

Communication Driver KOYO

Driver for Communication with
Automation Direct Devices
Using DirectLOGIC Ethernet Modules

Contents

INTRODUCTION	2
GENERAL INFORMATION.....	3
DEVICE CHARACTERISTICS	3
LINK CHARACTERISTICS.....	3
DRIVER CHARACTERISTICS	3
CONFORMANCE TESTING.....	4
INSTALLING THE DRIVER.....	6
CONFIGURING THE DRIVER	7
SETTING THE COMMUNICATION PARAMETERS.....	7
CONFIGURING THE STANDARD DRIVER WORKSHEET	9
CONFIGURING THE MAIN DRIVER WORKSHEET	13
DEVICE CONFIGURATION	14
EXECUTING THE DRIVER.....	15
TROUBLESHOOTING	16
SAMPLE APPLICATION	18
REVISION HISTORY	19

Introduction

The KOYO driver enables communication between the Studio system and some of the Automation Direct devices using their CCM protocol by Ethernet or Serial, according to the specifications discussed in this document.

This document was designed to help you install, configure and execute the KOYO driver. The information is organized as follows:

- **Introduction:** Provides an overview of the driver documentation.
- **General Information:** Provides information needed to identify all the required components (hardware and software) used to implement communication between Studio and the KOYO driver.
- **Installing the Driver:** Explains how to install the KOYO driver.
- **Configuring the Driver:** Explains how to configure the communication 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 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 product’s *Technical Reference Manual*.
- This document also assumes that you are familiar with the Windows NT/2000/XP environment. If you are unfamiliar with Windows NT/2000/XP, 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 driver and devices.

The information is organized into the following sections:

- Device Characteristics
- Link Characteristics
- Driver Characteristics

Device Characteristics

This driver is compatible with the following devices:

- **Manufacturer:** Automation Direct
- **Compatible Equipment:**
 - DirectLog 205 Koyo Family (DL240/DL250 CPUs) + DirectLOGIC Ethernet modules (H2-ECOM)
- **Automation Direct PLC programmer software:** DIRECTSOFT32

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

Link Characteristics

To establish communication, you must use links with the following specifications:

- **Device communication port:** Ethernet or COM port
- **Physical protocol:** Ethernet or Serial
- **Logic protocol:** IPX, TCP/IP + CCM Protocol or Serial
- **Device Runtime software:** Host Engineering Library to DirectLOGIC Ethernet modules
- **Specific PC Board:** Ethernet board
- **Adapters/Converters:** DirectLOGIC Ethernet modules
- **Cable Wiring:** Ethernet cable

Driver Characteristics

The KOYO driver is composed of the following files:

- **KOYO.INI:** Internal file of the driver. It should not be modified by the user.
- **KOYO.MSG:** Error messages for each error code. It should not be modified.
- **KOYO.PDF:** Provides detailed documentation about the driver
- **KOYO.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 `KOYO.PDF` document.

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

- Windows 9x
- Windows 2000
- Windows NT

- Windows CE (x86 / SH3 / SH4 / MIPs / ARM / PPC)

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

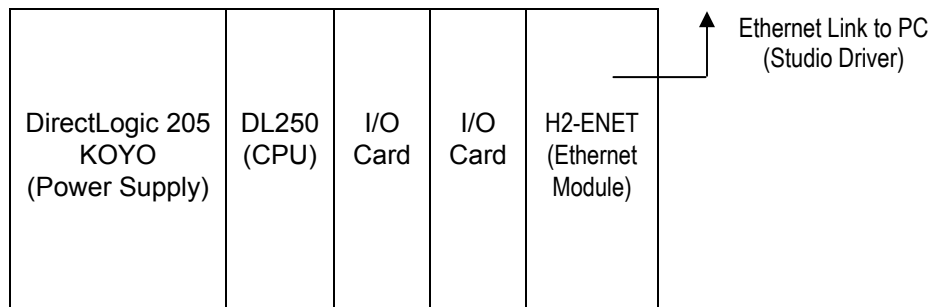
The KOYO driver supports the following registers:

Register Type	Length	Write	Read	Bit	Integer	Float
X or V (Input Point)	1 Bit	–	•	•	•	–
Y or V (Output Point)	1 Bit	•	•	•	•	–
C or V (Control Relay)	1 Bit	•	•	•	•	–
SP or V (Special Relay)	1 Bit	•	•	•	•	–
V (Timer Current Value)	2 Bytes	•	•	•	•	–
T or V (Timer Status Bits)	1 Bit	•	•	•	•	–
V (Counter Current Value)	2 Bytes	•	•	•	•	–
CT or V (Counter Status Bits)	1 Bit	•	•	•	•	–
V (Data Words)	2 Bytes	•	•	•	•	–
V (Stages)	2 Bytes	•	•	•	•	–
V (System Parameters)	2 Bytes	•	•	•	•	–

Conformance Testing

The following hardware/software was used for conformance testing:

- Configuration:**
 - PLC Family: DirectLogic 205
 - CPU Module: DL250 CPU
 - Ethernet Module: H2-ENET
 - Ethernet Module Port: 10BASE-T ETHERNET PORT
 - H2-ENET Module IP Address: 216.68.246.57
- Cable:** Ethernet cable.



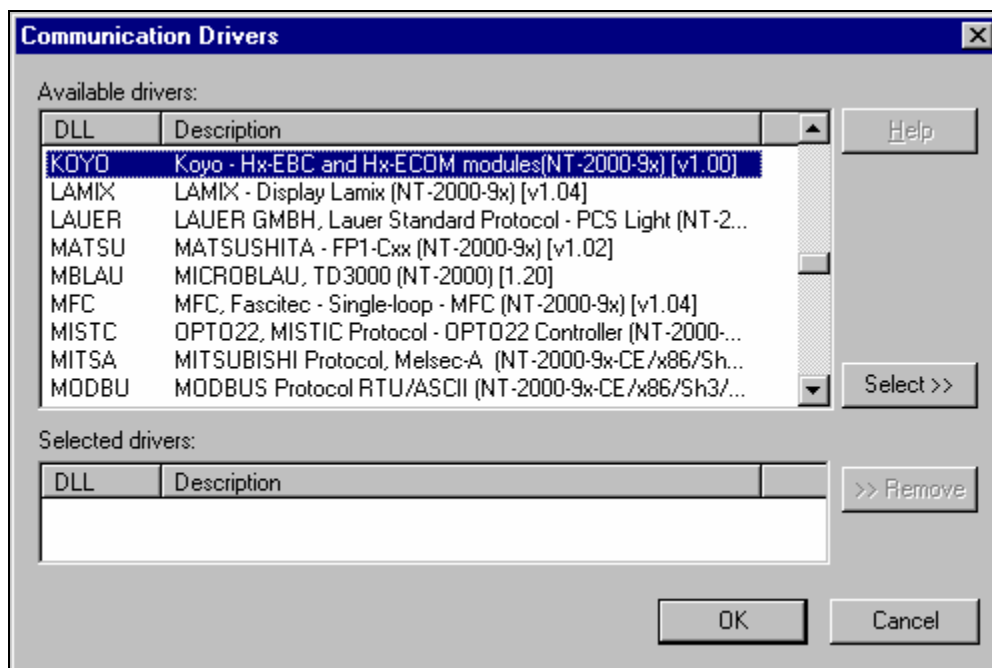
Driver Version	Studio Version	Operating System (development)	Operating System (target)	Equipment
1.00	4.1	Windows 2000 + Service Pack 1	Windows 2000 + Service Pack 1	DL205 CPU + F2-08SIM + D2-08TR + "H2-ENET"

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



Communication Drivers Dialog

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

 **Note:**

It is not necessary to install any other software on the PC to enable communication between the host and the device. However, to download the custom program to the device, it is necessary to install a version of KOYO programmer software, such as DIRECTSOFT32. Please see the DIRECTSOFT32 documentation for installation procedures.

 **Caution:**

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 KOYO driver, you must configure the driver. Configuring the KOYO driver is done in two parts:

- Specifying communication parameters
- Defining communication tags and controls in the Communication tables or Driver Worksheets

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.

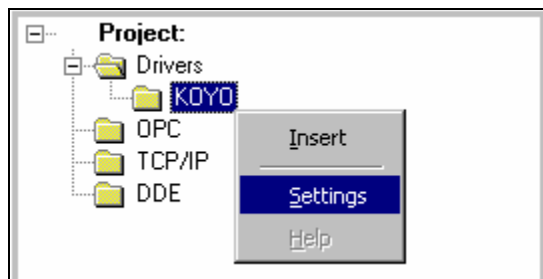
 **Note:**

For a detailed description of the Studio *Standard* and *MAIN* Driver Worksheets, and information about configuring the standard fields, review the product's *Technical Reference Manual*.

Setting the Communication Parameters

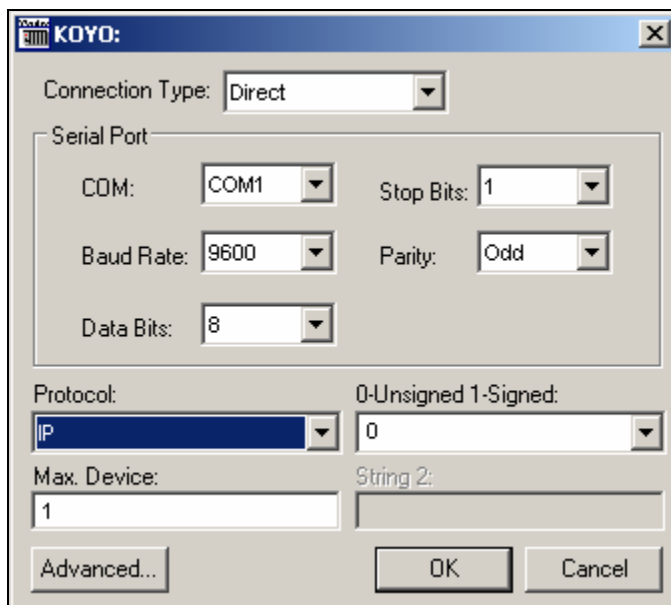
Use the following steps to configure the communication parameters (which are valid for all Driver Worksheets configured in the system):

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 *KOYO* 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 KOYO: Communication Parameters dialog displays (as follows).



KOYO: Communication Parameters Dialog

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

Parameter	Default Value	Valid values	Description
Protocol	0	IP, IPX or Serial	Transport protocol used in the Ethernet or Serial configuration of the computer
Max. Device	0	0 to 32	Maximum number of DirectLOGIC devices included in the network.
0-Unsigned 1-Signed	0	0 or 1	Set type of data. (Unsigned or Signed)



Caution:

The IPX transport protocol is not supported for the Windows CE operating system.



Note:

The device MUST be configured with exactly the same values defined in the *Communication Parameters* window of the KOYO driver.

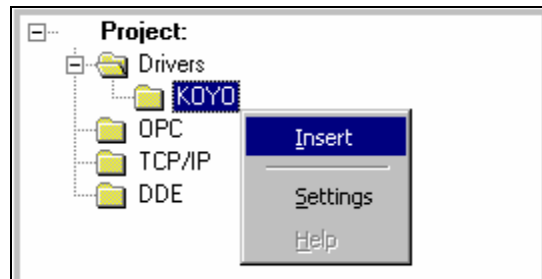
5. Clicking the **Advanced** button opens the *Advanced Settings* dialog, which contains additional communication parameters. Although *you should not change the default settings at this time*, you can consult the *Studio Technical Reference Manual* for information about configuring these parameters for future reference.

Configuring the Standard Driver Worksheet

This section explains how to configure a Standard Driver Worksheet (or communication table). You can configure multiple Driver Worksheets — each of which is divided into a Header section and a Body section.

Use the following steps to create a new Standard Driver Worksheet:

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 *KOYO* subfolder.
3. When the pop-up menu displays (as shown in the following figure), select the **Insert** option.



Inserting a New Worksheet

⇒ **Tip:**

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 period in which each group of tags must be written or read. Also, we recommend configuring the communication addresses in sequential blocks to improve performance.

The KOYO.drv dialog box displays (similar to the following figure).

The screenshot shows a dialog box titled "Koyo001.drv". It contains several configuration fields and a table. The fields are:

- Description: A text box containing 'X' and a checkbox for "Increase priority".
- Read Trigger: A text box.
- Enable Read when Idle: A text box containing "erd[1]".
- Read Completed: A text box.
- Read Status: A text box.
- Write Trigger: A text box.
- Enable Write on Tag Change: A text box containing "ewr[1]".
- Write Completed: A text box.
- Write Status: A text box.
- Station: A text box containing "N:UNIT1".
- Header: A text box containing "X".
- Min: A text box with an unchecked checkbox.
- Max: A text box with an unchecked checkbox.

 Below the fields is a table with the following data:

	Tag Name	Address	Div	Add
1	DataX[1]	0		
2	DataX[2]	1		
3	DataX[3]	5		
4	DataX[4]	7		
5	DataX[5]	10		

KOYO Driver Worksheet

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 configure the **Station**, **Header** and **Address** fields on this worksheet.
- Use the **Station** field to define the PLC address. This value must match the Device Number parameter configured in the device using the Automation Direct PLC programmer software. If the communication is **Serial**, enter the ID Number.

Valid values: **N:**<Device Name> or **I:**<ID Number>

- Use the **Header** field to define the type of variables that will be read from/written to the device. You must use the following syntax when defining the **Header**:

<Type>

Where:

<Type> is the register type.

Valid values: **x** (default) (Input Points - Bit), **y** (Output Points - Bit), **c** (Control Relays - Bit), **sp** (Special Relays - Bit), **t** (Timers - Bit), **ct** (Counters), **s** (Stages) or **v** (V memory)

Studio automatically verifies the **Header** to ensure the syntax is valid. If the syntax is incorrect, Studio automatically inserts the default value (**x**) into this field.

You can type {Tag} into this field, but you must be certain that the tag's value is correct and that you are using the correct syntax; otherwise, you will get an invalid Header error. The following table lists all the data types and address ranges that are valid for the KOYO driver.

Header Field Information			
Data Types	Sample Syntax	Valid Range of Initial Addresses per Worksheet	Comments
Input Points	X	X0 – X777	Input Points in the bit data format
	V	V40400 – V40437	Input Points in the word data format
Output Points	Y	Y0 – Y777	Output Points in the bit data format
	V	V40500 – V40537	Output Points in the word data format
Control Relays	C	C0 – C1777	Control Relays in the bit data format
	V	V40600 – V40677	Control Relays in the word data format
Special Relays	SP	SP0 – C777	Special Relays in the bit data format
	V	V41200 – V41237	Special Relays in the word data format
Timer Current Value	V	V0 – V377	Timer Current Values in the word data format
Timer Status Bits	T	T0 – T377	Timer Status Bits in the bit data format
	V	V41100 – V41117	Timer Status Bits in the word data format
Counter Current Value	V	V1000 – V1177	Counter Current Values in the word data format
Counter Status Bits	T	CT0 – CT177	Counter Status Bits in the bit data format
	V	V41140 – V41147	Counter Status Bits in the word data format
Data Words	V	V1400 – V7377 V10000 – V17777	Data Words in the word data format
Stages	S	S0 – S1777	Stages in the bit data format
	V	V41000 – V41077	Stages in the word data format
System Parameters	V	V7400 – V7777 V37000 – V37777	System Parameters in the word data format

Note:
 Most of the Register types can be treated as bit or as word data. When the Discrete Memory Syntax (X, Y, C, SP, T, CT or S) is used, the bits are always addressed individually. When the Word Memory Syntax (V) is used, the word memory (32 bits) is addressed.

- **Address field:** Use this field to associate each tag to its respective device address. The **Tag Name** column must be filled with the application tag name. The **Address** column must be filled with the address of the device registers, according to the following syntax:

To Discrete Memory Registers (X, Y, C, SP, T, CT and S):

<Address> (for example, 5)

To Word Memory Registers (V):

<Address> . <Bit> (for example, 40400 . 5) – Data in decimal format

<Address> : B (for example, 1000 : B) – Data in BCD format

Where:

<Address>: Register address number. The address is ALWAYS configured in OCTAL format

<Bit>: Bit number (from 0 to 15) from the word address. This is an optional parameter.

Address Field Samples			
Register Type	Address on the Device (octal)	Header Field	Address Field (octal)
Input Points	X0000	X	0
		V	40400.0
	X0030	X	30
		V	40401.8
	X0000 to X0017 (1 word)	V	40400
Output Points	Y0007	Y	7
		V	40500.7
	Y0026	Y	26
		V	40501.6
	Y0020 to Y0037 (1 word)	V	40501
Control Relays	C0010	C	10
		V	40600.8
	C0053	C	53
		V	40602.11
	C0000 to C0017 (1 word)	V	40600
Special Relays	SP0017	SP	17
		V	41200.15
	SP0020	SP	20
		V	41201.0
	SP0040 to SP0057 (1 word)	V	41202
Timer Current Value	V0000 – Decimal Format	V	0
	V0000 – BCD Format	V	0:B
	V0012 – Decimal Format	V	12
	V0012 – BCD Format	V	12:B
Timer Status Bits	T0007	T	7
		V	41100.7
	T0021	T	21
		V	41101.1
	T0000 to T0017 (1 word)	V	41100

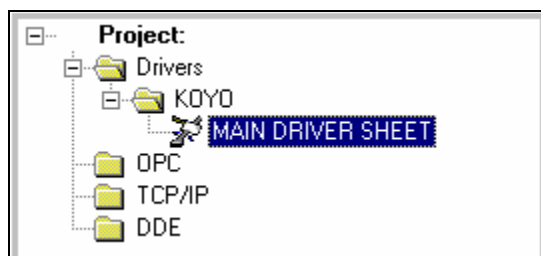
Counter Current Value	V1000 – Decimal Format	V	1000
	V1000 – BCD Format	V	1000:B
	V1012 – Decimal Format	V	1012
	V1012 – BCD Format	V	1012:B
Counter Status Bits	CT0007	CT	7
		V	41140.7
	CT0021	CT	21
		V	41141.1
CT0000 to CT0017 (1 word)		V	41140
Data Words	V1400 – Decimal Format	V	1400
	V1400 – BCD Format	V	1400:B
	V12000 – Decimal Format	V	12000
	V12000 – BCD Format	V	12000:B
Stages	S0007	S	7
		V	41000.7
	S0021	S	21
		V	41001.1
S0000 to S0017 (1 word)		V	41000
System Parameters	V7772 – BCD Format (Day of week)	V	7772:B
	V7773 – BCD Format (Month)	V	7773:B
	V7774 – BCD Format (Year)	V	7774:B

➔ Caution:

You must not configure a range of addresses greater than the maximum block size (data buffer length) supported within the same worksheet: 126 for V registers or 254 for X, Y, C, SP, T, CT or S registers.

Configuring the Main Driver Worksheet

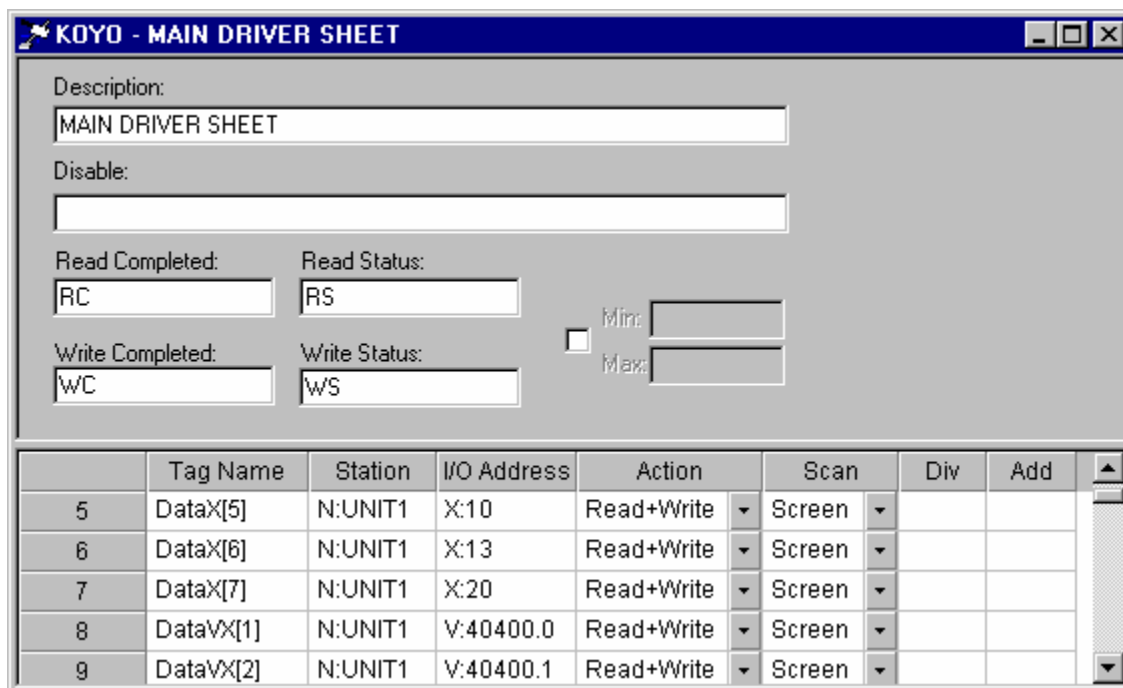
When you add the KOYO driver to your application, Studio automatically adds a *MAIN Driver Worksheet (MDS)* to the KOYO driver folder, as shown in the following figure.



Main Driver Sheet in the KOYO Folder

The MAIN DRIVER SHEET provides a simple way to associate Studio tags to addresses in the PLC. Most of the MAIN DRIVER SHEET entries are standard for any driver. Refer to the Studio *Technical Reference Manual* for configuration of the standard fields. The fields that require specific syntax for this driver are described below.

1. Double-click the **MAIN DRIVER SHEET** icon to open the *MAIN* Driver Worksheet, similar to the following:



Main Driver Worksheet

2. Use the following information to configure the **Station** and **I/O Address** fields:
 - **Station:** PLC address (**N:<Device Name>** or **I:<ID Number>**). This value must match the Device Number parameter configured in the device using the Automation Direct PLC programmer software for IP or IPX Communication). For Serial Communication, enter the **ID Number**
 - **I/O Address:** Address of each register from the PLC. The syntax used in this field is described below:

To Discrete Memory Registers (X, Y, C, SP, T, CT and S):

<Type>: <Address> (for example, **X: 5**)

To Word Memory Registers (V):

<Type>: <Address> . <Bit> (for example, **V:40400.5**) – Data in decimal format

<Type>: <Address> : B (for example, **V:1000:B**) – Data in BCD format

Where:

<Type>: Register type. **Valid values:** **x** (Inputs Points - Bit), **y** (Output Points - Bit), **c** (Control Relays - Bit), **sp** (Special Relays - Bit), **t** (Timers - Bit), **ct** (Counters), **s** (Stages) or **v** (V memory)

<Address>: Register address number. The address is ALWAYS configured in OCTAL format

<Bit>: Bit number (from 0 to 15) from the word address. This is an optional parameter.

Device Configuration

The Koyo PLC must be configured to exchange data by the H2-ENET (Ethernet Module).

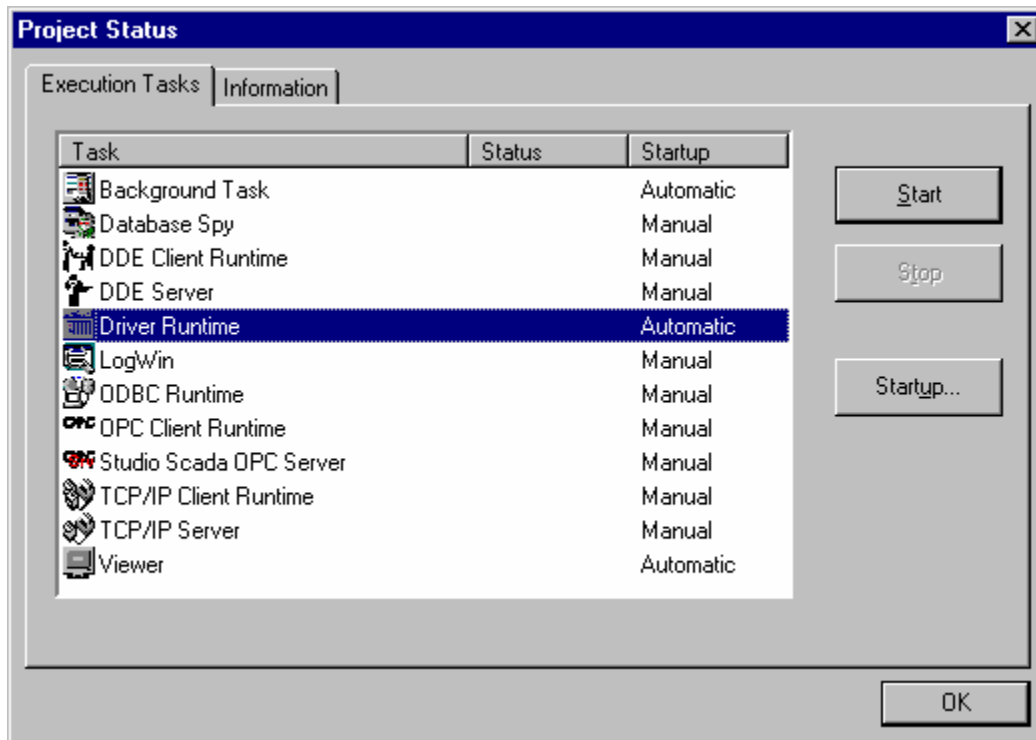
Executing the Driver

After you add the KOYO 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 KOYO 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 the failure that occurred.

Error Code	Description	Possible Causes	Procedure to Solve
0	OK	Communication without problems	None
1	Block Size Error	Address offset greater than 126 to "V" operands, and greater than 254 to other operands.	Check on the worksheet receiving the error. If the configured offset is greater than the limit, correct it.
2	Invalid Header	An invalid Header has been typed, or the tag has an invalid configuration.	Type a valid Header either on the header field or on the tag value. See the "Header Field Information" table in this document for valid headers.
5	Error opening transport	Transport protocol (IP or IPX) must be installed and configured correctly in the computer.	Check the network configuration. Install the correct transport protocol and configure it.
6	Nack response	Invalid configuration in the header or in the address	- Check the configuration of operand block size. - Check the operand inside the device.
7	Invalid answer	Unexpected length of the answer	- Check the cable wiring. - Check the PLC state. It must be RUN. - Check the station number. - Check the configuration.
8	Timeout Start Message	- Disconnected cables - PLC turned off, or in Stop or error mode - Station number is wrong	- Check the cable wiring. - Check the PLC state. It must be RUN - Check the station number. - Check the configuration.
9	Invalid device number	ID number or Device name configured with invalid value	Check the device configuration (Id number and device name). Make corrections in the station field.
10	Error opening device	- Disconnected cables - PLC turned off, or in Stop or error mode - Station number is wrong	- Check the cable wiring. - Check the PLC state. It must be RUN - Check the station number. - Check the configuration.
100	Invalid Protocol	Protocol not compatible	Verify that the PLC supports the DirectLOGIC Protocol.
101	Communication Error	Error in communication	- Check the station number. - Verify that the addresses are valid.
102	LRC Error	Invalid LRC Check	Make sure the cable is plugged in correctly.
-15	Timeout Start Message	- Disconnected cables - PLC turned off, or in Stop or error mode - Station number is wrong - RTS/CTS control settings are wrong	- Check the cable wiring. - Check the PLC state. It must be RUN - Check the station number. - Check the RTS/CTS configuration (see the Studio <i>Technical Reference Manual</i> for valid configurations).
-17	Timeout between rx char.	- PLC in stop or error mode - Station number is wrong - Parity is wrong - RTS/CTS configuration settings are wrong	- Check the cable wiring. - Check the PLC state. It must be RUN - Check the station number. - Check the RTS/CTS configuration (see the Studio <i>Technical Reference Manual</i> for valid configurations).

⇒ **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 enable the log at the unit (**Tools** → **Remote Logwin**) and verify the `ce1og.txt` file created at the target unit.

If you are unable to establish communication with the PLC, try to establish communication between the PLC Programming Tool and the PLC. Quite frequently, communication is not possible because you have a hardware or cable problem, or a PLC configuration error. After successfully establishing communication between the device's Programming Tool and the PLC, you can retest the supervisory driver.

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**.
- **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

You will find a sample application in the `/COMMUNICATION EXAMPLES/KOYO` directory. We strongly recommend that you use this sample application to test the KOYO driver before configuring your own customized application, for the following reasons:

- To better understand the information provided in each 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.
3. Execute the *Viewer* module in Studio to display information about the driver communication.



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
A	1.00	Roberto V. Junior	23-Apr-2001	First driver version
B	1.01	Roberto V. Junior	11-Jun-2001	<ul style="list-style-type: none">▪ Updated to support WindowsCE operating system▪ The driver no longer requires additional libraries.
C	1.02	Roberto V. Junior	13-Jul-2001	Fixed memory leak bug
D	1.03	Roberto V. Junior	19-Jul-2001	Modified algorithm for compatibility with ARM processor
E	1.04	Roberto V. Junior	07-May-2002	<ul style="list-style-type: none">▪ Fixed multiple station bug▪ Improved performance
F	1.05	Roberto V. Junior	17-May-2002	Included Signed or Unsigned value
G	1.06	Leandro G. Coeli	11-Mar-2005	Included Serial Communication
H	1.07	Leandro G. Coeli	07-Apr-2005	Fixed problems over IPX protocol