

**Communication Driver Virgo**

Communication driver with Virgo PC  
Based Control for Windows CE 3.0

**Index**

<b>1</b>	<b>INTRODUCTION.....</b>	<b>2</b>
<b>2</b>	<b>GENERAL CHARACTERISTICS .....</b>	<b>3</b>
2.1	DEVICE CHARACTERISTICS .....	3
2.2	LINK CHARACTERISTICS.....	3
2.3	DRIVER CHARACTERISTICS .....	3
2.4	INFORMATION ABOUT CONFORMANCE TESTING.....	4
<b>3</b>	<b>INSTALLATION .....</b>	<b>4</b>
3.1	INSTALLING THE DRIVER .....	4
3.2	OTHER SOFTWARE REQUIREMENTS .....	5
<b>4</b>	<b>DRIVER CONFIGURATION .....</b>	<b>5</b>
4.1	SETTINGS - COMMUNICATION PARAMETERS.....	5
4.2	STANDARD DRIVER WORKSHEET .....	6
4.3	STATION AND HEADER CONFIGURATION.....	7
4.4	ADDRESS CONFIGURATION .....	7
4.5	DEVICE CONFIGURATION .....	8
<b>5</b>	<b>EXECUTION .....</b>	<b>8</b>
<b>6</b>	<b>TROUBLESHOOTING.....</b>	<b>9</b>
<b>7</b>	<b>APPLICATION SAMPLE .....</b>	<b>10</b>
<b>8</b>	<b>HISTORY OF VERSIONS.....</b>	<b>11</b>

# 1 Introduction

The Virgo driver enables communication between Studio system and Virgo Run-Time for Windows CE 3.0 using the Altersys API provided with the file "IXLMGR.dll", in accordance with the characteristics covered in this document.

This document contains 8 parts, as follows:

- **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:** Altersys
- **Compatible Equipment:**
  - SH4 and x86 HMI with Windows CE 3.0 and Virgo Run Time installed.

↳ **Tip:** Refers to section 2.4 to see the Equipment used in the standard conformance tests for this driver.

### 2.2 Link Characteristics

- **Device Runtime software:** Virgo Run-Time
- **Specific PC Board:** None

### 2.3 Driver Characteristics

- **Operating System:**
  - Windows CE (x86 / SH3)

↳ **Tip:** 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:

- **VIRGO.INI:** Internal file of the driver, it should not be modified by the user.
- **VIRGO.MSG:** Error messages for each error code. It should not be modified.
- **VIRGO.PDF:** Provides detailed documentation about the driver.
- **VIRGO.DLL:** Compiled 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:** Windows CE HMI with SH4 processor
- **Configuration:**  
Station: 1100
- **Operating System (development):** Windows 2000
- **Operating System (target):** Windows CE v3.00
- **Studio Version:** 4.1
- **Driver version:** 1.00

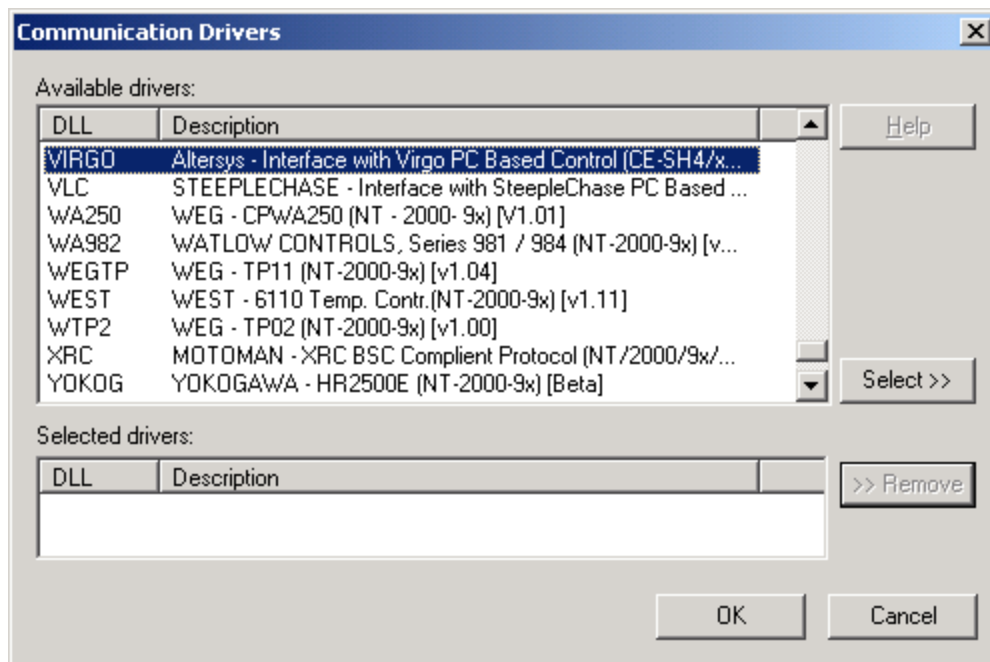
## 3 Installation

### 3.1 Installing the Driver

When you install the Studio, 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 **VIRGO Driver** and push the button **Select >>** (the VIRGO driver must appear in the column **Selected Drivers**).
4. Press **OK**.



### 3.2 Other software requirements

It is necessary to install Virgo Run-Time interface (IXLMGR.DLL). It must be located in the “\Windows” folder or in the same folder where CEView.exe file is located.

## 4 Driver Configuration

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

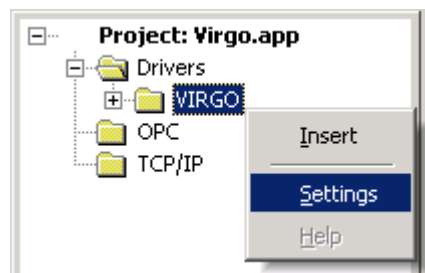
The driver configuration is two parts:

- The Settings or Communication parameters, it is only one configuration to the whole driver;
- The communication tables or Driver Worksheets, where the communication tags are defined. There are two types of communication tables: **Standard Tables** and **MAIN DRIVER SHEET**. This driver version uses just Standard Tables.

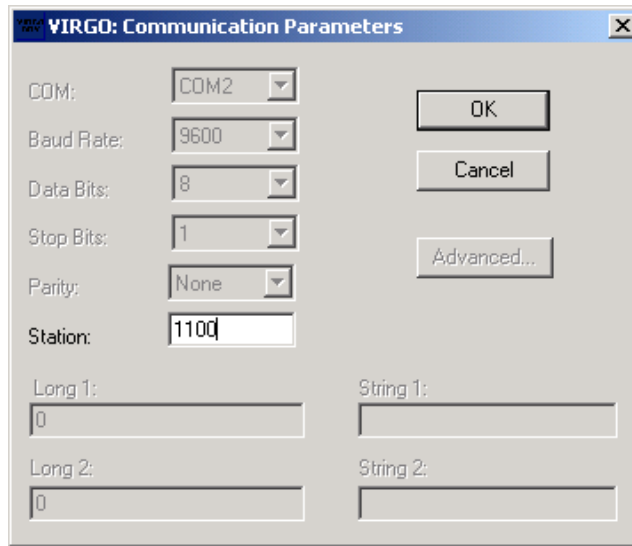
### 4.1 Settings - Communication Parameters

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 **VIRGO**.
3. Right click on the **VIRGO** subfolder and select the option **Settings**.



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

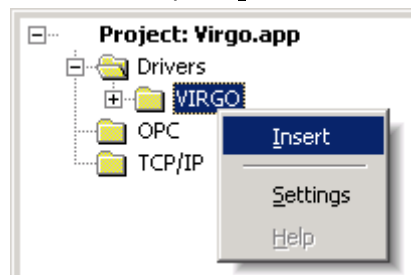


Parameter	Default Value	Valid values	Description
Station	1100	1000 – 2000	Application station number

## 4.2 Standard Driver Worksheet

It is possible to configure many driver worksheets, but this driver works with just one. It will be composed of a Header and Body. To create it, follow these steps:

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



All entries at the Driver Worksheet, exception by the **Station**, **Header** and **Address** are standard to all communication drivers. You should refer to Studio Technical Reference Manual 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	-	1 to 2000	Virgo Run-Time Address
Header	IP Address	-	Virgo Run-Time IP address

After editing the field **Header**, the system will check if it is valid or not. If the syntax were incorrect, the default value (127.0.0.1) will be automatically placed in this field.

### 4.4 Address Configuration

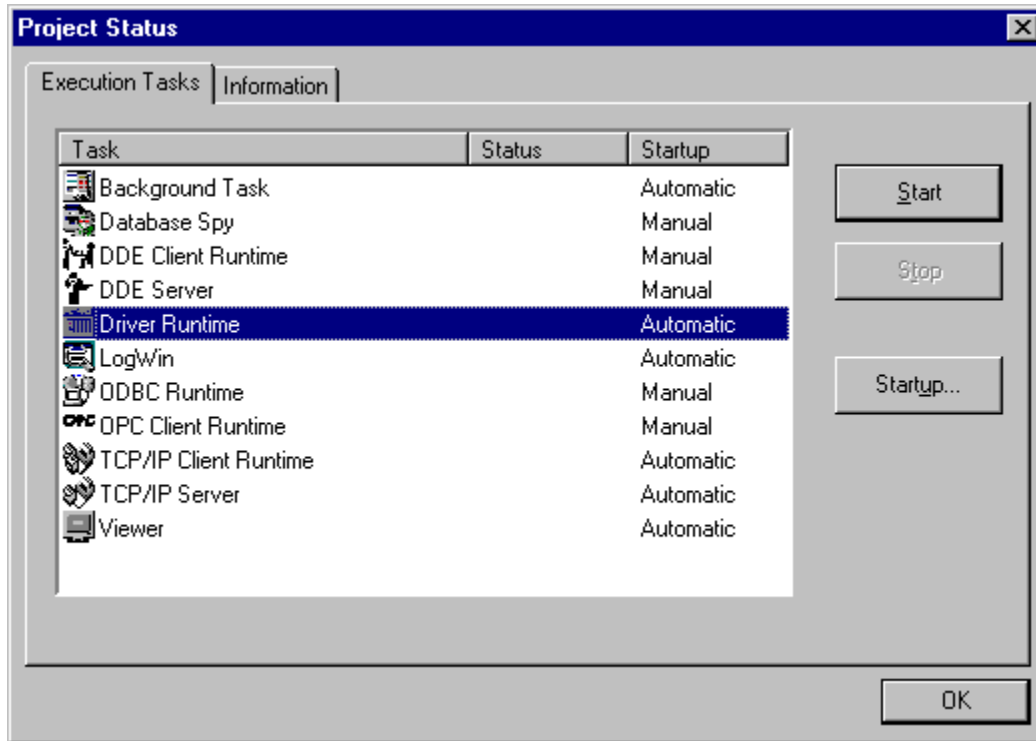
The body of the driver worksheet allows you to associate each tag to its respective address in the device. In the column **Tag Name**, you must type the tag from your application database. This tag will receive or send values from or to an address on the device. The address cells should be filled with Virgo Run-Time tag names configured on its workbench.

## 4.5 Device Configuration

Altersys software should be used to configure the device. Refer to its manual for further details.

## 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	None
1	Invalid Header	Wrong Header typed at the driver configuration worksheet.	- See at the section 4.3 some samples of valid Headers.
2	Invalid Address	The Virgo Run-Time tag typed does not exist.	- Use workbench to check the tag and create it if necessary.

 **Tip:** The communication status can be verified by the **output** Window of the Studio's environment or by the **LogWin** module. 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 testing the communication with the Studio, you should first use the application sample described at item 7 (if it's available), 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

Studio provides 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.

 **Tip:** 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	José L. Teodoro	08-Mar-2001	▪ First driver version
1.01	José L. Teodoro	26-Jul-2001	▪ Fixed problem when getting tag information from Altersys API
1.02	José L. Teodoro	14-Nov-2001	▪ Fixed problem when starting driver
1.03	José L. Teodoro	03-May-2002	▪ Changes to fix the register variables problem