

OPTO Communication Driver

Driver for Serial Communication
with Opto 22 Devices Using the Optomux Protocol

Contents

INTRODUCTION2

GENERAL INFORMATION.....3

 DEVICE SPECIFICATIONS.....3

 NETWORK SPECIFICATIONS.....3

 DRIVER CHARACTERISTICS3

SELECTING THE DRIVER5

CONFIGURING THE DRIVER6

 CONFIGURING THE COMMUNICATION SETTINGS6

 CONFIGURING THE DRIVER WORKSHEETS9

EXECUTING THE DRIVER 14

TROUBLESHOOTING 15

REVISION HISTORY..... 17

Introduction

The OPTO driver enables communication between the Studio system and Opto 22 analog and digital I/O modules using the Optomux protocol, according to the specifications discussed in this document.

This document will help you to select, configure and execute the OPTO driver, and it is organized as follows:

- **Introduction:** This section, which provides an overview of the document.
- **General Information:** Identifies all of the hardware and software components required to implement communication between the Studio system and the target device.
- **Selecting the Driver:** Explains how to select the OPTO driver in the Studio system.
- **Configuring the Device:** Describes how the target device must be configured to receive communication from the OPTO driver.
- **Configuring the Driver:** Explains how to configure the OPTO driver in the Studio system, including how to associate database tags with device registers.
- **Executing the Driver:** Explains how to execute the OPTO driver during application runtime.
- **Troubleshooting:** Lists the most common errors for this driver, their probable causes, and basic procedures to resolve them.
- **Sample Application:** Explains how to use a sample application to test the OPTO driver configuration
- **Revision History:** Provides a log of all changes made to the driver and this documentation.

Notes:

- This document assumes that you have read the “Development Environment” chapter in Studio’s *Technical Reference Manual*.
- This document also assumes that you are familiar with the Microsoft Windows NT/2000/XP environment. If you are not familiar with Windows, then we suggest using the **Help** feature (available from the Windows desktop **Start** menu) as you work through this guide.

General Information

This chapter identifies all of the hardware and software components required to implement serial communication between the OPTO driver in Studio and a target Opto 22 device.

The information is organized into the following sections:

- Device Specifications
- Network Specifications
- Driver Characteristics
- Conformance Testing

Device Specifications

To establish communication, your target device must meet the following specifications:

- **Manufacturer:** Opto 22
- **Compatible Equipment:** Opto 22 analog and digital I/O modules

Network Specifications

To establish communication, your device network must meet the following specifications:

- **Physical Protocol:** RS-485
- **Logic Protocol:** Optomux
- **Device Runtime Software:** None

Driver Characteristics

The OPTO driver package consists of the following files, which are automatically installed in the \DRV subdirectory of Studio:

- **OPTO.INI:** Internal driver file. *You must not modify this file.*
- **OPTO.MSG:** Internal driver file containing error messages for each error code. *You must not modify this file.*
- **OPTO.PDF:** This document, which provides detailed information about the OPTO driver.
- **OPTO.DLL:** Compiled driver.

 **Note:**

You must use Adobe Acrobat® Reader™ to view the **OPTO.PDF** document. You can install Acrobat Reader from the Studio installation CD, or you can download it from Adobe's Web site.

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

- Windows NT/2000/XP

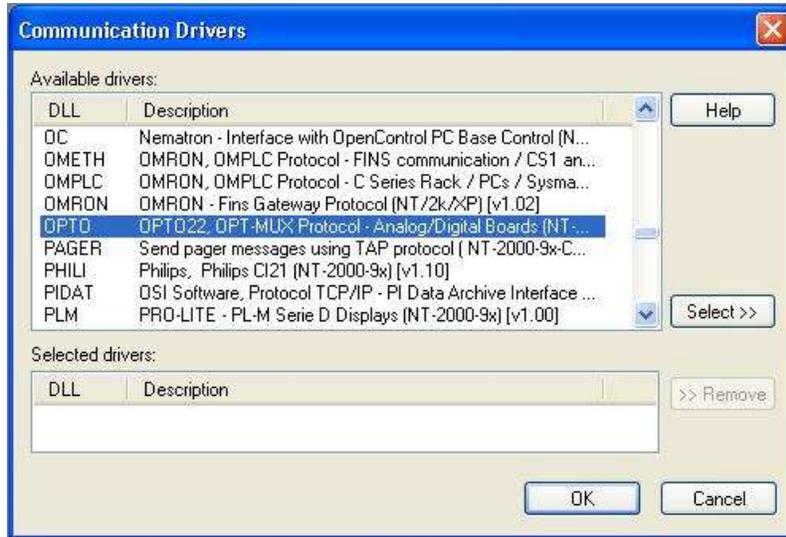
The OPTO driver supports the following registers:

Commands	Digital	Analog	Read	Write
A (PowerUP-Clear)	•	•	–	•
B (Reset)	•	•	–	•
C (Set TurnAround Delay)	•	•	–	•
D (Set Digital WatchDog Delay)	•	–	–	•
E (Set Protocol)	•	•	–	•
G (Configure Positions)	•	•	–	•
J (Write OutPut - Turn ON/OFF)	•	•	–	•
K (Active OutPut - Turn ON)	•	–	•	•
L (Deactive OutPut - Turn OFF)	•	–	•	•
M (Read ON/OFF Status)	•	–	•	–
N (Set Latch Edges)	•	–	–	•
O (Set OFF-to-ON Latches)	•	–	–	•
P (Set ON-to-OFF Latches)	•	–	–	•
S (Clear Latches)	•	–	–	•
T (Start/Stop Counters)	•	–	–	•
U (Start Counters)	•	–	–	•
V (Stop Counters)	•	–	–	•
W (Read Counter)	•	–	•	–
X (Read and Clear Counter)	•	–	•	–
Y (Clear Counters)	•	–	–	•
m (Enhanced Digital WatchDog)	•	–	–	•

Selecting the Driver

When you install Studio, all of the communication drivers are automatically installed in the `\DRV` subdirectory but they remain dormant until manually selected for specific applications. To select the OPTO driver for your Studio application:

1. From the main menu bar, select **Insert** → **Driver** to open the *Communication Drivers* dialog.
2. Select the **OPTO** driver from the *Available Drivers* list, and then click the **Select** button.



Communication Drivers Dialog

3. When the **OPTO** driver is displayed in the **Selected Drivers** list, click the **OK** button to close the dialog. The driver is added to the *Drivers* folder, in the *Comm* tab of the Workspace.

Note:

It is not necessary to install any other software on your computer to enable communication between Studio and your target device. However, this communication can only be used by the Studio application; it cannot be used to download control logic to the device. To download control logic to an Opto 22 device, you must also install the Opto 22 programming software. For more information, please consult the documentation provided by the device manufacturer.

Attention:

For safety reasons, you must take special precautions when installing any physical hardware. Please consult the manufacturer's documentation for specific instructions.

Configuring the Driver

Once you have selected the OPTO driver in Studio, you must properly configure it to communicate with your target device. First, you must set the driver's communication settings to match the parameters set on the device. Then, you must build driver worksheets to associate database tags in your Studio application with the appropriate addresses (registers) on the device.

Configuring the Communication Settings

The communication settings are described in detail in the "Communication" chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

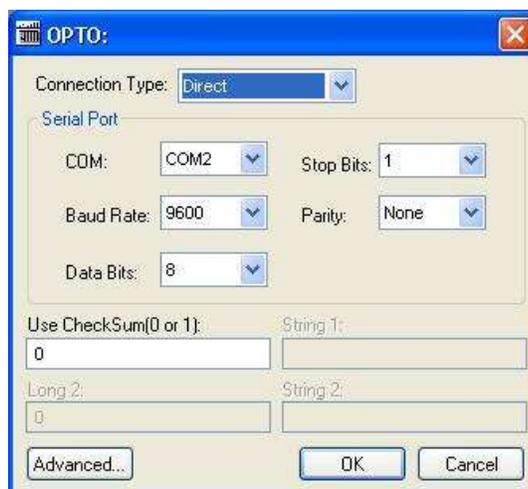
For the purposes of this document, only OPTO driver-specific settings and procedures will be discussed here. To configure the communication settings for the OPTO driver:

1. In the *Workspace* pane, select the *Comm* tab and then expand the *Drivers* folder. The OPTO driver is listed here as a subfolder.
2. Right-click on the *OPTO* subfolder and then select the **Settings** option from the pop-up menu:



Select Settings from the Pop-Up Menu

The *OPTO: Communication Settings* dialog is displayed:



OPTO: Communication Settings Dialog

- In the *Communication Settings* dialog, configure the driver settings to enable communication with your target device. To ensure error-free communication, the driver settings must *exactly match* the corresponding settings on the device. Please consult the manufacturer’s documentation for instructions how to configure the device and for complete descriptions of the settings.

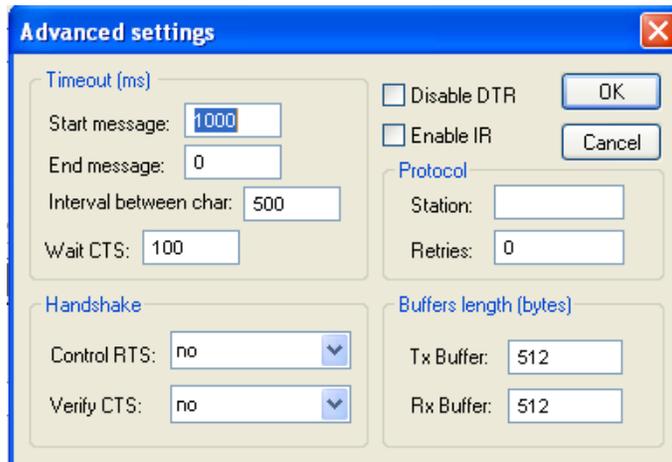
Depending on your circumstances, you may need to configure the driver *before* you have configured your target device. If this is the case, then take note of the driver settings and have them ready when you later configure the device.

➔ Attention:
 For safety reasons, you **must** take special precautions when connecting and configuring new equipment. Please consult the manufacturer’s documentation for specific instructions.

The communication settings and their possible values are described in the following table:

Parameters	Default Value	Valid Values	Description
Use CheckSum(0 or 1)	0	0	Use CheckSum
		1	Do not use CheckSum

- If you are using a Data Communication Equipment (DCE) converter (e.g., 232/485) between your PC and your target device, then you must also adjust the **Control RTS** (Request to Send) setting to account for the converter. In the *Communication Settings* dialog, click the **Advanced** button to open the *Advanced Settings* dialog:



Advanced Settings Dialog

When the dialog is displayed, configure the **Control RTS** setting using the following information:

Setting	Default	Values	Description
Control RTS	no	no	Do not set the RTS (Request to Send) handshake signal. IMPORTANT: If you are using Windows 95/98 or Windows CE with the correct RS232/RS485 adapter (i.e. without RTS control), then you must select this option.
		yes	Set the RTS (Request to Send) handshake signal before communication. IMPORTANT: If you are using Windows NT and the Cutler-Hammer RS232/RS485 adapter, then you must select this option.
		yes+echo	Set the RTS (Request to Send) handshake signal before communication, and echo the signal received from the target device.

➤ **Attention:**

If you incorrectly configure the **Control RTS** setting, then runtime communication will fail and the driver will generate a –15 error. See “Troubleshooting” for more information.

You do not need to change any other advanced settings at this time. You can consult the *Studio Technical Reference Manual* later for more information about configuring these settings.

5. Click **OK** to close the *Advanced Settings* dialog, and then click **OK** to close the *Communication Settings* dialog.

Configuring the Driver Worksheets

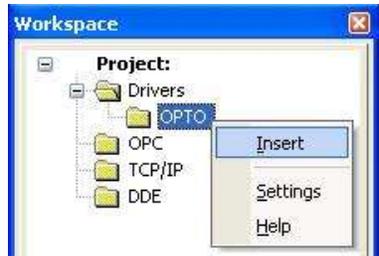
A selected driver includes one or more driver worksheets, which are used to associate database tags in Studio with registers on the target device. Each worksheet is triggered by specific application behavior, so that the tags / registers defined on that worksheet are scanned only when necessary — that is, only when the application is doing something that requires reading from or writing to those specific tags / registers. Doing this optimizes communication and improves system performance.

The configuration of these worksheets is described in detail in the “Communication” chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

Note:
We recommend configuring device registers in sequential blocks in order to maximize performance.

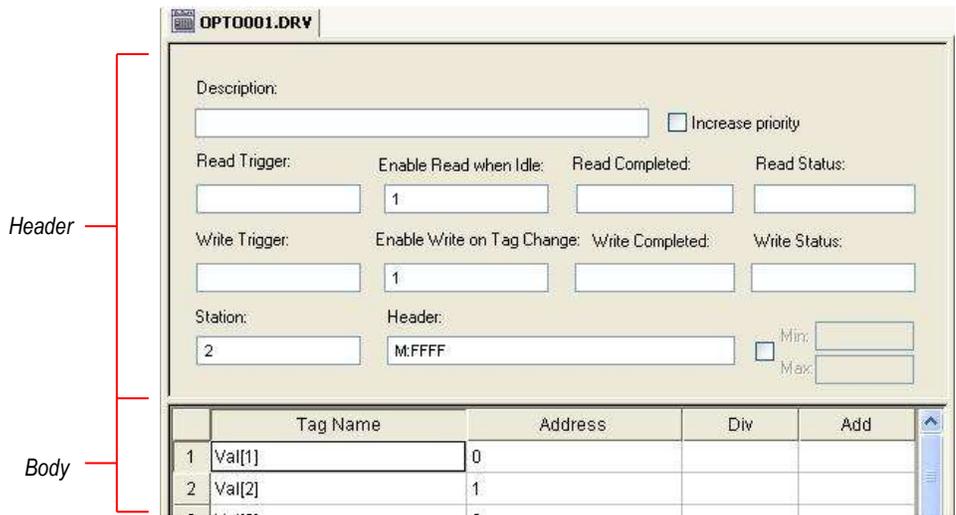
To insert a new driver worksheet:

1. In the *Comm* tab, open the *Drivers* folder and locate the *OPTO* subfolder.
2. Right-click on the *OPTO* subfolder, and then select **Insert** from the pop-up menu:



Inserting a New Worksheet

A new OPTO driver worksheet is inserted into the *OPTO* subfolder, and the worksheet is opened for configuration:



OPTO Driver Worksheet

Note:
 Worksheets are numbered in order of creation, so the first worksheet is **OPTO001 .drv**.

Most of the fields on this worksheet are standard for all drivers; see the “Communication” chapter of the *Technical Reference Manual* for more information on configuring these fields. However, the **Station**, **Header**, and **Address** fields use syntax that is specific to the OPTO driver.

3. Configure the **Station** and **Header** fields as follows:

- **Station** field: Specify the Optomux address (from 0 to 255) of the target device on the serial network.
- **Header** field: Specify the command to be executed and the configuration of the board, using the following syntax:

<Command>: <Configuration>

Examples — **E: 61B7** or **M: FFFF**

Where:

- **<Command>** is a one-letter command code, as described in the table on the following page.
- **<Configuration>** is a four-character hexadecimal that represents the data attached to the specified **<Command>**. The meaning of the data varies according to the command; see the table on the following page. When the data have no predefined meaning, 0 equals Input and 1 equals Output.

The table below shows how the data is converted to a hexadecimal. *This is only an example.*

POSITION	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA	0	1	1	0	0	0	0	1	1	0	1	1	0	1	1	1
Hexadecimal	6				1				B				7			

Attention:
 You cannot leave the **Station** and **Header** fields blank; you must specify some values.

The following table describes all of the available commands for the **Header** field, as well as how the attached data is used with the command. For more information about these commands, please refer to the Opto 22 / Optomux documentation.

Command	Action	Valid information	Comment																											
A	PowerUp-Clear (Write)	Don't use DATA or POSITIONS	Prevents Optomux from returning a Power-up Clear Expected error message in response to the first instruction following application of power.																											
B	Reset (Write)	Don't use DATA or POSITIONS	Digital: <ul style="list-style-type: none"> ▪ All outputs turned off ▪ All positions then configured as inputs ▪ Protocol as set by jumper B10 ▪ Watchdog timer disabled ▪ Turnaround delay = 0 ▪ Counters/duration timers cancelled ▪ Latches cleared ▪ Timer resolution = 10 ms 																											
			Analog: <ul style="list-style-type: none"> ▪ 0 scale written to all output modules ▪ All positions then configured as inputs ▪ Protocol as set by jumper B10 ▪ Watchdog timer disabled. ▪ Turnaround Delay = 0 ▪ All offsets set to 0 ▪ All gain coefficients set to 1 ▪ All averaging cancelled ▪ All temperature probe types cancelled 																											
C	Set TurnAround Delay (Write)	Use only DATA	Tells Optomux to wait for a specified time before responding to commands sent from host. <table border="0"> <thead> <tr> <th>Data</th> <th>Time</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No Delay</td> </tr> <tr> <td>1</td> <td>10 ms</td> </tr> <tr> <td>2</td> <td>100 ms</td> </tr> <tr> <td>3</td> <td>500 ms</td> </tr> </tbody> </table>	Data	Time	0	No Delay	1	10 ms	2	100 ms	3	500 ms																	
Data	Time																													
0	No Delay																													
1	10 ms																													
2	100 ms																													
3	500 ms																													
D	Set Digital WatchDog Delay(Write)	Use only DATA	Instructs digital Optomux to monitor activity on the serial communications link and to take a predetermined action if there is no activity within a specified time. <table border="0"> <thead> <tr> <th>Data</th> <th>Time</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>0</td> <td></td> <td>Watchdog disabled</td> </tr> <tr> <td>1</td> <td>10 secs</td> <td>Turn all outputs OFF</td> </tr> <tr> <td>2</td> <td>1 min</td> <td>Turn all outputs OFF</td> </tr> <tr> <td>3</td> <td>10 mins</td> <td>Turn all outputs OFF</td> </tr> <tr> <td>4</td> <td></td> <td>Watchdog disabled</td> </tr> <tr> <td>5</td> <td>10 secs</td> <td>Turn output 0 ON all other outputs OFF</td> </tr> <tr> <td>6</td> <td>1 min</td> <td>Turn output 0 ON all other outputs OFF</td> </tr> <tr> <td>7</td> <td>10 mins</td> <td>Turn output 0 ON all other outputs OFF</td> </tr> </tbody> </table>	Data	Time	Action	0		Watchdog disabled	1	10 secs	Turn all outputs OFF	2	1 min	Turn all outputs OFF	3	10 mins	Turn all outputs OFF	4		Watchdog disabled	5	10 secs	Turn output 0 ON all other outputs OFF	6	1 min	Turn output 0 ON all other outputs OFF	7	10 mins	Turn output 0 ON all other outputs OFF
Data	Time	Action																												
0		Watchdog disabled																												
1	10 secs	Turn all outputs OFF																												
2	1 min	Turn all outputs OFF																												
3	10 mins	Turn all outputs OFF																												
4		Watchdog disabled																												
5	10 secs	Turn output 0 ON all other outputs OFF																												
6	1 min	Turn output 0 ON all other outputs OFF																												
7	10 mins	Turn output 0 ON all other outputs OFF																												
E	Set Protocol (Write)	Use only DATA	<table border="0"> <thead> <tr> <th>Data</th> <th>Protocol</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Two-Pass Protocol</td> </tr> <tr> <td>1</td> <td>Four-Pass Protocol</td> </tr> </tbody> </table>	Data	Protocol	0	Two-Pass Protocol	1	Four-Pass Protocol																					
Data	Protocol																													
0	Two-Pass Protocol																													
1	Four-Pass Protocol																													

Command	Action	Valid information	Comment
G	Configure Positions	Use only POSITIONS	Identifies the function of positions as either inputs or outputs. <ul style="list-style-type: none"> ▪ If the value of Tag is 0, module position configured in address is configured to Inputs. ▪ If the value of Tag is 1, module position configured in address is configured to Output.
J	Write OutPut - Turn ON/OFF (Digital) (Write)	Use only POSITIONS	Turns output modules ON and OFF. <ul style="list-style-type: none"> ▪ If the value of Tag is 0 module position configured in address go to OFF. ▪ If the value of Tag is 1 module position configured in address go to ON.
	Write Analog OutPut - (Write)	Use DATA and POSITIONS	[position] - to set position [data]- value to be written to each position
K	Active OutPut - Turn ON (Read/Write)	Use only POSITIONS	Turns output modules ON. The module position configured in Address go to ON.
L	Deactive OutPut - Turn OFF (Read/Write)	Use only POSITIONS	Turns output modules OFF. The module position configured in Address go to OFF.
M	Read ON/OFF Status (Read)	Use only POSITIONS	Read state of module position (ON/OFF) configured in Address.
N	Set Latch Edges	Use only POSITIONS	Set positions configured as inputs and configured in the Address to latch on either ON-to-OFF or OFF-to-ON transitions.
O	Set OFF-to-ON Latches	Use only POSITIONS	Set positions configured as inputs and configured in the Address to latch on either OFF-to-ON transitions.
P	Set ON-to-OFF Latches	Use only POSITIONS	Set positions configured as inputs and configured in the Address to latch on either ON-to-OFF transitions.
Q	Read Latches	Use only POSITIONS	Read data indicating which of the module position configured in the address have latched. If value is 1 module position have been latched.
R	Read and Clear Latches	Use only POSITIONS	Read data indicating which of the inputs have latched and then resets the latches for specified input positions to the unlatched state. If value is 1 module position have been latched.
S	Clear Latches	Use only POSITIONS	Sets latches for specified input positions to unlatched state. The latches for specified input positions are configured in Address.
T	Start/Stop Counters	Use only POSITIONS	Starts and stops counting of OFF-to-ON transitions at specified input positions. The specified input positions are configured in Address.
U	Start Counters	Use only POSITIONS	Starts counting of OFF-to-ON transitions at specified input positions. The specified input positions are configured in Address.
V	Stop Counters	Use only POSITIONS	Stops counting of OFF-to-ON transitions at specified input positions. The specified input positions are configured in Address.
W	Read Counter	Use only POSITIONS	Returns the counter values for the specified input positions.

Command	Action	Valid information	Comment
x	Read and Clear Counter	Use only POSITIONS	Returns the counter values for the specified positions and then sets the counters for those positions to 0.
y	Clear Counters	Use only POSITIONS	Clear counters of specified input positions. The specified input positions are configured in Address.
m	Enhanced Digital WatchDog (Write)	Use DATA and POSITIONS	Instructs digital Optomux to monitor activity on the serial communications link and to take a specified action if there is no activity within a specified time.

- For each row of the table (i.e. each Tag association), configure the **Address** field either with the string **DATA** (to pass on the data from the **Header** field) or with a I/O register address.

The following table shows how register addresses are composed using the values in the **Header** and **Address** fields. *These are only examples.*

Header field	Tag Name	Address field	Description
E: 61B7	Protocol (value of protocol)	DATA	Set Protocol (write).
M: FFFF	Val [1]	0	Configure all with input.
	Val [2]	1	
	Val [3]	2	
	Val [4]	3	
	Val [5]	4	

 **Notes:**

- If the specified command uses only Data, configure the **Address** field with the string **DATA**.
- If the specified command uses Data and Positions, then configure only one row with the string **DATA**. If more than one row is configured with **DATA**, only the last row will be evaluated.

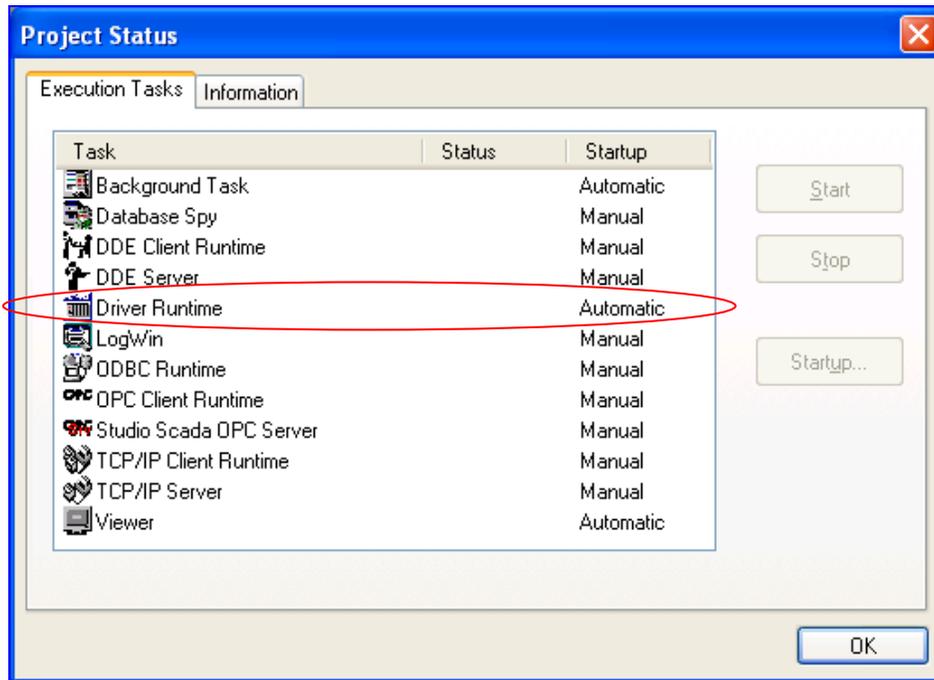
For more information about registers and addressing, please consult the manufacturer’s documentation.

Executing the Driver

By default, Studio will automatically execute your selected communication driver(s) during application runtime. However, you may verify your application's runtime execution settings by checking the *Project Status* dialog.

To verify that the communication driver(s) will execute correctly:

1. From the main menu bar, select **Project** → **Status**. The *Project Status* dialog displays:



Project Status Dialog

2. Verify that the *Driver Runtime* task is set to **Automatic**.
 - If the setting is correct, then proceed to step 3 below.
 - If the **Driver Runtime** task is set to **Manual**, then select the task and click the **Startup** button to toggle the task's *Startup* mode to **Automatic**.
3. Click **OK** to close the *Project Status* dialog.
4. Start the application to run the driver.

Troubleshooting

If the OPTO driver fails to communicate with the target device, then the database tag(s) that you configured for the **Read Status** or **Write Status** fields of the Main Driver Sheet 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
41	Invalid Address	Wrong address typed at driver worksheet	Check if the addresses values are valid.
42	Invalid Header	<ul style="list-style-type: none"> ▪ Wrong Header typed at the driver configuration worksheet. ▪ Invalid Read Command ▪ Invalid Write Command 	<ul style="list-style-type: none"> ▪ See at the section 4.3 some samples of valid Headers. ▪ Check if the read is allowed. ▪ -Check if the writing is allowed.
43	Invalid Command	Invalid command in header field	Check the command parameters in the header field
44	Error Checksum	Invalid CHECKSUM in response message	<ul style="list-style-type: none"> ▪ Check the cable wiring. ▪ Check the station number. ▪ Check the RTS/CTS configuration (see Studio Technical Reference Manual for valid configurations).
60	Power-Up Clear Expected - Command Ignored	A command other than "A" (Power-Up Clear) was attempted after power-up or power failure.	<ul style="list-style-type: none"> ▪ Once the error is received, it is unnecessary to execute Power-Up Clear. The next command will be executed normally. ▪ It will be necessary to reinitialize the Opto 22 I/O module.
71	Undefined Command	The command character was not a legal command character.	Check the command character in the address field.
82	Checksum Error	The checksum received by Optomux did not match the sum of the characters in the command.	<ul style="list-style-type: none"> ▪ Check the cable wiring. ▪ Check the station number. ▪ Check the RTS/CTS configuration (see Studio Technical Reference Manual for valid configurations).
93	Input Buffer Overrun	The received command contained more than 71 characters for analog or 16 characters for digital boards. The command was ignored.	Check the command typed in the address fields.
104	Non-printable ASCII Character Received - Command Ignored	Only characters from 21 hex to 7F hex are permitted within commands.	Check characters used in the fields.
115	Data Field Error	Not enough characters received.	Check the number of characters used in the fields.
126	Communication Link Watchdog Time-out Error	Communication Time-out	<ul style="list-style-type: none"> ▪ Check the cable wiring. ▪ Check the station number.
137	Specified Limits Invalid	Invalid Limits specified in the parameters	Check the limits specified in the application.
-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. Review the Communication Parameters section for valid RTS/CTS configurations.

Error Code	Description	Possible Causes	Procedure to Solve
-17	Timeout between rx characters.	<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 configuration. Review the Communication Parameters section for valid RTS/CTS configurations.

⇒ **Tip:**

You can monitor communication status by establishing an event log in Studio's *Output* window (*LogWin* module). To establish a log for **Field Read Commands**, **Field Write Commands** and **Serial Communication**, right-click in the *Output* window and select the desired options from the pop-up menu.

You can also use the *LogWin* module (**Tools** → **LogWin**) to establish an event log on a remote unit that runs Windows CE. The log is saved on the unit in the `ce1log.txt` file, which can be downloaded later.

If you are unable to establish communication between Studio and the target device, then try instead to establish communication using the device's own programming software. Quite often, communication is interrupted by a hardware or cable problem or by a device configuration error. If you can successfully communicate using the programming software, then recheck the driver's communication settings in Studio.

To test communication between Studio and the device, 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.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of Changes
A	1.01	Roberto V. Junior	Jun-1998	Include new commands(digital)
B	1.02	Lidiane Amorim Moreira	Jan-2003	Conformance to CE
C	1.03	Graziane C. Forti	Nov-2006	Included Read Analog Input and Output, Write Analog Output command
D	1.03	Graziane C. Forti	Jun-2007	Edited for language and usability.