IEC60870 Slave and Server

straton user guide – Rev.10

sales@straton-plc.com

CP_R31_006_H

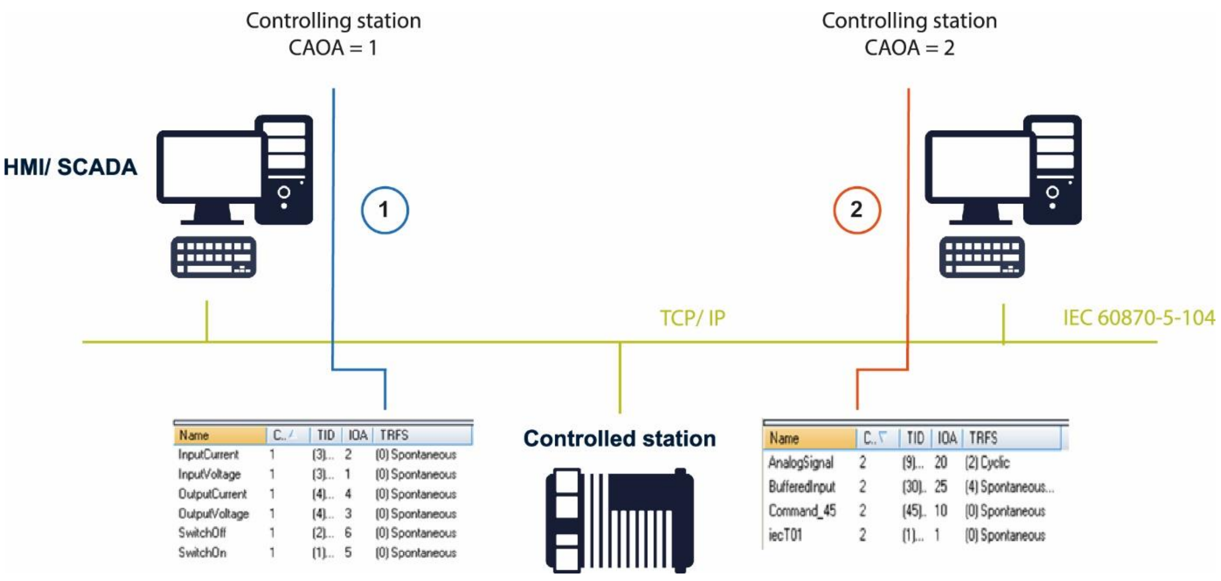
straton

Content

1. Install Editor and Runtime

Download and install from https://straton-plc.com/en/downloads/

2. Example of architecture



Name	C..	TID	IOA	TRFS
InputCurrent	1	(3)...	2	(0) Spontaneous
InputVoltage	1	(3)...	1	(0) Spontaneous
OutputCurrent	1	(4)...	4	(0) Spontaneous
OutputVoltage	1	(4)...	3	(0) Spontaneous
SwitchOff	1	(2)...	6	(0) Spontaneous
SwitchOn	1	(1)...	5	(0) Spontaneous

Name	C..	TID	IOA	TRFS
AnalogSignal	2	(9)...	20	(2) Cyclic
BufferedInput	2	(30)..	25	(4) Spontaneous...
Command_45	2	(45)..	10	(0) Spontaneous
iecT01	2	(1)...	1	(0) Spontaneous

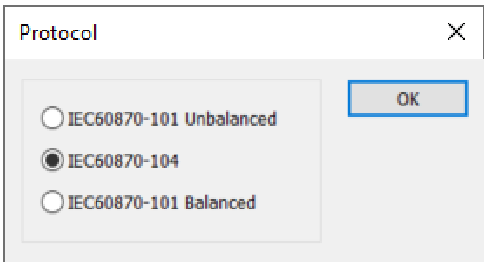
3. Configuration

3.1. Create the configuration

Open the IO Drivers window using the tool bar ( ) or right click on the project > Insert Shortcut > Fieldbus Configurations

Insert a new Fieldbus using tool bar ( ) or menu Insert > Insert configuration and select the "IEC60870 Slave" driver.

Insert a first level using tool bar ( ) or menu Insert > Insert Master/Port. A port describes the physical layer of IEC60870. Select 104 for Ethernet or 101 for serial link. Note that since straton 9.3, the Balanced mode is also supported for serial communications:



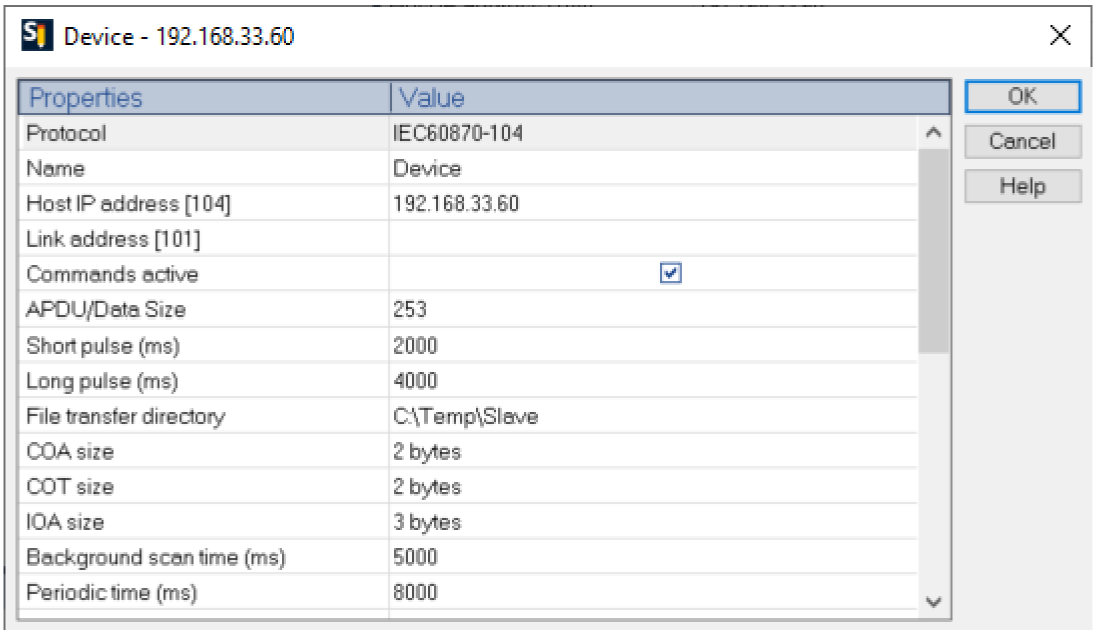
○ IEC60870-101 Unbalanced
[✓ ● IEC60870-104]
○ IEC60870-101 Balanced

Insert a device level using tool bar ( ) or menu Insert > Insert Slave/Data Block. A device describes a virtual channel for communication with its specific properties.

If several Clients must connect to this Slave, then add several "Devices" levels.

For the moment this is **not** possible to use IEC101 and IEC104 at the same time on the same Slave.

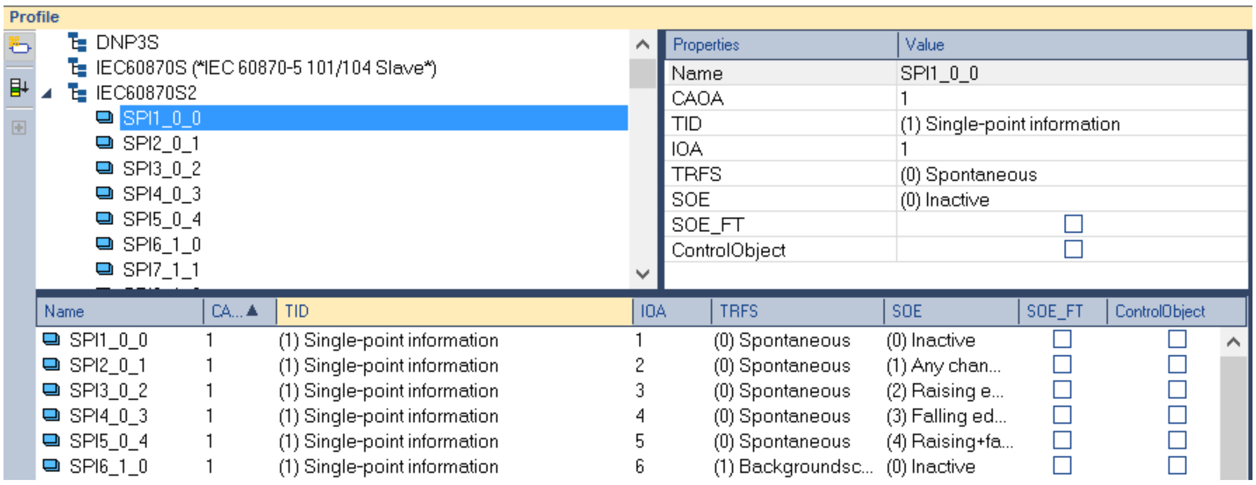
If you use IEC60870-5-104 then just enter the Host IP address



Properties	Value
Protocol	IEC60870-104
Name	Device
Host IP address [104]	192.168.33.60
Link address [101]	
Commands active	✓
APDU/Data Size	253
Short pulse (ms)	2000
Long pulse (ms)	4000
File transfer directory	C:\Temp\Slave
COA size	2 bytes
COT size	2 bytes
IOA size	3 bytes
Background scan time (ms)	5000
Periodic time (ms)	8000

Open the profile editor using the tool bar ( ) or right click on the project > Insert Shortcut > Profiles

Create the IEC60870S2 mapping according to the different variables and data types.



Properties	Value
Name	SPI1_0_0
CAOA	1
TID	(1) Single-point information
IOA	1
TRFS	(0) Spontaneous
SOE	(0) Inactive
SOE_FT	☐
ControlObject	☐

Name	CA...	TID	IOA	TRFS	SOE	SOE_FT	ControlObject
SPI1_0_0	1	(1) Single-point information	1	(0) Spontaneous	(0) Inactive	☐	☐
SPI2_0_1	1	(1) Single-point information	2	(0) Spontaneous	(1) Any chan...	☐	☐
SPI3_0_2	1	(1) Single-point information	3	(0) Spontaneous	(2) Raising e...	☐	☐
SPI4_0_3	1	(1) Single-point information	4	(0) Spontaneous	(3) Falling ed...	☐	☐
SPI5_0_4	1	(1) Single-point information	5	(0) Spontaneous	(4) Raising+fa...	☐	☐
SPI6_1_0	1	(1) Single-point information	6	(1) Backgroundsc...	(0) Inactive	☐	☐

3.2. Communication parameters

For each variable the type of transfer (TRFS) can be set to the following values:

0: SPONTANEOUS

Values are transferred in spontaneous mode each time a value changes.

1: BACKGROUND SCAN

Values are transferred cyclically in the time grid set.

Values are only transferred cyclically if no spontaneous value is transferred within the time set.

This is activated if a time is specified in the device parameters (Background Scan Time).

Supported IDs:

M_SP_NA_1 (Single-point information)

M_DP_NA_1 (Double-point information)

M_ST_NA_1 (Step position information)

M_BO_NA_1 (Bitstring of 32 bit)

M_ME_NA_1 (Measured value, normalized value)

M_ME_NB_1 (Measured value, scaled value)

M_ME_NC_1 (Measured value, short floating point value)

2: CYCLIC

Values are transferred cyclically and not spontaneously.

This is activated if a time is specified in the device parameters (Periodic Time).

Supported IDs:

M_ME_NA_1 (Measured value, normalized value)

M_ME_NB_1 (Measured value, scaled value)

M_ME_NC_1 (Measured value, short floating point value)

3: OFF

Values are not transferred spontaneously.

4: SPONTANEOUS – BUFFERED

Spontaneous values will be queued when the Master is offline. Values are buffered until these are read by the Master.

An independent queue/buffer is created by "Device" level. The size is limited to 2048 events.

Supported IDs:

M_SP_TB_1 (Single-point information with time tag CP56Time2a)

M_DP_TB_1 (Double-point information with time tag CP56Time2a)

M_ST_TB_1 (Step position information with time tag CP56Time2a)

M_BO_TB_1 (Bitstring of 32 bit with time tag CP56Time2a)

M_ME_TD_1 (Measured value, normalized value with time tag CP56Time2a)

M_ME_TE_1 (Measured value, scaled value with time tag CP56Time2a)

M_ME_TF_1 (Measured value, short floating point value with time tag CP56Time2a)

M_IT_TB_1 (Integrated totals with time tag CP56Time2a)

3.3. Quality bits

The IEC 60870-5-101 and IEC 60870-5-104 manage some additional information to the values like the time stamping and the quality of the data.

Note: Variable status and time stamp are stored in the database if the "Allocate status bits for variables with embedded properties" option is activated and if the variable has a defined profile. Please refer to the "Variable status bits" section in the online help.

BITS	Identifier	Description	IEC870
18	_VSB_I_BIT	Invalid	This bit is set if there is a problem in the communication with the driver or with a single variable. The evaluation of single variables is only supported by spontaneous drivers. Most of the zenon drivers however are polling drivers, thus only a general problem in the communication can be indicated (and not a separate one for each variable).
44	_VSB_BL_BIT	IEC 60870 Status: blocked	Signals IEC status blocked in accordance with standard 60870-101 or 104. The value is blocked for transmission and stays in this status it has before it was blocked. This status bit can be selected in Multi reaction matrices, in Combined elements and in the Interlocking formula.
45	_VSB_SP_BIT	Substituted	Signals IEC status substituted in accordance with standard 60870-101 or 104. The value was set by an operator or an automatic source. This status bit can be selected in Multi reaction matrices, in Combined elements and in the Interlocking formula.

BITS	Identifier	Description	IEC870
46	_VSB_NT_BIT	IEC 60870 Status: not topical	Signals IEC status not topical in accordance with standard 60870-101 or 104. The value was not updated or was not available for a certain period of time. This status bit can be selected in Multi reaction matrices, in Combined elements and in the Interlocking formula.
47	_VSB_OV_BIT	Overflow	Signals Overflow in accordance with standard 60870-101 or 104. The value lies beyond the predefined bandwidth.
48	_VSB_SE_BIT	Select	This S/E bit is used in accordance with standard IEC 60870-101 or 104 together with functionality Select before operate and serves for distinction between Select and Execute states of a command.
49	not defined	Time invalid	T_INVAL is set by driver IEC870 if the received real time stamp is marked as invalid. In this case, the local PC time is stamped. In the IEC870 Slave of the Process Gateways the T_INVAL is forwarded in message direction in the time stamp.

4. Double Point Position

When using double point information for Status or Control, the values stand for the states as listed below:

0: Indeterminate or intermediate state

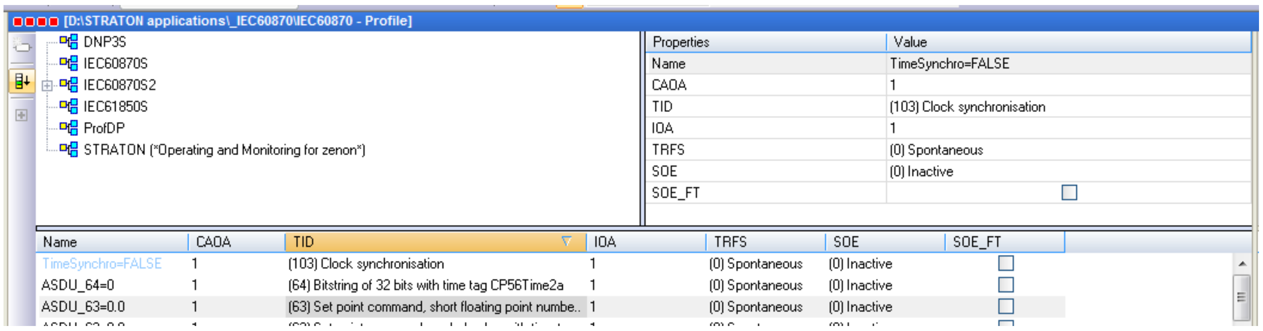
1: Off

2: On

3: Indeterminate state

5. Time Synchronization

The Time Synchronization depends on the OEM integration. This function can be deactivated creating a Boolean variable set to False connected to: "TID (103) Clock synchronization"



[D:\STRATON applications_IEC60870\IEC60870 - Profile]

- DNP3S
- IEC60870S
- IEC60870S2
- IEC61850S
- ProfDP
- STRATON (*Operating and Monitoring for zenon*)

Properties	Value
Name	TimeSynchro=FALSE
CAOA	1
TID	(103) Clock synchronisation
IOA	1
TRFS	(0) Spontaneous
SOE	(0) Inactive
SOE_FT	☐

Name	CAOA	TID	IOA	TRFS	SOE	SOE_FT
TimeSynchro=FALSE	1	(103) Clock synchronisation	1	(0) Spontaneous	(0) Inactive	☐
ASDU_64=0	1	(64) Bitstring of 32 bits with time tag CP56Time2a	1	(0) Spontaneous	(0) Inactive	☐
ASDU_63=0.0	1	(63) Set point command, short floating point numbe..	1	(0) Spontaneous	(0) Inactive	☐

6. Control Object

Some variables in the 60870 Slave can be set by the Master, for example:

T45 – Single command

T46 – Double command

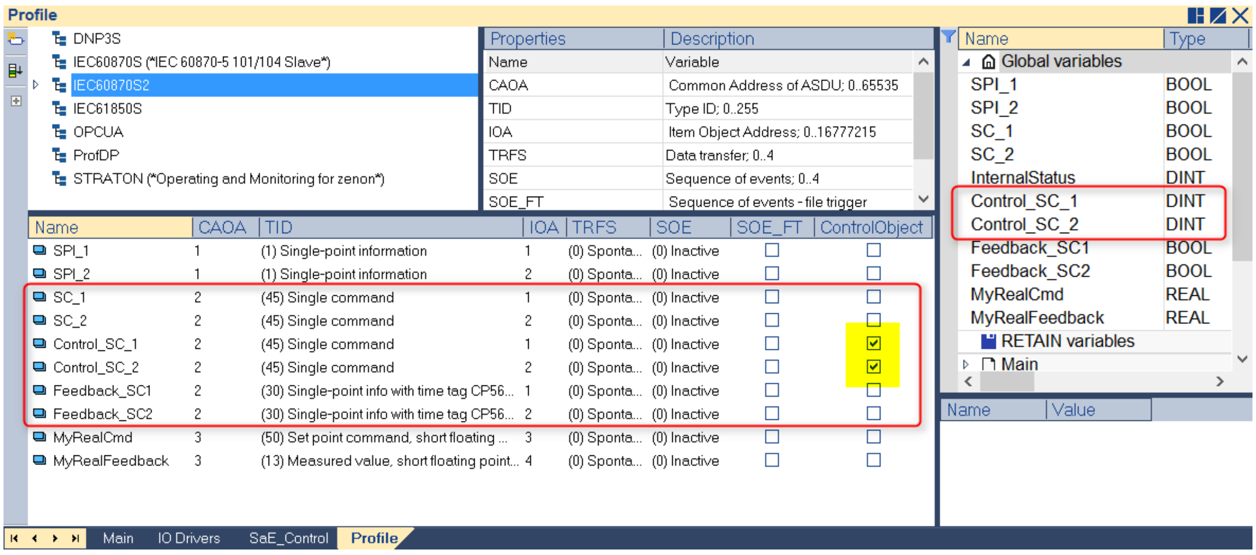
…

These variables can be set easily by the 60870 Master using the IEC60870_EXEC block ("Direct execute"). However, it is also possible to assign a Control-Object to some of them in order to use a "Select before execute" scheme.

There are two specific cases, described in the following parts.

6.1. TID <45..47> and <58..60>

The Control-Object must be a DINT, must have the same CAOA and IOA as the controlled variable and must be a "ControlObject". For example:



Profile
DNP3S
IEC60870S (*IEC 60870-5 101/104 Slave*)
IEC60870S2
IEC61850S
OPCUA
ProfDP
STRATON (*Operating and Monitoring for zenon*)
Main
IO Drivers
SaE_Control
Profile

Properties	Description
Name	Variable
CAOA	Common Address of ASDU; 0..65535
TID	Type ID; 0..255
IOA	Item Object Address; 0..16777215
TRFS	Data transfer; 0..4
SOE	Sequence of events; 0..4
SOE_FT	Sequence of events - file trigger

Name	CAOA	TID	IOA	TRFS	SOE	SOE_FT	ControlObject
SPI_1	1	(1) Single-point information	1	(0) Sponta...	(0) Inactive	☐	☐
SPI_2	1	(1) Single-point information	2	(0) Sponta...	(0) Inactive	☐	☐
SC_1	2	(45) Single command	1	(0) Sponta...	(0) Inactive	☐	☐
SC_2	2	(45) Single command	2	(0) Sponta...	(0) Inactive	☐	☐
Control_SC_1	2	(45) Single command	1	(0) Sponta...	(0) Inactive	☐	☑
Control_SC_2	2	(45) Single command	2	(0) Sponta...	(0) Inactive	☐	☑
Feedback_SC1	2	(30) Single-point info with time tag CP56...	1	(0) Sponta...	(0) Inactive	☐	☐
Feedback_SC2	2	(30) Single-point info with time tag CP56...	2	(0) Sponta...	(0) Inactive	☐	☐
MyRealCmd	3	(50) Set point command, short floating ...	3	(0) Sponta...	(0) Inactive	☐	☐
MyRealFeedback	3	(13) Measured value, short floating point...	4	(0) Sponta...	(0) Inactive	☐	☐

Name	Type
Global variables	
SPI_1	BOOL
SPI_2	BOOL
SC_1	BOOL
SC_2	BOOL
InternalStatus	DINT
Control_SC_1	DINT
Control_SC_2	DINT
Feedback_SC1	BOOL
Feedback_SC2	BOOL
MyRealCmd	REAL
MyRealFeedback	REAL
RETAIN variables	
Main	

Name	Value

Then, on the Master side, the IEC60870_SELECT and IEC60870_EXEC blocks can be used (only if the Master is straton or zenon logic). The "Value" parameter in the blocks must be the value to obtain for the Slave variable.

On the Slave side, the Control Object variable is composed of different specific bits

7	6	5	4	3	2	1	0
/	Timeout	Select	Execute	Cancel	Positive response	Negative response	Termination
	Set by the driver's stack				To be set by the Slave		

For example, when the Master sends a Select command then, on the Slave side, the bit #5 is automatically set by the stack (Select received).

After receiving this command, the bit #1 or the bit #2 (positive or negative response) must be set by the Slave application (some program), this will automatically send a message to the Master, containing the Cause of Transmission (COT) information.

From this COT result, the Master is able, or not, to send an Execute command (see the online help or the 60870-5-101 international standard for the different values of the COT).

Typically, the classical behavior, 'Select before Executing', can be described like that:

Step #	Description	COT*	ControlObject on the Slave side**
1	Master sends a SELECT command	<6> Act	0b0010_0000 (32)
2	Slave answers positively	<7> ActCon	0b0010_0100 (36)
3	Master sends an EXEC command	<6> Act	0b0011_0000 (48)
4	Slave answers positively	<3> Spont	0b0011_0100 (52)
5	Slave – Activation confirmation	<7> ActCon	0b0011_0101 (53)
6	Slave – Activation termination	<10> ActTerm	0b0000_0000 (0)

*The Cause of Transmission is automatically set by the driver
** In blue = automatically set by the driver / In red = must be set (by a program) in the Slave application

The 'Cancel' procedure can be described like that:

Step #	Description	COT*	ControlObject on the Slave side**
1	Master sends a SELECT command	<6> Act	0b0010_0000 (32)
2	Slave answers positively	<7> ActCon	0b0010_0100 (36)
3	Master sends an CANCEL command	<8> DeAct	0b0010_1000 (40)
4	Slave – Deactivation confirmation	<9> DeActCon	0b0010_1100 (44)
5	(No termination)	/	0b0000_0000 (0)

The 'Timeout reached' behavior can be described like that:

Step #	Description	COT*	ControlObject on the Slave side**
1	Master sends a SELECT command	<6> Act	0b0010_0000 (32)
2	Slave answers positively	<7> ActCon	0b0010_0100 (36)
3	Timeout reached	<10> ActTerm	0b0110_0000 (96)
4	(Terminated)	/	0b0000_0000 (0)

NOTES: the timeout is set in the Slave application, in the Fieldbus Configurations, "Select Timeout (ms)" parameter. The timeout bit is automatically set when this timeout is reached.

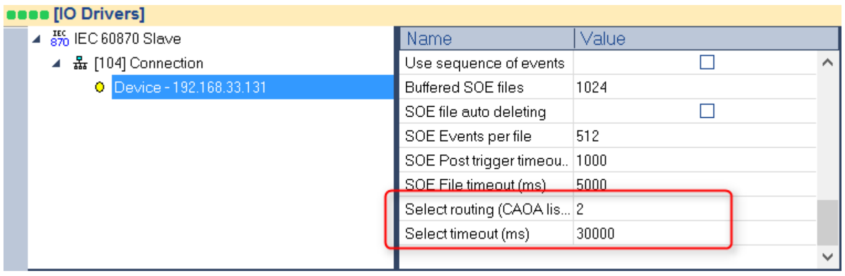
IMPORTANT:

In the Slave application, do not forget to precise the sectors (Common Address of ASDUs – CAOA) that uses the 'Select' model.

Here in this example, the variables in the CAOA number 2 uses a Control Object.

The "Commands active" parameter must be selected and the "Select timeout" parameter must also be set.

Values sent as feedback of a command must be configured manually. By default, if these are send spontaneously,n these will have the COT = 3 = Spontaneous. This can be changed; see the FAQ part of this document.



[IO Drivers]

- IEC 60870 Slave
  - [104] Connection
    - Device - 192.168.33.131

Name	Value
Use sequence of events	☐
Buffered SOE files	1024
SOE file auto deleting	☐
SOE Events per file	512
SOE Post trigger timeou..	1000
SOE File timeout (ms)	5000
Select routing (CAOA lis...	2
Select timeout (ms)	30000

An example of a simple Slave application is available in Annex A of this document as well as an example of state machine managing control commands on single point data.

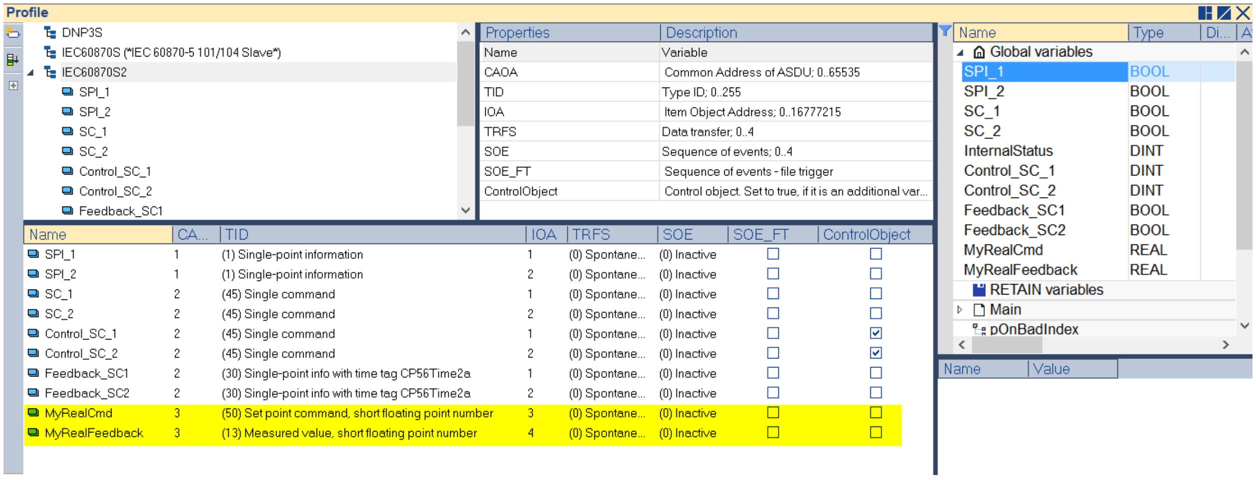
6.2. TID <48..50>

The principle for the "Set Point Command", T48, T49 and T50, is almost the same, except that:

No ControlObject variable is needed.

The positive answer after a SELECT command (COT <6> Act) is sent automatically by the Slave (COT <7> ActCon). It means this has not to be managed by the Slave application.

The CAOA of these variables must not be in the "Select routing" list parameter of the configuration.



Profile

- DNP3S
- IEC60870S (*IEC 60870-5 101/104 Slave*)
- IEC60870S2
  - SPI_1
  - SPI_2
  - SC_1
  - SC_2
  - Control_SC_1
  - Control_SC_2
  - Feedback_SC1

Properties	Description
Name	Variable
CAOA	Common Address of ASDU; 0..65535
TID	Type ID; 0..255
IOA	Item Object Address; 0..16777215
TRFS	Data transfer; 0..4
SOE	Sequence of events; 0..4
SOE_FT	Sequence of events - file trigger
ControlObject	Control object. Set to true, if it is an additional var...

Name	CA...	TID	IOA	TRFS	SOE	SOE_FT	ControlObject
SPI_1	1	(1) Single-point information	1	(0) Spontane...	(0) Inactive	☐	☐
SPI_2	1	(1) Single-point information	2	(0) Spontane...	(0) Inactive	☐	☐
SC_1	2	(45) Single command	1	(0) Spontane...	(0) Inactive	☐	☐
SC_2	2	(45) Single command	2	(0) Spontane...	(0) Inactive	☐	☐
Control_SC_1	2	(45) Single command	1	(0) Spontane...	(0) Inactive	☐	☑
Control_SC_2	2	(45) Single command	2	(0) Spontane...	(0) Inactive	☐	☑
Feedback_SC1	2	(30) Single-point info with time tag CP56Time2a	1	(0) Spontane...	(0) Inactive	☐	☐
Feedback_SC2	2	(30) Single-point info with time tag CP56Time2a	2	(0) Spontane...	(0) Inactive	☐	☐
MyRealCmd	3	(50) Set point command, short floating point number	3	(0) Spontane...	(0) Inactive	☐	☐
MyRealFeedback	3	(13) Measured value, short floating point number	4	(0) Spontane...	(0) Inactive	☐	☐

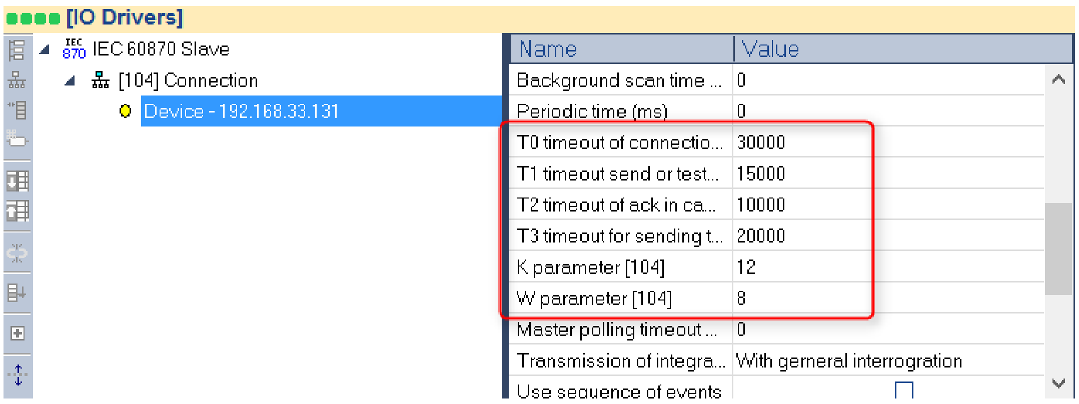
Name	Type	Di...	A
Global variables			
SPI_1	BOOL		
SPI_2	BOOL		
SC_1	BOOL		
SC_2	BOOL		
InternalStatus	DINT		
Control_SC_1	DINT		
Control_SC_2	DINT		
Feedback_SC1	BOOL		
Feedback_SC2	BOOL		
MyRealCmd	REAL		
MyRealFeedback	REAL		
RETAIN variables			
Main			
pOnBadIndex			

Name	Value

7. Frequently Asked Questions

I have connection problems...

Make sure that the SCADA and the iec870 Slave have the same settings for timeouts T0 to T3 and numbers of frames K and W.



Name	Value
Background scan time ...	0
Periodic time (ms)	0
T0 timeout of connectio...	30000
T1 timeout send or test...	15000
T2 timeout of ack in ca...	10000
T3 timeout for sending t...	20000
K parameter [104]	12
W parameter [104]	8
Master polling timeout ...	0
Transmission of integra...	With gemeral interrogration
Use sequence of events	☐

Note: these settings must be documented in the Interoperability List of the Master.

Most problems are also made by Masters implemented according to Edition 1 of the Standard and never updated to Edition 2 (which is the only one valid). Then the Master has wrong handling of timeout T1. With such Master the Slave can communicate only when you deactivate in the Slave the T1 timeout check by setting value 0.

Do I have to use IEC60870S or IEC60870S2 Profile?

IEC60870S2 profiles must be used, the previous one is only specific to old applications.

How to check that a Command is well sent by the Master and treated by the Slave?

Some feedback variable can be created on the Slave side, for example:

SC_1	CAOA = 1	T45	IOA = 1
Feedback_SC1	CAOA = 1	T30	IOA = 10

In an ST program:
```
Feedback_SC1 := SC_1;
```

How to use a different COT for feedback variables?

Set the option "**Use specific COT**" on the third level of the Fieldbus Configurations. Then COT 11 and COT 12 (return info caused by a remote/local command) using the VSI bits _VSB_TCB0, to _VSB_TCB3 can be used.

For example, in order to send the *Feedback_SC1* variable with COT = 11 use:

```
VSISETBIT(Feedback_SC1, _VSB_TCB0, TRUE);
VSISETBIT(Feedback_SC1, _VSB_TCB1, TRUE);
VSISETBIT(Feedback_SC1, _VSB_TCB3, TRUE);
```

How to establish a serial connection on IEC60870-5-101

When you want to create a serial connection, you will need to connect a PORT COM on Windows with a PORT /dev/ttyUSBx on Linux.

On Windows:

Port Number: Put the number of the PORT COM which was opened.

Port COM1 : Port Number:1

On the Raspberry

- Step 1

Check which USB port is used with this command:

# ls /dev/tty*

You should create a symbolic link between the /dev/ttUSBx and the /dev/ttySx used on Straton.

Example: if you use the /dev/ttUSB0, you should link it with /dev/ttS1 with this command:

# sudo ln -s /dev/ttyUSB0 /dev/ttyS1

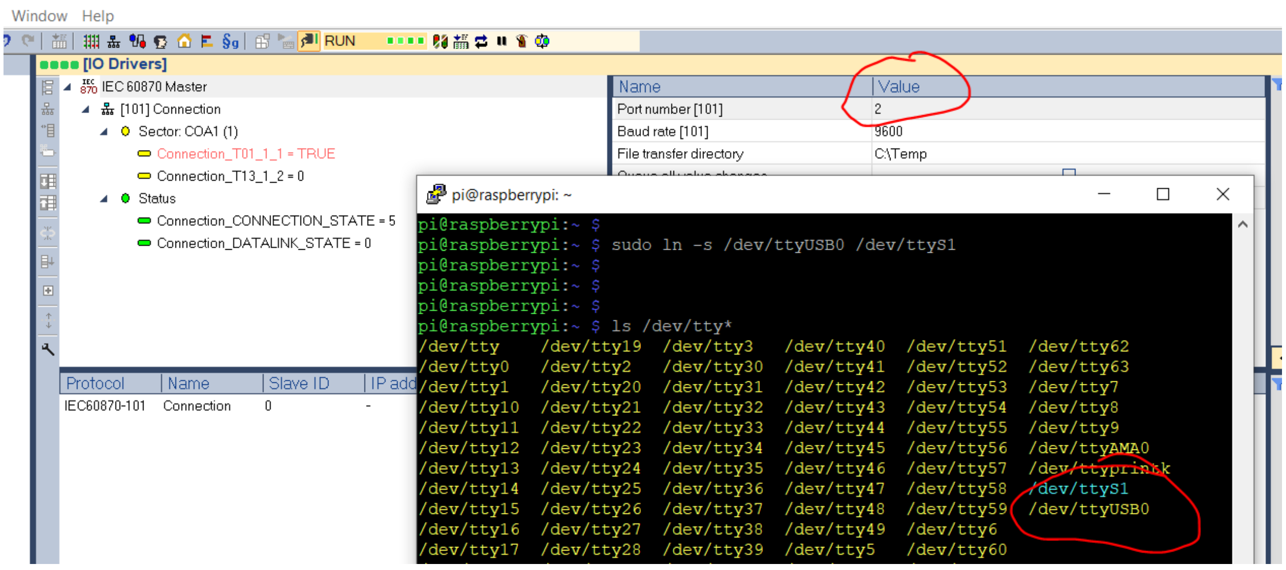
- Step 2

There is a lag of 1 between the Raspberry and the Straton. Like that:

ttyS1 = Port Number : 2

ttyS2 = Port Number : 3

You can look on the example below.



IEC 60870 Master
[101] Connection
Sector: COA1 (1)
Connection_T01_1_1 = TRUE
Connection_T13_1_2 = 0
Status
Connection_CONNECTION_STATE = 5
Connection_DATALINK_STATE = 0

Name	Value
Port number [101]	2
Baud rate [101]	9600
File transfer directory	C:\Temp

Protocol	Name	Slave ID	IP add
IEC60870-101	Connection	0	-

pi@raspberrypi: ~

```
pi@raspberrypi:~ $
pi@raspberrypi:~ $ sudo ln -s /dev/ttyUSB0 /dev/ttyS1
pi@raspberrypi:~ $
pi@raspberrypi:~ $
pi@raspberrypi:~ $
pi@raspberrypi:~ $ ls /dev/tty*
/dev/tty     /dev/tty19  /dev/tty3   /dev/tty40  /dev/tty51  /dev/tty62
/dev/tty0    /dev/tty2   /dev/tty30  /dev/tty41  /dev/tty52  /dev/tty63
/dev/tty1    /dev/tty20  /dev/tty31  /dev/tty42  /dev/tty53  /dev/tty7
/dev/tty10   /dev/tty21  /dev/tty32  /dev/tty43  /dev/tty54  /dev/tty8
/dev/tty11   /dev/tty22  /dev/tty33  /dev/tty44  /dev/tty55  /dev/tty9
/dev/tty12   /dev/tty23  /dev/tty34  /dev/tty45  /dev/tty56  /dev/ttyAMA0
/dev/tty13   /dev/tty24  /dev/tty35  /dev/tty46  /dev/tty57  /dev/ttyprintk
/dev/tty14   /dev/tty25  /dev/tty36  /dev/tty47  /dev/tty58  /dev/ttyS1
/dev/tty15   /dev/tty26  /dev/tty37  /dev/tty48  /dev/tty59  /dev/ttyUSB0
/dev/tty16   /dev/tty27  /dev/tty38  /dev/tty49  /dev/tty6
/dev/tty17   /dev/tty28  /dev/tty39  /dev/tty5   /dev/tty60
```

In linux to open a Port COM

# cat /dev/ttyUSB0

To write on:

# echo Hello > /dev/ttyUSB0

8. Annex A – Example of Slave application managing the Control Object variable

MASTER APPLICATION:

If the Master is straton or zenon logic, use the IEC60870_SELECT and IEC60870_EXEC blocks to send commands. Think to use the "COT" output of these blocks.

SLAVE APPLICATION:

This is only an example! There are many ways to implement the Control-Object management. In this example, the Main program calls a function managing the Control Object.

THE MAIN PROGRAM IS:

```
(* Main *)
```

```
// Function SaE (Select and Execute) is called continuously
// When its output is TRUE, program enters in the IF loop
IF SaE_Control    (Control_SC_1, // Control variable, to be set in the Profiles
                  FALSE,      // SelectNegative
                  FALSE,      // ExecuteNegative
                  TRUE, //  ExecuteDone  -  Hardware  always  is  able  to  execute
successfully
                  SC1   // Target value received during Select and Operate
                  ) THEN
    // Command execution on Hardware - only if an EXEC is sent by the Master
    Feedback_SC1 := SC1;
END_IF;
```

THE SAE_CONTROL FUNCTION IS:

```
(* Subprogram
    STATE-MACHINE for Control of 'Select and Execute' sequence of command
execution. It has no individual Timeouts (else it must be a UDFB)
PARAMETERS
  IN/OUT:
   DataObject_Control value of iec870 slave - state machine
        bit: 7 6 5 4 3 2 1 0
             x | | | | | | |
               | | | | | | + termination        \
               | | | | | + - negative response   - to set in program here -- OUT
               | | | | + - - positive response  /
               | | | |
               | | | + - - - Cancel     \
               | | + - - - - Execute     - set by driver's stack -- IN
               | + - - - - - Select     /
               + - - - - - - Timeout   /
  IN:
    SelectNegative  - when TRUE then Select will be refused - responded negative
    ExecuteNegative - when TRUE then Execute or termination will be responded
negative
    ExecuteDone     - when TRUE the current Execute will be terminated (HW done
the command)
    TargetValue     - target value, in order to check the Select and Operate
matches

  OUT:
    ExecuteOnHW - TRUE means the command should be executed (send to) Hardware
    /!\ It is TRUE for at least one cycle; from Execute activation until
termination (ExecuteDone)

  INOUT:
    DataObject_Control - The Control Object variable
    /!\ Think to set the option "Complex variables in a separate segment" in the
    project settings, otherwise INOUT is not supported by sub-programs
*)
```

```
ExecuteOnHW := FALSE;

CASE DataObject_Control OF

  2#00100000 : // 32 - wants Select

    SelectValue := Target;  // Save the target state in the Select
    IF SelectNegative THEN
        DataObject_Control += 2; // negative - sets bit number 1
    ELSE
        DataObject_Control += 4; // positive - sets bit number 2
    END_IF;
    PRINTF('Wants Select - DOControl = %ld', DataObject_Control);


  2#00110000 : // 48 - wants Execute, Selected

    IF ExecuteNegative OR (SelectValue <> Target) THEN  // If we want to send a
neg. resp. OR Select value does not correspond to Exec value
        DataObject_Control += 2; // negative
    ELSE
        DataObject_Control += 4; // positive
        ExecuteOnHW := TRUE;     // flag for command execution on Hardware
    END_IF;
    PRINTF('Wants Execute - DOControl = %ld', DataObject_Control);



  2#00110100 : // 52 - in Execution, waits until HD done command

    IF ExecuteDone THEN
        IF ExecuteNegative THEN
            DataObject_Control := 2#00110011; //negative termination
        ELSE
            DataObject_Control := 2#00110101; //positive termination
        END_IF;
        PRINTF('Execution on HW finished - DOControl = %ld', DataObject_Control);
    ELSE
```

```
        ExecuteOnHW := TRUE;
    END_IF;

  2#00101000 : // 40 - wants Cancel (of Select)

    DataObject_Control += 4; //positive
    PRINTF('Wants Cancel - DOControl = %ld', DataObject_Control);

ELSE
    // Timeout-bit from stack - Control.6 by CONNECTION LOSS
    // (or not considered CASE)
    IF DataObject_Control.6 THEN
        PRINTF('Timeout from stack - DOControl = %ld', DataObject_Control);
        DataObject_Control := 0; //idle
        PRINTF('DOControl = %ld - Idle', DataObject_Control);
    END_IF;

END_CASE;
```