

**OMRON Communication Driver**

Driver for Communication with OMRON  
Devices Using the FinsGateway and/or SYSMAC Gateway

**Contents**

**INTRODUCTION .....2**

**GENERAL INFORMATION .....3**

    DEVICE CHARACTERISTICS .....3

    LINK CHARACTERISTICS .....3

    DRIVER CHARACTERISTICS .....4

    CONFORMANCE TESTING .....5

**INSTALLING THE DRIVER .....6**

**CONFIGURING THE DRIVER.....7**

    SETTING THE COMMUNICATION PARAMETERS .....7

    CONFIGURING THE DRIVER WORKSHEETS .....9

    DEVICE CONFIGURATION ..... 19

**FINSGATEWAY ..... 20**

    FINSGATEWAY CONFIGURATION..... 20

    COMMUNICATION TEST..... 21

    MEMORY MONITOR ..... 21

    FINSGATEWAY HELP ..... 22

**SYSMAC GATEWAY ..... 23**

    SYSMAC GATEWAY SETUP: ..... 24

    EXECUTING THE DRIVER ..... 25

**TROUBLESHOOTING..... 26**

**REVISION HISTORY ..... 28**

## Introduction

The OMRON driver enables communication between the Studio system and devices using the OMRON FinsGateway or the SYSMAC Gateway, according to the specifications discussed in this document.



### Attention:

This driver is available only for customers who acquired the software from OMRON. If you purchased the software from any supplier other than OMRON, the driver will not be executed during the runtime. In this case, consult your supplier for information about other communication drivers that support the OMRON devices.

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

- **Introduction:** Provides an overview of the OMRON driver documentation.
- **General Information:** Provides information needed to identify all the required components (hardware and software) used to implement communication between Studio and the OMRON driver.
- **Installing the Driver:** Explains how to install the OMRON driver.
- **Configuring the Driver:** Explains how to configure the OMRON driver.
- **FinsGateway:** Explains the FinsGateway configuration.
- **SYSMAC Gateway:** Explains the SYSMAC Gateway configuration.
- **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 OMRON 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 Environment. If you are unfamiliar with Windows, 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 OMRON driver and the OMRON 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:** OMRON
- **Compatible Equipment:** Any OMRON PLC that supports communications with the FINS or SYSMAC Gateways
- **Programmer Software:** CX-Programmer

For a list of the devices used for conformance testing, see “Conformance Testing” on page 5.

### Link Characteristics

In order to communicate using the **FINS Gateway**, you can use any **Physical Protocol** supported by the **FINS Gateway**, such as Ethernet, Serial, CLINK, CS1, directly through the Ethernet or Serial ports, or any of the available networks. The **OMRON** driver communicates with the **FINS Gateway** and the **FINS Gateway** is the one that goes and establishes the link with the PLCs using whatever network is configured.

On the same token, In order to communicate using the **SYSMAC Gateway**, you can use any **Physical Protocol** supported by the **SYSMAC Gateway**, such as Ethernet/IP, Serial, USB, and so forth. The **OMRON** driver communicates with the **SYSMAC Gateway** and the **SYSMAC Gateway** is the one that goes and establishes the link with the PLCs using whatever network is configured.

## **Driver Characteristics**

The OMRON driver is composed of the following files:

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

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

- Windows XP
- Windows 7/8

For a list of the operating systems used for conformance testing, see “Conformance Testing” on page 5.

When using the FINS Gateway, the OMRON driver supports the following registers:

Memory Area	Length	Write	Read
A (Auxiliary Bit Area)	2 Bytes	•	•
CIO (Core IO Area)	2 Bytes	•	•
CF (Counter Area – Completion Flag)	1 Byte	–	•
C (Counter Area – PV)	2 Bytes	•	•
D (Data Memory Area)	2 Bytes	•	•
En (Extension Memory – Bank)	2 Bytes	•	•
E (Extension Memory – Current Bank)	2 Bytes	•	•
H (Holding Bit Area)	2 Bytes	•	•
TK (Task Flag)	1 Byte	–	•
TF (Timer Area – Completion Flag)	1 Byte	–	•
T (Timer Area – PV)	2 Bytes	•	•
W (Work Area)	2 Bytes	•	•

When using the SYSMAC Gateway, it communicates through Tag Names:

Tag Communication	Description
TAG	Tag Name on the PLC See Section Sysmac Gateway

### **Conformance Testing**

The following hardware/software was used for conformance testing:

- **Cable:** Ethernet cable

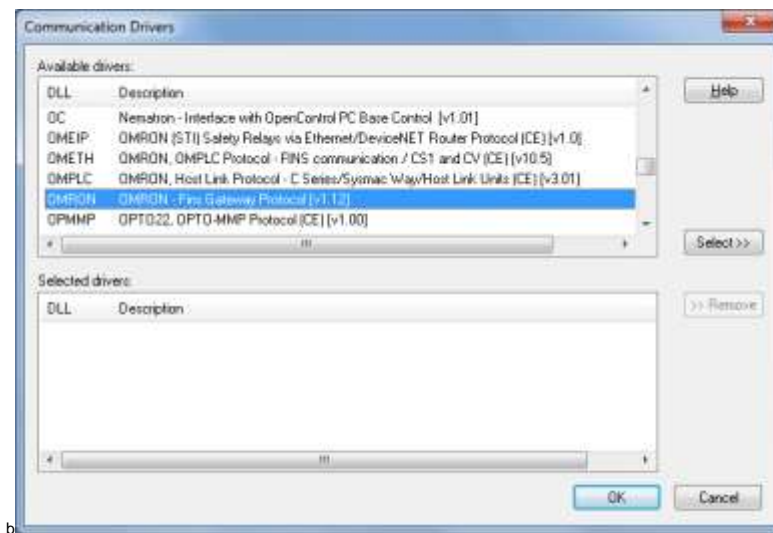
Driver Version	Studio Version	Operating System - Development	Operating System - Runtime	Equipment
1.21	8.1 + SP2	Windows 7/8	Windows XP	CJ1M CPU12 CJ2F-CPU64-EIP CS1 CPU-45H NJ501-1300

## Installing the Driver

When you install Studio version 6.1 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 **OMRON** 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 **OMRON** driver displays in the **Selected Drivers** list, click the **OK** button to close the dialog.



**Note:**

It is necessary to install the Fins or SYSMAC Gateway software on your computer to enable communication between Studio and the PLC. However, to download the program to your PLC, you must install the CX-Programmer software. Consult your CX-Programmer software 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 in this area.

## Configuring the Driver

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

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

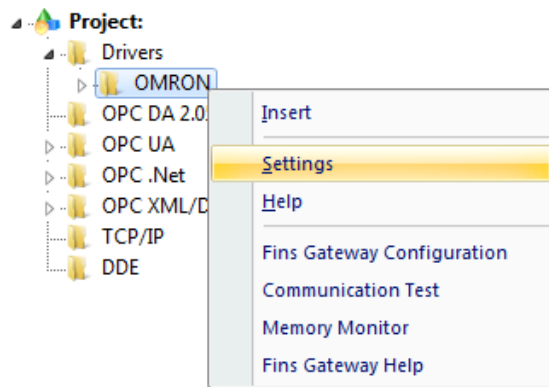
Worksheets are divided into two sections, a *Header* and a *Body*. The fields contained in these two sections are standard for all communications drivers — except the **Station**, **Header**, and **Address** fields, which are driver-specific. This document explains how to configure the **Station**, **Header**, and **Address** fields only.

**Note:**  
For a detailed description of the Studio *MAIN* and *STANDARD DRIVER SHEETS*, and information about configuring the standard fields, review the product's *Technical Reference Manual*.

### Setting the Communication Parameters

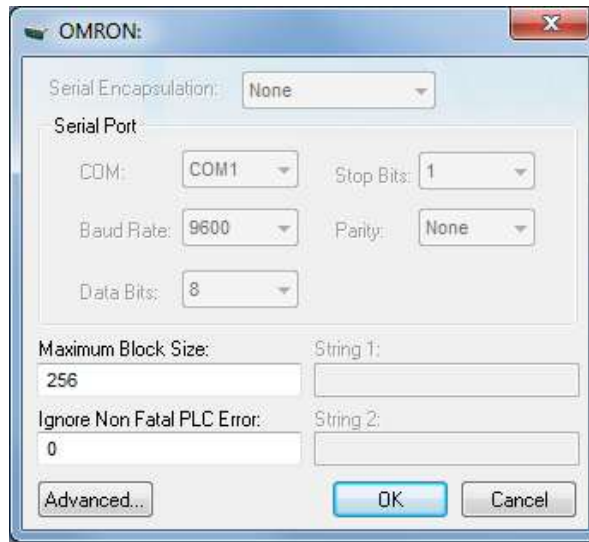
Use the following steps to configure the communication parameters, which are valid for all driver worksheets configured in the system:

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 OMRON 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 *OMRON: Communications Parameters* dialog displays (as follows).



**Communication Parameters Dialog**

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

Parameters	Default Values	Valid Values	Description
<b>Maximum Block Size</b>	256	1...999	Maximum block size (Words)
<b>Ignore Non Fatal PLC Error</b>	0	0 or 1	If this parameter is set to 1, the OMRON driver ignores the Non Fatal PLC Error, and traces a warning at LogWin (Output Window). This parameter affects only the communication through the FINS Gateway

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

**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.

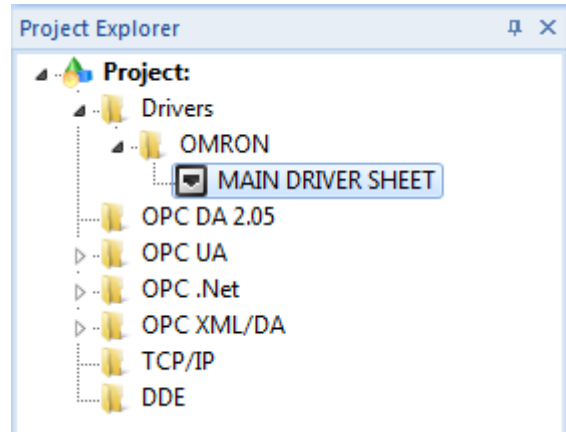


## Configuring the Driver Worksheets

This section explains how to configure the *MAIN* and *STANDARD DRIVER SHEET*s (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 a *Body* section.

### Configuring the MAIN DRIVER SHEET

When you add the OMRON 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.



**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 field:**

**For FINS Gateway:**

The Station field must comply with the following syntax:

<NetID>.<NodeID>.<Unit#>:<optMode>

Where:

- <NetID> = Network ID number (0...255)
- <NodeID> = Node ID number (0...255)
- <Unit#> = Unit number (0...255)
- <optMode> = The user should be able to specify the mode (:CS, :CJ or :CV). If the user does not specify the mode, the driver should execute the command *Controller Data Read (0501)* to check the PLC Model and execute the correct mode, according to the PLC. If the driver cannot identify the PLC model by the response (e.g. another computer), the CV mode is used by default.

**For SYSMAC Gateway:**

The Station field must comply with the following syntax:

<IP>:<Port ID>:<optMode>

Where:

- <IP> = The target IP Address
- <Port ID> = The Port ID (see the SYSMAC Gateway reference for more information)
- <optMode> = The user should be able to specify the mode (:CJ2, :NE1S or :NJ). If the user does not specify the mode, the driver will use the CJ2 mode

- I/O Address field: The Address field must support the following syntax:

For FINS Gateway:

- To Clock command:

Syntax: CLK<Number>

Number	Description
0	Year
1	Month
2	Day
3	Hour
4	Minute
5	Second
6	Day of week

- To Status command:

Syntax: STAT<Number>

Number	Description
0	<b>Status</b> The operating status of the CPU Unit is as follows: <b>00h:</b> Stop (user program not being executed) <b>01h:</b> Run (user program being executed) <b>80h:</b> CPU on standby (waiting for a signal from a device such as a SYSMAC BUS Remote I/O Slave Unit).
1	<b>Mode</b> The CPU Unit operating mode is as follows: <b>00h:</b> PROGRAM <b>02h:</b> MONITOR <b>04h:</b> RUN

- To CPU Unit command:

**Description:** Reads the following data:

- CPU Unit model
- CPU Bus Unit configuration
- CPU Unit version
- Remote I/O data
- Area data
- CPU Unit information

Syntax: CPU0

- To CMD command:

**Description:** Command (support any data type, by typing its memory area code – Hex). When reading, the driver sends the <Command> configured in the **Address** and stores the value (answer) in the tag configured in the **Tag Name** column. When writing, the driver concatenates the <Command> configured in the **Address** with the value of the tag configured in the **Tag Name** column and sends this command to the PLC.

Syntax: CMD:<Command>

▪ **To Memory Area command:**

**Syntax:** <MemoryArea><Address>.<optBitNumber>:<optDataType>

Where:

- <MemoryArea>: Specify the type of register, according to the following list:

Driver Syntax	Description	CX Programmer Syntax
A	Auxiliary Bit Area	A, AC, AR
CF	Counter Area (Completion Flag)	-
C	Counter Area (PV)	C
CIO	Core IO Area	CIO, <None>
D	Data Memory Area	D, DM
E	Extension Memory – current bank	E, EM
E<n>_	Extension Memory – Bank <n> (<n> = 0...12)	En
H	Holding Bit Area	H, HR
T	Task Flag	T, TK
TF	Timer Area (Completion Flag)	-
T	Timer Area (PV)	T, TC
TK	Task Flag	TK
W	Work Area	W, WR

- <Address>: Offset for the Memory Area

- <optBitNumber>: Bit number from 0 to 15

- <optDataType>: Specifies the format of the values read/written. When this parameter is omitted, the default value for each *Memory Area* type is used.

Syntax supported by the OMRON driver	Bytes
CHANNEL	2
DINT	4
INT	2
LINT	8
NUMBER	8
REAL	4
LREAL	8
UDINT	4
UDINT_BCD	4
UINT	2
UINT_BCD	2
ULINT	8
ULINT_BCD	8
ASCII<NumberOfChars>	<N>

**For SYSMAC Gateway:**

▪ **To TAG command:**

**Syntax:** TAG<PLC Tag Name>

PLC Tag Name	I/O Address
Temperature	TAG:Temperature
Pressure[2].CV	TAG:Pressure[2].CV

The current supported data types when communicating using **SYSMAC Gateway**, subject to the PLC supporting these types as well, are:

- Boolean: **BOOL**
- Bit strings: **BYTE, WORD, DWORD, and LWORD**
- Integers:
  - Signed: **SINT, INT, DINT, and LINT**
  - Unsigned: **USINT, UINT, UDINT, and ULINT**
- Real numbers **REAL** and **LREAL**
- Durations: **TIME**
- Dates: **DATE**
- Times: **TIME\_OF\_DAY**
- Date and times: **DATE\_AND\_TIME**
- Text strings: **STRING[n]**
- ARRAYS
- STRUCTURES
- POU Variables



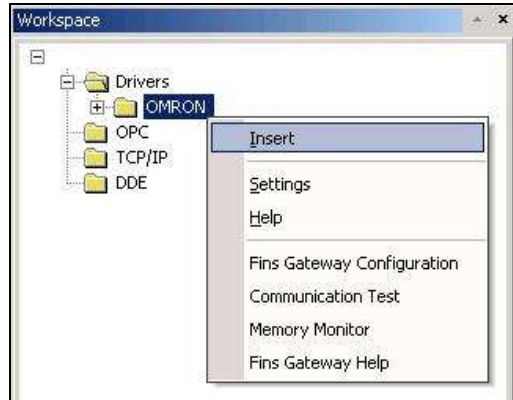
**Note:**

When using TAG of the following OMRON datatypes DATE , DATE\_TIME, TIME and TIME\_OF\_DAY that are related to time and date please use the following format, for DATE\_TIME mm/dd/yyyy hh:mm:ss.mmm, for DATE mm/dd/yyyy and for TIME can be represented as [-]d.hh:mm:ss.ff, where the optional minus sign indicates a negative time interval, the *d* component is days, *hh* is hours as measured on a 24-hour clock, *mm* is minutes, *ss* is seconds, and *ff* is fractions of a second.

### 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 *OMRON* 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[1]	1:INT		
2	Tag[2]	2:INT		
3	Tag[3]	3:INT		
4	Tag[4]	4:INT		
5				

**STANDARD DRIVER SHEET**

In general, all parameters on the Driver Worksheet (except the **Station**, **Header** and **Address** fields) are exploding 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*.

- Use the following information to complete the **Station**, **Header** and **Address** fields on this worksheet.

**For FINS Gateway:**

- Station field:** The Station field must support the following syntax:

`<NetID>.<NodeID>.<Unit#>:<optMode>`

Where:

- `<NetID>` = Network ID number (0...255)
- `<NodeID>` = Node ID number (0...255)
- `<Unit#>` = Unit number (0...255)
- `<optMode>` = The user should be able to specify the mode (:CS, :CJ :CV). If the user does not specify the mode, the driver should execute the command *Controller Data Read (0501)* to check the PLC Model and execute the correct mode, according to the PLC. If the driver cannot identify the PLC model by the response (e.g. another computer), the CV mode is used by default.

**For SYSMAC Gateway:**

If using TAG communication type using the following syntax:

`<IP>:<Port ID>:<optMode>`

Where:

- `<IP>` = The target IP Address
- `<Port ID>` = The Port ID (see the SYSMAC Gateway reference for more information)
- `<optMode>` = The user should be able to specify the mode (:CJ2, :NE1S or :NJ). If the user does not specify the mode, the driver will use the CJ2 mode, which is also valid if you want to communicate 2 computers using the SYSMAC Gateway

**Note:**  
 For communicating with NJ the version of Compolet must be CX-Compolet with SGW V1.31 or higher

**Header field:**

**For FINS Gateway:**

Use the information in the following table to define the type of memory area that will be read from or written to the device (default value is A).

These variables must comply with the following syntax:

<MemoryArea>

Where:

- <MemoryArea>: Specify the type of register, according to the following list:

Driver Syntax	Description	CX Programmer Syntax
A	Auxiliary Bit Area	A, AC, AR
CF	Counter Area (Completion Flag)	-
C	Counter Area (PV)	C
CIO	Core IO Area	CIO, <None>
D	Data Memory Area	D, DM
E	Extension Memory – current bank	E, EM
E<n>_	Extension Memory – Bank <n> (<n> = 0...12)	En
H	Holding Bit Area	H, HR
T	Task Flag	T, TK
TF	Timer Area (Completion Flag)	-
T	Timer Area (PV)	T, TC
TK	Task Flag	TK
W	Work Area	W, WR
CLK	Clock (year, month, date, minute, second, and day of the week)	-
CPU	CPU Model <ul style="list-style-type: none"> <li>▪ CPU Unit model</li> <li>▪ CPU Bus Unit configuration</li> <li>▪ CPU Unit version</li> <li>▪ Remote I/O data</li> <li>▪ Area data</li> <li>▪ CPU Unit information</li> </ul>	-
STAT	CPU Status <ul style="list-style-type: none"> <li>▪ Status</li> <li>▪ Model</li> </ul>	-
CMD	Command (support any data type, by typing its memory area code – Hex)	-

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.

**For SYSMAC Gateway:**

Driver Syntax	Description	CX Programmer Syntax
TAG	Communicates with PLC Tag Names	<PLC Tag Name>



**Address field:**

**For FINS Gateway:**

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:

- **To CLK Header:**

**Syntax:** <Number>

Number	Description
0	Year
1	Month
2	Day
3	Hour
4	Minute
5	Second
6	Day of week

- **To STAT Header:**

**Syntax:** <Number>

Number	Description
0	<b>Status</b> The operating status of the CPU Unit is as follows: <b>00h:</b> Stop (user program not being executed) <b>01h:</b> Run (user program being executed) <b>80h:</b> CPU on standby (waiting for a signal from a device such as a SYSMAC BUS Remote I/O Slave Unit).
1	<b>Mode</b> The CPU Unit operating mode is as follows: <b>00h:</b> PROGRAM <b>02h:</b> MONITOR <b>04h:</b> RUN

- **To CPU Header:**

**Syntax:** 0

- **To Memory Area command:**

<Address>.<optBitNumber>:<optDataType>

Where:

- <Address>: Offset for the *Memory Area*

- <optBitNumber>: Bit number from 0 to 15

- <optDataType>: Specifies the format of the values read/written. When this parameter is omitted, the default value for each *Memory Area* type is used.

Syntax supported by the OMRON driver	Bytes
CHANNEL	2
DINT	4
INT	2
LINT	8
NUMBER	8
REAL	4
LREAL	8
UDINT	4
UDINT_BCD	4
UINT	2
UINT_BCD	2
ULINT	8
ULINT_BCD	8
ASCII<NumberOfChars>	<N>

➤ **Attention:**  
 You must not configure a range of addresses greater than the maximum block size (data buffer length) supported by each PLC within the same worksheet. The maximum data buffer length for this driver is 998 words per *STANDARD DRIVER SHEET*.

➤ **Attention:**  
 In ASCII Type, you should read and write in the Number of Chars signed on the Address field.

➤ **Attention:**  
 The user should pay attention when using data types different from the memory area defaults. In these cases, the user should take special care with the correct PLC memory mapping to avoid undesirable results (the memory area offset value ALWAYS considers the default data type for that area) .

**For SYSMAC Gateway:**

**Syntax:** <PLC Tag Name>

**Write Group using TAGWBINT and TAGWBREAL**

For performance enhancements, the OMRON driver supports two headers for writing blocks when using Sysmac Gateway, TAGWBINT for writing INT arrays and TAGWBREAL for writing REAL arrays. This feature only works on Standard Driver Sheet. You should always specify the entire array. You will not be able to read using this header, it is used for write only.

Example of using TAGWBINT:

On the PLC create a INT array like intArrayTest[0..4]

Header: TAGWBINT

I/O Address
intArrayTest[0]
intArrayTest[1]
intArrayTest[2]
intArrayTest[3]
intArrayTest[4]

Example of using TAGWBREAL:

On the PLC create a REAL array like realArrayTest[0..4]

Header: TAGWBREAL

I/O Address
realArrayTest[0]
realArrayTest[1]
realArrayTest[2]
realArrayTest[3]
realArrayTest[4]

### **Device Configuration**

Because there are multiple devices that use the Fins and the SYSMAC Gateway, we cannot define a standard device configuration.

## FinsGateway

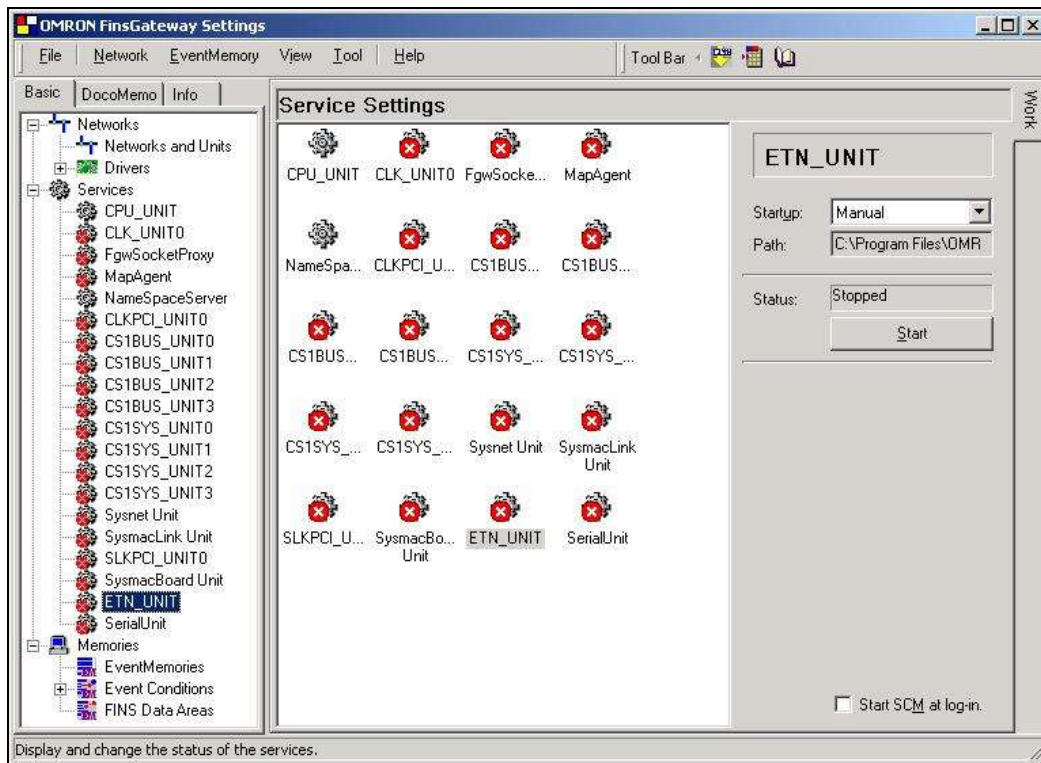
The OMRON driver interfaces with the FINS Gateway Library in order to exchange data with the devices. When you right-click on the OMRON driver from the *Workspace*, shortcuts to the following programs display:

- Fins Gateway Configuration
- Communication Test
- Memory Monitor
- Fins Gateway Help

More information about each of these modules is available in the FinsGateway Help file.

### FinsGateway Configuration

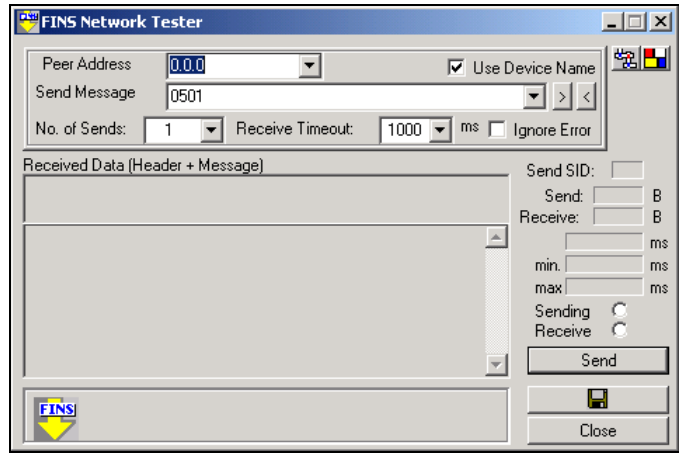
The FinsGateway configuration program performs the basic settings required to use FinsGateway and allows the user to configure the routing tables.



**FinsGateway Configuration**

### **Communication Test**

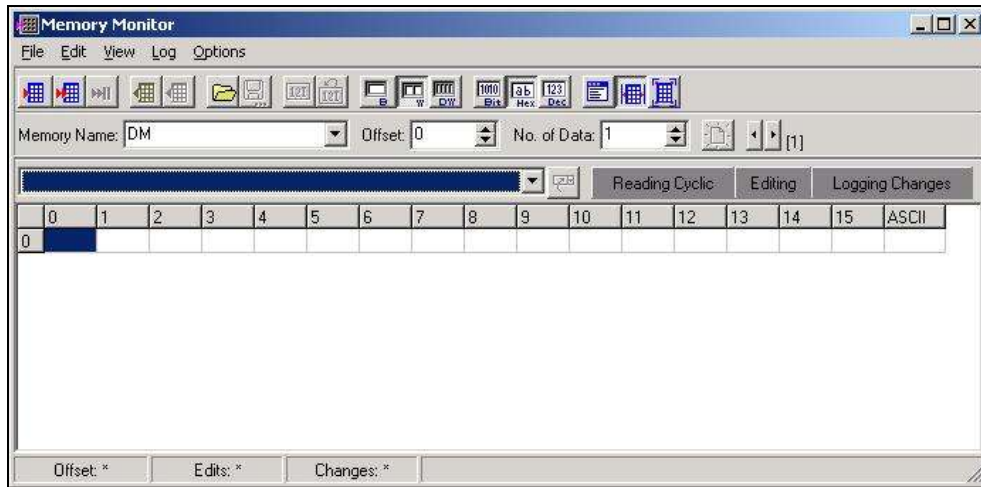
This is a debugging tool useful for testing the connection with the remote devices and to execute FINS commands.



**Communication Test**

### **Memory Monitor**

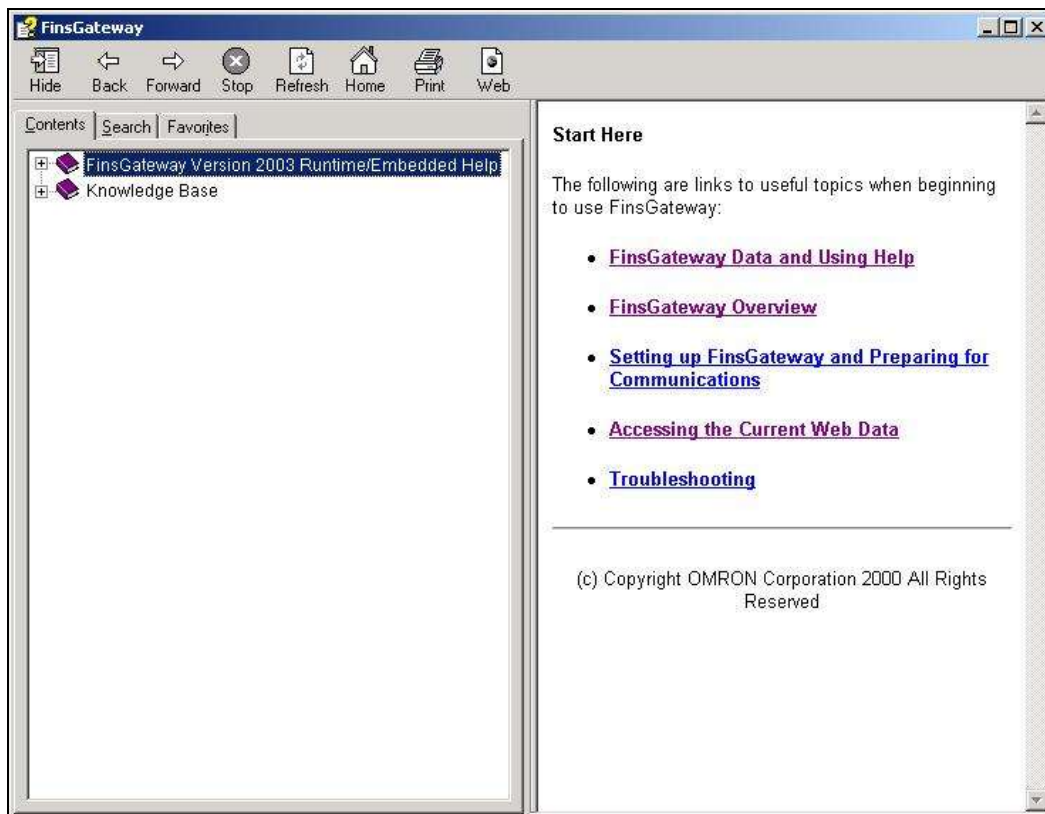
This is a debugging tool useful for monitoring the value of the Memory Area registers.



**Memory Monitor**

## **FinsGateway Help**

This is the Help system for the FinsGateway modules.



## Sysmac Gateway

The OMRON driver interfaces with the SYSMAC GATEWAY in order to exchange data tags with the devices.

The OMRON driver has the following capabilities when interfacing with SYSMAC GATEWAY:

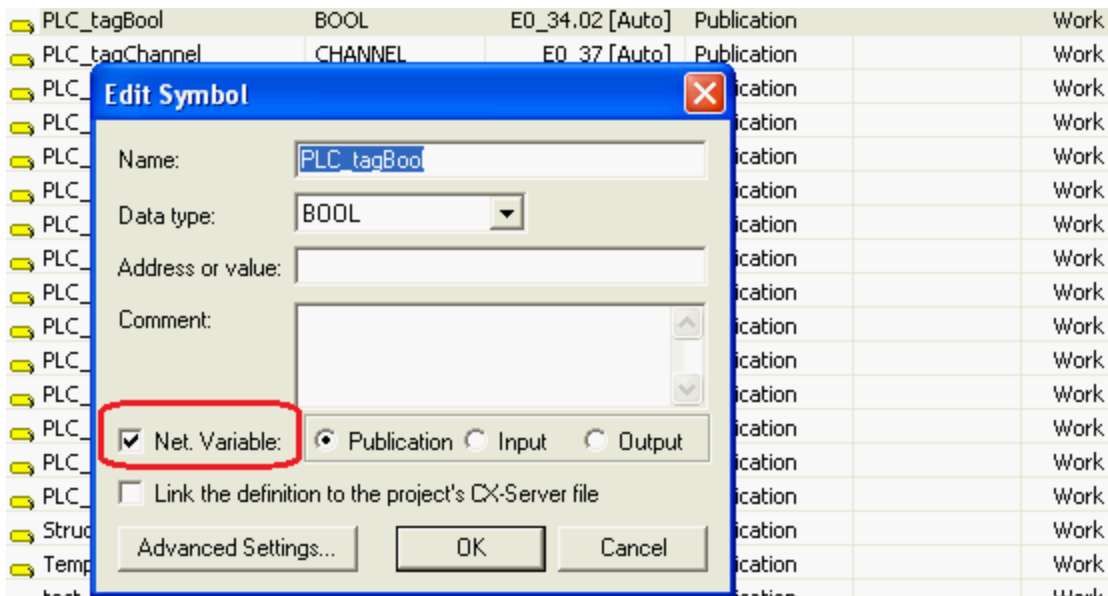
- Read Tag Variables by Name (Synchronous operation)
- Write Tag Variables by Name (Synchronous operation)

When configuring tags using **SYSMAC STUDIO** to communicate through **SYSMAC GATEWAY**, you have to be sure to set the proper **Network Publish Settings**. **Publish Only** is the most recommended Setting.

Name	Data Type	Initial Value	AT	Retain	Constant	Network Publish	Comment
IReal1	LREAL			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
IReal2	LREAL			<input type="checkbox"/>	<input type="checkbox"/>	Do not publish	
IReal3	LREAL			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
IReal4	LREAL			<input type="checkbox"/>	<input type="checkbox"/>	Input Output	

*Configuring Tags on SYSMAC Studio*

Also, when using **CX-Programmer** to configure a **CJ2**, as you configure the Symbols that you want to communicate with using the **OMRON** driver, please be sure to configure the proper **Usage**, to **Publication**. You can do that by setting the **Symbol->Advanced** parameter **Net. Variable->Publication**

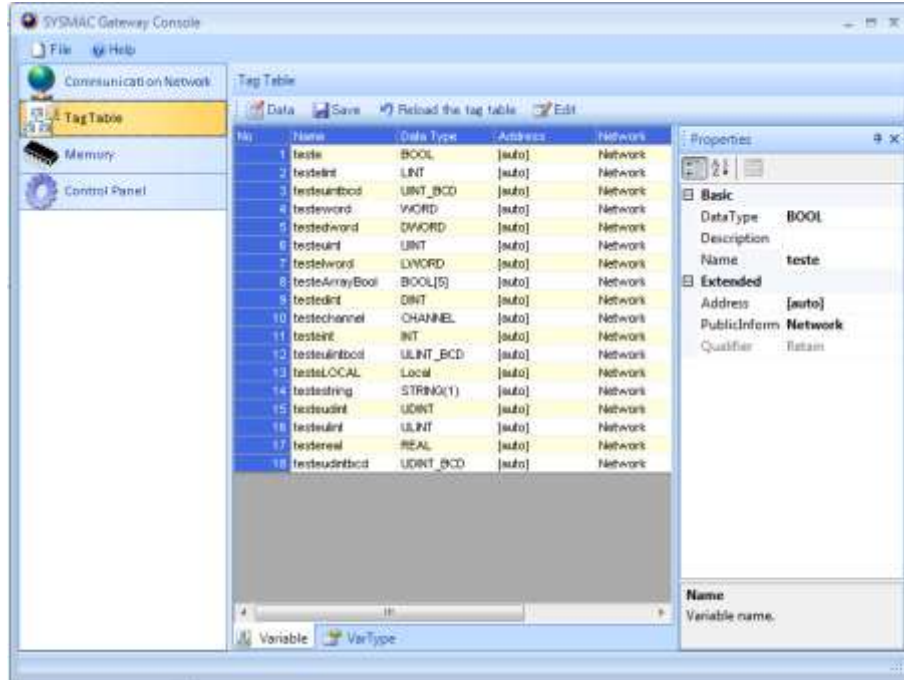


*Configuring Symbols on CX-Programmer*

**Sysmac Gateway Setup:**

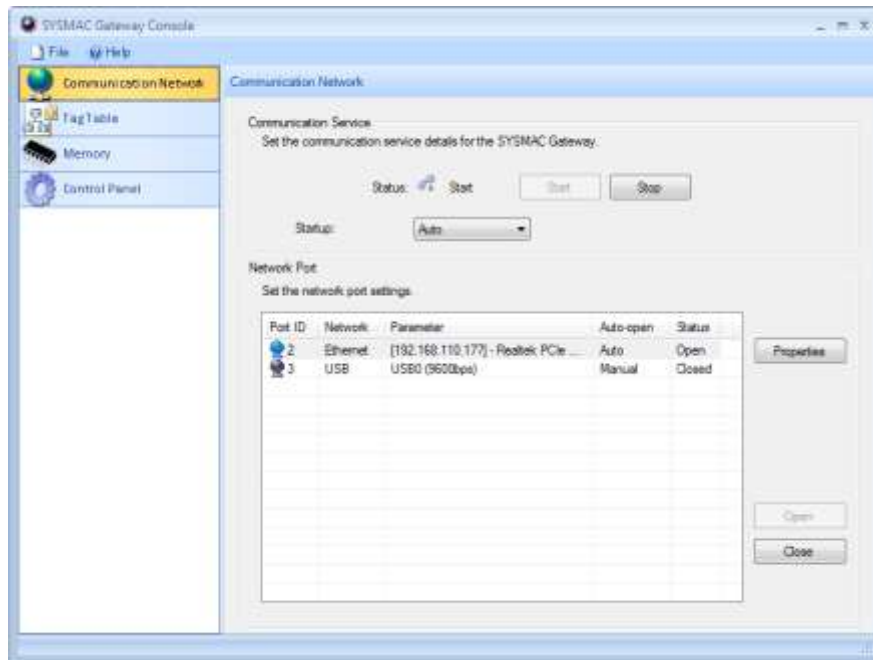
Make sure that SYSMAC GATEWAY is started.

If you are reading and writing data to and from the Event Memory within the Sysmac gateway, please add to the tag table the Name, Data Type and Address.



**Adding Tags to the Sysmac Gateway Console**

Start the Communication Service:



**Starting the SYSMAC Gateway Communication Service**

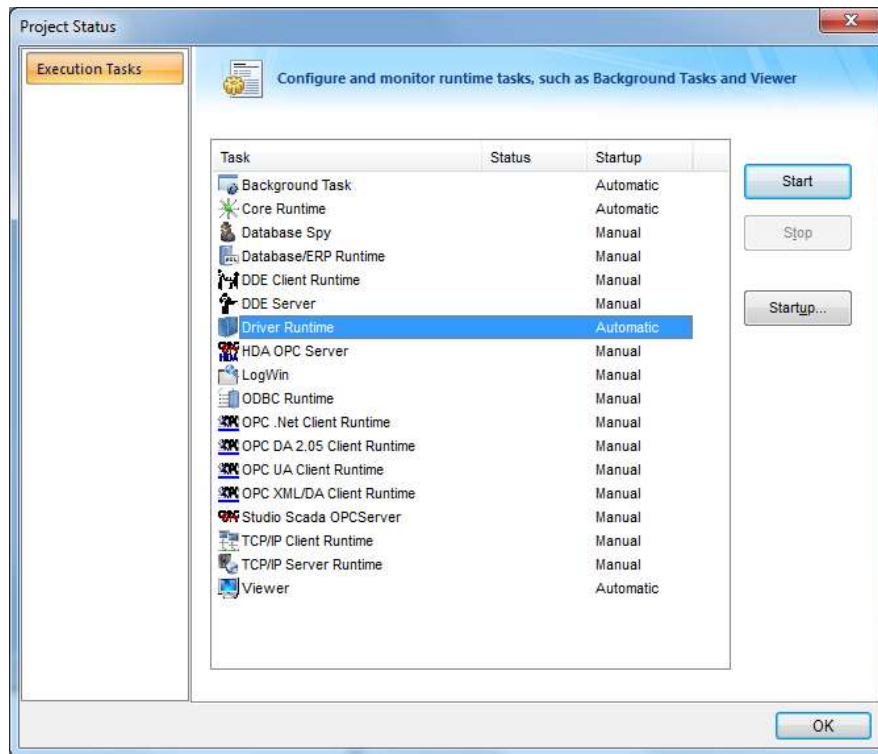


## Executing the Driver

After adding the OMRON 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** → **Tasks** 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 OMRON 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	Head Compose Error	Error in the communication	Verify that the FinsGateway Service Manager is configured correctly.
2	Send Data Error	Error sending data	Verify that the FinsGateway Service Manager is configured correctly.
3	Receive Data Error	Error receiving data	Verify that the FinsGateway Service Manager is configured correctly.
4	Invalid Header	Invalid Memory Area configured in the Header field	Check the memory area configured in the Header field.
5	Invalid Data Type	Invalid Data Type configured in the Address field	Check the data type configured in the Address field.
6	Request Version Error	Error in the request version	Verify that the FinsGateway Service Manager is configured correctly.
8	Invalid Command	The command specificity cannot read or write	Verify that the command is valid to read or write.
-15	Timeout Start Message	<ul style="list-style-type: none"> <li>•Disconnected cables</li> <li>•PLC is turned off, in stop mode, or in error mode</li> <li>•Wrong station configuration</li> </ul>	<ul style="list-style-type: none"> <li>• Check cable wiring.</li> <li>• Check the PLC state – it must be RUN or Monitor.</li> <li>• Check the station fields.</li> <li>• .Bad Routing Table configuration</li> </ul>
101	Unable to connect	<ul style="list-style-type: none"> <li>•The SYSMAC GATEWAY is down</li> </ul>	<ul style="list-style-type: none"> <li>• Check if the SYSMAC GATEWAY is running</li> </ul>
102	Error Reading Data	<ul style="list-style-type: none"> <li>•The tag that is configured does not exist on the PLC</li> </ul>	<ul style="list-style-type: none"> <li>• Use the Tag Browser to avoid typing wrong tag names</li> </ul>
103	Error Writing Data	<ul style="list-style-type: none"> <li>•Wrong tag typed</li> <li>•Invalid Value</li> </ul>	<ul style="list-style-type: none"> <li>• Use the Tag Browser to avoid typing wrong tag names</li> </ul>
104	Invalid Array Format	<ul style="list-style-type: none"> <li>•The format is invalid</li> </ul>	<ul style="list-style-type: none"> <li>• Example for testArray[10] :</li> <li>• Wrong format: testArray1 testArray testArray testArray[20]</li> <li>• Correct: testArray[4]</li> </ul>
105	Error Loading COMPOLET API Libraries (FEATURE DISABLED)	<ul style="list-style-type: none"> <li>•The SYSMAC GATEWAY is not installed</li> <li>•The library used to communicate through Sysmac Gateway is not compatible with this driver version</li> </ul>	<ul style="list-style-type: none"> <li>• Install the SYSMAC GATEWAY and run the driver again</li> <li>• Update the Driver using its SETUP file (OMRON.exe) which will also update the file SysmacLoader.dll, which is installed on the Studio's DRV\API folder</li> </ul>
106	Timeout	<ul style="list-style-type: none"> <li>•Disconnected cables</li> <li>•PLC is turned off, in stop mode, or in error mode</li> <li>•Wrong station number</li> </ul>	<ul style="list-style-type: none"> <li>• Check cable wiring.</li> <li>• Check the PLC state – it must be RUN or Monitor.</li> <li>• Check the station format.</li> </ul>

→ **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 **Protocol Analyzer**, 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 remote target, you can use the Remote LogWin of Studio (**Remote Logwin**) to get the log events from the target unit remotely.

If you are unable to establish communication with the PLC, try to establish communication between the FINS or SYSMAC Gateway Testing tools and the PLC. If you can get their testing tools to work, the **OMRON** driver will work as well

If you must contact us for technical support, please have the following information available:

- **Operating System** (type and version): To find this information, select **Help->Support 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 **Protocol Analyzer** for the Output window.
- **Device Model and Boards**: Consult the hardware manufacturer's documentation for this information.

## Revision History

Doc. Revision	Driver Version	Author	Date	Description of changes
A	1.00	Eric Vigiani	Set/24/2004	Initial version
B	1.01	Leandro G. Coeli	Apr/25/2005	Fixed problems on MDS
C	1.02	Eric Vigiani	Jul/15/2005	Modified the BOOL data type to accept Bit
D	1.03.2-Beta	Adolfo Aguiar	Dec/05/2006	Fixed problems with the En_m syntax
E	1.03.3-Beta	Plínio M. Santana	Dec/28/2006	Fixed problems with ASCII Format.
F	1.04	Plínio M. Santana	May/09/2007	Document Revision.
G	1.06	Rafael R. Fernandes	Aug/21/2007	Fixed problem with address field length
H	1.07	Eric Vigiani / Rafael R. Fernandes	Sep/05/2007	Included a counter ID responsible by message identification. Implemented an option to ignore the Non Fatal PLC Error. Fixed bug with "CMD" operand.
I	1.09	Lourenco Teodoro	Nov/18/2009	Added support to multiple connections with the limitation of just 1 per station
J	1.10	André Körbes	Sep/23/2010	Implemented buffer overrun protections
K	1.11	André Körbes	Aug/18/2011	Minor fixes
L	1.12	Paulo Balbino	Jun/12/2012	Added CX-Compolet API (SYSMAC GATEWAY) Added Support for NJ and CJ CPUs Added Time and Date data types
M	1.13	Paulo Balbino	Jul/16/2013	Fixed problems with simultaneous connection Fixed problem with SYSMAC GATEWAY datatypes such as LREAL , DATE , DATE_TIME, TIME and TIME_OF_DAY Increased the maximum number of characters supported by the ASCII data type)from 40 to 1024 characters
N	1.14	André Körbes	Oct/21/2013	Added support for SINT, USINT and BYTE for NJ series PLC
O	1.15	Paulo Balbino	Jan/14/2014	Added support for writing groups using Cx-Compolet Fixed issue with Virtual Group creation when using tags on the Station field of the Main Driver Sheet
P	1.16	Paulo Balbino	May/16/2014	Fixed Write problem on CX-Compolet Fixed problem on write group for FINS
Q	1.17	Paulo Balbino	Jun/26/2014	Fixed Write problem with CV PLC
R	1.18	Paulo Balbino	Jan/22/2015	Fixed problem with String Added note regarding the Sysmac Gateway
S	1.19	Anushree Phanse	Apr/18/2017	Fixed issue of log not showing correct invalid items. Removed Case Sensitivity when using Tag Names which caused errors in communication.
T	1.20	Anushree Phanse	Apr/11/2018	Fixed issue of the driver crashing in specific conditions.
U	1.21	Anushree Phanse	Jan/24/2019	Fixed an issue with address validation