IEC870IP – CitectSCADA Driver for IEC60870-5-104 Protocol

CitectSCADA version 5.xx and 6.xx
User information and design

Driver version history

Version	Modified By	Details
2.46.00(+104v1.00)	MHW	Add IEC60870-5-104 support.
2.46.00(+104v1.01)	MHW	Add Software Protection support.
2.46.00(+104v1.01)	JN – Aug 2005	Driver document updated

Table of Contents

1. Introduction

The “IEC870IP” CitectSCADA driver is a 32-bit driver that provides communication services for devices that use the IEC870-5-104 Communication Protocol.

Many protocol configuration parameters are supported, and all of the commonly used functions are implemented. See the IEC Interoperability List in appendix A for details.

The driver supports multiple serial channels and multiple devices on each serial channel. In addition, support is provided for installations where the IEC Common address is different from the Link address.

CitectSCADA’s underlying “TCP/IP” driver is used for TCP/IP communication for IEC870-5-104.

1.1 Driver Operation

The driver employs “front-end / back-end” architecture, which means that a memory image of a current state device’s I/O data is maintained in the driver.

Updating of this memory image is not driven by CitectSCADA’s Read operations to the driver. When CitectSCADA performs a read operation, the value CitectSCADA receives comes from the IEC driver’s memory image. In addition to the current value of the points the IEC driver also maintains information about the online / offline status of devices.

However these two processes are not only linked by the memory image data - they are also linked by information about the validity of that data. When a unit goes off-line, all data is marked as “not topical”. Each datum then becomes “topical” once it has successfully been read from the device.

The driver is interrupt-driven, with an interrupt service routine called on data arrival. The interrupt routine does validity checking of the data, discarding any invalid data, then queues it for processing.

The next poll from CitectSCADA to the driver then triggers reading and processing of this queue. This is different from most CitectSCADA drivers that are either completely poll-driven, or switch off CitectSCADA polls to the driver completely. The problem with the latter approach is that interrupt routines become too long and complex, and too much code ends up in common between interrupt and normal processing - increasing the risk of conflicts.

Note: The outcome of this is that the “PollTime” parameter should NOT be made zero. It can comfortably be made a very small period though, as the polling is now an insignificant overhead - all it does is check for a non-empty receive queue and a non-empty send queue. “PollTime” defaults to 100 milliseconds.

2. Getting Started

The driver contains many configuration options. Fortunately, not all of these options need to be configured for basic communication functionality.

Ensure that configuration details for the RTU's are available – how both communications and protocol are configured. Due to the nature of the IEC protocol, differences in the configuration details may stop the RTU and CitectSCADA from communicating correctly.

NOTE: Familiarity with CitectSCADA is assumed. Please see your CitectSCADA documentation for more information about how to configure a CitectSCADA system.

This document also assumes the reader is familiar with the IEC870-1-105 protocol.

The list below provides a basic set of instructions for connecting an RTU to CitectSCADA via the "IEC870IP" driver.

2.1 Driver Installation

Run the installation program setup.exe on the distribution disk. This will install the driver executables and documentation onto a directory on your hard disk.

The following files will be copied to a 32-bit CitectSCADA installation's \CitectSCADA\bin\:

- IEC870IP.dll : Driver Dynamic Link Library
- IEC870IP.dbf: Driver configuration file
- Iecioa.dbf and IecUnit.dbf (note that these are sample configuration files only **and are not used by default**).

2.2 CitectSCADA I/O Forms Configuration

See the CitectSCADA documentation for more information about configuring IO Forms.

1. Make an entry in the CitectSCADA “Boards” form for each serial channel as follows:
   For Serial:
   - Board Type = Comx;
   - Address = 0;

   For TCPIP:
   - Board Type = TCPIP;
   - Address = 0;

2. Make an entry in the CitectSCADA “Ports” form for each serial channel as follows:
   For Serial:
   - Port Number = COM port number;
   - Baud Rate = Baud rate of connected devices, e.g. 9600;
   - Data Bits = 8
   - Stop Bits = 1
   - Parity = EVEN_P

   For TCPIP:
   - Special Opt: IP Address, Port Number TCP or UDP, e.g.-I192.1.1.182 -P2404 -T
     Note: the IEC870-104 protocol default port number is 2404.

3. Make an entry in the CitectSCADA “IO Devices” form for each device as follows:
   - Address = Device address of the connected device, e.g. 1;
   - Protocol = IEC870IP;
   - Port Name = one of the ports defined above.

2.3 Example IEC870IP CitectSCADA Project:

The IEC870IP driver disk also contains a backup of an example project that utilises the driver.

To install this project restore the project as described in the CitectSCADA user documentation. You should refer to the IEC 870 standards for more information about the protocol and its terminology.

3.2 General Parameters : [IEC870IP] Section

The table below outlines the parameters that effect all configured IEC ports.

Name	Default Value	Valid Values	Description
DefaultSetPtType	NORM	NORM SCALE	Selects whether Set Point Outputs should default to “Normalised Value” (“NORM”), or to “Scaled Value” (“SCALE”). Set Points can be overridden from this default individually in the IOA Configuration File

Name	Default Value	Valid Values	Description
Class2PollTimeMs	500	0 – 32767	How often to check for Class 2 data (msec). "0" switches off polling. This parameter is ignored in Balanced mode – no polling.
Class1PollTimeMs	5000	0 – 32767	How often to check for Class 1 data (msec). "0" switches off polling. This parameter is ignored in Balanced mode. If the RTU has class one data, this is shown in the response to the poll for class 2 data, in this case the driver will automatically poll for class 1 data I
GIPollTimeSec	3600	0 – 32767	How often to perform General Interrogations (seconds). "0" switches off polling. A General Interrogation flags the device to return the current state of all points. General Interrogations should not be necessary except after communications breaks, but it's a good idea to perform them occasionally to provide an integrity check of the database.
CounterPollTimeSec	1800	0 – 32767	How often to perform Counter Interrogations (seconds). "0" switches off polling. A Counter Interrogation flags the device to return the current state of all counters in Class 1 data (which normally only contain changes to data). Some devices will return counter changes in Class1 events, in which case frequent Counter Interrogations are not necessary. Other devices return counter data ONLY in response to Counter Interrogations, so the polling rate needs to be set as per the required update rate. Note that this driver currently does not support getting counter values "exactly" on o'clock etc boundaries, such as is needed by some EMS systems etc. A fast polling rate would be needed to approach this (if counter change events are not sent by the device).

Name	Default Value	Valid Values	Description
CounterPollType	1	0 1 2 or 3	Type of Counter Interrogation message to send. This parameter will affect Counter Interrogations done as a regular poll, but not those done on demand through the "FORCECI" output (which are always without freeze and reset). This parameter can be one of the following: 0 - no freeze or reset 1 - freeze without reset 2 - freeze with reset 3 - reset only
TimeSyncPollTimeSec	1800	0 - 32767	How often to set a device's time (seconds). Default is 30 minutes. Setting 0 disables time synchronisation
TimeSyncOnInit	1	0 or 1	Whether to set a device's time as it comes on line (1) or not (0).
TxTestLink	5000	0 – 32767	How often to send Test Link messages (milliseconds). "0" switches off transmitted Test Link messages. Note that this has effect only for Balanced Mode, because Test Link messages are undefined in Unbalanced Mode and will not be sent.
TimeoutCount	3	0 – 32767	Number of timeouts to occur before a device is marked as Offline
RxInactivityTimeout	0	0 – 32767	How long to allow a link to be idle before taking the device offline (seconds). Default is no limit. The limit will only apply if no polling is done on the link – i.e. the driver is in balanced mode and TxTestLink parameter is set to 0. Typically, this parameter would be set to about 3 times the device's Test Link polling period or to 0 (to disable the timeout) if no regular polling occurs from the device.
DefaultSOEMode	0	0 or 1	Each Input IOA can be configured to support SOEs or not in the driver. This sets the default for all IOAs that do not override this. SOEs will default to enabled if this parameter is 1, disabled if 0. If SOEs are to be used, try to ensure that only those IOAs for which CitectSCADA will use SOE data have SOEs enabled.

Name	Default Value	Valid Values	Description
DefaultAnalogSOEMode	0	0 or 1	Each Input Analogue IOA can be configured to support SOEs or not in the driver. This sets the default for all Analogue Input IOAs that do not override this. SOEs will default to enabled if this parameter is 1, disabled if 0. It is recommended that SOE not be used for analogue points
MaxSOEServer	1	1 to 32767	How many servers are defined on the CitectSCADA network that read SOEs? In other words, how many different CitectSCADA processes will read each SOE? See section 3.3
SOEQueue	500	0 – 32767	Size of SOE queue - per device.
SOERmUnusedMs	5000	0 – 32767	SOE Remove Unused “Hold” time to keep unread SOEs on the queue before we delete them (milliseconds). See section: SOE Hold Mechanism
SOEHoldMs	2000	0 - 32767	Time to keep SOE entries in the queue that have been read at least once, but not by the configured number of alarm servers. See section: SOE Hold Mechanism
SOERmCheckMs	1000	0 –32767	SOE Remove Check How often to run through SOE queues checking for timed-out SOEs to be removed (milliseconds). See section: SOE Hold Mechanism
ReceiveQueueSize	250	0 – 32767	Size of the Interrupt data Receive Queue, in number of messages per channel. Allow AT LEAST one entry for all the messages in the largest GI initiation plus 20.
StoreDigitalInQual	0	0 or 1	If set it places the state of the point (either single or double digital points) is stored inside the quality descriptor. This allows a single tag to be used for both value and quality information – thus reducing the total number of tags required
AccumFreeze-Invalidate	1	0 or 1	Whether to mark accumulators affected by a freeze as "not valid" when the freeze is done. This will have the effect of forcing them to be re-read from the device the next time a "read" request for them is received from CitectSCADA. 0 Disables 1 Enables

Name	Default Value	Valid Values	Description
[Kernel] Debugstr	PortName	Valid port name	The port name (from the CitectSCADA "Ports" form) of a serial port is specified here to log debug information about driver activity. The information is logged to the CitectSCADA Kernel and the "\winnt\syslog.dat" file. When the first "IEC870IP" channel is initialised the configuration files are read and processed. At this point kernel debug messages will be generated for any errors in the file. So in order to see these it is advisable to set the debug parameter in the "citect.ini" file to show all debug messages for the first channel - at least on initial runs, until the configuration file has been validated. E.g.: [IEC870IP] debugstr=ChannelName ALL Where: "ChannelName" is the port name (from the CitectSCADA "Ports" form) of the first serial port that will be used with the IEC870IP driver.

3.3 Channel Parameters

The following port specific parameters may be added to the [IEC870IP] section.

Name	Default Value	Valid Values	Description
<PortName>IOAOctets	2	1 2 3	Size of Information Object Addresses, in octets
<PortName>LinkAddrOctets	2	1 2	Size of Link addresses, in octets
<PortName>CommonAddrOctets	2	0 1 2	Size of Common addresses, in octets. 0 is only valid for balanced mode
<PortName>BalancedMode	0	0 1	Selects balanced mode (1) Or unbalanced mode (0).
<PortName>COTLength	1	1 2	Length of the cause of transmission, in octets
<PortName>Station	A	A or B	This applies only in Balanced mode, and allows the CitectSCADA end of a balanced link to be either “Station A” or “Station B”. See the IEC870-5-101 standard for more information
<PortName>InterCharTimeoutMs	50	0 – 32767	Allowable gap between characters in a message from a device. The intercharacter timeout is an extra check to ensure the end of a message is recognised as such, should things get out of step. The ideal theoretical value is dependent on baud rate, as this determines the expected intercharacter gap. However, the timeout typically needs to be a bit larger due to timing inaccuracies, so 50ms or more is recommended. The number of timeouts can be monitored in the driver statistics.
<PortName>AppTimeoutMs	4000		Application-level timeout for the device. The normal “Timeout” parameter applies to link-level responses, but application-level data may take longer due to processing taking place in the device. This is used for example when waiting for a Select Confirmation after a Control Select has been issued, or when waiting for all Interrogation data to arrive.

Note: Many (but not all) IEC 870-5-104 devices will provide an "Activation Termination" message at the end of Interrogation data, so no application timeout is needed. Otherwise, the timeout will allow the driver to know when all Interrogation data has been received, for example GI data when a device is being brought on line.

Note: The lower the timeout value, the more quickly units can be brought back on line.
If the timeout it is set too low then CitectSCADA may begin reading data from the driver before it is all refreshed – and "data not yet valid" errors will occur.

3.4 Alarm Server Configuration Parameters [Alarm]

The following two parameters for the alarm server should be set (see the CitectSCADA documentation for more information)

HresType=7
Twenty-four hour: millisecond timer - stored as decimal, Currently this is the only time stamp format supported by the IEC 870-5-104 driver.

HighResOff=1
Use millisecond accuracy for active and inactive alarm transitions (the default setting is to only time stamp alarm active transitions).

4. Optional Configuration Files

4.1 Unit Configuration File: IecUnit.dbf

The IEC870-5-104 protocol provides three different levels of addressing:

The Link address: This is the physical address of the device that the CitectSCADA master will talk to at the data link level, i.e. that the CitectSCADA master is physically linked to. It roughly corresponds to a CitectSCADA "I/O Device Address".

The Common address. The logical address of the device that the CitectSCADA master will communicate with at the application level. Usually it is the same as the Link address, but if the device physically linked to be connected to other devices, then the Common address distinguishes between them, allowing a particular one to be addressed. It also roughly corresponds to a CitectSCADA "I/O Device Address".

Information Object Address (IOA); This is the object to be read or written to within the device. It corresponds to a CitectSCADA "Variable Tag Address"

Catering for both a link address and a common address is done by:
- By default, the CitectSCADA I/O device address is used as the common address and link address, if they are always the same.
- Where some common and link addresses are different, use an address configuration file for the driver that maps Link addresses to Common addresses. The CitectSCADA I/O Device Address will then always be an index that maps to an entry in this configuration file.

This address configuration file is "IECunits.dbf" by default, and is optional. An example file is provided with the IEC Driver.

A Configuration Parameter defines whether this file is to be used. If in use, ALL units will need to be specified, one per row.

Fields for each row are:
- I/O Device Address: an index number referenced from CitectSCADA's I/O Device Address field;
- Link address;
- Common address.

Column	Description
QUALIF	Control Qualifier **for Output points only, leave blank for other point types** Single or Double Control Output, or Regulating Step Command 0 No additional definition 1 Short Duration Pulse 2 Long Duration Pulse 3 Persistent Output 4..31 Reserved, as per IEC Standard The default mode is specified by the driver configuration parameter, DefaultDOQual Set Point Command 0 Default 1..127 Reserved, as per IEC Standard
SELEX	Select / Execute mode for output points **leave blank for other point types** 0 Direct execute (single stage), 1 Select/Execute (three stage). The default is specified by the driver configuration parameter, DefaultSEMode
SOEENABL	SOE Enable flag **for input points only leave blank for other point types** Enables or Disables SOE reporting for this point 0 SOEs Disabled, 1 SOEs Enabled. The default is specified by the driver configuration parameter, DefaultSOEMode

4.2.1 Configuration of Double Digital Outputs

In many cases, the IOA configuration file won't need to be used.

However all Digital Outputs default to SDO (Single Digital Output), so any DDOs (Double Digital Outputs) or RSOs (Regulating Step Outputs) needed must be configured in the IOA configuration File.

If for example, one Select/Execute Short Pulse DDO with IOA = 1221 is to be configured on an RTU with Common address = 43 on the "Serial1" channel, then put one row in the IOA configuration File as follows:

CHANNEL	COMMADDR	IOA	TYPE	QUALIF	SELEX	SOEENABL
Serial1	43	1221	DDO	1	1	0

Configuring IO Points

4.3 Quality Descriptors

For IEC870-5-104 protocol, the CitectSCADA Quality Descriptor word has bits defined as follows:

Bit Number	Description
0 (least significant)	For Counters: Overflow or counter carry indication For Single Points: if the StoreDigitalInQual parameter is 1 this bit contains the current point value For Double Points: If the StoreDigitalInQual parameter is set this bit contains the lower bit of the point value (DPI)
1	Transient for integer inputs For Double Points (if the StoreDigitalInQual Parameter is 0) Indeterminate / intermediate state for double digital inputs (DDI) For Double Points (if the StoreDigitalInQual Parameter is 1) The upper bit of the point value (DPI)
2	For Double Points (if the StoreDigitalInQual Parameter is 0) Invalid state for double digital inputs (DDI)
3	Blocked
4	Substituted
5	Not topical
6	Invalid
7	Counter adjusted since last reading
8..12	Counter Sequence number (number from 0 to 31)
13	Quality Descriptor out of date
14	Timestamp was provided by PC, not RTU
15 (most significant)	Timestamp from RTU has “invalid” set (this bit is undefined if timestamp from PC, i.e. bit 14 is set)

Note: When the StoreDigitalInQual Parameter is zero

The IEC standards specify that the two bit DPI field is “Indeterminate or Intermediate” when DPI=0, and “Indeterminate” when DPI=3.

To determine the value of the DPI from D and QD

If QD Bit 1 and Bit 2 are not set and D is 0, then DPI = 1
If QD Bit 1 and Bit 2 are not set and D is 1, then DPI=2
If QD Bit 2 is set, then DPI=0
If QD Bit 3 is set then DPI=3

On start up all input IOAs are marked “Not topical” and stay that way until the driver receives data for them, at which time the “Not topical” flags in this data are used. IOAs that don’t exist on a device will remain marked “Not topical” forever.

Incoming protocol messages generally have Quality Descriptors associated with them that include indications of the data being invalid or not topical (out of date). These indications are returned from

the driver to the CitectSCADA kernel in “CitectSCADA Quality Descriptor” pseudo-registers, one per input data point.

Other indications (such as “transient” for integer inputs or “intermediate” for DDIs) do not come from the protocol data packet’s Quality Descriptor, but are returned in the driver’s CitectSCADA Quality descriptor pseudo-registers.
Consequently **all** input data points will have CitectSCADA Quality Descriptor pseudo-registers, even those which do not have a protocol Quality Descriptor implemented.

Note: that an input point could be read from a device as unpacked protocol data packets with Quality Descriptors, or in packed form without. Hence a bit is allocated in the CitectSCADA Quality Descriptor word to flag that the current data is without a protocol quality descriptor, and that some other bits in the CitectSCADA Quality Descriptor are therefore “old”. The “old” bits would be those from the Protocol Quality Descriptor:

- *Invalid*
- *Not Topical*
- *Substituted*
- *Blocked*
- *Overflow*

5. Processing Time Stamped Data

Note: The term Sequence of Events (SOE) processing is used to describe time tagged data events

5.1 Time Stamps

Timestamps in this protocol are received from devices as either:
- Three byte integers containing the number of milliseconds since the last “o’clock”, i.e. the minutes, seconds and milliseconds components of the time of day; or
- Seven byte integers containing the above plus information up to and including years (a recent extension to the standard), as in the protocol’s Clock Synchronisation command.

Currently the driver converts these into 32-bit integers containing milliseconds since midnight, as supported by CitectSCADA.(HresType=7)

CitectSCADA has built-in support for time stamped alarm transitions, but not for other time stamped data. However, IEC870IP extends time stamping to potentially any incoming data.

CitectSCADA’s usual method for obtaining timestamps is as follows:
- The value of the datum is repeatedly read in the alarm scan
- When it changes, CitectSCADA then reads its timestamp
- The driver takes this as an indication that this state change has been processed and can be popped from its queue.

The above is currently implemented by CitectSCADA as part of its alarm handling. However for non-alarm points it should be implemented by application CiCode.

5.2 Time Stamped Data Queues

In some applications, there may be a need to receive and store regularly polled data even when its value doesn’t change, for example hourly meter readings. The hourly readings may need to be recorded in a database as discrete readings even if sequential readings for a point have identical values. To allow for this, an optional timestamp method has been allowed for as follows:

- The timestamp of the datum is repeatedly read
- When it changes, CitectSCADA then reads its value
- The driver takes this as an indication that this state change has been processed and can be popped from its queue.

*Note: that for **either** method, Quality Descriptor words associated with time stamped data should be read **before** the driver is signalled to pop the data from the queue.*

Cicode Note: This can be achieved by putting a function into the timestamp field of the time stamped alarm. The arguments passed into the function are (in order), Fn(QDS, ST) then the function simple returns the ST but has captured the QDS for later processing

An additional consideration with the protocol’s extended use of timestamps is in queuing of changes.

Normally in an SOE-enabled driver, all time stamped changes are put into the SOE queue. Queue entries not retrieved by CitectSCADA remain until they time out, which isn’t usually a problem because time stamped changes will be detected by CitectSCADA’s alarms scanning.

With the extended use of timestamps in this protocol, the SOE queues may be overloaded with unused entries. To prevent this points, are individually configurable as to whether the driver keeps SOE data for them or not.

5.3 SOE Hold Mechanism

The driver implements a SOE hold mechanism to provide:

- efficient memory usage
- ensure SOE data is not lost
- Support for multiple SOE server tasks.

The driver will keep the SOE event in its buffer until one of the following conditions is met

A) All SOE server tasks have read the event (i.e. the number of SOE servers as configured by the MaxSOEServer Parameter)
B) At least ONE SOE server task has read the event and hold time has elapsed (i.e. the SOEHoldMs time has elapsed)
C) The SOERmUnusedMs time has elapsed

The processing that clears removes the data from the SOE queue if performed periodically, the period is determined by the SOERmCheckMs parameter.

6. Accessing Device Data

Once devices have been configured into CitectSCADA, various register addresses are available for accessing that data.

6.1 Register Addresses

CitectSCADA Variable Tags are mapped to device I/O points through register addresses, that are be configured with the following syntax.

Note that two forms of IOA are provided –
Unstructured: <ioa> represents a decimal number with a range dependent on the number of configured IOA octets, i.e. it could be 0..255, 0..65535 or 0..16777215.
Structured: <ioa1>,<ioa2> and <ioa3> represent decimal numbers in the range 0..255.

Note: Output points cannot be read from the Driver by CitectSCADA – this reflects the IEC standard, as the protocol does not cater for reading of output points.

6.1.1 Digital Inputs – used for Single-Point Information and Double-Point Information.

Description	Unstructured Address	Structured Address
Current State	D<ioa>	D<ioa1>.<ioa2>.<ioa3>
Quality Descriptor, Current State:	QD<ioa>	QD<ioa1>.<ioa2>.<ioa3>
Value-triggered SOE Value	SD<ioa>	SD<ioa1>.<ioa2>.<ioa3>
Value-triggered SOE Timestamp	ST<ioa>	ST<ioa1>.<ioa2>.<ioa3>
Time-triggered SOE Value	TTSD<ioa>	TTSD<ioa1>.<ioa2>.<ioa3>
Time-triggered SOE Timestamp	TTST<ioa>	TTST<ioa1>.<ioa2>.<ioa3>
Quality Descriptor, SOE Value	SQD<ioa>	SQD<ioa1>.<ioa2>.<ioa3>

Double-Point Intermediate and Indeterminate states will be flagged in the Quality Descriptor.

6.1.2 Integer Inputs – Step position information, and Measured Values

Note: that both forms of Measured Value, scaled and normalised are returned as signed two's complement integers.

Normalised values, according to the IEC standards, range over raw values from –32768 to +32767.

Description	Unstructured Address	Structured Address
Current Value	I<ioa>	I<ioa1>.<ioa2>.<ioa3>
Quality Descriptor, Current State:	QD<ioa>	QD<ioa1>.<ioa2>.<ioa3>
Value-triggered SOE Value	SI<ioa>	SI<ioa1>.<ioa2>.<ioa3>
Value-triggered SOE Timestamp	ST<ioa>	ST<ioa1>.<ioa2>.<ioa3>
Time-triggered SOE Value	TTSI<ioa>	TTSI<ioa1>.<ioa2>.<ioa3>
Time-triggered SOE Timestamp	TTST<ioa>	TTST<ioa1>.<ioa2>.<ioa3>
Quality Descriptor, SOE Value	QDS<ioa>	QDS<ioa1>.<ioa2>.<ioa3>

In the protocol Step position information is a single byte with the MSB a “transient” indicator (set if equipment is in transient state), and the other 7 bits holding a value from –64 to +63. The driver will convert it to a 16-bit signed integer and separate the transient indicator out into the Quality Descriptor.

6.1.3 Long Integer Inputs – used for Integrated total (Counters)

Description	Unstructured Address	Structured Address
Current Value	L<ioa>	L<ioa1>.<ioa2>.<ioa3>
Quality Descriptor, Current State:	QD<ioa>	QD<ioa1>.<ioa2>.<ioa3>
Value-triggered SOE Value	SL<ioa>	SL<ioa1>.<ioa2>.<ioa3>
Value-triggered SOE Timestamp	ST<ioa>	ST<ioa1>.<ioa2>.<ioa3>
Time-triggered SOE Value	TTSL<ioa>	TTSL<ioa1>.<ioa2>.<ioa3>
Time-triggered SOE Timestamp	TTST<ioa>	TTST<ioa1>.<ioa2>.<ioa3>
Quality Descriptor, SOE Value	QDS<ioa>	QDS<ioa1>.<ioa2>.<ioa3>

6.1.4 Float Inputs – used for Short Floating Measurand Values

Description	Unstructured Address	Structured Address
Current Value	F<ioa>	F<ioa1>.<ioa2>.<ioa3>
Quality Descriptor, Current State:	QD<ioa>	QD<ioa1>.<ioa2>.<ioa3>

6.1.5 Digital Outputs – used for Single Command (Single Digital Output), Double Command (Double Digital Output) and Regulating Step Command

Description	Unstructured Address	Structured Address
New Value	C<ioa>	C<ioa1>.<ioa2>.<ioa3>
New Value	C<ioa>.Sel	C<ioa1>.<ioa2>.<ioa3>.Sel
New Value	C<ioa>.Desel	C<ioa1>.<ioa2>.<ioa3>.Desel
New Value	C<ioa>.Ex	C<ioa1>.<ioa2>.<ioa3>.Ex
Return Confirm Code (Integer)	C<ioa>.Conf	C<ioa1>.<ioa2>.<ioa3>.Conf

Double Commands are only ever used for trip/close operations, so only 2 states are valid. Hence they will be treated as Single Digital Output points by CitectSCADA, and translated in the driver. 0 becomes (0,1) (trip), and 1 becomes (1,0) (close).

Regulating Step Commands are similar to Single Commands, except that writing a 0 means “next step lower”, and 1 means “next step higher”.

The control outputs in the IEC870-5-104 protocol are pure outputs, therefore there is no associated quality information.

C<ioa>.Sel is to transmit select request.
C<ioa>. Desel is to transmit break off command (deselect request).

<0x 45> := unknown cause of transmission
<0x 46> := unknown common address of ASDU
<0x 47> := unknown information object address

note: to reset C<ioa>. Conf variable to zero by WRITE operation.

C<ioa> is to transmit direct execute command or transmit select request and execute request based on the point type (direct or S/E).

6.1.6 Integer Outputs – used for Set Point Command:

Description	Unstructured Address	Structured Address
New Value	CI<ioa>	CI<ioa1>.<ioa2>.<ioa3>

The control outputs in the IEC870-5-104 protocol are pure outputs, therefore there is no associated quality information.

6.1.7 Force General Interrogation

Writing “1” to this digital register will force a General Interrogation of the device.

FORCEGI

6.1.8 Force Counter Interrogation

Writing “1” to this digital register will force a Counter Interrogation of the device (without freeze or reset of the counters).

FORCECI

6.1.9 Force Clock Synchronisation

Writing “1” to this digital register will force a Clock Synchronisation of the device.

FORCECS

6.1.10 Counter Freeze

Writing “1” to this digital register will perform a counter freeze on the device.

COUNTFZ

6.1.11 Counter Reset

Writing “1” to this digital register will perform a counter reset on the device.

COUNTRS

6.1.12 Counter Freeze and Reset

Writing “1” to this digital register will perform a counter freeze and reset on the device.

COUNTFZRS

7. Configuring the CitectSCADA Alarm Server

7.1.1 Parameters

See section 3.4: Alarm Server Configuration Parameters [Alarm].

7.1.2 Alarm Points

To configure an alarm point first you need to configure the following

- Create a variable tag for the value of the SOE, i.e. the value triggered SOE value
- Create variable tag for the timestamp, value-triggered SOE Timestamp.
- Create a time stamped alarm, set the variable to first tag, i.e. the value of the SOE and the time to the second tag, i.e. the time stamp.

For example to configure an alarm for a single digital input with Information Object Address (IOA) equal to 400.

Create variable tag, VALUE with address SD400,
Create variable tag, TIMESTAMP with address TTDT400
Create a time stamped alarm, ALARM with variable tag VALUE and time TIMESTAMP.

(Alarm pages and Alarm summary pages can be created from the standard CitectSCADA templates).

7.2 Driver Statistics

The following special counters are accumulated by the driver. They can be viewed at run-time by typing “page Driver” in the Kernel window to display the Driver window, and pressing the down-arrow or “v” key in that window to activate Verbose mode.

Number	Label	Purpose/Meaning of this counter
1	Data CRC Error	The number of messages received that had a bad checksum.
2	Unit Offline	The number of times any device has been made off-line.
3	Negative Reply Rx	The number of times a device has indicated in a Command Response that there was something wrong with the command sent to it.
4	NOT IMPL Replies	The number of times the driver has received a NOT IMPLEMENTED message
5	NOT AVAIL Replies	The number of time the driver has received a NOT AVAILABLE message
6	Busy Replies	The number of times the driver has received a BUSY message
7	FCB Errors	The number of times the driver has received a message with an incorrect Frame Control Bit (FCB), this indicates that not all messages are being received correctly
8	Unknown Addr Rx	The number of times the driver has received a message containing invalid address information, indicates that the slave device may be incorrectly configured
9	SOE Q Overflows	The number of times that the driver has lost data because its internal buffers have become full
10	Intercharacter timeouts	The number of times message have been rejected because the time between bytes in a message was too large. This may indicate that communication parameters need to be adjusted
11	SOE Q Length	The number of SOE buffers currently used. When no changes are occurring this value will be equal to the number of SOE tags in the system. If this number exceeds the configured number of SOE

Number	Label	Purpose/Meaning of this counter
		buffers data will be lost
12	Poll Count	In unbalanced mode: The number of Class 1 or Class 2 poll messages that have been transmitted since CitectSCADA started
13	Request Link Count	The number of request link status message that have been transmitted since CitectSCADA started
14	Test Link Count	In balanced mode: The number of Test Link Messages that have been transmitted since CitectSCADA started
15	General Int Count	The number of general interrogation messages that have been transmitted since CitectSCADA started
16	Time Sync Count	The number of time synchronisation messages that have been transmitted since CitectSCADA started

8. Protocol Issues

There are some ambiguities in the IEC standards that leave room for minor interoperability problems. This section documents the way that some of these ambiguities have been interpreted and implemented for IEC870IP.

8.1 Handling of Non-Timestamped Input Data

A given IOA can be send by an RTU in several different message types, some containing timestamps and some not. Any timestamped data will be processed by the driver as a new event and added to the SOE queues. Non-timestamped data is processed by the driver for each IOA as follows:

- If no previous data for this IOA has been received then the new data is stored;
- if the new value/state or quality descriptor is different than the previous stored value then the new data is stored;
- If the value/state and quality descriptor are the same as the previously stored value new data is discarded.

If the new data is stored then the timestamp is set to the time on the PC when the message arrived, and the “timestamp set on PC” flag in the quality descriptor is set.

8.2 Events During General Interrogations

The can be an issue if while retrieving data from the RTU for a General Interrogation (GI), a state change / spontaneous event occurs. Depending on the RTU’s implementation this new state change event may remain queued on the RTU until the GI is complete or may be sent immediately, in the middle of the GI.

The driver assumes that the RTU will correctly use Cause of Transmission fields to identify data being sent in response to a GI as opposed to spontaneous transmission. If the driver receives non-GI data for an IOA while a GI is underway then this data will be stored and the IOA will be marked to prevent any GI data overwriting it for the duration of the current GI.

9. Testing The Driver

Once the driver has been configured correctly, connect the IEC RTU to the configured communication port on CitectSCADA PC.

Check that the following protocol parameters match between the device and the CitectSCADA driver. See section **Error! Reference source not found.** for details on these parameters and their default values. If their default values don't suit then set them correctly:

[IEC870IP]

- <PortName>IOAOctets=
- <PortName>LinkAddrOctets=
- <PortName>CommonAddrOctets=
- <PortName>BalancedMode=
- <PortName>COTLength=
- <PortName>Station= (Note: only needed for Balanced Mode)

Enable basic debug output in the CitectSCADA Kernel window with entries in the "\winnt\citect.ini" file as follows:

- [DEBUG]
  Kernel=1
- [IEC870IP]
  debugstr=<PortName>

Start the CitectSCADA Runtime system (by clicking on the RUN icon on the CitectSCADA Explorer speed bar). If everything in configured correctly and the cabling is correct, the device should now come online.

If the system is still not communicating, confirm all the configuration parameter are correct and both the CitectSCADA values and the RTU value match.

10. Troubleshooting

IEC60870-5-104 is a complex protocol with many interoperability issues. Due to the number of message types defined in the protocol, only those that are likely to be used in practice have been implemented (see the Interoperability List for details). If you find that a message type you need is not yet supported in the driver please contact your supplier, and it will be added as soon as possible.

Interoperability problems (like the above) or configuration errors may cause communication errors to appear. The best way to diagnose these is to look at the low-level data packets passing back and forth between devices and the CitectSCADA I/O Server PC. A data analyser is ideal for this, but CitectSCADA's low-level "COMx" logging can also be used.

10.1 "COMx" Logging

The CitectSCADA COMx serial driver is used by IEC870IP driver to communication via the serial ports of the PC. This driver includes some basic debug facilities that enable the user to see the actual data transmitted and received from the communication port. See CitectSCADA knowledge base article Q2404 for more information.

Example configuration: the following entries in the "\winnt\citect.ini" file, in the [COMx] section, will enable "COMx" logging:

```
[COMx]
WritePortName=PORT_1
WriteDebugLevel=1
ReadPortName=PORT_1
ReadDebugLevel=1
```

Where PORT_1 is port name as it appears in the CitectSCADA->Communications->Ports form.

The logs will be created in data files named after the ports, in the windows directory. For the example port names above, the files created would be:

```
WPORT_1.dat
RPORT_1.dat
WSerial1.dat
RSerial1.dat
```

10.2 Gathering Information for a Support Request

Should a communications problem arise requiring support from CiT, please do as much as possible of the following to help diagnosis:

- Provide a description of the problem, and the circumstances that cause it;
- Enable COMx logging as described in section 10.1 above, and enable driver debug "debugstr=" logging as described in section **Error! Reference source not found.**, then reproduce the problem;
- Provide the log files produced above;
- Provide your "citect.ini".
- Provide information on configuration of the device, including communications parameters (e.g. baud rate) and IEC parameters (e.g. IOA length etc);
- Try to reproduce the problem in a simple CitectSCADA project such as the test project provided with this driver, and provide a backup of this project;

Appendix A. IEC 870 - 5 - 101 Interoperability List

The interoperability list refers to section 8 of IEC 60870-5-101.

8.1 *Network Configuration* (network specific parameter)

- ■ Point-to-point
- ■ Multiple point-to-point
- ■ Multipoint-party line
- ☐ Multipoint-star

8.2 *Physical layer* (network-specific parameter)

Transmission Speed (Control Direction)

Unbalanced interchange circuit V.24/V.28 Standard	Unbalanced interchange circuit V.24/V.28 Recomm. if>1200bit/s	Balanced interchange circuit X.24/X.27
■ 100 bit/s	■ 2400 bit/s	☐ 2400 bit/s ☐ 56000 bit/s
■ 200 bit/s	■ 4800 bit/s	☐ 4800 bit/s ☐ 64000 bit/s
■ 300 bit/s	■ 9600 bit/s	☐ 9600 bit/s
■ 600 bit/s		☐ 19200 bit/s
■ 1200 bit/s		☐ 38400 bit/s

Transmission Speed (Monitor Direction)

Unbalanced interchange circuit V.24/V.28 Standard	Unbalanced interchange circuit V.24/V.28 Recomm. if>1200bit/s	Balanced interchange circuit X.24/X.27
■ 100 bit/s	■ 2400 bit/s	☐ 2400 bit/s ☐ 56000 bit/s
■ 200 bit/s	■ 4800 bit/s	☐ 4800 bit/s ☐ 64000 bit/s
■ 300 bit/s	■ 9600 bit/s	☐ 9600 bit/s
■ 600 bit/s		☐ 19200 bit/s
■ 1200 bit/s		☐ 38400 bit/s

8.3 *Link Layer* (Network-specific parameter)

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

Link Transmission Procedure

- ■ Balanced transmission
- ■ Unbalanced transmission

Address Field of the Link

- ■ Not present (balanced transm. only)
- ■ One octet
- ■ Two octets
- □ Structured
- ■ Unstructured

Frame Length

262 Max. length L (number of octets)

8.4 *Application Layer*

Transmission Mode for Application Data

Mode 1 (Least significant octet first), as defined in clause 4.10 of IEC 870-5-4, is used exclusively in this companion standard.

Common address of ASDU (System-specific parameter)

- ■ One octet
- ■ Two octets

Information Object Address (System-specific parameter)

- ■ One octet
- ■ Two octets
- ■ Three octets
- ■ Structured
- ■ Unstructured

Cause of Transmission (System-specific parameter)

- ■ One octet
- ■ Two octets (with originator address)

Selection of Standard ASDUs

Process information in monitor direction (station-specific parameter)

■	<1> := Single-point information	M_SP_NA_1
■	<2> := Single-point information with time tag	M_SP_TA_1
■	<3> := Double-point information	M_DP_NA_1
	<4> := Double-point information with time tag	M_DP_TA_1
	<5> := Step position information	M_ST_NA_1
	<6> := Step position information with time tag	M_ST_TA_1
	<7> := Bitstring of 32 bit	M_BO_NA_1
	<8> := Bitstring of 32 bit with time tag	M_BO_TA_1
	<9> := Measured value, normalized value	

Process information in control direction (station-specific parameter)

■	<45>:= Single command	C_SC_NA_1
■	<46>:= Double command	C_DC_NA_1
■	<47>:= Regulating step command	C_RC_NA_1
■	<48>:= Set point command, normalized value	C_SE_NA_1
■	<49>:= Set point command, scaled value	C_SE_NB_1
□	<50>:= Set point command, short floating point value	C_SE_NC_1
□	<51>:= Bitstring of 32 bit	C_BO_NA_1

System information in monitor direction (station-specific parameter)

■	<70>:= End of initialization	M_EI_NA_1

System information in control direction(station-specific parameter)

■	<100> := Interrogation command	C_IC_NA_1
■	<101> : = Counter interrogation command	C_CI_NA_1
☐	<102> : = Read command	C_RD_NA_1
■	<103> : = Clock synchronisation command	C_CS_NA_1
☐	<104> : = Test command	C_TS_NB_1
☐	<105> : = Reset process command	C_RP_NC_1
☐	<106> : = Delay acquisition command	C_CD_ NA_1

Parameter in control direction (station-specific parameter)

☐	<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)

☐	<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

8.5 *Basic Application Functions*

Station Initialisation (station-specific parameter)

- [x] Remote initialisation

General Interrogation (system- or station-specific parameter)

- [x] global
- [ ] group 1
- [ ] group 2
- [ ] group 3
- [ ] group 4
- [ ] group 5
- [ ] group 6
- [ ] group 7
- [ ] group 8
- [ ] group 9
- [ ] group 10
- [ ] group 11
- [ ] group 12
- [ ] group 13
- [ ] group 14
- [ ] group 15
- [ ] group 16

addresses per group have to be defined

Clock Synchronisation (station-specific parameter)

- [x] Clock synchronisation

Command Transmission (object-specific parameter)

- [x] Direct command transmission
- [x] Direct set point command transmission
- [x] Select + execute command
- [x] Select + execute set point command
- [ ] C_SE ACTTERM used

- [x] No additional definition
- [x] Short pulse duration (duration determined by a system parameter in the outstation)
- [x] Long pulse duration (duration determined by a system parameter in the outstation)
- [x] Persistent output

Transmission of integrated totals (station- or object-specific parameter)

■	Counter request	■	General request counter
■	Counter freeze without reset	☐	Request counter group 1
■	Counter freeze with reset	☐	Request counter group 2
■	Counter reset	☐	Request counter group 3
	Addresses per group have to be defined	☐	Request counter group 4

Parameter Loading (object-specific parameter)

☐	Threshold value
☐	Smoothing factor
☐	Low limit for transmission of measured value
☐	High limit for transmission of measured value

Parameter Activation (object-specific parameter)

☐	Act/deact of persistent cyclic or periodic transmission of the addressed object

File Transfer (station-specific parameter)

☐	File transfer in monitor direction
☐	File transfer in control direction

Appendix B.IEC 870 - 5 - 104 Interoperability List

The interoperability list refers to section 9 of IEC 60870-5-104.

9.1 *System or device*

- ☐ System definition
- ☐ Control station definition(master)
- ■ Control station definition(slave)

9.2 *Network Configuration* (network specific parameter)

- ■ ~~Point-to-point~~
- ■ ~~Multiple-point-to-point~~
- ■ ~~Multipoint-party-line~~
- ■ ~~Multipoint-star~~

9.3 *Physical layer* (network-specific parameter)

Transmission Speed (Control Direction)

Unbalanced interchange circuit V.24/V.28 Standard

- ■ ~~100 bit/s~~
- ■ ~~200 bit/s~~
- ■ ~~300 bit/s~~
- ■ ~~600 bit/s~~
- ■ ~~1200 bit/s~~

Unbalanced interchange circuit V.24/V.28 Recomm. if>1200bit/s

- ■ ~~2400 bit/s~~
- ■ ~~4800 bit/s~~
- ■ ~~9600 bit/s~~

Balanced interchange circuit X.24/X.27

- ■ ~~2400 bit/s~~
- ■ ~~4800 bit/s~~
- ■ ~~9600 bit/s~~
- ■ ~~19200 bit/s~~
- ■ ~~38400 bit/s~~
- ■ ~~56000 bit/s~~
- ■ ~~64000 bit/s~~

Transmission Speed (Monitor Direction)

Unbalanced interchange circuit V.24/V.28 Standard

- ■ ~~100 bit/s~~
- ■ ~~200 bit/s~~
- ■ ~~300 bit/s~~
- ■ ~~600 bit/s~~
- ■ ~~1200 bit/s~~

Unbalanced interchange circuit V.24/V.28 Recomm. if>1200bit/s

- ■ ~~2400 bit/s~~
- ■ ~~4800 bit/s~~
- ■ ~~9600 bit/s~~

Balanced interchange circuit X.24/X.27

- ■ ~~2400 bit/s~~
- ■ ~~4800 bit/s~~
- ■ ~~9600 bit/s~~
- ■ ~~19200 bit/s~~
- ■ ~~38400 bit/s~~
- ■ ~~56000 bit/s~~
- ■ ~~64000 bit/s~~

9.4 *Link Layer* (Network-specific parameter)

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

Link Transmission Procedure

- ■ ~~Balanced transmission~~
- ■ ~~Unbalanced transmission~~

Address Field of the Link

- ■ ~~Not present (balanced transm. only)~~
- ■ ~~One octet~~
- ■ ~~Two octets~~
- ■ ~~Structured~~
- ■ ~~Unstructured~~

Frame Length

262 Max. length L (number of octets)

9.5 *Application Layer*

Transmission Mode for Application Data

Mode 1 (Least significant octet first), as defined in clause 4.10 of IEC 870-5-4, is used exclusively in this companion standard.

Common address of ASDU (System-specific parameter)

- ■ ~~One octet~~
- ■ Two octets

Information Object Address (System-specific parameter)

- ■ ~~One octet~~
- ■ ~~Two octets~~
- ■ Three octets
- ■ Structured
- ■ Unstructured

Cause of Transmission (System-specific parameter)

- ■ ~~One octet~~
- ■ Two octets (with originator address)

Length of APDU

253 Max. length L (number of octets)

Selection of Standard ASDUs

Process information in monitor direction (station-specific parameter)

■	<1> := Single-point information	M_SP_NA_1
■	<2> := Single-point information with time tag	M_SP_TA_1
■	<3> := Double-point information	M_DP_NA_1
■	<4> := Double-point information with time tag	M_DP_TA_1
■	<5> := Step position information	M_ST_NA_1
□	<6> := Step position information with time tag	M_ST_TA_1
□	<7> := Bitstring of 32 bit	M_BO_NA_1
■	<8> := Bitstring of 32 bit with time tag	M_BO_TA_1
■	<9> := Measured value, normalized value	M_ME_NA_1
■	<10>:= Measured value, normalized value with time tag	M_ME_TA_1
	<11>:= Measured value, scaled value	M_ME_NB_1
	<12>:= Measured value, scaled value with time tag	M_ME_TB_1
□	<13>:= Measured value, short floating point value	M_ME_NC_1
■	<14>:= Measured value, short floating point value with time tag	M_ME_TC_1
	<15>:= Integrated totals	M_IT_NA_1
	<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 event 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	

☐ <33>:= Bitstring of 32 bit with time tag CP56Time2a M_BO_TB_1

■ <34>:= Measured value, normalised value with CP56Time2a M_ME_TD_1

■ <35>:= Measured value, scaled value with CP56Time2a M_ME_TE_1

☐ <36>:= Measured value, short floating point value with CP56Time2a M_ME_TF_1

■ <37>:= Integrated totals with CP56Time2a M_IT_TB_1

☐ <38>:= Event of protection equipment with CP56Time2a M_EP_TD_1

☐ <39>:= Packed start events of protection equipment with CP56Time2a M_EP_TE_1

☐ <40>:= Packed output circuit information of protection equipment with CP56Time2a M_EP_TF_1

Process information in control direction (station-specific parameter)

■ <45>:= Single command C_SC_NA_1

■ <46>:= Double command C_DC_NA_1

■ <47>:= Regulating step command C_RC_NA_1

■ <48>:= Set point command, normalized value C_SE_NA_1

■ <49>:= Set point command, scaled value C_SE_NB_1

☐ <50>:= Set point command, short floating point value C_SE_NC_1

☐ <51>:= Bitstring of 32 bit C_BO_NA_1

System information in monitor direction (station-specific parameter)

■ <70>:= End of initialization M_EI_NA_1

System information in control direction(station-specific parameter)

■	<100> := Interrogation command	C_IC_NA_1
■	<101> : = Counter interrogation command	C_CI_NA_1
☐	<102> : = Read command	C_RD_NA_1
■	<103> : = Clock synchronisation command	C_CS_NA_1
☐	~~<104> : = Test command~~	~~C_TS_NB_1~~
☐	<105> : = Reset process command	C_RP_NC_1
☐	~~<106> : = Delay acquisition command~~	~~C_CD_ NA_1~~

Parameter in control direction (station-specific parameter)

☐	<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)

☐	<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

9.6 *Basic Application Functions*

Station Initialisation (station-specific parameter)

■ Remote initialisation

General Interrogation (system- or station-specific parameter)

■ global

☐ group 1	☐ group 7	☐ group 13
☐ group 2	☐ group 8	☐ group 14
☐ group 3	☐ group 9	☐ group 15
☐ group 4	☐ group 10	☐ group 16
☐ group 5	☐ group 11	
☐ group 6	☐ group 12	addresses per group have to be defined

Clock Synchronisation (station-specific parameter)

■ Clock synchronisation

Command Transmission (object-specific parameter)

■ Direct command transmission

■ Direct set point command transmission

■ Select + execute command

■ Select + execute set point command

☐ C_SE ACTTERM used

■ No additional definition

■ Short pulse duration (duration determined by a system parameter in the outstation)

■ Long pulse duration (duration determined by a system parameter in the outstation)

■ Persistent output

Transmission of integrated totals (station- or object-specific parameter)

■	Counter request	■	General request counter
■	Counter freeze without reset	☐	Request counter group 1
■	Counter freeze with reset	☐	Request counter group 2
■	Counter reset	☐	Request counter group 3
	Addresses per group have to be defined	☐	Request counter group 4

Parameter Loading (object-specific parameter)

☐	Threshold value
☐	Smoothing factor
☐	Low limit for transmission of measured value
☐	High limit for transmission of measured value

Parameter Activation (object-specific parameter)

☐	Act/deact of persistent cyclic or periodic transmission of the addressed object

File Transfer (station-specific parameter)

☐	File transfer in monitor direction
☐	File transfer in control direction