

SNPP Communication Driver

Driver to send messages to Pagers using
Simple Network Paging Protocol (SNPP)

Contents

INTRODUCTION2

GENERAL INFORMATION.....3

 DEVICE SPECIFICATIONS.....3

 NETWORK SPECIFICATIONS.....3

 DRIVER CHARACTERISTICS3

 CONFORMANCE TESTING4

SELECTING THE DRIVER5

CONFIGURING THE DEVICE6

CONFIGURING THE DRIVER6

 CONFIGURING THE COMMUNICATION SETTINGS6

 CONFIGURING THE DRIVER WORKSHEETS9

EXECUTING THE DRIVER 12

TROUBLESHOOTING 13

SAMPLE APPLICATION 15

REVISION HISTORY..... 16

Introduction

The SNPP driver enables the capability to send messages between the Studio system and pagers using the Simple Network Paging Protocol (SNPP), according to the specifications discussed in this document.

This document will help you to select, configure and execute the SNPP 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 SNPP driver in the Studio system.
- **Configuring the Device:** Describes how the target device must be configured to receive communication from the SNPP driver.
- **Configuring the Driver:** Explains how to configure the SNPP driver in the Studio system, including how to associate database tags with device registers.
- **Executing the Driver:** Explains how to execute the SNPP driver during application runtime.
- **Troubleshooting:** Lists the most common errors for this driver, their probable causes, and basic procedures to resolve them.
- **Sample Application:** Explains how to use a sample application to test the SNPP driver configuration
- **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 Technical Reference Manual*.
- This document also assumes that you are familiar with the Microsoft Windows 2000/XP/Vista 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 communication between the SNPP driver in Studio and a target device using the Ethernet.

The information is organized into the following sections:

- Device Specifications
- Network Specifications
- Driver Characteristics
- Conformance Testing

Device Specifications

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

- **Manufacturer:** Any device using the SNPP over TCP/IP.
- **Device Runtime Software:** None

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

Network Specifications

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

- **Device Communication Port:** Ethernet Port
- **Physical Protocol:** Ethernet / TCP/IP
- **Logic Protocol:** SNPP
- **Specific PC Board:** Any TCP/IP Adapter (Ethernet board)

Driver Characteristics

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

- **SNPP.INI:** Internal driver file. *You must not modify this file.*
- **SNPP.MSG:** Internal driver file containing error messages for each error code. *You must not modify this file.*
- **SNPP.PDF:** This document, which provides detailed information about the SNPP driver.
- **SNPP.DLL:** Compiled driver.

 **Note:**

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

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

- Windows 2000/XP/Vista

For a description of the operating systems used to test driver conformance, see “Conformance Testing” below.

Conformance Testing

The following hardware/software was used for conformance testing:

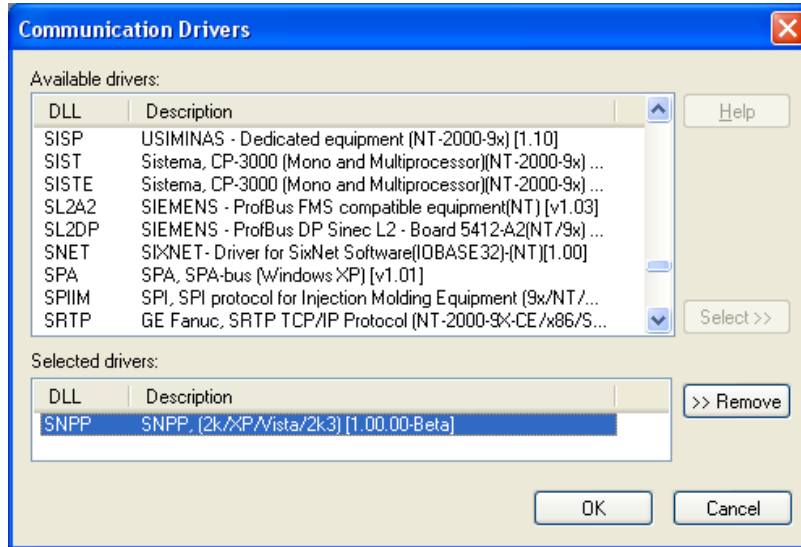
- **Driver Configuration:**
 - **Operation System (development):** Windows XP SP2
 - **Operation System (target):** Windows XP
 - **Studio Version:** 6.1 SP4
 - **Driver Version:** 1.00
 - **Cable:** Ethernet Cable

Driver Version	Studio Version	Operating System (development)	Operating System (runtime)	Equipment
1.00	6.1 + SP4	Win XP + SP2	▪ Win XP+SP2	▪ Pager

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 SNPP driver for your Studio application:

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



Communication Drivers Dialog

3. When the **SNPP** 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.



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 Device

Once the selected driver and the target device are both properly configured, it is not necessary to install any other software on your computer to enable communication between the host and the device. All runtime communication is handled within your Studio application project. However, programming the device itself — that is, developing control logic and downloading it to the device — still requires using the device’s own programming tool.

Configuring the Driver

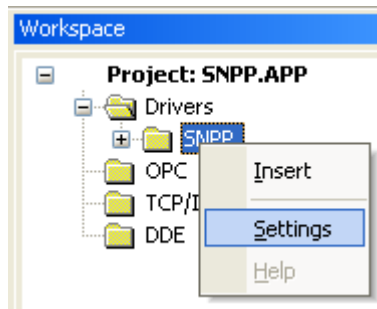
Once you have selected the SNPP driver in Studio, you must properly configure it to communicate with your target device. First, you must set the driver’s communication settings to match the parameters set on the device. Then, you must build driver worksheets to associate database tags in your Studio 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.

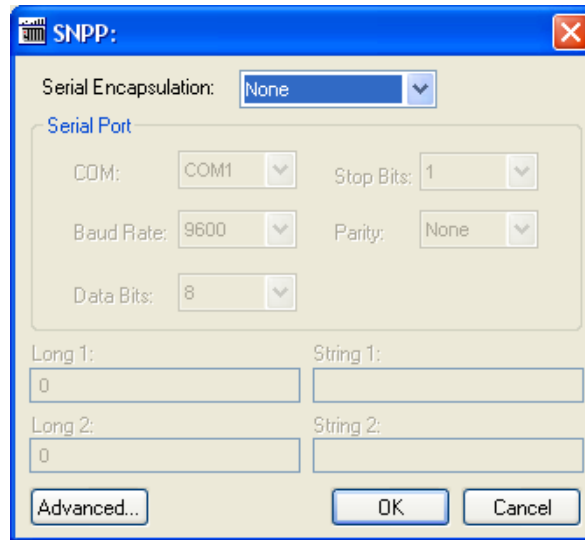
For the purposes of this document, only SNPP driver-specific settings and procedures will be discussed here. To configure the communication settings for the SNPP driver:

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



Select Settings from the Pop-Up Menu

The *SNPP: Communication Settings* dialog is displayed:



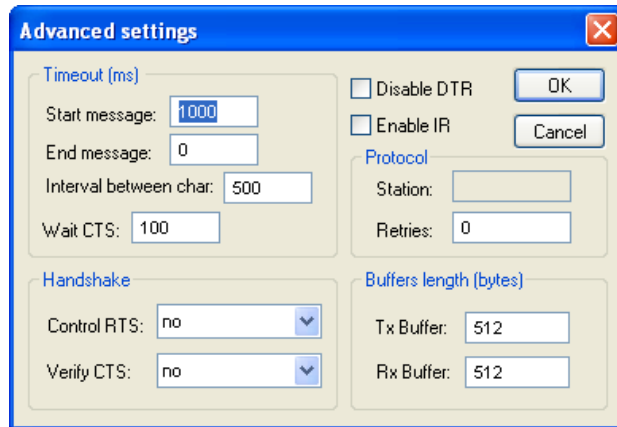
SNPP: Communication Settings Dialog

3. In the *Communication Settings* dialog, configure the driver 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.



Advanced Settings Dialog

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.

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

Configuring the Driver Worksheets

Each selected driver includes a one or more *Standard Driver Worksheets*. *Standard Driver Worksheets* can be inserted to define additional tag/register associations that are triggered by specific application behaviors.

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.

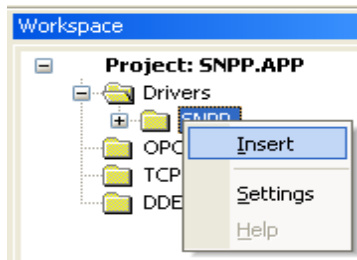
For the purposes of this document, only SNPP driver-specific parameters and procedures are discussed here.

STANDARD DRIVER WORKSHEET

When you select the SNPP driver, you may insert *Standard Driver Worksheets* to define tag/register associations that are 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.

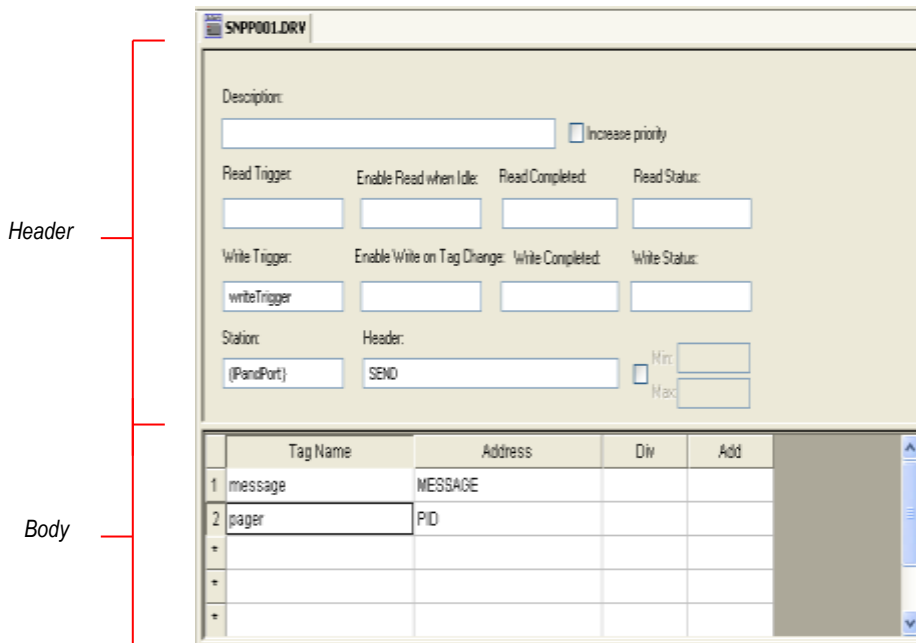
To insert a new *Standard Driver Worksheet*:

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



Inserting a New Worksheet

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



SNPP 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 SNPP driver.

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

- **Station** field: Specify the IP Address of the device and the slot number, using the following syntax:

<IP Address>:[Port Number]

Example — **192.168.125.31:4000**

Where:

- **<IP Address>** is the SNPP server IP Address on your network or Internet.
- **[Port Number]** is the port number for the SNPP TCP protocol. This is an optional parameter. If the parameter is omitted, then the driver assumes 444;

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

➤ **Attention:**

- You must use a non-zero value in the **Station** field, and you cannot leave the field blank.

- **Header** field: should have the text “SEND”

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 **SEND**

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

- **Address** field: For each table row (i.e., each tag/register association), configure the **Address** field using **PID and MESSAGE**.

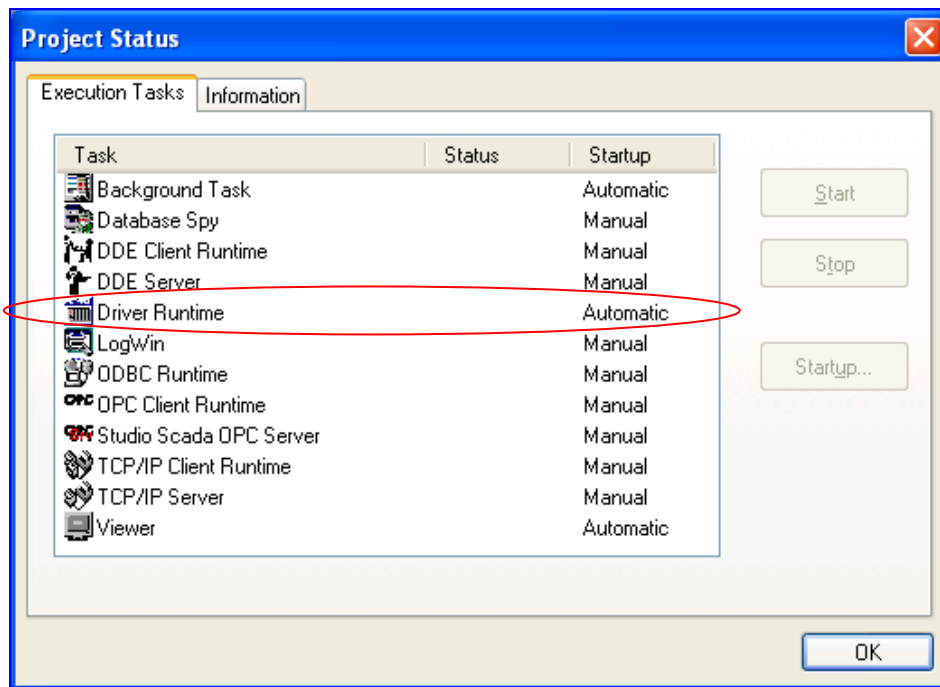
Address	Tag Type	Description
PID	String	The pager's Phone ID
MESSAGE	String	The message to be sent

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 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 SNPPdriver 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 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 PROTOCOL	The message is larger than 80 characters in length	Validate the length of the message being sent
3	INVALID BLOCK SIZE	Too many rows configured on body of the driver worksheet	Make sure only 2 rows are configured per sheet
4	ERROR CONNECTING	The driver could not establish a TCP/IP connection to the server.	Check the cable connection. Check the server IP address and port entered in the station field. Check for a connection to the IP address using PING for the command prompt Try to connect to the server using telnet to the IP address and port
5	INVALID COMMAND	Read trigger Enable Read When Idle or Enable Write on Tag Change are configured	Make sure these fields are blank
6	INVALID RESPONSE	A Not OK response was received from the server	Check the PID and server IP
1005	Timeout Error	The time answer from the equipment was higher than the expected.	Check the cable connection. Check the server IP address and port entered in the station field. Check for a connection to the IP address using PING for the command prompt Try to connect to the server using telnet to the IP address and port

⇒ **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 `ceLog.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. 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.

Sample Application

There was not an official sample application available for this driver by the time that this document was written.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of changes
A	1.00	Mark Duren	May/06/2008	Initial release
A	1.01	Lourenço Teodoro	Dec/16/2008	Updated driver version, no change in the contents.