

# IEC 60870-5-101 Master Driver

© 2018 PTC Inc. All Rights Reserved.

# Table of Contents

<b>IEC 60870-5-101 Master Driver</b> .....	<b>1</b>
<b>Table of Contents</b> .....	<b>2</b>
IEC 60870-5-101 Master Driver .....	4
Overview .....	4
<b>Setup</b> .....	<b>5</b>
Channel Properties — General .....	5
Channel Properties — Serial Communications .....	6
Channel Properties — Write Optimizations .....	8
Channel Properties — Advanced .....	9
Channel Properties — Communication Serialization .....	10
Channel Properties — 60870 Settings .....	11
Device Properties — General .....	13
Device Properties — Scan Mode .....	15
Device Properties — Auto-Demotion .....	16
Device Properties — Time Synchronization .....	16
Device Properties — Communications .....	17
Event Playback .....	20
Timing .....	21
Device Properties — Redundancy .....	21
<b>Data Type Descriptions</b> .....	<b>22</b>
<b>Address Descriptions</b> .....	<b>23</b>
<b>Error Descriptions</b> .....	<b>30</b>
Device <device name> failed to complete clock synchronization. ....	30
Device <device name> failed to complete counter interrogation. ....	30
Device <device name> failed to complete general interrogation. ....	31
Device <device name> is not responding. ....	31
Error loading XML file for channel <channel name>. Unbalanced mode does not support a link address size of zero. ....	32
Error loading XML file for device <device name>. When the transmission mode is balanced, all devices must have the same link address. The link address of device <device name> is <link address>. ....	32
Error loading XML file for device <device name>. Common Address is invalid. Valid range is <low limit> to <high limit>. ....	32
Error loading XML file for device <device name>. The link address (<link address>) and common address (<common address>) pair is already in use on device <device name>. Link and common address pairs must be unique. ....	33

---

Error loading XML file for device <device name>. Link address is invalid. Valid range is <low limit> to <high limit>. ....	33
Read failed on tag <tag name>. ....	33
Read failed on tag <tag name> due to data type mismatch. ....	34
Reverting channel <channel name> link mode to Unbalanced Transmission. Balanced Transmission is not compatible with communication serialization or port sharing. ....	34
Too many events received in <device name>; the buffer has overflowed. Increase the event playback buffer size. ....	34
Unable to write to address <address> on device <device>. ....	35
Write failed on tag <tag name>. ....	35
<b>IEC 60870-5-101 Interoperability Guide</b> .....	<b>36</b>
<b>Index</b> .....	<b>50</b>

---

## IEC 60870-5-101 Master Driver

---

Help version 1.035

### CONTENTS

#### Overview

What is the IEC 60870-5-101 Master Driver?

#### Channel Setup

How do I configure a channel for use with this driver?

#### Device Setup

How do I configure devices for use with this driver?

#### Data Types Description

What data types does the IEC 60870-5-101 Master Driver support?

#### Address Descriptions

How do I address a data location on a device?

#### Error Descriptions

What error messages are produced by the IEC 60870-5-101 Master Driver?

#### IEC 60870-5-101 Interoperability Guide

Where can I find specific implementation information for the IEC 60870-5-101 Master Driver?

---

### Overview

---

IEC 60870 is a commonly used substation communication protocol similar to DNP. The IEC 60870-5-101 protocol uses specific terms to describe the communications pathway. Descriptions are as follows:

- **Channel:** This describes a communications path between two endpoints.
- **Session:** This describes a logical connection between a 101 master node (server channel) and a 101 slave node (server device). It comprises one or more 101 Sectors.
- **101 Sector:** This groups related data. It has its own Information Object Address (IOA) space. In the IEC 60870-5-101 Master Driver, a 101 session/sector pair is represented as server devices for each channel.

Thus, the server channel describes the communications pathway over which the master and slave will communicate. The other endpoint of the 101 channel may have one or more slave nodes and sectors available. A server device must be created for each sector in the endpoint.

## Setup

### Communication Protocol

IEC 60870-5-101 Master

For more information, refer to [IEC 60870-5-101 Interoperability Guide](#).

### Supported Devices

Any IEC 60870-5-101 slave device or gateway.

### Maximum Channels and Devices

The maximum number of channels supported is 256. The maximum number of devices supported per channel is 1024.

**Note:** When using Ethernet Encapsulation, this driver uses one socket per channel.

## Channel Properties — General

This server supports the use of simultaneous multiple communications drivers. Each protocol or driver used in a server project is called a channel. A server project may consist of many channels with the same communications driver or with unique communications drivers. A channel acts as the basic building block of an OPC link. This group is used to specify general channel properties, such as the identification attributes and operating mode.

Property Groups	<input type="checkbox"/> <b>Identification</b>	
General	Name	
Write Optimizations	Description	
Advanced	Driver	
	<input type="checkbox"/> <b>Diagnostics</b>	
	Diagnostics Capture	Disable

### Identification

**Name:** User-defined identity of this channel. In each server project, each channel name must be unique. Although names can be up to 256 characters, some client applications have a limited display window when browsing the OPC server's tag space. The channel name is part of the OPC browser information.

For information on reserved characters, refer to "How To... Properly Name a Channel, Device, Tag, and Tag Group" in the server help.

**Description:** User-defined information about this channel.

Many of these properties, including Description, have an associated system tag.

**Driver:** Selected protocol / driver for this channel. This property specifies the device driver that was selected during channel creation. It is a disabled setting in the channel properties.

**Note:** With the server's online full-time operation, these properties can be changed at any time. This includes changing the channel name to prevent clients from registering data with the server. If a client has already acquired an item from the server before the channel name is changed, the items are unaffected. If, after the channel name has been changed, the client application releases the item and attempts to re-acquire using the old channel name, the item is not accepted. With this in mind, changes to the properties should not be made once a large client application has been developed. Utilize the User Manager to prevent operators from changing properties and restrict access rights to server features.

## Diagnostics

**Diagnostics Capture:** When enabled, this option makes the channel's diagnostic information available to OPC applications. Because the server's diagnostic features require a minimal amount of overhead processing, it is recommended that they be utilized when needed and disabled when not. The default is disabled.

- **Note:** This property is not available if the driver does not support diagnostics.
- For more information, refer to "Communication Diagnostics" in the server help.

## Channel Properties — Serial Communications

Serial communication properties are available to serial drivers and vary depending on the driver, connection type, and options selected. Below is a superset of the possible properties.

Click to jump to one of the sections: [Connection Type](#), [Serial Port Settings](#) or [Ethernet Settings](#), and [Operational Behavior](#).

- **Note:** With the server's online full-time operation, these properties can be changed at any time. Utilize the User Manager to restrict access rights to server features, as changes made to these properties can temporarily disrupt communications.

Property Groups		
General	[-] <b>Connection Type</b>	
<b>Serial Communications</b>	Physical Medium	COM Port
Write Optimizations	Shared	No
Advanced	[-] <b>Serial Port Settings</b>	
Communication Serialization	COM ID	6
	Baud Rate	9600
	Data Bits	8
	Parity	Even
	Stop Bits	1
	Flow Control	None
	[-] <b>Operational Behavior</b>	
	Report Comm. Errors	Enable
	Close Idle Connection	Enable
	Idle Time to Close (s)	15

## Connection Type

**Physical Medium:** Choose the type of hardware device for data communications. Options include COM Port, None, Modem, and Ethernet Encapsulation. The default is COM Port.

- **None:** Select None to indicate there is no physical connection, which displays the [Operation with no Communications](#) section.
- **COM Port:** Select Com Port to display and configure the [Serial Port Settings](#) section.
- **Modem:** Select Modem if phone lines are used for communications, which are configured in the [Modem Settings](#) section.
- **Ethernet Encap.:** Select if Ethernet Encapsulation is used for communications, which displays the [Ethernet Settings](#) section.
- **Shared:** Verify the connection is correctly identified as sharing the current configuration with another channel. This is a read-only property.

## Serial Port Settings

**COM ID:** Specify the Communications ID to be used when communicating with devices assigned to the channel. The valid range is 1 to 9991 to 16. The default is 1.

**Baud Rate:** Specify the baud rate to be used to configure the selected communications port.

**Data Bits:** Specify the number of data bits per data word. Options include 5, 6, 7, or 8.

**Parity:** Specify the type of parity for the data. Options include Odd, Even, or None.


**Stop Bits:** Specify the number of stop bits per data word. Options include 1 or 2.

**Flow Control:** Select how the RTS and DTR control lines are utilized. Flow control is required to communicate with some serial devices. Options are:

- **None:** This option does not toggle or assert control lines.
- **DTR:** This option asserts the DTR line when the communications port is opened and remains on.
- **RTS:** This option specifies that the RTS line is high if bytes are available for transmission. After all buffered bytes have been sent, the RTS line is low. This is normally used with RS232/RS485 converter hardware.
- **RTS, DTR:** This option is a combination of DTR and RTS.
- **RTS Always:** This option asserts the RTS line when the communication port is opened and remains on.
- **RTS Manual:** This option asserts the RTS line based on the timing properties entered for RTS Line Control. It is only available when the driver supports manual RTS line control (or when the properties are shared and at least one of the channels belongs to a driver that provides this support).

RTS Manual adds an **RTS Line Control** property with options as follows:

- **Raise:** This property specifies the amount of time that the RTS line is raised prior to data transmission. The valid range is 0 to 9999 milliseconds. The default is 10 milliseconds.
- **Drop:** This property specifies the amount of time that the RTS line remains high after data transmission. The valid range is 0 to 9999 milliseconds. The default is 10 milliseconds.
- **Poll Delay:** This property specifies the amount of time that polling for communications is delayed. The valid range is 0 to 9999. The default is 10 milliseconds.

 **Tip:** When using two-wire RS-485, "echoes" may occur on the communication lines. Since this communication does not support echo suppression, it is recommended that echoes be disabled or a RS-485 converter be used.

## Operational Behavior

- **Report Comm. Errors:** Enable or disable reporting of low-level communications errors. When enabled, low-level errors are posted to the Event Log as they occur. When disabled, these same errors are not posted even though normal request failures are. The default is Enable.
- **Close Idle Connection:** Choose to close the connection when there are no longer any tags being referenced by a client on the channel. The default is Enable.
- **Idle Time to Close:** Specify the amount of time that the server waits once all tags have been removed before closing the COM port. The default is 15 seconds.

## Ethernet Settings

**Note:** Not all serial drivers support Ethernet Encapsulation. If this group does not appear, the functionality is not supported.

Ethernet Encapsulation provides communication with serial devices connected to terminal servers on the Ethernet network. A terminal server is essentially a virtual serial port that converts TCP/IP messages on the Ethernet network to serial data. Once the message has been converted, users can connect standard devices that support serial communications to the terminal server. The terminal server's serial port must be properly configured to match the requirements of the serial device to which it is attached. *For more information, refer to "How To... Use Ethernet Encapsulation" in the server help.*

- **Network Adapter:** Indicate a network adapter to bind for Ethernet devices in this channel. Choose a network adapter to bind to or allow the OS to select the default.  
 • *Specific drivers may display additional Ethernet Encapsulation properties. For more information, refer to Channel Properties — Ethernet Encapsulation.*

## Modem Settings

- **Modem:** Specify the installed modem to be used for communications.
- **Connect Timeout:** Specify the amount of time to wait for connections to be established before failing a read or write. The default is 60 seconds.
- **Modem Properties:** Configure the modem hardware. When clicked, it opens vendor-specific modem properties.
- **Auto-Dial:** Enables the automatic dialing of entries in the Phonebook. The default is Disable. *For more information, refer to "Modem Auto-Dial" in the server help.*
- **Report Comm. Errors:** Enable or disable reporting of low-level communications errors. When enabled, low-level errors are posted to the Event Log as they occur. When disabled, these same errors are not posted even though normal request failures are. The default is Enable.
- **Close Idle Connection:** Choose to close the modem connection when there are no longer any tags being referenced by a client on the channel. The default is Enable.
- **Idle Time to Close:** Specify the amount of time that the server waits once all tags have been removed before closing the modem connection. The default is 15 seconds.

## Operation with no Communications

- **Read Processing:** Select the action to be taken when an explicit device read is requested. Options include Ignore and Fail. Ignore does nothing; Fail provides the client with an update that indicates failure. The default setting is Ignore.

## Channel Properties — Write Optimizations

As with any server, writing data to the device may be the application's most important aspect. The server intends to ensure that the data written from the client application gets to the device on time. Given this goal, the server provides optimization properties that can be used to meet specific needs or improve application responsiveness.

Property Groups	<input checked="" type="checkbox"/> <b>Write Optimizations</b>	
General	Optimization Method	Write Only Latest Value for All Tags
<b>Write Optimizations</b>	Duty Cycle	10



## Write Optimizations

**Optimization Method:** controls how write data is passed to the underlying communications driver. The options are:

- **Write All Values for All Tags:** This option forces the server to attempt to write every value to the controller. In this mode, the server continues to gather write requests and add them to the server's internal write queue. The server processes the write queue and attempts to empty it by writing data to the device as quickly as possible. This mode ensures that everything written from the client applications is sent to the target device. This mode should be selected if the write operation order or the write item's content must uniquely be seen at the target device.
- **Write Only Latest Value for Non-Boolean Tags:** Many consecutive writes to the same value can accumulate in the write queue due to the time required to actually send the data to the device. If the server updates a write value that has already been placed in the write queue, far fewer writes are needed to reach the same final output value. In this way, no extra writes accumulate in the server's queue. When the user stops moving the slide switch, the value in the device is at the correct value at virtually the same time. As the mode states, any value that is not a Boolean value is updated in the server's internal write queue and sent to the device at the next possible opportunity. This can greatly improve the application performance.
  - **Note:** This option does not attempt to optimize writes to Boolean values. It allows users to optimize the operation of HMI data without causing problems with Boolean operations, such as a momentary push button.
- **Write Only Latest Value for All Tags:** This option takes the theory behind the second optimization mode and applies it to all tags. It is especially useful if the application only needs to send the latest value to the device. This mode optimizes all writes by updating the tags currently in the write queue before they are sent. This is the default mode.

**Duty Cycle:** is used to control the ratio of write to read operations. The ratio is always based on one read for every one to ten writes. The duty cycle is set to ten by default, meaning that ten writes occur for each read operation. Although the application is performing a large number of continuous writes, it must be ensured that read data is still given time to process. A setting of one results in one read operation for every write operation. If there are no write operations to perform, reads are processed continuously. This allows optimization for applications with continuous writes versus a more balanced back and forth data flow.

● **Note:** It is recommended that the application be characterized for compatibility with the write optimization enhancements before being used in a production environment.

## Channel Properties — Advanced

This group is used to specify advanced channel properties. Not all drivers support all properties; so the Advanced group does not appear for those devices.

Property Groups	[-] <b>Non-Normalized Float Handling</b>	
General	Floating-Point Values	Replace with Zero
Write Optimizations	[-] <b>Inter-Device Delay</b>	
<b>Advanced</b>	Inter-Device Delay (ms)	0

**Non-Normalized Float Handling:** A non-normalized value is defined as Infinity, Not-a-Number (NaN), or as a Denormalized Number. The default is Replace with Zero. Drivers that have native float handling may default to Unmodified. Non-normalized float handling allows users to specify how a driver handles non-normalized IEEE-754 floating point data. Descriptions of the options are as follows:

- **Replace with Zero:** This option allows a driver to replace non-normalized IEEE-754 floating point values with zero before being transferred to clients.
- **Unmodified:** This option allows a driver to transfer IEEE-754 denormalized, normalized, non-number, and infinity values to clients without any conversion or changes.

● **Note:** This property is not available if the driver does not support floating point values or if it only supports the option that is displayed. According to the channel's float normalization setting, only real-time driver tags (such as values and arrays) are subject to float normalization. For example, EFM data is not affected by this setting.

● *For more information on the floating point values, refer to "How To ... Work with Non-Normalized Floating Point Values" in the server help.*

**Inter-Device Delay:** Specify the amount of time the communications channel waits to send new requests to the next device after data is received from the current device on the same channel. Zero (0) disables the delay.

● **Note:** This property is not available for all drivers, models, and dependent settings.

## Channel Properties — Communication Serialization

The server's multi-threading architecture allows channels to communicate with devices in parallel. Although this is efficient, communication can be serialized in cases with physical network restrictions (such as Ethernet radios). Communication serialization limits communication to one channel at a time within a virtual network.

The term "virtual network" describes a collection of channels and associated devices that use the same pipeline for communications. For example, the pipeline of an Ethernet radio is the master radio. All channels using the same master radio associate with the same virtual network. Channels are allowed to communicate each in turn, in a "round-robin" manner. By default, a channel can process one transaction before handing communications off to another channel. A transaction can include one or more tags. If the controlling channel contains a device that is not responding to a request, the channel cannot release control until the transaction times out. This results in data update delays for the other channels in the virtual network.

Property Groups	<input type="checkbox"/> <b>Channel-Level Settings</b>	
General	Virtual Network	None
Serial Communications	Transactions per Cycle	1
<b>Communication Serialization</b>	<input type="checkbox"/> <b>Global Settings</b>	
	Network Mode	Load Balanced

### Channel-Level Settings

**Virtual Network** This property specifies the channel's mode of communication serialization. Options include None and Network 1 - Network 500. The default is None. Descriptions of the options are as follows:

- **None:** This option disables communication serialization for the channel.
- **Network 1 - Network 500:** This option specifies the virtual network to which the channel is assigned.

**Transactions per Cycle** This property specifies the number of single blocked/non-blocked read/write transactions that can occur on the channel. When a channel is given the opportunity to communicate, this number of transactions attempted. The valid range is 1 to 99. The default is 1.

## Global Settings

- **Network Mode:** This property is used to control how channel communication is delegated. In **Load Balanced** mode, each channel is given the opportunity to communicate in turn, one at a time. In **Priority** mode, channels are given the opportunity to communicate according to the following rules (highest to lowest priority):
  - Channels with pending writes have the highest priority.
  - Channels with pending explicit reads (through internal plug-ins or external client interfaces) are prioritized based on the read's priority.
  - Scanned reads and other periodic events (driver specific).

The default is Load Balanced and affects *all* virtual networks and channels.

🔴 Devices that rely on unsolicited responses should not be placed in a virtual network. In situations where communications must be serialized, it is recommended that Auto-Demotion be enabled.

Due to differences in the way that drivers read and write data (such as in single, blocked, or non-blocked transactions); the application's Transactions per cycle property may need to be adjusted. When doing so, consider the following factors:

- How many tags must be read from each channel?
- How often is data written to each channel?
- Is the channel using a serial or Ethernet driver?
- Does the driver read tags in separate requests, or are multiple tags read in a block?
- Have the device's Timing properties (such as Request timeout and Fail after x successive timeouts) been optimized for the virtual network's communication medium?

## Channel Properties — 60870 Settings

The 60870 Settings group is used to specify the IEC 60870-5-101 Master Driver settings for communications with a specific channel on a 101 slave. Descriptions of the properties are below.

Property Groups		
General		
Serial Communications		
Write Optimizations		
Advanced		
Communication Serialization		
<b>60870 Settings</b>		
<b>Address Settings</b>		
Cause of Transmission (COT) Size	One Octet	
Originator Address	0	
<b>Link Settings</b>		
Link Mode	Unbalanced Transmission	
RX Buffer Size	255	
Link Confirm Timeout (ms)	2000	
Class-1 Poll Interval (ms)	0	
Class-2 Poll Interval (ms)	500	
Class-1 Pending Delay (ms)	0	
Class-2 Pending Delay (ms)	0	
<b>Size Settings</b>		
Link Address Size	One Octet	
Common Address (ASDU) Size	Two Octets	
Information Object Address (IOA) Size	Two Octets	
<b>Timing Settings</b>		
Incremental Timeout (ms)	30000	
First Char Wait (ms)	0	

### Address Settings

- **Cause of Transmission (COT) Size:** This property specifies the number of octets in an ASDU COT field. When Two Octets is selected, the master's originator address is included in messages sent to the slave. When One Octet is selected, the originator address is not included and is disabled in the configuration. The default setting is Two Octets.
- **Originator Address:** This property specifies the second byte of the Cause of Transmission (COT) field, which is used by a dual-mode device to route responses from a slave to the correct master. The valid range is 0 to 254. The default setting is 0.

### Link Settings

- **Link Mode:** This property specifies the mode in which the device will be configured. Options include Unbalanced Transmission and Balanced Transmission. The default setting is Unbalanced Transmission. Descriptions of the options are as follows:
  - **Balanced Transmission:** This mode allows a master to communicate with one device, and allows the device to send unsolicited updates.
  - **Unbalanced Transmission:** This mode allows a master to communicate with multiple devices. It does not support unsolicited updates; as such, all data from each device must be polled.
- **RX Buffer Size:** This property specifies the maximum data size it is possible to receive. Messages containing more than the specified size are discarded. This value can range between 6 and 255. The default setting is 255 bytes.
- **Link Confirm Timeout:** This property specifies the maximum amount of time (in milliseconds) to wait for a confirmation of a link frame. The default setting is 2000 ms.
- **Class-1 Poll Interval:** This property specifies the time (in milliseconds) between each Class 1 data poll when the master is not expecting data. It is used by the unbalanced transmission link mode to acquire high-priority data, such as events, periodically. The default value is 0, which disables it. Most

devices do not require periodic Class 1 polling because Class 2 polling notifies the driver when class 1 data exists.

- **Class-2 Poll Interval:** This property specifies the time (in milliseconds) between each Class 2 data poll when the master is not expecting data. It is used by the unbalanced transmission link mode to acquire low-priority data, such as cyclic data, periodically. The default value is 500. Many devices notify the driver of Class 1 data during Class 2 polling. When this occurs, a Class 1 poll is automatically scheduled, regardless of the Class 1 poll interval. Class 2 polling cannot be disabled.
- **Caution:** The unbalanced mode class poll interval should be set appropriately according to how other master operations are configured. The class poll interval should be frequent enough to service all master requests and slave responses during that time frame. For example, a configuration that could perform poorly would be a Class 2 Poll Interval of 20,000 milliseconds and a Test Procedure Period of 5 seconds. In general, the class poll interval should be at least twice as fast as any other master operations.
- **Class-1 Pending Delay:** This property specifies the time (in milliseconds) between each Class 1 data poll when the master is expecting data. The default value is 0.
- **Class-2 Pending Delay:** This property specifies the time (in milliseconds) between each Class 2 data poll when the master is expecting data. The default value is 0.

### Size Settings

- **Link Address Size:** This property specifies the number of octets in a device link address. Options include Zero, One Octet, and Two Octets. Balanced communications support zero, one, or two octets. Unbalanced communications support one or two octets. The number of octets that are selected determines the maximum link address allowed in the device setup. The default setting is One Octet.
- **Common Address (ASDU) Size:** This property specifies the number of octets in a device common address. Options include One Octet and Two Octets. The number of octets that are selected determines the maximum common address allowed in the device. The default setting is Two Octets.
- **Information Object Address (IOA) Size:** This property specifies the number of octets in a tag information object address. Options include One Octet, Two Octets, or Three Octets. The number of octets that are selected determines the maximum address allowed as a tag IOA. The default setting is Two Octets.

### Timing Settings

- **Incremental Timeout (ms):** This defines the maximum time to wait for a response from a device when a command is outstanding. The default setting is 30000 ms.
- **First Char Wait (ms):** This property specifies how long to wait (in milliseconds) after receiving a character before attempting to transmit a character. The default setting is 0 ms.

## Device Properties — General

---

A device represents a single target on a communications channel. If the driver supports multiple controllers, users must enter a device ID for each controller.

Property Groups	Identification	
General	Name	
Scan Mode	Description	
	Channel Assignment	
	Driver	
	Model	
	ID Format	Decimal
	ID	2
	Operating Mode	
	Data Collection	Enable
	Simulated	No

## Identification

**Name:** This property specifies the name of the device. It is a logical user-defined name that can be up to 256 characters long, and may be used on multiple channels.

● **Note:** Although descriptive names are generally a good idea, some OPC client applications may have a limited display window when browsing the OPC server's tag space. The device name and channel name become part of the browse tree information as well. Within an OPC client, the combination of channel name and device name would appear as "ChannelName.DeviceName".

● *For more information, refer to "How To... Properly Name a Channel, Device, Tag, and Tag Group" in server help.*

**Description:** User-defined information about this device.

● Many of these properties, including Description, have an associated system tag.

**Channel Assignment:** User-defined name of the channel to which this device currently belongs.

**Driver:** Selected protocol driver for this device. This property specifies the driver selected during channel creation. It is disabled in the channel properties.

**Model:** This property specifies the specific type of device that is associated with this ID. The contents of the drop-down menu depends on the type of communications driver being used. Models that are not supported by a driver are disabled. If the communications driver supports multiple device models, the model selection can only be changed when there are no client applications connected to the device.

● **Note:** If the communication driver supports multiple models, users should try to match the model selection to the physical device. If the device is not represented in the drop-down menu, select a model that conforms closest to the target device. Some drivers support a model selection called "Open," which allows users to communicate without knowing the specific details of the target device. For more information, refer to the driver help documentation.

**ID:** This property specifies the device's station / node / identity / address. The type of ID entered depends on the communications driver being used. For many drivers, the ID is a numeric value. Drivers that support a Numeric ID provide users with the option to enter a numeric value whose format can be changed to suit the needs of the application or the characteristics of the selected communications driver. The ID format can be Decimal, Octal, and Hexadecimal. If the driver is Ethernet-based or supports an unconventional station or node name, the device's TCP/IP address may be used as the device ID. TCP/IP addresses consist of four values that are separated by periods, with each value in the range of 0 to 255. Some device IDs are string based. There may be additional properties to configure within the ID field, depending on the driver.

## Operating Mode

**Data Collection:** This property controls the device's active state. Although device communications are enabled by default, this property can be used to disable a physical device. Communications are not attempted when a device is disabled. From a client standpoint, the data is marked as invalid and write operations are not accepted. This property can be changed at any time through this property or the device system tags.

**Simulated:** This option places the device into Simulation Mode. In this mode, the driver does not attempt to communicate with the physical device, but the server continues to return valid OPC data. Simulated stops physical communications with the device, but allows OPC data to be returned to the OPC client as valid data. While in Simulation Mode, the server treats all device data as reflective: whatever is written to the simulated device is read back and each OPC item is treated individually. The item's memory map is based on the group Update Rate. The data is not saved if the server removes the item (such as when the server is reinitialized). The default is No.

● **Notes:**

1. This System tag (\_Simulated) is read only and cannot be written to for runtime protection. The System tag allows this property to be monitored from the client.
2. In Simulation mode, the item's memory map is based on client update rate(s) (Group Update Rate for OPC clients or Scan Rate for native and DDE interfaces). This means that two clients that reference the same item with different update rates return different data.

● Simulation Mode is for test and simulation purposes only. It should never be used in a production environment.

## Device Properties — Scan Mode

The Scan Mode specifies the subscribed-client requested scan rate for tags that require device communications. Synchronous and asynchronous device reads and writes are processed as soon as possible; unaffected by the Scan Mode properties.

Property Groups	☐ <b>Scan Mode</b>	
General	Scan Mode	Respect Client-Specified Scan Rate ▼
<b>Scan Mode</b>	Initial Updates from Cache	Disable

**Scan Mode:** specifies how tags in the device are scanned for updates sent to subscribing clients.

Descriptions of the options are:

- **Respect Client-Specified Scan Rate:** This mode uses the scan rate requested by the client.
- **Request Data No Faster than Scan Rate:** This mode specifies the maximum scan rate to be used. The valid range is 10 to 99999990 milliseconds. The default is 1000 milliseconds.
  - **Note:** When the server has an active client and items for the device and the scan rate value is increased, the changes take effect immediately. When the scan rate value is decreased, the changes do not take effect until all client applications have been disconnected.
- **Request All Data at Scan Rate:** This mode forces tags to be scanned at the specified rate for subscribed clients. The valid range is 10 to 99999990 milliseconds. The default is 1000 milliseconds.
- **Do Not Scan, Demand Poll Only:** This mode does not periodically poll tags that belong to the device nor perform a read to get an item's initial value once it becomes active. It is the client's

responsibility to poll for updates, either by writing to the `_DemandPoll` tag or by issuing explicit device reads for individual items. *For more information, refer to "Device Demand Poll" in server help.*

- **Respect Tag-Specified Scan Rate:** This mode forces static tags to be scanned at the rate specified in their static configuration tag properties. Dynamic tags are scanned at the client-specified scan rate.

**Initial Updates from Cache:** When enabled, this option allows the server to provide the first updates for newly activated tag references from stored (cached) data. Cache updates can only be provided when the new item reference shares the same address, scan rate, data type, client access, and scaling properties. A device read is used for the initial update for the first client reference only. The default is disabled; any time a client activates a tag reference the server attempts to read the initial value from the device.

## Device Properties — Auto-Demotion

The Auto-Demotion properties can temporarily place a device off-scan in the event that a device is not responding. By placing a non-responsive device offline for a specific time period, the driver can continue to optimize its communications with other devices on the same channel. After the time period has been reached, the driver re-attempts to communicate with the non-responsive device. If the device is responsive, the device is placed on-scan; otherwise, it restarts its off-scan time period.

Property Groups	Auto-Demotion	
General	Demote on Failure	Enable
Scan Mode	Timeouts to Demote	3
Timing	Demotion Period (ms)	10000
Auto-Demotion	Discard Requests when Demoted	Disable

**Demote on Failure:** When enabled, the device is automatically taken off-scan until it is responding again.

**Tip:** Determine when a device is off-scan by monitoring its demoted state using the `_AutoDemoted` system tag.

**Timeouts to Demote:** Specify how many successive cycles of request timeouts and retries occur before the device is placed off-scan. The valid range is 1 to 30 successive failures. The default is 3.

**Demotion Period:** Indicate how long the device should be placed off-scan when the timeouts value is reached. During this period, no read requests are sent to the device and all data associated with the read requests are set to bad quality. When this period expires, the driver places the device on-scan and allows for another attempt at communications. The valid range is 100 to 3600000 milliseconds. The default is 10000 milliseconds.

**Discard Requests when Demoted:** Select whether or not write requests should be attempted during the off-scan period. Disable to always send write requests regardless of the demotion period. Enable to discard writes; the server automatically fails any write request received from a client and does not post a message to the Event Log.

## Device Properties — Time Synchronization

This group is used to specify the device's time zone and time synchronization properties. It primarily applies to time stamped data or information from battery-powered devices at remote locations where the device time may deviate (causing issues with the time-stamped data). To prevent this problem from occurring, users can specify that the server synchronize the device time.



Property Groups	<input type="checkbox"/> <b>Time Zone</b>	
General	Time Zone	(UTC-05:00) Eastern Time (US & Canada)
Scan Mode	Respect Daylight Saving Time	Yes
Timing	<input type="checkbox"/> <b>Synchronization</b>	
Auto-Demotion	Time Sync Method	<b>Absolute</b>
Tag Generation	Time Sync Threshold (sec)	0
<b>Time Synchronization</b>	Sync Absolute	12:00:00 AM
Settings		
EFM Meters		
Trend		
Redundancy		

● **Note:** Not all drivers and models support all options.

**Time Zone:** This property specifies the device's time zone. To ignore the time zone, select one of the first four options in the list (which do not have an offset). The default is the time zone of the local system.

● **Note:** The driver uses this property both when syncing the device time and when converting EFM timestamps from the device to UTC time.

**Respect Daylight Saving Time:** Select Yes to follow Daylight Saving Time offset when syncing the device time. Select No to ignore Daylight Saving Time. Only time zones that observe Daylight Saving Time will be affected. The default is No (disabled).

● **Note:** When enabled, the time of the device is adjusted by +1 hour for Daylight Saving Time (in the spring), and adjusted by -1 hour after Daylight Saving Time (in the fall).

**Time Sync Method:** This property specifies the method of synchronization. Options include Disabled, Absolute, and Interval. The default is Disabled. Descriptions of the options are as follows:

- **Disabled:** No synchronization.
- **Absolute:** Synchronizes to an absolute time of day specified through the Time property (appears only when Absolute is selected).
- **Interval:** Synchronizes on startup and every number of minutes specified through the Sync Interval property (appears only when Interval is selected). The default is 60 minutes.
- **OnPoll:** Synchronizes when poll is completed (applicable only to EFM devices).

**Time Sync Threshold:** This property specifies the maximum allowable difference, in seconds, between the device time and the system time before syncing the device time to the system time. If the threshold is set to 0, a time synchronization occurs every time. The default is 0 seconds. The maximum allowable threshold is 600 seconds.

## Device Properties — Communications

The Communications group is used to specify the settings for communications with a specific sector on a 101 slave. Descriptions of the properties are below.

Property Groups	<input type="checkbox"/> <b>Address Settings</b>	
General	Common Address	3
Scan Mode	Link Address	0
Auto-Demotion	<input type="checkbox"/> <b>General Settings</b>	
Time Synchronization	Polled Reads	Yes
<b>Communications</b>	Delay Acquisition	No
Redundancy	<input type="checkbox"/> <b>Timing Settings</b>	
	Request Timeout (ms)	10000
	Interrogation Request Timeout (milliseconds)	60000
	Attempt Count	3
	Interrogation Attempt Count	3
	<input type="checkbox"/> <b>Initialization Settings</b>	
	Time Sync Initialization	End of Initialization
	GI Initialization	End of Initialization
	CI Initialization	End of Initialization
	<input type="checkbox"/> <b>Periodic Settings</b>	
	Periodic GI Interval (min.)	720
	Periodic CI Interval (min)	0
	<input type="checkbox"/> <b>Test Procedure Settings</b>	
	Test Procedure	Yes
	Test Procedure Period (s)	15
	<input type="checkbox"/> <b>Event Playback Settings</b>	
	Playback Events	Yes
	Playback Buffer Size	100
	Playback Rate (ms)	2000

### Address Settings

- **Common Address:** This property specifies whether to permit the addressing of the whole station and, optionally, a particular sector within a station. A station represents a physical device; a sector represents a set of data within a device. Sectors are commonly used by pass-through devices (which must separate data from the multiple devices they represent) and by devices that want to segment their data based on type. The valid range is 0 to 254 or 65534, depending on whether the Common Address (ASDU) Size setting is set to one or two octets. The default setting is 3.
- **Link Address:** This property identifies a session, which is a physical or logical device in the network. The valid range is 0 to 254 or 0 to 65534, depending on whether the Link Address Size setting is set to one or two octets. The default setting is 0. In Balanced Transmission mode, this setting is not used when the Link Address Size is set to zero octets.

#### Notes:

1. Devices within a channel must have unique link/common address pairs because the master must have unique sector references within a channel.
2. When a channel link mode is set to Balanced Transmission, all devices are required to use the same Link Address because this link mode requires the channel be connected to a single slave (point-to-point) connection.

### General Settings

- **Polled Reads:** When enabled, unbuffered tags perform polled reads when last-read data is older than the scan rate. When disabled, only previously received cyclic and background scan data is reported. If polled reads are disabled, the driver reports "Uncertain" quality until cyclic or background scan updates have been received. Default is enabled.

- **Delay Acquisition:** Acquisition is used to account for transmission delay between the master and slave. When enabled, the master performs Delay Acquisition before issuing Time Sync commands to the device. Delay Acquisition is disabled by default.

### Timing Settings

- **Request Timeout (ms):** This property specifies an interval that determines how long the driver waits for a general response from the target device to complete. The default setting is 10000 ms. General Interrogation and Counter Interrogation use their own timeout setting: Interrogation Request Timeout.
- **Interrogation Request Timeout (ms):** This property specifies the General and Counter Interrogation request timeouts. Having a separate timeout for interrogation enables larger timeouts for these longer requests.
- **Attempt Count:** This property specifies how many times the driver tries a communication request before considering the request to have failed and the device to be in error. The valid range is 1 to 10. The default setting is 3. General Interrogation and Counter Interrogation use their own setting for the number of attempts: Interrogation Attempt Count.
- **Interrogation Attempt Count:** This property specifies the number of General and Counter Interrogation attempts.

### Initialization Settings

- **Time Sync Initialization:** Specifies if and when a time synchronization should be sent to the slave during initialization. The default settings is "End of Initialization." This property is independent of the Synchronization setting of the Time Synchronization property group and Command.Timesync tag.
  - **End of Initialization:** Perform Time Sync when the device notifies the master that it has ended initialization.
  - **Connect/Reconnect:** Perform Time Sync when a connection is established. This occurs any time the master reestablishes communication with the slave.
  - **None:** Do not perform an initial Time Sync.
- **GI Initialization:** Specifies if and when a General Interrogation (GI) should be sent to the slave during initialization. The default setting is "End of Initialization." This property is independent of the Periodic GI Interval setting and the Command.GI command tag.
  - **End of Initialization:** Perform GI when the device notifies the master that it has ended initialization.
  - **Connect/Reconnect:** Perform GI when a connection is established. This occurs any time the master reestablishes communication with the slave.
  - **None:** Do not perform an initial GI.
- **CI Initialization:** Specifies if and when a Counter Interrogation (CI) should be sent to the slave during initialization. The default setting is "End of Initialization." This property is independent of the Periodic CI Interval setting and the Command.CI command tag.
  - **End of Initialization:** Perform CI when the device notifies the master that it has ended initialization.
  - **Connect/Reconnect:** Perform CI when a connection is established. This occurs any time the master reestablishes communication with the slave.
  - **None:** Do not perform an initial CI.

### Periodic Settings

- **Periodic GI Interval (minutes):** Configures the master to perform a General Interrogation based on a specified time interval. The default setting is 720 minutes (12 hours). When clients are connected, General Interrogations are sent every time the specified period elapses. Specifying an interval of 0 disables periodic GI.
- **Periodic CI Interval (minutes):** Configures the master to perform a Counter Interrogation based on a specified time interval. When clients are connected, Counter Interrogations are sent every time the specified period elapses. The default setting is 0 minutes (disabled). Specifying an interval of 0 disables periodic CI.

### Test Procedure Settings

- **Test Procedure:** Specifies whether or not test command is enabled. When enabled, the master periodically sends a test command ASDU. This command can be used to determine device error state. It is recommended to use this when devices are expected to go long periods without communicating. The default setting is enabled.
- **Test Procedure Period (seconds):** Test commands are sent at the specified rate. The default setting is 15 seconds.

### Playback Settings

- **Playback Events:** Specifies whether or not event playback is enabled. When disabled, all tags associated with event playback report the most recent data and no data is buffered. When enabled, events are played back based on the buffer size and playback rate. The default is enabled. See [Event Playback](#).
- **Playback Buffer Size:** Maximum number of events buffered for each Information Object Address (IOA) buffer. The default is 100.
- **Playback Rate (ms):** Rate at which events are played back. The default setting is 2000 ms.

## Event Playback

A device may send many updates in one transaction. To address this, event playback handles multiple value changes for data points that are then displayed (played back) in the client. For example, if a device sends a set of packed events that represent some history of an analog event (such as voltage values during a fault), the data is played back in the historical sequence. The driver plays back each data point at a set interval to allow the client to read each update in the order it was received from the device.

Event playback also guarantees that each change to a data point is read. If the playback interval expires before the data has been accessed, the data is maintained until the client references it.

To guarantee that no data is missed, it is recommended that client update rates be at least as fast as the event playback interval. This is not required with a single client because data is held until at least one read has occurred; however, it is required in the case of multiple clients reading the same data point to ensure that each client receives all the data.

The quality of a buffered tag is “uncertain” until data associated with the tag is received. If a DNR occurs during event playback, all buffered events are played back before reporting “bad” quality.

### Notes:

1. Please see the OPC DA Compliance note for OPC DA Compliance settings that can affect event playback.

- The OPC timestamp is based on the device time (if available) or the server time. When the OPC timestamp is based on server time, it represents the time of the data arrival.

## Timing

Timing settings come in two varieties: link-layer timing and application-layer timing. Link-layer timing settings are used to control or time out the acquisition of application layer data. Application-layer data contains the individual commands to read or write. For example, General Interrogation is performed as an application-layer command, but the acquisition of individual data is accomplished through the link layer.

Device-level application-layer timeouts are controlled through the request timeout device settings. These settings are the only timing properties that can affect tag quality.

The channel-level timing settings affect the link layer. They do not directly affect the quality of a tag, but they can be tuned to allow for delays or timeouts during the acquisition of individual data pieces. These settings can introduce complex interactions in data acquisition and should not be changed without a compelling reason.

**Note:** If Link Confirm Timeout is less than Request Timeout, the first communication timeout to a device fails based on the Link Confirm Timeout. The device error state is not set because it is not an application-layer failure. However, subsequent failures are treated as application-layer timeouts. To ensure that communications timeouts always result in the device entering error state, the device Attempt Count should be greater than 1.

## Device Properties — Redundancy

Property Groups	<b>Redundancy</b>	
General	Secondary Path	...
Scan Mode	Operating Mode	Switch On Failure
Timing	Monitor Item	
<b>Redundancy</b>	Monitor Interval (s)	300
	Return to Primary ASAP	Yes

Redundancy is available with the Media-Level Redundancy Plug-In.

**Note:** Consult the website, a sales representative, or the user manual for more information.

## Data Type Descriptions

---

Data Type	Description
Boolean	Single bit
Float	32-bit floating-point value
Byte	Unsigned 8-bit value Bit 0 is the low bit Bit 7 is the high bit
Date	64-bit value representing the number of milliseconds since December 31, 1899
Char	Signed 8-bit data
Word	Unsigned 16-bit data
DWord	Unsigned 32-bit data
Short	Signed 16-bit data
Long	Signed 32-bit data

## Address Descriptions

There are two forms of tag addressing. The first is information object address (IOA) based tag addressing, which allows access to each individual data point on a device. IOA-based tag addressing includes monitor ASDUs and control ASDUs. The second form is device-level command-based tag addressing, which allows control of device-specific commands.

### IOA-Based Tag Addressing

#### Monitor ASDUs

#### Control ASDUs

### Command-Based Tag Addressing

## IOA-Based Tag Addressing

Tag addressing takes the form *TYPEID.IOA.SUB-TYPE.OPTIONAL.OPTIONAL*, where:

- *TYPEID* is the ASDU type.
- *IOA* is the Information Object Address (IOA).
- *SUB-TYPE* is the point's driver-specific attribute.
- *OPTIONAL* is an attribute that is not required.

● **Note:** The Information Object Address (IOA) represents a slave device's point address. It is an integer between 1 and 254, 65535, or 16777215; depending on whether the Information Object Address Size property is set to one, two, or three octets (respectively).

### **TYPEID**

The TYPEID represents the ASDU type as defined by the protocol. Types include monitor or control. Each monitor TYPEID supports multiple protocol-defined ASDU numbers. Monitor types represent data that is read only. Control types represent data that can be written.

#### Monitor ASDUs

#### Control ASDUs

## Monitor ASDUs

Monitor TYPEID	Data Type	ASDU	Description
M_SP	Boolean	1, 2, 30	Single-Point Information
M_DP	Char	3, 4, 31	Double-Point Information: Possible values are 0, 1, 2, and 3. 0: Indeterminate or Intermediate State 1: Determined State OFF 2: Determined State ON 3: Indeterminate State
M_ST	Char	5, 6, 32	Step Position Information: Possible values are -64 to +63. Supports the Transient sub-type.
M_BO	DWord	7, 8, 33	Bit string of 32 bit. Support bit indexing into the following sub-types: Value and CurrentValue.

Monitor TYPEID	Data Type	ASDU	Description
M_ME_NV	Float	9, 10, 21, 34	Measured Value, Normalized Value: This represents a value between -1 and 0.999969482421875. Devices transmit this value and the driver converts this into its appropriate floating-point value.
M_ME_SV	Short	11, 12, 35	Measured Value, Scaled Value. Possible values are -32768 to +32767.
M_ME_FV	Float	13, 14, 36	Measured Value, Short Floating Point Number.
M_IT	Long	15, 16, 37	Integrated Totals. Supports the SQ sub-type. Does not support the unbuffered "Current" sub-types.

**SUB-TYPE**

The SUB-TYPES that are available are determined by the TYPEID. Monitor TYPEIDs can only use monitor SUB-TYPES. For more information, refer to the table below.

SUB-TYPE	Data Type	Description
Value	See TYPEID	The Event Playback value. Values received during General Interrogation (GI) and event acquisition are played back with this tag. If the quality descriptor IV (invalid) bit is set, this tag shows bad quality. When used with M_BO, this supports bit indexing. For example, to access bit 0 in the value tag, use the sub-type Value with .0 appended: Value.0.
CurrentValue*	See TYPEID	This reports all unbuffered data; which includes cyclic, background scan, and polled reads. If the quality descriptor IV (invalid) bit is set, this tag shows bad quality. When used with M_BO, this supports bit indexing. For example, to access bit 0 in the value tag, use the sub-type CurrentValue with .0 appended: CurrentValue.0. Not supported for M_IT. Polled reads are performed with the read command: C_RD Type ID 102. Although the standard considers this a control operation, the physical reads are performed with the monitor TYPEIDs.
Timestamp	Date	The Event Playback timestamp. Timestamps received during GI, event acquisition, cyclic updates, and background scans are played back with this tag. If the device provides data without a timestamp, the Timestamp tag value is set to 1899-12-30T00:00:00.000 and it is set to bad quality.
CurrentTimestamp*	Date	Same as Timestamp, but with unbuffered data.
QualityDesc	Byte	The Event Playback quality description field. Quality descriptions received during GI, event acquisition, cyclic updates, and background scans will be played back with this tag.
CurrentQualityDesc*	Byte	Same as QualityDesc, but with unbuffered data.
COT	Word	The Event Playback Cause of Transmission (COT) field. For COT values as specified in 60870-5-101, section 7.2.3, refer to the table below.
CurrentCOT*	Word	Same as COT, but with unbuffered data.



SUB-TYPE	Data Type	Description
Transient	Boolean	The Event Playback Transient bit associated with M_ST ASDU type. 0: Equipment is not in transient state. 1: Equipment is in transient state.
CurrentTransient*	Boolean	The same as the Transient, but with unbuffered data.
SQ	Char	The event playback sequence field associated with the M_IT ASDU type. A number between 0 and 31 represents the sequence notation.

\* For all the SUB-TYPES with “Current” in the name, cyclic or background data is reported if it has been recently received. If there has not been an update for a significant period of time and polled reads are enabled, a polled read with the read command ASDU is performed. This is accomplished by monitoring a tag's scan rate. Cyclic and background scan data is reported only if it is newer than a tag's scan rate. If the quality descriptor IV (invalid) bit is set, this tag shows bad quality.

**Caution** : Updating too many unbuffered Current tags at a fast rate can cause performance problems. Unbuffered tags perform physical reads from the IEC 60870-5 slave any time the tag has not been recently updated. Typically updates can be obtained using buffered tag types, which has minimal impact on performance.

**Cause of Transmission (COT)**

The Cause of Transmission field is enumerated according to the table below, which is based on the IEC 60870-5-101 specification.

**Note**: All causes of transmission (with the exception of requested transmission, background scan, and periodic / cyclic) are handled by event playback. Users can access events, general interrogation, and counter interrogation by utilizing the Value tag, Timestamp tag, QualityDesc tag, and COT tag. Requested, background scan, and cyclic unbuffered data can be accessed by utilizing the CurrentValue tag, CurrentTimestamp tag, CurrentQualityDesc tag, and CurrentCOT tag.

COT Value	Cause
0	Not used
1	Periodic, cyclic
2	Background scan
3	Spontaneous
4	Initialized
5	Request or requested
6	Activation
7	Activation confirmation
8	Deactivation
9	Deactivation confirmation
10	Activation termination
11	Return information caused by a remote command
12	Return information caused by a local command
13	File transfer

COT Value	Cause
14-19	Reserved for further compatible definitions
20	Interrogated by station interrogation
21-36	Interrogated by group 1-16 interrogation
37	Requested by general counter request
38-41	Requested by group 1-4 counter request
42-43	Reserved for further compatible definitions
44	Unknown type identification
45	Unknown Cause of Transmission
46	Unknown Common Address (ASDU)
47	Unknown Information Object Address (IOA)
48-63	For special use (private range)

### Buffered, Monitor ASDUs Examples

If a client needs access to double-point data in IOA 200 to capture events generated for this point. Events (COT = Spontaneous) are buffered so that all events are captured and played back to the client in chronological order. The following tags expose all the data related to IOA 200.

*M\_DP.200.VALUE*  
*M\_DP.200.TIMESTAMP*  
*M\_DP.200.COT*  
*M\_DP.200.QUALITYDESC*

ASDUs that contain structured data expose additional sub-types to expose the data. For example, integrated totals (M\_IT) contain a count (exposed in the VALUE sub-type) and sequence (exposed in the SQ sub-type). The following tags expose all data related to integrated total data in IOA 800.

*M\_IT.800.VALUE*  
*M\_IT.800.SQ*  
*M\_IT.800.TIMESTAMP*  
*M\_IT.800.COT*  
*M\_IT.800.QUALITYDESC*

### Unbuffered, Monitor ASDUs

If a client needs access to measured, floating-point data in IOA 700 and the slave is configured to transmit data periodically (COT = Cyclic) every 60 seconds. The clients should access “current” (unbuffered) tags. All ASDUs and sub-type are available with “current” sub-types, except integrated totals (M\_IT) because the protocol does not support polled reads or cyclic transmission of integrated totals. The following tags expose all the data related to IOA 700.

*M\_ME\_FV.700.CURRENTVALUE*  
*M\_ME\_FV.700.CURRENTTIMESTAMP*  
*M\_ME\_FV.700.CURRENTCOT*  
*M\_ME\_FV.700.CURRENTQUALITYDESC*

If the client update rate is greater than or equal to the slave’s configured cyclic transmission rate (60 seconds in this example), the driver only performs a polled read to determine the tag’s initial value. After the

initial read, the driver receives cyclic updates and provides the client the latest cyclic slave transmission. If the client update rate is faster than the slave cyclic transmission rate, the driver performs a polled read anytime the data is older than the client scan rate. For example, if the client update rate is 10 seconds, the driver performs a polled read every 10 seconds unless the cyclic data is less than 10 seconds old. Polled reads can be disabled in the device configuration.

### Control ASDUs

These Control ASDUs are write only.

Control TYPEID	Data Type	ASDU	Description
C_SC	Boolean	45	Single Command
C_DC	Char	46	Double Command. Possible values are 1 and 2. 0: Not Permitted 1: Determined State OFF 2: Determined State ON 3: Not Permitted
C_RC	Char	47	Regulating Step Command. Possible values are 1 and 2. 0: Not Permitted 1: Next Step LOWER 2: Next Step HIGHER 3: Not Permitted
C_SE_NV	Float	48	Measured Value, Normalized Value: This represents a value between -1 and 0.999969482421875. Values outside the range fail and report bad quality.
C_SE_SV	Short	49	Set point command, scaled value.
C_SE_FV	Float	50	Set point command, Short Floating Point Number.
C_BO	DWord	51	Bit string of 32 bits. Does not support bit indexing.

### SUB-TYPE

The SUB-TYPES that are available are determined by the TYPEID. Control TYPEIDs can only use control SUB-TYPES. For more information, refer to the table below.

SUB-TYPE	Data Type	Description
SelectExecuteValue	See TYPEID	The two-step write value. Writing to this causes the associated command ASDU to write to a device using select-and-execute. In this mode, writes occur with a dual command/response from the device.
DirectValue	See TYPEID	The single-step write value. Writing to this causes the associated command ASDU to write to a device using direct execute. In this mode, writes occur with a single command/response with the device.

### OPTIONAL

The properties below can be included with select Control TYPEIDs, but are not required. These allow access to additional Control SUB-TYPE options. Optional addressing can be used in any order.

OPTION	Applicable TypeIDs	Description
ShortPulse	C_SC C_DC C_RC	Control ASDU includes ShortPulse QoC. Only one of ShortPulse, LongPulse, or Persistent may be used at a time. Not specifying allows the device to use its default setting.
LongPulse	C_SC C_DC C_RC	Control ASDU includes LongPulse QoC. Only one of ShortPulse, LongPulse, or Persistent may be used at a time. Not specifying allows the device to use its default setting.
Persistent	C_SC C_DC C_RC	Control ASDU includes Persistent QoC. Only one of ShortPulse, LongPulse, or Persistent may be used at a time. Not specifying allows the device to use its default setting.

“No additional definition” (value of 0) is used for command tags that support the Qualifier of Command (QoC) syntax, but have no QoC option is specified.

### Control ASDUs Examples

If a client must control single-point data monitored with IOA 100, it can be accomplished with control ASDUs. In 60870-5-101, monitor IOAs are read-only (used to transfer data from the slave to the master) and cannot be written. The slave; however, may be configured to expose a single control IOA at point 2100 such that it correlates to the monitor IOA for the same point. When the master changes the value of IOA 2100, the change is reflected in the monitored single-point IOA 100. The following tags can be used to send control ASDUs to the slave for single-point 2100.

*C\_SC.2100.SELECTEXECUTEVALUE*  
*C\_SC.2100.DIRECTVALUE*

The client should use the tag that represents the desired write strategy. In addition to the write strategy, an optional write qualifier can be used to specify the slave's behavior. The following tags represent several options that can adjust the write qualifier. The write qualifier can be used with C\_SC, C\_DC, and C\_RC control ASDUs. All write qualifiers are supported for both the DIRECTVALUE and SELECTEXECUTEVALUE sub-types.

*C\_SC.2100.SELECTEXECUTEVALUE.PERSISTENT*  
*C\_SC.2100.DIRECTVALUE.LONGPULSE*  
*C\_SC.2100.DIRECTVALUE.PERSISTENT*  
*C\_SC.2100.SELECTEXECUTEVALUE.SHORTPULSE*

### Command-Based Tag Addressing

Tag addressing takes the form *TYPE.COMMAND*, where:

- *TYPE* is the string: Command.
- *COMMAND* is the command to be run.

#### COMMAND

An example command address is: “Command.GI”.

SUB-TYPE	Description
Command.GI	Global General Interrogation (GI) is issued when a client writes any value to this tag. This asks a device for any data assigned to general interrogation.
Command.CI	Global Counter Interrogation (CI) is issued when a client writes any value to this tag. This asks any integrated totals.
Command.FreezeCounters	Global Freeze Counter Interrogation is issued when a client writes any value to this tag. This stops counting integrated totals.
Command.ResetCounters	Global Reset Counter Interrogation is issued when a client writes any value to this tag. This returns all integrated totals to 0.
Command.FreezeResetCounters	Global Freeze and Reset Counter Interrogation is issued when a client writes any value to this tag. This stops all integrated totals and returns all integrated totals to 0.
Command.TimeSync	Time Sync is issued when a client writes any value to this tag. This synchronizes the master clock with the slave clock.
Command.TestProcedure	Test Procedure is issued when a client writes any value to this tag. This checks a device to see if it is connected.

## Error Descriptions

---

The following messages may be generated. Click on the link for a description of the message.

[Device <device name> failed to complete clock synchronization.](#)

[Device <device name> failed to complete counter interrogation.](#)

[Device <device name> failed to complete <procedure name>.](#)

[Device <device name> is not responding.](#)

[Error loading XML file. <Reason>.](#)

[Error loading XML file for channel.](#)

[Error loading XML file for device. Common address is invalid.](#)

[Error loading XML file for device. Link address is invalid.](#)

[Error loading XML file for device. Link and common address pair must be unique.](#)

[Error loading XML file for device. When the transmission mode is balanced, all devices must have the same link address. The link address of device is <link address>.](#)

[Read failed on tag <tag name>.](#)

[Read failed on tag <tag name> due to data type mismatch.](#)

[Reverting channel <channel> link mode to Unbalanced Transmission. Balanced Transmission is not compatible with communication serialization or port sharing.](#)

[Too many events received.](#)

[Unable to write to address <address> on device <device>.](#)

[Write failed on tag <tag name>.](#)

### **Device <device name> failed to complete clock synchronization.**

---

#### **Error Type:**

Warning

#### **Possible Cause:**

1. Communications were lost.
2. Communications timed out.

#### **Solution:**

1. Service the device connection.
2. Increase the value of the device's clock synchronization request timeout setting.

### **Device <device name> failed to complete counter interrogation.**

---

#### **Error Type:**

Serious

#### **Possible Cause:**

1. Communications were lost.
2. Communications timed out. This can occur when the time to send all counter interrogation data exceeds the Interrogation Request Timeout.

**Solution:**

1. Service the device connection.
2. Increase the value of the device interrogation request timeout setting.

**Device <device name> failed to complete general interrogation.**

---

**Error Type:**

Warning

**Possible Cause:**

1. Communications were lost.
2. Communications timed out. This can occur when the time to send all general interrogation data exceeds the Interrogation Request Timeout.

**Solution:**

1. Service the device connection.
2. Increase the value of the device's interrogation request timeout setting.

**Device <device name> is not responding.**

---

**Error Type:**

Serious

**Possible Cause:**

1. The connection between the device and the host PC is broken.
2. The communications parameters for the connection are incorrect.
3. The named device may have been assigned an incorrect network ID.
4. A device on the channel is unresponsive, due to improper timing settings or a broken communications link.
5. There are multiple channels using DNS host names that resolve to the same IP address.
6. The response from the device took longer to receive than the amount of time specified in the Request Timeout device setting

**Solution:**

1. Verify the cabling between the IEC 60870 master and the IEC 60870 slave device.
2. Verify that the specified communications parameters match those of the device.
3. Verify that the network ID given to the named device matches that of the actual device.
4. Locate the unresponsive device and correct the timing settings or fix the broken communications link.
5. Ensure that all channels are using a unique Destination Host.
6. Increase the Request Timeout setting so that the entire response can be handled.

---

**Error loading XML file for channel <channel name>. Unbalanced mode does not support a link address size of zero.**

---

**Error Type:**

Error

**Possible Cause:**

The XML project being loaded specifies a link address size of 0. A link address is required when the link mode is set to Unbalanced Transmission.

**Solution:**

1. Change the link mode to Balanced Transmission.
2. Increase Link Address Size to 1 or 2 octets.

**See Also:**

[Channel Setup](#)

---

**Error loading XML file for device <device name>. When the transmission mode is balanced, all devices must have the same link address. The link address of device <device name> is <link address>.**

---

**Error Type:**

Error

**Possible Cause:**

Devices don't have the same link address when the link mode is Balanced Transmission. This link mode requires that the channel contain one (1) master and one (1) slave.

**Solution:**

Verify that every device in the channel has the same link address, but unique common addresses.

**See Also:**

[Channel Setup](#)

---

**Error loading XML file for device <device name>. Common Address is invalid. Valid range is <low limit> to <high limit>.**

---

**Error Type:**

Error

**Possible Cause:**

The XML project being loaded specifies a common address that is not valid with the Common Address Size setting.

**Solution:**



1. Increase the Common Address Size. Possible values are 1 octet or 2 octets.
2. Update the common address so that it is within the allowed range.

**Error loading XML file for device <device name>. The link address (<link address>) and common address (<common address>) pair is already in use on device <device name>. Link and common address pairs must be unique.**

---

**Error Type:**

Error

**Possible Cause:**

More than one device in the channel is using the same link address / common address pair.

**Solution:**

Verify that every device in the channel has a unique link address /common address pair.

**See Also:**

[Channel Setup](#)

**Error loading XML file for device <device name>. Link address is invalid. Valid range is <low limit> to <high limit>.**

---

**Error Type:**

Error

**Possible Cause:**

The XML project being loaded specifies a link address that is not valid with the Link Address Size setting.

**Solution:**

1. Increase the Link Address Size. Possible values are 1 octet or 2 octets.
2. Update the link address so that it is within the allowed range.

**Read failed on tag <tag name>.**

---

**Error Type:**

Warning

**Possible Cause:**

A device is connected, but the tag is not appropriate for the device. For example, the ASDU type or IOA may not be appropriate for the data on the device.

**Solution:**

Correct the tag address to match the device's configuration.

**See Also:**

## [Address Descriptions](#)

---

### **Read failed on tag <tag name> due to data type mismatch.**

**Error Type:**

Warning

**Possible Cause:**

Tag ASDU type is incorrect. This error occurs when data is received from the device at this IOA, but it represents a different ASDU type.

**Solution:**

Verify or correct the ASDU type. Correct the tag address to match the device configuration.

**See Also:**

[Address Descriptions](#)

---

### **Reverting channel <channel name> link mode to Unbalanced Transmission. Balanced Transmission is not compatible with communication serialization or port sharing.**

**Error Type:**

Warning

**Possible Cause:**

1. The channel was assigned to a virtual network from the Advanced Channel Settings property sheet with the link mode set to Balanced Transmission.
2. The channel was added to a com port network that is in used by at least one other channel with the link mode set to Balanced Transmission.

**Solution:**

The Balanced Transmission link mode is designed to work with a point-to-point serial network configuration. In a point-to-point network, the master and slave may transmit at any time. This link mode is not compatible with communication serialization or port sharing because there is no way to avoid transmit collisions when multiple slaves are connected.

---

### **Too many events received in <device name>; the buffer has overflowed. Increase the event playback buffer size.**

**Error Type:**

Warning

**Possible Cause:**

The slave is producing events faster than the client can consume them.

**Solution:**

Increase the event playback buffer size. The per-item event buffer size and playback rate should be set to handle the data at the highest per-item event transmission speed. As a simplified example, assume a slave may transmit up to 100 events, all at once, every 5 seconds for an item. The buffer size and playback rate

must be set so all 100 events can be stored and played back before more arrive. The buffer size must be set to at least 100 events and the playback rate must be 50 milliseconds to store and then playback 100 events every 5 seconds.

---

**Unable to write to address <address> on device <device>.**

---

**Error Type:**

Warning

**Possible Cause:**

1. Incorrect common address; the device is not responding.
2. The point does not exist in the slave.
3. The address has an incorrect data type.

**Solution:**

1. Correct the common address to match the slave.
2. Correct the Information Object Address(IOA).
3. Correct the data type for the address.

**See Also:**

[Address Descriptions](#)

---

**Write failed on tag <tag name>.**

---

**Error Type:**

Warning

**Possible Cause:**

Attempted to write to a tag outside the supported range.

**Solution:**

Note the minimum and maximum values and limit writes within the supported range.

**See Also:**

[Address Descriptions](#)

## IEC 60870-5-101 Interoperability Guide

The following reference material is from the International Electrotechnical Commission (IEC).

### Interoperability

This companion standard presents sets of parameters and alternatives from which subsets have to be selected to implement particular telecontrol systems. Certain parameter values, such as the number of octets in the COMMON ADDRESS of ASDUs represent mutually exclusive alternatives. This means that only one value of the defined parameters is admitted per system. Other parameters, such as the listed set of different process information in command and in monitor direction allow the specification of the complete set or subsets, as appropriate for given applications. This Clause summarizes the parameters of the previous Clauses to facilitate a suitable selection for a specific application. If a system is composed of equipment stemming from different manufacturers, it is necessary that all partners agree on the selected parameters.

The selected parameters should be marked in the white boxes as follows:

<input type="checkbox"/>	Function or ASDU is not used
<input checked="" type="checkbox"/>	Function or ASDU is used as standardized (default)
<input checked="" type="checkbox"/>	Function or ASDU is used in reverse mode
<input checked="" type="checkbox"/>	Function or ASDU is used in standard and reverse mode

The possible selection (blank, X, R, or B) is specified for each specific clause or parameter.

● **Note:** In addition, the full specification of a system may require individual selection of certain parameters for certain parts of the system, such as the individual selection of scaling factors for individually addressable measured values.

### System or device

(system-specific parameter, indicate the station's function by marking one of the following with 'X')

<input type="checkbox"/>	System definition
<input checked="" type="checkbox"/>	Controlling station definition (Master)
<input type="checkbox"/>	Controlling station definition (Slave)

### Network configuration

(network-specific parameter, all configurations that are used are to be marked with an "X")

<input checked="" type="checkbox"/>	Point-to-point
<input checked="" type="checkbox"/>	Multiple point-to-point
<input type="checkbox"/>	Multipoint-party line
<input type="checkbox"/>	Multipoint-star

### Physical layer

(network-specific parameter, all interfaces and data rates that are used are to be marked with an "X")

**Transmission speed (control direction)**

Unbalanced interchange Circuit V.24/V.28 Standard		Unbalanced interchange Circuit V.24/V.28 Recommended if >1 200 bit/s		Balanced interchange Circuit X.24/X.27	
<input type="checkbox"/>	100 bit/s	<input checked="" type="checkbox"/>	2400 bit/s	<input checked="" type="checkbox"/>	2400 bit/s
<input type="checkbox"/>	200 bit/s	<input checked="" type="checkbox"/>	4800 bit/s	<input checked="" type="checkbox"/>	4800 bit/s
<input checked="" type="checkbox"/>	300 bit/s	<input checked="" type="checkbox"/>	9600 bit/s	<input checked="" type="checkbox"/>	9600 bit/s
<input checked="" type="checkbox"/>	600 bit/s			<input checked="" type="checkbox"/>	19200 bit/s
<input checked="" type="checkbox"/>	1200 bit/s			<input checked="" type="checkbox"/>	38400 bit/s
				<input checked="" type="checkbox"/>	56000 bit/s
				<input type="checkbox"/>	64000 bit/s

**Transmission speed (monitor direction)**

Unbalanced interchange Circuit V.24/V.28 Standard		Unbalanced interchange Circuit V.24/V.28 Recommended if >1 200 bit/s		Balanced interchange Circuit X.24/X.27	
<input type="checkbox"/>	100 bit/s	<input checked="" type="checkbox"/>	2400 bit/s	<input checked="" type="checkbox"/>	2400 bit/s
<input type="checkbox"/>	200 bit/s	<input checked="" type="checkbox"/>	4800 bit/s	<input checked="" type="checkbox"/>	4800 bit/s
<input checked="" type="checkbox"/>	300 bit/s	<input checked="" type="checkbox"/>	9600 bit/s	<input checked="" type="checkbox"/>	9600 bit/s
<input checked="" type="checkbox"/>	600 bit/s	<input type="checkbox"/>		<input checked="" type="checkbox"/>	19200 bit/s
<input checked="" type="checkbox"/>	1200 bit/s	<input type="checkbox"/>		<input checked="" type="checkbox"/>	38400 bit/s
<input type="checkbox"/>		<input type="checkbox"/>		<input checked="" type="checkbox"/>	56000 bit/s
<input type="checkbox"/>		<input type="checkbox"/>		<input type="checkbox"/>	64000 bit/s

**Link layer**

(network-specific parameter, all options that are used are to be marked with an "X". Specify the maximum frame length. If a non-standard assignment of class 2 messages is implemented for unbalanced transmission, indicate the type ID and COT of all messages assigned to class 2.)

Frame format FT 1.2, single character 1 and the fixed time out interval are used exclusively in this companion standard.

**Link transmission procedure**

<input checked="" type="checkbox"/>	Balanced transmission
<input checked="" type="checkbox"/>	Unbalanced transmission

**Address field of link**

<input checked="" type="checkbox"/>	Not present (balanced transmission only)
<input checked="" type="checkbox"/>	One octet
<input checked="" type="checkbox"/>	Two octets
<input type="checkbox"/>	Structured
<input type="checkbox"/>	Unstructured

### Frame length

<b>255</b>	Maximum length L (control direction)
<b>255</b>	Maximum length L (monitor direction)
<b>Configurable</b>	Time during which repetitions are permitted (Trp) or number of repetitions

When using an unbalanced link layer, the following ASDU types are returned in class 2 messages (low priority) with the indicated causes of transmission:

The standard assignment of ASDUs to class 2 messages is used as follows:

Type identification	Cause of transmission
9, 11, 13, 21	<1>

A special assignment of ASDUs to class 2 messages is used as follows:

Type identification	Cause of transmission

**Note:** In response to a class 2 poll, a controlled station may respond with class 1 data when there is no class 2 data available.

### Application Layer

Transmission mode for application data

Mode 1 (Least significant octet first), as defined in clause 4.10 of IEC 60870-5-4, is used exclusively in this companion standard.

### Common address of ASDU

(system-specific parameter, all configurations that are used are to be marked "X")

<input checked="" type="checkbox"/>	One octet
<input checked="" type="checkbox"/>	Two octets

### Information object address

(system-specific parameter, all configurations that are used are to be marked "X")

<input checked="" type="checkbox"/>	One octet
<input checked="" type="checkbox"/>	Two octets
<input checked="" type="checkbox"/>	Three octets
<input type="checkbox"/>	Structured
<input type="checkbox"/>	Unstructured

### Cause of transmission

(system-specific parameter, all configurations that are used are to be marked "X")

<input checked="" type="checkbox"/>	One octet
<input checked="" type="checkbox"/>	Two octets (with originator address)

**Note:** Set to zero in case of no originator address.

### Selection of standard ASDUs

#### Process information in monitor direction

(station-specific parameter, mark each type ID with an "X" if it is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

<input checked="" type="checkbox"/>	<1>	:=	Single-point information	M_SP_NA_1
<input checked="" type="checkbox"/>	<2>	:=	Single-point information with time tag	M_SP_TA_1
<input checked="" type="checkbox"/>	<3>	:=	Double-point information	M_DP_NA_1
<input checked="" type="checkbox"/>	<4>	:=	Double-point information with time tag	M_DP_TA_1
<input checked="" type="checkbox"/>	<5>	:=	Step position information	M_ST_NA_1
<input checked="" type="checkbox"/>	<6>	:=	Step position information with time tag	M_ST_TA_1
<input checked="" type="checkbox"/>	<7>	:=	Bitstring of 32 bit	M_BO_NA_1
<input checked="" type="checkbox"/>	<8>	:=	Bitstring of 32 bit with time tag	M_BO_TA_1
<input checked="" type="checkbox"/>	<9>	:=	Measured value, normalized value	M_ME_NA_1
<input checked="" type="checkbox"/>	<10>	:=	Measured value, normalized value with time tag	M_ME_TA_1
<input checked="" type="checkbox"/>	<11>	:=	Measured value, scaled value	M_ME_NB_

			1
X	<12>	:= Measured value, scaled value with time tag	M_ME_TB_1
X	<13>	:= Measured value, short floating point value	M_ME_NC_1
X	<14>	:= Measured value, short floating point value with time tag	M_ME_TC_1
X	<15>	:= Integrated totals	M_IT_NA_1
X	<16>	:= Integrated totals with time tag	M_IT_TA_1
	<17>	:= Event of protection equipment with time tag	M_EP_TA_1
	<18>	:= Packed start events of protection equipment with time tag	M_EP_TB_1
	<19>	:= Packed output circuit information of protection equipment with time tag	M_EP_TC_1
	<20>	:= Packed single-point information with status change detection	M_PS_NA_1
X	<21>	:= Measured value, normalized value without quality descriptor	M_ME_ND_1
X	<30>	:= Single-point information with time tag CP56Time2a	M_SP_TB_1
X	<31>	:= Double-point information with time tag CP56Time2A	M_DP_TB_1
X	<32>	:= Step position information with time tag CP56Time2A	M_ST_TB_1
X	<33>	:= Bitstring of 32 bit with time tag CP56Time2A	M_BO_TB_1
X	<34>	:= Measured value, normalized value with time tag CP56Time2A	M_ME_TD_1
X	<35>	:= Measured value, scaled value with time tag CP56Time2A	M_ME_TE_1
X	<36>	:= Measured value, short floating point value with time tag CP56Time2A	M_ME_TF_1
X	<37>	:= Integrated totals with time tag CP56Time2A	M_IT_TB_1
	<38>	:= Event of protection equipment with time tag CP56Time2A	M_EP_TD_1
	<39>	:= Packed start events of protection equipment with time tag CP56time2A	M_EP_TE_1
	<40>	:= Packed output circuit information of protection equipment with time tag CP56Time2a	M_EP_TF_1

Either ASDUs of the set <2>, <4>, <6>, <8>, <10>, <12>, <14>, <16>, <17>, <18>, <19>, or of the set <30 -40> are used.

### Process information in control direction

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions)

X	<45>	:= Single command	C_SC_NA_1
X	<46>	:= Double command	C_DC_NA_1
X	<47>	:= Regulating step command	C_RC_NA_1



X	<48>	:=	Set-point command, normalized value	C_SE_NA_1
X	<49>	:=	Set-point command, scaled value	C_SE_NB_1
X	<50>	:=	Set-point command, short floating point value	C_SE_NC_1
X	<51>	:=	Bitstring of 32 bit	C_BO_NA_1

### System information in monitor direction

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions)

X	<70>	:=	End of initialization	M_EI_NA_1
---	------	----	-----------------------	-----------

### System information in control direction

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions)

X	<100>	:=	Interrogation command	C_IC_NA_1
X	<101>	:=	Counter interrogation command	C_CI_NA_1
X	<102>	:=	Read command	C_RD_NA_1
X	<103>	:=	Clock synchronization command	C_CS_NA_1
X	<104>	:=	Test command	C_TS_NB_1
	<105>	:=	Reset process command	C_RP_NC_1
X	<106>	:=	Delay acquisition command	C_CD_NA_1

### Parameter in control direction

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions)

	<110>	:=	Parameter of measured value, normalized value	P_ME_NA_1
	<111>	:=	Parameter of measured value, scaled value	P_ME_NB_1
	<112>	:=	Parameter of measured value, short floating point value	P_ME_NC_1
	<113>	:=	Parameter activation	P_AC_NA_1

### File transfer

(station-specific parameter, mark each Type ID 'X' if it is only used in the standard direction, 'R' if only used in the reverse direction, and 'B' if used in both directions)

	<120>	:=	File ready	F_FR_NA_1
	<121>	:=	Section ready	F_SR_NA_1

<122>	:=	Call directory, select file, call file, call section	F_SC_NA_1
<123>	:=	Last section, last segment	F_LS_NA_1
<124>	:=	Ack file, ack section	F_AF_NA_1
<125>	:=	Segment	F_SG_NA_1
<126>	:=	Directory	F_DR_TA_1

## Type identification and cause of transmission assignments

(station-specific parameters)

Shaded boxes are not required.

Blank = function or ASDU is not used.

Mark type identification/cause of transmission combinations:

“X” if used only in the standard direction;

“R” if used only in the reverse direction;

“B” if used in both directions.

Type Identification		Cause of Transmission																		
		Periodic, cyclic	Background scan	Spontaneous	Initialized	Request or requested	Activation	Activation confirmation	Deactivation	Deactivation confirmation	Activation termination	Return info caused by a remote cmd	Return info caused by a local cmd	File transfer	Interrogated by group <number>	Request by group <n> counter request	Unknown type identification	Unknown cause of transmission	Unknown common address of ASDU	Unknown information object address
		1	2	3	4	5	6	7	8	9	10	11	12	13	20-36	37-41	44	45	46	47
<1>	M_SP_NA_1			X		X						X	X		X					
<2>	M_SP_TA_1			X		X						X	X		X					
<3>	M_DP_NA_1			X		X						X	X		X					
<4>	M_DP_TA_1			X		X						X	X							
<5>	M_ST_NA_1			X		X						X	X		X					

<6>	M_ST_TA_1			X		X						X	X						
<7>	M_BO_NA_1			X		X									X				
<8>	M_BO_TA_1			X		X													
<9>	C_ME_NA_1			X		X									X				
<10>	M_ME_TA_1			X		X													
<11>	M_ME_NB_1			X		X									X				
<12>	M_ME_TB_1			X		X													
<13>	M_ME_NC_1			X		X									X				
<14>	M_ME_TC_1			X		X													
<15>	M_IT_NA_1			X															
<16>	M_IT_TA_1			X															
<17>	M_EP_TA_1																		
<18>	M_EP_TB_1																		
<19>	M_EP_TC_1																		
<20>	M_PS_NA_1																		
<21>	M_ME_ND_1			X		X									X				
<30>	M_SP_TB_1			X		X						X	X		X				
<31>	M_DP_			X		X						X	X						

	TB_1																		
<32>	M_ST_TB_1			X		X						X	X						
<33>	M_BO_TB_1			X		X													
<34>	M_ME_TD_1			X		X													
<35>	M_ME_TE_1			X		X													
<36>	M_ME_TF_1			X		X													
<37>	M_IT_TB_1			X															
<38>	M_EP_TD_1																		
<39>	M_EP_TE_1																		
<40>	M_EP_TF_1																		
<45>	C_SC_NA_1						X												
<46>	C_DC_NA_1						X												
<47>	C_RC_NA_1						X												
<48>	C_SE_NA_1						X												
<49>	C_SE_NB_1						X												
<50>	C_SE_NC_1						X												
<51>	C_BO_NA_1						X												
<70>	M_EI_NA_1*																		
<100>	C_IC_NA_1						X												
<101>	C_CL_NA_1						X												

<102>	C_RD_NA_1																	
<103>	C_CS_NA_1					X												
<104>	C_TS_NA_1					X												
<105>	C_RP_NA_1																	
<106>	C_CD_NA_1					X												
<110>	M_ME_NA_1																	
<111>	M_ME_NB_1																	
<112>	P_ME_NC_1																	
<113>	P_AC_NA_1																	
<120>	F_FR_NA_1																	
<121>	F_SR_NA_1																	
<122>	F_SC_NA_1																	
<123>	F_LS_NA_1																	
<124>	F_AF_NA_1																	
<125>	F_SG_NA_1																	
<126>	F_DR_TA_1*																	

\* Note: Blank or X only.

### Basic Application Functions

#### Station initialization

(station-specific parameter, mark with an "X" if function is used)

<b>X</b>	Remote initialization
----------	-----------------------

**Cyclic data transmission**

(station-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

<b>X</b>	Cyclic data transmission
----------	--------------------------

**Read procedure**

(station-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

<b>X</b>	Read procedure
----------	----------------

**Spontaneous transmission**

(station-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

<b>X</b>	Spontaneous transmission
----------	--------------------------

**Double transmission of information objects with cause of transmission spontaneous**

(Station-specific parameter, mark each information type with an "X" where both a type ID without time and corresponding type ID with time are issued in response to a single spontaneous change of a monitored object.)

The following type identifications may be transmitted in succession caused by a single status change of an information object. The particular information object addresses for which double transmission is enabled are defined in a project-specific list.

<input type="checkbox"/>	Single-point information	M_SP_NA_1, M_SP_TA_1, M_SP_TB_1, M_PS_NA_1
<input type="checkbox"/>	Double-point information	M_DP_NA_1, M_DP_TA_1, M_DP_TB_1
<input type="checkbox"/>	Step position information	M_ST_NA_1, M_ST_TA_1, M_ST_TB_1
<input type="checkbox"/>	Bitstring of 32 bit	M_BO_NA_1, M_BO_TA_1, M_BO_TB_1*
<input type="checkbox"/>	Measured value, normalized value	M_ME_NA_1, M_ME_TA_1, M_ME_ND_1, M_ME_TD_1
<input type="checkbox"/>	Measured value, scaled value	M_ME_NB_1, M_ME_TB_1, M_ME_TE_1
<input type="checkbox"/>	Measured value, short floating point number	M_ME_NC_1, M_ME_TC_1, M_ME_TF_1

\***Note:** If defined for a specific project, see 7.2.1.1.

**Station interrogation**

(station-specific parameter, mark each Type ID "X" if it is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

<b>X</b>	global		
<input type="checkbox"/>	group 1	<input type="checkbox"/>	group 13

<input type="checkbox"/>	group 2	<input type="checkbox"/>	group 8	<input type="checkbox"/>	group 14
<input type="checkbox"/>	group 3	<input type="checkbox"/>	group 9	<input type="checkbox"/>	group 15
<input type="checkbox"/>	group 4	<input type="checkbox"/>	group 10	<input type="checkbox"/>	group 16
<input type="checkbox"/>	group 5	<input type="checkbox"/>	group 11		
<input type="checkbox"/>	group 6	<input type="checkbox"/>	group 12		

**\*Note:** Addresses per group have to be defined.

**Clock synchronization**

(station-specific parameter, mark each Type ID "X" if it is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

<input checked="" type="checkbox"/>	Clock synchronization
<input type="checkbox"/>	Day of week used
<input type="checkbox"/>	RES1, GEN (time tag substituted/ not substituted) used
<input type="checkbox"/>	SU-bit (summertime) used

**Command transmission**

(object-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

<input checked="" type="checkbox"/>	Direct command transmission
<input checked="" type="checkbox"/>	Direct set point command transmission
<input checked="" type="checkbox"/>	Select and execute command
<input checked="" type="checkbox"/>	Select and execute set point command
<input type="checkbox"/>	C_SE_ACTTERM used
<input checked="" type="checkbox"/>	No additional definition
<input checked="" type="checkbox"/>	Short pulse duration (duration determined by a system parameter in the outstation)
<input checked="" type="checkbox"/>	Long pulse duration (duration determined by a system parameter in the outstation)
<input checked="" type="checkbox"/>	Persistent output

**Transmission of integrated tools**

(station-specific parameter, mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

<input type="checkbox"/>	Mode A: Local freeze with spontaneous
<input type="checkbox"/>	Mode B: Local freeze with counter
<input type="checkbox"/>	Mode C Freeze and transmit by counter interrogation
<input type="checkbox"/>	Mode C Freeze by counter-interrogation command, frozen values reported

<input checked="" type="checkbox"/>	Counter read
<input checked="" type="checkbox"/>	Counter freeze without reset
<input checked="" type="checkbox"/>	Counter freeze with reset
<input checked="" type="checkbox"/>	Counter reset
<input checked="" type="checkbox"/>	General request counter
<input type="checkbox"/>	Request counter group 1
<input type="checkbox"/>	Request counter group 2
<input type="checkbox"/>	Request counter group 3
<input type="checkbox"/>	Request counter group 4

**Parameter loading**

(object-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

<input type="checkbox"/>	Threshold value
<input type="checkbox"/>	Smoothing factor
<input type="checkbox"/>	Low limit for transmission of measured value
<input type="checkbox"/>	High limit for transmission of measured value

**Parameter activation**

(object-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

<input type="checkbox"/>	Activation / deactivation of persistent cyclic or periodic transmission of the addressed object
--------------------------	---

**Test procedure**

(station-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

<input checked="" type="checkbox"/>	Test procedure
-------------------------------------	----------------

**File transfer**

(station-specific parameter, mark with an "X" if function is used)

File transfer in monitor direction

<input type="checkbox"/>	Transparent file
<input type="checkbox"/>	Transmission of disturbance data of protection
<input type="checkbox"/>	Transmission of sequences of events
<input type="checkbox"/>	Transmission of sequences of recorded analog values



File transfer in control direction

Transparent file

**Background scan**

(station-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

Background scan

**Acquisition of transmission delay**

(station-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

Acquisition of transmission delay

# Index

## 6

60870 Settings 11

## A

Absolute 17

Acquisition of transmission delay 49

Address Descriptions 23

Address field of link 37

Address Settings 12

Application Layer 38

ASDU 36

Attempt Count 19

## B

Background scan 49

Balanced Transmission 12

Boolean 22

Buffer Size 12

Buffered, Monitor ASDUs 26

Byte 22

## C

Cause of transmission 39

Cause of Transmission 12, 25

Channel Assignment 14

Char 22

Class 1 Pending Delay 13

Class 1 Poll Interval 12

Class 2 Pending Delay 13

Class 2 Poll Interval 13

Clock synchronization 47

COMMAND 28

Command Based 28  
Command transmission 47  
Common Address 18  
Common address of ASDU 38  
Communications 17  
Control ASDUs 28  
COT 12, 25  
Counter Interrogation 20  
Cyclic data transmission 46

## D

Data Collection 15  
Data Types Description 22  
Date 22  
Daylight Saving Time 17  
Delay Acquisition 19  
Demote on Failure 16  
Demotion Period 16  
Description 14  
Device <device name> is not responding. 31  
Device failed to complete clock synchronization. 30  
Device failed to complete counter interrogation. 30  
Device failed to complete general interrogation. 31  
Device Properties — Auto-Demotion 16  
Device Properties — General 13  
Discard Requests when Demoted 16  
Do Not Scan, Demand Poll Only 15  
Double transmission 46  
Driver 14  
DWord 22

## E

Error Descriptions 30  
Error loading XML file for channel. Unbalanced mode does not support a link address size of zero. 32  
Error loading XML file for device. Common Address is invalid. Valid range is <low limit> to <high limit>. 32  
Error loading XML file for device. Link address is invalid. Valid range is <low limit> to <high limit>. 33

Error loading XML file for device. The link address and common address pair is already in use on device.  
Link and common address pairs must be unique. 33

Error loading XML file for device. When the transmission mode is balanced, all devices must have the  
same link address. The link address of device is link address. 32

Ethernet Encapsulation 5

Event Playback 20

## **F**

File transfer 41

First Char Wait 13

Float 22

Frame length 38

## **G**

General Interrogation 20

## **H**

Help Contents 4

## **I**

ID 14

IEC 36

IEC 60870-5-101 Interoperability Guide 36

Incremental Timeout 13

Information object address 39

Information Object Address (IOA) 23

Initial Updates from Cache 16

Initialization 19

Interrogation Attempt Count 19

Interrogation Request Timeout 19

Interval 17, 20

IOA 23

**L**

Link layer 37  
Link Mode 12  
Link Settings 12  
Link transmission procedure 37  
Long 22

**M**

Master 36  
Messages 30  
Method 17  
Model 14  
Monitor ASDUs 23

**N**

Name 14  
Network configuration 36

**O**

OnPoll 17  
OPTIONAL 27  
Originator Address 12  
Overview 4

**P**

Parameter 41  
Parameter activation 48  
Parameter loading 48  
Periodic Settings 19  
Physical layer 36  
Playback 20  
Playback Buffer Size 20  
Playback Rate 20

Polled Reads 18  
Process information 39  
Protocol 5

## R

Read failed on tag due to data type mismatch. 34  
Read failed on tag. 33  
Read procedure 46  
Redundancy 21  
Request All Data at Scan Rate 15  
Request Data No Faster than Scan Rate 15  
Request Timeout 19  
Respect Client-Specified Scan Rate 15  
Respect Tag-Specified Scan Rate 16  
Reverting channel <channel name> link mode to Unbalanced Transmission. Balanced Transmission is not compatible with communication serialization or port sharing. 34

## S

Scan Mode 15  
Setup 5  
Short 22  
Simulated 15  
Size Settings 13  
Slave 36  
Socket 5  
Spontaneous transmission 46  
Standard ASDUs 39  
Station initialization 45  
Station interrogation 46  
SUB-TYPE 27  
SUBTYPE 24  
System information 41

## T

Test procedure 48

Test Procedure 20  
the buffer has overflowed. Increase the event playback buffer size. 34  
Time Sync Initialization 19  
Time Sync Threshold 17  
Time Synchronization 16  
Time Zone 17  
Timeouts to Demote 16  
Timing 21  
Timing Settings 13  
Too many events received in device name 34  
Transmission of integrated tools 47  
Transmission speed 37  
Type identification 38  
TYPEID 23

## **U**

Unable to write to address <address> on device <device>. 35  
Unbalanced Transmission 12  
Unbuffered, Monitor ASDUs 26

## **W**

Word 22  
Write failed on tag. 35