

Communication Driver WTP2

Driver for serial communication with
Weg TP02 devices using RS-232

Index

1 INTRODUCTION..... 2

2 GENERAL CHARACTERISTICS 3

2.1 DEVICE CHARACTERISTICS 3

2.2 LINK CHARACTERISTICS..... 3

2.3 DRIVER CHARACTERISTICS 4

2.4 INFORMATION ABOUT CONFORMANCE TESTING..... 5

3 INSTALLATION 6

3.1 INSTALLING THE DRIVER 6

3.2 OTHER SOFTWARE REQUIREMENTS 6

4 DRIVER CONFIGURATION 7

4.1 SETTINGS - COMMUNICATION PARAMETERS..... 7

4.2 DRIVER WORKSHEET..... 9

4.3 STATION AND HEADER CONFIGURATION..... 10

4.4 ADDRESS CONFIGURATION 12

4.5 DEVICE CONFIGURATION 13

5 EXECUTION 14

6 TROUBLESHOOTING..... 15

7 APPLICATION SAMPLE..... 17

8 HISTORY OF VERSIONS..... 17

1 Introduction

The WTP2 driver enables communication between Studio system and some of the Weg Tp02 devices using their Host protocol by RS 232, in accordance with the characteristics covered in this document.

This document contains 8 parts, as follow:

- Introduction: Provides an overview of the driver documentation.
 - General characteristics: Provides information necessary to identify all the required components (hardware and software) necessary to implement the communication and global characteristics about the communication.
 - Installation: Explains the procedures that must be followed to install the software and hardware required for the communication.
 - Driver configuration: Provides the required information to configure the communication driver such as the different permutations for configuration and its default values.
 - Execution: Explain the steps to test whether the driver was correctly installed and configured.
 - Troubleshooting: Supplies a list of the most common error codes for this protocol and the procedures to fix them.
 - Application Sample: Provides a sample application for testing the configuration the driver.
 - History of versions: Provides a log of all the modifications done in driver.
- ☞ Note: This document presumes that the user has read the chapter *Driver Configuration* of the Studio's Technical reference manual.

2 General Characteristics

2.1 Device Characteristics

- Manufacturer: WEG
- Compatible Equipment
 - Weg TP02 Series
- Weg PLC programmer software, for example: PC12.

☞ Note: Please refers to section 2.4 to see the Equipment used in the standard conformance tests for this driver.

2.2 Link Characteristics

- Device communication port: MMI port (EIA RS-422A)
- Physical protocol: RS232/RS422
- Logic protocol: Weg Tp02 Host Protocol
- Device Runtime software: None
- Specific PC Board: None
- Adapters / Converters: To run this communication you must use the Weg TP232 PG (RS232/422 adapter).
- Cable Wiring: TP-202MC.
 - **PG Mode:** Signal of PG/COM(Pin 4) must is hi.
 - **Computer Connection Mode:** Signal of PG/COM(Pin 4) must is low. (connect Pin4 with Pin5)

To use the driver WTP2 with Studio the communication mode must be “Computer Connection Mode”.

2.3 Driver Characteristics

- Operating System:
 - Windows 9x
 - Windows 2000
 - Windows NT

☞ Note: Please refer to section 2.4 to see the Operating System used in the conformance tests for this driver.

The driver is composed of the following files:

- WTP2.INI: Internal file of the driver, it should not be modified by the user.
- WTP2.MSG: This file contains the error messages for each error code. It is an internal file of the driver, the user should not modify it.
- WTP2.PDF: This document provides detailed documentation about the driver.
- WTP2.DLL: This is the compiled library for the driver.

☞ Note: All the files above must to be in the subdirectory /DRV of the Studio's installation directory.

2.4 Information about conformance testing

- Equipment: Weg TP02 – 28MR

Configuration:

PLC program: Test
Baud Rate: 19200
Protocol: Weg Tp02 Host Protocol
Data Bits: 7
Stop Bits: 1
Parity: None
COM port: COM2

Response Time: 4

- Cable: According link specification, section 2.2.
- Operating System (development): Windows NT 4.0 + Service pack 6
- Operating System (target): Windows NT 4.0 + Service Pack 6
- Studio Version: 3.0+Service Pack 1
- Driver version: 1.00

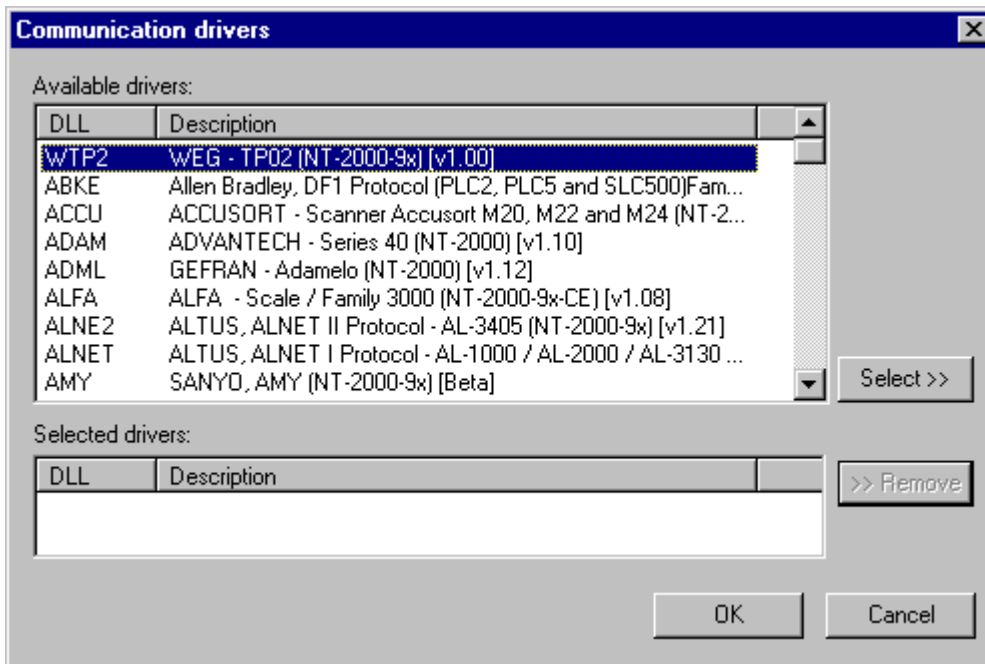
3 Installation

3.1 Installing the Driver

When you install the Studio v3.0 or higher, the communication drivers are already installed. You need now to select the driver at the applications where it will be used.

The steps to select the driver inside an application are:

1. Execute the Studio and select the proper application.
2. Select the menu *Insert + Driver...*
3. In the column **Available Drivers**, select the **WTP2 Driver** and push the button **ADD>>>** (the driver WTP2 must appear in the column **Selected Drivers**).
4. Press **OK**.



3.2 Other software requirements

It is not necessary to install any other software in the PC to enable the communication between the host and the Device. However, to download the custom program to the device, it is necessary to install one of the Weg programmer software, for example, PC12. Please see the PC12 documentation about the procedure to install their software.

☞ Note: Special precautions must be taken when installing the physical hardware. Refer to the hardware manufacturer documentation for specific instructions in this area.

4 Driver Configuration

After the driver is installed and selected in the Studio (see section 3.1), you should proceed to the driver configuration.

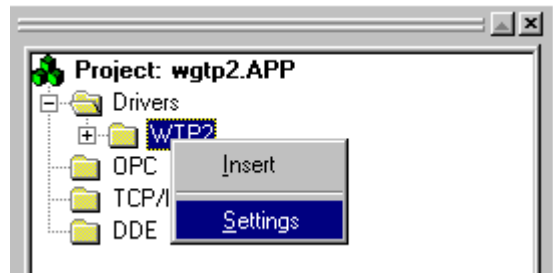
The driver configuration is two parts:

The Settings or Communication parameters, it is only one configuration to the whole driver, then you have the communication tables or Driver Worksheets, where the communication tags are defined.

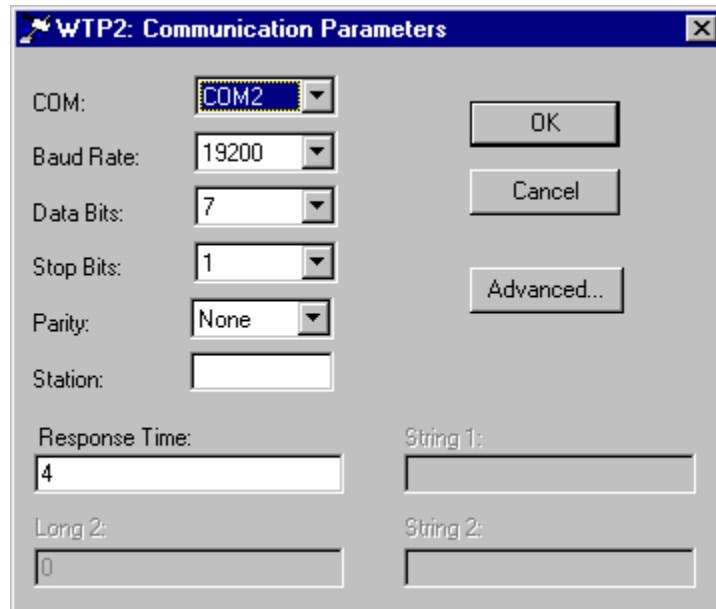
4.1 Settings - Communication Parameters

These parameters are valid for all driver worksheets configured in the system. To open the window for configuring the **Communication parameters**, follow these steps:

1. In the **Workspace** of the Studio environment, select the **Comm** table.
2. Expand the folder **Drivers** and select the subfolder **WTP2**.
3. Right click on the **WTP2** subfolder and select the option **Settings**.



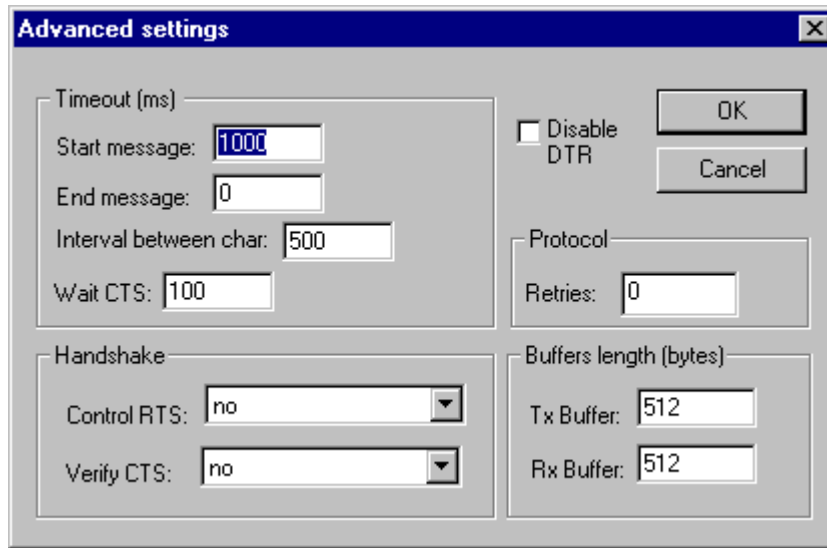
When selecting the Settings, there is the following dialog to configure:



Parameter	Default Value	Valid values	Description
COM	COM1	COM1 to COM8	Serial port of the PC used to communication with the device
Baud Rate	19200	110 to 57600bps	Communication rate of data
Data Bits	7	5 to 8	Number of data bits used in the protocol
Stop Bits	1	1or 2	Number of stop bits used in the protocol
Parity	None	even, odd, none, space or mark	Parity of the protocol
Station	0	0	Not used for the driver
Response Time	4(40ms)	0-0ms 1-10ms 2-20ms 3-30ms . . 14-500ms 15-600ms	Limit time between execution command and response to computer.

Note: These Parameters must be just the same as the configured on the Weg Tp02 device. The Weg Tp02 Programmable Controller Manual indicates the following serial communication settings:
 Baud Rate: 19200, Data bits: 7, Stop bits: 1, Parity: None, Response Time : 4

By clicking on the button **Advanced...** in the window **Communication Parameters**, you open additional communication parameters.



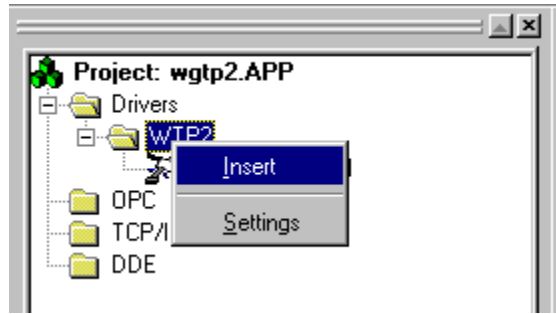
The Advanced setting parameters are explained at the Studio Technical Reference Manual, and you should keep the default values to all fields. Only the field described at the next table should be configured:

Parameter	Default Value	Valid values	Description
Control RTS	No	no, yes or yes + echo	Define if the handshake signal of RTS (Request to Send) is set before communication and if there is an echo in the communication. You must to choose the option "no". Important: Wrong settings on this field will not let the driver work, having the Timeout waiting start a message error.

4.2 Driver Worksheet

It is possible to configure many driver worksheets, each one will be composed of a Header and Body. To create a new driver worksheet, follow these steps:

1. In the **Workspace** of the Studio environment, select the table **Comm**.
2. Expand the folder **Drivers** and select the subfolder **WTP2**.
3. Right click on the **WTP2** subfolder and select the option **I**nsert.



Note: To optimize communication and ensure better performance for the system, it is important to tie the tags in different driver sheets according to the events that must trigger the communication of each group of tags and the periodicity for which each group of tags must be written or read. In addition, it is recommended to configure the addresses of communication in sequential blocks.

When creating a communication table, you have the following window:

Description:
 Increase read priority

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

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

Station: Header: Min:
 Max:

	Tag Name	Address	Div	Add
1	data[1]	1		
2	data[2]	2		
3	data[3]	3		
4	data[4]	4		
5	data[5]	5		
6	data[6]	6		
7	data[7]	7		
8	data[8]	8		
9	data[9]	9		

All entries at the Driver Worksheet, exception by the **Station**, **Header** and **Address** are standard to all communication drivers. You should refer to Studio Communication Driver documentation about the configuration of the standard fields. This document describes the Station, Header and Address fields, which are specific to each communication driver.

4.3 Station and Header configuration

Parameter	Default Value	Valid values	Description
Station	-	00 to 99	The PLC's Address (Decimal value).
Header	MCR:X0000	Vide next table	Defines the type of variable to be read or written from or to the device and the reference of the initial address.

The **Header** field defines the type of variables that will be read or written from or to the device. It complies with the syntax: <Command><type of operand><initial address reference>. After editing the field **Header**, the system will check if it is valid or not. If the syntax were incorrect, the default value (MCR:X0000) will be automatically placed in this field.

You can type Tag between curly brackets into this field, but be sure that the Tag's value is correct, with the right syntax, or you will get the Invalid Header error. The right syntax, both for the field typing and Tag value is described below:

- **Command:** MCR (coil state - Read), MRV(register value - Read), RBP(boolean program - Read), SCS(coil state - Write), WRV(register value - Write) and WBP(boolean program - Write)
- **Type of Operand:** X,Y,C,S (SC), V, D, S(WS), C(WC), F(file)
- **Initial address:** As the communication driver establishes serial communications using messages blocks, this driver has been developed to use an initial address and a block size. So, all you need to do to type a right device's variable address is to type here the initial address and on the *Address* cells (Driver Configuration's Body), just type the offset regarding this initial address. The tables below have a lot of examples how to get the same variable's value using different headers.

Information regarding the parameter "Header"			
Command	Type of Operand	Valid range of initial Address	Comment
MCR (coil state)	X	1 – 384	Read only command. This commands work with bit value. Example : MCR:X0000
	Y	1 – 384	
	C	1 – 2048	
	S (SC)	1 – 128	
MRV (register value)	V	1 – 1024	Read only command. This commands work with word value. Example: MRV:V0000
	D	1 – 2048	
	S(WS)	1 – 128	
	C(WC)	1 – 912	
	F(file)	1 – 130	
RBP (boolean program)	None	0–4000 (TP02– 40MR/60MR) 0-1500(TP02-20MR/28MR)	Read only comamnd. This command work with word value. Example: RBP:0000
SCS (coil state)	Y	1 – 384	Write only command. This command work with bit value. Example: SCS:Y0000
	C	1 – 2048	
	S(SC)	1 – 128	

WRV (register value)	V	1 – 1024	Write only command. This command work with word value. Example: WRV:D0000
	D	1 – 2048	
	S (WS)	1 – 128	
	C (WC)	1 – 912	
	F	1 – 130	
WBP (boolean program)	None	0 – 4000	Write only command. This command work with word value. Example: WBP:0000
PSR (Read PLC Status)	-	-	Read only command. This command will receive : 0 – STOP 1- RUN 2- Error
STP (Stop PLC)	-	-	Write only command.
RUN (Run PLC)	-	-	Write only command.

4.4 Address Configuration

The body of the driver worksheet allows you to associate each tag to its respective address in the device. In the column **Tag Name**, you must type the tag from your application database. This tag will receive or send values from or to an address on the device.

The address cells complies to the following syntax:

<offset>.<bit number>

offset: the **offset** is a number that when added to the **initial address reference** configured in the **Header** field defines an address in the device.

bit: The **bit** is optional and when used it specifies a bit, always from 0 to 15 even in the I/Os, of the address configured.

Sample of Addressing Configuration		
Address on the Device	Header Field	Address Field
X0001(Read)	MCR:X0001	0
X0015(Read)	MCR:X0000	15
X0015(Read)	MCR:X0015	0
C0001(Read)	MCR:C0001	0
C0012(Read)	MCR:C0000	12
C0010(Read)	MCR:C0010	0
V0001(Read)	MRV:V0001	0
V0001(Read)	MRV:V0000	1

V0010(Read)	MRV:V0005	5
WC0001(Read)	MRV:C0001	0
WC00010(Read)	MRV:C0000	10
Y0001(Write)	SCS:Y0001	0
Y0015(Write)	SCS:Y0000	15
Y0015(Write)	SCS:Y0015	0
C0001(Write)	SCS:C0001	0
C0012(Write)	SCS:C0000	12
C0010(Write)	SCS:C0010	0
V0001(Write)	WRV:V0001	0
V0001(Write)	WRV:V0000	1
V0010(Write)	WRV:V0005	5
WC0001(Write)	WRV:C0001	0
WC00010(Write)	WRV:C0000	10
Address 0000(Prog. Bool.) (Read)	RBP:0000	0
Address 0010(Prog. Bool.) (Read)	RBP:0010	0
Address 0010(Prog. Bool.) (Read)	RBP:0000	10
Address 0000(Prog. Bool.) (Write)	WBP:0000	0
Address 0010(Prog. Bool.) (Write)	WBP:0010	0
Address 0010(Prog. Bool.) (Write)	WBP:0000	10
Read PLC Status	PSR	0
Stop PLC	STP	0
Run PLC	RUN	0

Note: In the previous table there are several ways to set the same variable on the device because of the variable's number defined by the sum of the **initial address reference** defined in the field header and the **offset** defined in the Address Field.

4.5 Device Configuration

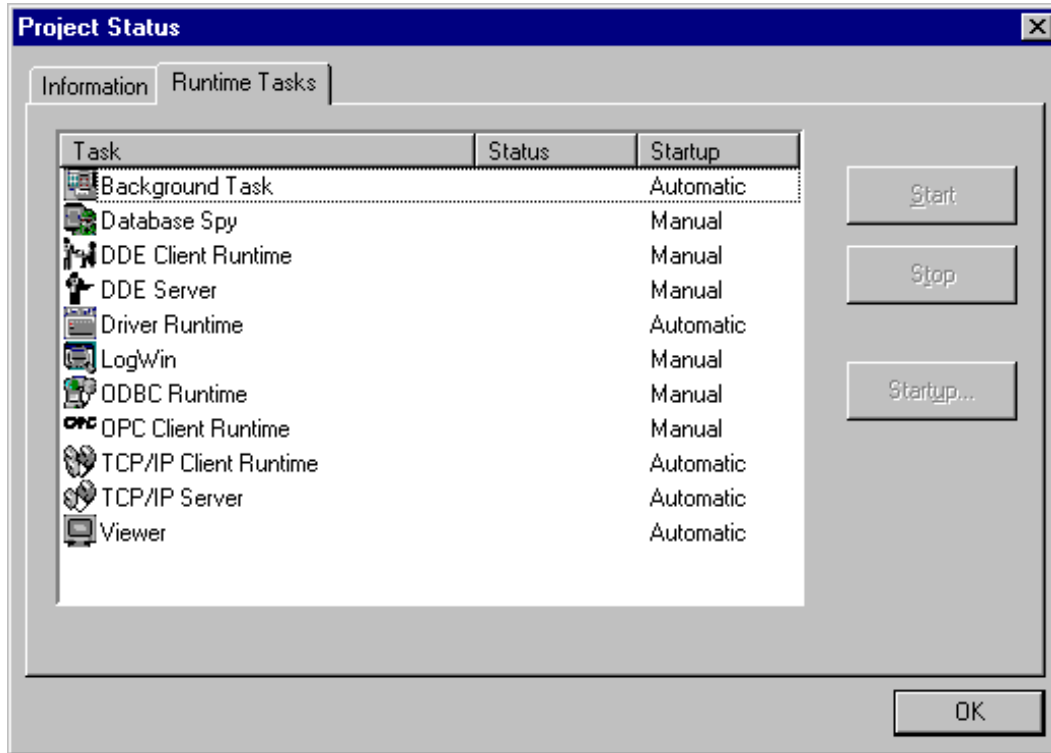
The Weg TP02 Programmable Controller Manual indicates the following serial communication settings:

Baud Rate: 19200
 Data bits: 7
 Stop bits: 1
 Parity: None
 Response Time: 4

Besides that, it is required to select the PLC station number, using the PLC programming tools.

5 Execution

- When installing the driver, it is automatically selected to execute when you start-up the Runtime Environment. To verify the if the driver is correctly enabled to start, use the menu option **Project + Status...**, and verify the task Driver Runtime



6 Troubleshooting

After each attempt to communicate using this driver, the tag configured in the field **Read Status** or **Write Status** will receive the error code regarding the kind of failure that occurred. The error messages are:

Error Code	Description (*)	Possible causes	Procedure to solve
0	OK	Communication without problems	
2	Invalid Header	An invalid Header has been typed or the tag that is inside this field has an invalid configuration.	Type a valid Header either on the header field or on the tag value. A lot of different valid headers are shown on the section 4.2
4	Block Size Error	Address offset greater then possible.	Check on the driver configuration's worksheet that is getting this error if the configured offsets is greater then limit operand.. If so, correct it.
5	Error Protocol	It's a protocol error	Check the serial communication configuration. Verify if the settings on the Communication Parameters and on the device are the same.
8	Invalid Command	An invalid Header has been typed or the tag that is inside this field has an invalid configuration.	Type a valid Header either on the header field or on the tag value. A lot of different valid headers are shown on the section 4.2 - Remember, there are read and write commands. Don't try read with a write command or write with a read command.
-15	Timeout waiting start a 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. - Check the right configuration. See on the section 2.2 the different RTS/CTS valid configurations.
-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.

☞ Note: The results of the communication may be verified in the **output** Window of the Studio's environment. To set a log of events for **Field Read Commands**, **Field Write Commands** and **Serial Communication** click with the right button of the mouse on the output window and chose the option setting to select these log events. When testing under a Windows CE target, you can enable the log at the unit (Tools/Logwin) and verify the file celog.txt created at the target unit.

When you are not able to establish the communication with the PLC, you should first of all establish the communication between the PLC Programming Tool and the PLC. Very frequently the communication it is not possible due to a hardware or cable problem, or due an error or lack of configuration at the PLC. Only after the communication between the PLC Programming Software and the PLC is working fine, you can test again the supervisory.

When testing the communication with the Studio, you should first use the application sample described at item 7, instead of the new application that you are creating.

If is required to contact technical support, please have the following information available:

- Operating System (type and version): To find this information use the Tools/System Information option
- Project information: It is displayed using the option Project/Status from the Studio menu
- Driver version and communication log: Available from Studio Output when running the driver
- Device model and boards: please refer to hardware manufacture's documentation

7 Application Sample

The Studio contains a configured project to test the driver. It is strongly recommended to do some tests with this application before beginning the configuration of the customized project, for the follow reasons:

- To understand better the information covered in section 4 of this document.
- To verify that your configuration is working.
- To certify that the hardware used in the test (device + adapter + cable + PC) is in working conditions before beginning the configuration of the applications.

☞ Note: The Application Sample is not available for all drivers.

The Studio application is in the directory: **/COMMUNICATION EXAMPLES/<Driver Name>**

To perform the test, you need to follow these steps:

- Configure the device communication parameters using manufacturer programmer software..
- Open the application **/COMMUNICATION EXAMPLES/<Driver Name>**
- Execute the application
- To display the following screen with some information about the communication, please execute the Viewer module in the Studio.

☞ Note: The application for testing may be used like a maintenance screen for the custom application.

8 History of Versions

Version	By	Date	Description of changes
1.00	Roberto V. Junior	17-jan-2000	▪ Include PSR, RUN and STP command (Release Version).
1.00	Roberto V. Junior	22-dec-1999	▪ First driver version(Beta Version).