HMF01 Communication Driver

Driver for Serial Communication with Devices
Using V720-series Electromagnetic Inductive RFID System

Contents

Introduction

The HMF01 driver enables communication between the Studio system and V700-series Electromagnetic Inductive RFID System devices using the proprietary protocol communicating over serial, according to the specifications discussed in this document.

This document was designed to help you install, configure, and execute the HMF01 driver to enable communication with these devices. The information in this document is organized as follows:

- **Introduction**: Provides an overview of the HMF01 driver documentation.

- **General Information**: Provides information needed to identify all the required components (hardware and software) used to implement communication between Studio and the HMF01 driver.

- **Installing the Driver**: Explains how to install the HMF01 driver.

- **Configuring the Driver**: Explains how to configure the HMF01 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 HMF01 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*.

General Information

This chapter explains how to identify all the hardware and software components used to implement communication between the Studio HMF01 driver and the V700-series Electromagnetic Inductive RFID System.

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 (RFID)**: OMRON
- **Manufacturer (CFC Reader):** OMROM
- **Compatible Equipment (RFID)**:
  - The V720S-HMF01 RFID
- **Compatible Equipment (CFC Reader)**:
  - CF CARD TYPE RFID UNIT for CE
- **Programmer Software**: None

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**: Serial Port
- **Physical Protocol**: None
- **Logic Protocol**: Low Level
- **Device Runtime Software**: None
- **Specific PC Board**: None

Driver Characteristics

The HMF01 driver is composed of the following files:

- **HMF01.INI**: Internal driver file. *You must not modify this file*.
- **HMF01.MSG**: Internal driver file containing error messages for each error code. *You must not modify this file*.
- **HMF01.PDF**: Document providing detailed information about the **HMF01** driver.
- **HMF01.DLL**: Compiled driver.

✎ **Notes:**
  • All of the preceding files are installed in the `/DRV` subdirectory of the Studio installation directory.
  • You must use Adobe Acrobat® Reader™ (provided on the Studio installation CD-ROM) to view the *HMF01.PDF* document.

You can use the HMF01 driver on the following operating systems:

- Windows CE

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

Conformance Testing

The following hardware/software was used for conformance testing:

- **Equipment**: OMRON (V720S-HMF01 RFID / CF CARD TYPE RFID UNIT for CE)
- **Driver Configuration**:
    - **Baud Rate**: 9600
    - **Parity**: Even
    - **Data Bits**: 8
    - **Stop Bits**: 1
    - **COM Port**: COM 5
- **Operating System (*development*):** Windows XP with Service Pack 1
- **Operating System (*runtime*):** Windows CE
- **Studio Version**: 6.0
- **Driver Version**: 1.0

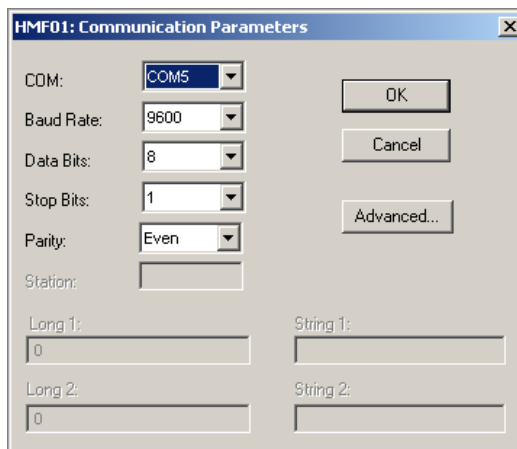## *Setting the Communication Parameters*

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

1. From the Studio development environment, select the *Comm* tab located below the *Workspace*.
2. Click on the *Drivers* folder in the *Workspace* to expand the folder.
3. Right-click on the HMF01 subfolder and when the pop-up menu displays, (as shown in the following figure) select the **Settings** option.
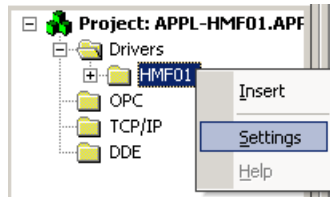
Select Settings from the Pop-Up Menu

The *HMF01: Communications Parameters* dialog displays (as follows).

Communication Parameters Dialog

> ✎ **Note:**
>
> The device must be configured with *exactly the same* parameters that you configured in the *HMF01 Communication Parameters* dialog.

4. Click the **Advanced** button on the *Communication Parameters* dialog to open the *Advanced Settings* dialog and configure the settings that are necessary.

> ✎ **Notes:**
> - Do not change any of the other *Advanced* parameters at this time. You can consult the Studio *Technical Reference Manual* for information about configuring these parameters for future reference.

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 **HMF01** driver from the *Available Drivers* list (as shown in the following figure), and then click the **Select** button.
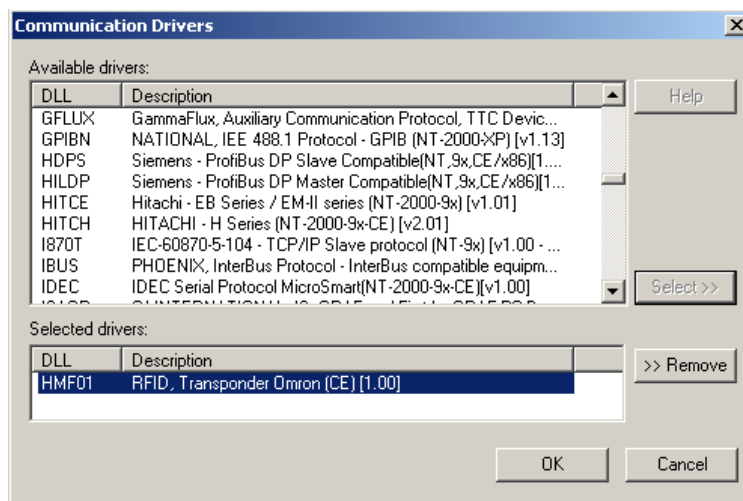
Communication Drivers Dialog Box

5. When the **HMF01** driver displays in the **Selected Drivers** list, click the **OK** button to close the dialog.

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

Configuring the Driver

After opening Studio and selecting the HMF01 driver, you must configure the driver. Configuring the HMF01 driver is done in one part:

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

Setting the Communication Parameters

Click the **Advanced** button on the *Communication Parameters* dialog to open the *Advanced Settings* dialog and configure the settings that are necessary.

> ✎ **Notes:**
> ▪ Do not change any of the other *Advanced* parameters at this time. You can consult the Studio *Technical Reference Manual* for information about configuring these parameters for future reference.
>
> ▪ Generally, you must change the *Advanced* parameter settings if you are using a DCE (Data Communication Equipment) converter (232/485 for example), modem, and so forth between the PC, driver, and the host. You must be familiar with the DCE specifications before adjusting these configuration parameters.

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 *<Driver Name>* subfolder.
3. When the pop-up menu displays (as shown in the following figure), select the **Insert** option.
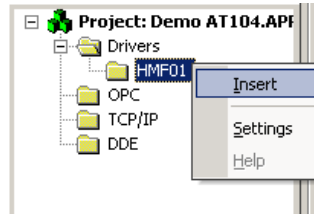
Inserting a New Worksheet

> ✍ **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).

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:

     <ANY>

     Where:

     – This field can be blank

   ▪ **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 C1*:0*)

     These variables must comply with the following syntax:

     <Type>:<AddressReference> (For example: CI*:4*)

Where:

– **Type** is the register type. (**C1,CI**)
– **AddressReference** is the bank of the configured type (CI accepts 0-15 and C1 only accepts 0).

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 HMF01 driver.

Header Field Information			
Data Types	Sample Syntax	Valid Range of Initial Addresses per Worksheet	Comments
C1	C1:0	0-15	For I-CODE1 chip
CI	CI:0	0-15	For I-CODESLI chip

▪ **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:

<Page1>.<Page2>.<Page3>.<Page4> (For example: 0.1, *0.3.5.4*, *5.2.6*, *0*)

Where:

– **Page1** is a parameter that references the page of the bank. (This parameter is obligatory)
– **Page2** like **Page1** but this is not obligatory.
– **Page3** like **Page1** but this is not obligatory.
– **Page4** like **Page1** but this is not obligatory.

| Address Configuration Sample – I-CODESLI		
Device Address	Header Field	Address Field
0-15 –> Bank 0	CI:0	0.1.2.3
16-31 –> Bank 9	CI:9	4.5.6.7
8-15,1-7,24-31 -> Bank 5	CI:5	1.0.3
23-8 -> Bank 15	CI:15	5.4.3.2

Executing the Driver

After adding the HMF01 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.
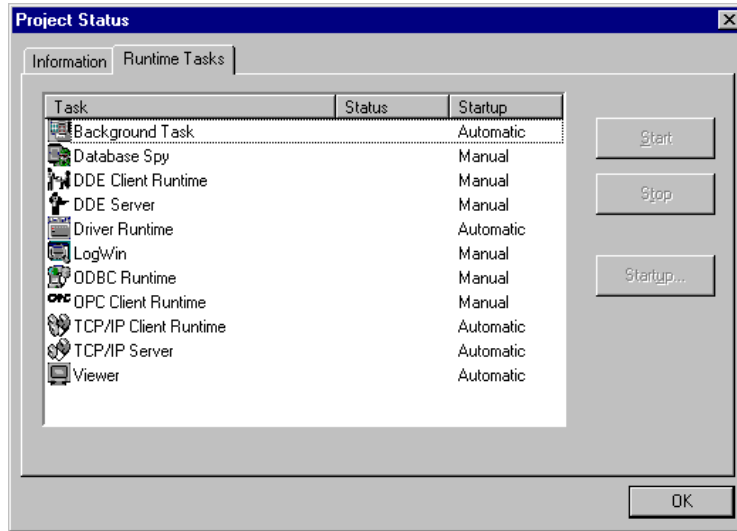
Project Status Dialog Box

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 HMF01 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 what kind of failure occurred.

Error Code	Description	Possible Causes	Procedure to Solve
0	OK	Communication without problems	None required.
1	ERROR_PROTOCOL	Device sent more of 16 bytes.	Try a new reading or a new writing.
3	ERROR_RESPONSE_READ	Device didn't send the expected byte on the reading.	Try a new reading.
4	ERROR_RESPONSE_WRITE	Device didn't send the expected byte on the writing.	Try a new writing.

> **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 testing a Windows CE target, you can use the Remote LogWin of Studio (**Tools → Remote Logwin**) to get the log events from the target unit remotely.

To test communication with Studio, 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**.
- **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 for **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

You will find a sample application for drivers in the **/COMMUNICATION EXAMPLES/<*Driver Name*>** directory. We strongly recommend that you check if there is a sample application for this driver and use it to test the driver before configuring your own customized application, for the following reasons:

- To better understand the information provided in the section of this document.
- To verify that your configuration is working satisfactorily.
- To certify that the hardware used in the test (device, adapter, cable, and PC) is working satisfactorily before you start configuring your own customized applications.

> ✎ **Note:**
> This application sample is not available for all drivers.

Use the following procedure to perform the test:

1. Configure the device's communication parameters using the manufacturer's documentation.
2. Open and execute the sample application.

> ⇨ **Tip:**
> You can use the sample application screen as the maintenance screen for your custom applications.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of changes
0	1.0	Leandro Coeli	Oct/01/2004	Initial revision