

Elipse FTP Client Driver

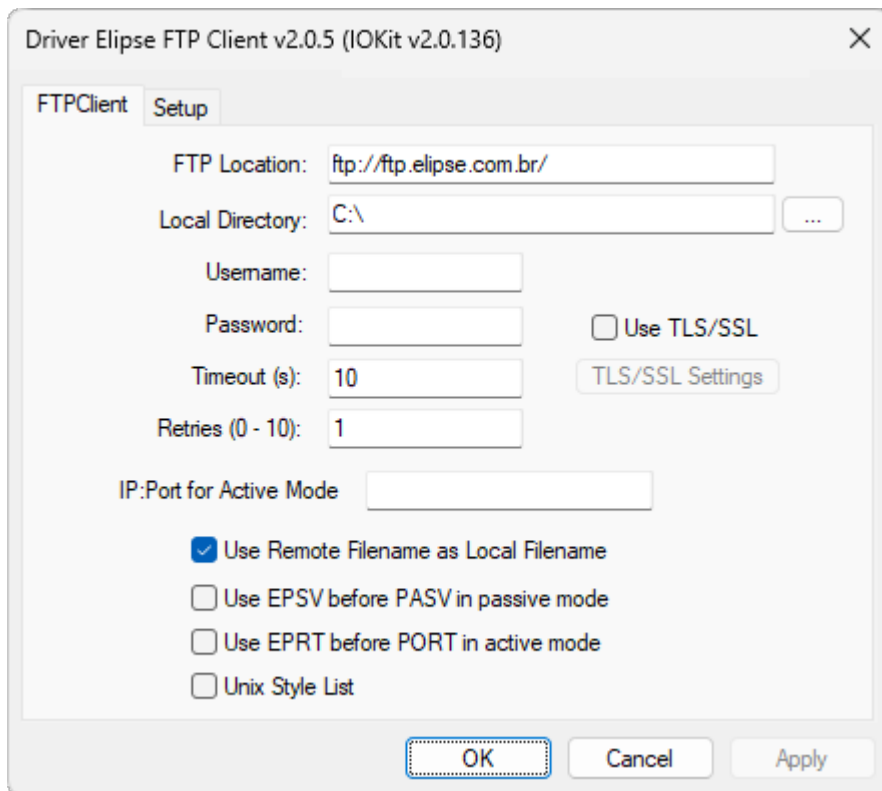
File Name	FTPClient.dll
Manufacturer	Elipse Software
Devices	
Protocol	FTP (File Transfer Protocol)
Version	2.0.8
Last Update	09/02/2025
Platform	Win32
Dependencies	IOKit version 2.0 or later
Superblock Readings	No
Level	0

Introduction

The Elipse FTP Client Driver can be used to list directories and retrieve files from FTP servers, saving them on a directory on the local computer.

Driver Configuration

Some general configurations must be informed on this Driver's configuration window, on the **FTPClient** tab, as displayed on the next figure.



FTPClient tab

The available options on this tab are described on the next table.

Available options on the FTPClient tab

OPTION	DESCRIPTION
FTP Location	Inform an IP address or path of an FTP server, such as "ftp://ftp.elipse.com.br/"
Local Directory	Local directory to where files must be copied, such as "C:\Mydir\"
Username	Inform the name of a user from an FTP server. For an anonymous connection, leave this option blank
Password	Inform a password for a user on an FTP server. For an anonymous connection, leave this option blank
Use TLS/SSL	If this option is selected, this Driver forces the use of TLS/SSL (<i>Transport Layer Security</i> or <i>Secure Sockets Layer</i> cryptography) security protocol and enables certificate authentication. When this option is not selected, this Driver tries to use TLS/SSL protocol without certificate authentication and, if it fails, then tries again without TLS/SSL protocol. To perform specific settings, use the TLS/SSL Settings option
Timeout (s)	Time to wait, in seconds, for a response from an individual command. After this time passes, that operation is aborted
Retries	How many times a transaction with a server is reattempted in case it fails. After all retries are exhausted, this Driver assumes a failed connection to a server
TLS/SSL Settings	Use this option for specific settings of the TLS/SSL protocol, described on topic TLS/SSL Settings Window
IP:Port for Active Mode	The FTP protocol provides Active and Passive data transfer modes. The default is the Passive mode, where a client requests a transfer and data arrives using the same channel. In the Active mode, a client informs an IP address and a TCP/IP port from where a server connects, and then sends those requests back. If users want to use the Active mode, inform in this option a value different from empty, as described on table Available options for the IP:Port for the Active Mode option
Use Remote Filename as Local Filename	If this option is selected and the name of the local file is not informed, this Driver tries to use the same file name requested as the local name
Use EPSV before PASV in passive mode	In Passive mode, this Driver tries an EPSV command before a PASV command. The EPSV command is not supported by all servers
Use EPRT before PORT in active mode	In Active mode, this Driver tries an EPRT command before a PORT command. The EPRT command is not supported by all servers
Unix Style List	This Driver, when receiving a list of files from a directory, must extract information about file name and size, for example. There are differences among FTP servers and some servers use a format more common to Windows applications, the NT format, and others use the Unix format. Select this option if the format is Unix . Please

OPTION	DESCRIPTION
	check items NT Format and Unix Format for examples of these formats

Available options for the IP:Port for Active Mode option

SYNTAX	EXAMPLE	MEANING
IP:Port	192.168.102.23:4567	Informs that a server must connect to the IP address 192.168.102.23 using TCP/IP port 4567
IP:Range	192.168.102.23:4500-4600	Informs a range of available TCP/IP ports
IP	192.168.102.23	A server tries to connect using a random TCP/IP port
-	- (dash)	Uses the current IP address and a random TCP/IP port

NT Format

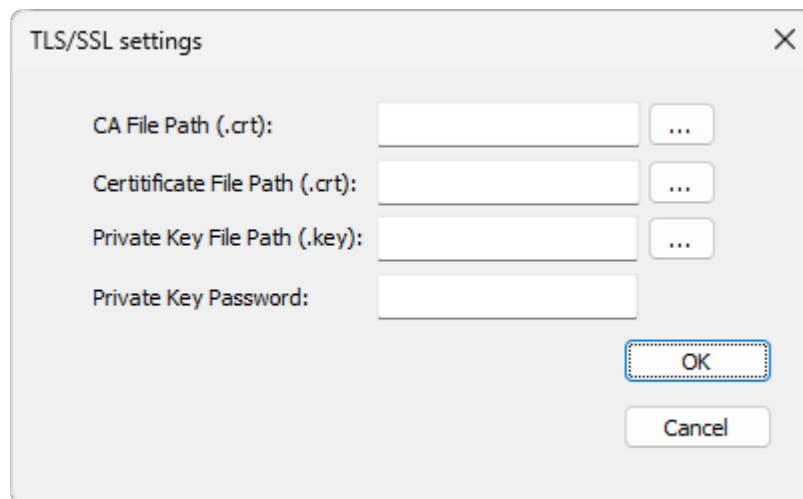
```
06-03-99 10:57PM      1 NotCurrentYear.txt
03-06-00 09:11PM      1 CurrentYear.txt
03-06-00 09:18PM      <DIR> tmp
```

Unix Format

```
-rw-r--r--  1 vglass  vglass   3651 Feb  2  2002 .screenrc
drwxr-xr-x  3 vglass  vglass   4096 Jul 23 17:42 Archive
-rw-r--r--  1 vglass  vglass   1096 Jul 23 13:35 GZipTest1.class
-rw-r--r--  1 vglass  vglass    733 Jul 23 13:35 GZipTest1.java
```

TLS/SSL Settings Window

When clicking **TLS/SSL Settings** on the **FTPClient** tab, the window on the next figure is opened.



TLS/SSL settings window

The available options on this window are described on the next table.

Available options on the TLS/SSL settings window

OPTION	DESCRIPTION
CA File Path (.crt)	Name and path of a file for the Certificate of Authority (CA). Click <input type="text" value="..."/> to select a file
Certificate File Path (.crt)	Name and path of a file for a client's SSL Certificate. Click <input type="text" value="..."/> to select a file
Private Key File Path (.key)	Name and path of a file with private keys for a client's TLS and SSL Certificates. Click <input type="text" value="..."/> to select a file
Private Key Password	Password for the private key, if applicable

All settings on this window are optional and can be left blank if not applicable.

Tag Reference

This section contains information about the configuration of this Driver's **[N/B]** Tags.

Abort Read

Write-Only

N1	9 (nine)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Write any value to this Tag to cancel an ongoing reading operation.

Abort Write

Write-Only

N1	11
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Write any value to this Tag to cancel an ongoing writing operation.

Copy Directory

Write-Only

N1	13
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Copies a directory and all its files, without sub-directories, from an FTP server to the local folder (value zero) or from the local folder to an FTP server (value one). The name of the directory to copy must be informed in the **Remote Folder Name** Tag, regardless of the operation's direction.

Users can copy all files from a server's base directory. To do so, inform in the **Remote Folder Name** Tag only the value "/".

Create or Delete Directory

Write-Only

N1	12
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Creates (value zero) or deletes (value one) a directory in an FTP server, which must be specified in the **Remote Folder Name** Tag.

Users can delete the whole content of a server on a single operation. To do so, inform in the **Remote Folder Name** Tag only the value "/". The target directory of this operation is the base directory of a server.

Directory List

Read-Only

B1	7 (seven)
B2	0 (zero)
B3	0 (zero)
B4	0 (zero)

This must be a Block Tag with four Elements, which returns the response from a command to list directories. The Elements of this Block Tag are the following:

- **Element 0:** Time
- **Element 1:** 0 (zero) if the item is a file and 1 (one) if the item is a directory
- **Element 2:** Size, if available
- **Element 3:** Item's name

Last Error

Read-Only

N1	1 (one)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Informes the last transaction's error code as a number, according to the following values:

- **0:** OK
- **1:** UNSUPPORTED_PROTOCOL
- **2:** FAILED_INIT
- **3:** URL_MALFORMAT
- **5:** COULDNT_RESOLVE_PROXY
- **6:** COULDNT_RESOLVE_HOST
- **7:** COULDNT_CONNECT
- **8:** FTP_WEIRD_SERVER_REPLY
- **9:** REMOTE_ACCESS_DENIED (does not mean a login failure)
- **11:** FTP_WEIRD_PASS_REPLY
- **13:** FTP_WEIRD_PASV_REPLY
- **14:** FTP_WEIRD_227_FORMAT
- **15:** FTP_CANT_GET_HOST
- **17:** FTP_COULDNT_SET_TYPE
- **18:** PARTIAL_FILE
- **19:** FTP_COULDNT_RETR_FILE

- **21:** QUOTE_ERROR (failure when using a **QUOTE** command)
- **22:** HTTP_RETURNED_ERROR
- **23:** WRITE_ERROR
- **25:** UPLOAD_FAILED (failure when using an **UPLOAD** command)
- **26:** READ_ERROR (cannot open or read a file)
- **27:** OUT_OF_MEMORY
- **28:** OPERATION_TIMEDOUT (time-out reached)
- **30:** FTP_PORT_FAILED (failure when using an **FTP PORT** operation)
- **31:** FTP_COULDNT_USE_REST (failure when using a **REST** command)
- **33:** RANGE_ERROR (failure when using a **RANGE** command)
- **34:** HTTP_POST_ERROR
- **35:** SSL_CONNECT_ERROR (error when connecting using SSL)
- **36:** BAD_DOWNLOAD_RESUME (cannot restart a download)
- **37:** FILE_COULDNT_READ_FILE
- **38:** LDAP_CANNOT_BIND
- **39:** LDAP_SEARCH_FAILED
- **41:** FUNCTION_NOT_FOUND
- **42:** ABORTED_BY_CALLBACK
- **43:** BAD_FUNCTION_ARGUMENT
- **45:** INTERFACE_FAILED (failure in **CURLOPT_INTERFACE**)
- **47:** TOO_MANY_REDIRECTS (error in endless redirection loops)
- **48:** UNKNOWN_TELNET_OPTION (user specified an unknown option)
- **49:** TELNET_OPTION_SYNTAX (malformed **TELNET** option)
- **51:** PEER_FAILED_VERIFICATION (invalid certificate)
- **52:** GOT_NOTHING
- **53:** SSL_ENGINE_NOTFOUND (SSL engine not found)
- **54:** SSL_ENGINE_SETFAILED (SSL engine cannot be configured as default)
- **55:** SEND_ERROR (failure when sending network data)
- **56:** RECV_ERROR (failure when receiving network data)
- **57:** OBSOLETE57 (not used)
- **58:** SSL_CERTPROBLEM (problem with local certificate)
- **59:** SSL_CIPHER (cannot use specified cipher)

- **60:** SSL_CACERT (problem with CA certificate)
- **61:** BAD_CONTENT_ENCODING (unrecognized transfer encoding)
- **62:** LDAP_INVALID_URL (invalid uRL for LDAP protocol)
- **63:** FILESIZE_EXCEEDED (maximum file size exceeded)
- **64:** USE_SSL_FAILED (failure on requested FTP SSL level)
- **65:** SEND_FAIL_REWIND (sending data requires a rewind operation that failed)
- **66:** SSL_ENGINE_INITFAILED (failure when initializing SSL engine)
- **67:** LOGIN_DENIED (user and password not accepted and login failed)
- **68:** TFTP_NOTFOUND (file not found on the server)
- **69:** TFTP_PERM (permission problem on the server)
- **70:** REMOTE_DISK_FULL (disk is full on the server)
- **71:** TFTP_ILLEGAL (illegal **TFTP** operation)
- **72:** TFTP_UNKNOWNID (unknown transfer ID)
- **73:** REMOTE_FILE_EXISTS (file already exists)
- **74:** TFTP_NOSUCHUSER (user not found)
- **75:** CONV_FAILED (failure during conversion)
- **76:** CONV_REQD (user must register conversion callbacks using the **curl_easy_setopt** options **CURLOPT_CONV_FROM_NETWORK_FUNCTION**, **CURLOPT_CONV_TO_NETWORK_FUNCTION**, and **CURLOPT_CONV_FROM_UTF8_FUNCTION**)
- **77:** SSL_CACERT_BADFILE (cannot load CACERT file, this file does not exist, or its format is wrong)
- **78:** REMOTE_FILE_NOT_FOUND (remote file not found)
- **79:** SSH (error on SSH layer)
- **80:** SSL_SHUTDOWN_FAILED (failure when finishing an SSL connection)
- **81:** AGAIN (socket is not ready for **SEND** or **RECV** commands, please wait and try again)
- **82:** SSL_CRL_BADFILE
- **83:** SSL_ISSUER_ERROR

Last Error String

Read-Only

N1	2 (two)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Informes the last transaction's error code as a text, according to the codes listed in the **Last Error** Tag.

Local File Name

Write-Only

N1	4 (four)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Allows informing a local file name to save to the default local directory.

Recursive Copy Directory

Write-Only

N1	14
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Recursive version of the **Copy Directory** Tag. Copies a directory and its **whole** content, including sub-directories, from an FTP server to the local folder (value zero) or from the local folder to an FTP server (value one). The name of the directory to copy must be informed in the **Remote Folder Name** Tag, regardless of the operation's direction.

Users can copy the whole content of a server on a single operation. To do so, inform in the **Remote Folder Name** Tag only the value "/". The target directory of this operation is the base directory of a server.

Remote File Name

Write-Only

N1	3 (three)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Allows informing a remote file name for an operation. This Tag allows informing a wildcard character of an asterisk (*), such as "*.pdf" to indicate all files in **PDF** format.

NOTES

- If users inform in the **Request Read File** Tag a value of 2 (two) in its **Value** property, at the end of the transfer a removal operation is performed on all files that match the value informed in this Tag. However, not all FTP servers accept the asterisk (*) character in removal operations.
- When using a wildcard character, the name of each local file is identical to the name of each remote file, thus the configuration performed in the **Local File Name** Tag is disregarded.

Remote Folder Name

Write-Only

N1	8 (eight)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Allows informing the name of a remote directory for an operation.

Request Directory

Write-Only

N1	6 (six)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Requests a directory list using the name informed in the **Remote Folder Name** Tag. Any value can be written to this Tag.

Request Read File

Write-Only

N1	5 (five)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Requests a file transfer from a remote directory to the local directory. The remote file is the file informed in the **Remote File Name** Tag, located inside the directory informed in the **Remote Folder Name** Tag. The resulting local file is the file informed in the **Local File Name** Tag.

Write 2 (two) to this Tag's **Value** property so that, at the end of the file transfer, the file is deleted from the FTP server. Any other value in the **Value** property only transfers a file, without deleting it from the FTP server.

Request Write File

Write-Only

N1	10
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Requests a file transfer from the local directory to a remote directory. The resulting remote file is the file informed in the **Remote File Name** Tag, located inside the directory informed on the **Remote Folder Name** Tag. The local file is the file informed on the **Local File Name** Tag. Any value can be written to this Tag.

Status

Read-Only

N1	0 (zero)
N2	0 (zero)
N3	0 (zero)
N4	0 (zero)

Informs the current status of all requested transactions, according to the following values:

- 0: FTP_STATUS_IDLE
- 1: FTP_STATUS_TRANSFERRING
- 2: FTP_STATUS_FINISH_OK
- 3: FTP_STATUS_FINISH_ERROR
- 4: FTP_STATUS_CANCELLED
- 5: FTP_STATUS_DIR_READY

Configuration Parameters via IOKit Library

To configure this Driver using commands from **IOKit** library, inform the parameters on the next table on the configuration dialog box.

Configuration parameters of this Driver for IOKit library

OPTION	PARAMETER	DATA TYPE
FTP Location	FTPClient.FTPLocation	Text
Local Directory	FTPClient.LocalDir	Text
Username	FTPClient.UserName	Text
Password	FTPClient.Password	Text
Use Remote Filename As Local Filename	FTPClient.UseRemoteAsLocal	Byte
Timeout (s)	FTPClient.Timeout	DWord
IP:Port for Active Mode	FTPClient.IPPortActive	Text
Use EPRT before PORT in active Mode	FTPClient.EPRT	Byte
Use EPSV before PASV in passive Mode	FTPClient.EPSV	Byte
Unix Style List	FTPClient.UnixStyle	Byte

Documentation of I/O Interfaces

This section contains the documentation about I/O Interfaces referring to **FTPClient** Driver.

General Configurations

This section contains information about the configuration of general **I/O Tags** and **Properties** of I/O Interfaces.

I/O Tags

General I/O Interfaces Tags (N2/B2 = 0)

The Tags described next are provided for all supported I/O Interfaces.

IO.CommunicationStatus

Type of Tag	I/O Tag
Type of Access	Reading
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	6 (six)
String Configuration	IO.CommunicationStatus

This Tag informs the communication status of a Driver. It indicates how communication works relative to receiving valid data within a time period arbitrated in the configuration. For more information, please check topic **Setup Tab**. Possible values are **0 - Inactive communication**: The Driver did not receive valid data or stopped receiving data after *n* milliseconds, as configured in the properties window, or **1 - Active communication**: The Driver is receiving valid data.

IO.IOKitEvent

Type of Tag	Block Tag
Type of Access	Read-Only
B1 Parameter	-1 (minus one)
B2 Parameter	0 (zero)
B3 Parameter	0 (zero)
B4 Parameter	1 (one)
Size Property	4 (four)
ParamItem Property	IO.IOKitEvent

This Block returns Driver events generated by several sources in I/O Interfaces. The **TimeStamp** property of this Block represents the moment this event occurred. The Block Elements are the following:

- **Element 0**: Type of event. Possible values are **0**: Information, **1**: Warning, or **2**: Error
- **Element 1**: Source of an event. Possible values are **0**: Driver (specific of a Driver), **-1**: IOKit (generic events of I/O Interfaces), **-2**: **Serial** Interface, **-3**: **Modem** Interface, **-4**: **Ethernet** Interface, or **-5**: **RAS** Interface
- **Element 2**: Error number, specific for each source of event
- **Element 3**: Message of an event, a **String** specific for each event

NOTE

A Driver keeps a maximum number of 100 events internally. If additional events are reported, older events are discarded.

IO.PhysicalLayerStatus

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	2 (two)
String Configuration	IO.PhysicalLayerStatus

This Tag indicates the status of a physical layer. Possible values are the following:

- **0:** Physical layer stopped, that is, a Driver is in **Offline** mode, the physical layer failed when initializing, or exceeded the maximum number of reconnection attempts
- **1:** Physical layer started but not connected, that is, a Driver is in **Online** mode but the physical layer is not connected. If the **Connection management** option is configured with the value **Automatic**, the physical layer can be connecting, disconnecting, or waiting for a reconnection attempt. If the **Connection management** option is configured with the value **Manual**, then the physical layer remains in this status until forced to connect
- **2:** Physical layer connected, that is, the physical layer is ready for use. This **DOES NOT** mean a device is connected, only that the access layer is working

IO.SetConfigurationParameters

Type of Tag	Block Tag
Type of Access	Read-Only
B1 Parameter	-1 (minus one)
B2 Parameter	0 (zero)
B3 Parameter	0 (zero)
B4 Parameter	3 (three)
Size Property	2 (two)
ParamItem Property	IO.SetConfigurationParameters

Use this Tag to change any property of a Driver's configuration dialog box at run time.

This Tag works only while a Driver is in **Offline** mode. To start a Driver in **Offline** mode, select the **Start driver OFFLINE** option on that Driver's configuration dialog box. Users can write to a PLC Tag or to a Block Tag containing the parameters to change. Writing individual Block Elements is not supported, the whole Block must be written at once.

In **Eclipse SCADA**, users must use a Block Tag. Every parameter to configure uses two Block Elements. For example, if users want to configure 3 (three) parameters, then the size of the Block must be 6 (six, 3×2). The first Element is the property's name, as a **String**, and the second Element is the property's value, according to the next example.

```
// 'Block' must be a Block Tag with automatic reading,
// scan reading, and automatic writings disabled.
// Configure all parameters
Block.element001 = "IO.Type" // Parameter 1
Block.element002 = "Serial"
Block.element003 = "IO.Serial.Port" // Parameter 2
Block.element004 = 1
Block.element005 = "IO.Serial.BaudRate" // Parameter 3
Block.element006 = 19200
// Writes the whole Block
Block.Write()
```

When using **Elipse E3**, the ability to create arrays at run time allows using an I/O Tag as well as a Block Tag. Users can use the **Write** method of a Driver to send the parameters directly to that Driver, without creating a Tag, according to the next example.

```
Dim arr(6)
' Configure all array elements
arr(1) = "IO.Type"
arr(2) = "Serial"
arr(3) = "IO.Serial.Port"
arr(4) = 1
arr(5) = "IO.Serial.BaudRate"
arr(6) = 19200
' There are two methods to send parameters
' Method 1: Using an I/O Tag
tag.WriteEx arr
' Method 2: Without using a Tag
Driver.Write -1, 0, 0, 3, arr
```

A variation of the previous example uses a bidimensional array.

```
Dim arr(10)
' Configure all array elements. Notice the array was resized
' to 10 elements. Empty array elements are ignored by a Driver
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.Serial.BaudRate", 19200)
Driver.Write -1, 0, 0, 3, arr
```

A Driver does not validate parameter names or passed values, therefore be careful when writing parameters and values. The **Write** method fails if the configuration array is incorrectly created. Users can check the log of a Driver or use the *writeStatus* parameter of the **WriteEx** method to find out the exact cause of an error.

```
Dim arr(10), strError
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.Serial.BaudRate", 19200)
If Not Driver.WriteEx -1, 0, 0, 3, arr, , , strError Then
    MsgBox "Failed configuring Driver parameters: " + strError
End If
```

IO.WorkOnline

Type of Tag	I/O Tag
Type of Access	Reading or Writing
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	4 (four)
String Configuration	IO.WorkOnline

This Tag informs the current status of a Driver and allows starting or stopping the physical layer. Possible values are the following:

- **0 - Driver Offline:** Physical layer is closed or stopped. This mode allows a dynamic configuration of a Driver's parameters using the **IO.SetConfigurationParameters** Tag
- **1 - Driver Online:** Physical layer is open or executing. While in **Online** mode, the physical layer can be connected or disconnected and its current status can be checked using the **IO.PhysicalLayerStatus** Tag

In the next example, using **Elipse E3**, a Driver is configured to **Offline** mode, its COM port is changed, and then configured to **Online** mode again.

```
'Configure to Offline mode
Driver.Write -1, 0, 0, 4, 0
'Change port to COM2
Driver.Write -1, 0, 0, 3, Array("IO.Serial.Port", 2)
'Configure to Online mode
Driver.Write -1, 0, 0, 4, 1
```

The **Write** method may fail when configuring a Driver to **Online** mode, that is, writing the value 1 (one). In this case, this Driver remains in **Offline** mode. The cause of failure can be:

- Type of physical layer incorrectly configured, probably an invalid value was configured in the **IO.Type** property
- This Driver may have run out of memory
- Physical layer probably did not create its working thread. Search the log file for a message "Failed to create physical layer thread!"
- Physical layer could not start. The cause of this failure depends on the type of physical layer. It can be an invalid serial port number, a failure when starting Windows Sockets, or a failure when starting TAPI (modem), among others. This cause is recorded on the log file

IMPORTANT

Even if the configuration of a Driver to **Online** mode is successful, this does not necessarily mean the physical layer is ready to use, that is, ready to execute input and output operations with an external device. The **IO.PhysicalLayerStatus** Tag must be checked to ensure the physical layer is connected and ready for communication.

Properties

These are general properties of all supported I/O Interfaces.

IO.ConnectionMode

9 Controls the management mode of a Connection. Possible values are **0**: Automatic mode, in which a Driver manages the connection or **1**: Manual mode, in which an application manages the connection.

IO.GiveUpEnable

When configured to True, defines a maximum number of reconnection attempts. If all reconnection attempts fail, a Driver enters the **Offline** mode. When configured to False, a Driver tries until a reconnection is successful.

IO.GiveUpTries

9 Number of reconnection attempts before this one is aborted. For example, if the value of this property is equal to 1 (one), a Driver tries only one reconnection when the connection is lost. If this one fails, this Driver enters the **Offline** mode.

IO.InactivityEnable

☑ Configure to True to enable and to False to disable inactivity detection. The physical layer is disconnected if inactive for a certain period of time. The physical layer is considered inactive only if it is capable of sending data but not capable of receiving it back.

IO.InactivityPeriodSec

9 Number of seconds to check for inactivity. If the physical layer is inactive for this period of time, it is then disconnected.

IO.RecoverEnable

☑ Configure to True to enable a Driver to recover lost connections and to False to leave a Driver in **Offline** mode when a connection is lost.

IO.RecoverPeriodSec

9 Delay time between two connection attempts, in seconds.

NOTE

The first reconnection is executed immediately after a connection is lost.

IO.StartOffline

☑ Configure to True to start a Driver in **Offline** mode and to False to start a Driver in **Online** mode.

NOTE

It is pointless to change this property at run time, as it can only be changed when a Driver is already in **Offline** mode. To configure a Driver in **Online** mode at run time, write the value 1 (one) to the **IO.WorkOnline** Tag.

IO.TimeoutMs

9 Defines a time-out for the physical layer, in milliseconds. One second is equal to 1000 milliseconds.

IO.Type

A Defines the type of physical interface used by a Driver. Possible values are the following:

- **N or None:** Does not use a physical interface, that is, a Driver must provide a customized interface
- **S or Serial:** Uses a local serial port (COM n)
- **M or Modem:** Uses a local modem, internal or external, accessed via TAPI (*Telephony Application Programming Interface*)
- **E or Ethernet:** Uses a TCP/IP or UDP/IP socket

- **R or RAS:** Uses a **RAS** (*Remote Access Server*) Interface. A Driver connects to a RAS device using the **Ethernet** Interface and then sends an **AT** (*dial*) command

Configuration of a Driver

I/O Interface configuration is performed on a Driver's configuration dialog box. To access the configuration of this dialog box in **Eclipse E3** in version 1.0, follow these steps:

1. Right-click a Driver object (IODriver).
2. Select the **Properties** item on the contextual menu.
3. Select the **Driver** tab.
4. Click **Other parameters**.

In **Eclipse E3** version 2.0 or later, click **Configure driver**  on a Driver's toolbar. In **Eclipse SCADA**, follow these steps:

1. Open the Organizer.
2. Select a Driver on Organizer's tree.
3. Click **Extras** on the **Driver** tab.

Currently, an I/O Interface allows opening only one connection for each Driver. This means that, if users want to access two serial ports, they must add two Drivers to an application and then configure each one of these Drivers for each serial port.

Configuration Dialog Box

The dialog box of I/O Interfaces allows configuring the I/O connection used by a Driver. This dialog box contains the **Setup**, **Serial**, **Ethernet**, **Modem**, and **RAS** tabs, described on the next topics. If a Driver does not implement a specific I/O connection, its corresponding tab is not available for configuration. Some Drivers may contain additional tabs, specific for that Driver, on the configuration dialog box.

Setup Tab

The **Setup** tab contains general configurations of a Driver. This tab is divided into the following groups:

- **General configurations:** Configurations of a Driver's physical layer, time-out, and initialization mode
- **Connection management:** Configurations on how the I/O Interface keeps a connection and which recovery policy is used on failure
- **Logging options:** Controls the generation of log files

Setup

Physical Layer: Ethernet Start driver OFFLINE

Timeout: 1000 ms Communication check time: 5000 ms

Connection management

Mode: Automatic (managed by the driver)

Retry failed connection every 20 seconds

Give up after 1 failed retries

Disconnect if non-responsive for 0 seconds

Logging Options

Log to File: C:\eeLogs\MicrolokII_%DATE%.log

File size limit (MB): 0 ('0' is unlimited)

Setup tab

General options on the Setup tab

OPTION	DESCRIPTION
Physical Layer	Select the physical layer on a list. Available options are Serial , Ethernet , Modem , and RAS . The selected interface must be configured on its specific tab
Timeout	Configure a time-out, in milliseconds, for the physical layer. This is the amount of time an I/O interface waits to receive any byte from the reception's buffer
Communication check time	Set the time, in milliseconds, to define the interval at which communication is considered to be in an inactive state. As long as an I/O Driver receives valid data, its communication state is considered active. However, if during operation an I/O Driver does not receive valid data inside this period of time, the state is considered inactive. The communication state is shown in the IO.CommunicationStatus Tag
Start driver OFFLINE	Select this option so that a Driver starts in Offline mode or stopped. This means that the I/O interface is not created until this Driver is configured to Online mode by using a Tag in an application. This mode enables a dynamic configuration of an I/O interface at run time

Options on the Connection management group

OPTION	DESCRIPTION
Mode	Selects a management mode of a connection. Selecting the Automatic option allows a Driver to manage the connection automatically, as specified in the next options. Selecting the Manual option allows an application to fully manage a connection
Retry failed connection every ... seconds	Select this option to enable a Driver's connection retry in a certain interval, in seconds. If the Give up after failed retries option is not selected, this Driver keeps retrying until a connection is performed, or until the application is stopped
Give up after ... failed retries	Enable this option to define a maximum number of connection retries. When the specified number of consecutive connection retries is reached, a Driver goes to the Offline mode, assuming that a hardware problem was detected. If a Driver establishes a successful connection, the number of unsuccessful retries is cleared. If this new connection is lost, then the retry counter starts at zero
Disconnect if non-responsive for ... seconds	Enable this option to force a Driver to disconnect if no byte was received by the I/O interface during the specified time-out, in seconds. This time-out must be greater than the time-out configured in the Timeout option

Options on the Logging Options group

OPTION	DESCRIPTION
<p>Log to File</p>	<p>Enable this option and configure the name of a file to write a log. Log files can be large, so use this option for short periods of time, only for testing and debugging purposes. If the %PROCESS% macro is used in the log file name, it is replaced by the identifier of the current process. This option is particularly useful when using several instances of the same Driver in Elipse E3, thus allowing each instance to generate a separate log file. For example, when configuring this option with value "c:\e3logs\drivers\sim_%PROCESS%.log", it generates a file named c:\e3logs\drivers\sim_00000FDA.log for process OFDAh. Users can also use the %DATE% macro in the file name. In this case a log file is generated every day, in the format aaaa_mm_dd. For example, when configuring this option with value "c:\e3logs\drivers\sim_%DATE%.log", it generates a file named c:\e3logs\drivers\sim_2005_12_31.log in 12/31/2005 and a file named c:\e3logs\drivers\sim_2006_01_01.log in 01/01/2006. Similarly, the %DATE_HOUR% macro generates one log file per hour, in the format aaaa_mm_dd_hh</p>
<p>File size limit (MB)</p>	<p>Configure the log file size limit, in megabytes. A value equal to 0 (zero) means that there is no size limit for the log file</p>

Statistical Configuration

This section contains information about the configuration of **I/O Tags** and **Properties** of I/O Interfaces statistics.

I/O Tags

Tags of I/O Interface Statistics (N2/B2 = 0)

The Tags described next display statistics for all I/O Interfaces.

IO.Stats.Partial.BytesRecv

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1101
Configuration by String	IO.Stats.Partial.BytesRecv

This Tag returns the number of bytes received in the current connection.

IO.Stats.Partial.BytesSent

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1100
Configuration by String	IO.Stats.Partial.BytesSent

This Tag returns the number of bytes sent through the current connection.

IO.Stats.Partial.TimeConnectedSeconds

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1102
Configuration by String	IO.Stats.Partial.TimeConnectedSeconds

This Tag returns the number of seconds a Driver is connected in the current connection or 0 (zero) if a Driver is disconnected.

IO.Stats.Partial.TimeDisconnectedSeconds

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1103
Configuration by String	IO.Stats.Partial.TimeDisconnectedSeconds

This Tag returns the number of seconds a Driver is disconnected since the last connection ended or 0 (zero) if a Driver is connected.

IO.Stats.Total.BytesRecv

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1001
Configuration by String	IO.Stats.Total.BytesRecv

This Tag returns the number of bytes received since a Driver was loaded.

IO.Stats.Total.BytesSent

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1000
Configuration by String	IO.Stats.Total.BytesSent

This Tag returns the number of bytes sent since a Driver was loaded.

IO.Stats.Total.ConnectionCount

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1004
Configuration by String	IO.Stats.Total.ConnectionCount

This Tag returns the number of connections a Driver already established, successfully, since it was loaded.

IO.Stats.Total.TimeConnectedSeconds

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1002
Configuration by String	IO.Stats.Total.TimeConnectedSeconds

This Tag returns the number of seconds a Driver remained connected since it was loaded.

IO.Stats.Total.TimeDisconnectedSeconds

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	0 (zero)
N4 Parameter	1003
Configuration by String	IO.Stats.Total.TimeDisconnectedSeconds

This Tag returns the number of seconds a Driver remained disconnected since it was loaded.

Properties

Currently, there are no properties defined specifically to display I/O Interface statistics at run time.

Driver Revision History

VERSION	DATE	AUTHOR	COMMENTS
2.0.8	09/02/2025	M. Ludwig	<ul style="list-style-type: none"> Driver updated to IOKit library version 3.0 and Visual Studio 2022 (Case 37959).
2.0.7	10/16/2024	A. Fetzner	<ul style="list-style-type: none"> Removed from log the FTPClient.Password and IO.Ethernet.SSLKeyPassword parameters, for security reasons (Case 36760).
2.0.6	08/16/2023	Pedro H. Santos	<ul style="list-style-type: none"> Implemented operations for creating, deleting, and copying a directory in an FTP server (Case 26213).

VERSION	DATE	AUTHOR	COMMENTS
2.0.5	12/22/2021	H. Coelho	<ul style="list-style-type: none"> Implemented an operation for uploading local files to a remote server (<i>Case 25880</i>). Implemented the automatic addition of slash (/) or backslash (\) characters at the end of a user-written path, when necessary (<i>Case 31862</i>). Fixed the implementation of SSL/TLS cryptography (<i>Case 31906</i>).
2.0.4	12/14/2020	M. Ludwig	<ul style="list-style-type: none"> Implemented settings for SSL/TLS cryptography (<i>Case 25903</i>).
2.0.3	12/28/2016	C. Mello	<ul style="list-style-type: none"> Platform update on this Driver's source code (<i>Case 27512</i>).
2.0.2	12/02/2016	M. Salvador	<ul style="list-style-type: none"> Implemented a batch file transfer using a wildcard character (*) and implemented an option to delete a file after a transfer (<i>Case 21714</i>). Created a Remote Folder Name Tag, <i>N1</i> equal to 8 (eight), to inform a directory on an FTP server (<i>Case 21678</i>).
2.0.1	11/21/2016	F. Englert	<ul style="list-style-type: none"> Driver ported to IOKit library version 2.0 (<i>Case 21688</i>).
1.0.1	07/19/2012	M. Salvador	<ul style="list-style-type: none"> Initial version of this Driver.

Headquarters

**Rua Mostardeiro, 322/Cj. 902, 1001 e
1002**

90510-002 — Porto Alegre — RS

Phone: (+55 51) 3346-4699

Fax: (+55 51) 3222-6226

E-mail: elipse-rs@elipse.com.br

Branch in Taiwan

9F., No.12, Beiping 2nd St., Sanmin Dist.

807 — Kaohsiung City — Taiwan

Phone: (+886 7) 323-8468

Fax: (+886 7) 323-9656

E-mail: evan@elipse.com.br

Check our website for information about a representative in your country.

www.elipse.com.br

kb.elipse.com.br

forum.elipse.com.br

www.youtube.com/elipsesoftware

elipse@elipse.com.br



Gartner, Cool Vendors in Brazil 2014, April 2014.

Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability of fitness for a particular purpose.

Microsoft Partner
Gold Independent Software Vendor (ISV)