IEC 61850 Server Driver

File Name	IEC61850Srv.dll
Manufacturer	International Electrotechnical Commission
Devices	
Protocol	IEC 61850, MMS, and XMPP
Version	1.0.20
Last Update	01/26/2026
Platform	Win32 and Win64
Dependencies	IOKit version 2.0 or later
Superblock Readings	No
Level	31308

Introduction

The IEC 61850 Server Driver can communicate with other IEC 61850 applications or systems that implement a **Client** mode. To do so, this Driver can import a database in **ICD** format, which can be created by any edition tool for configuration files in IEC 61850 standard. This Driver allows the following actions:

- Communication with a single client per Driver
- Importing Tags from Drivers
- Support for Reports (Buffered or Unbuffered)
- Polling
- Support for quality and timestamp information with a precision of one millisecond

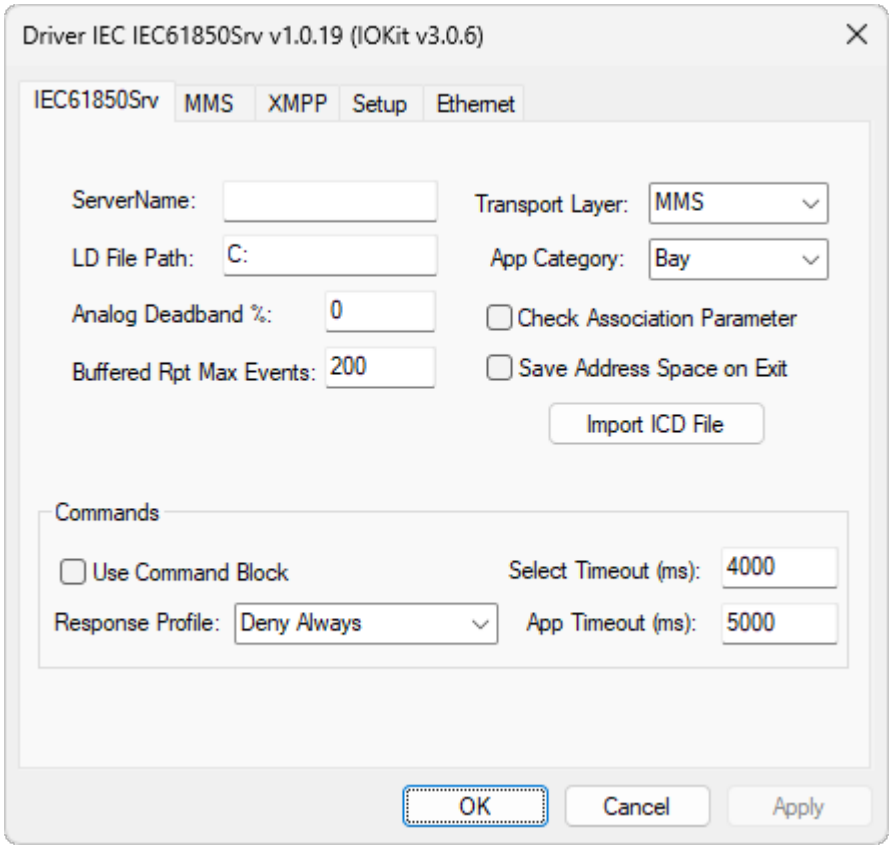
Additional information about how this Driver works can be checked on topic **Technical Specifications**.

Driver Configuration

This Driver's **[P]** parameters are not used. All settings are performed on this Driver's configuration window.

IEC61850Srv Tab

The **IEC61850Srv** tab configures this Driver's general behavior, according to the next figure.



IEC61850Srv tab

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

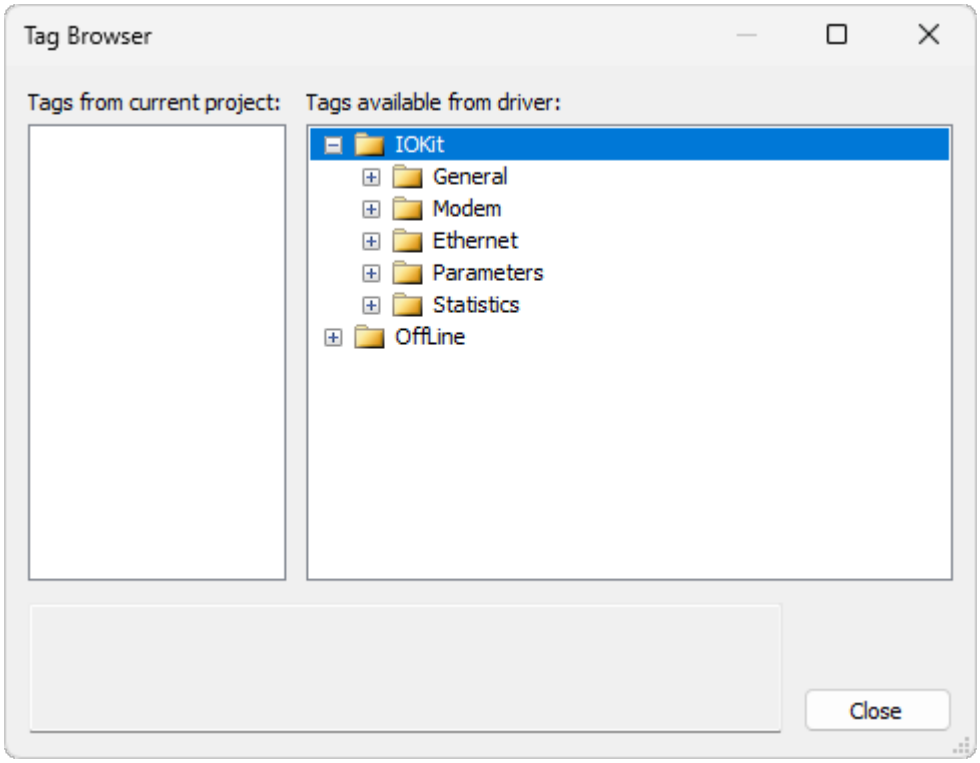
Available options on the IEC61850Srv tab

OPTION	DESCRIPTION
ServerName	Defines a default name for this Driver, used in the name of files in **LDS** format
Transport Layer	Defines a transport layer. Possible values are **MMS** (default) or **XMPP**
LD File Path	Directory where files in **LDS** format are saved containg an IEC 61850 database. Users must inform a valid directory
App Category	When receiving commands, this Driver uses the item selected in this option to compare with the **OrCat** (*Origin Category*) property sent by a client. The available options are **Bay**, **Station**, **Remote**, or **Maintenance**. Commands with unexpected types are not executed
Analog Deadband %	Indicates a minimum percentage for a change in the value of an analog variable, relative to its previous value, to be considered an event that can be sent by a report. Users can also specify an individual dead band in the *N1* parameter of each Tag
Check Association Parameter	Defines if, when receiving a connection request from a client, the client addresses **P** (Presentation), **T** (Transport), or **S** (Session) accept or not a connection
Buffered Report Max Events	Maximum number of events stored in **BRCBs** (buffered reports)

OPTION	DESCRIPTION
Save Address Space on Exit	Indicates whether a file in **LDS** format must be saved after finishing the execution of this Driver, updating default values that may have been changed relative to what was imported by a file in **ICD** format
Import ICD File	Allows selecting a file in **ICD** format to use as this Driver's database. All parameters that can be imported are copied to the available properties. The other parameters, such as definitions for Logical Devices and Logical Nodes, are saved to a proprietary file with an .lds extension on the path indicated in the **LD File Path** option and also a file for each Logical Device is created with the name in the format **ServerName_LogicalDeviceName.LDS**
Use Command Block	Informs whether the process of importing Tags, or Tag browsing, creates individual Tags for **Select** and **Operate** commands or a block is created allowing to inform an operation before the request
Select Timeout (ms)	Time-out between a **Select** operation and an **Operate** command. If after this time the **Operate** command is not sent, then the **Select** operation is canceled
Response Profile	Informs how commands are handled by this Driver. Possible values are **Deny Always**: Commands are responded immediately, denying them with a **Temporarily Unavailable** code, **Accept Always**: Commands are responded immediately, passing their value to an application, or **Treat by Application**: Commands are evaluated by an application. If positive, sends a success message to the client. Otherwise, sends a response with a **Temporarily Unavailable** code
App Timeout (ms)	Default time-out of an application layer

Tag Browser

Click **Tag Browser** on this Driver's toolbar in **Elipse E3**, **Elipse Power**, or **Elipse Water** to open the Tag Browser window, which allows listing and dragging to this Driver all existing Tags on the database, according to the next figure.



Tag Browser window

The available options on the **Tags available from driver** list are the following:

- **IOKit**: Contains default **IOKit** library Tags, allowing to read and write connection and status parameters, among others
- **Offline**: Displays a folder with the name of a server and, on this folder, a folder for each Logical Device found. Each Logical Device contains Logical Nodes and their respective objects and properties

To use these Tags in an application, drag a Tag or folder from the **Tags available from driver** list and drop it on the **Tags from current project** list.

About IEC 61850 Standard

The **IEC 61850** is an international standard for communication in electrical substation automation systems, widely used in smart grids. This standard defines a robust, flexible, and interoperable communication model for protection, control, monitoring, and measurement devices in power systems.

Key Features

- **Standardized Data Model**: Uses data objects, or Logical Nodes, to represent functions and devices, such as circuit breakers and transformers, ensuring consistency and interoperability
- **Ethernet-Based Communication**: Supports protocols **MMS** (*Manufacturing Message Specification*), **XMPP** (*Extensible Messaging and Presence Protocol*), **GOOSE** (*Generic Object-Oriented Substation Event*), and **SV** (*Sampled Values*) for real-time data exchange
- **SCL Configuration**: Uses the SCL (*Substation Configuration Language*) language to describe device and system configurations, simplifying engineering and maintenance
- **High Reliability and Scalability**: Designed to meet performance requirements in critical environments, with support for redundancy and efficient communication

Benefits

- Interoperability between devices from different manufacturers
- Reduced configuration and maintenance costs.
- Support for advanced applications, such as automation and real-time monitoring

The implementation of IEC 61850 standard in an application allows an efficient integration with compatible devices, ensuring reliable and adaptable operation in modern substations.

Transport Layer

IEC 61850 standard was originally developed for Ethernet-based communication using the **MMS** protocol, as specified in IEC 61850-8-1, focused on local substation networks. In 2018, IEC 61850-8-2 standard introduced an alternative based on the **XMPP** protocol, primarily focusing on communication outside a local network, expanding its applicability. Thus, this Driver allows users to select one of these protocols as the transport layer. This standard also defines an Ethernet-based data exchange known as **GOOSE**, which is not currently implemented in this Driver.

Structuring Information

Systems implementing this standard as data servers, typically IEDs or protection relays, organize information based on the following entities:

- **Logical Devices or LD**: Represent logical devices corresponding to a physical or functional device, such as a bay mapped to an IED (*Intelligent Electronic Device*). They correspond to the first hierarchical level in an IEC 61850 Server, grouping related logical functions to facilitate modeling and management
- **Logical Nodes or LN**: Virtual representations of functions or components of a real device, mapped within a Logical Device. For example, a circuit breaker in a bay is represented by a Logical Node with the standard class **XCBR**. A Logical Device typically contains multiple Logical Nodes, each corresponding to a specific function, such as protection, control, or measurement
- **Data Objects or DO and Data Attributes or DA**: Information within a Logical Node is organized into Data Objects, which represent specific data sets related to a function. Each Data Object consists of Data Attributes, which describe detailed properties, such as values, states, or configurations. A set of Data Attributes in a Data Object follows a defined structure known as **CDC** (*Common Data Class*), ensuring standardization and interoperability
- **Functional Constraint or FC**: Specific services that can be applied to each Data Attribute and are part of an object's path

The format used by this Driver for all Tags representing Data Objects or Data Attributes is the following:

- **Device Parameter**: Server:LD, such as "AL_07:AL_07PRO"
- **Item Parameter**: LNFCData$DataAttribute, such as "XCBR1STMod$stVal"

Tag Reference

The **[N]** parameters of Tags are not used. Tags are addressed only with the **Device** and **Item** parameters, using the following format:

- **Device parameter**: ServerName:LDName
- **Item parameter**: LNFCData$DataAttribute. For more information, please check the next table

- **N1 Parameter**: For analog variables, defines an individual dead band for this Tag multiplied by 10. For example, to specify a 1.5% dead band, configure the *N1* parameter with the value 15. If the *N1* parameter is equal to 0 (zero), a default dead band is used, that is, the value configured in the **Analog Deadband %** option on the **IEC61850Srv** tab

Available options for the Item parameter

ITEM	OPERATION	MEANING
LNFCData$DataAttribute	Write-only	Any system Tag, whose value is informed using Links with Tags from other Drivers, if the file in **ICD** format contains a default value for the Data Attribute
LNRPurcbXXXX ou LNRPbcrbXXXX	--	This Driver does not display the tree corresponding to **Reports**, for the purpose of simplicity and using less Tags. All processing is performed internally
LNCOxxxx ou LNSPxxxxx	Reading and writing	Command or control block. This Driver allows interacting with commands using a Block Tag with 3 (three) Elements or using an individual Tag for each operation. Please check the next text for more information

A Block Tag is created with the value "$CommandBlock" and added at the end of the command's Data Object, with the Elements described on the next table.

Elements of a Block Tag

ELEMENT	DESCRIPTION
0 (zero): OperName	A **String** that indicates the name of the requested operation, usually "Select" or "Operate"
1 (one): CtlVal	A number code for a command
2 (two): Cause	Used to respond a specific error code for an IEC 61850 Client, when the **Command Profile** option is defined with the value **Treat By Application**. Please check the next table for all possible values

Possible values for the Cause Element

VALUE	DESCRIPTION
0	CmdErrUnknown
1	CmdErrNotSupported
2	CmdErrBlockedBySwitchingHierarchy
3	CmdErrSelectFailed
4	CmdErrInvalidPosition
5	CmdErrPositionReached

VALUE	DESCRIPTION
6	CmdErrParameterChangeInExecution
7	CmdErrStepLimit
8	CmdErrBlockedByMode
9	CmdErrBlockedByProcess
10	CmdErrBlockedByInterlocking
11	CmdErrBlockedBySynchrocheck
12	CmdErrCommandAlreadyInExecution
13	CmdErrBlockedByHealth
14	CmdErr1_Of_N_Control
15	CmdErrAbortionByCancel
16	CmdErrTimeLimitOver
17	CmdErrAbortionByTrip
18	CmdErrObjectNotSelected

When using individual *Select* or *Operate* Tags, the value of these Tags corresponds to the **CtlVal** (*Operation*) code.

Whenever a new command is received, the Block or PLC Tag triggers an **OnRead** event. Blocks Tags or PLC Tags with a configured **EnableDriverEvent** property also trigger an **OnTagRead** event. On this event, an application must send the requested operation to another system or Driver and then report the result. To report a result, the Block Tag or PLC Tag must be written back. If the value of **CtlVal** is the same, this operation is considered successful, otherwise it is considered as a failure.

Technical Specifications

This section contains information about the technical specifications of IEC 61850 Server Driver. The available specifications are the following:

- **Protocol Implementation Conformance Statement (PICS)**
- **Model Implementation Conformance Statement (MICS)**
- **Protocol Implementation Extra Information for Testing (PIXIT)**
- **Technical Issues Conformance Statement (TICS)**

Terms and Abbreviations

ABBREVIATION	MEANING
ACSI	Abstract Communication Service Interfaces
BDA	Basic Data Attribute (not structured)
DA	Data Attributes
DO	DATA in IEC 61850-7-2, data object type or instance
FCD	Functionally Constrained Data
FCDA	Functionally Constrained Data Attribute

ABBREVIATION	MEANING
ID	Identifier
IED	Intelligent Electronic Device
LD	Logical Device
LN	Logical Node
MSV	Multicast Sampled Value
RCB	Report Control Block
GCB	GOOSE Control Block or GSSE Control Block
SCL	Substation Configuration Language
SCSM	Specific Communication Service Mapping
XML	Extensible Markup Language
GSSE	Generic Substation State Events
GOOSE	Generic Object Oriented Substation Events
SCD	Substation Configuration Description File
ICD	IED Configuration Description
CID	Configured IED Description
PICS	Protocol Implementation Conformance Statement
MICS	Model Implementation Conformance Statement
PIXIT	Protocol Implementation eXtra Information for Testing
TICS	Tissue Implementation Conformance Statement

PICS

This section contains the technical specification **Protocol Implementation Conformance Statement (PICS)**.

Introduction

The technical specification **Protocol Implementation Conformance Statement (PICS)** contains information regarding **Abstract Communication Service Interface (ACSI, IEC 61850-7-2 First Edition 2003-05)** components that can be tested.

This document presents a summary of what was implemented for ACSI components in IEC 61850 Server Driver version **1.0** for **Elipse E3**, **Elipse Power**, or **Elipse Water**.

Basic Conformance

Y: Supported Service or N: Unsupported Service

		CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENTS
Client/Server Roles				
B11	**Server side (of TWO PARTY APPLICATION ASSOCIATION)**	-	Y	
B12	**Client side (of TWO PARTY APPLICATION**	N	-	

		CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENTS
	ASSOCIATION)			
SCSMs Supported				
B21	SCSM: IEC 6185-8-1 used	N	Y	
B22	SCSM: IEC 6185-9-1 used	N	N	
B23	SCSM: IEC 6185-9-2 used	N	N	
B24	SCSM: other	N	N	
Generic substation event model (GSE)				
B31	Publisher Side	-	N	
B32	Subscriber Side	N	-	
Transmission of sampled value model (SVC)				
B41	Publisher Side	-	N	
B42	Subscriber Side	N	-	

ACSI Models Conformance Statement

Y: Supported Service or N: Unsupported Service

		CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	VALUE OR COMMENTS
If Server or Client side (B11/12) supported				
M1	Logical device	N	Y	
M2	Logical node	N	Y	
M3	Data	N	Y	
M4	Data set	N	Y	
M5	Substitution	N	Y	
M6	Setting group control	N	N	
	Reporting			
M7	Buffered report control	N	Y	
M7-1	sequence-number	N	Y	
M7-2	report-time-stamp	N	Y	
M7-3	reason-for-inclusion	N	Y	
M7-4	data-set-name	N	Y	
M7-5	data-reference	N	Y	
M7-6	buffer-overflow	N	Y	
M7-7	entryID	N	Y	
M7-8	BufTim	N	Y	

		CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	VALUE OR COMMENTS
M7-9	**IntgPd**	N	Y	
M7-10	**GI**	N	Y	
M7-11	**conf-revision**	N	Y	
M8	**Unbuffered report control**	N	Y	
M8-1	**sequence-number**	N	Y	
M8-2	**report-time-stamp**	N	Y	
M8-3	**reason-for-inclusion**	N	Y	
M8-4	**data-set-name**	N	Y	
M8-5	**data-reference**	N	Y	
M8-6	**BufTim**	N	Y	
M8-7	**IntgPd**	N	Y	
M8-8	**GI**	N	Y	
M8-9	**conf-revision**	N	Y	
	Logging	N	N	
M9	**Log control**	N	N	
M9-1	**IntgPd**	N	N	
M10	**Log**	N	N	
M11	**Control**	N	N	
If GSE (B31/32) is supported				
M12	**GOOSE**	N	N	
M13	**GSSE**	N	N	
If SVC (41/42) is supported				
M14	**Multicast SVC**	N	N	
M15	**Unicast SVC**	N	N	
If Server or Client side (B11/12) supported				
M16	**Time**	N	N	
M17	**File Transfer**	N	N	

ACSI Service Conformance Statement

AA: Application Association, TP: Two-Party, or MC: Multicast

	SERVICES	AA: TP OR MC	CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENTS
Server					
S1	**GetServerDirectory**		N	Y	
Application Association					

	SERVICES	AA: TP OR MC	CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENTS
S2	**Associate**	TP	N	Y	
S3	**Abort**	TP	N	Y	
S4	**Release**	TP	N	Y	
Logical Device					
S5	**GetLogicalDeviceDirectory**	TP	N	Y	
Logical Node					
S6	**GetLogicalNodeDirectory**	TP	N	Y	
S7	**GetAllDataValues**	TP	N	Y	
Data					
S8	**GetDataValues**	TP	N	Y	
S9	**SetDataValues**	TP	N	Y	
S10	**GetDataDirectory**	TP	N	Y	
S11	**GetDataDefinition**	TP	N	Y	
Data Set					
S12	**GetDataSetValues**	TP	N	Y	
S13	**SetDataSetValues**	TP	N	Y	
S14	**CreateDataSet**	TP	N	Y	
S15	**DeleteDataSet**	TP	N	Y	
S16	**GetDataSetDirectory**	TP	N	Y	
Substitution					
S17	**SetDataSetValues**	TP	N	N	
Setting Group Control					
S18	**SelectActiveSG**	TP	N	N	
S19	**SelectEditSG**	TP	N	N	
S20	**SetSGValues**	TP	N	N	
S21	**ConfirmEditSGValues**	TP	N	N	
S22	**GetSGValues**	TP	N	N	
S23	**GetSGCBValues**	TP	N	N	
Reporting					
Buffered Report Control Block					
S24	**Report**	TP	N	Y	

	SERVICES	AA: TP OR MC	CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENTS
S24-1	**data-change (dchg)**	TP	N	Y	
S24-2	**quality-change (qchg)**	TP	N	Y	
S24-3	**data-update (dupd)**	TP	N	Y	
S25	**GetBRCBValues**	TP	N	Y	
S26	**SetBRCBValues**	TP	N	Y	
Unbuffered Report Control Block					
S27	**Report**	TP	N	Y	
S27-1	**data-change (dchg)**	TP	N	Y	
S27-2	**quality-change (qchg)**	TP	N	Y	
S27-3	**data-update (dupd)**	TP	N	Y	
S28	**GetURCBValues**	TP	N	Y	
S29	**SetURCBValues**	TP	N	Y	
Logging					
Log Control Block					
S30	**GetLCBValues**	TP	N	N	
S31	**SetLCBValues**	TP	N	N	
Log					
S32	**QueryLogByTime**	TP	N	N	
S33	**QueryLogAfter**	TP	N	N	
S34	**GetLogStatusValues**	TP	N	N	
Generic substation event model (GSE)					
GOOSE Control Block					
S35	**SendGOOSEMessage**	TP	N	N	
S36	**GetGoReference**	TP	N	N	
S37	**GetGOOSEElementNumber**	TP	N	N	
S38	**GetGoCBValues**	TP	N	N	
S39	**SetGoCBValues**	TP	N	N	
GSSE Control Block					
S40	**SendGSSEMessage**	TP	N	N	

	SERVICES	AA: TP OR MC	CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENTS
S41	**GetGsReference**	TP	N		
S42	**GetGSSEDataOffset**	TP	N		
S43	**GetGsCBValues**	TP	N		
S44	**SetGsCBValues**	TP	N		
Transmission of sampled value model (SVC)					
Multicast SVC					
S45	**SendMSVMessage**	TP	N	N	
S46	**GetMSVCBValues**	TP	N	N	
S47	**SetMSVCBValues**				
Unicast SVC					
S48	**SendUSVMessage**	TP	N	N	
S49	**GetUSVCBValues**	TP	N	N	
S50	**SetUSVCBValues**	TP	N	N	
Control					
S51	**Select**	TP	N	Y	
S52	**SelectWithValue**	TP	N	Y	
S53	**Cancel**	TP	N	Y	
S54	**Operate**	TP	N	Y	
S55	**Command-Termination**	TP	N	Y	
S56	**TimeActivated-Operate**	TP	N	Y	
File Transfer					
S57	**GetFile**	TP	N	N	
S58	**SetFile**	TP	N	N	
S59	**DeleteFile**	TP	N	N	
S60	**GetFileAttributeValues**	TP	N	N	
Time					
T1	**Time Resolution of internal clock**		10	N	Nearest negative power of 2 (two), in seconds
T2	**Time Accuracy of internal clock**		N	Y	T0
			N	Y	T1
			N	N	T2

	SERVICES	AA: TP OR MC	CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENTS
			N	N	T3
			N	N	T4
			N	N	T5
T3	**Supported Timestamp resolution**		10	N	Nearest negative power of 2 (two), in seconds

SCL Conformance Degree

SCL CONFORMANCE DEGREE		CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENT
SCL.1	SCL File for Implementation available offline		Y	A third-party SCL configuration tool must be used
SCL.2	SCL File available for Implementation online		N	
SCL.3	SCL implementation reconfiguration supported online		Y	SCL File can be reloaded online

SUPPORTED ACSI SERVICES FOR SCL.2 AND SCL.3		CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENT
	GetFileAttributeValues		N	
	GetFile		N	
	SetFile		N	
	DeleteFile		N	
	GetDataValues		N	
	SetDataValues		N	
	SCL Control Block		N	
	SCL File Structure		N	
	Remote Creation of SCL File		N	

ADDITIONAL MMS SERVICES FOR SCL.2 AND SCL.3		CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENT
	GetCapabilityList		N	
	GetDomainAttributes		N	
	LoadDomainContent		N	
	StoreDomainContent		N	

DEFINITION OF SCL CONTROL BLOCK (IEC61850-8-1 COMPONENT NAME)		CLIENT OR SUBSCRIBER	SERVER OR PUBLISHER	COMMENT
	Validade		N	
	ValState		N	
	Activate		N	

MICS

This section contains the technical specification **Model Implementation Conformance Statement (MICS)**.

Introduction

The technical specification **Model Implementation Conformance Statement (MICS)** applies to IEC 61850 Server Driver version **1.0**. This document specifies modeling extensions compared to version 1 (one) of IEC 61850 protocol.

Supported Common Data Classes

STATUS INFORMATION	
SPS	Single Point Status
DPS	Double Point Status
INS	Integer Status
ACT	Protection Activation Information
ACD	Directional Protection Activation Information
SEC	Security Violation Counting
BCR	Binary Counter Reading

MENSURAND INFORMATION	
MV	Single Point Status
CMV	Double Point Status
SAV	Integer Status
WYE	Phase to ground related measured values of a three phase system
DEL	Phase to phase related measured values of a three phase system
SEQ	Sequence
HMV	Harmonic Value
HWYE	Harmonic Valur for WYE
HDEL	Harmonic Value for DEL

CONTROLLABLE STATUS INFORMATION	
SPC	Controllable Single Point
DPC	Controllable Double Point
INC	Controllable Integer Status
BSC	Binary Controlled step position information
ISC	Integer controlled step position information

CONTROLLABLE ANALOG INFORMATION	
APC	Controllable Analog set point information

STATUS CONFIGURATION SPECIFICATIONS	
SPG	Single Point Setting
ING	Integer Status Setting

ANALOG CONFIGURATION SPECIFICATIONS	
ASG	Analog Setting
CURVE	Setting Curve

SPECIFICATIONS FOR DESCRIPTION INFORMATION	
DPL	Device Name plate
CURVE	Logical Node Name plate
CSD	Curve Shape Description

Unsupported Common Data Classes

STATUS CONFIGURATION SPECIFICATIONS (BY GROUP OF OPERATIONS)	
SPG	Single Point Setting
ING	Integer Status Setting

PIXIT

This section contains the technical specification **Protocol Implementation Extra Information for Testing (PIXIT)**.

Introduction

This document contains the technical specification **Protocol Implementation Extra Information for Testing (PIXIT)** of IEC 61850 Server Driver version **1.0**, referenced in this document as "System".

Together with technical specifications **PICS** and **MICS**, the technical specification **PIXIT** forms the basis for a conformance test based on IEC 61850-10 standard.

PIXIT for Association Model

Y: Supported Service or N: Unsupported Service

IDENTIFIER	DESCRIPTION	VALUE
As1	Maximum number of clients that can set up an association simultaneously	1 (one)
As2	TCP_KEEPALIVE value	Defined on Windows
As3	Lost Connection Detection Time	Configurable
As4	Is authentication supported	N
As5	What association parameters are necessary for successful association	Transport Selector: Y; Session Selector: Y; Presentation Selector: Y; AP Title: Y; AE Qualifier: Y
As6	If association parameters are necessary for successful association, describe the correct values	Configurable
As7	What is the maximum and minimum MMS PDU size	Max MMS PDU Size: 32000, Min MMS PDU Size: 16000
As8	What is the maximum startup time after a power supply interrupt	After windows startup (depends on machine) it takes 15 seconds

PIXIT for Server Model

Y: Supported Service or N: Unsupported Service

IDENTIFIER	DESCRIPTION	VALUE
Sr1	What analogue value (MX) quality bits are supported or can be set by server?	Good: Y; Invalid: Y; Reserved: N; Questionable: Y; Overflow: Y; OutofRange: Y; BadReference: N; Oscillatory: Y; Failure: Y; OldData: Y; Inconsistent: N; Inaccurate: Y; Process: N; Substituted: Y; Test: N; OperatorBlocked: N
Sr2	Which Status Value (ST) quality bits are supported or can be set by server?	Good: Y; Invalid: Y; Reserved: N; Questionable: Y; Overflow: Y; OutofRange: Y; BadReference: N; Oscillatory: Y; Failure: Y; OldData: Y; Inconsistent: N; Inaccurate: Y; Process: N; Substituted: Y; Test: N; OperatorBlocked: N (please check text after this table for more information)
Sr3	What is the maximum number of data values in one GetDataValues request?	65000
Sr4	What is the maximum number of data values in one SetDataValues request?	65000
Sr5	Which Mode or Behavior values are supported?	On: N; Blocked: N; Test: N; Test or Blocked: N; Off: N

Quality bits are received in the **Quality** property of each Tag variable. The **Quality** property follows OPC DA specification and contains a fixed byte mapping, **QQSSSSLL**, in which **QQ** means mapped to good, invalid, reserved, and questionable bits in the order Good, Bad, Bad, and Uncertain OPC values, **SSSS** means maps all other bits, and **LL** means not mapped.

- **Good,Process**: Good non specific,192
- **Good, Substituted**: Local Override, 216
- **Invalid, Overflow**: Bad, non specific, 0 (zero)
- **Invalid, Out of Range**: Bad, Config Error, 4 (four)
- **Invalid, Bad Reference**: Bad, Config Error, 4 (four)
- **Invalid, Oscillatory**: Bad, Sensor Failure, 16
- **Invalid, Failure**: Bad, Device Failure, 12
- **Invalid, Old Data**: Bad, Last Known Value, 20
- **Invalid**: Bad, non specific, 0 (zero)
- **Questionable, Out of Range or Bad reference or Oscillatory**: Uncertain, Sensor not accurate, 80
- **Questionable, Old Data**: Uncertain, Last Usable Value, 68
- **Questionable, Inaccurate or Inconsistent**: Uncertain, Sensor not Accurate, 80

- **Questionable, Overflow**: Uncertain, EU Exceeded, 84
- **Questionable**: Uncertain, non specific, 64

Quality codes are carried together with their timestamp and value and can be configured as a Link to any object in **Elipse E3**, **Elipse Power**, or **Elipse Water**.

PIXIT for Dataset Model

IDENTIFIER	DESCRIPTION	VALUE
Ds1	What is the maximum number of data elements in one data set (compare ICD setting)?	65000
Ds2	How many persistent data sets can be created by one or more clients?	65000
Ds3	How many non-persistent data sets can be created by one or more clients?	Not supported

PIXIT for Substitution Model

IDENTIFIER	DESCRIPTION	VALUE
Sb1	Are substituted values stored in volatile memory?	Not supported

PIXIT for Setting Group Model

IDENTIFIER	DESCRIPTION	VALUE
Sg1	What is the number of supported setting groups for each logical device or compare NumSG in the SGCB?	Not supported
Sg2	What is the effect of when and how the non-volatile storage is updated or compare IEC 61850-8-1 $16.2.4?	-
Sg3	Can multiple clients edit the same setting group?	-
Sg4	What happens if the association is lost while editing a setting group?	-
Sg5	Is EditSG value 0 (zero) allowed?	-

PIXIT for Reporting Model

Y: Supported Service or N: Unsupported Service

IDENTIFIER	DESCRIPTION	VALUE
Rp1	The supported trigger conditions are (compare PICS)	integrity: Y; data change: Y; quality change: Y; data update: Y; general interrogation: Y

IDENTIFIER	DESCRIPTION	VALUE
Rp2	The supported optional fields are	sequence-number: Y; report-time-stamp: Y; reason-for-inclusion: Y; data-set-name: Y; data-reference: Y; buffer-overflow: Y; entryID: Y; conf-rev: Y; segmentation: Y
Rp3	Can the server send segmented reports?	Y
Rp4	Mechanism on second internal data change notification of the same analogue data value within buffer period (Compare IEC 61850-7-2 $14.2.2.9)	Send report immediately OR Replace analogue value in pending report
Rp5	Multi client URCB approach (compare IEC 61850-7-2 $14.2.1)	Each URCB is visible to all clients
Rp6	What is the format of EntryID?	Based on datetime and sequence number in hex: DDMMYYHHMMSS SEQNUMBER
Rp7	What is the buffer size for each BRCB or how many reports can be buffered?	Not explicitly limited, depends on available memory
Rp8	Pre-configured RCB attributes that cannot be changed online when RptEna = FALSE (see also the ICD report settings)	N
Rp9	May the reported data set contain: structured data objects and data attributes?	Y
Rp10	What is the scan cycle for binary events? Is this fixed, configurable	Configurable
Rp11	Does the device support to pre-assign a RCB to a specific client in the SCL?	N

PIXIT for Logging Model

IDENTIFIER	DESCRIPTION	VALUE
Lg1	What is the default value of LogEna (Compare IEC 61850-8-1 $17.3.3.2.1, the default value should be FALSE)?	Not available
Lg2	What is the format of EntryID (Compare IEC 61850-8-1 $17.3.3.3.1)?	Not available
Lg3	If there are multiple Log Control Blocks that specify the Journaling of the same MMS NamedVariable and TrgOps and the Event Condition (Compare IEC 61850-8-1 $17.3.3.3.2)	Not available
Lg4	Pre-configured LCB attributes that cannot be changed online	Not available

PIXIT for Control Model

Y: Supported Service or N: Unsupported Service

IDENTIFIER	DESCRIPTION	VALUE
Ct1	Which control modes are supported?	status-only: Y; direct-with-normal-security: Y; sbo-with-normal-security: Y; direct-with-enhanced-security: Y; sbo-with-enhanced-security: Y
Ct2	Is the control model fixed, configurable or online changeable?	Configurable and online changeable
Ct3	Is Time activated operate (operTm) supported?	N
Ct4	Is operate-many supported?	N
Ct5	Will the DUT activate the control output when the test attribute is set in the SelectWithValue or Operate request (when N test procedure Ctl2 is applicable)?	N
Ct6	What are the conditions for the time (T) attribute in the SelectWithValue or Operate request?	DUT ignores the time and execute the command as usual
Ct7	Is pulse configuration supported?	N
Ct8	What is the behaviour of the DUT when the check conditions are set. Is this behaviour fixed, configurable, or online changeable?	DUT ignores the check value
Ct9	What additional cause diagnosis are supported?	Blocked-by-switching-hierarchy: Y/N; Select-failed: Y/N; Invalid-position: Y/N; Position-reached: Y/N; Parameter-change-in-execution: Y/N; Step-limit: Y/N; Blocked-by-Mode: Y/N; Blocked-by-process: Y/N; Blocked-by-interlocking: Y/N; Blocked-by-synchrocheck: Y/N; Command-already-in-execution: Y/N; Blocked-by-health: Y/N; 1-of-n-control: Y/N; Abortion-by-cancel: Y/N; Time-limit-over: Y/N; Abortion-by-trip: Y/N
Ct10	How to force a test-not-ok respond with SelectWithValue request?	
Ct11	How to force a test-not-ok respond with Select request?	
Ct12	How to force a test-not-ok respond with Operate request?	
Ct13	Which origin categories are supported?	
Ct14	What happens if the orCat value is not supported?	

IDENTIFIER	DESCRIPTION	VALUE
Ct15	Does the IED accept a SelectWithValue or Operate with the same ctlVal as the current status value?	
Ct16	Does the IED accept a select or operate on the same control object from two different clients at the same time?	
Ct17	Does the IED accept a Select or SelectWithValue from the same client when the control object is already selected (tissue 334)?	SBOns: Y; SBOes: Y
Ct18	Is for SBOes the internal validation performed during the SelectWithValue or Operate step?	SelectWithValue: Y; Operate: Y; SelectWithValue and Operate: Y
Ct19	Can a control operation be blocked by Mod=Off or Blocked?	N
Ct20	Does the IED support local or remote operation?	N
Ct21	Does the IED send an InformationReport with LastApplError as part of the Operate response- for control with normal security?	SBOns: N; Dons: N

PIXIT for Time and Synchronization Model

Y: Supported Service or N: Unsupported Service

IDENTIFIER	DESCRIPTION	VALUE
Tm1	What quality bits are supported (may be set by the IED)?	LeapSecondsKnown: N; ClockFailure: N; ClockNotSynchronized: N
Tm2	Describe the behavior when the time synchronization signal or messages are lost	
Tm3	When is the time quality bit ClockFailure set?	
Tm4	When is the time quality bit Clock not synchronised set?	
Tm5	Is the timestamp of a binary event adjusted to the configured scan cycle?	
Tm6	Does the device support time zone and daylight saving?	
Tm7	Which attributes of the SNTP response packet are validated?	Leap indicator not equal to 3: N; Mode is equal to SERVER: N; OriginateTimestamp is equal to value sent by the SNTP client as Transmit Timestamp: N; RX/TX timestamp fields are checked for reasonableness: N; SNTP version 3 or 4: N; other (describe): N

TICS

This section contains the technical specification **Technical Issues Conformance Statement (TICS)**.

Introduction

IEC 61859 standard series was published as Edition 1 between 2002 and 2004. As part of the maintenance process for the standard, technical issues, called TISSUES, were collected from the beginning in cooperation with the UCA international users group, which is a community for users of IEC 61850 standard. The collected TISSUES can be categorized in two groups:

- TISSUES that can affect interoperability between implementations of the standard and need either corrections or clarifications (**IntOp**-type TISSUES)
- TISSUES that propose new features to implement in future versions of the standard (**Ed. 2**-type TISSUES)

During the October 2006 meeting IEC TC57, the working group 10 decided that **IntOp** Tissues are mandatory for IEC 61850 edition 1 and **Ed. 2** Tissues must not be implemented. The next topic contains an overview of all **IntOp** Tissues implemented.

This document declares the technical issues addressed for IEC 61850 Server Driver version **1.0** for **Elipse E3**, **Elipse Power**, or **Elipse Water**.

Mandatory Technical Issues

PART	TITLE	STATUS	COMMENTS
Part 6			
1	Syntax	OK	
5	tExtensionAttributeNameEnum is restricted	OK	
8	SIUnit enumeration for W	OK	
10	Base type for bitstring usage	OK	
17	DAI/SDI elements syntax	OK	
169	Ordering of enum differs from 7-3	OK	
Part 7-2			
30	control parameter T	OK	
31	Typo	OK	
32	Typo in syntax	OK	
35	Typo Syntax Control time	OK	
36	Syntax parameter DSet-Ref missing	-	GOOSE not implemented
37	Syntax GOOSE *T* type	-	GOOSE not implemented
38	Syntax *AppID* or *GoID*	-	GOOSE not implemented
39	Add DstAddr to GoCB	-	GOOSE not implemented

PART	TITLE	STATUS	COMMENTS
40	GOOSE Message *AppID* to *GoID*	-	GOOSE not implemented
41	GsCB *AppID* to *GsID*	-	GOOSE not implemented
42	SV timestamp: *EntryTime* to *TimeStamp*	-	**SampledValues** are not supported
43	Control *T* semantic	OK	
44	AddCause - Object not sel	OK	
45	Missing AddCauses (neg range)	OK	
46	Synchro check cancel	-	
47	. (dot) in LD Name?	OK	
49	BRCB TimeOfEntry (part of #453)	OK	
50	LNName start with number?	OK	
51	ARRAY [0..num] missing	OK	**Array** not supported
52	Ambiguity GOOSE SqNum	-	GOOSE not implemented
53	Add DstAddr to GsCB, SV	-	GOOSE not implemented
151	Name constraint for control blocks etc.	OK	
166	DataRef attribute in Log	-	Logging not implemented
185	Logging - Integrity period	-	Logging not implemented
189	SV Format		**SampledValues** are not supported
190	BRCB: EntryId and TimeOfEntry (part of #453)	OK	
191	BRCB: Integrity and buffering reports (part of #453)	OK	
234	New type CtxInt (Enums are mapped to eight-bit integer)	OK	
275	Confusing statement on GI usage (part of #453)	OK	
278	EntryId not valid for a server (part of #453)	OK	
Part 7-3			
28	Definition of APC	OK	
54	Point def xVal, not cVal	OK	
55	Ineut = Ires?	OK	
60	Services missing in tables	OK	
63	mag in CDC CMV	OK	
219	operTm in ACT	OK	

PART	TITLE	STATUS	COMMENTS
270	WYE and DEL rms values	OK	
Part 8-1			
116	GetNameList with empty response?	OK	
165	Improper Error Response for GetDataSetValues	OK	
183	GetNameList error handling	OK	

Documentation of I/O Interfaces

This section contains the documentation of I/O Interfaces referring to **IEC61850Srv** Driver.

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 **Elipse 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 **Elipse E3** version 2.0 or later, click **Configure driver** on a Driver's toolbar. In **Elipse 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

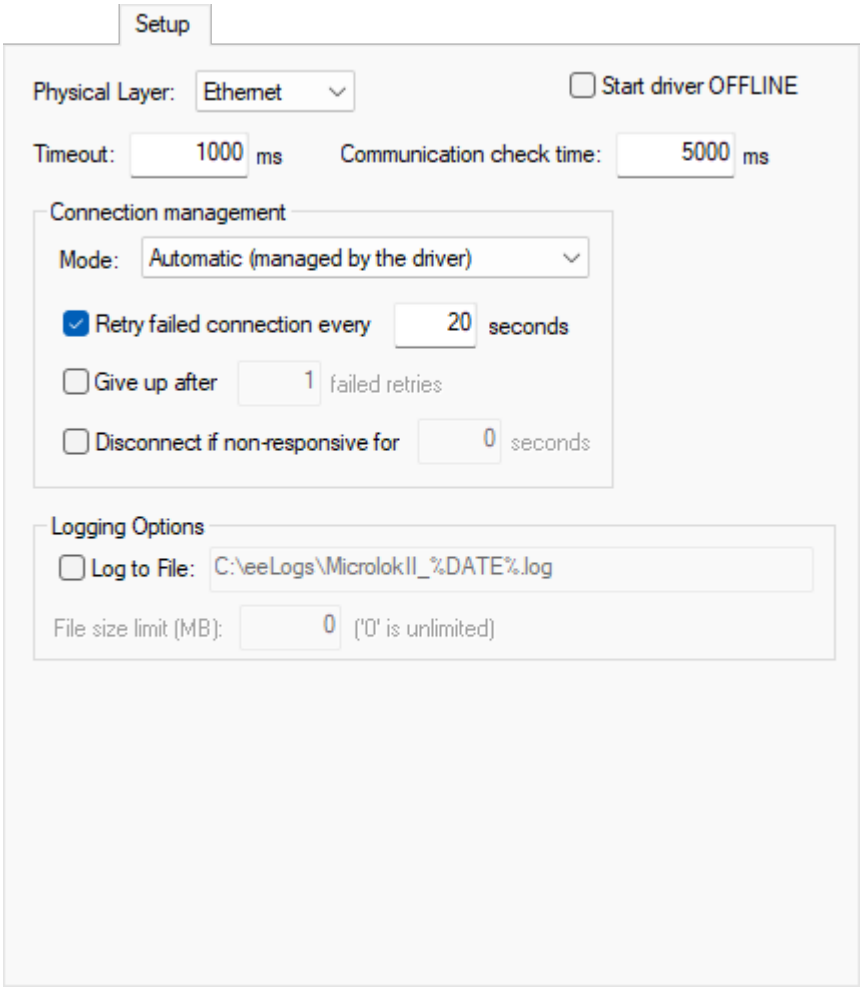
The dialog box of I/O Interfaces allows configuring the I/O connection used by a Driver. This dialog box contains the **Setup**, **Serial**, **Ethernet**, **Modem**, and **RAS** tabs, described on the next topics. If a Driver does not implement a specific I/O connection, its corresponding tab is not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

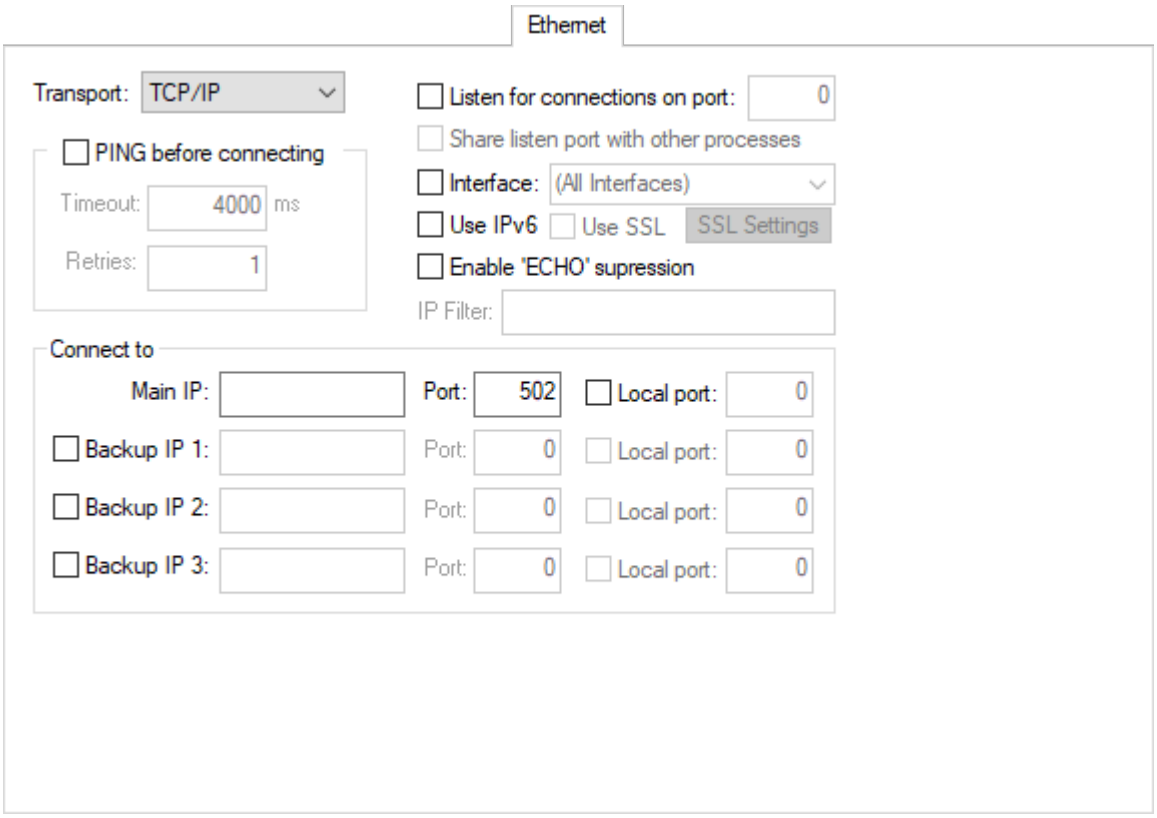
OPTION	DESCRIPTION
Mode	Selects a management mode of a connection. Selecting the **Automatic** option allows a Driver to manage the connection automatically, as specified in the next options. Selecting the **Manual** option allows an application to fully manage a connection
Retry failed connection every ... 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
Log to File	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 **0FDAh**. 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**
File size limit (MB)	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

Ethernet Tab

Use this tab to configure parameters of an **Ethernet** Interface. These parameters, except port configurations, must also be configured for use in the **RAS** Interface.



Ethernet tab

Available options on the Ethernet tab

OPTION	DESCRIPTION
Transport	Select the value **TCP/IP** for a TCP socket (*stream*) or select the value **UDP/IP** to use a UDP socket (*connectionless datagram*)
Listen for connections on port	Use this option to wait for new connections in a specific IP port, common in Slave Drivers. If this option remains unselected, a Driver connects to the address and port specified in the **Connect to** option
Share listen port with other processes	Select this option to share the listening port with other Drivers and processes
Interface	Select the local network interface, identified by its IP address, that a Driver uses to establish and receive connections, or select the value **(All Interfaces)** to allow connection in any network interface
Use IPv6	Select this option to force a Driver to use addresses in **IPv6** format on all Ethernet connections. Leave this option deselected to use the **IPv4** format
Enable 'ECHO' suppression	Enable this option to remove the echo from received data. An echo is a copy of sent data, which can be returned before a reply message
IP Filter	List of restricted or allowed IP addresses from where a Driver accepts connections (*Firewall*). Please check the **IO.Ethernet.IPFilter** property for more information
PING before connecting	Enable this option to execute a **ping** command, that is, to check whether a device can be reached on a network, for a device before trying a socket connection. This is a quick way

OPTION	DESCRIPTION
	of determining a successful connection before trying to open a socket with a device. The time-out of a connection with a socket can be very high. The available options are: • **Timeout**: Specify the number of milliseconds to wait for a reply from a **ping** command. Users must use a **ping** command to check the normal reply time, configuring this option for a value above that average. Usually this value can be configured between 1000 and 4000 milliseconds, that is, between 1 (one) and 4 (four) seconds • **Retries**: Number of retries of a **ping** command, not counting the first attempt. If all attempts fail, then the socket connection is aborted

Available options on the Connect to group

OPTION	DESCRIPTION
Main IP	Type the IP address of a remote device. Users can use an IP address separated by dots, as well as a URL. In case of a URL, a Driver uses the available DNS service to map that URL to an IP address, such as "192.168.0.13" or "Server1"
Port	Type the IP port of a remote device, between 0 (zero) and 65535
Local port	Select this option to use a fixed local IP port when connecting to a remote device
Backup IP 1, 2, and 3	Indicate the IP address, the IP port, and the fixed local IP port of up to 3 (three) backup addresses of a remote device

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 vlaue **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 **Elipse 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

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.

Ethernet Interface Configuration

This section contains information about the configuration of **I/O Tags** and **Properties** of an **Ethernet** Interface.

I/O Tags

Tags of an Ethernet Interface (N2/B2 = 4)

The Tags described next allow controlling and identifying an **Ethernet** Interface at run time and they are also valid when the **RAS** Interface is selected.

IMPORTANT
These Tags are available **ONLY** while a Driver is in **Online** mode.

IO.Ethernet.IPSelect

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

Indicates the active IP address. Possible values are **0**: The main IP address is selected, **1**: The first alternative or backup IP address is selected, **2**: The second alternative or backup IP address is selected, or **3**: The third alternative or backup IP address is selected.

If the **Ethernet** or **RAS** Interface is connected, this Tag indicates which one of the four configured IP addresses is in use. If the Interface is disconnected, this Tag indicates which IP address is used first on the next attempt to connect.

During the connection process, if the active IP address is not available, the I/O Interface tries to connect using the other IP address. If the connection with the alternative IP address works, it is configured as the active IP address (automatic switchover).

To force a manual switchover, write values from 0 (zero) to 3 (three) to this Tag. This forces a reconnection with the specified IP address (**0**: Main address or **1, 2, 3**: Alternative address) if a Driver is currently connected. If a Driver is disconnected, this Tag configures the active IP address for the next attempt to connect.

IO.Ethernet.IPSwitch

Type of Tag	I/O Tag
Type of Access	Write-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	4 (four)
N4 Parameter	1 (one)
String Configuration	IO.Ethernet.IPSwitch

Any value written to this Tag forces a manual switchover. If the main IP address is active, then the first alternative or backup IP address is activated, and so on for all alternative IP addresses and returning to the main address until a connection is established.

If a Driver is disconnected, this Tag configures the active IP address for the next attempt to connect.

IO.Ethernet.SocketState

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	4 (four)
N4 Parameter	2 (two)
String Configuration	IO.Ethernet.SocketState

The Value property of this Tag corresponds to socket states as a map of bits:

- **Bit 0**: 0 (zero, not listening) or 1 (one, listening)
- **Bit 1**: 0 (zero, disconnected) or 1 (one, connected)

Properties

These properties control the configuration of an **Ethernet** Interface.

NOTE

The **Ethernet** Interface is also used by the **RAS** Interface.

IO.Ethernet.AcceptConnection

Configure to False if a Driver must not accept external connections, that is, if a Driver behaves as a master, or configure to True to enable the reception of connections, that is, if a Driver behaves as a slave.

IO.Ethernet.BackupEnable[2,3]

Configure to True to enable an alternative or backup IP address. If the reconnection attempt with the main IP address fails, a Driver tries to use an alternative IP address. Configure to False to disable its usage.

IO.Ethernet.BackupIP[2,3]

Alternative or backup IP address of a remote device. Users can use a numerical address, as well as a device's host name, such as "192.168.0.7" or "SERVER2".

IO.Ethernet.BackupLocalPort[2,3]

Local port number to be used when connecting to an alternative IP address of a remote device. Used only if **IO.Ethernet.BackupLocalPortEnable** is equal to True.

IO.Ethernet.BackupLocalPortEnable[2,3]

Configure to True to force the use of a specific local port when connecting to an alternative or backup IP address or configure to False to use any available local port.

IO.Ethernet.BackupPort[2,3]

Port number of an alternative or backup IP address of a remote device, used with the **IO.Ethernet.BackupIP** property.

IO.Ethernet.IPFilter

List with a comma-separated IPv4 or IPv6 addresses, which defines from which addresses a Driver accepts or blocks connections. Users can use asterisks, such as "192.168.*.*", or intervals, such as "192.168.0.41-50", in any part of IP addresses. To block an IP address or a range of IP addresses, use the tilde ("~") character at the beginning of the address, according to the next examples:

- **192.168.0.24**: Accepts only connections from IPv4 address 192.168.0.24
- **192.168.0.41-50**: Accepts connections from IPv4 addresses in the interval between 192.168.0.41 and 192.168.0.50
- **192.168.0.***: Accepts connections from IPv4 addresses in the interval between 192.168.0.0 and 192.168.0.255
- **fe80:3bf:877::*:* (expands to fe80:03bf:0877:0000:0000:0000:*:*)**: Accepts connections from IPv6 addresses in the interval between fe80:03bf:0877:0000:0000:0000:0000:0000 and fe80:03bf:0877:0000:0000:0000:ffff:ffff
- **192.168.0.10, 192.168.0.15, 192.168.0.20**: Accepts connections from IPv4 addresses 192.168.0.10, 192.168.0.15, and 192.168.0.20
- **~192.168.0.95, 192.168.0.***: Accepts connections from IPv4 addresses in the interval between 192.168.0.0 and 192.168.0.255, except the IPv4 address 192.168.0.95

When a Driver receives a connection attempt, the list of filters is scanned sequentially from left to right, searching for a specific authorization or block for the IP address where the connection comes from. If no element on the list corresponds to the IP address, the authorization or block are dictated by the last element of that list:

- If the last element on the list is an authorization, such as "192.168.0.24", then all IP addresses not found on the list are blocked
- If the last element on the list is a block, such as "~192.168.0.24", then all IP addresses not found on the list are authorized

If an IP address appears on more than one filter on the list, the leftmost filter has precedence. For example, in case of "~192.168.0.95, 192.168.0.*", the IP address 192.168.0.95 fits both rules, but the rule that wins is the leftmost one, "~192.168.0.95", and therefore this IP address is blocked.

When **IOKit** blocks a connection, it logs a message "Blocked incoming socket connection from {IP}!".

In case of UDP connections in broadcast listening mode, in which a Driver can receive packets from different IP addresses, blocks or permissions are performed at each packet received. If a packet is received from a blocked IP address, it logs a message "Blocked incoming packet from {IP} (discarding {N} bytes)!".

IO.Ethernet.ListenIP

A IP address of the local network interface that a Driver uses to establish and accept connections. Leave this property empty to establish and accepts connections using any local network interface.

IO.Ethernet.ListenPort

9 Number of the IP port used by a Driver to listen to connections.

IO.Ethernet.MainIP

A IP address of a remote device. Users can use a numerical address, as well as a device's host name, such as "192.168.0.7" or "SERVER2".

IO.Ethernet.MainLocalPort

9 Local port number to use when connecting to the main IP address of a remote device. This value is only used if the **IO.Ethernet.MainLocalPortEnable** property is equal to True.

IO.Ethernet.MainLocalPortEnable

◩ Configure to True to force the use of a specific local port when connecting to the main IP address of a remote device or configure to False to use any available local port.

IO.Ethernet.MainPort

9 Number of the IP port of a remote device, used with the **IO.Ethernet.MainIP** property.

IO.Ethernet.PingEnable

◩ Configure to True to enable sending a **ping** command to the IP address of a remote device, before trying to connect to the socket. This socket's connection time-out cannot be controlled, therefore sending a **ping** command before connecting is a fast way to detect if the connection is going to fail. Configure to False to disable a **ping** command.

IO.Ethernet.PingTimeoutMs

9 Delay time to wait for a response from a **ping** command, in milliseconds.

IO.Ethernet.PingTries

9 Maximum number of attempts of a **ping** command. Minimum value is 1 (one), including the first **ping** command.

IO.Ethernet.ShareListenPort

Configure to True to share a listening port with other Drivers and processes or False to open a listening port in exclusive mode. To successfully share a listening port, all Drivers and processes that use that port must open it in shared mode. When a listening port is shared, each incoming connection is distributed to one of the processes listening. This way, if a Slave Driver only supports one connection at a time, users can use several instances of this Driver listening on the same port, therefore simulating a Driver with support for multiple connections.

IO.Ethernet.SupressEcho

Configure to True to eliminate echoes in communication. An echo is the unwanted reception of an exact copy of all data packets a Driver sent to a device.

IO.Ethernet.Transport

Defines a transport protocol. Possible values are **T or TCP**: Uses the TCP/IP protocol or **U or UDP**: Uses the UDP/IP protocol.

IO.Ethernet.UseIPv6

Configure to True to use IPv6 addresses on all Ethernet connections or configure to False to use IPv4 addresses (default).

Driver Revision History

VERSION	DATE	AUTHOR	COMMENTS
1.0.20	01/26/2026	M. Ludwig	• Driver updated to **IOKit** library version **3.0** and Visual Studio 2022 (*Case 37981*).
1.0.19	06/05/2025	M. Salvador	• Now the *N1* parameter, if greater than 0 (zero), allows defining an individual dead band as a percentage, multiplied by 10 (*Case 37082*). • Implemented the **Owner** property in BRCBs (*Case 37084*). • Fixed a problem in which the **ResvTms** property was not being used correctly (*Case 37083*). • Revision of **XMPP** transport layer (*Case 37740*).
1.0.18	01/06/2025	M. Salvador	• This Driver now adds the name of a Logical Device when requesting an **MMS READ** command from any item whose Domain is equal to "LD0" (*Case 18069*). • The path of files in **LDS** format is now relative, it

VERSION	DATE	AUTHOR	COMMENTS
			must start with a backslash (\) and may or may not finish with a backslash (*Case 22021*). • Added support to the optional transport layer IEC 61850-8-2 (**XMPP**) (*Case 23785*). • Improvements for stress tests with 10,000 **XMPP** connections (*Case 30848*). • Now this Driver accepts more than connection with clients at the same TCP/IP port, 102, when selecting the **Share listen port** option (*Case 31150*). • Adjustment in the Link layer to not include the *CalledApTitle* parameter if that parameter is not included in the request (*Case 19761*). • Fixed the behavior of commands when an object contains the **OperTm** member (*Case 27051*). • Fixed a problem when writing the **RptEna** parameter from Reports if the written value is the same as the current value (*Case 27188*). • Implemented corrections for this Driver to work with other 61850 clients (*Case 30201*). • Fixed a problem when writing items with functional constraint equal to **CF** (*Configuration*) (*Case 30565*). • Revised the usage of dynamic data sets (*Case 31536*). • The message **DefineNamedVariableListResponse** is no longer sending the encoding of the **MMSConfirmedResponsePDU** command (*Case 33518*). • Revised the algorithm for sending buffered and

VERSION	DATE	AUTHOR	COMMENTS
			unbuffered reports. Specially in case of unbuffered reports, there was a problem to send a second report and so on containing only spontaneous changes (*Case 33514*). • Revised the handling of commands to also allow using analog variables (*Case 34194*). • Revised the handling of commands (*Case 35639*). • Now the bit 2 (two) of **Time Quality** (*Clock not synchronized*) is cleared as soon as any value is written by an application (*Case 35908*). • Documentation updated (*Case 27040*). • Added the inversion of **BitStrings** processing when the name of an item is equal to "stVal" (*Case 21890*).
1.0.1	02/24/2017	M. Salvador	• First commercial 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 2014 Cool Vendor

OPC FOUNDATION MEMBER

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)