DLMS Driver

File Name	DLMS.dll
Manufacturer	IEC
Devices	SL7K and other IEC 62056 standard-compliant power meters
Protocol	IEC 62056 (Master)
Version	2.0.32
Last Update	09/02/2025
Platform	Win32
Dependencies	IOKit version 2.0 or later
Superblock Readings	No
Level	31202

Introduction

The DLMS Driver must be used for communication between **Elipse Software** systems and Itron SL7000, Itron ACE6000, Landis&Gyr E550, Landis&Gyr E650, and other compatible meters.

Driver Setup

This Driver's **[P]** configuration parameters are not used. All configurations must be performed on this Driver's properties dialog box. For more information about the **Setup**, **Serial**, **Ethernet**, **Modem**, and **RAS** tabs, please check topic **Documentation of I/O Interfaces**.

Select the **DLMS** tab to define a destination address of a DLMS meter using the **HDLC Logical Device** and **HDLC Physical Device** options.

Usually the **HDLC Logical Device** option's default value is 1 (one), while the **HDLC Physical Device** option defines the physical address set to a DLMS meter.

The **HDLC Client Address** option specifies an access profile to a meter. Its default value is 1 (one, *Client Management Process*), but can assume other values depending on the security profile required to grant access to data. For example, for Landis&Gyr meters, the **HDLC Client Address** option can have the value 32 for access with LLS security or can have the value 48 for access with HLS security.

In case a DLMS meter works with a data profile different from the MAE standard, enable the **Disable MAE compatibility mode** option.

To execute a new reading try from an eventual failure when receiving a device's response, enable the **Retry failed responses** option, also informing how many times this Driver must repeat this reading in case the failure persists by defining the **Number of retries** option. Optionally, this Driver can force resending a request command between each try by enabling the **Force resend of the request command every retry** option.

Configuring Properties

The **DLMS** tab of this Driver's properties window contains all options that can be configured by users.

In addition to this properties window, these settings can also be defined at run time in **Elipse E3**, **Elipse Power**, or **Elipse Water** applications. To do so, initialize this Driver in **Offline** mode, that is, execute an application with the **Start driver OFFLINE** option enabled, which is configured on the **Setup** tab of the properties window.

The configuration options of this Driver are described on the next table.

DLMS Driver Configuration Options

OPTION	DESCRIPTION	OFFLINE STRING	FORMAT
Meter Model	Configures the model of a meter	DLMS.MeterModel	Number. Possible values are **0**: Itron SL7000, **1**: Itron ACE6000, **2**: Landis&Gyr E550, **3**: Landis&Gyr E650, **4**: CEWE Prometer 100, **5**: Secure Apex 100 or **6**: AEC ADDAD
Name Referencing	Configures the type of address supported by a meter	DLMS.NameReferencing	Number. Possible values are **0**: Automatic, **1**: Logical Name Referencing, or **2**: Short Name Referencing
HDLC Logical Device	Configures the Logical Device Address of a meter	DLMS.DefaultSlaveLogicalAddress	Number. Possible values are in the range from 0 (zero) to 16383
HDLC Physical Device	Configures the Physical Device Address of a meter	DLMS.DefaultSlavePhysicalAddress	Number. Possible values are in the range from 0 (zero) to 16383
HDLC Client Address (read profile)	Configures a user's access profile for reading data	DLMS.HDLCClientAddress	Number. Possible values are in the range from 0 (zero) to 127
Security Mechanism (read profile)	Configures a user's access security level for reading data	DLMS.SecurityMechanism	Number. Possible values are **0**: Lowest (no authentication), **1**: LLS (password authentication), or **2**: HLS (encrypted password authentication, specific to each manufacturer)
HDLC Client Address (write profile)	Configures a user's access profile for writing data	DLMS.HDLCClientAddressWrite	Number. Possible values are in the range from 0 (zero) to 127
Security Mechanism (write profile)	Configures a user's access security level for writing data	DLMS.SecurityMechanismWrite	Number. Possible values are **0**: Lowest (no authentication), **1**: LLS (password authentication), or **2**: HLS (encrypted password authentication, specific to each manufacturer)
Retry failed responses	Enables or disables this Driver's internal function for retries after a lack of response from a meter after its time-out expires	DLMS.UseRetryFailedResponses	Number. Possible values are **0**: Disabled or **1**: Enabled
Number of Retries	Configures how many internal retries this Driver	DLMS.NumberOfRetries	Number. Possible values are in the range from 0 (zero) to

OPTION	DESCRIPTION	OFFLINE STRING	FORMAT
	performs, only if the **Retry failed responses** option is enabled		100
Force resend of the request command every retry	Enables or disables resending a request between each Driver's internal retry, only if the **Retry failed responses** option is enabled	DLMS.UseResendRequestCommand	Number. Possible values are **0**: Disabled or **1**: Enabled
Disable MAE compatibility mode	Enables or disables non-standard MAE data handling	DLMS.DisableMAEExtensions	Number. Possible values are **0**: Disabled or **1**: Enabled
Minimum time adjustment (seconds)	Configures a minimum value allowed to synchronize with a meter's clock	DLMS.TimeSync.MinAdjustSec	Number. Possible values are in the range from 0 (zero) to 3600
Maximum time adjustment (seconds)	Configures a maximum value allowed to synchronize with a meter's clock	DLMS.TimeSync.MaxAdjustSec	Number. Possible values are in the range from 0 (zero) to 86400
Protect against time adjustment across intervals (seconds)	Configures a protected value within the integration period to synchronize with a meter's clock	DLMS.TimeSync.IntervalProtectSec	Number. Possible values are in the range from 0 (zero) to 600

All offline properties must be configured via PLC Tags in **String** format, by using parameters *N1* equal to -1 (minus one), *N2* equal to 0 (zero), *N3* equal to 0 (zero), and *N4* equal to 3 (three). For more details and examples, please check topic **Documentation of I/O Interfaces**.

Tag Reference

All Tags of this Driver are configured with numeric values using *N* and *B* parameters. For **Elipse E3**, **Elipse Power**, or **Elipse Water** applications, users can manually access an object by informing its respective OBIS (*Object Identification System*) code in the **ParamItem** property.

Use the *N1* or *B1* parameter to define a physical address of a destination DLMS meter or leave this value in 0 (zero) to use the default value defined in this Driver's properties.

To send a command to a meter, use the *N2* or *B2* parameter.

List of OBIS Codes Associated to Tags

Based on the SL7K meter, OBIS (*Object Identification System*) codes associated with the Tags implemented by this Driver are described on the next tables.

Meter Tags

N2/B2	COMMAND	TYPE	OPERATION	OBIS
02	Change COSEM password	PLC	Write-only	Not used
03	Set up meter's clock time	PLC	Write-only	0-0:1.0.0
81	Current date and time	PLC	Read-only	0-0:1.0.0

N2/B2	COMMAND	TYPE	OPERATION	OBIS
31	MAE number	PLC	Read-only	1-0:0.0.0 1-0:0.0.1
32	Serial number	PLC	Read-only	0-0:96.1.0
33	CV and VT ratio	Block	Read-only	0-0:148.2.2
34	Constant of integration	PLC	Read-only	0-0:136.0.1 0-0:136.1.1
35	Threshold Cut/Sag/Swell	Block	Read-only	0-0:148.1.2
36	External firmware revision	PLC	Read-only	0-0:142.1.1
37	Internal firmware revision	PLC	Read-only	0-0:142.1.2
38	Channel parameters	Block	Read-only	0-0:136.0.2 0-0:136.1.2

Mass Memory Tags

N2/B2	COMMAND	TYPE	OPERATION	OBIS
99	Mass memory	Block	Write-only	0-0:136.0.2 0-0:136.1.2 0-0:99.128.1 0-0:99.129.1 0-0:136.0.1 0-0:136.1.1 0-0:99.1.2 0-0:99.2.2
991	MM values	Block	Read-only	Not used
992	MM status	Block	Read-only	Not used

Alarm Tags

N2/B2	COMMAND	TYPE	OPERATION	OBIS
88	Alarms	PLC	Write-only	0-0:1.0.0 0-0:99.98.0
881	AL values	Block	Read-only	Not used
882	AL status	Block	Read-only	Not used

Quality Tags

N2/B2	COMMAND	TYPE	OPERATION	OBIS
77	Quality	PLC	Write-only	0-0:148.1.2 1-0:99.10.1 1-0:99.10.2 1-0:99.10.3
771	QL values	Block	Read-only	Not used
772	QL status	Block	Read-only	Not used

Instant Values Tags

N2/B2	COMMAND	TYPE	OPERATION	OBIS
501	Phase angles	Block	Read-only	1-1:98.128.1
502	Zero sequence	Block	Read-only	1-1:98.128.2
503	Instantaneous powers	Block	Read-only	1-1:98.128.3
504	RMS values	Block	Read-only	1-1:98.128.4
505	Instantaneous power factors	Block	Read-only	1-1:98.128.5
506	Offsets	Block	Read-only	1-1:98.128.6

Manual Configuration of OBIS Codes

Manual configuration of OBIS (*Object Identification System*) codes can only be performed via **Elipse E3**, **Elipse Power**, or **Elipse Water** applications using Tag's **ParamItem** property.

To read a user-defined object's attribute manually, leave the *N2* parameter with the value 0 (zero) and fill in the OBIS code in the **ParamItem** property, using the following syntax:

```
<A>-<B>:<C>.<D>[.<E>[.<F>]][*<Attr>]
```

The **<E>** and **<F>** parameters are optional. If not informed, they assume a value of 255. The **<Attr>** attribute is optional. If not informed, such as "1-1:1.8.0", a register's value is read and the scale configured in the object is then applied. This behavior can vary depending on the object's class:

- **Class 3 (REGISTER)**: Reads the value of attribute 2 (two, **Value**) and applies the scale configured in attribute 3 (three, **UnitScale**)
- **Class 4 (EXTENDED REGISTER)**: Reads the value of attribute 2 (two, **Value**) and applies the scale configured in attribute 3 (three, **UnitScale**)
- **Class 5 (DEMAND REGISTER)**: Reads the value of attribute 2 (two, **CurrentAverageValue**) and applies the scale configured in attribute 4 (four, **UnitScale**)
- **Other Classes**: Tag returns an error, because only REGISTER classes are allowed in this mode

If the **<Attr>** parameter is informed, such as "1-1:1.8.0*3", only the value of object's attribute is read. The classes or attributes supported by this Driver are described on the next table.

List of Classes and their Attributes

CLASS	ATTRIBUTES	DESCRIPTION
Class 0 (CLASS)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
Class 1 (DATA)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Value
Class 3 (REGISTER)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Value
	Attribute 3	UnitScale (a block with two elements, the first one indicating the scale and the second one the unit)
Class 4 (EXTENDED REGISTER)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Value
	Attribute 3	UnitScale (a block with two elements, the first one indicating the scale and the second one the unit)
	Attribute 4	Status
	Attribute 5	Capture Time
Class 5 (DEMAND REGISTER)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Current Average Value
	Attribute 3	Last Average Value
	Attribute 4	UnitScale (a block with two elements, the first one indicating the scale and the second one the unit)
	Attribute 5	Status
	Attribute 6	Capture Time
	Attribute 7	Start Time Current
	Attribute 8	Period
	Attribute 9	Number Of Periods
Class 6 (REGISTER ACTIVATION)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Register Assignment
	Attribute 3	Mask List
	Attribute 4	Active Mask
Class 7 (PROFILE GENERIC)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Buffer
	Attribute 3	Capture Objects

CLASS	ATTRIBUTES	DESCRIPTION
	Attribute 4	Capture Period
	Attribute 5	Sort Method
	Attribute 6	Sort Object
	Attribute 7	Entries In Use
	Attribute 8	Profile Entries
Class 8 (CLOCK)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Time
	Attribute 3	Time Zone
	Attribute 4	Clock Status
	Attribute 5	Daylight Savings Begin
	Attribute 6	Daylight Savings End
	Attribute 7	Daylight Savings Deviation
	Attribute 8	Daylight Savings Enabled
	Attribute 9	Clock Base
Class 9 (SCRIPT TABLE)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Scripts
Class 10 (SCHEDULE)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Entries
Class 12 (ASSOCIATION SN)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Object List
	Attribute 8	Reply HLS Challenge Service
Class 15 (ASSOCIATION LN)	Attribute 1	LogicalName (a **String** containing the OBIS code of this object)
	Attribute 2	Object List
	Attribute 3	Associated Partners ID
	Attribute 4	Application Context Name
	Attribute 5	DLMS Context Info
	Attribute 6	Authentication Mechanism Name
	Attribute 7	LLS Secret
	Attribute 8	Association Status

The other supported classes are described next:

- Class 17 (SAP ASSIGNMENT)
- Class 19 (IEC LOCAL PORT SETUP)

- Class 20 (ACTIVITY CALENDAR)
- Class 21 (REGISTER MONITOR)
- Class 22 (SINGLE ACTION SCHEDULE)
- Class 23 (IEC HDLC SETUP)
- Class 24 (IEC TWISTED PAIR SETUP)
- Class 26 (UTILITY TABLES)
- Class 27 (PSTN MODEM CONFIGURATION)
- Class 28 (PSTN AUTO ANSWER)
- Class 29 (PSTN AUTO DIAL)

Please check the documentation of DLMS protocol for a full list of attributes.

Meter Tags

This section contains information about Tags with general data about a meter.

Change COSEM Password

Write-Only

N1	Physical address of a device
N2	2 (two)
N3	Type of access profile
N4	Type of authentication

A PLC Tag in **Text** format to change a meter's COSEM password.

The *N3* parameter of this Tag selects a type of access profile. Possible values are **0**: Access profile for data reading or **1**: Access profile for data writing.

The *N4* parameter of this Tag selects an access method. Possible values are **0**: Defining a password without authentication (MAE standard) or **1**: Defining a password with authentication.

NOTE

Configuring the *N4* parameter with value 1 (one) only works if the **Disable MAE compatibility mode** option is enabled to ignore MAE standard.

Set Up Meter's Clock Time

Write-Only

N1	Physical address of a device
N2	3 (three)
N3	0 (zero)
N4	0 (zero)

A PLC Tag in **Date and Time** format to set up a meter's clock.

- **OBIS**: 0-0:1.0.0
- **ClassId**: 8 (eight)
- **AttributeId**: 2 (two, current date and time)

Synchronize Meter's Clock Time

Write-Only

N1	Physical address of a device
N2	4 (four)
N3	0 (zero)
N4	0 (zero)

A PLC Tag in **Number** format to synchronize a meter's clock with the computer's local time, with the option to inform a meter's location (timezone), in minutes, referring to GMT, if a meter's model has this attribute.

For example, if a meter is located in Brazil, its timezone must be set to GMT -03:00 by typing -180 in this Tag, which is the equivalent value in minutes of the equivalent timezone (GMT -3 = -3 * 60 = -180).

Meter's Clock Syncing Properties

This Tag to synchronize a meter's clock time works with the following options, defined in this Driver's properties window.

- **Minimum time adjustment (seconds)**: Defines a minimum adjustment to be performed by this Driver, in seconds. Adjustments less than this time are not performed, that is, this Driver returns success when writing, but does not perform that adjustment
- **Maximum time adjustment (seconds)**: Defines a maximum adjustment to be performed by this Driver, in seconds. Adjustments greater than this time are reduced to this maximum adjustment
- **Protect against time adjustment across intervals (seconds)**: Defines a value to enable a protection against clock settings that exceed the current collection interval. The indicated value determines the number of seconds that are protected at the end of the current collection interval

For example, if the current meter's time is 09:13:35 and the integration period is 15 minutes, or 900 seconds, this is the scenario for a protected clock adjustment:

- **Current collection interval**: 09:00:00 to 09:15:00
- **Adjustment's protection value**: 60 seconds

Based on this protection parameter, the maximum allowable clock adjustment is from 09:00:00 to 09:14:00, that is, 09:15:00 minus 60 seconds. Adjustments that exceed these limits are truncated so that they remain within that limit.

Current Date and Time

Read-Only

N1	Physical address of a device
N2	81
N3	0 (zero)
N4	0 (zero)

A PLC Tag in **Date and Time** format to read a meter's clock.

- **OBIS**: 0-0:1.0.0
- **ClassId**: 8 (eight)
- **AttributeId**: 2 (two, current date and time)

MAE Number

Read-Only

N1	Physical address of a device
N2	31
N3	0 (zero)
N4	0 (zero)

A PLC Tag in **Text** format to read an alphanumeric value containing 14 characters of a MAE identifier.

NOTE

MAE is a short for *Mercado Atacadista de Energia* (*Power Wholesale Market*), currently CCEE (*Câmara de Comercialização de Energia Elétrica* or *Power Trade Chamber*).

- **OBIS**: 1-0:0.0.0
- **ClassId**: 1 (one)

- **AttributeId**: 2 (two, *Utility Id 1 Parameters*)

- **OBIS**: 1-0:0.0.1
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Utility Id 2 Parameters*)

Serial Number

Read-Only

N1	Physical address of a device
N2	32
N3	0 (zero)
N4	0 (zero)

A PLC Tag in **Text** format to read a serial identification number from a meter.

- **OBIS**: 0-0:96.1.0
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Serial Number Parameters*)

CT and VT Ratio

Read-Only

B1	Physical address of a device
B2	33
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format, containing the next Elements:

- **TC**: Ratio between the configured voltage and current for a meter
- **TP**: Ratio between the configured voltage and power for a meter
- TC numerator
- TC denominator
- TP numerator
- TP denominator

- **OBIS**: 0-0:148.2.2
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Secondary Metrology Installation Parameters*)

Number of Integration Constant

Read-Only

N1	Physical address of a device
N2	34
N3	Profile ID
N4	0 (zero)

A PLC Tag in **Number** format to indicate the integration time configured in a meter, in minutes. Default value is 5 (five) minutes.

- **OBIS**: 0-0:136.0.1
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Load Profiling 1 Parameters*)

- **OBIS**: 0-0:136.1.1
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Load Profiling 2 Parameters*)

Threshold Cut/Sag/Swell

Read-Only

B1	Physical address of a device
B2	35
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the next Elements:

- *Volt Cut High Threshold*
- *Volt Cut Low Threshold*

- *Volt Sag High Threshold*
- *Volt Sag Low Threshold*
- *Volt Swell High Threshold*
- *Volt Swell Low Threshold*

- **OBIS**: 0-0:148.1.2
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Qualimetry Voltage Threshold Parameters*)

External Firmware Revision

Read-Only

N1	Physical address of a device
N2	36
N3	0 (zero)
N4	0 (zero)

A PLC Tag in **Text** format to read in **XX.YY** format the firmware's external version, in which **XX** indicates a major version number and **YY** indicates a minor version number.

- **OBIS**: 0-0:142.1.1
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *External Firmware Id Parameters*)

Internal Firmware Revision

Read-Only

N1	Physical address of a device
N2	37
N3	0 (zero)
N4	0 (zero)

A PLC Tag in **Text** format to read in **XX.YY** format a firmware's internal version, in which **XX** indicates a major version number and **YY** indicates a minor version number.

- **OBIS**: 0-0:142.1.2
- **ClassId**: 1 (one)

- **AttributeId**: 2 (two, *Internal Firmware Id Parameters*)

Channel Parameters

Read-Only

B1	Physical address of a device
B2	38
B3	Profile ID
B4	0 (zero)

A Block Tag in **Number** format, containing the next Elements:

- *Channel Number*
- *QuantityId*
- *Scaler*
- *Unit*
- *Excess Energy Channel*
- *Working Mode*
- *Fluid*
- *Quantity Type*

For more information about channel parameters, please check the **COSEM LoadProfilenChannelParameters** object on a meter's documentation.

For a description of all codes returned by this Block Tag, please check a meter's interface spreadsheet and click the **quant_listing** (*QuantityId*) and **units_listing** (*Unit*) spreadsheets.

NOTE

Use a Block Tag's **OnRead** event to receive these values in **List** mode.

- **OBIS**: 0-0:136.0.2
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Load Profile 1 Channel Parameters*)

- **OBIS**: 0-0:136.1.2
- **ClassId**: 2 (two)
- **AttributeId**: 2 (two, *Load Profile 2 Channel Parameters*)

Query a Description of the Last Error Occurred

Read-Only

N1	Physical address of a device
N2	1001
N3	0 (zero)
N4	0 (zero)

A PLC Tag in **Text** format to return a description of the most recent error that occurred with a meter.

Mass Memory Tags

This section contains information about Tags executing a process of collecting memory mass for a user-defined range of dates.

Mass Memory

Write-Only

B1	Physical address of a device
B2	99
B3	Profile ID
B4	0 (zero)

A Block Tag in **Date and Time** format to define a time interval to collect mass memory, containing the next Elements:

- **Initial Date**: The starting date of a data request
- **Final Date**: The ending date of a data request

NOTE

If the initial date is null or greater than the current date, data is collected until the most recent period of a meter.

- **OBIS**: 0-0:136.0.2
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Load Profile 1 Channel Parameters*)

- **OBIS**: 0-0:136.1.2
- **ClassId**: 1 (one)

- **AttributeId**: 2 (two, *Load Profile 2 Channel Parameters*)

- **OBIS**: 0-0:99.128.1
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, *Load Profile 1 Information*)

- **OBIS**: 0-0:99.129.1
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, *Load Profile 2 Information*)

- **OBIS**: 0-0:136.0.1
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Load Profiling 1 Parameters*)

- **OBIS**: 0-0:136.1.1
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, *Load Profiling 2 Parameters*)

- **OBIS**: 0-0:99.1.2
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, *Load Profile 1 Condensed*)

- **OBIS**: 0-0:99.2.2
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, *Load Profile 2 Condensed*)

MM Values

Read-Only

B1	Physical address of a device
B2	991
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format to define a reading of mass memory data according to a user-defined interval. The timestamp defines an integration time. This Tag contains the next Elements:

- Values for collecting mass memory on ProfileId 1: (MAE standard)
  - ProfileID (returns a value of one)
  - Supplied Active Power (KWh)
  - Received Active Power (KWh)
  - Supplied Reactive Power (KVARh)
  - Received Reactive Power (KVARh)
  - Voltage Phase A (V)
  - Voltage Phase B (V)
  - Voltage Phase C (V)

NOTE

Values referring to quantities of Supplied Active or Reactive Power and Received Active or Reactive Power are returned with their values divided by 1000 by this Driver, except if the **Disable MAE compatibility mode** option is enabled to bypass the MAE standard.

- Values for collecting mass memory on ProfileId 2: (MAE standard)
  - ProfileID (returns a value of two)
  - Current Phase A (A)
  - Current Phase B (A)
  - Current Phase C (A)
  - Power Factor
  - Frequency (Hz)
  - External 1 (one)
  - External 2 (two)

MM Status

Read-Only

B1	Physical address of a device
B2	992
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format to indicate data status, containing the next Elements:

- **Status**: Indicates how many subtasks are still active before this operation ends. When status is equal to 0 (zero), all data in a meter for the requested time interval were read and are available in the **MM Values** Tag
- Active thread
- Number of Elements in the buffer
- Total number of readings

Alarm Tags

This section contains information about Tags executing a process of collecting alarm events.

Alarms

Write-Only

N1	Physical address of a device
N2	88
N3	0 (zero)
N4	0 (zero)

A PLC Tag that executes a writing command to start collecting alarm events.

- **OBIS**: 0-0:1.0.0
- **ClassId**: 8 (eight)
- **AttributeId**: 2 (two, current date and time)

- **OBIS**: 0-0:99.98.0
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, Logbook)

AL Values

Read-Only

B1	Physical address of a device
B2	881
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format. The timestamp defines the integration time. It contains the next Elements:

- Event ID
- Event parameter
- Event sequence
- Event status

AL Status

Read-Only

B1	Physical address of a device
B2	882
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format to indicate data status, containing the next Elements:

- **Status**: Indicates how many subtasks are still active before this operation ends. When status is equal to 0 (zero), all data in a meter for the requested time interval were read and are available in the **AL Values** Tag
- Active thread
- Number of Elements in the buffer
- Total number of readings

Power Quality Tags

This section contains information about Tags executing a process of collecting quality values.

Quality

Write-Only

N1	Physical address of a device
N2	77
N3	0 (zero)
N4	0 (zero)

A PLC Tag that executes a writing command to start collecting quality values.

- **OBIS**: 0-0:148.1.2
- **ClassId**: 1 (one)
- **AttributeId**: 2 (two, Qualimetry Voltage Threshold Parameters)

- **OBIS**: 1-0:99.10.1
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, Sag Elements)

- **OBIS**: 1-0:99.10.2
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, Swell Elements)

- **OBIS**: 1-0:99.10.3
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, Cut Elements)

QL Values

Read-Only

B1	Physical address of a device
B2	771
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format and the timestamp indicates the integration time, containing the next Elements:

- Type
- Phase number
- Duration
- Magnitude
- Log Book ID
- Maximum Sag
- Minimum Swell

QL Status

Read-Only

B1	Physical address of a device
B2	772
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format indicating data status, containing the next Elements:

- **Status**: Indicates how many subtasks are still active before this operation ends. When status is equal to 0 (zero), all data in a meter for the requested time interval were read and are available in the **QL Values** Tag
- Active thread
- Number of Elements in the buffer
- Total number of readings

Snapshot Values Tags

This section contains information about Tags for reading snapshot values from a meter.

Itron Meters

Use a Block Tag reading to query any of the many snapshot value options provided by Itron meters.

Phase Angles

Read-Only

B1	Physical address of a device
B2	501
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the next Elements:

- Angle U1 I1
- Angle U2 I2
- Angle U3 I3
- Angle U1 U2

- Angle U1 U3
- Angle U2 U3

- **OBIS**: 1-1:98.128.1
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, All Phase Angles)

Zero Sequence

Read-Only

B1	Physical address of a device
B2	502
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the next Elements:

- Zero Sequence U
- Zero Sequence I

- **OBIS**: 1-1:98.128.2
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, Zero Sequence)

Snapshot Powers

Read-Only

B1	Physical address of a device
B2	503
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the next Elements:

- Import Active Power Phase1
- Import Active Power Phase2

- Import Active Power Phase3
- Import Active Power Aggregate
- Export Active Power Phase1
- Export Active Power Phase2
- Export Active Power Phase3
- Export Active Power Aggregate
- Import Reactive Power Phase1
- Import Reactive Power Phase2
- Import Reactive Power Phase3
- Import Reactive Power Aggregate
- Export Reactive Power Phase1
- Export Reactive Power Phase2
- Export Reactive Power Phase3
- Export Reactive Power Aggregate
- Reactive Power Aggregate Q1
- Reactive Power Aggregate Q2
- Reactive Power Aggregate Q3
- Reactive Power Aggregate Q4
- Import Apparent Power Phase 1
- Import Apparent Power Phase 2
- Import Apparent Power Phase 3
- Import Apparent Power Aggregate
- Export Apparent Power Phase 1
- Export Apparent Power Phase 2
- Export Apparent Power Phase 3
- Export Apparent Power Aggregate

- **OBIS**: 1-1:98.128.3
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, All Snapshot Powers)

RMS Values

Read-Only

B1	Physical address of a device
B2	504
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the next Elements:

- RMS Current Value Phase 1
- RMS Current Value Phase 2
- RMS Current Value Phase 3
- RMS Voltage Value Phase 1
- RMS Voltage Value Phase 2
- RMS Voltage Value Phase 3

- **OBIS**: 1-1:98.128.4
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, All RMS Values)

Snapshot Power Factors

Read-Only

B1	Physical address of a device
B2	505
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the next Elements:

- Snapshot Power Factor Phase 1
- Snapshot Power Factor Phase 2
- Snapshot Power Factor Phase 3
- Snapshot Power Factor Aggregate

- **OBIS**: 1-1:98.128.5

- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, All Snapshot Power Factors)

Offsets

Read-Only

B1	Physical address of a device
B2	506
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the next Elements:

- Offset Current Phase 1
- Offset Current Phase 2
- Offset Current Phase 3
- Offset Voltage Phase 1
- Offset Voltage Phase 2
- Offset Voltage Phase 3

- **OBIS**: 1-1:98.128.6
- **ClassId**: 7 (seven)
- **AttributeId**: 2 (two, All Offsets)

Landis+Gyr Meters

Use a Block Tag reading to query any of the many snapshot value options provided by Landis+Gyr meters.

Phase Angles

Read-Only

B1	Physical address of a device
B2	501
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Phase Angles Tag

ELEMENT	QUANTITY	OBIS CODE
1	Angle U1 I1	1-1:81.7.40
2	Angle U2 I2	1-1:81.7.51
3	Angle U3 I3	1-1:81.7.62
4	Angle U2 U1	1-1:81.7.10
5	Angle U1 U3	1-1:81.7.2
6	Angle U3 U2	1-1:81.7.21

Zero Sequence

Read-Only

B1	Physical address of a device
B2	502
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Zero Sequence Tag

ELEMENT	QUANTITY	OBIS CODE
1	Zero Sequence U	1-1:91.7.0
2	Zero Sequence I	1-1:92.7.0

Snapshot Powers

Read-Only

B1	Physical address of a device
B2	503
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Snapshot Powers Tag

ELEMENT	QUANTITY	OBIS CODE
1	Import Active Power Phase 1	1-1:21.7.0
2	Import Active Power Phase 2	1-1:41.7.0
3	Import Active Power Phase 3	1-1:61.7.0
4	Import Active Power Aggregate	1-1:1.29.1
5	Export Active Power Phase 1	1-1:22.7.0
6	Export Active Power Phase 2	1-1:42.7.0
7	Export Active Power Phase 3	1-1:62.7.0
8	Export Active Power Aggregate	1-1:2.29.4
9	Import Reactive Power Phase 1	1-1:23.7.0
10	Import Reactive Power Phase 2	1-1:43.7.0
11	Import Reactive Power Phase 3	1-1:63.7.0
12	Import Reactive Power Aggregate	1-1:3.29.1
13	Export Reactive Power Phase 1	1-1:24.7.0
14	Export Reactive Power Phase 2	1-1:44.7.0
15	Export Reactive Power Phase 3	1-1:64.7.0
16	Export Reactive Power Aggregate	1-1:6.29.4
17	Reactive Power Aggregate Q1	1-1:5.7.0
18	Reactive Power Aggregate Q2	1-1:6.7.0
19	Reactive Power Aggregate Q3	1-1:7.7.0
20	Reactive Power Aggregate Q4	1-1:8.7.0
21	Import Apparent Power Phase 1	1-1:29.7.0
22	Import Apparent Power Phase 2	1-1:49.7.0
23	Import Apparent Power Phase 3	1-1:69.7.0
24	Import Apparent Power Aggregate	1-1:9.7.0
25	Export Apparent Power Phase 1	1-1:30.7.0
26	Export Apparent Power Phase 2	1-1:50.7.0
27	Export Apparent Power Phase 3	1-1:70.7.0
28	Export Apparent Power Aggregate	1-1:10.7.0

RMS Values

Read-Only

B1	Physical address of a device
B2	504
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the RMS Values Tag

ELEMENT	QUANTITY	OBIS CODE
1	RMS Current Value Phase 1	1-1:31.7.0
2	RMS Current Value Phase 2	1-1:51.7.0
3	RMS Current Value Phase 3	1-1:71.7.0
4	RMS Voltage Value Phase 1	1-1:32.7.0
5	RMS Voltage Value Phase 2	1-1:52.7.0
6	RMS Voltage Value Phase 3	1-1:72.7.0

Snapshot Power Factors

Read-Only

B1	Physical address of a device
B2	505
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Snapshot Power Factors Tag

ELEMENT	QUANTITY	OBIS CODE
1	Snapshot Power Factor Phase 1	1-1:33.7.0
2	Snapshot Power Factor Phase 2	1-1:53.7.0
3	Snapshot Power Factor Phase 3	1-1:63.7.0
4	Snapshot Power Factor Aggregate	1-1:13.7.0

Offsets

Read-Only

B1	Physical address of a device
B2	506
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Offsets Tag

ELEMENT	QUANTITY	OBIS CODE
1	Offset Current Phase 1	1-1:96.50.1
2	Offset Current Phase 2	1-1:96.50.2
3	Offset Current Phase 3	1-1:96.50.3
4	Offset Voltage Phase 1	1-1:96.50.4
5	Offset Voltage Phase 2	1-1:96.50.5
6	Offset Voltage Phase 3	1-1:96.50.6

CEWE Meters

Use a Block Tag reading to query any of the many snapshot value options provided by CEWE meters.

Phase Angles

Read-Only

B1	Physical address of a device
B2	501
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Phase Angles Tag

ELEMENT	QUANTITY	OBIS CODE
1	Voltage 1 to Current 1 Angle	1-0:81.7.4
2	Voltage 2 to Current 2 Angle	1-0:81.7.15
3	Voltage 3 to Current 3 Angle	1-0:81.7.26
4	Voltage Angle 1-2	1-0:81.7.1
5	Voltage Angle 2-3	1-0:81.7.12

ELEMENT	QUANTITY	OBIS CODE
6	Voltage Angle 3-1	1-0:81.7.20

Snapshot Powers

Read-Only

B1	Physical address of a device
B2	503
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Snapshot Powers Tag

ELEMENT	QUANTITY	OBIS CODE
1	Active Power (Total) - L1	1-0:21.7.0
2	Active Power (Total) - L2	1-0:41.7.0
3	Active Power (Total) - L3	1-0:61.7.0
4	Active Power (Total) - All phases	1-0:1.7.0
5	Reactive Power - L1	1-0:23.7.0
6	Reactive Power - L2	1-0:43.7.0
7	Reactive Power - L3	1-0:63.7.0
8	Reactive Power - All phases	1-0:3.7.0
9	Apparent Power - L1	1-0:29.7.0
10	Apparent Power - L2	1-0:49.7.0
11	Apparent Power - L3	1-0:69.7.0
12	Apparent Power - All phases	1-0:9.7.0
13	Active Import Total (Q1+Q4) Energy	1-0:1.8.0
14	Active Export Total (Q2+Q3) Energy	1-0:2.8.0
15	Reactive Import (Q1+Q2) Energy	1-0:3.8.0
16	Reactive Export (Q3+Q4) Energy	1-0:4.8.0
17	Reactive Q1 Energy	1-0:5.8.0
18	Reactive Q2 Energy	1-0:6.8.0
19	Reactive Q3 Energy	1-0:7.8.0
20	Reactive Q4 Energy	1-0:8.8.0
21	Apparent While Active Import (Q1+Q4) Energy	1-0:9.8.0

ELEMENT	QUANTITY	OBIS CODE
22	Apparent While Active Export (Q2+Q3) Energy	1-0:10.8.0
23	Active Import Total (Q1+Q4) Energy - L1	1-0:21.8.0
24	Active Import Total (Q1+Q4) Energy - L2	1-0:41.8.0
25	Active Import Total (Q1+Q4) Energy - L3	1-0:61.8.0
26	Active Export Total (Q2+Q3) Energy - L1	1-0:22.8.0
27	Active Export Total (Q2+Q3) Energy - L2	1-0:42.8.0
28	Active Export Total (Q2+Q3) Energy - L3	1-0:62.8.0

RMS Values

Read-Only

B1	Physical address of a device
B2	504
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the RMS Values Tag

ELEMENT	QUANTITY	OBIS CODE
1	Line Current - L1	1-0:31.7.0
2	Line Current - L2	1-0:51.7.0
3	Line Current - L3	1-0:71.7.0
4	Average Line Current - All phases	1-0:11.7.0
5	Phase Voltage - L1	1-0:32.7.0
6	Phase Voltage - L2	1-0:52.7.0
7	Phase Voltage - L3	1-0:72.7.0
8	Average Phase Voltages - All phases	1-0:12.7.0

Snapshot Power Factors

Read-Only

B1	Physical address of a device
B2	505
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Snapshot Power Factors Tag

ELEMENT	QUANTITY	OBIS CODE
1	Power Factor - L1	1-0:33.7.0
2	Power Factor - L2	1-0:53.7.0
3	Power Factor - L3	1-0:73.7.0
4	Power Factor - All phases	1-0:13.7.0

Secure Meters

Use a Block Tag reading to query any of the many snapshot value options provided by Secure meters.

Phase Angles

Read-Only

B1	Physical address of a device
B2	501
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Phase Angles Tag

ELEMENT	QUANTITY	OBIS CODE
1	R - Current Phase Angle	1-0:81.7.4
2	Y - Current Phase Angle	1-0:81.7.15
3	B - Current Phase Angle	1-0:81.7.26
4	RY - Voltage Phase Angle	1-0:81.7.1
5	YB - Voltage Phase Angle	1-0:81.7.12
6	BR - Voltage Phase Angle	1-0:81.7.20

Snapshot Powers

Read-Only

B1	Physical address of a device
B2	503
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Snapshot Powers Tag

ELEMENT	QUANTITY	OBIS CODE
1	R - Active Power (Total)	1-0:21.7.0
2	Y - Active Power (Total)	1-0:41.7.0
3	B - Active Power (Total)	1-0:61.7.0
4	System Active Power (Total)	1-0:1.7.0
5	R - Reactive Power	1-0:23.7.0
6	Y - Reactive Power	1-0:43.7.0
7	B - Reactive Power	1-0:63.7.0
8	System Reactive Power	1-0:3.7.0
9	R - Apparent Power	1-0:29.7.0
10	Y - Apparent Power	1-0:49.7.0
11	B - Apparent Power	1-0:69.7.0
12	System Apparent Power	1-0:9.7.0
13	All Phase Active Import Total (Q1+Q4) Energy	1-0:1.8.0
14	All Phase Active Export Total (Q2+Q3) Energy	1-0:2.8.0
15	All Phase Reactive Import (Q1+Q2) Energy	1-0:3.8.0
16	All Phase Reactive Export (Q3+Q4) Energy	1-0:4.8.0
17	All Phase Reactive Q1 Energy	1-0:5.8.0
18	All Phase Reactive Q2 Energy	1-0:6.8.0
19	All Phase Reactive Q3 Energy	1-0:7.8.0
20	All Phase Reactive Q4 Energy	1-0:8.8.0
21	All Phase Apparent While Active Import (Q1+Q4) Energy	1-0:9.8.0

ELEMENT	QUANTITY	OBIS CODE
22	All Phase Apparent While Active Export (Q2+Q3) Energy	1-0:10.8.0
23	R - Active Import Total (Q1+Q4) Energy	1-0:21.8.0
24	Y - Active Import Total (Q1+Q4) Energy	1-0:41.8.0
25	B - Active Import Total (Q1+Q4) Energy	1-0:61.8.0
26	R - Active Export Total (Q2+Q3) Energy	1-0:22.8.0
27	Y - Active Export Total (Q2+Q3) Energy	1-0:42.8.0
28	B - Active Export Total (Q2+Q3) Energy	1-0:62.8.0

RMS Values

Read-Only

B1	Physical address of a device
B2	504
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the RMS Values Tag

ELEMENT	QUANTITY	OBIS CODE
1	R - Line Current	1-0:31.7.0
2	Y - Line Current	1-0:51.7.0
3	B - Line Current	1-0:71.7.0
4	Average Line Currents	1-0:11.7.0
5	R - Phase Voltage	1-0:32.7.0
6	Y - Phase Voltage	1-0:52.7.0
7	B - Phase Voltage	1-0:72.7.0
8	Average Phase Voltages	1-0:12.7.0

Snapshot Power Factors

Read-Only

B1	Physical address of a device
B2	505
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Snapshot Power Factors Tag

ELEMENT	QUANTITY	OBIS CODE
1	R - Power Factor	1-0:33.7.0
2	Y - Power Factor	1-0:53.7.0
3	B - Power Factor	1-0:73.7.0
4	System Power Factor	1-0:13.7.0

AEC Meters

Use a Block Tag reading to query any of the many snapshot value options provided by AEC meters.

Zero Sequence

Read-Only

B1	Physical address of a device
B2	502
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Zero Sequence Tag

ELEMENT	QUANTITY	OBIS CODE
1	INST_NEUTRAL_CURRENT	1-0:91.7.0

Snapshot Powers

Read-Only

B1	Physical address of a device
B2	503
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Snapshot Powers Tag

ELEMENT	QUANTITY	OBIS CODE
1	INST_L1_PWR_PACT	1-0:21.7.0
2	INST_L2_PWR_PACT	1-0:41.7.0
3	INST_L3_PWR_PACT	1-0:61.7.0
4	INST_PWR_PACT	1-0:1.7.0
5	INST_L1_PWR_NACT	1-0:22.7.0
6	INST_L2_PWR_NACT	1-0:42.7.0
7	INST_L3_PWR_NACT	1-0:62.7.0
8	INST_PWR_NACT	1-0:2.7.0
9	INST_L1_PWR_PREAC	1-0:23.7.0
10	INST_L2_PWR_PREAC	1-0:43.7.0
11	INST_L3_PWR_PREAC	1-0:63.7.0
12	INST_PWR_PREAC	1-0:3.7.0
13	INST_L1_PWR_NREAC	1-0:24.7.0
14	INST_L2_PWR_NREAC	1-0:44.7.0
15	INST_L3_PWR_NREAC	1-0:64.7.0
16	INST_PWR_NREAC	1-0:4.7.0
17	INST_L1_PWR_Q1	1-0:25.7.0
18	INST_L2_PWR_Q1	1-0:45.7.0
19	INST_L3_PWR_Q1	1-0:65.7.0
20	INST_PWR_Q1	1-0:5.7.0
21	INST_L1_PWR_Q2	1-0:26.7.0
22	INST_L2_PWR_Q2	1-0:46.7.0
23	INST_L3_PWR_Q2	1-0:66.7.0
24	INST_PWR_Q2	1-0:6.7.0
25	INST_L1_PWR_Q3	1-0:27.7.0
26	INST_L2_PWR_Q3	1-0:47.7.0
27	INST_L3_PWR_Q3	1-0:67.7.0

ELEMENT	QUANTITY	OBIS CODE
28	INST_PWR_Q3	1-0:7.7.0
29	INST_L1_PWR_Q4	1-0:28.7.0
30	INST_L2_PWR_Q4	1-0:48.7.0
31	INST_L3_PWR_Q4	1-0:68.7.0
32	INST_PWR_Q4	1-0:8.7.0
33	INST_L1_PWR_PAPP	1-0:29.7.0
34	INST_L2_PWR_PAPP	1-0:49.7.0
35	INST_L3_PWR_PAPP	1-0:69.7.0
36	INST_PWR_PAPP	1-0:9.7.0
37	INST_L1_PWR_NAPP	1-0:30.7.0
38	INST_L2_PWR_NAPP	1-0:50.7.0
39	INST_L3_PWR_NAPP	1-0:70.7.0
40	INST_PWR_NAPP	1-0:10.7.0

RMS Values

Read-Only

B1	Physical address of a device
B2	504
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the RMS Values Tag

ELEMENT	QUANTITY	OBIS CODE
1	INST_L1_CURRENT	1-0:31.7.0
2	INST_L2_CURRENT	1-0:51.7.0
3	INST_L3_CURRENT	1-0:71.7.0
4	INST_L1_MAX_CURRENT	1-0:31.6.0
5	INST_L2_MAX_CURRENT	1-0:51.6.0
6	INST_L3_MAX_CURRENT	1-0:71.6.0
7	INST_L1_VOLTAGE	1-0:32.7.0
8	INST_L2_VOLTAGE	1-0:52.7.0
9	INST_L3_VOLTAGE	1-0:72.7.0
10	INST_L1_MAX_VOLTAGE	1-0:32.6.0

ELEMENT	QUANTITY	OBIS CODE
11	INST_L2_MAX_VOLTAGE	1-0:52.6.0
12	INST_L3_MAX_VOLTAGE	1-0:72.6.0

Snapshot Power Factors

Read-Only

B1	Physical address of a device
B2	505
B3	0 (zero)
B4	0 (zero)

A Block Tag in **Number** format containing the Elements described on the next table.

Elements of the Snapshot Power Factors Tag

ELEMENT	QUANTITY	OBIS CODE
1	INST_L1_PF	1-0:33.7.0
2	INST_L2_PF	1-0:53.7.0
3	INST_L3_PF	1-0:73.7.0
4	INST_TOTAL_PF	1-0:13.7.0

Documentation of I/O Interfaces

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

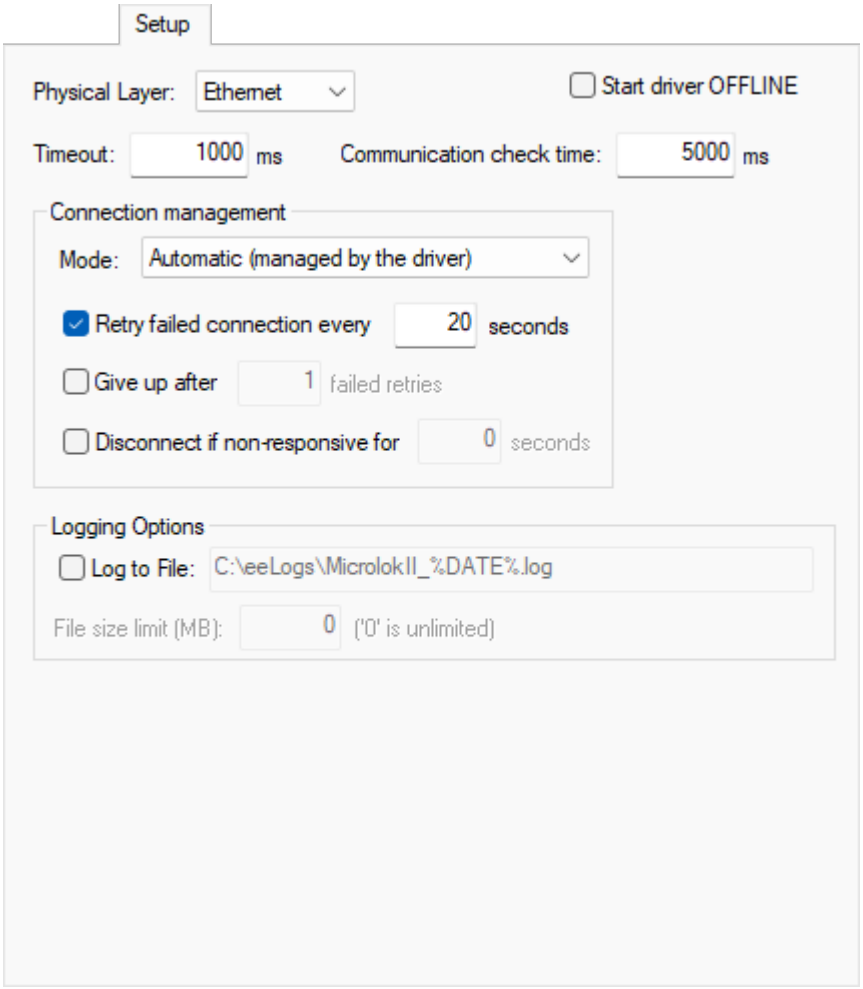
Configuration Dialog Box

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

OPTION	DESCRIPTION
	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

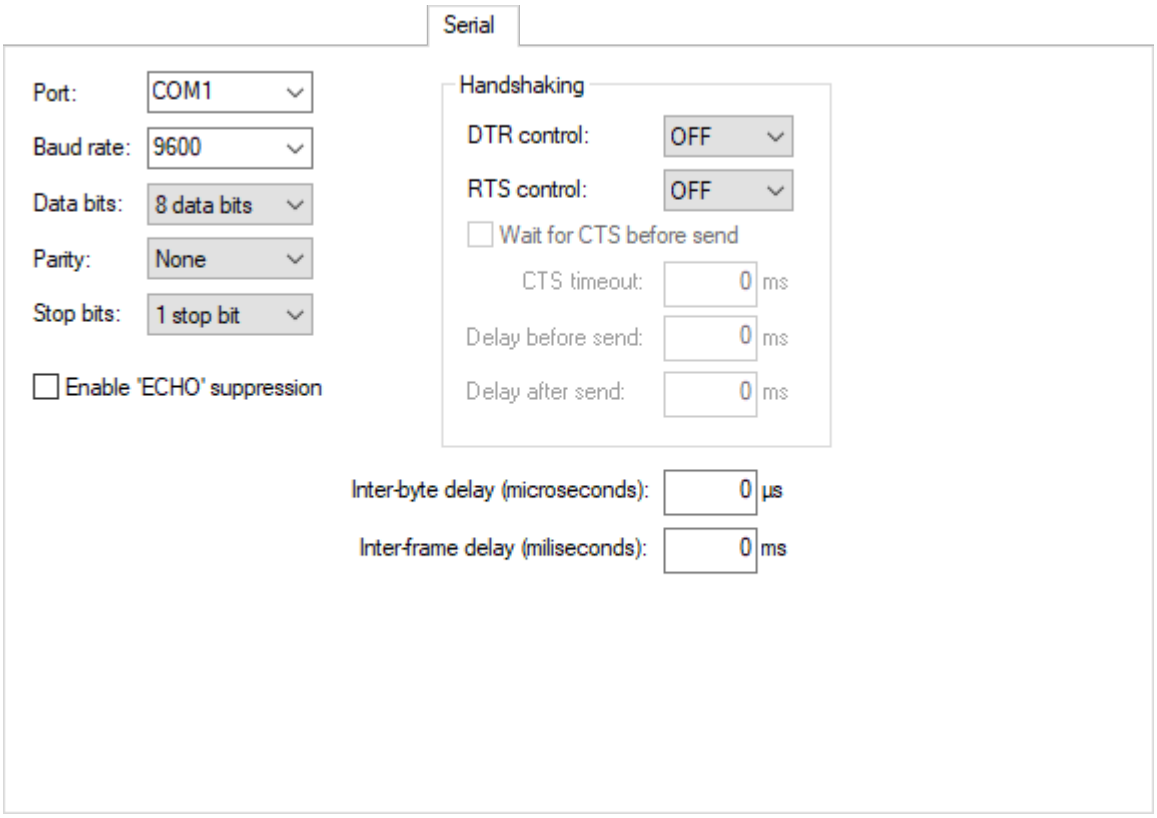
OPTION	DESCRIPTION
Mode	Selects a management mode of a connection. Selecting the **Automatic** option allows a Driver to manage the connection automatically, as specified in the next options. Selecting the **Manual** option allows an application to fully manage a connection
Retry failed connection every ... 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

Serial Tab

Use this tab to configure parameters for a **Serial** Interface.



Serial tab

General options on the Serial tab

OPTION	DESCRIPTION
Port	Select a serial port on the list, from **COM1** to **COM4**, or type the name of a serial port in the format **COM*n***, such as "COM15". When typing the name of a serial port manually, the dialog box only accepts names of serial ports starting with the expression "COM"
Baud rate	Select a baud rate on the list (**1200**, **2400**, **4800**, **9600**, **19200**, **38400**, **57600**, or **115200**) or type a baud rate, such as 600
Data bits	Select 7 (seven) or 8 (eight) data bits on the list
Parity	Select a parity on the list. The available options are **None**, **Even**, **Odd**, **Mark**, or **List**
Stop bits	Select the number of stop bits on the list. The available options are **1**, **1.5**, or **2** stop bits
Enable 'ECHO' supression	Enable this option to remove the echo received after the I/O Interface sends data via serial port. If this echo is not equal to the bytes just sent, then the I/O Interface aborts communication
Inter-byte delay (microseconds)	Defines a delay between each byte transmitted by the I/O Interface, in millionths of a second, that is, 1000000 is equal to a second. This option must be used with small delays of less than a millisecond
Inter-frame delay (milliseconds)	Defines a delay between packets sent or received by the I/O Interface, in thousandths of a second, that is, 1000 is equal to a second. This delay is applied if the I/O Interface

OPTION	DESCRIPTION
	sends two consecutive packets, or between a received packet and the next sending

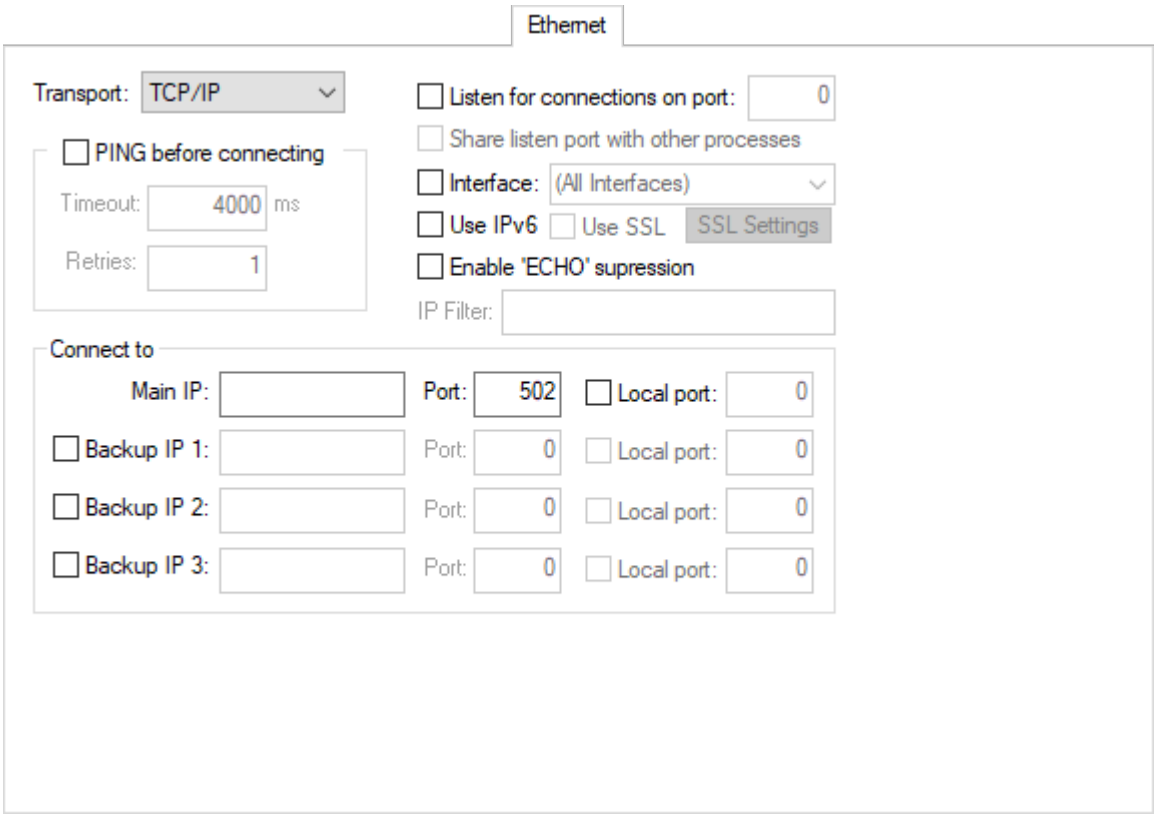
The **Handshaking** group configures the usage of **RTS**, **CTS**, and **DTR** signals in the handshaking process, that is, it controls when data can be sent or received via serial line. Most of the time, configuring the **DTR control** option to **ON** and the **RTS control** option to **Toggle** works with **RS232**-type serial lines as well as with **RS485**-type serial lines.

Available options on the Handshaking group

OPTION	DESCRIPTION
DTR control	Select the value **ON** to keep the **DTR** signal always on while the serial port is open. Select the value **OFF** to turn the **DTR** signal off while the serial port is open. Some devices require the **DTR** signal always on to allow communication
RTS control	Select the value **ON** to keep the **RTS** signal always on while the serial port is open. Select the value **OFF** to turn the **RTS** signal off while the serial port is open. Select the value **Toggle** to turn the **RTS** signal on while sending bytes via serial port and turn it off when not sending bytes, therefore enabling the reception
Wait for CTS before send	Available only when the **RTS control** option is configured with the value **Toggle**. Use this option to force a Driver to check the **CTS** signal before sending bytes via serial port, after turning the **RTS** signal on. In this mode, the **CTS** signal is handled as a permission flag for sending
CTS timeout	Determines a maximum time, in milliseconds, that a Driver waits for the **CTS** signal after turning the **RTS** signal on. If the **CTS** signal is not turned on within this time-out, that Driver then fails the current communication and returns an error
Delay before send	Some serial port devices have a delay when enabling a data sending circuit after the **RTS** signal is turned on. Configure this option to wait a certain number of milliseconds after turning the **RTS** signal on and before sending the first byte. **IMPORTANT**: This delay must be used carefully, because it uses 100% of CPU resources while waiting. System's general performance degrades as this value increases
Delay after send	This is the same effect of the **Delay before send** option, but in this case the delay is performed after sending the last byte, before turning the **RTS** signal off

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

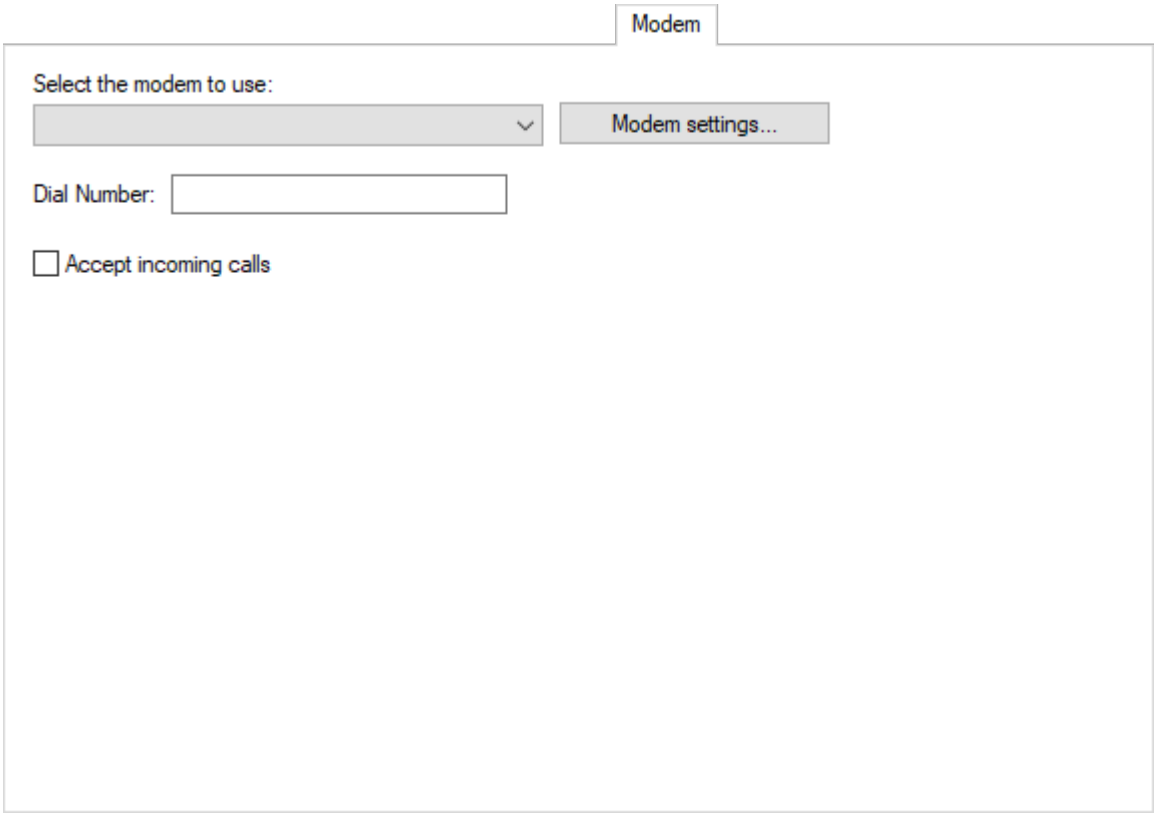
OPTION	DESCRIPTION
	of determining a successful connection before trying to open a socket with a device. The time-out of a connection with a socket can be very high. The available options are: • **Timeout**: Specify the number of milliseconds to wait for a reply from a **ping** command. Users must use 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

Modem Tab

Use this tab to configure parameters of a **Modem** Interface. Some options on the **Serial** tab affect the configuration of a modem, therefore users must also configure the **Serial** Interface.



Modem tab

The **Modem** Interface uses the TAPI modems installed on the computer.

Available options on the Modem tab

OPTION	DESCRIPTION
Select the modem to use	Select a modem on the list of available modems on the computer. If the value **Default modem** is selected, then the first available modem is used. Selecting this option is recommended specially when an application is used on another computer
Modem settings	Click to open the configuration window of the selected modem
Dial Number	Type a default number for dialing. This value can be changed at run time. Users can use the **w** character to represent a pause or a waiting time for a dial tone. For example, "0w33313456" dials the number 0 (zero), waits, and then dials the number "33313456"
Accept incoming calls	Enable this option so that a Driver answers the phone when receiving an external call. To use this option, users must configure the **Connection management** option on the **Setup** tab to the value **Manual**

RAS Tab

Use this tab configure parameters of a **RAS** Interface. Users must also configure the **Ethernet** tab.

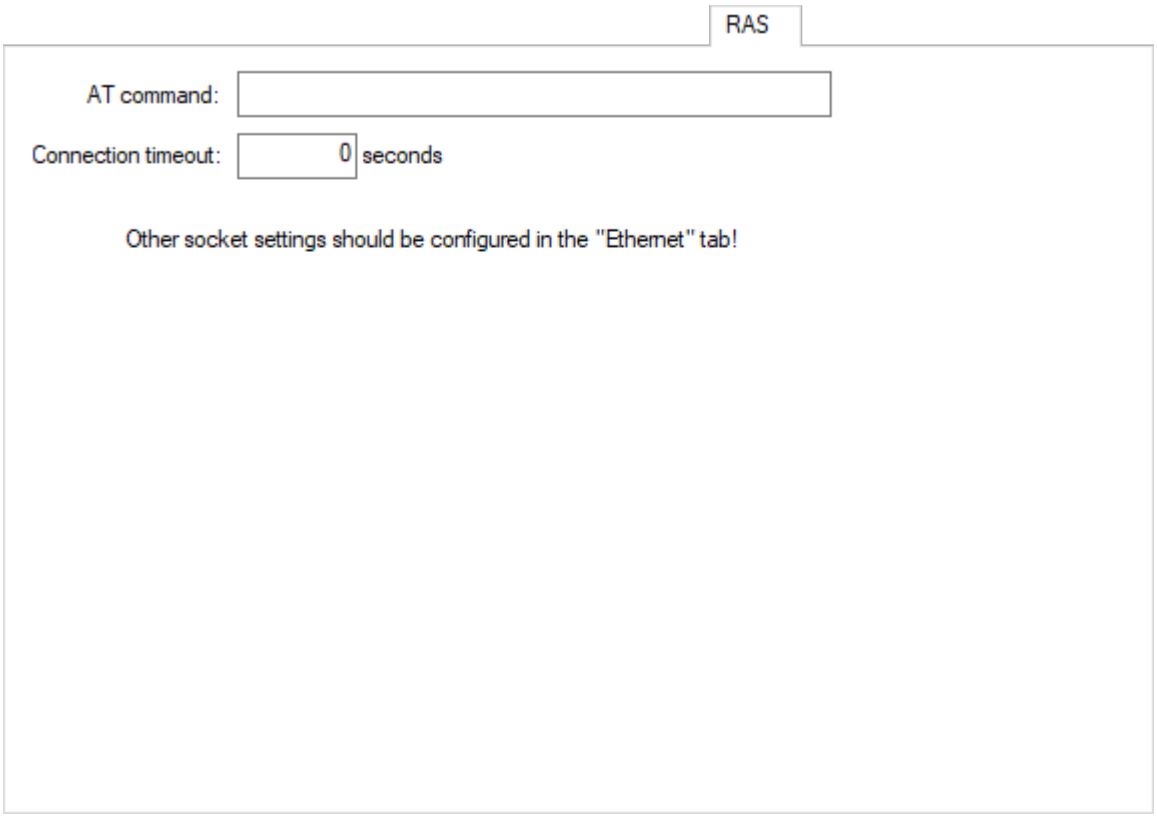
A **RAS** Interface opens a socket connection with a RAS device. A RAS device is a server of modems available through TCP/IP, waiting for socket connections on an IP port. For each connection accepted on this port, users have access to one modem.

When connecting to a RAS device, first the I/O Interface**IOKit** connects to the socket on the IP address and port configured on the **Ethernet** tab. After opening the socket, the following initialization or connection steps are performed:

1. Clears the socket, that is, removes any **TELNET** greeting message received from a RAS device.
2. Sends an **AT** dial message, in **ASCII** format, in the socket.
3. Waits for a **CONNECT** reply.
4. If the time-out expires, the connection is aborted.
5. If the **CONNECT** reply is received within the time-out, the socket is available for communication with a device, that is, the connection was established.

If step 5 (five) is successful, then the socket behaves as a normal socket, with the RAS device working as a router between a Driver and the device. Bytes sent by a Driver are received by the RAS device and sent to the destination device using a modem. Bytes received by the modem's RAS device are sent back to a Driver using the same socket.

After establishing a connection, the **RAS** interface monitors data received by a Driver. If a "NO CARRIER" **String** is found, the socket is closed. If the RAS device does not send a **NO CARRIER** signal, the **RAS** Interface cannot detect when the modem connection between the RAS device and the final I/O device fails. To recover from this failure, users are strongly advised to enable the **Disconnect if non-responsive** option on the **Setup** tab.



RAS tab

Available options on RAS tab

OPTION	DESCRIPTION
AT command	A **String** with the full **AT** command used to dial to a destination device. For example, "ATDT33313456" dials by tone to number "33313456"
Connection timeout	Number of seconds to wait for a modem's **CONNECT** reply, after sending an **AT** command

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

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

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

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

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

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

IO.Type

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

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

IO.Ethernet.PingTries

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).

Modem Interface Configuration

This section contains information about the configuration of **I/O Tags** and **Properties** of a **Modem** (TAPI) Interface.

I/O Tags

Tags of a Modem Interface (N2/B2 = 3)

The Tags described next allow controlling and diagnosing a **Modem** (TAPI) Interface at run time.

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

IO.TAPI.ConnectionBaudRate

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	3 (three)
N4 Parameter	5 (five)
String Configuration	IO.TAPI.ConnectionBaudRate

Indicates a baud rate value for the current connection. If a modem is not connected, returns the value 0 (zero).

IO.TAPI.Dial

Type of Tag	I/O Tag
Type of Access	Write-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	3 (three)
N4 Parameter	1 (one)
String Configuration	IO.TAPI.Dial

Write any value to this Tag to force a **Modem** Interface to start a call. This is an asynchronous command, only starting the call process. Users can monitor the **IO.TAPI.IsModemConnected** Tag to detect when a call is established.

IO.TAPI.HangUp

Type of Tag	I/O Tag
Type of Access	Write-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	3 (three)
N4 Parameter	4 (four)
String Configuration	IO.TAPI.HangUp

Any value written to this Tag hangs the current call up.

NOTE

Use this command only when managing the physical layer manually or when explicitly trying to force a Driver to restart the communication. If the physical layer is configured for automatic reconnection, a Driver immediately tries to reestablish the connection.

IO.TAPI.IsModemConnected

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	3 (three)
N4 Parameter	3 (three)
String Configuration	IO.TAPI.IsModemConnected

This Tag indicates the status of a modem connection. Possible values are **0**: The modem is not connected, but it may be performing or receiving an external call or **1**: The modem is connected and a Driver completed or received an external call successfully. While it is in this status, the physical layer can send or receive data.

IO.TAPI.IsModemConnecting

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	3 (three)
N4 Parameter	6 (six)
String Configuration	IO.TAPI.IsModemConnecting

This Tag indicates the status of a modem connection, with more details than the **IO.TAPI.IsModemConnected** Tag. Possible values are **0**: Modem is not connected, **1**: Modem is connecting, that is, performing or receiving an external call, **2**: Modem is connected. While in this status, the physical layer can send or receive data, or **3**: Modem is disconnecting the current call.

IO.TAPI.ModemStatus

Type of Tag	I/O Tag
Type of Access	Read-Only
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	3 (three)
N4 Parameter	2 (two)
String Configuration	IO.TAPI.ModemStatus

Returns a **String** with the current status of a modem. Possible values are the following:

- **"No status!"**: The **Modem** Interface was not open yet or was already closed
- **"Modem initialized OK!"**: The **Modem** Interface was initialized successfully
- **"Modem error at initialization!"**: A Driver could not initialize modem's line. Check that Driver's log file for more details
- **"Modem error at dial!"**: A Driver could not start or accept a call
- **"Connecting..."**: A Driver started a call successfully, and is currently processing that call
- **"Ringing..."**: Indicates that the modem is receiving an external call, but it did not accepted it yet
- **"Connected!"**: A Driver connected successfully, that is, completed or accepted an external call
- **"Disconnecting..."**: A Driver is turning the current call off
- **"Disconnected OK!"**: A Driver turned the current call off
- **"Error: no dial tone!"**: A Driver aborted a call because the available line signal was not detected
- **"Error: busy!"**: A Driver aborted a call because the line was busy
- **"Error: no answer!"**: A Driver aborted a call because no answer was received from the other modem
- **"Error: unknown!"**: Current call was aborted because of an unknown error

IO.TAPI.PhoneNumber

Type of Tag	I/O Tag
Type of Access	Reading or Writing
N1 Parameter	-1 (minus one)
N2 Parameter	0 (zero)
N3 Parameter	3 (three)
N4 Parameter	0 (zero)
String Configuration	IO.TAPI.PhoneNumber

This Tag is a **String** that reads or changes the telephone number used by the **IO.TAPI.Dial** Tag. When changing this Tag, the new value is used only on the next **Dial** command.

Properties

These properties control the configuration of a **Modem** (TAPI) Interface.

IO.TAPI.AcceptIncoming

Configure to False if a modem cannot accept external calls, that is, if a Driver behaves as a master, and configure to True to enable receiving calls, that is, if a Driver behaves as a slave.

IO.TAPI.ModemID

This is the identification number of a modem. This ID is created by Windows and used internally to identify a modem on a list of devices installed on a computer. This ID may not remain valid if a modem is reinstalled or an application is executed on another computer.

NOTE

It is advisable to configure this property as 0 (zero), indicating that a Driver must use the first available modem.

IO.TAPI.PhoneNumber

A telephone number used by **Dial** commands, such as "0w01234566", in which the "w" character forces a modem to wait for a call sign.

RAS Interface Configuration

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

I/O Tags

Tags of a RAS Interface (N2/B2 = 5)

Currently, there are no Tags defined specifically to manage a **RAS** Interface at run time.

Properties

These properties control the configuration of a **RAS** Interface.

NOTE

A **RAS** Interface uses the **Ethernet** Interface, which therefore must be also configured.

IO.RAS.ATCommand

An **AT** command to send through a socket to force a RAS device to perform a call using the current RAS channel, such as "ATDT6265545".

IO.RAS.CommandTimeoutSec

Time to wait for a **CONNECT** message in response to an **AT** command, in seconds.

Serial Interface Configuration

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

I/O Tags

Tags of a Serial Interface (N2/B2 = 2)

Currently, there are no Tags defined specifically to manage a **Serial** Interface at run time.

Properties

These properties control the configuration of a **Serial** Interface.

IO.Serial.Baudrate

Specifies a baud rate of a serial port, such as 9600.

IO.Serial.CTSTimeoutMs

Time to wait for a **CTS** signal, in milliseconds. After turning the **RTS** signal on, a timer is started to wait for a **CTS** signal. If this timer expires, a Driver aborts sending bytes through the serial port. Available only when the **IO.Serial.RTS** property is configured with the value **Toggle** and the **IO.Serial.WaitCTS** property is configured to True.

IO.Serial.DataBits

Specifies the number of data bits to configure a serial port. Possible values are **5**: Five data bits, **6**: Six data bits, **7**: Seven data bits, or **8**: Eight data bits.

IO.Serial.DelayAfterMs

Number of milliseconds to delay after the last byte is sent through a serial port, but before turning the **RTS** signal off. Available only when the **IO.Serial.RTS** property is configured with the value **Toggle** and the **IO.Serial.WaitCTS** property is configured to False.

IO.Serial.DelayBeforeMs

Number of milliseconds to delay after turning the **RTS** signal on, but before data is sent. Available only when the **IO.Serial.RTS** property is configured with the value **Toggle** and the **IO.Serial.WaitCTS** property is configured to False.

IO.Serial.DTR

Indicates how a Driver deals with the **DTR** signal. Possible values are **OFF**: **DTR** signal is always turned off or **ON**: **DTR** signal is always turned on.

IO.Serial.InterbyteDelayUs

Delay time, in milliseconds (1/1000000 of a second), for each two bytes sent through a **Serial** Interface.

IO.Serial.InterframeDelayMs

Delay time, in milliseconds, before sending a packet after the last packet sent or received.

IO.Serial.Parity

Specifies a parity for the configuration of a serial port. Possible values are **E or Even**: Even parity, **N or None**: No parity, **O or Odd**: Odd parity, **M or Mark**: Mark parity, or **S or Space**: Space parity.

IO.Serial.Port

Number of the local serial port. Possible values are **1**: Uses the COM1 port, **2**: Uses the COM2 port, **3**: Uses the COM3 port, or **n**: Uses the COM*n* port.

IO.Serial.RTS

Indicates how a Driver deals with the **RTS** signal. Possible values are **OFF**: **RTS** signal always off, **ON**: **RTS** signal always on, or **Toggle**: Turns the **RTS** signal on when transmitting data and turns the **RTS** signal off when not transmitting data.

IO.Serial.StopBits

Specifies the number of stop bits for the configuration of a serial port. Possible values are **1**: One stop bit, **2**: One and a half stop bit, or **3**: Two stop bits.

IO.Serial.SupressEcho

Use a value different from 0 (zero) to enable suppressing the echo or 0 (zero) to disable it.

IO.Serial.WaitCTS

Configure to True to force a Driver to wait for the **CTS** signal before sending bytes when the **RTS** signal is turned on. Available only when the **IO.Serial.RTS** property is configured with the value **Toggle**.

Driver Revision History

VERSION	DATE	AUTHOR	COMMENTS
2.0.32	09/02/2025	M. Ludwig	• Driver updated to **IOKit** library version **3.0** and Visual Studio 2022 (*Case 37953*).
2.0.31	12/20/2021	C. Mello	• Added support for Secure Apex 100 meters (*Case 30806*). • Added support for CEWE Prometer 100 meters (*Case 30807*). • Added support for AEC ADDAD meters (*Case 31831*).
2.0.30	04/22/2021	C. Mello	• Added support for interpreting data in **Float32** and **Float64** formats (*Case 30771*).

VERSION	DATE	AUTHOR	COMMENTS
2.0.29	09/04/2019	C. Mello	• Platform update of this Driver's source code (*Case 27486*).
2.0.28	12/13/2018	F. Englert	• Added support for reading user-defined OBIS codes via **ParamItem** property of a Tag (*Case 23430*). • Improvements when accessing Short Name Referencing for different firmwares of Landis+Gyr E550 and E650 meters (*Case 23208*). • Improvements in the Clock Synchronization Tag to prevent interference with the current collection interval (*Case 22124*). • Improvements when reading alarm events for Landis+Gyr meters (*Case 21529*). • Implemented support for dual access passwords, specific to reading and writing profiles (*Case 21464*). • Added support for access via Short Name Referencing to Landis+Gyr E650 meters (*Case 21415*). • Implemented HLS authentication support for Landis+Gyr meters (*Case 21343*). • Implemented a Clock Synchronization Tag, by informing a timezone (*Case 21121*). • Added support for access via Short Name Referencing to Landis+Gyr E550 meters (*Case 20134*). • Implemented a Tag for a descriptive query of the most recent error occurred with a meter (*Case 20624*). • Improvements in the disconnection process with a meter (*Case 20621*). • Improvements in the interpretation of AARE packages (*Case 20135*).

VERSION	DATE	AUTHOR	COMMENTS
			• Fixed a problem when checking the firmware version of Itron meters (*Case 19881*). • Implemented support to user-configurable HDLC Client Address (*Case 19575*). • Improvements when validating the protocol's initialization process (*Case 18811*). • Implemented support for collecting mass memory of data profiles different from the MAE standard (*Case 18579*). • Fixed an internal problem that could interrupt a mass memory collecting (*Case 18488*).
		C. Mello	• Improvements in the reception of data with noise-generated byte discarding (*Case 20442*). • Fixed a problem when collecting mass memory for partial periods less than a full day (*Case 19844*).
2.0.1	05/29/2014	C. Mello	• Adjustments for migration to **IOKit** library version **2.0** (*Case 15478*).
1.9.1	02/01/2013	C. Mello	• Added a value for the corresponding Profile ID in the Block Tag of the process of collecting mass memory (*Case 13782*). • Fixed problems with delays when updating the Status Tag for the process of collecting mass memory (*Case 13730*). • Fixed failures when interpreting data blocks in a format not recognized by this Driver when performing a process of collecting mass memory (*Case 13729*). • Added options for managing communication failures on this Driver's properties (*Case 13592*).

VERSION	DATE	AUTHOR	COMMENTS
			• Adjustments to prevent a possible lock when reading continuous bytes not expected by the protocol (*Case 13403*). • Improvements in the process of collecting mass memory for a partial period of the current day (*Case 13388*). • Adjustments to the command for changing a meter's clock (*Case 13386*). • Added support for interpreting protocol-provided error codes (*Case 13034*). • Fixed a calculation of quantities for the process of collecting mass memory of Profile ID 2 (*Case 12971*). • Implemented a hardkey-protection level (*Case 12592*).
1.8.1	06/18/2010	C. Mello	• Added support for addressing meters using Tag's *N1* or *B1* parameter (*Case 11518*).
1.7.1	05/26/2010	C. Mello	• Adjustments to perform a process of collecting mass memory with any group of channels (*Case 11242*). • Added support for reading Snapshot Values (the *N2* parameter between 501 and 506) of a meter (*Case 11191*). • Added support for a process of collecting mass memory and for reading meter parameters for other Profile IDs (*Case 11142*).
1.6.1	01/07/2010	C. Mello	• Added support for recognizing periods of power outage during a process of collecting mass memory (*Case 10958*).
1.5.1	10/15/2009	C. Mello	• Adjustments to prevent problems during a process of collecting mass memory

VERSION	DATE	AUTHOR	COMMENTS
			for periods with partial days (*Case 10706*). • Moved DLMS addressing of **[P]** parameters to Logical Device and Physical Device properties (*Case 10762*).
1.4.1	06/24/2009	C. Mello	• Adjustments to perform a process of collecting mass memory with support for several data structures (*Case 10387*). • Added a command for reading Channel Parameters (the *N2* parameter equal to 38) (*Case 10453*).
1.3.1	12/17/2008	C. Mello	• Adjustments to perform a process of collecting mass memory with a precision of hours for an informed period (*Case 10129*).
1.2.1	07/25/2008	C. Mello	• Fixed the consistency of Tag parameters (*Case 9660*).
1.1.1	09/18/2007	C. Mello	• Added four Block Elements for a **CT/VT** command (the *N2* parameter equal to 33) to return raw Numerators and Denominators for CT/VT ratio (*Case 8479*). • Added internals retries (*Case 8323*). • Added a command for reading an External Firmware Version (the *N2* parameter equal to 36) (*Case 7551*). • Added a command for reading an Internal Firmware Version (the *N2* parameter equal to 37) (*Case 7551*). • Fixed the process of collecting mass memory (*Case 7551*). • Added a **Default Slave Address** mode.
1.0.1	11/05/2003	Airgate	• First 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)