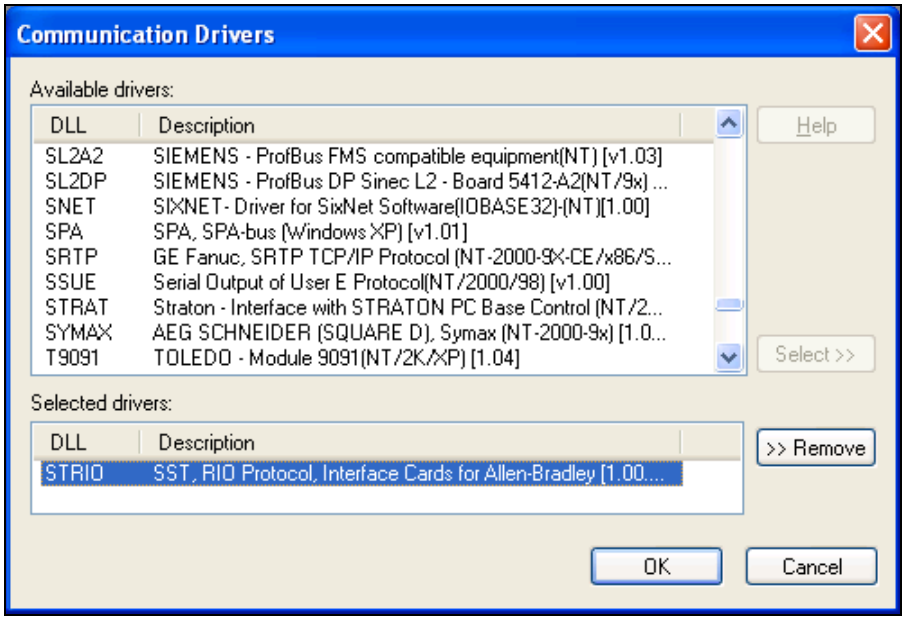
Driver Version	Studio Version	Operating System (development)	Operating System (runtime)	Equipment
1.2	7.0+SP1	WinXP +SP3	▪ WinXP+SP3	▪ **PC**: PC with Windows XP SP3 ▪ **RIO Board**: SST 5136-SD-PCI board ▪ **Field Equipment**: Allen Bradley PLC5/80E. CH1B as RIO

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 **STRIO** driver from the *Available Drivers* list (as shown in the following figure), and then click the **Select** button.



Communication Drivers Dialog Box

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

> **Note:**
> It is necessary to install the board driver software on your computer to enable communication between the PC and the PLC over RIO. Furthermore, to download the PLC application program to your device, you must install RSLogix 5. Consult your Rockwell RSLogix 5 documentation for installation instructions.

> **Attention:**
> For safety reasons, you must use special precautions when installing the physical hardware. Consult the hardware manufacturer's documentation for specific instructions.

Configuring the Driver

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

- Specifying communication parameters
- Defining tags and controls in the *STANDARD DRIVER SHEET*s (or Communication tables)

  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 *MAIN* and *STANDARD DRIVER SHEET*s, and information about configuring the standard fields, review the product's *Technical Reference Manual*.

Setting the Communication Parameters

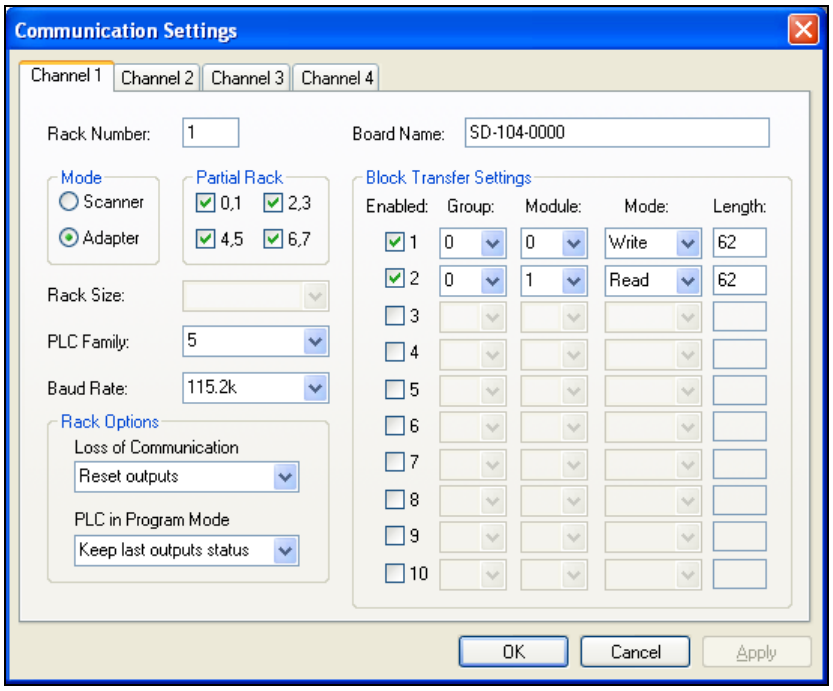
Use the following steps to configure the communication parameters, which are valid for all driver worksheets configured in the system:

From the Studio development environment, select the *Comm* tab located below the *Workspace*.

1. Click on the *Drivers* folder in the *Workspace* to expand the folder.
2. Right-click on the STRIO subfolder. When the pop-up menu displays (as shown in the following figure), select the **Settings** option.

Select Settings from the Pop-Up Menu

The *STRIO:* Communication *Settings* dialog displays (as follows).



Communication Settings Dialog

3. Specify the parameters as noted in the following table:

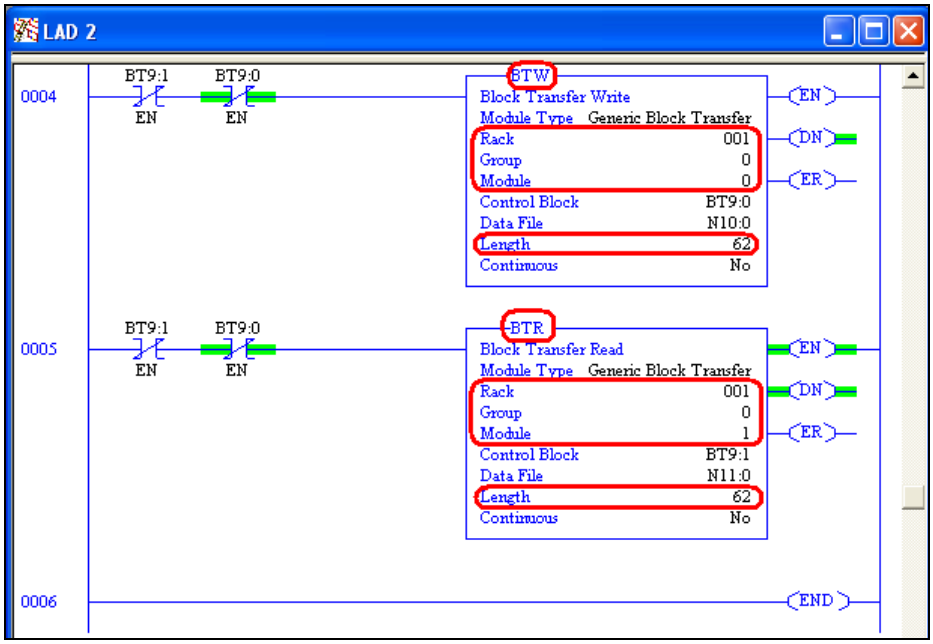
Parameters	Default Values	Valid Values	Description
Rack Number	-	1-37(octal)	Rack Number
Board Name	-	Any	SST Board Name. You can get this name from the SST Installation Program.
Mode	-	Scanner or Adapter	RIO Protocol Mode. If the PLC is *Scanner*, the Board is *Adapter.*
Partial Rack	-	0,1 – 2,3 – 4,5 – 6,7	Partial Rack (only for Adapter Type). It must match the PLC configuration.
Rack Size	-	¼, ½, ¾ or Full Rack	Rack Size (only for Scanner Type).
PLC Family	-	2, 3 or 5	PLC Family
Baud Rate	-	57,6k, 115,2k or 230,4k	Set the Baud Rate.
Loss of Communication	-	Keep last outputs status, Reset outputs	If *Reset Outputs* is selected, DEM clears the outputs to 0 when communication with the PLC is lost.
PLC in Program Mode	-	Keep last outputs status, Reset outputs	If *Reset Outputs* is selected, DEM clears the outputs to 0 when the PLC is in program mode.
BTS - Enabled	-	Check or not	Enable the Block Transfer.
BTS - Group	-	0-7	Block Transfer Group: I/O group number (0-7) which specifies the position of target module in the chassis.
BTS - Module	-	0-1	Block Transfer Module: Enter the slot number (0-1) within the group.

BTS - Mode	-	Read or Write	Block Transfer Mode
BTS - Length	-	0-64	Block Transfer Length

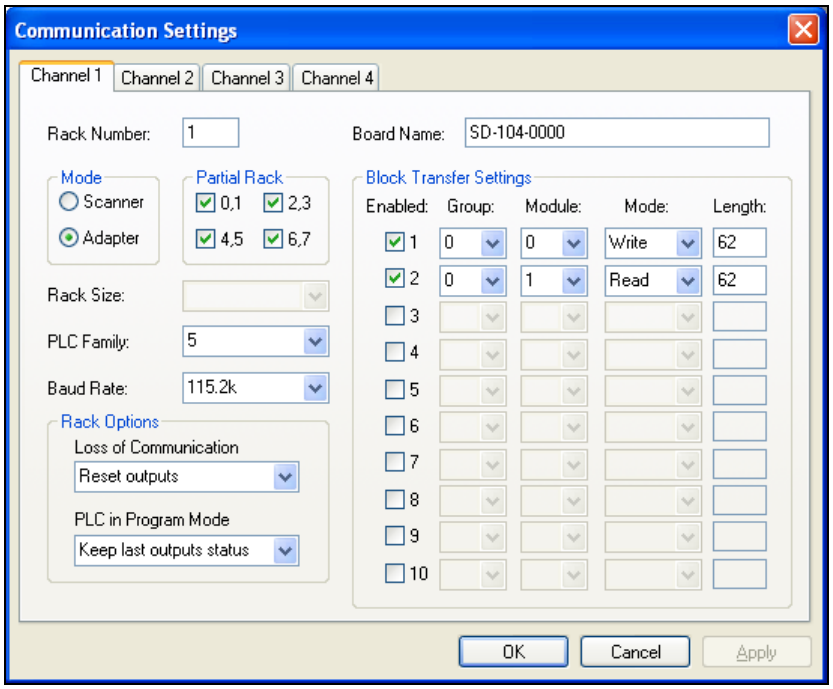
> **Note:**
> The device must be configured with *exactly the same* parameters that you configured in the *STRIO Communication Settings* dialog.

Example

The following example shows the configuration of a Block Transfer Write and a Block Transfer Read block, using RSLogix 5:



In the STRIO driver, the Block Transfer type (Read or Write), Rack, Group, Module and Length need to match exactly the configuration in the PLC. It will be similar to the following:



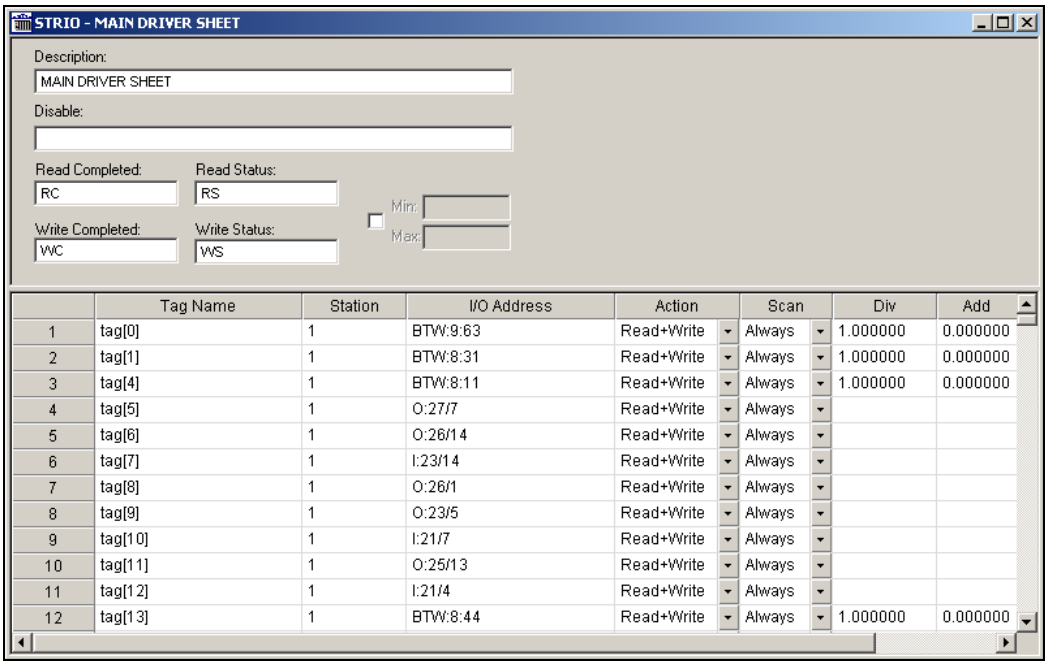
Configuring the Driver Worksheets

This section explains how to configure the *MAIN* and *STANDARD DRIVER SHEETs* (or communication tables) to associate application tags with the device addresses. You can configure multiple Driver Worksheets — each of which is divided into a *Header* section and *Body* section.

Configuring the MAIN DRIVER SHEET
When you add the STRIO driver to your application, Studio automatically adds a *MAIN DRIVER SHEET* to the driver folder, as shown in the following figure:

Main Driver Sheet

You use this worksheet (similar to the following figure) to associate Studio tags to addresses in the PLC.



	Tag Name	Station	I/O Address	Action	Scan	Div	Add
1	tag[0]	1	BTW:9:63	Read+Write	Always	1.000000	0.000000
2	tag[1]	1	BTW:8:31	Read+Write	Always	1.000000	0.000000
3	tag[4]	1	BTW:8:11	Read+Write	Always	1.000000	0.000000
4	tag[5]	1	O:27/7	Read+Write	Always		
5	tag[6]	1	O:26/14	Read+Write	Always		
6	tag[7]	1	I:23/14	Read+Write	Always		
7	tag[8]	1	O:26/1	Read+Write	Always		
8	tag[9]	1	O:23/5	Read+Write	Always		
9	tag[10]	1	I:21/7	Read+Write	Always		
10	tag[11]	1	O:25/13	Read+Write	Always		
11	tag[12]	1	I:21/4	Read+Write	Always		
12	tag[13]	1	BTW:8:44	Read+Write	Always	1.000000	0.000000

Main Driver Sheet

> **Note:**
> Most of the *MAIN DRIVER SHEET* parameters are standard for all drivers, and are not discussed in this document. Instructions for configuring these standard parameters are provided in the Studio *Technical Reference Manual*.

Use the following information to configure the **Station** and **Address** parameters specific to this driver:

- **Station:** Type the PLC Address (ID number).
- **I/O Address:** Type the address of each register in the PLC using the following syntax (for Input and Outputs):

  `<Type>:<Address>/[Bit]` (for example, *O:20, O:20/7*) (for O and I types)

  `<Type>:<BT Number>:<Address>/[Bit]` (for example, BTR*:1:20,* BTR*:1:20/7*) (for BTR and BTW types)

  Where:

  – **`<Type>`***:* Register type (**I, O, BTR or BTW**)

  – **`<BT Number>`:** Block Transfer Number enabled in the *Communication Settings*

  – **`<Address>`**: Address of the device's Input, Output or Memory Area inside a Block Transfer

  – **`<Bit>`** (*optional*): Use this parameter to indicate the bit number to be read or written to the device.

Configuring the *STANDARD DRIVER SHEET*

Use the following steps to create a new *STANDARD DRIVER SHEET*:

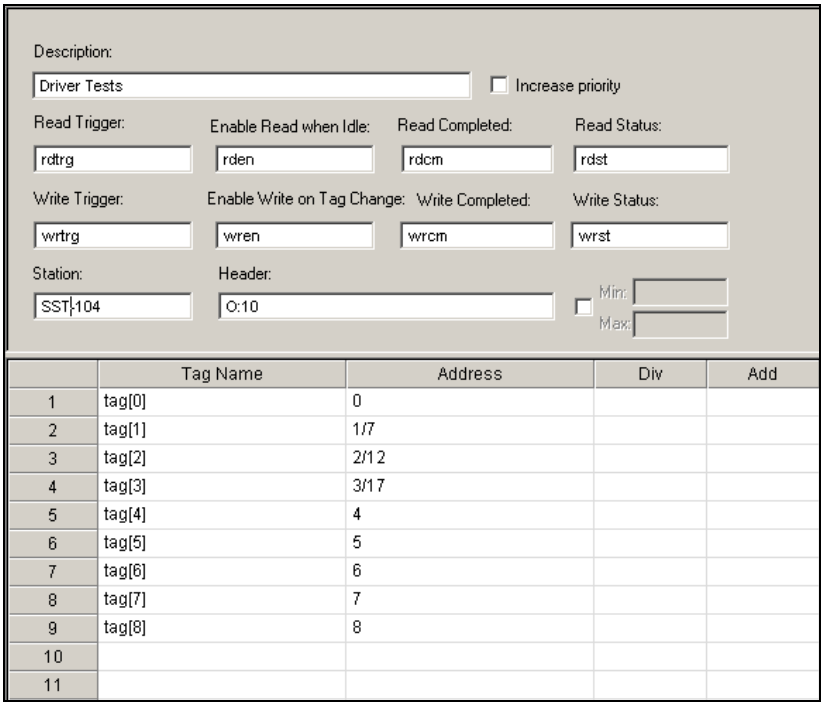
1. From the Studio development environment, select the *Comm* tab, located below the *Workspace* pane.
2. In the *Workspace* pane, expand the *Drivers* folder and right-click the *STRIO* subfolder.

3. When the pop-up menu displays (as shown in the following figure), 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 *STANDARD DRIVER SHEET* displays (similar to the following figure).



	Tag Name	Address	Div	Add
1	tag[0]	0		
2	tag[1]	1/7		
3	tag[2]	2/12		
4	tag[3]	3/17		
5	tag[4]	4		
6	tag[5]	5		
7	tag[6]	6		
8	tag[7]	7		
9	tag[8]	8		
10				
11				

Standard Driver Sheet

In general, all parameters on the Driver Worksheet (except the **Station**, **Header** and **Address** fields) are standard for all communication drivers, but they will not be discussed in this document. 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: Specify the device using the following syntax:

     <CHANNEL NUMBER>

     Where:

     – **<CHANNEL NUMBER>** is the same as that configured in the *Communication Settings*.

   - **Header** field: Use the information in the following table to define the type of variables that will be read from or written to the device, and a reference to the initial address. (Default value is *O:0*)

     These variables must comply with the following syntax:

     <Type>:<Rack Number><AddressReference> (For example: *O:10*) (For O and I types)

     <Type>:<Block Transfer Number> (For example: *BTR:1*) (For BTR and BTW types)

     Where:

     – **<Type>**is the register type (**O**, **I**, **BTR** or **BTW**).
     – **<Rack Number>** is the number of the configured rack.
     – **<AddressReference>** is the initial address (reference) of the O and I types.
     – **<Block Transfer Number>** is the number of the Block Transfer configured in the *Communication Settings*.

   After you edit the **Header** field, Studio checks the syntax to determine if it is valid. If the syntax is incorrect, Studio automatically inserts the default value in the **Header** field.

   Also, you can type a tag string in brackets **{Tag}** into the **Header** field, but you must be certain that the tag's value is correct and that you are using the correct syntax, or you will get an invalid Header error.

   The following table lists all of the data types and address ranges that are valid for the STRIO driver.

Header Field Information		
Data Types	**Sample Syntax**	**Comments**
O	O:10	Discrete Outputs
I	I:21	Discrete Inputs
BTR	BTR:1	Block Transfer Reads
BTW	BTW:4	Block Transfer Writes

   - **Address** field: Use the information in the next table to associate each tag to its respective device address.

     Type the tag from your application database into the **Tag Name** column. This tag will receive values from or send values to an address on the device. The address must comply with the following syntax:

     <AddressOffset>/[Bit] (For example: *4*, *10/3*, *15/7*, *25*)

     Where:

     – **<AddressOffset>** is a parameter added to the **<AddressReference>** parameter (configured in the **Header** field) to compose the group address configured in the **Header** field.
     – **[Bit]** (*optional* parameter used for O and I types [in octal] and BTR and BTW types [in decimal]) is the bit number to be read from or written to the device.

Attention:

- If types are O or I, the Bits must be 0-7 or 10-17 (octal base).
- If types are BTW or BTR, the Bits must be 0-15 (decimal base).

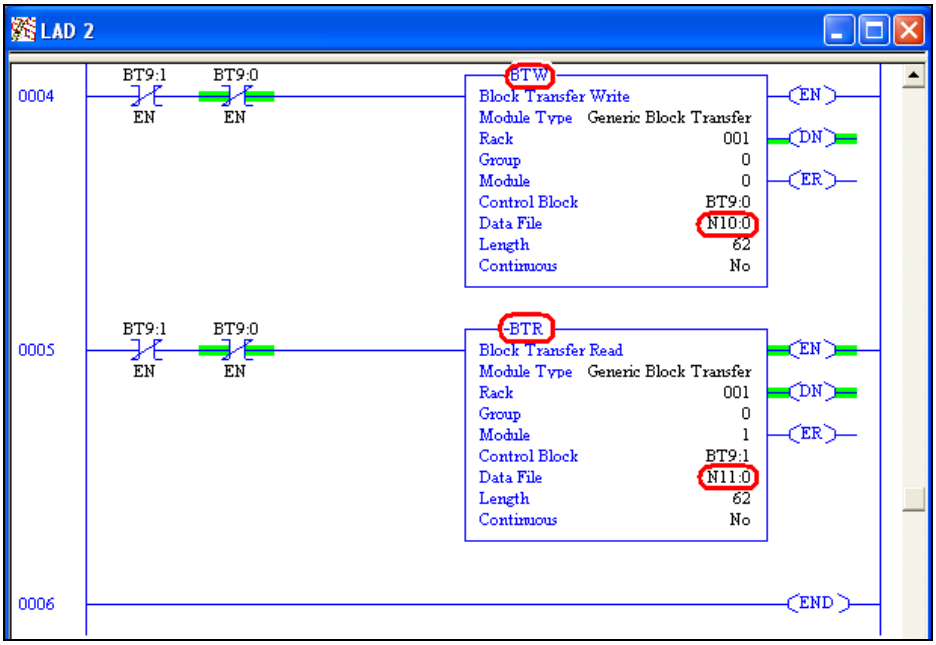
Address Configuration Sample (with Header Type O and Rack 1)		
Device Address	**Header Field**	**Address Field**
O:010	O:10	0
O:012	O:10	2
O:017	O:17	0

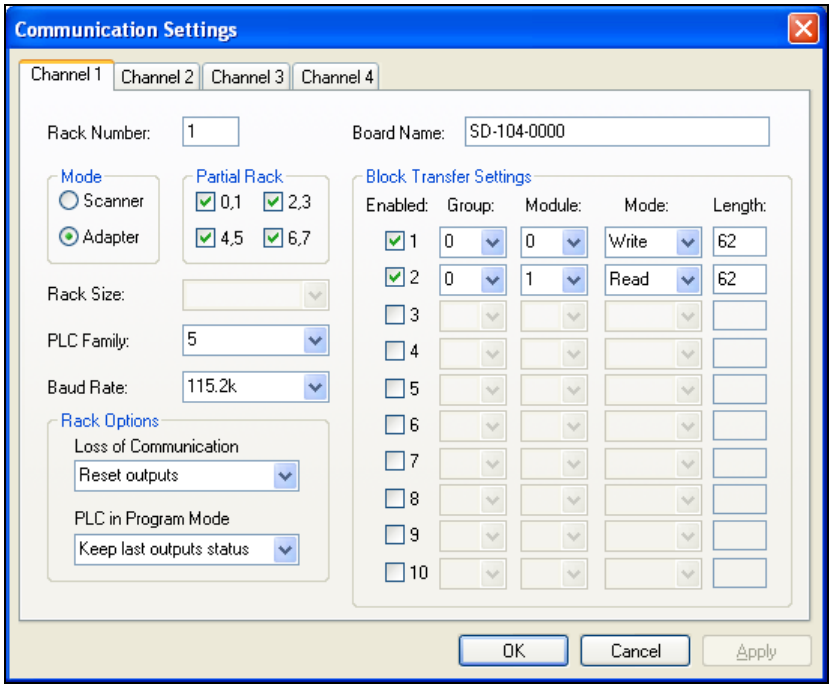
Address Configuration Sample (with Header Type O and Rack 3)		
Device Address	**Header Field**	**Address Field**
O:030/0	O:30	0/0
O:035/17	O:30	5/17
O:032/7	O:32	0/7

Address Configuration Sample (with Header Type I and Rack 2)		
Device Address	**Header Field**	**Address Field**
I:020	I:20	0
I:022/5	I:20	2/5
I:27	I:27	0

Address Configuration Sample (with Header Type I and Rack 3)		
Device Address	**Header Field**	**Address Field**
I:030	I:30	0
I:035/10	I:35	0/10
I:035/10	I:30	5/10

The Block Transfer example below shows the PLC and Driver configuration.





Address Configuration Sample (with Header Type BTR and BT 1)		
Device Address	**Header Field**	**Address Field**
N11:0	BTR::1	0
N11:50/9	BTR:1	50/9
N11:62	BTR:1	62

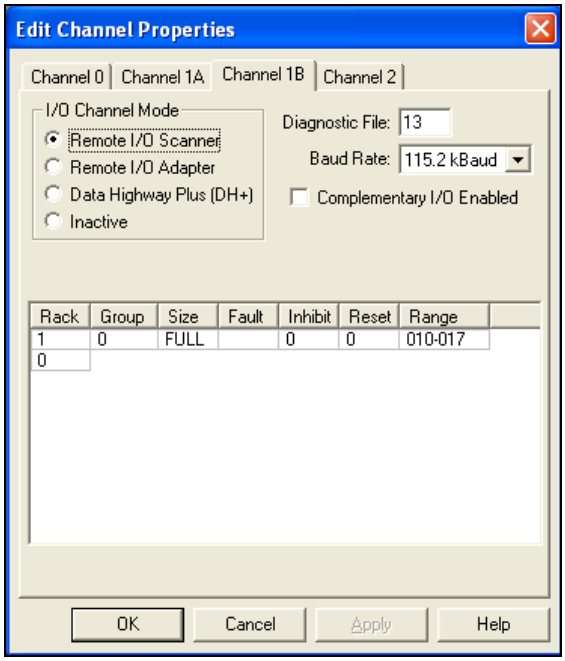
Address Configuration Sample (with Header Type BTW and BT 2)		
Device Address	**Header Field**	**Address Field**
N10:0	BTW:2	0
N10:15/8	BTW:2	15/8
N17:62	BTW:2	62

Attention:

You must not configure a range of addresses that is greater than the maximum supported. Output supports 512 Words, and Input supports 512 Words. The sizes of Block Transfers are configured in the *Communication Settings*.

Device Configuration

Multiple Remote I/O configurations are possible. One example follows:



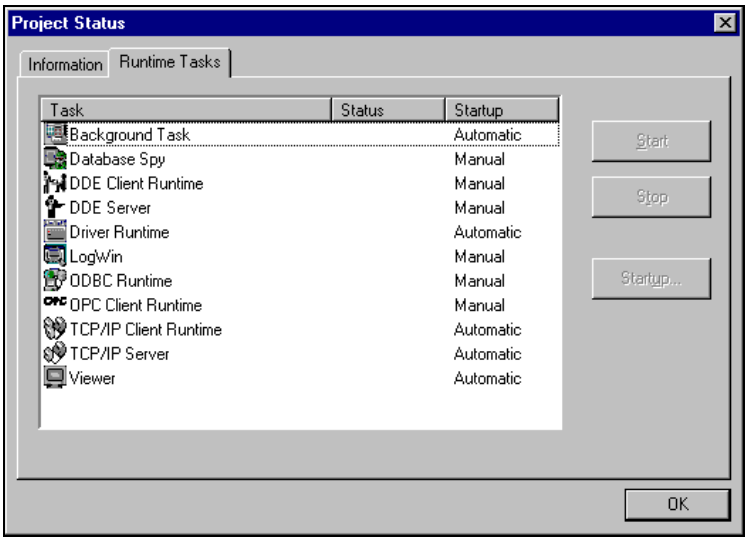
Executing the Driver

After adding the STRIO 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 box displays, as follows.



Project Status Dialog Box

2. Verify that the *Driver Runtime* task is set to **Automatic**.
   - If the setting is correct, click **OK** to close the dialog box.
   - 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 STRIO 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 that occurred.

Error Code	Description	Possible Causes	Procedure to Solve
0	OK	Communication without problems	None required
1	Scanner type writes Input	Attempted writing an Input when Scanner is configured in the *Communication Settings.*	Type a valid header or a valid configuration in the *Communication Settings*
2	Adapter type reads Output	Attempted writing an Output when Adapter is configured in the *Communication Settings*.	Type a valid header or a valid configuration in the *Communication Settings*
3	Initializing API functions	It does not load the API functions.	See if the SST software is installed.
4	Invalid Protocol	Neither Adapter nor Scanner is configured in the *Communication Settings*.	Select a valid configuration in the *Communication Settings*.
5	Invalid Baud Rate	The Baud Rate is not configured in the *Communication Settings*.	Select a valid configuration in the *Communication Settings*.
6	Invalid Rack Size Configuration in Scanner Mode	The Rack Size is not configured in the *Communication Settings*.	Select a valid configuration in the *Communication Settings*.
7	Invalid Family Configuration	The family configured in the *Communication Settings* is different from the family of the connected PLC, or the PLC Family is not configured in the *Communication Settings*.	Select a valid configuration in the *Communication Settings*.
8	Invalid Header field	Invalid value in the **Header** field.	Type a valid rack value in the Header field. The rack value must be the same of the Rack field configured in the *Communication Settings*.
9	Invalid Rack	Invalid Rack is configured.	Type a valid configuration in the *Communication Settings*. The Rack must be between 0 and 37 in octal base.
10	Invalid Mode	Mode is not configured in the *Communication Settings,* but the enabled field is checked.	Select a valid configuration in the *Communication Settings*.
11	DEM not load	DEM module is not present or running.	Check if DEM module is present and running.
12	Invalid Station	The station field value is different from 1, 2, 3 or 4.	The station field must be the same as the Channel Number in the *Communication Settings*.
13	Invalid Writing Block Transfer	▪ Attempted writing when Adapter is configured in the *Communication Settings.* ▪ Attempted reading when Scanner is configured in the *Communication Settings*.	Type a valid configuration in the *Communication Settings*.
14	Invalid Reading Block Transfer	▪ Attempted reading when Adapter is configured in the *Communication Settings.* ▪ Attempted writing when Scanner is configured in the *Communication Settings*.	Type a valid configuration in the *Communication Settings*.
15	Invalid Address	Invalid value in the I/O Address field on the Main Driver Sheet	Type a valid configuration in the *Main Driver Sheet*.
16	Invalid Board Name	The value of the Name in the *Communication Settings* is blank, and the channel is configured in the **Station** field.	Type a valid configuration in the *Communication Settings*.

⇨ **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 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 driver.

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**.
- **Studio version**: To find this information, select **Help** → **About.**
- **Driver Version**: To find this information, read the full description of the driver on the **Communication Drivers** Dialog Box.
- **Communication Log**: Displays in the Studio *Output* window (or *LogWin* window) when the driver is running. Be sure to enable the **Field Read Commands**, **Field Write Commands** and **Serial Communication** for the LogWin window.
- **Device Model** and **Boards**: Consult the hardware manufacturer's documentation for this information.

Sample Application

This driver does not have a sample application.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of changes
A	1.00	Leandro G. Coeli	Dec/8/2004	First version
B	1.01	Leandro G. Coeli	Sep/6/2005	Fixed problems with offsets
C	1.2	Lourenco Teodoro	Jan/15/2012	• Fixed issues with rack number higher than 7 • Fixed issue with bit access higher than 15 • Fixed issues with Bit writing • Fixed stability issues