




# BE1-FLEX

Protection, Automation, and Control System

*Communication Integration*



 **WARNING:** California's Proposition 65 requires special warnings for products that may contain chemicals known to the state of California to cause cancer, birth defects or other reproductive harm. Please note that by posting this Proposition 65 warning, we are notifying you that one or more of the Proposition 65 listed chemicals may be present in products we sell to you. For more information about the specific chemicals found in this product, please visit <https://www.basler.com/Prop65>.

# Preface

This instruction manual provides the information needed to integrate the BE1-FLEX Protection, Automation, and Control System into Ethernet- or serial-based communication. To accomplish this, the following information is provided:

- Communication setup
- Modbus communication
- DNP communication
- Synchrophasor configuration
- IEC 61850 communication

## ***Conventions Used in this Manual***

---

Important safety and procedural information is emphasized and presented in this manual through warning, caution, and note boxes. Each type is illustrated and defined as follows.

### **Warning!**

Warning boxes call attention to conditions or actions that may cause personal injury or death.

### **Caution**

Caution boxes call attention to operating conditions that may lead to equipment or property damage.

### **Note**

Note boxes emphasize important information pertaining to installation or operation.



12570 State Route 143  
Highland IL 62249-1074 USA

[www.basler.com](http://www.basler.com)

[info@basler.com](mailto:info@basler.com)

Tel: +1 618.654.2341

Fax: +1 618.654.2351

© 2023 by Basler Electric

All rights reserved

First printing: May 2021

### Warning!

READ THIS MANUAL. Read this manual before installing, operating, or maintaining this equipment. Note all warnings, cautions, and notes in this manual as well as on the product. Keep this manual with the product for reference. Only qualified personnel should install, operate, or service this system. Failure to follow warning and cautionary labels may result in personal injury or property damage. Exercise caution at all times.

### Caution

Installing previous versions of firmware may result in compatibility issues causing the inability to operate properly and may not have the enhancements and resolutions to issues that more recent versions provide. Basler Electric highly recommends using the latest version of firmware at all times. Using previous versions of firmware is at the user's risk and may void the warranty of the unit.

Basler Electric does not assume any responsibility to compliance or noncompliance with national code, local code, or any other applicable code. This manual serves as reference material that must be well understood prior to installation, operation, or maintenance.

For terms of service relating to this product and software, see the *Commercial Terms of Products and Services* document available at [www.basler.com/terms](http://www.basler.com/terms).

This publication contains confidential information of Basler Electric Company, an Illinois corporation. It is loaned for confidential use, subject to return on request, and with the mutual understanding that it will not be used in any manner detrimental to the interests of Basler Electric Company and used strictly for the purpose intended.

It is not the intention of this manual to cover all details and variations in equipment, nor does this manual provide data for every possible contingency regarding installation or operation. The availability and design of all features and options are subject to modification without notice. Over time, improvements and revisions may be made to this publication. Before performing any of the following procedures, contact Basler Electric for the latest revision of this manual.

The English-language version of this manual serves as the only approved manual version.

# Revision History

A historical summary of the changes made to this instruction manual is provided below. Revisions are listed in reverse chronological order.

Visit [www.basler.com](http://www.basler.com) to download the latest hardware, firmware, and BESTCOMSPlus® revision histories.

## Instruction Manual Revision History

Manual Revision and Date	Change
B, Jul 2023	<ul style="list-style-type: none"><li>• Updated description of Indicators and Remote Indicators in IEC 61850</li><li>• Updated GOOSE and Ethernet Settings description in IEC 61850</li></ul>
A, Feb 2023	<ul style="list-style-type: none"><li>• Added <i>IEC 61850 Communication</i> chapter</li><li>• Added description of SBO Operate Registers in Modbus</li><li>• Minor text edits throughout manual</li></ul>
—, May 2021	<ul style="list-style-type: none"><li>• Initial release</li></ul>



# Contents

Communication Setup.....	1-1
Modbus® Communication.....	2-1
DNP Communication.....	3-1
Synchrophasor Configuration.....	4-1
IEC 61850 Communication .....	5-1



# 1 • Communication Setup

## Introduction

The BE1-FLEX can support Modbus®, DNP3, Synchrophasor, and IEC 61850 communication protocols. Applying a protocol requires the configuration of the appropriate communication ports and settings.

## Communication Interface

Depending upon the BE1-FLEX style specified, RS-485 and/or Ethernet communication is possible.

### RS-485 Port

The BE1-FLEX RS-485 interface consists of three terminals located on the back-panel power supply board connector. These terminals are designated Send/Receive A (terminal 1), Send/Receive B (terminal 2), and Signal Ground (C [terminal 3]).

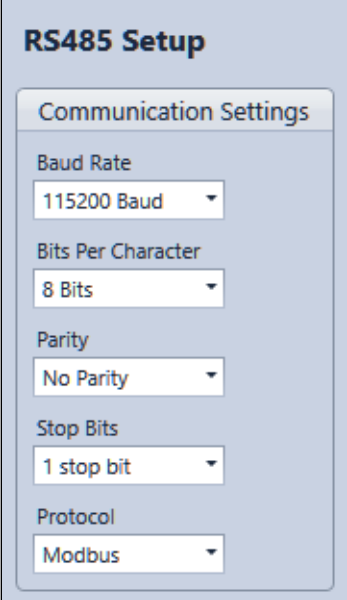
### Ethernet Port

Depending upon the style specified, the BE1-FLEX may be equipped with either an RJ45 copper Ethernet port or a fiber optic Ethernet port. The port is located on the relay back panel. Fiber optic Ethernet communication uses a 100Base-FX, LC connector. Copper Ethernet communication uses a 10/100/1000Base-T, RJ45 connector. 10Base-T applies only to copper Ethernet port 1. All copper ports support 100/1000Base-T communication. Refer to the *Specifications* chapter of publication 9579200990 for the Ethernet port operating specifications.

## RS-485 Port Settings

**Navigation Path:** [Communications, RS-485 Setup](#)

RS-485 communication requires configuration of the RS-485 port settings found on the RS-485 Setup screen (Figure 1-1) of BESTCOMSP<sup>Plus</sup>.



The screenshot displays the 'RS485 Setup' configuration window. It features a title bar 'RS485 Setup' and a sub-section 'Communication Settings'. Below this, there are five dropdown menus for configuration: 'Baud Rate' is set to '115200 Baud', 'Bits Per Character' is set to '8 Bits', 'Parity' is set to 'No Parity', 'Stop Bits' is set to '1 stop bit', and 'Protocol' is set to 'Modbus'.

Figure 1-1. RS-485 Setup Settings



## 2 • Modbus® Communication

### Introduction

BE1-FLEX systems equipped with Protocol Package 01 or 02 have Modbus® capability. In Modbus client/server transactions (queries), the BE1-FLEX always functions as a server responding to the client.

#### Caution

This product contains one or more nonvolatile memory devices. Nonvolatile memory is used to store information that needs to be preserved when the product is power-cycled or otherwise restarted. Established nonvolatile memory technologies have a physical limit on the number of times they can be erased and written. In this product, the limit is 20 TB writes. During product application, consideration should be given to communications, logic, and other factors that may cause frequent/repeated writes of settings or other information that is retained by the product. Applications that result in such frequent/repeated writes may reduce the useable product life and result in loss of information and/or product inoperability.

### Modbus Communication Settings

When equipped with Protocol Package 01 or 02, the BE1-FLEX supports Modbus communication via RS-485 or Ethernet. Using either communication method requires configuration of the appropriate settings. Refer to Chapter 1, *Communication Setup* for information about general RS-485 settings.

**Navigation Path:** [Communications](#), [Modbus](#), [Modbus Settings](#)

The BESTCOMSPi.us Modbus settings screen of Figure 2-1 provides Modbus configuration settings for RS-485 and Ethernet.

Enabling the Auto Save setting enables group saves of settings which reduces write times.

**Modbus Settings**

RS-485 Settings

Serial Unit ID  
1

Response Delay (ms)  
10

Ethernet Settings

Ethernet Unit ID  
1

Auto Save Settings

Auto Save  
Disabled

Figure 2-1. Modbus Settings

## Modbus Map

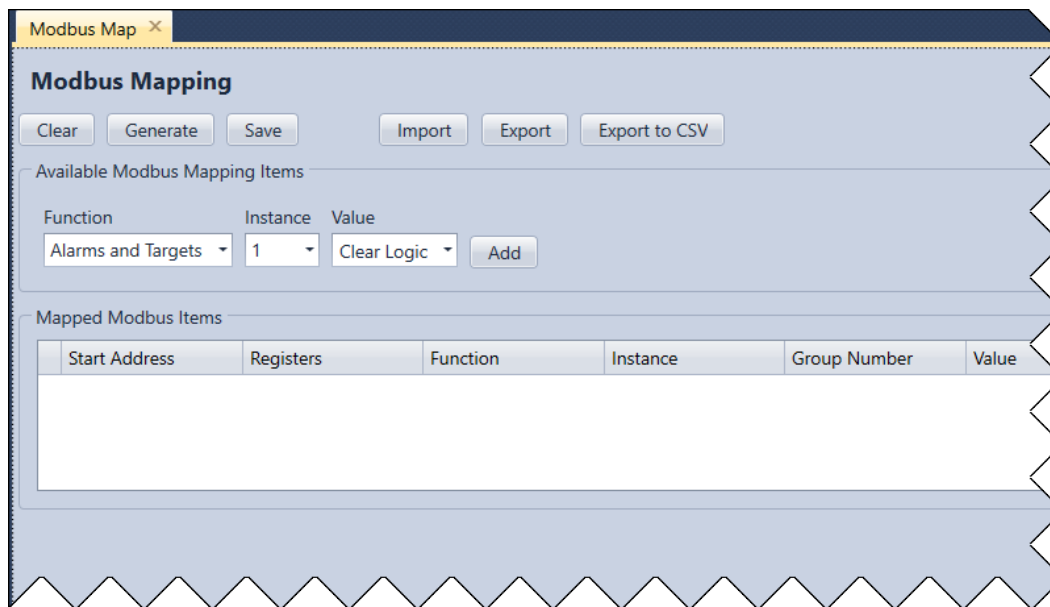
**Navigation Path:** Communications, Modbus, Modbus Map

The Modbus map of a BE1-FLEX is fully customizable in BESTCOMSP<sup>l</sup>us<sup>®</sup> to meet application requirements. See Figure 2-2. The address of a mapped item can be edited and unneeded variables can be deleted as desired. A custom map is easily exported from or imported to BESTCOMSP<sup>l</sup>us<sup>®</sup> for use with multiple BE1-FLEX systems. A Modbus map also can be exported directly to a CSV file for documentation purposes.

### Map Creation

The BE1-FLEX defaults to an empty Modbus map. Clicking the Generate button will display an Auto Populate Items window showing the available object categories for quick setup. Selecting the desired object categories will populate the map with all items related to the selected categories.

Individual items can be added to the map by using the Function, Instance, and Value controls of the Available Modbus Mapping Items section. An item and its attributes can be selected and then placed in the map by using the Add button.



**Figure 2-2. Modbus Mapping Window**

### Map Example

A sample map, for illustration only, is shown in Figure 2-3. This illustration shows all of the labels and parameters associated with the selected items. A filter function is available for each column and is useful for displaying only the data of interest in the map.

Mapped Modbus Items									
Start Address	Registers	Function	Instance	Group Number	Value	Datatype	Read/Write		
72	5	Device Info Service		1	Application Version	StringType	ReadOnly	Delete	
77	7	Com Service		1	Address 1	StringType	ReadWrite	Delete	
85	2	Database		1	Save_Key Save	IntType	ReadWrite	Delete	
99	2	I/O Slot		2	Sensing Level	IntType	ReadWrite	Delete	
103	2	Logic Constants		1	Logic 1	BoolType	ReadOnly	Delete	
105	2	Logic Counter		1	Trigger Count	UIntType	ReadWrite	Delete	
107	2	Modbus		1	Ethernet Unit ID	IntType	ReadWrite	Delete	
109	2	System Status		1	LCD Backlight Fault	BoolType	ReadOnly	Delete	
111	2	System Status		1	Slot 2 Hardware Er...	BoolType	ReadOnly	Delete	
113	2	Undervoltage (27)		1	Group0	FloatType	ReadWrite	Delete	
115	2	Alarm Status		1	Alarms Reset	BoolType	ReadOnly	Delete	
117	2	Setting Group		1	Active Setting Group	IntType	ReadOnly	Delete	

Figure 2-3. Modbus Map Example

## Map Editing

All items in a map can be removed with the Clear button. Individual items can be removed with the Delete button located to the right of the item.

An item's Start Address can be clicked and changed to meet the application requirements. An address of 72 is the minimum permissible value; lower address numbers are reserved for security functions. Additionally, the map sequence can be edited by clicking and dragging any item within the map. Start Address and the Delete buttons are the only active edit fields in the map.

## Map Saving, Exporting, and Importing

Mapped items can be saved as part of the BE1-FLEX settings file by clicking the Save button.

The Export button is used to save a Modbus map for use with other BE1-FLEX systems. Clicking the Export button displays a Save Modbus Mapping dialog window where the file name and location can be assigned. Modbus map export files are appended with a ".bstm" extension. To retrieve a previously exported Modbus map, click the Import button. This displays an Open Modbus Mapping dialog window that provides navigation to and selection of the desired map file with a ".bstm" extension.

The Export to CSV button exports a Modbus map as a comma-separated values file. Map items are saved in a tabular format that can be viewed in most spreadsheet applications.

## Security Registers

A group of registers provide security login and logout functions and cannot be remapped. Security register functions are summarized in Table 2-1.

Table 2-1. Security Registers

Description	Address	Type	Bytes	Writable	Range
Username	2	String	64	Read/Write	0–64
Password	34	String	64	Read/Write	0–64
Attempt login	66	String	4	Read/Write	Any value written triggers a login with the username and password of addresses 2 and 34.
Logout	68	Integer	4	Read/Write	Write a '1' to trigger logout.

## Database Register

The Database Function "Save" register saves settings to the BE1-FLEX database when the Auto Save setting is disabled. See the *Modbus Communication Settings* section of the *Communication Setup* chapter for more information.

This parameter's default value is 0 and is set to 1 when written and will save changed settings to the BE1-FLEX database. Table 2-2 summarizes the Save register functions.

**Table 2-2. Database Register**

Description	Address	Type	Bytes	Writable	Range
Save_Key Save		IntType	4	Read/Write	0–1

## SBO Registers

The Select Before Operate (SBO) registers operate the control blocks in the BE1-FLEX such as Virtual Switches (43) and Breaker Control (101) Switches. Table 2-3 summarizes the SBO Operate register functions.

**Table 2-3. SBO Operate Register**

Description	Type	Bytes	Writable	Range
Virtual Switches (43-x)	IntType	4	Read/Write	1 = Set 2 = Reset 3 = Pulse
Breaker Control Switch (101-x)	IntType	4	Read/Write	1 = Trip 2 = Close

## Message Structure

A Modbus message consists of four fields: device address, function code, data block, and error check.

### Device Address (Serial Unit ID or Ethernet Unit ID)

The device address field contains the unique Modbus address of the server being queried. The addressed server repeats the address in the device address field of the response message. This field is one byte in length.

The range of permissible device addresses is 1 to 247. This address is user-selectable and can be altered during operation.

### Function Code

The function code field defines the action to be taken by the addressed server. This field is echoed in the response message and altered by setting the most significant bit (MSB) of the field to 1 if the response is an error response. This field is one byte in length.

The BE1-FLEX maps all available data into the Modicon 984 holding register address space and supports the following function codes:

- 03 (03<sub>16</sub>): read holding registers
- 06 (06<sub>16</sub>): preset single register
- 08 (08<sub>16</sub>):
  - Sub-function 00: diagnostics, return query data
  - Sub-function 01: diagnostics, restart communication option
  - Sub-function 04: diagnostics, force listen-only mode
- 16 (10<sub>16</sub>): preset multiple registers

### Data Block

The query data block field contains additional information needed by the server to perform the requested function. The response data block contains data collected by the server for the queried function. An error response will substitute an exception response code for the data block. The length of this field varies with each query.

## Error Check

The error check field provides a method for the server to validate the integrity of the query message contents and allows the client to confirm the validity of response message contents. This field is two bytes in length.

## Modes of Operation

The BE1-FLEX supports Modbus RTU (remote terminal unit) communications over the RS-485 serial interface and Modbus TCP/IP over Ethernet.

A client can query servers individually or universally. A universal (“broadcast”) query, when allowed, evokes no response from any server device. If a query to an individual server device requests actions that cannot be performed by the server, the server response message contains an exception response code defining the error detected. Exception response codes are frequently enhanced by the information provided in the Error Details block of holding registers.

The Modbus protocol defines a simple Protocol Data Unit (PDU) independent of the underlying communication layers. The mapping of the Modbus protocol on specific buses or networks can introduce some additional fields on the Application Data Unit (ADU). See Figure 2-4.

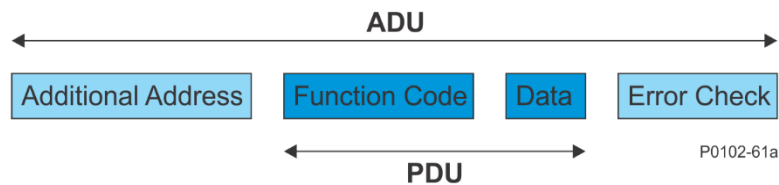


Figure 2-4. General Modbus Frame

The client that initiates a Modbus transaction builds the Modbus Application Data Unit. The function code indicates to the server which kind of action to perform.

## Modbus via RS-485

The BE1-FLEX supports serial communication through an RS-485 interface located on the rear panel.

Each eight-bit byte in a message contains two four-bit hexadecimal characters. The message is transmitted in a continuous stream with the LSB of each byte of data transmitted first. Transmission of each eight-bit data byte occurs with one start bit and either one or two stop bits. When enabled, parity checking is performed and can be either odd or even. The transmission baud rate is user-selectable and can be set at installation and altered during operation. The BE1-FLEX supports a baud rate of 1200, 2400, 4800, 9600, 19200, 38400, 57600, or 115200. The Baud Rate, Bits Per Character, Parity and Stop Bits are configured on the *BESTCOMSPi* RS-485 Setup settings screen. See the *RS-485 Port Settings* section of the *Communication Setup* chapter for more information.

### Message Framing and Timing Considerations

When receiving a message via the RS-485 communication port, the BE1-FLEX requires an inter-byte latency of 3.5 character times before considering the message complete.

Once a valid query is received, the BE1-FLEX waits a specified amount of time before responding. This time delay is set in the Response Delay field of the *BESTCOMSPi* Modbus Settings tab.

Table 2-4 lists the response message transmission time (in seconds) and 3.5-character times (in milliseconds) for various message lengths and baud rates.

Table 2-4. Timing Considerations

Baud Rate	3.5 Character Time (ms)	Message Tx Time (s)	
		128 Bytes	256 Bytes
1200	32.08	1.17	2.34
2400	16.04	0.59	1.17
4800	8.021	0.29	0.59
9600	4.0104	0.15	0.29
19200	2.0052	0.07	0.15
38400	1.0026	0.04	0.07
57600	0.6684	0.02	0.04
115200	0.3342	0.01	0.02

## Modbus via TCP/IP

The encapsulation of a request or response when it is carried on a Modbus TCP/IP network is illustrated in Figure 2-5 and described as follows.

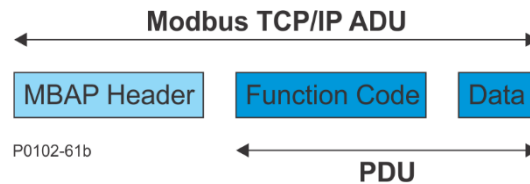


Figure 2-5. Request/Response Over Modbus TCP/IP

A dedicated header is used on TCP/IP to identify the Modbus Application Data Unit. It is called the MBAP header (Modbus Application Protocol header).

This header provides some differences compared to the Modbus RTU application data unit used on a serial line:

- The Modbus 'server address' field usually used on Modbus Serial Line is replaced by a single byte 'Unit Identifier' within the MBAP header. The 'Unit Identifier' is used to communicate via devices such as bridges, routers, and gateways that use a single IP address to support multiple independent Modbus end units.
- All Modbus requests and responses are designed in such a way that the recipient can verify that a message is finished. For function codes where the Modbus PDU has a fixed length, the function code alone is sufficient. For function codes carrying a variable amount of data in the request or response, the data field includes a byte count.
- When Modbus is carried over TCP, additional length information is carried in the MBAP header to allow the recipient to recognize message boundaries even if the message has been split into multiple packets for transmission. The existence of explicit and implicit length rules and use of a CRC-32 error check code (on Ethernet) results in an infinitesimal chance of undetected corruption to a request or response message.

### MBAP Header Description

Table 2-5 lists the fields contained in an MBAP header. A header is seven bytes long. All Modbus/TCP ADU are sent via TCP on registered port 502.

Table 2-5. MBAP Header Fields

Field	Length	Description	Client	Server
Transaction Identifier	2 bytes	Identifies a Modbus request/response transaction. The Modbus server copies in the response the transaction identifier of the request.	Initialized by the client.	Recopied by the server from the received request.
Protocol Identifier	2 bytes	Used for intra-system multiplexing. The Modbus protocol is identified by the value 0.	Initialized by the client.	Recopied by the server from the received request.
Length	2 bytes	A byte count of the fields that follow, including the Unit Identifier and data fields.	Initialized by the client (request).	Initialized by the server (response).
Unit Identifier	1 byte	Used for intra-system routing, it is typically used to communicate to a Modbus serial line server through a gateway between an Ethernet TCP/IP network and a Modbus serial line. This field is set by the Modbus client in the request and must be returned with the same value in the response by the server.	Initialized by the client.	Recopied by the server from the received request.

## ***Detailed Message Query and Response for RTU Transmission Mode***

A detailed description of BE1-FLEX supported message queries and responses is provided as follows.

### **Read Holding Registers**

#### Query

This query message requests a register or block of registers to be read. The data block contains the starting register address and the quantity of registers to be read. A register address of N will read holding register N+1. If the query is a broadcast (device address = 0), no response message is returned.

Device Address  
 Function Code = 03<sub>16</sub>  
 Starting Address Hi  
 Starting Address Lo  
 No. of Registers Hi  
 No. of Registers Lo  
 CRC Hi error check  
 CRC Lo error check

The number of registers cannot exceed 125 without causing an error response with the exception code for an illegal function.

#### Response

The response message contains the data queried. The data block contains the block length in bytes followed by the data (one Data Hi byte and one Data Lo byte) for each requested register.

Reading an unassigned holding register returns a value of zero.

Device Address  
 Function Code = 03<sub>16</sub>  
 Byte Count  
 Data Hi (For each requested register, there is one Data Hi and one Data Lo.)  
 Data Lo  
 .  
 .  
 Data Hi  
 Data Lo  
 CRC Hi error check  
 CRC Lo error check

## Return Query Data

This query contains data to be returned (looped back) in the response. The response and query messages should be identical. If the query is a broadcast (device address = 0), no response message is returned.

Device Address  
 Function Code = 08<sub>16</sub>  
 Subfunction Hi = 00<sub>16</sub>  
 Subfunction Lo = 00<sub>16</sub>  
 Data Hi = xx (don't care)  
 Data Lo = xx (don't care)  
 CRC Hi error check  
 CRC Lo error check

## Restart Communications Option

This query causes the remote communications function of the BE1-FLEX to restart, terminating an active listen only mode of operation. No effect is made upon primary BE1-FLEX operations. Only the remote communications function is affected. If the query is a broadcast (device address = 0), no response message is returned.

If the BE1-FLEX receives this query while in the listen only mode, no response message is generated. Otherwise, a response message identical to the query message is transmitted prior to the communications restart.

Device Address  
 Function Code = 08<sub>16</sub>  
 Subfunction Hi = 00<sub>16</sub>  
 Subfunction Lo = 01<sub>16</sub>  
 Data Hi = xx (don't care)  
 Data Lo = xx (don't care)  
 CRC Hi error check  
 CRC Lo error check

## Listen Only Mode

This query forces the addressed BE1-FLEX to the listen only mode for Modbus communications, isolating it from other devices on the network. No responses are returned.

While in the listen only mode, the BE1-FLEX continues to monitor all queries. The BE1-FLEX does not respond to any other query until the listen only mode is removed. All write requests with a query to Preset Multiple Registers (Function Code = 16) are also ignored. When the BE1-FLEX receives the restart communications query, the listen only mode is removed.

Device Address  
 Function Code = 08<sub>16</sub>  
 Subfunction Hi = 00<sub>16</sub>  
 Subfunction Lo = 04<sub>16</sub>  
 Data Hi = xx (don't care)  
 Data Lo = xx (don't care)  
 CRC Hi error check  
 CRC Lo error check

## Preset Multiple Registers

A preset multiple registers query could address multiple registers in one server or multiple servers. If the query is a broadcast (device address = 0), no response message is returned.

### Query

A Preset Multiple Register query message requests a register or block of registers to be written. The data block contains the starting address and the quantity of registers to be written, followed by the Data Block byte count and data. The BE1-FLEX will perform the write when the device address in query is a broadcast address or the same as the BE1-FLEX Modbus Unit ID (device address).

A register address of N will write Holding Register N+1.

Data will cease to be written if any of the following exceptions occur.

- Queries to write to Read Only registers result in an error response with Exception Code of “Illegal Data Address”.
- Queries attempting to write more than 100 registers cause an error response with Exception Code “Illegal Function”.
- An incorrect Byte Count will result in an error response with Exception Code of “Illegal Data Value”.
- There are several instances of registers that are grouped together to collectively represent a single numerical BE1-FLEX data value (i.e. - floating point data, 32-bit integer data, and strings). A query to write a subset of such a register group will result in an error response with Exception Code “Illegal Data Address”.
- A query to write a not allowed value (out of range) to a register results in an error response with Exception Code of “Illegal Data Value”.

Device Address  
 Function Code =           10<sub>16</sub>  
 Starting Address Hi  
 Starting Address Lo  
 No. of Registers Hi  
 No. of Registers Lo  
 Byte Count  
 Data Hi  
 Data Lo  
 .  
 .  
 Data Hi  
 Data Lo  
 CRC Hi error check  
 CRC Lo error check

### Response

The response message echoes the starting address and the number of registers. There is no response message when the query is a broadcast (device address = 0).

Device Address  
 Function Code =           10<sub>16</sub>  
 Starting Address Hi  
 Starting Address Lo  
 No. of Registers Hi  
 No. of Registers Lo  
 CRC Hi Error Check  
 CRC Lo Error Check

## **Data Formats**

BE1-FLEX systems support the following data types.

- BoolType: 0 = FALSE and 1 = TRUE, a 1 is located in the same location as an IntType value of 1. Mapped to two registers.

- IntType: signed integer 32 (Int32). Mapped to two registers.
- UIntType: unsigned integer 32 (UInt32). Mapped to two registers.
- FloatType: floating point. Mapped to two registers
- StringType: any number of registers,
- DoubleType: 64-bit, double-precision, floating-point number. Mapped to four registers.

### Floating Point Data Format (Float)

The Modbus floating point data format uses two consecutive holding registers to represent a data value. The first register contains the low-order 16 bits of the following 32-bit format:

- MSB is the sign bit for the floating-point value (0 = positive).
- The next 8 bits are the exponent biased by 127 decimal.
- The 23 LSBs comprise the normalized mantissa. The most-significant bit of the mantissa is always assumed to be 1 and is not explicitly stored, yielding an effective precision of 24 bits.

The value of the floating-point number is obtained by multiplying the binary mantissa times two raised to the power of the unbiased exponent. The assumed bit of the binary mantissa has the value of 1.0, with the remaining 23 bits providing a fractional value. Table 2-6 shows the floating-point format.

**Table 2-6. Floating Point Format**

Sign	Exponent + 127	Mantissa
1 Bit	8 Bits	23 Bits

The floating-point format allows for values ranging from approximately  $8.43 \times 10^{-37}$  to  $3.38 \times 10^{38}$ . A floating-point value of all zeroes is the value zero. A floating-point value of all ones (not a number) signifies a value currently not applicable or disabled.

**Example:** The value 95,800 represented in floating-point format is hexadecimal 47BB1C00. This number will read from two consecutive holding registers as follows:

Holding Register	Value
K (Hi Byte)	hex 1C
K (Lo Byte)	hex 00
K+1(Hi Byte)	hex 47
K+1(Lo Byte)	hex BB

The same byte alignments are required to write.

### Integer Data Format (UInt, Nit, Bool)

The Modbus long integer data format uses two consecutive holding registers to represent a 32-bit data value. The first register contains the low-order 16 bits and the second register contains the high-order 16 bits.

**Example:** The value 95,800 represented in long integer format is hexadecimal 0x00017638. This number will read from two consecutive holding registers as follows:

Holding Register	Value
K (Hi Byte)	hex 76
K (Lo Byte)	hex 38
K+1(Hi Byte)	hex 00
K+1(Lo Byte)	hex 01

The same byte alignments are required to write.

## Double Data Format (binary64)

The Modbus double data format uses four consecutive holding registers to represent a data value. The first register contains the low-order 16 bits of the following 64-bit format:

- MSB is the sign bit for the floating-point value (0 = positive).
- The next 11 bits are the exponent biased by 1023 decimal.
- The 52 LSBs comprise the normalized mantissa. The most-significant bit of the mantissa is always assumed to be 1 and is not explicitly stored, yielding an effective precision of 53 bits.

The value of the Double number is obtained by multiplying the binary mantissa times two raised to the power of the unbiased exponent. The assumed bit of the binary mantissa has the value of 1.0, with the remaining 52 bits providing a fractional value. Table 2-7 shows the floating-point format.

**Table 2-7. Double Format**

Sign	Exponent + 1023	Mantissa
1 Bit	11 Bits	52 Bits

The double format allows for values ranging from approximately  $2.23 \times 10^{-38}$  to  $1.80 \times 10^{308}$ . A double value of all zeroes is the value zero. A double value of all ones (not a number) signifies a value currently not applicable or disabled.

**Example:** The value 95,800 represented in floating-point format is hexadecimal 47F7638000000000. This number will read from four consecutive holding registers as follows:

<u>Holding Register</u>	<u>Value</u>
K (Hi Byte)	hex 00
K (Lo Byte)	hex 00
K+1(Hi Byte)	hex 00
K+1(Lo Byte)	hex 00
K+2(Hi Byte)	hex 63
K+2(Lo Byte)	hex 80
K+3(Hi Byte)	hex 47
K+3(Lo Byte)	hex F7

## String Data Format (String)

The Modbus string data format uses one or more holding registers to represent a sequence, or string, of character values. If the string contains a single character, the holding register high byte will contain the ASCII character code and the low byte will be zero.

**Example:** The string "PASSWORD" represented in string format will read as follows:

<u>Holding Register</u>	<u>Value</u>
K (Hi Byte)	'P'
K (Lo Byte)	'A'
K+1(Hi Byte)	'S'
K+1(Lo Byte)	'S'
K+2(Hi Byte)	'W'
K+2(Lo Byte)	'O'
K+3(Hi Byte)	'R'
K+3(Lo Byte)	'D'

**Example:** If the above string is changed to "P", the new string will read as follows:

<u>Holding Register</u>	<u>Value</u>
K (Hi Byte)	'P'
K (Lo Byte)	hex 00
K+1(Hi Byte)	hex 00

K+1(Lo Byte)	hex 00
K+2(Hi Byte)	hex 00
K+2(Lo Byte)	hex 00
K+3(Hi Byte)	hex 00
K+3(Lo Byte)	hex 00

The same byte alignments are required to write.

### CRC Error Check

This field contains a 2-byte CRC value for transmission error detection. The client first calculates the CRC and appends it to the query message. The BE1-FLEX recalculates the CRC value for the received query and performs a comparison to the query CRC value to determine if a transmission error has occurred. If so, no response message is generated. If no transmission error has occurred, the server calculates a new CRC value for the response message and appends it to the message for transmission.

The CRC calculation is performed using all bytes of the device address, function code, and data block fields. A 16-bit CRC register is initialized to all 1's. Then each 8-bit byte of the message is used in the following algorithm:

First, exclusive-OR the message byte with the low-order byte of the CRC register. The result, stored in the CRC register, will then be right-shifted eight times. The CRC register MSB is zero-filled with each shift. After each shift, the CRC register LSB is examined. If the LSB is a 1, the CRC register is exclusive-ORed with the fixed polynomial value A001 (hex) prior to the next shift. Once all bytes of the message have undergone the above algorithm, the CRC register will contain the message CRC value to be placed in the error check field.

### Register Table

Parameters are mapped into the holding register address space in blocks according to the Modbus Map settings.

Any holding register not listed in the map is an unassigned holding register. Reading or writing to unassigned holding registers is not allowed and error code "Illegal Data Address" will be reported.

Reading/Writing any data (variable) as its complete (atomic) value is the only legal operation. Partially requested data will return an error. For example, request for reading/writing of/to only a single register will return an error if this register is part of data that is mapped to two registers. Reading of two registers is correct only if reading/writing of both registers are requested in the same query.

### Conventions

The Type column uses the following abbreviations:

- String - ASCII string
- Float - Floating point
- Int32 - Integer (32-bit integer)
- Uint32 - Unsigned Integer (32-bit integer)
- Double – Double precision (64-bit floating point)

## 3 • DNP Communication

### Introduction

BE1-FLEX systems equipped with Protocol Package 01 or 02 are capable of communication using the Distributed Network Protocol (DNP3). BE1-FLEX systems are classified as intelligent electronic devices (IEDs) that are capable of reacting or responding to specific requests conforming to a level-two server device, as defined in the *DNP Subset Definitions* document.

#### Caution

This product contains one or more nonvolatile memory devices. Nonvolatile memory is used to store information that needs to be preserved when the product is power-cycled or otherwise restarted. Established nonvolatile memory technologies have a physical limit on the number of times they can be erased and written. In this product, the limit is 20 TBW (terabytes written). During product application, consideration should be given to communications, logic, and other factors that may cause frequent/repeated writes of settings or other information that is retained by the product. Applications that result in such frequent/repeated writes may reduce the useable product life and result in loss of information and/or product inoperability.

### References

- BE1-FLEX Protection, Automation, and Control System, Publication 9579200990
- DNP3 Basic 4 Document Set
- DNP Subset Definitions Document
- DNP Website ([www.dnp.org](http://www.dnp.org))

### Device Profile Document

Table 3-1 provides a profile document for the BE1-FLEX in the standard format defined in the DNP3 subset definition document. This device profile document, the implementation table (TABLE), and the points available through the BESTCOMSPPlus® software provide a complete application configuration guide for incorporating the BE1-FLEX into any DNP environment.

**Table 3-1. Device Profile Document**

<b>DEVICE PROFILE DOCUMENT</b>	
Vendor Name: Basler Electric Company	
Device Name: BE1-Flex	
Highest DNP Level Supported: Level 2.	Device Function: <input type="checkbox"/> Client <input checked="" type="checkbox"/> <b>Server</b>
Notable objects, functions, and/or qualifiers supported in addition to the highest DNP levels supported (the complete list is described in DNP3 Implementation Table): <ul style="list-style-type: none"> <li>- For static (non-change-event) object requests, request qualifier codes 00 and 01 (start-stop), 07 and 08 (limited quantity), and 17 and 28 (index) are supported in addition to request qualifier code 06 (no range – or all points).</li> <li>- Static object requests sent with qualifiers 00,01,06,07, and 08, will be responded to with qualifiers 00 or 01.</li> <li>- Static object requests sent with qualifiers 17 and 28 will be responded to with qualifiers 17 or 28.</li> </ul>	

## DEVICE PROFILE DOCUMENT

- Only Analog Inputs and Binary Inputs are included in Class 0.
- The user configures Class 0 data by mapping Analog Input points and Binary Input points in the DNP Map configured with BESTCOMSPlus®.
- A Dead band for each Analog Input point in the DNP Map is configurable via the BESTCOMSPlus DNP Map or via DNP object 34.
- The class assignment for each reporting binary and analog input point is configured via BESTCOMSPlus DNP Map or via DNP assign class function code 22.
- Each Analog Input point has a configurable scaling factor set through BESTCOMSPlus DNP Map.
- Enabling and disabling unsolicited responses is configured on a Class-by-Class basis.
- Device supports DNP over serial (RS-485) port and via Ethernet. Device provides a setting to select the type of Internet Protocol connection to establish with a client datagram end point or a TCP listening end point.
- Default variations for objects 1, 2, 30, 32, and 40 are programmable through BESTCOMSPlus.
- All DNP settings configured through BESTCOMSPlus are saved in non-volatile memory.
- Dead bands and assigned classes, changed via DNP, are saved in non-volatile memory.
- This device does not support Collision Avoidance.
- Select to Operate time delay is 30 seconds.
- Maximum delay measurement error is 50 ms.
- Maximum internal time reference error when set from protocol is 250 ms.
- Maximum response time is 150 ms.
- For Unsolicited Responses, if the number of maximum retries has been reached, the outstation continues to transmit unsolicited responses with larger intervals (off-line intervals) indefinitely until confirmation is received from the client.

Maximum Data Link Frame Size (octets):  
 Transmitted 292  
 Received 292

Maximum Application Fragment Size (octets):  
 Transmitted configurable up to 4096  
 Received 1024

Maximum Data Link Re-tries:  
 None  
 **Fixed at 2 (only if frame sent with confirm requested)**  
 Configurable

Maximum Application Layer Re-tries:  
 None  
 Fixed at  
 **Configurable**

Requires Data Link Layer Confirmation:  
 Never  
 Always  
 Sometimes  
 **Configurable. Default is NEVER.**

Requires Application Layer Confirmation:  
 Never  
 Always (not recommended)  
 **When reporting Event Data**  
 **When sending multi-fragment responses**

Timeouts while waiting for:

Data Link Confirm	<input type="checkbox"/> None	<input checked="" type="checkbox"/> <b>Fixed at 3,000 ms</b>	<input type="checkbox"/> Variable	<input type="checkbox"/> Configurable
Complete Appl. Fragment	<input checked="" type="checkbox"/> <b>None</b>	<input type="checkbox"/> Fixed at _____	<input type="checkbox"/> Variable	<input type="checkbox"/> Configurable
Application Confirm	<input type="checkbox"/> None	<input type="checkbox"/> Fixed at _____	<input type="checkbox"/> Variable	<input checked="" type="checkbox"/> <b>Configurable.</b> <b>5,000 ms default</b>
Complete Appl. Response	<input checked="" type="checkbox"/> <b>None</b>	<input type="checkbox"/> Fixed at _____	<input type="checkbox"/> Variable	<input type="checkbox"/> Configurable

<b>DEVICE PROFILE DOCUMENT</b>				
Sends/Executes Control Operations:				
WRITE Binary Outputs	<input checked="" type="checkbox"/> <b>Never</b>	<input type="checkbox"/> Always	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
SELECT/OPERATE	<input type="checkbox"/> Never	<input checked="" type="checkbox"/> <b>Always</b>	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
DIRECT OPERATE	<input type="checkbox"/> Never	<input checked="" type="checkbox"/> <b>Always</b>	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
DIRECT OPERATE - NO ACK	<input type="checkbox"/> Never	<input checked="" type="checkbox"/> <b>Always</b>	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
Count > 1	<input checked="" type="checkbox"/> <b>Never</b>	<input type="checkbox"/> Always	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
Pulse On	<input type="checkbox"/> Never	<input checked="" type="checkbox"/> <b>Always</b>	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
Pulse Off	<input checked="" type="checkbox"/> <b>Never</b>	<input type="checkbox"/> Always	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
Latch On	<input type="checkbox"/> Never	<input checked="" type="checkbox"/> <b>Always</b>	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
Latch Off	<input type="checkbox"/> Never	<input checked="" type="checkbox"/> <b>Always</b>	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
Queue	<input checked="" type="checkbox"/> <b>Never</b>	<input type="checkbox"/> Always	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
Clear Queue	<input checked="" type="checkbox"/> <b>Never</b>	<input type="checkbox"/> Always	<input type="checkbox"/> Sometimes	<input type="checkbox"/> Configurable
Reports Binary Input Change Events when no specific variation requested:	<input type="checkbox"/> Never <input type="checkbox"/> Only time-tagged <input type="checkbox"/> Only non-time-tagged <input checked="" type="checkbox"/> <b>Configurable to send time-tagged or non-time-tagged (default is time-tagged)</b>			
Reports time-tagged Binary Input Change Events when no specific variation requested:	<input type="checkbox"/> Never <input type="checkbox"/> Binary Input Change With Time <input type="checkbox"/> Binary Input Change With Relative Time <input checked="" type="checkbox"/> <b>Configurable</b>			
Sends Unsolicited Responses:	<input type="checkbox"/> Never <input checked="" type="checkbox"/> <b>Configurable</b> <input type="checkbox"/> Only certain objects <input type="checkbox"/> Sometimes (attach explanation) <input checked="" type="checkbox"/> <b>ENABLE/DISABLE UNSOLICITED Function codes supported</b>			
Sends Static Data in Unsolicited Responses:	<input checked="" type="checkbox"/> <b>Never</b> <input type="checkbox"/> When Device Restarts <input type="checkbox"/> When Status Flags Change  No other options are permitted.			
Default Counter Object/Variation:	<input checked="" type="checkbox"/> <b>No Counters Reported</b> <input type="checkbox"/> Configurable (attach explanation) <input type="checkbox"/> Default Object Default Variation <input type="checkbox"/> Point-by-point list attached			
Counters Roll Over at:	<input checked="" type="checkbox"/> <b>No Counters Reported</b> <input type="checkbox"/> Configurable (attach explanation) <input type="checkbox"/> 16 Bits <input type="checkbox"/> 32 Bits <input type="checkbox"/> Other Value: <input type="checkbox"/> Point- by-point list attached			
Sends Multi-Fragment Responses: <input checked="" type="checkbox"/> <b>Yes</b> <input type="checkbox"/> No				

### **Implementation Table**

The basic DNP objects supported by the BE1-FLEX are defined in Table 3-2.

Table 3-2. Supported DNP Objects

<p><b>Binary Input Points</b></p> <p><i>Static Object Number: 1</i>  <i>Change Event Object Number: 2</i>  <i>Request Function Codes Supported: 1 (read), 22 (assign class)</i>  <i>Static Variation Reported When Variation 0 Requested: 1 (binary input without status), Configurable</i>  <i>Change Event Variation Reported When Variation 0 Requested: 2 (binary input change with time), Configurable</i></p>
<p><b>Binary Output Status Points</b></p> <p><i>Object Number: 10</i>  <i>Variations Supported: 1, 2</i>  <i>Request Function Codes Supported: 1 (read)</i>  <i>Default Variation Reported When Variation 0 Requested: 2 (binary output status)</i></p>
<p><b>Control Relay Output Blocks</b></p> <p><i>Object Number: 12</i>  <i>Variations Supported: 1</i>  <i>Request Function Codes Supported: 3 (select), 4 (operate), 5 (direct operate), 6 (direct operate, no ack)</i></p>
<p><b>Analog Inputs</b></p> <p><i>Static Object Number: 30</i>  <i>Change Event Object Number: 32</i>  <i>Request Function Codes Supported: 1 (read), 22 (assign class)</i>  <i>Static Variation Reported When Variation 0 Requested: User programmable—default “variation 0” is 3 (32-bit analog input without flag).</i>  <i>Change Event Variation Reported When Variation 0 Requested: User programmable—default “variation” is 1 (32-bit analog change event without time).</i>  <i>Analog Input Deadband Object: 34</i></p>
<p><b>Analog Output Status Points</b></p> <p><i>Object Number: 40</i>  <i>Variations Supported: 1, 2, 3</i>  <i>Request Function Codes Supported: 1 (read)</i>  <i>Default Variation Reported When Variation 0 Requested: 2 (16-bit analog output status)</i></p>
<p><b>Analog Output Blocks</b></p> <p><i>Object Number: 41</i>  <i>Variations Supported: 1, 2, 3</i>  <i>Request Function Codes Supported: 3 (select), 4 (operate), 5 (direct operate), 6 (direct operate), no ack</i></p>
<p><b>8-Bit Unsigned Integer (DNP Octet String)</b></p> <p><i>Object Number: 110</i>  <i>Variations Supported: 1</i>  <i>Request Function Codes Supported: 1 (read)</i></p>

Table 3-3 identifies the object variations, function codes, and qualifiers supported by the BE1-FLEX in both request messages and response messages.

For static (non-change event) objects, requests sent with qualifiers 00, 01, 06, 07, or 08 will be responded to with qualifiers 00 or 01. Static object requests sent with qualifiers 17 or 28 will be responded to with qualifiers 17 or 28.

Qualifiers 17 and 28 are always responded to for change-event objects.

**Table 3-3. BE1-FLEX DNP Implementation Table**

OBJECT			REQUEST (BE1-FLEX will parse)		RESPONSE (BE1-FLEX will respond with)	
Obj.	Var.	Description	Function Codes (dec)	Qualifier Codes (hex)	Function Codes (hex)	Qualifier Codes (hex)
1	0	Binary Inputs – (Variation 0 is used to request default variation)	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)		
1	1 (default, see note 1)	Single-Bit Binary Input	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81 (response)	00,01 (start-stop) 17,28 (index)
1	2	Binary Input with Status	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81 (response)	00,01 (start-stop) 17,28 (index)
2	0	Binary Input Change (Variation 0 is used to request default variation)	1 (read)	06 (no range) 07,08 (limited qty)		
2	1	Binary Input Change without time	1 (read)	06 (no range) 07,08 (limited qty)	81 (response)	17,28 (index)
2	2 (default, see note 1)	Binary Input Change with time	1 (read)	06 (no range) 07,08 (limited qty)	81 (response)	17,28 (index)
10	0	Binary Output – (Variation 0 is used to request default variation)	1 (read)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)		
10	2 (default)	Binary Output Status	1 (read)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
12	1	Control Relay Output Block	3 (select) 4 (operate) 5 (direct op) 6 (dir op Noack)	00,01 (start-stop) 07,08 (limited qty) 17,28 (index)	81	echo of request
30	0	Analog Input (Variation 0 is used to request default variation)	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81 (response)	00,01 (start-stop) 17,28 (index)
30	1	32-Bit Analog Input With Flag	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)

OBJECT			REQUEST (BE1-FLEX will parse)		RESPONSE (BE1-FLEX will respond with)	
Obj.	Var.	Description	Function Codes (dec)	Qualifier Codes (hex)	Function Codes (hex)	Qualifier Codes (hex)
30	2	16-Bit Analog Input With Flag	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
30	3 (default, see notes 1, 2)	32-Bit Analog Input Without Flag	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
30	4	16-Bit Analog Input Without Flag	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
30	5	Short float analog input	1 (read) 22 (assign class)	00,01 (start- stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
32	0	Analog Change Event (Variation 0 is used to request default variation)				
32	1 (default, see notes 1, 2)	32-Bit Analog Input without time	1 (read)	06 (no range) 07,08 (limited qty)	81	17,28 (index)
32	2	16-Bit Analog Input without time	1 (read)	06 (no range) 07,08 (limited qty)	81	17,28 (index)
32	3	32-Bit Analog Input with time	1 (read)	06 (no range) 07,08 (limited qty)	81	17,28 (index)
32	4	16-Bit Analog Input with time	1 (read)	06 (no range) 07,08 (limited qty)	81	17,28 (index)
32	5	Short float analog change event without time	1 (read)	06 (no range) 07,08 (limited qty)	81	17,28 (index)
32	7	Short float analog change event with time	1 (read)	06 (no range) 07,08 (limited qty)	81	17,28 (index)
34	0	(Variation 0 is used to request default variation)	1 (read) 2 (write)			00,01 (start-stop) 17,28 (index)
34	1 (see note 5)	16-Bit Analog Input Deadband	1 (read) 2 (write)	For read: 00,01,06,07,08,17,28 For write: 00,01,07,08,17,28	81	00,01 (start-stop) 17,28 (index)
34	2 (default, see note 6)	32-Bit Analog Input Deadband	1 (read)	For read: 00,01,06,07,08,17,28 For write: 00,01,07,08,17,28	81	00,01 (start-stop) 17,28 (index)

OBJECT			REQUEST (BE1-FLEX will parse)		RESPONSE (BE1-FLEX will respond with)	
Obj.	Var.	Description	Function Codes (dec)	Qualifier Codes (hex)	Function Codes (hex)	Qualifier Codes (hex)
40	0	Analog Output Status – (Variation 0 is used to request default variation)	1	00,01 (start-stop) 06 (no range) 07,08 (limited qty) 17,28 (index)		
40	1	32-bit Analog Output Status	1 (read)	00,01 (start-stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
40	2 (default)	16-bit Analog Output Status	1 (read)	00,01 (start-stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
40	3	Short float Analog Output Status	1 (read)	00,01 (start-stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
41	1	32-bit Analog Output Block	2 (select) 3 (operate) 4 (direct op) 6 (dir op noack)	00,01 (start-stop) 07,08 (limited qty) 17,28 (index)	81	echo of request
41	2	16-bit Analog Output Block	3 (select) 4 (operate) 5 (direct op) 6 (dir op noack)	00,01 (start-stop) 07,08 (limited qty) 17,28 (index)	81	echo of request
41	3	Short Float Analog Output Block	3 (select) 4 (operate) 5 (direct op) 6 (dir op noack)	00,01 (start-stop) 07,08 (limited qty) 17,28 (index)	81	echo of request
50	1	Time and Date	1 (read) 2 (write)	00,01 (start-stop) 06 (no range or all) 07 (limited qty=1) 08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
60	1	Class 0 Data (Note 4)	1 (read)	06 (no range or all)	81	
60	2	Class 1 Data	1 (read) 20 (enable unsol) 21 (disable unsol) 22 (assign class)	06 (no range or all) 07,08 (limited quantity)	81	
60	3	Class 2 Data	1 (read) 20 (enable unsol) 21 (disable unsol) 22 (assign class)	06 (no range or all) 07,08 (limited quantity)	81	

OBJECT			REQUEST (BE1-FLEX will parse)		RESPONSE (BE1-FLEX will respond with)	
Obj.	Var.	Description	Function Codes (dec)	Qualifier Codes (hex)	Function Codes (hex)	Qualifier Codes (hex)
60	4	Class 3 Data	1 (read) 20 (enable unsol) 21 (disable unsol) 22 (assign class)	06 (no range or all) 07,08 (limited quantity)	81	
80	1	Internal Indications	2 (write)	00 (start-stop) (index=7)		
110	(see note 7)	Octet String	1 (read)	00,01 (start-stop) 06 (no range) 07,08 (limited qty) 17,28 (index)	81	00,01 (start-stop) 17,28 (index)
		No Object(function code only) (See Note 3)	13 (cold restart)			
		No Object(function code only) (See Note 3)	14 (warm restart)			
		No Object (function code only)	23 (delay measurement)			

### Table 3-3 Notes

1. A default variation refers to the variation responded to when variation 0 is requested and/or in class 0, 1, 2, or 3 scans.
2. This is a default “default variation”. Objects 1, 2, 30, 32, and 40 have configurable default variation.
3. A cold restart is implemented as a warm restart – the DNP process is restarted, not BE1-FLEX.
4. By default, Class 0 includes all binary and analog input points in the DNP Map.
5. Maximum number of demands that can be written in on request for Variation 2 (32 bit) is 60 demands. Use multiple requests to write all demands.
6. Maximum number of octet strings that can be read is dependent upon Application Layer Response Fragment Size.

## DNP Configuration Settings

### Navigation Path: Communications, DNP, DNP Settings

The BE1-FLEX has two sets of DNP data change buffers and connection information. One set supports RS-485 communication and the other supports Ethernet. Any Ethernet port can support DNP as long as the device has the applicable protocol package and another Ethernet port is not already holding the DNP connection.

DNP settings are configurable through *BESTCOMSP/us*.

Dead bands for analog inputs are configured in the DNP map or via DNP object 34.

Class assignments for events are configured in the DNP map and/or via the DNP assign class function code 22.

Figure 3-1 illustrates the *BESTCOMSP/us* configuration settings for DNP communication.

### DNP Settings

#### Miscellaneous Settings

<h5>Ethernet</h5> <p>Type of End Point  <input type="text" value="UDP Datagram"/></p> <p>Local Port Number  <input type="text" value="20,000"/></p> <p>Client IP Address  <input type="text" value="0.0.0.0"/></p>	<h5>UDP End Point</h5> <p>UDP End Port  <input type="text" value="20,000"/></p> <p>Datagram Association Timeout (ms)  <input type="text" value="300,000"/></p> <p>Destination UDP Port for Other Responses  <input type="text" value="0"/></p> <p>0) Use Source Port Number</p>
<h5>Application Layer</h5> <p>Response Fragment Size  <input type="text" value="2,048"/></p> <p>Application Confirmation Timeout (ms)  <input type="text" value="5,000"/></p>	<h5>Link Layer</h5> <p>Link Layer Device Address  <input type="text" value="1"/></p> <p>Link Layer Data Confirmation  <input type="text" value="Never"/></p>
<h5>Listening End Point</h5> <p>TCP Keep Alive Timer (ms)  <input type="text" value="300,000"/></p>	

#### Unsolicited Response Support

<p>Unsolicited Support  <input type="text" value="Disabled"/></p> <p>Master Data Link Address  <input type="text" value="5"/></p> <p>Unsolicited Confirmation Timeout (ms)  <input type="text" value="5,000"/></p> <p>Number of Retries  <input type="text" value="2"/></p> <p>Off-Line Interval (ms)  <input type="text" value="12,000"/></p>	<h5>Trigger Conditions</h5> <p>Number of Class 1 Events  <input type="text" value="1"/></p> <p>Number of Class 2 Events  <input type="text" value="1"/></p> <p>Number of Class 3 Events  <input type="text" value="1"/></p>
--	---

#### Default Variations

<h5>Single-Bit Binary Inputs</h5> <p>Default Variation for Binary Input (object 1)  <input type="text" value="Binary Input"/></p> <p>Default Variation for Binary Input Change (object 2)  <input type="text" value="Without Time"/></p>
<h5>Analog Inputs</h5> <p>Default Variation for Analog Input (object 30)  <input type="text" value="32-bit with Flag"/></p> <p>Default Variation for Analog Change Event (object 32)  <input type="text" value="32-bit without Time"/></p>
<h5>Analog Output Status</h5> <p>Default Variation for Analog Output Status (object 40)  <input type="text" value="32-bit"/></p>

Figure 3-1. DNP Communication Settings

## DNP Configuration Settings Available Through BESTCOMSPlus

DNP settings that can be configured through BESTCOMSPlus are summarized as follows.

### DNP Over Ethernet Settings

- Type of End Point (TCP Listening or UDP Datagram)
- Local port number
- Client IP address

### TCP Listening End Point Setting

TCP Keep Alive Timer (TCP connection timeout) in milliseconds

### UDP End Point Settings

- Destination UDP Port for initial unsolicited null response
- Destination UDP Port for Other Responses
- Datagram Association timeout in milliseconds

### Link Layer Settings

- Data Link Layer Device address (device DNP address)
- Data Link Layer Data Confirmation

### Application Layer Settings

- Application Response Fragment Size
- Application Confirmation Timeout in milliseconds

### Unsolicited Response Support Settings

- Enable/Disable
- Client Data Link Address (DNP unsolicited response destination address)
- Unsolicited Response Confirmation Timeout in milliseconds
- Number of Unsolicited Retries
- Unsolicited Off-Line Interval in milliseconds
- Unsolicited response trigger conditions (number of class 1, 2, or 3 events)

### Objects Default Variations Settings

- Default Variation for Binary Input (object 1)
- Default Variation for Binary Input Change (object 2)
- Default Variation for Analog Input (object 30)
- Default Variation for Analog Input Change (object 32)
- Analog Output Status (object 40)

## DNP Map

### Navigation Path: [Communications](#), [DNP](#), [DNP Map](#)

The DNP map of a BE1-FLEX is fully customizable in BESTCOMSPlus to meet application requirements. See Figure 3-3. The address of a mapped item can be edited and unneeded variables can be deleted as desired. The DNP Map is easily exported from or imported to BESTCOMSPlus for use with multiple BE1-FLEX systems. A DNP map also can be exported directly to a CSV files for documentation purposes.

### Map Creation

The BE1-FLEX defaults to an empty DNP map. Clicking the Generate button will display an Auto Populate Items window (Figure 3-2) showing the available object categories for quick setup. Selecting the desired object categories will populate the map with all items related to the selected categories.

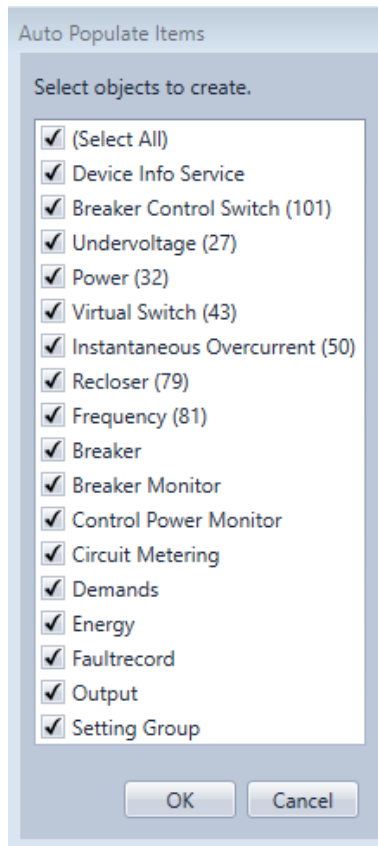


Figure 3-2. Auto Populate Items Window

Individual items can be added to the map by using the Function, Instance, and Value controls of the Available DNP Mapping items section. An item and its attributes can be selected and then placed in the map by using the Add button.

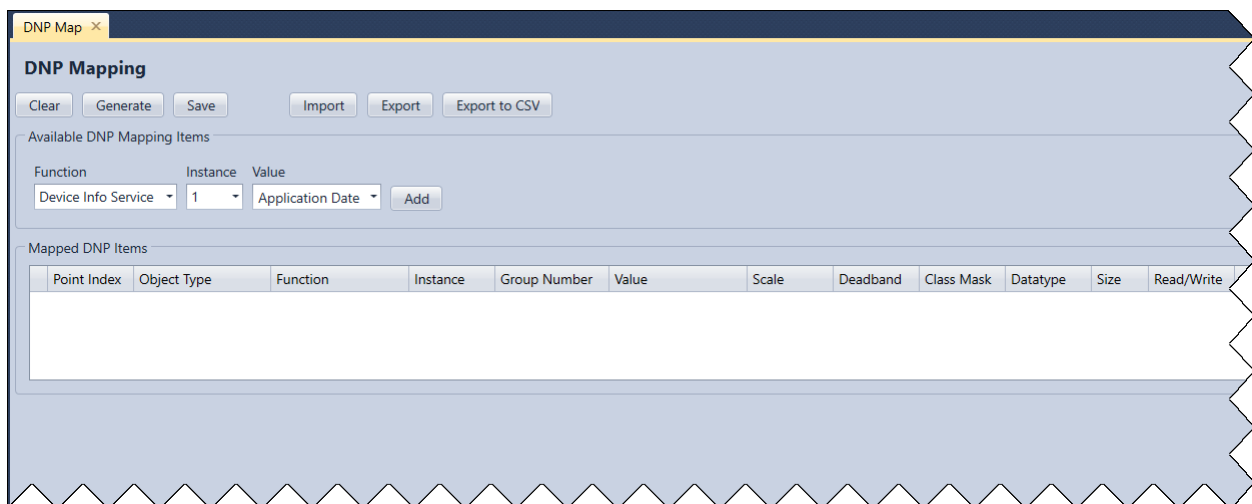


Figure 3-3. DNP Mapping Window

### Map Example

A sample map, for illustration only, is shown in Figure 3-4. This illustration shows all of the labels and parameters associated with the selected items. A filter function is available for each column and is useful for displaying only the data of interest in the map.

Point Index	Object Type	Function	Instance	Group Number	Value	Scale	Deadband	Class Mask	Datatype	Size	Read/Write	Delete
0	Binary Input	Breaker Control Switch (101)	1		Blocked			Class 0	BoolType	4	ReadOnly	Delete
1	Binary Input	Breaker Control Switch (101)	1		TSC			Class 0	BoolType	4	ReadOnly	Delete
2	Binary Input	Undervoltage (27)	1		Pickup			Class 0	BoolType	4	ReadOnly	Delete
3	Binary Input	Undervoltage (27)	1		Trip			Class 0	BoolType	4	ReadOnly	Delete
0	Binary Output	Breaker Control Switch (101)	1		Block Tag				IntType	4	ReadWrite	Delete
1	Binary Output	Breaker Control Switch (101)	1		Direct Operate				IntType	4	ReadWrite	Delete
2	Binary Output	Breaker Control Switch (101)	1		Block Untag				IntType	4	ReadWrite	Delete
0	Analog Input	Circuit Metering	1		Freq	1	0	Class 0	FloatType	4	ReadOnly	Delete
1	Analog Input	Circuit Metering	1		Primary V1 Mag	1	0	Class 0	FloatType	4	ReadOnly	Delete
0	Analog Output	Output	1		Override Enable				IntType	4	ReadWrite	Delete
1	Analog Output	Output	2		Override Enable				IntType	4	ReadWrite	Delete
0	DNP Strings	Device Info Service	1		Device ID				StringType	64	ReadOnly	Delete
1	DNP Strings	Device Info Service	1		Model Number				StringType	64	ReadOnly	Delete
2	DNP Strings	Device Info Service	1		Serial Number				StringType	9	ReadOnly	Delete

Figure 3-4. DNP Map Example

## Map Editing

All items in a map can be removed with the Clear button. Individual items can be removed with the Delete button located to the right of the item.

Class Mask, Analog Input Scaling, Deadband and the Delete buttons are the active edit fields in the map.

A Binary Input or Analog Input Object Type can be altered to select which event class or classes generate unsolicited responses.

Additionally, the map sequence can be edited by clicking and dragging any item within the map. All DNP points for each object start at zero. All points are automatically renumbered to be contiguous.

### Scaling of (Default) Analog Input Points

When an analog point value exceeds the range of the currently active object variation, the reported value is the maximum amount for that variation and object variation that includes the setting of an Over Range flag.

The Over Range status can be avoided if the value is scaled with the appropriate scaling factor. The default value for any point is 1.000 and the available range is from 0.001 to 1,000,000,000.000.

Each analog input point and analog output point has an adjustable scaling factor. Scaling is set from the Mapped DNP Items list.

## Map Saving, Exporting, and Importing

Mapped items can be saved as part of the BE1-FLEX settings file by clicking the Save button.

The Export button is used to save a DNP map for use with other BE1-FLEX systems. Clicking the Export button displays a Save DNP Mapping dialog window where the file name and location can be assigned. DNP map export files are appended with a "bstd" extension. To retrieve a previously exported DNP map, click the Import button. This displays an Open DNP Mapping dialog window that provides navigation to and selection of the desired map file with a "bstd" extension.

The Export to CSV button exports a DNP map as a comma-separated values file. Map items are saved in a tabular format that can be viewed in most spreadsheet applications.

## 4 • Synchrophasor Configuration

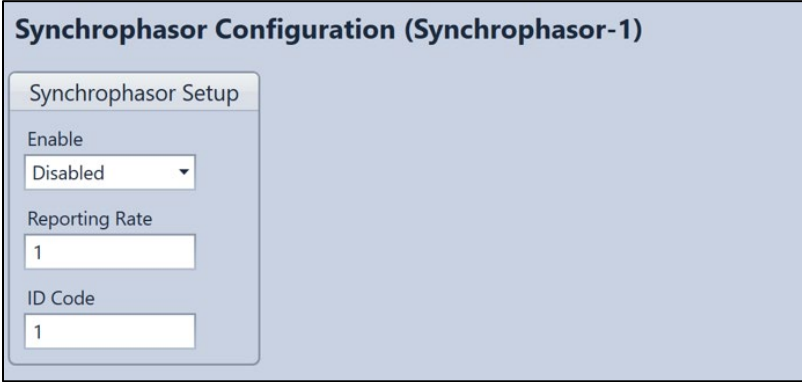
Synchronized phasors (synchrophasors) provide highly-accurate, time-synchronized magnitude and angle measurements across an Ethernet communication network. The ability to see individual resource information from a wide area control point can provide improvements to system stability and resource optimization.

### **Synchrophasor Configuration Settings**

---

**Navigation Path:** Configuration, Circuit Summary, Circuit, Synchrophasor Configuration

Enable synchrophasor broadcasting for a circuit by navigating to the circuit in the BESTCOMSPi<sup>us</sup>® Configuration, Circuit Summary branch and selecting Synchrophasor Configuration. An example synchrophasor settings screen is shown in Figure 4-1.



The screenshot displays a web-based configuration interface titled "Synchrophasor Configuration (Synchrophasor-1)". It features a "Synchrophasor Setup" panel with the following settings:

- Enable:** A dropdown menu currently set to "Disabled".
- Reporting Rate:** A text input field containing the value "1".
- ID Code:** A text input field containing the value "1".

**Figure 4-1. Synchrophasor Configuration Settings**

Settings are provided to enable the circuit synchrophasor, set the reporting frequency, and assign an identification code for the BE1-FLEX.

### **Data Reported**

---

Synchrophasor data reported by the BE1-FLEX includes sensed voltage (VA, VB, VC, VAB, VBC, and VCA) and sensed current (IA, IB, and IC).

### **Data Format**

---

BE1-FLEX synchrophasor data is reported in a format consistent with IEEE C37.118.2 – *IEEE Standard for Synchrophasor Measurements for Power Systems*.

### **Reporting**

BE1-FLEX synchrophasors are reported to UDP (user datagram protocol) multicast address 239.253.10.1 on port 4713.

A PDC (phasor data concentrator) identifies a PMU (phase measurement unit) by its identification code.

### **Control Commands**

Control commands are sent via UDP unicast sideband communication to the IP address of the BE1-FLEX on port 4712.



# 5 • IEC 61850 Communication

## Introduction

BE1-FLEX systems equipped with Protocol Package 02 have IEC 61850 capability. Functionality conformance statements are available in this document.

### Caution

This product contains one or more nonvolatile memory devices. Nonvolatile memory is used to store information that needs to be preserved when the product is power-cycled or otherwise restarted. Established nonvolatile memory technologies have a physical limit on the number of times they can be erased and written. In this product, the limit is 20 TB writes. During product application, consideration should be given to communications, logic, and other factors that may cause frequent/repeated writes of settings or other information that is retained by the product. Applications that result in such frequent/repeated writes may reduce the useable product life and result in loss of information and/or product inoperability.

This section describes the Basler Electric IEC 61850 protocol implementation in the BE1-FLEX. A BE1-FLEX is classified as an Intelligent Electronic Device (IED) that is capable of sending and receiving IEC 61850 messages simultaneously.

The IEC 61850 protocol is an option available when ordering a BE1-FLEX. An Ethernet connection is required for IEC 61850 operation.

IEC 61850 is a communications infrastructure that allows seamless integration of IEDs into higher level devices. IEDs from different vendors can be integrated together in this vendor-independent infrastructure. A typical station bus is illustrated in Figure 5-1.

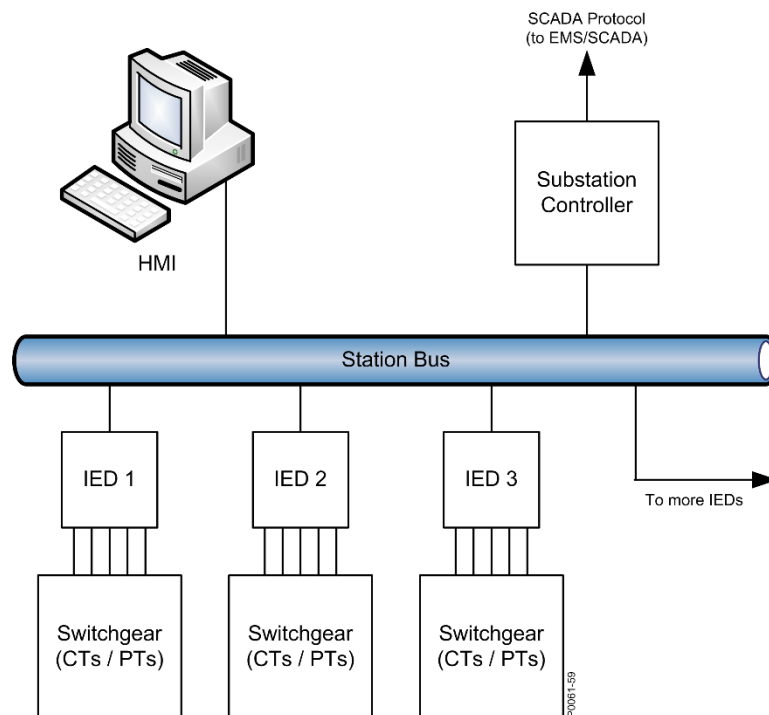


Figure 5-1. Typical Station Bus

Protection, control, and metering for the substation are defined in an SCL (Substation Configuration Language) file. The BE1-FLEX comes with an SCL compatible ICD (IED Capability Design) file that is uploaded to the BE1-FLEX as a CID (Configured ICD) file. The substation design developer uses the CID file to create the portion of the SCD (Substation Configuration Description) file for the overall substation design.

The BE1-FLEX is capable of transmitting the following items on an IEC 61850 network:

- Configuration settings
- Metering
- Status
- Alarms
- Targets
- Breaker operations
- Direct control
- Fault records
- Load profile
- Oscillographic records
- Sequence of events reports
- Unsolicited reports
- Peer-to-peer control GOOSE
- Device discovery

## ***IEC 61850 Configuration***

---

IEC 61850 communication relies on several configuration file types to be discussed throughout this manual. Substation Configuration Language is the XML based language used within Substation Configuration Description (\*.scl) and Configured IED Description (\*.cid) files. The CID file is specific to each IED in a system. The SCD file contains all IEDs in the network, communication configuration, and substation description. The IEC 61850 standard defines the parts for communicating information between IEDs in a substation.

Since the BE1-FLEX can be configured in countless configurations, a fixed map does not exist. Below are the components utilized to configure a BE1-FLEX for IEC61850 communications.

### **Substation Configuration Language (SCL) Files**

This section describes how to create, import, and export an SCL file.

#### *SCL File Import and Export*

SCL files are imported or exported on the IEC 61850 File screen (Figure 5-2) in BESTCOMSP<sup>Plus</sup>®.

[BESTCOMSP<sup>Plus</sup> Navigation Path: Settings Explorer, Communications, IEC 61850, IEC 61850 File](#)

Once the BE1-FLEX settings file has the desired configured instances, navigate to the IEC 61850 File screen and export the \*.cid file. The \*.cid file can be viewed with a text editor for edits such as changing the IP address.

If GOOSE messaging and advanced configuration are not required, rename the \*.cid file type to \*.scl. Then select Import SCD on the IEC 61850 File screen. Select the IED and Access Point desired and click OK. The SCD is now imported into the BE1-FLEX settings file.

For GOOSE mapping and additional configuration, open the exported \*.cid or \*.scl file in the Substation Configuration Tool of choice. Configure each IED as desired. Export the SCD file, then select Import SCD from BESTCOMSP<sup>Plus</sup>.

Once the \*.scl file is imported, GOOSE Inputs and Outputs will be available in BESTlogic™<sup>Plus</sup>. See the BESTlogic<sup>Plus</sup> section in the main BE1-FLEX instruction manual for further details on logic capabilities and configuration.

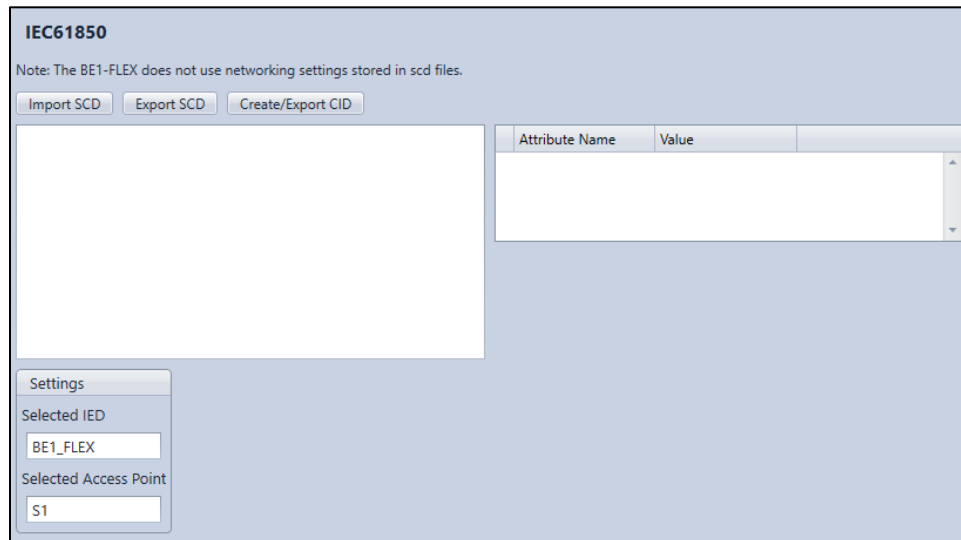


Figure 5-2. IEC 61850 File Screen

### GOOSE and Ethernet Settings

GOOSE and Ethernet settings are configured on the IEC 61850 Settings screen (Figure 5-3) in *BESTCOMSPPlus*.

**BESTCOMSPPlus Navigation Path:** Settings Explorer, Communications, IEC 61850, IEC 61850 Settings

Generic Input and Output logical nodes, CTL\_GGIO and IND\_GGIO, are created by adding Remote Indicator and Indicator instances from within the *BESTLogicPlus* Programmable Logic folder.

The IEC 61850 Settings screen defines which Ethernet port IEC 61850 will use and defines the scan rate and MMS password. Ethernet port 2 is used in conjunction with general Ethernet settings for HSR or PRP redundancy with Ethernet boards that support redundancy.

GOOSE Settings enable GOOSE blocks 1 through 4. Each block is disabled by default.

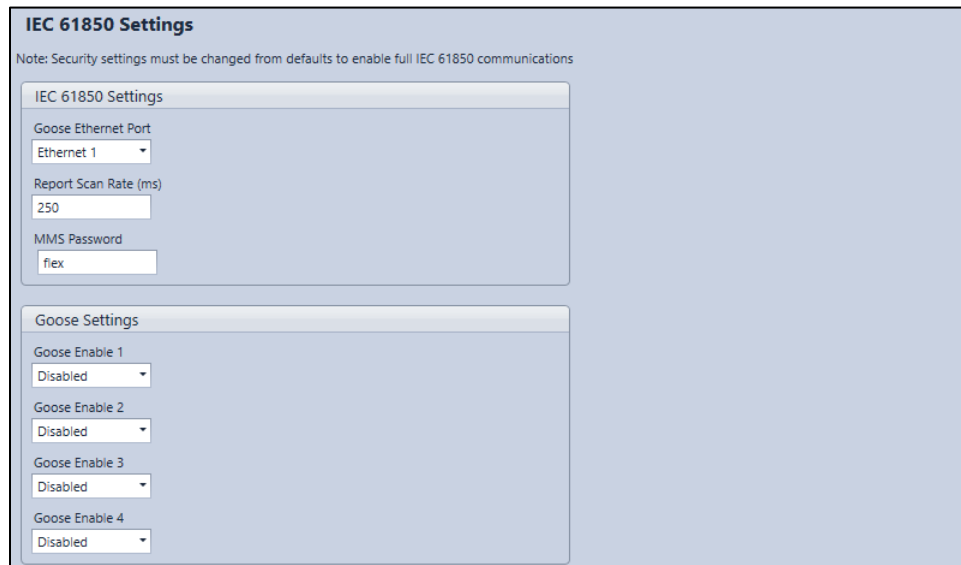


Figure 5-3. IEC 61850 Settings Screen

## ***IEC 61850 Engineering***

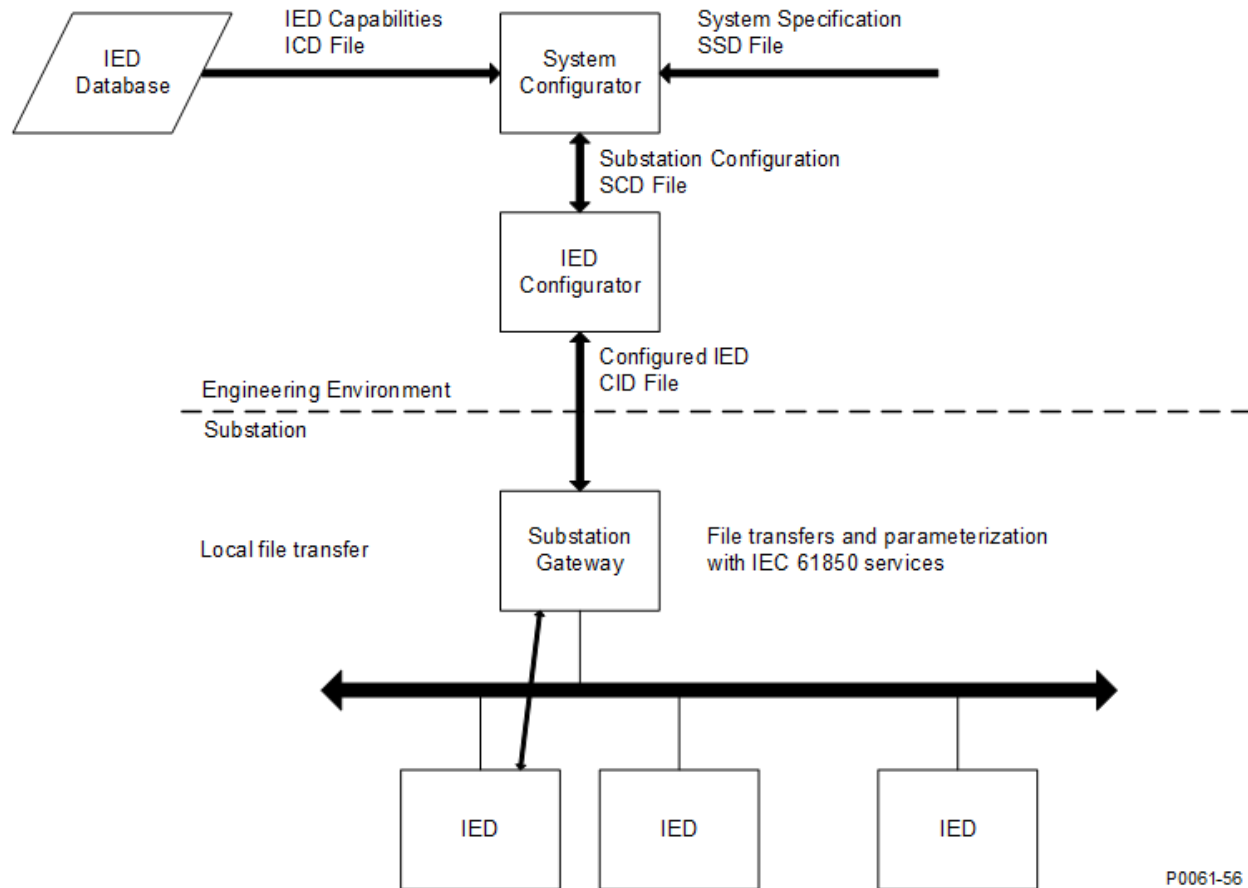
The engineering process and configuration of a substation is described in IEC 61850-Part 6:

*Configuration description language for communication in electrical substations related to IEDs.*

This section contains information about the standard that may be referenced when working with BESTCOMSPlus® software.

## Engineering Process in IEC 61850

Figure 5-4 illustrates the reference model for information flow in the configuration process.



**Figure 5-4. Reference Model for Information Flow in the Configuration Process**

The IEC 61850 standard defines two tools and four file types for Engineering as illustrated in Figure 5-4. The tools and file types are defined below.

### System Configurator

The System Configurator performs the following functions:

- Imports and exports configuration files
- Imports configuration files from several IEDs for system level engineering
- Generates a substation related configuration file
- Reads an SSD file as a base for starting system engineering or for comparisons

The complete station configuration is exported in the SCD file for further use by the various IED configuration tools.

### IED Configurator Tool

Configuration tools perform, but are not exclusively limited to, the following functions:

- Imports an SCD file and extracts the necessary information for each IED in the system

- Maps subscribed GOOSE messages between devices
- Configures published GOOSE messages
- Configures DataSets
- Configures Report Control Blocks (RCB)
- Creates a CID file and uploads it to the IED or exports it to a file
- Performs any additional engineering operations

#### ICD File (IED Capability Description File)

The capabilities of an IED are described in this file. The file contains logical node type definitions and may contain an optional substation section.

#### SSD File (System Specification Description File)

This file describes the single line diagram of the substation and the required logical nodes. It contains a substation description section and the data templates and logical node definitions.

#### SCD File (Substation Configuration Description File)

This file contains the data exchanged from the system configuration tool to the IED configuration tool. It contains all IEDs, a communication configuration section, and a substation description.

#### CID File (Configured IED Description File)

This file contains the data exchanged from the IED configuration tool to the IED.

### **SCL Object Model**

The SCL (Substation Configuration Language) object model consists of the following parts:

- The primary (power) system structure
- The communications system
- The application level communication
- Each IED
- Instantiable logical node (LN) type definitions
- The relations between instantiated logical nodes and their hosting IEDs on one side and the switchyard (function) parts on the other side

Figure 5-5 shows an instance of the SCL object model: a simple example of an SA (Substation Architecture) system used for a switchyard. The switchyard has a 110 kV voltage level E1. It is a double bus bar system with two line bays =E1Q1 and =E1Q3, and a bus coupler =E1Q2. The IEDs are already assigned to switchyard functionality (for example the bay controller –E1Q1SB1 as a product is assigned to bay =E1Q1, and its Logic Node CSWI1 controls the circuit breaker =E1Q1QA1 via the Logic Node XCBR1 on the IED –E1Q1QA1B1). Figure 5-5 uses a – (minus) sign to indicate the product-related designations. The functional name is not repeated. The station level communication sub network is named W1. There are three additional sub networks at process level (W2, W3, and W4).

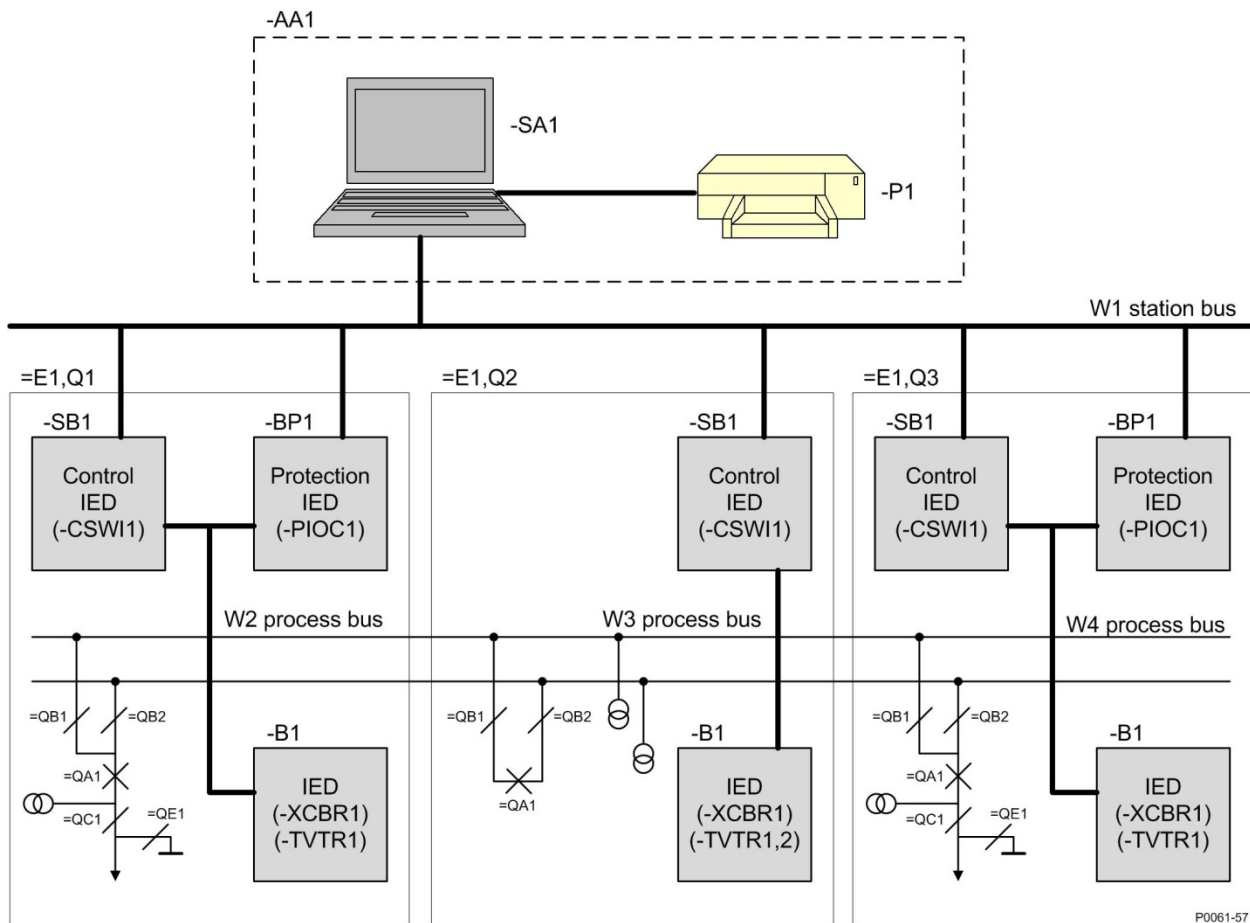


Figure 5-5. Configuration Example (from IEC 61850-6)

The IEC 61850 standard is based on the hierarchical addressing and information model in a station. It follows, in this way, the structure of the substation equipment independent of IED structure and organization. This information and addressing model is also visible in the IEC 61850 telegrams because the address is presented in MMS (Manufacturing Message Specification) as an ASCII string, so the address can be seen directly in a readable form.

The IEC 61850 standard defines the information and its exchange in a way that it is independent of a concrete implementation (i.e., it uses abstract models). The standard also uses the concept of virtualization. Virtualization provides a view of those aspects of a real device that are of interest for the information exchange with other devices. Only those details that are required to provide interoperability of devices are defined in the IEC 61850 standard.

The approach of the standard is to decompose the application functions into the smallest entities, which are used to exchange information. These entities are called logical nodes (for example, a virtual representation of a circuit breaker class, with the standardized class name XCBR). Several logical nodes build a logical device (for example, a representation of a Bay unit). A logical device is always implemented in one IED, therefore logical devices are not distributed.

Real devices on the right-hand side of Figure 5-6 are modeled as a virtual model in the middle of the figure. The logical nodes defined in the logical device (for example, Bay) correspond to well-known functions in the real devices. In this example, the logical node XCBR represents a specific circuit breaker of the bay to the right.

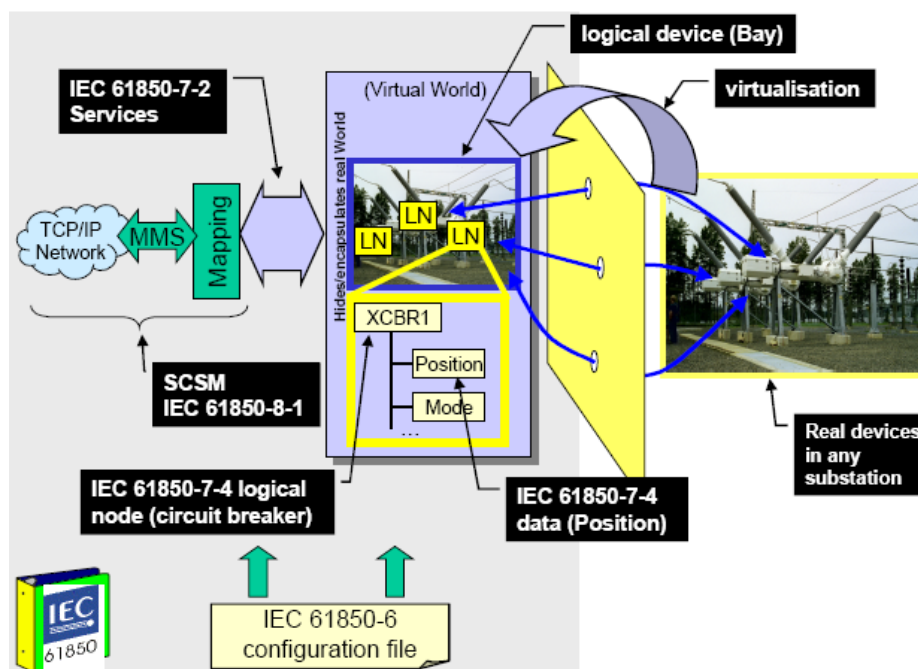


Figure 5-6. Modeling Approach Example

### Substation Configuration Description Language (SCL)

The SCL language is based on XML (Extensible Markup Language). Detailed knowledge about an XML file is not required, but the information within the file may be of interest. An SCL XML file is divided into the following five sections as specified in IEC 61850-6:

- Header
- Substation description
- Communication system description
- IED description
- Data Type Templates

The Header identifies an SCL configuration file and version, and specifies options for the mapping of names to signals. The Substation section describes the functional structure of a substation, and defines the primary devices and their electrical connections.

Default ICD files in configuration software are used to define the Header and Substation description sections. Logical nodes of the IED are logically linked to the substation section. A configuration tool is used for signal engineering and routing of the signals. BESTCOMSP<sup>lus</sup> must be used to define the substation and communication sections prior to using a configuration tool. Configuration tools are used to configure the DataSets and Control Blocks that are located in the IED section. The DataSets and Control Blocks (Report, GOOSE, and Setting) are logically defined as part of the logical nodes per IEC 61850-7-2. GOOSE engineering requires that the configuration tool has the correct configured Communication description section. The Data Type Templates section gives the correct content description of each logical node type to the clients. Logical node type definitions may vary between each IED and vendor.

Figure 5-7 shows the principle structure of the SCL XML file. The arrows show the link between the different sections given when an IED is integrated in the substation structure and/or in the communication structure. All available logical nodes of an IED are linked to the substation section.

A reference to the GoCB (GOOSE Control Block) is included in the communication section when the GoCB is configured. The graphical presentation of the XML file is a standard view of XML editors. It gives a better picture about the structure of the XML contents.

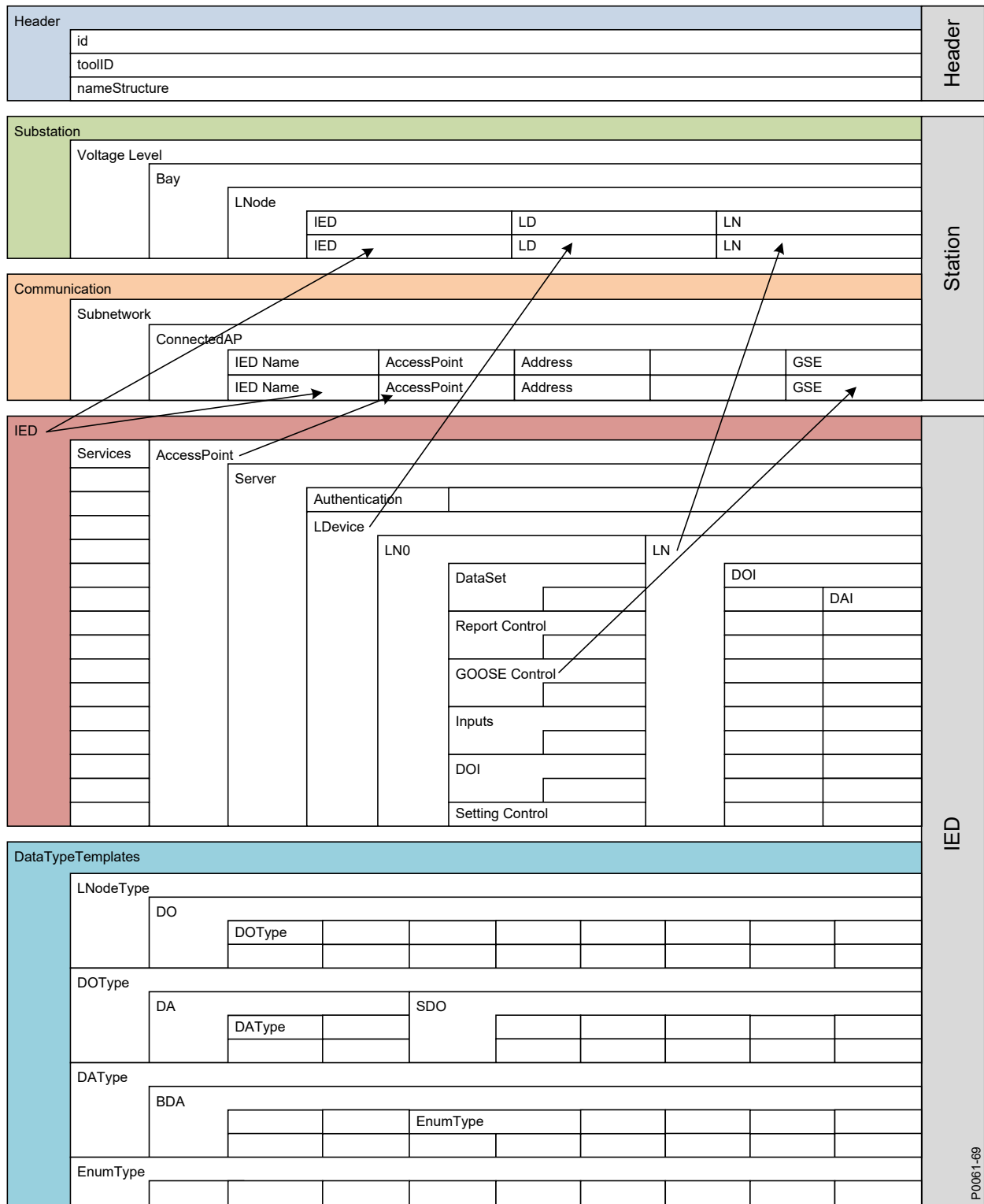
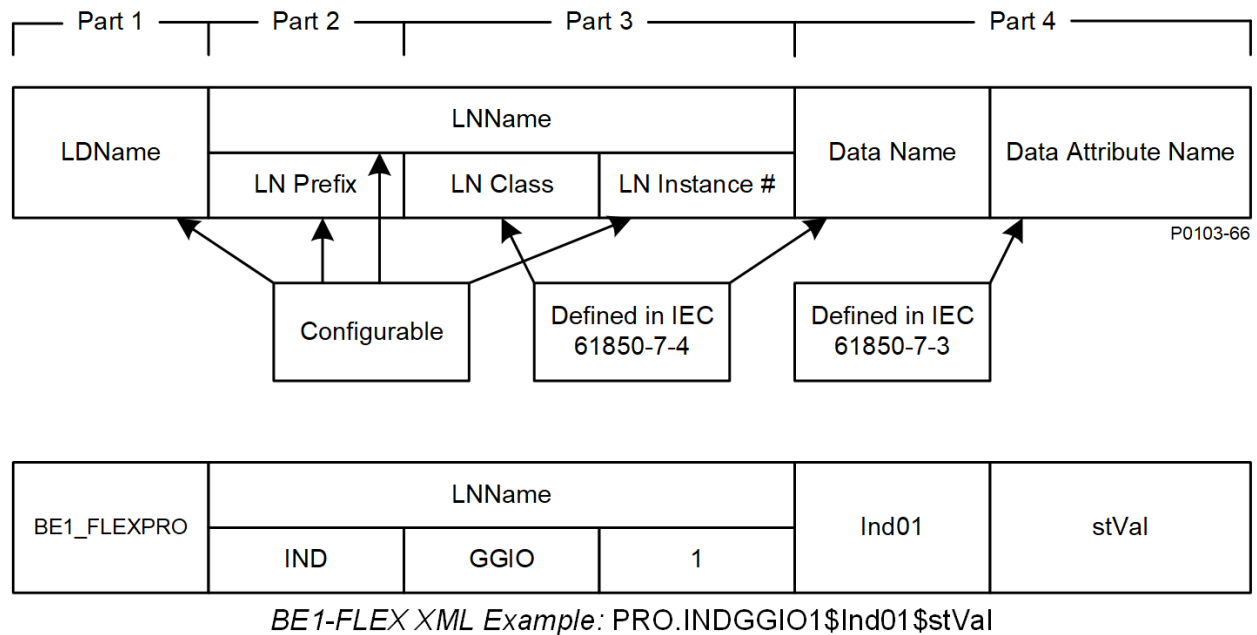


Figure 5-7. Principle Structure of the SCL XML File

Signal Identification

Elements of the signal identification as defined in IEC 61850-7-2 are illustrated in Figure 5-8.



**Figure 5-8. Elements of the Signal Identification as Defined in IEC 61850-7-2**

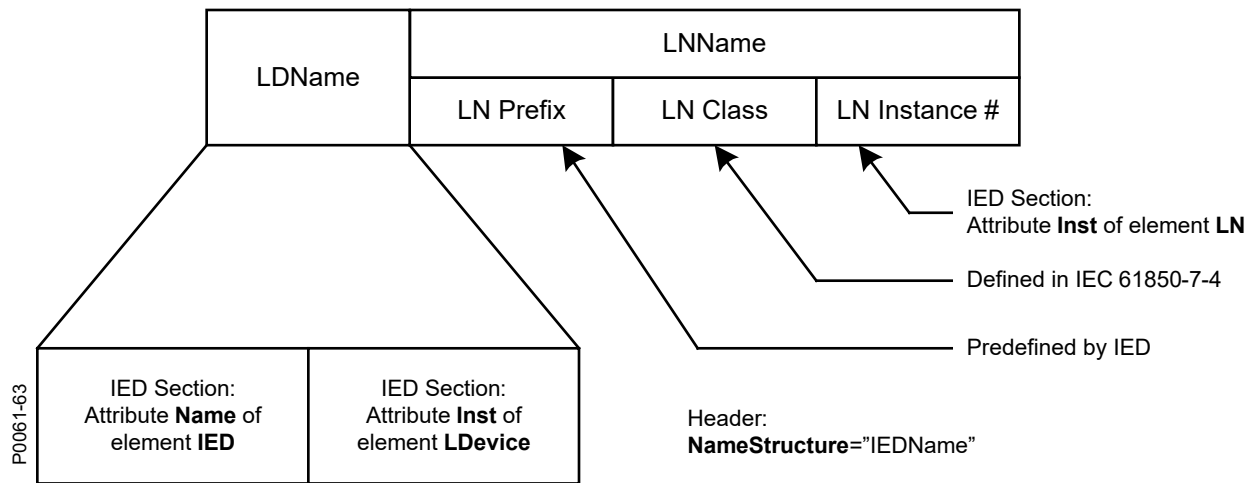
The signal designation consists of the following four parts:

- A user defined part identifying the Logical Device LD in the process (LDName).
- A function related part to distinguish several LNs of the same class within the same LD/IED (LN-Prefix).
- The standardized LN class name and the LN instance number, which distinguishes several LNs of the same class and prefix within the same LD/IED.
- A signal identification inside a LN consisting of data and attribute name as defined in IEC 61850-7-3 and IEC 61850-7-4.

#### IED Related Naming

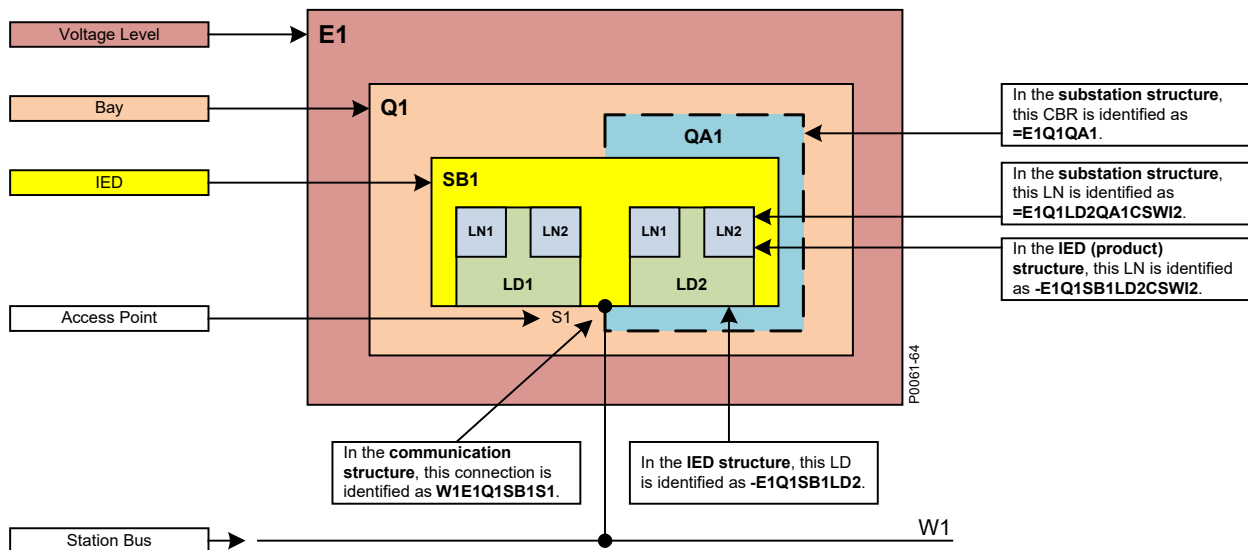
The LDName part is built out of the objects IED and LDevice. The LNName is the LN Prefix, the LN class and the LN instance #.

The LN prefix is used to identify several versions of a LN class. The link between the IED and the primary process (Substation, Voltage level, Bay) is given in the substation section of the station SCD file but not in the signal identification in the telegram. The IED name may be extended with a short form of the three missing levels. See Figure 5-9.



**Figure 5-9. Elements of the Signal Name Using Product Naming**

Figure 5-10 shows an example of an IED (SB1) with logical devices LD1 and LD2. Each logical device (LD1 and LD2) contain logical nodes LN1 and LN2, which control a circuit breaker QA1 of bay Q1 in voltage level E1.



**Figure 5-10. Names within Different Structures of the Object Model**

### Communication Network Section

The organization of the physical IEDs to the communication network is independent of the substation structure. The IEC 61850 standard defines the communication network with no relation to an existing media and protocol. The first mapping to an existing media and protocol is done in IEC 61850-8-1 with:

- Ethernet as medium
- MMS (Manufacturing Message Specification) protocol as defined in ISO 9506-1 and ISO 9506-2

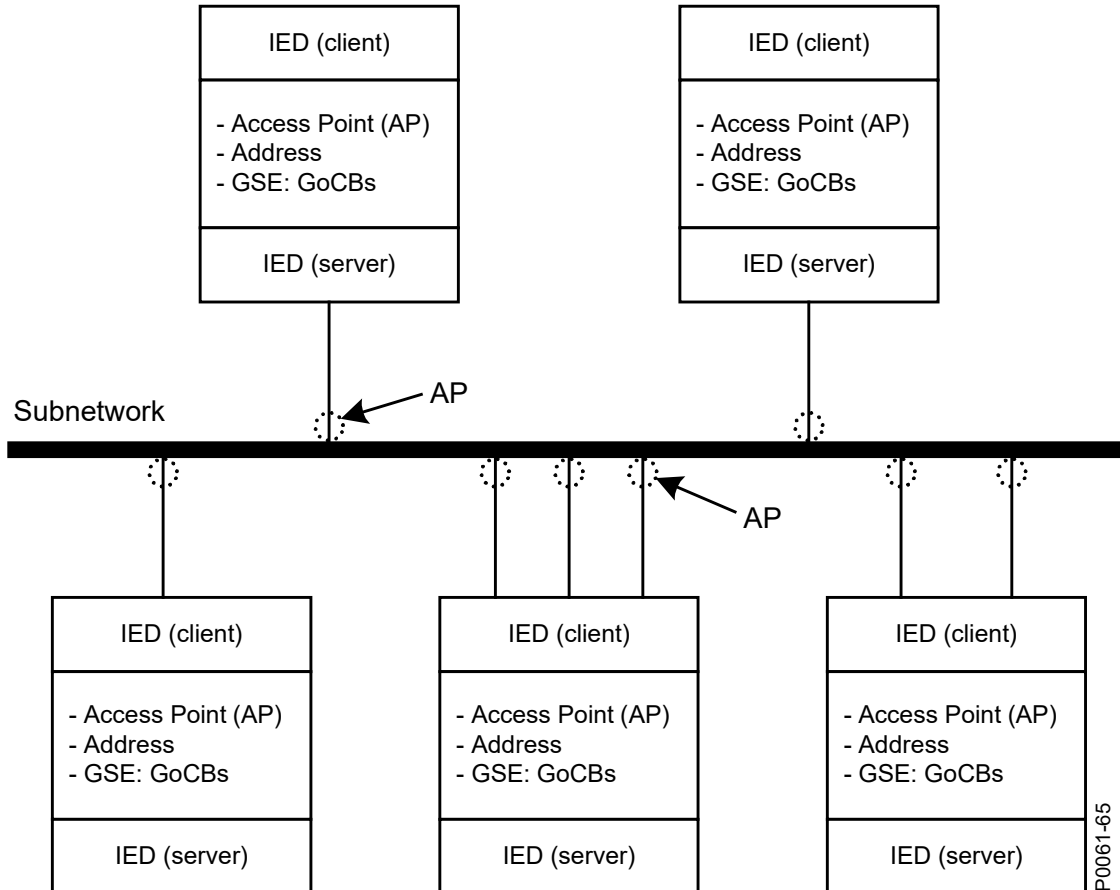
The IEC 61850 standard describes in part 7-2 the Abstract Communication Service Interface (ACSI) in a media and protocol independent form. Part 8-1 specifies the mapping of this ACSI to the existing MMS.

The communication section in the SCL file describes the minimum of what is needed to identify how information is routed between the IEDs in a project. This is:

- The used sub networks
- The IEDs connected to the different sub networks
- The access points per IED to the sub networks

- The address
- The IP address of the LAN network is an exception also part of the address elements
- Extended during signal engineering and routing, the link to the GoCB message in transmission direction

Figure 5-11 shows the IEC 61850-6: Communication Network.

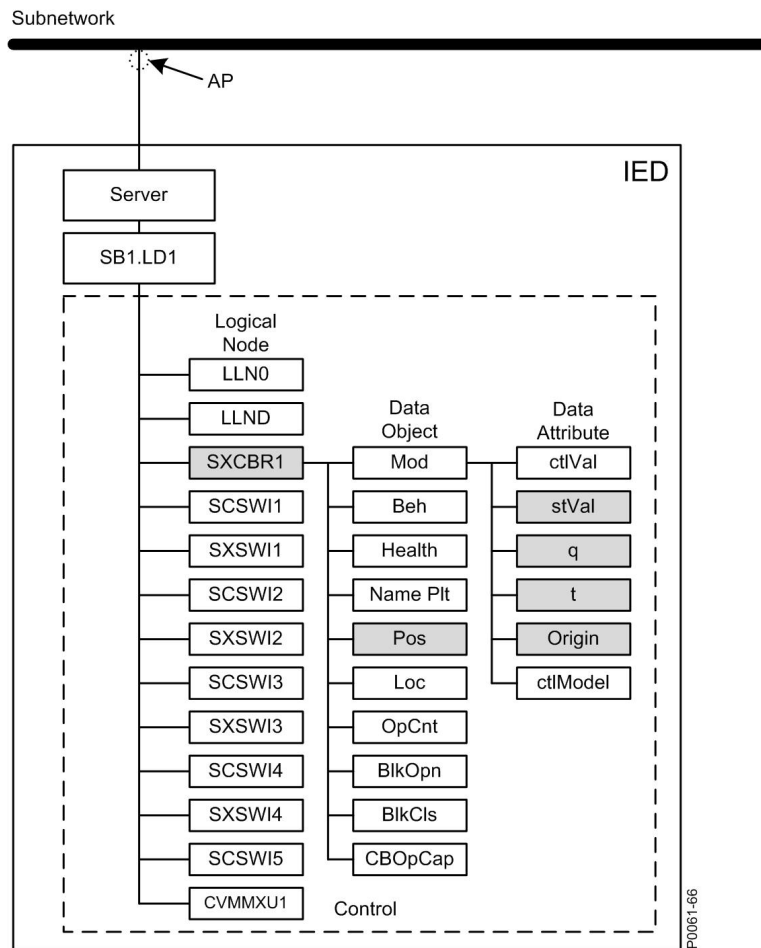


**Figure 5-11. IEC 61850-6: Communication Network**

### IED Section

The IED section describes the complete IED as it is needed for IEC 61850 communication. The Data Type Template part of an IED may be seen as part of the IED, even when separated in its own section. The IED's ICD file includes the description of the LNs, their Data Type Templates, and the used/supported services. The structure of the IED section follows the definitions made in the IEC 61850 standard.

Figure 5-12 illustrates organization of LDs, LNs, DOs, and DAs in an IED.



**Figure 5-12. Organization of LDs, LNs, DOs, and DAs in an IED**

An IED server represents the communication interface to the sub network (Ethernet).

- One or more Logical device(s) are connected to a server
- A set of Logical Nodes belong to a Logical Device
- The Logic Node LLN0 is a special Logic Node per Logic Device and contains the DataSets, GOOSE Control Blocks (GoCB), Report Control Blocks (RCB), and Setting Group Control Block (SGCB)
- The Logic Node LPHD is a special Logic Node per Logic Device and contains Data Objects (DO) which describe the status of the physical device (the IED)
- Each Logical Node represents a function and contains a number of Data Objects (DO)
- Each DO is represented by a number of Data Attributes (DA)

The data objects are representing information signals which may be routed to station level IEDs.

The signal engineering task is to select the requested signals (DOs) and link them to the client IEDs as receiver. The control services are not directly engineered. They are included in the data objects which handle both directions the command (control) and the response (monitoring). When routing the DO in monitoring direction the control is known by the clients. The organization of the IED from LD down to DAs can be viewed in the BEST61850 tool. This organization concept must be taken into consideration when DataSets are configured.

The number of data objects and data attributes per Logical Node is defined by the used LN type in this IED. The contents are taken from the Data Type Templates which belong to an IED type.

### Signal Engineering

Signal engineering consists of DataSets, Report Control Blocks (RCB), and Goose Control Blocks (GoCB).

#### DataSets

IEC 61850 has defined DataSets for signal transmission in Report Control Blocks. DataSets are also used for GOOSE messages. Figure 5-13 shows a DataSet where all position information of the apparatuses of a bay are put into one DataSet.

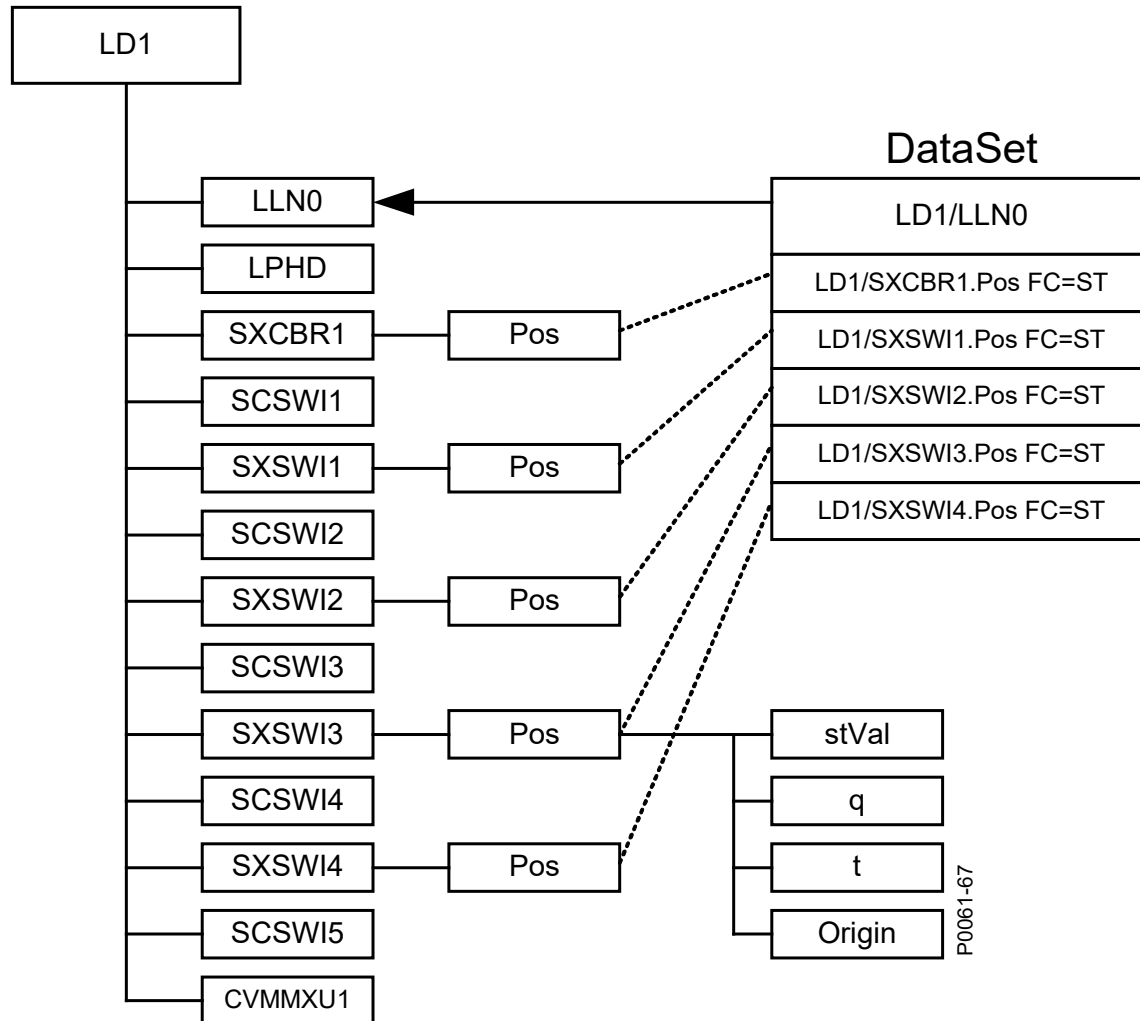


Figure 5-13. DataSet Example

General rules for DataSet configuration:

- All data objects or their data attributes can be selected for a DataSet.
- Only those data attributes of a data object are selected which have the same functional constraint (FC).
- Data objects with different FC can be selected for a DataSet. For example, DOs with FC=ST and DOs with FC=MX can be members in one DataSet.
- A single data attribute can be selected when it is specified with a trigger option. For example, the data attribute stVal of the data object Pos can be selected as a member of a DataSet, because it is specified with the trigger option data change detected (dchg).

The description of the DataSets and the list of data object members or FCDAs (Functionally Constrained Data Attributes) is included in the SCL file in the IED section in the Logical Device subsection. FCDAs are

also referred to as Data Tags. As specified in IEC 61850–7–2 clause 9, the DataSets are part of a Logical Node. They are included in the LLN0.

### Report Control Blocks (RCB)

The contents of an RCB are listed in IEC 61850–7–2 in clause 14, table 23. The RCB contains a list of attributes which handle and secure the communication between the client and the server.

- **Buffer Time** - This parameter describes how long the report should wait for other expected events before it sends the report to the client. When known, additional events are generated as a follow up. It is useful to wait for approximately 500 ms for additional events stored in the report. This feature reduces the number of telegrams transmitted in case of a burst of changes. But it increases the overall transaction time for events from IED input to presentation on HMI which is normally one second.
- **Trigger Options** - Data attributes have three trigger options (dchg, qchg, dupd). Within the RCB, two others can be defined integrity and general interrogation. The attribute Trigger Option is a multiple choice and allows masking of the supported trigger options in this RCB.
- **Integrity Period** - When period is selected in the trigger option attribute, it is needed to define an integrity period to force the transmission of all data listed in the DataSet. This is done by the attribute Integrity Period. This feature can be used as a background cycle to ensure that the process image in all partners is the same.
- **General Interrogation** - A general interrogation is only done on request from a client. Not all DataSets contain information which is needed for a general update of the client. For example, data with T(ransient) = TRUE are not part of a GI. When the RCB attribute general interrogation is set to TRUE a GI request from the client will be handled. The report handler will transmit all data defined in the DataSet with their actual values. The IEC 61850 standard defines that all buffered events are transmitted first before the GI is started. A running GI is stopped and a new GI is started, when a new GI request is received while a GI is running.

### Trigger Options

IEC 61850 has defined five different TrgOp. Three of them belong to data attributes and they are marked per data attribute in the column TrgOp of the CDC tables in part 7–3. The other two belong to the configuration of control blocks. The five trigger options are:

- **dchg = data-change** - The classical trigger. Whenever a process value has changed its value either binary or a measurement a transmission is done.
- **qchg = quality change** - Looking to the possibilities of the quality data attribute type (q) any changes in the quality description are transmitted.
- **dupd = data value update** - This trigger option gives the possibility to define that a transmission should be done on a condition which can be controlled by the application.
- **Period** - This trigger forces the transmission of all process values defined in the DataSet when a timer value (the integrity period) expires. For example, it can be used to do a process signal update in the background (e.g. every 15 minutes).
- **General Interrogation** - This trigger is forced by the clients (= station level IED; NCC gateway, station HMI, ...). Normally a GI is asked for when the client and the server start or restart a session. When the client is able to receive the actual values and when the Logical Device has scanned all process values at least once, an image of the actual process signal status can be transmitted to the client.

### Goose Control Blocks (GoCB)

The Generic Object Oriented Substation Event (GOOSE) class model is used to distribute input and output data values between IEDs on bay level through the use of multicast services. GOOSE messages bypass the server which results in a fast transmission from publisher to one or several subscribers (receivers).

The GOOSE message concept is used for all application functions where two or more IEDs are involved.

To send GOOSE messages, a GoCB must be defined and a DataSet is needed which contains the data objects of single data attributes to be sent. A GOOSE message is forced to be transmitted when a trigger change is detected for a data attribute. All members of the DataSet will be copied in the send buffer with their actual value and the message is sent. All subscribers which know the address of this GOOSE message will receive the telegram. The GOOSE message includes a sequence number and a state number to verify that all messages are received.

In the SCL communication section in the GSE element, the GoCB is listed under the ConnectedAP.

The IEDs which should receive a GOOSE message are informed in the SCL private section that they will receive GOOSE messages. This is given when the external Reference, the name of the IED, and the member of the DataSet is included in the LLN0 logic node under the structure of the Logic Device (LD) of the receiving IED. The IEDs which receive the GOOSE are selected by BEST61850.

The IEC 61850 principle operation of GOOSE messages is illustrated in Figure 5-14.

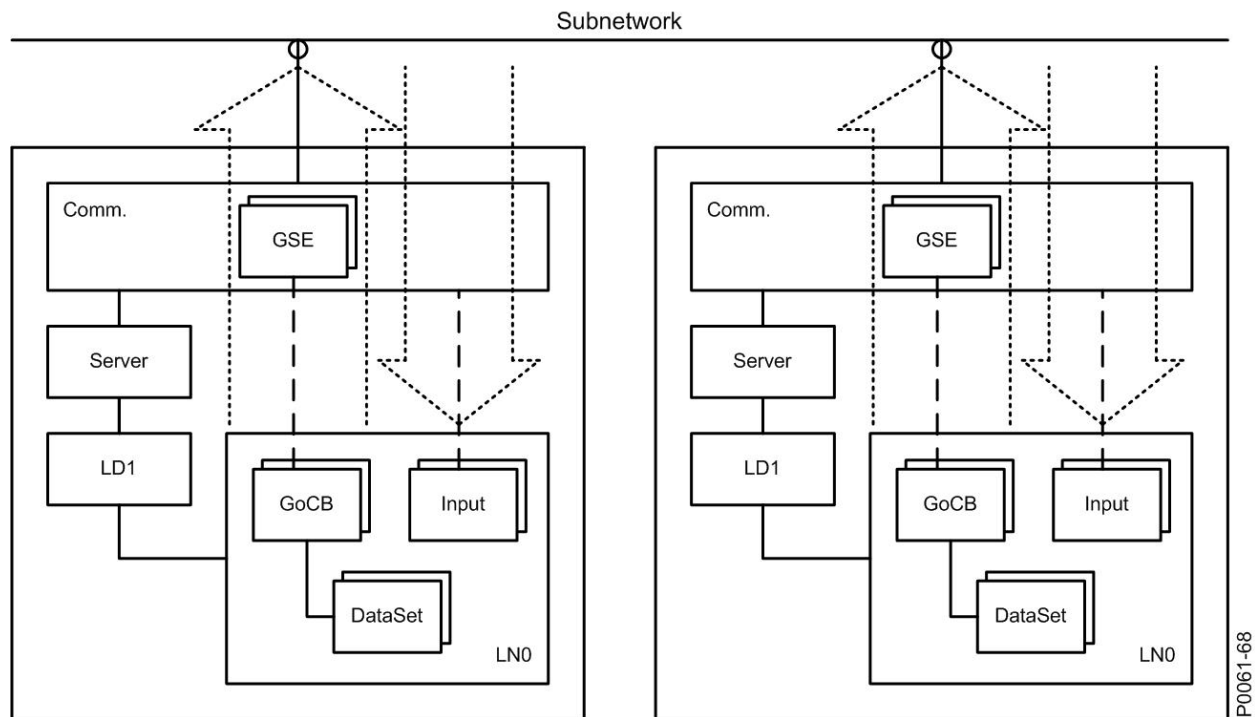


Figure 5-14. IEC 61850: Principle Operation of GOOSE Messages

## BESTCOMSPPlus®

BESTCOMSPPlus is a Windows®-based, PC application that provides a user-friendly, graphical user interface (GUI) for use with Basler Electric communicating products. BESTCOMSPPlus is used to program IEC 61850 logic. It is also used to program BE1-FLEX operational settings and other logic. For more information, refer to the *BESTLogic™Plus* chapter in the BE1-FLEX instruction manual.

### Indicators and Remote Indicators

The Indicator Logic Element is shown in Figure 5-15.

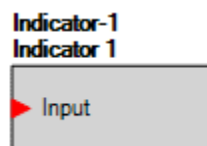


Figure 5-15. Indicator Logic Element

Figure 5-16 shows the Remote Indicator Alarm and Status logic blocks that are available in BESTlogicPlus. Remote Indicators can be set as failsafe TRUE or FALSE state in the event that communication is unavailable.

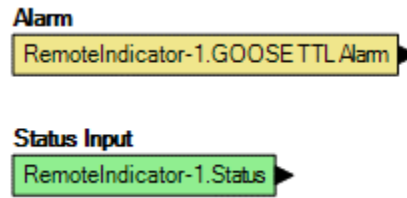


Figure 5-16. Remote Indicator Alarm and Status Logic Blocks

## Alarms

Figure 5-17 shows the Alarm logic blocks that are available in BESTlogicPlus.

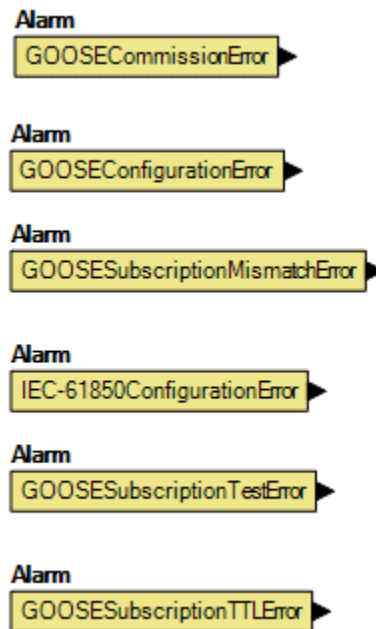


Figure 5-17. Alarm Logic Blocks

## System Configuration Software

System configuration software applications, not provided by Basler Electric for the BE1-FLEX, provide a means to configure IEC 61850 settings for the BE1-FLEX and may perform the following tasks:

- Reads a SCD or CID file and extracts the CID file
- Opens a CID file from a PC or network location
- Configures IED network settings
- Configures DataSets for report control blocks (RCB) and GOOSE control blocks (GoCB)
- Creates and configures Report Control Blocks (RCB)
- Maps GOOSE messages between IED subscribers
- Creates and configures GoCB for each GOOSE message
- Exports a SCD or CID file to a PC or network location

The following sections describe the functionality and settings utilized with System Configuration software applications with the BE1-FLEX.

## Device Information

Device information settings are described in Table 5-1.

**Table 5-1. Device Information Settings**

Setting	Range	Description
IED Name	String (32)	Name of IED (only alphanumeric and underscore characters)
Manufacturer	n/a	Manufacturer of the IED (read only)
Type	n/a	Type of IED (read only)
Description	String (255)	Description of IED (not sent to IED, only saved in BEST61850)
Model Number	n/a	BE1-FLEX Style Number
Application Version	n/a	BE1-FLEX firmware application version
IP Address	Dotted Decimal	IP Address of IED
Subnet Mask	Dotted Decimal	Subnet Mask of IED
Default Gateway	Dotted Decimal	Default Gateway of IED
Revision	n/a	Configuration revision

## DataSets

### Functional Constraints

The IEC 61850 standard defines a number of functional constraints that indicate the data attribute is used for some particular purpose, such as reporting (BR/RP), configuration (CF), control (CO), description (DC), extended definition (EX), GOOSE control (GO), logging (LG), measured analog value (MX), setting groups (SG), setpoint (SP), and status (ST). The functional constraints of a DATA instance determine the rights of services to read and/or write the DATA.

### Data Object

The DATA OBJECT or DATA class (abbreviated DO) like the LN class, is a key element of the IEC 61850 standard. Values of DATA instances represent meaningful information about substation devices, such as currents, voltages, power, phases, temperatures, status, timestamps, and so on. The DATA OBJECT may contain attributes which are themselves instances of the DATA class. Hence, it can be said that the DATA class is recursively defined.

### Data Attribute

The DATA ATTRIBUTE class (abbreviated DA) contains the smallest piece of data that can be defined in IEC 61850. The DA class includes definition of instance name, reference, type, and presence. Presence indicates whether or not the data is mandatory or optional.

## Published IED GOOSE

GOOSE publishing defines the DataSet points that are pushed from the BE1-FLEX and the method in which they are published.

## Subscribed IED GOOSE

GOOSE subscribing defines the DataSet points that are brought into the BE1-FLEX and the internal address to which they are mapped.

## Report Controls

Reporting is handled by the REPORT-CONTROL-BLOCK class of the ACSI. This class controls the procedures that are required for reporting values of DATA from one or more LNs to one client.

Three trigger options (data-change, quality-change, and data-update) can cause a report to be sent to a client. Report control is divided into two classes: BUFFERED-REPORTCONTROLBLOCK (abbreviated BRCB) and UNBUFFERED-REPORT-CONTROLBLOCK abbreviated (URCB).

The class BRCB allows for the sending of reports to be issued immediately, or for the events to be buffered for transmission after an amount of time specified by the Buffered Time setting. Furthermore, BRCB provides the sequence-of-events (SOE) functionality. If the connection is broken when reporting is to take place, the report is buffered and sent when the connection is re-established.

The class URCB only allows transmission of reports according to the time specified by the Buffered Time setting. If the connection is lost, buffering is terminated and the reports are discarded. URCB does not provide SOE functionality.

For both types of reporting, the server must restrict access to an instance of a report control block to one client at a time. The client will be associated with the control block and that client will be the only one receiving reports from the control until the association is released or aborted. In order for more than one client to receive reports of the same values of DATA, multiple instances of the report control block classes must be made available. (The BE1-FLEX contains four instances.) It is also defined in the standard how this should be achieved. In this context, it must be discerned between buffered reporting and unbuffered reporting.

In the case of buffered reporting, it is important that a client, whose connection is lost in the middle of the transmission of the report, is associated with the same report control instance the next time the client reconnects. For this reason, the report control can keep track of which report was successfully transmitted last, and thus, which reports are yet to be transmitted. For unbuffered reporting, this is not necessary. The class provides services for sending a buffered report and reading or writing attributes of a BRCB.

## ***File Transfer***

The BE1-FLEX can transmit the files listed in Table 5-2 when requested by a supported IEC 61850 client.

**Table 5-2. Files Transmitted by the BE1-FLEX**

<b>Description</b>	<b>File Naming Conventions</b>
Fault Record	RO-xxx_SUM.txt
Load Profile	LP-xx.cfg LP-xx.dat
Oscillographic Records	RO-xxx.cfg RO-xxx.dat RO-xxx.hdr
Sequence of Events Report	RO-xxx_SEQ.txt

## ***Data Tags***

The description of the DataSets and the list of data object members or FCDAs (Functionally Constrained Data Attributes) is included in the SCL file in the IED section in the Logical Device subsection. FCDAs are also referred to as Data Tags. As specified in IEC 61850-7-2 clause 9, the DataSets are part of a Logical Node. See the MICS section for Logical Nodes available and SCD or CID files for complete data tag capabilities.

## ***Conformance Statements***

IEC 61850 Self Certification was performed on the BE1-FLEX using the UniGrid SA2.1 tool from DNV-GL to validate all applicable logical nodes, services and test features.

The following conformance statements described in this chapter are applicable for a BE1-FLEX Protection, Automation and Control System.

- MICS (Model Implementation Conformance Statement)

- PICS (Protocol Implementation Conformance Statement)
- PIXIT (Protocol Implementation eXtra Information for Testing)
- TICS (TISSUES Implementation Conformance Statement)

## MICS

The MICS document specifies the modeling extensions in IEC 61850. Table 5-3 contains the list of logical nodes implemented in the device.

**Table 5-3. Logical Nodes**

<b>Node</b>	<b>Element</b>
Logical node zero: LLN0	LLN0
Physical device information: LPHD1	LPHD
Neutral Distance: P21NPDIS	21N
Phase Distance: P21PPDIS	21P
Volts Per Hz: PVPH	24
Undervoltage: PTUV	27
Directional Over Power: PDOP	32 Over Power
Directional Under Power: PDUP	32 Under Power
Time Undercurrent: PTUC	37
Loss of Excitation Reverse Var Based: P40QPDUP	40Q
Loss of Excitation Impedance Based: P40ZPDUP	40Z
Thermal Overload: P49RTDPTTR	49RTD
Instantaneous Overcurrent: PIOC	50
Time Overcurrent: PTOC	51
Voltage Control Overcurrent: PVOC	51 Voltage Control
Through Fault Counter Overcurrent: P51TFPTOC	51TF
Under Power Factor: PUPF	55
Overvoltage: PTOV	59
DC Time Overcurrent: P76PTOC	76
Phase Angle Measuring: P78VPPAM	78V
Phase Angle Measuring:P78PPAM	78OOS
Under Frequency: PTUF	81 Underfrequency
Over Frequency: PTOF	81 Overfrequency
Rate Of Change Frequency: PFRC	81 Rate-of-Change
Flux Balance: P87FBPDIF	87FB
Neutral Differential protection: P87NPDIF	87N
Phase Differential protection: P87PDIF	87
Protection Trip Conditioning: PTRC	Logic Points
Sync-check: RSYN	25
Auto synchronizer: P25ARSYN	25A
Breaker failure: RBRF	50BF
Automatic reclosing : RREC	79
Fault locator: RFLO	Fault Record

Node	Element
Disturbance Recorder Function: RDRE	Fault Record
Generic Process Control Inputs: CTLGGIO	Input Logic Points
Generic Process Indicator Outputs: INDGGIO	Output Logic Points
Multi-Phase Measurement: MMXU	Amps, Volts, Power, Frequency
Multi-Phase Current Measurement: IABCMMXU	Amps
Multi-Phase Voltage Measurement: VABCMMXU	Volts
Ground Voltage Measurement: VGMMXN	VG
Ground Current Measurement: IGMMXN	IG
Sequence Components: MSQI	I0, I1, I2, V0, V1, V2
Current Sequence Components: IABCMSQI	I0, I1, I2
Voltage Sequence Components: VABCMSQI	V0, V1, V2
DC Power, DC Current, DC Voltage: MMDCC	Watts, Amps DC, Volts DC
Circuit breaker: XCBR	Breaker, Breaker Monitor, 101

## PICS

The following ACSI conformance statements are used to provide an overview and details about the BE1-FLEX Protection, Automation and Control System.

- ACSI basic conformance statement (Table 5-4)
- ACSI models conformance statement (Table 5-5)
- ACSI service conformance statement (Table 5-6)

These statements specify the communication features mapped to IEC 61850-1.

Table notes:

- Yes = Supported with MMS services and objects necessary for implementing this capability.
- No = Not supported.
- Blank = Not applicable.

**Table 5-4. ACSI Basic Conformance Statement**

		Client/subscriber	Server/publisher	Value/comments
<b>CLIENT-SERVER ROLES</b>				
B11	Server side (of two-party application-association)		c1	Yes
B12	Client side (of two-party application-association)	c1		No
<b>SCSMS SUPPORTED</b>				
B21	SCSM: IEC 61850-8-1 used			Yes
B22	SCSM: IEC 61850-9-1 used			No
B23	SCSM: IEC 61850-9-2 used			No
B24	SCSM: other			No
<b>GENERIC SUBSTATION EVENT MODEL (GSE)</b>				
B31	Publisher side		O	Yes
B32	Subscriber side	O		Yes

TRANSMISSION OF SAMPLED VALUE MODEL (SVC)				
B41	Publisher side		O	No
B42	Subscriber side	O		No
<b>Notes:</b> c1 - shall be 'M' if support for LOGICAL-DEVICE model has been declared. O - Optional M - Mandatory				

Table 5-5. ACSI Models Conformance Statement

		Client/subscriber	Server/publisher	Value/comments
<b>IF SERVER SIDE (B11) SUPPORTED</b>				
M1	Logical device	c2	c2	Yes
M2	Logical node	c3	c3	Yes
M3	Data	c4	c4	Yes
M4	Data set	c5	c5	Yes
M5	Substitution	O	O	No
M6	Setting group control	O	O	Yes
<b>REPORTING</b>				
M7	Buffered report control	O	O	Yes
M7-1	sequence-number			Yes
M7-2	report-time-stamp			Yes
M7-3	reason-for-inclusion			Yes
M7-4	data-set-name			Yes
M7-5	data-reference			Yes
M7-6	buffer-overflow			Yes
M7-7	entryID			Yes
M7-8	BufTm			Yes
M7-9	IntgPd			Yes
M7-10	GI			Yes
M8	Unbuffered report control	O	O	Yes
M8-1	sequence-number			Yes
M8-2	report-time-stamp			Yes
M8-3	reason-for-inclusion			Yes
M8-4	data-set-name			Yes
M8-5	data-reference			Yes
M8-6	BufTm			Yes
M8-7	IntgPd			Yes
M8-8	GI			Yes
<b>Logging</b>				
M9	Log control	O	O	No
M9-1	IntgPd			No
M10	Log	O	O	No

		Client/subscriber	Server/publisher	Value/comments
M11	Control	M	M	Yes
<b>IF GSE (B31/32) IS SUPPORTED</b>				
GOOSE		O	O	Yes
M12-1	entryID			No
M12-2	DataRefInc			No
M13	GSSE	O	O	No
<b>IF SVC (B41/B42) IS SUPPORTED</b>				
M14	Multicast SVC	O	O	No
M15	Unicast SVC	O	O	No
M16	Time	M	M	Yes
M17	File transfer	O	O	Yes
<b>Notes:</b>				
c2 - shall be 'M' if support for LOGICAL-NODE model has been declared.				
c3 - shall be 'M' if support for DATA model has been declared.				
c4 - shall be 'M' if support for DATA-SET/Substitution/Report/Log Control/or Time model has been declared.				
c5 - shall be 'M' if support for Report/GSE/or SV models has been declared.				
O - Optional				
M - Mandatory				

Table 5-6. ACSI Service Conformance Statement

SERVICES		AA: TP/MC	Client/ subscriber	Server/ publisher	Value/ comments
<b>SERVER</b>					
S1	ServerDirectory	TP		M	Yes
<b>APPLICATION ASSOCIATION</b>					
S2	Associate		M	M	Yes
S3	Abort		M	M	Yes
S4	Release		M	M	Yes
<b>LOGICAL DEVICE</b>					
S5	LogicalDeviceDirectory	TP	M	M	Yes
<b>LOGICAL NODE</b>					
S6	LogicalNodeDirectory	TP	M	M	Yes
S7	GetAllDataValues	TP	O	M	Yes
<b>DATA</b>					
S8	GetDataValues	TP	M	M	Yes
S9	SetDataValues	TP	O	O	Yes
S10	GetDataDirectory	TP	O	M	Yes
S11	GetDataDefinition	TP	O	M	Yes
<b>DATA SET</b>					
S12	GetDataSetValues	TP	O	M	Yes
S13	SetDataSetValues	TP	O	O	No
S14	CreateDataSet	TP	O	O	No

SERVICES		AA: TP/MC	Client/ subscriber	Server/ publisher	Value/ comments
S15	DeleteDataSet	TP	O	O	No
S16	GetDataSetDirectory	TP	O	O	Yes
<b>SUBSTITUTION</b>					
S17	SetDataValues	TP	M	M	No
<b>SETTING GROUP CONTROL</b>					
S18	SelectActiveSG	TP	O	O	Yes
S19	SelectEditSG	TP	O	O	No
S20	SetSGValues	TP	O	O	No
S21	ConfirmEditSGValues	TP	O	O	No
S22	GetSGValues	TP	O	O	No
S23	GetSGCBValues	TP	O	O	No
<b>REPORTING</b>					
<b>BUFFERED REPORT CONTROL BLOCK (BRCB)</b>					
S24	Report	TP	c6	c6	Yes
S24-1	data-change (dchg)				Yes
S24-2	qchg-change (qchg)				Yes
S24-3	data-update (dupd)				Yes
S25	GetBRCBValues	TP	c6	c6	Yes
S26	SetBRCBValues	TP	c6	c6	Yes
<b>UNBUFFERED REPORT CONTROL BLOCK (URCB)</b>					
S27	Report	TP	c6	c6	Yes
S27-1	data-change (dchg)				Yes
S27-2	qchg-change (qchg)				Yes
S27-3	data-update (dupd)				Yes
S28	GetURCBValues	TP	c6	c6	Yes
S29	SetURCBValues	TP	c6	c6	Yes
<b>LOGGING</b>					
<b>LOG CONTROL BLOCK</b>					
S30	GetLCBValues	TP	M	M	No
S31	SetLCBValues	TP	M	M	No
<b>LOG</b>					
S32	QueryLogByTime	TP	c7	M	No
S33	QueryLogByEntry	TP	c7	M	No
S34	GetLogStatusValues	TP	M	M	No
<b>GENERIC SUBSTATION EVENT MODEL (GSE)</b>					
<b>GOOSE-CONTROL-BLOCK</b>					
S35	SendGOOSEMessage	MC	c8	c8	Yes
S36	GetReference	TP	O	c9	No
S37	GetGOOSEElementNumber	TP	c9	c9	No

SERVICES		AA: TP/MC	Client/ subscriber	Server/ publisher	Value/ comments
S38	GetGoCBValues	TP	O	O	Yes
S39	SetGoCBValues	TP	O	O	No
<b>GSSE-CONTROL-BLOCK</b>					
S40	SendGSSEMessage	MC	c8	c8	No
S41	GetReference	TP	O	c9	No
S42	GetGSSEElementNumber	TP	O	c9	No
S43	GetGsCBValues	TP	O	O	No
S44	SetGsCBValues	TP	O	O	No
<b>TRANSMISSION OF SAMPLE VALUE MODEL (SVC)</b>					
<b>MULTICAST SVC</b>					
S45	SendMSVMessage	MC	c10	c10	No
S46	GetMSVCBValues	TP	O	O	No
S47	SetMSVCBValues	TP	O	O	No
<b>UNICAST SVC</b>					
S48	SendUSVMessage	MC	c10	c10	No
S49	GetUSVCBValues	TP	O	O	No
S50	SetUSVCBValues	TP	O	O	No
<b>CONTROL</b>					
S51	Select	O	M	O	No
S52	SelectWithValue	TP	M	O	No
S53	Cancel	TP	O	O	No
S54	Operate	TP	M	M	Yes
S55	Command-Termination	TP	M	O	No
S56	TimeActivated-Operate	TP	O	O	No
<b>FILE TRANSFER</b>					
S57	GetFile	TP	O	M	Yes
S58	SetFile	TP	O	O	No
S59	DeleteFile	TP	O	O	No
S60	GetFileAttributeValues	TP	O	M	Yes
<b>TIME</b>					
T1	Time resolution of internal clock				T5
T2	Time accuracy of internal clock				T0 T1 T2 T3 T5 IRIG T1 NTP
T3	Supported TimeStamp resolution				T5

SERVICES		AA: TP/MC	Client/ subscriber	Server/ publisher	Value/ comments
<b>Notes:</b> c6 - shall declare support for at least one (BRCB or URCB). c7 - shall declare support for at least one (QueryLogByTime or QueryLogAfter). c8 - shall declare support for at least one (SendGOOSEMessage or SendGSSEMessage). c9 - shall declare support if TP association is available. c10 - shall declare support for at least one (SendMSVMessage or SendUSVMessage). TP - Two Party MC - Multicast O - Optional M - Mandatory					

## PIXIT

This document specifies the protocol implementation extra information for testing (PIXIT) of the IEC 61850 interface in the BE1-FLEX Protection, Automation and Control System. Together with the PICS and the MICS, the PIXIT forms the basis for a conformance test according to IEC 61850-10. Tables 5-7 through 5-15 specify the PIXIT for each applicable ACSI service model as structured in IEC 61850-10.

**Table 5-7. PIXIT for Association Model**

Description	Value / Clarification
Maximum number of clients that can set-up an association simultaneously	4
TCP_KEEPALIVE value	20 seconds
Lost connection detection time	20 seconds
Is authentication supported?	Y – Requires ACSE Auth. User configurable passcode
What association parameters are necessary for successful association?	Transport selector Y Session selector Y Presentation selector Y AP Title N AE Qualifier N
If association parameters are necessary for association, describe the correct values e.g.	Transport selector 0001 Session selector 0001 Presentation selector 00000001 AP Title na AE Qualifier na

Description	Value / Clarification
What is the maximum and minimum MMS PDU size?	Max MMS PDU size Negotiated up to 60,000 bytes Min MMS PDU size 500 bytes
What is the maximum startup time after a power supply interrupt?	40 seconds

Table 5-8. PIXIT for Server Model

Description	Value / Clarification
Which analogue value (MX) quality bits are supported (can be set by server)?	Validity: Y Good, Y Invalid, N Reserved, Y Questionable N Overflow Y OutofRange Y BadReference N Oscillatory N Failure N OldData N Inconsistent N Inaccurate Source: Y Process N Substituted N Test N OperatorBlocked
Which status value (ST) quality bits are supported (can be set by server)?	Validity: Y Good, Y Invalid, N Reserved, N Questionable Y BadReference N Oscillatory N Failure N OldData N Inconsistent N Inaccurate Source: Y Process N Substituted N Test N OperatorBlocked

Description	Value / Clarification
What is the maximum number of data values in one GetDataValues request?	Depends on the max MMS PDU size
What is the maximum number of data values in one SetDataValues request?	<i>Depends on the max MMS PDU size</i>

Table 5-9. PIXIT for DataSet Model

Description	Value / Clarification
What is the maximum number of data elements in one data set (compare ICD setting)?	Up to 64 elements per the 32 max pre-defined datasets Up to 32 elements in Dynamic datasets
How many persistent data sets can be created by one or more clients?	There are up to 16 total dynamic datasets supported (persistent + non-persistent)
How many non-persistent data sets can be created by one or more clients?	There are up to 16 total dynamic datasets supported (persistent + non-persistent)

Table 5-10. PIXIT for Setting Group Control Model

Description	Value / Clarification
What is the number of supported setting groups for each logical device (compare NumSG in the SGCB)?	4 (Selection of group only, Editing not supported)
What is the effect of when and how the non-volatile storage is updated? (compare IEC 61850-8-1 §16.2.4)	n/a
Can multiple clients edit the same setting group?	n/a
What happens if the association is lost while editing a setting group?	n/a
Is EditSG value 0 allowed?	n/a

Table 5-11. PIXIT for Reporting Model

Description	Value / Clarification
The supported trigger conditions are (compare PICS)	integrity Y data change Y quality change Y data update Y general interrogation Y

Description	Value / Clarification
The supported optional fields are	sequence-number Y report-time-stamp Y reason-for-inclusion Y data-set-name Y data-reference Y buffer-overflow Y entryID Y conf-rev Y segmentation Y
Can the server send segmented reports?	Y
Mechanism on second internal data change notification of the same analogue data value within buffer period (Compare IEC 61850-7-2 §14.2.2.9)	Send report immediately
Multi-client URCB approach (compare IEC 61850-7-2 §14.2.1)	All client can access all URCB's
What is the format of EntryID?	Octet string 8
What is the buffer size for each BRCB or how many reports can be buffered?	2,000,000 bytes for all report control blocks. More than 400 rcb blocks
Pre-configured RCB attributes that cannot be changed online when RptEna = FALSE (see also the ICD report settings)	<configuration revision>
May the reported data set contain: - structured data objects? - data attributes? - timestamp data attributes?	Y Y Y
What is the scan cycle for binary events? Is this fixed, configurable?	250 msec Configurable through BestComs+ between 5ms to 5000 ms

Table 5-12. PIXIT for Generic Substation Events Model

Description	Value / Clarification								
<p>What elements of a subscribed GOOSE header are checked to decide the message is valid and the allData values are accepted? If yes, describe the conditions. Note: the VLAN tag may be removed by a Ethernet switch and should not be checked</p>	<p>N source MAC address Y destination MAC address Y Ethertype = 0x88B8 Y gocbRef N timeAllowedtoLive Y datSet N Timestamp N goID N t N stNum N sqNum N test Y confRev Y ndsCom Y numDatSetEntries N AppID</p>								
<p>Can the test flag in the published GOOSE be turned on / off?</p>	<p>N</p>								
<p>What is the behavior when the GOOSE publish configuration is incorrect?</p>	<p>GOOSE Configuration Alarm gets set and none of the configured GOOSE publishes will be sent</p>								
<p>When is a subscribed GOOSE marked as lost? (TAL = time allowed to live value from the last received GOOSE message)</p>	<p>n/a</p>								
<p>What is the behavior when one or more subscribed GOOSE messages aren't received or syntactically incorrect (missing GOOSE)?</p>	<p>GOOSE configuration mismatch alarm gets set. DUT uses last received Goose until timeout occurs, then device will switch to the default value input from logic for each GGIO</p>								
<p>What is the behavior when a subscribed GOOSE message is out-of-order?</p>	<p>use last received Goose until timeout occurs, then device will switch to the default value input from logic for each GGIO</p>								
<p>What is the behavior when a subscribed GOOSE message is duplicated?</p>	<p>use last received Goose</p>								
<p>Does the device subscribe to GOOSE messages with/without the VLAN tag?</p>	<p>Y with the VLAN tag Y without the VLAN tag</p>								
<p>May the GOOSE data set contain: - structured data objects? - data attributes? - timestamp data attributes?</p>	<table border="0"> <tr> <td>Subscribed</td> <td>Published</td> </tr> <tr> <td>N</td> <td>N</td> </tr> <tr> <td>Y</td> <td>Y</td> </tr> <tr> <td>N</td> <td>N</td> </tr> </table> <p>DUT can only publish INDGGIO stVals DUT subscriptions can only be mapped to CTLGGIO stVals</p>	Subscribed	Published	N	N	Y	Y	N	N
Subscribed	Published								
N	N								
Y	Y								
N	N								
<p>What is the slow retransmission time? Is it fixed or configurable?</p>	<p>30,000 msec with TAL = 60,000 Configurable in SCL GCBs</p>								

Description	Value / Clarification
What is the fast retransmission scheme? Is it fixed or configurable?	1) 4 msec with TAL = 8 msec 2) 8 msec with TAL = 16 msec 3) 16 msec with TAL = 32 msec 4) 32 msec with TAL = 64 msec 5) 64 msec with TAL = 128 msec 6) 128 msec with TAL = 256 msec Configurable in SCL GCBs
Can the Goose publish be turned on / off by using SetGoCBValues (GoEna)	N

Table 5-13. PIXIT for Control Model

Description	Value / Clarification
What control modes are supported? (compare PICS)	Y status-only Y direct-with-normal-security (CTLGGIO, XCBR, RDRE) N sbo-with-normal-security N direct-with-enhanced-security N sbo-with-enhanced-security
Is the control model fixed, configurable and/or online changeable?	Fixed
Is Time activated operate (operTm) supported?	N
Is “operate-many” supported?	N
What is the behavior of the DUT when the test attribute is set in the SelectWithValue and/or Operate request?	DUT ignores the test value and executes the command as usual
What are the conditions for the time (T) attribute in the SelectWithValue and/or Operate request?	DUT ignores the time value and executes the command as usual
Is pulse configuration supported?	N
What is the behavior of the DUT when the check conditions are set?	N synchrocheck N interlock-check DUT ignores the check value and always perform the action
Is this behavior fixed, configurable, online changeable?	Fixed
What additional cause diagnoses are supported?	N Blocked-by-switching-hierarchy N Select-failed N Invalid-position N Position-reached N Parameter-change-in-execution N Step-limit N Blocked-by-Mode N Blocked-by-process N Blocked-by-interlocking N Blocked-by-synchrocheck N Command-already-in-execution N Blocked-by-health N 1-of-n-control N Abortion-by-cancel N Time-limit-over N Abortion-by-trip
How to force a “test-not-ok” respond with SelectWithValue request?	n/a

Description	Value / Clarification
How to force a “test-not-ok” respond with Select request?	n/a
How to force a “test-not-ok” respond with Operate request?	DOns: Operate is always possible SBOs: not supported DOes: not supported SBOes: not supported
Which origin categories are supported?	DOns: All SBOs: not supported DOes: not supported SBOes: not supported
What happens if the orCat is not supported?	no action
Does the IED accept a selectwithvalue/operate with the same ctlVal as the current status value?	DOns: Yes SBOs: not supported DOes: not supported SBOes: not supported
Does the IED accept a select/operate on the same control object from 2 different clients at the same time?	DOns: Yes SBOs: not supported DOes: not supported SBOes: not supported
Does the IED accept a select/selectwithvalue from the same client when the control object is already selected (tissue 334)	n/a
Is for SBOes the internal validation performed during the SelectWithValue and/or Operate step?	n/a
Can a control operation be blocked by Mod=Off or Blocked?	N
Does the IED support local / remote operation?	N

**Table 5-14. PIXIT for Time and Time Synchronization Model**

Description	Value / Clarification
What quality bits are supported?	N LeapSecondsKnown Y ClockFailure Y ClockNotSynchronized
Describe the behavior when the time synchronization signal/messages are lost	continues to use internal clock
When is the time quality bit "Clock not synchronized" set?	when SNTP or IRIG is not configured when SNTP or IRIG is configured but lost sync
Is the timestamp of a binary event adjusted to the configured scan cycle?	N
Does the device support time zone and daylight saving?	Y
Which attributes of the SNTP response packet are validated?	N Leap indicator not equal to 3? Y Mode is equal to SERVER N OriginateTimestamp is equal to value sent by the SNTP client as Transmit Timestamp N RX/TX timestamp fields are checked for reasonableness Y SNTP version 4 N other (describe)

Table 5-15. PIXIT for File Transfer Model

Description	Value / Clarification
What is structure of files and directories?	Zipped comtrade files /COMTRADE/<IEDNAME>_Fault_#. zip  <IEDNAME>_SequenceOfEvents.txt <IEDNAME>_iec61850.scd <IEDNAME>_logic.bslx <IEDNAME>_settings.bstx
Is the IETF FTP protocol also implemented?	N
Directory names are separated from the file name by	/ (forward slash)
The maximum file name size including path (recommended 64 chars)	up to 255 chars
Are directory/file name case sensitive?	case sensitive
Maximum file size	Maximum file size is not defined. Free space varies and size depends completely on configuration.
Is the requested file path included in the file name of the MMS fileDirectory respond?	Y
Is the wild char supported MMS fileDirectory request?	N
Is it allowed that 2 client get a file at the same time?	N

## TICS

This section provides a template for the tissues conformance statement. According to the UCA IUG QAP, the tissue conformance statement is required to perform a conformance test and is referenced on the certificate.

### Mandatory Edition 2 Tissues

Table 5-16 gives an overview of the applicable mandatory Tissues and their implementation status for the server.

Y = the server has implemented the respective tissue

NA = not applicable if the server does not support the corresponding ACSI service(s)

Table 5-16. Overview of Applicable Mandatory Tissues and Their Implementation Status for the Server

Part 6 Tissue	Description	Implemented Y/NA
658	Tracking related features	NA
663	FCDA element cannot be a "functionally constrained logical node"	NA
668	Autotransformer modeling	NA
687	SGCB ResvTms	NA
719	ConfDataSet - maxAttributes definition is confusing	Y
721	Log element name	NA
768	bType VisString65 is missing	Y
779	object references	Y
788	SICS S56 from optional to mandatory	NA

789	ConfLdName as services applies to both server and client	NA
804	valKind and IED versus System configuration	NA
806	Max length of log name inconsistent between -6 and -7-2	NA
807	Need a way to indicate if "Owner" present in RCB	Y
823	ValKind for structured data attributes	Y
824	Short addresses on structured data attributes	NA
825	Floating point value	NA
845	SGCB ResvTms	NA
853	SBO and ProtNs	Y
855	Recursive SubFunction	NA
856	VoltageLevel frequency and phases	NA
857	Function/SubFunction for ConductingEquipment	NA
886	Missing 8-1 P-types	NA
<b>Part 7-1 Tissue</b>	<b>Description</b>	<b>Implemented Y/NA</b>
828	Data model namespace revision IEC 61850-7-4:2007[A]	Y
1151	Simulated GOOSE disappears after 1 <sup>st</sup> appearance when LPHD.Sim = TRUE	NA
1196	Extensions to standardized LN classes made by third parties	Y
<b>Part 7-2 Tissue</b>	<b>Description</b>	<b>Implemented Y/NA</b>
778	AddCause values – add value not-supported	NA
780	What are unsupported trigger option at a control block?	NA
783	TimOperResp- ; Add Authorization check	NA
786	AddCause values 26 and 27 are switched	NA
820	Mandatory ACSI services (use for PICS template)	Y
858	Typo in enumeration ServiceType	Y
861	Dchg of ConfRev attribute	NA
876	GenLogicNodeClass and SGCB, GoCB, MsvCB, UsvCB	NA
1038	Loss of Info Detection After Resynch	Y
1062	Entrytime not used in CDC	NA
1071	Length of DO name	Y
1091	The sentence "The initial value of EditSG shall be 0", has to be stated in part 7.2 not in 8.1	Y
1127	Missing owner attribute in BTS and UTS	NA
1202	GI not optional	Y
<b>Part 7-3 Tissue</b>	<b>Description</b>	<b>Implemented Y/NA</b>
697	persistent command / PulseConfig	NA
698	Wrong case is BAC.dB attribute	NA

722	Units for 'h' and 'min' not in UnitKind enumeration.	Y
919	Presence Condition for sVC	Y
925	Presence of i or f attribute - Problem with writing	Y
926	Presence Conditions within RangeConfig	NA
<b>Part 7-4 Tissue</b>	<b>Description</b>	<b>Implemented Y/NA</b>
671	Mistake in definition of Mod & Beh	NA
674	CDC of ZRRRC.LocSta is wrong	NA
676	Same data object name used with different CDC	NA
677	MotStr is used with different CDC in PMMS and SOPM LN classes	NA
679	Remove CycTrMod Enum	NA
680	SI unit for MHYD.Cndct	Y
681	Enum PIDAlg	NA
682	ANCR.ParColMod	NA
683	Enum QVVR.IntrDetMth	NA
685	Enum ParTraMod	NA
686	New annex H - enums types in XML	NA
694	Data object CmdBik	NA
696	LSVS.St (Status of subscription)	NA
712	interpretation of quality operatorBlocked	Y
713	DO Naming of time constants in FFIL	NA
724	ANCR.Auto	NA
725	Loc in LN A-group	NA
734	LLN0.OpTmh vs. LPHD.OpTmh	NA
735	ISAF.Alm and ISAF.AlmReset	NA
736	PFSign	NA
742	GAPC.Str, GAPC.Op and GAPC.StrVal	NA
743	CCGR.PmpCtl and CCGR.FanCtl	NA
744	LN STMP, EEHealth and EENAME	NA
772	LPHD.PwrUp/PwrDn shall be transient	NA
773	Loc, LockKey and LocSta YPSH and YLTC	NA
774	ITCI.LockKey	NA
775	KVLV.ClsLim and OpnLim	NA
776	LPHD.OutOv/InOv and LCCH.OutOv/InOv	NA
800	Misspelling in CSYN	NA
802	CCGR and Harmonized control authority	NA
808	Presence condition of ZMOT.DExt and new DOs	NA
831	Setting of ConfRevNum in LGOS	NA
838	Testing in Beh=Blocked	Y

844	MFLK.PhPiMax, MFLK.PhPiLoFil, MFLK.PhPiRoot DEL->WYE	NA
849	Presence conditions re-assessing in case of derived statistical calculation	NA
877	QVUB -settings should be optional	NA
909	Remove ANCR.ColOpR and ColOpL	NA
920	Resetable Counter is NOT resetable	NA
932	Rename AVCO.SptVol to AVCO.VolSpt	NA
939	Change CDC for ANCR.FixCol	NA
991	LGOS: GoCBRef (as well as LSVS.SvCBRef) should be mandatory	NA
1007	PTRC as fault indicator - Update of description required	Y
1044	TapChg in AVCO	NA
1077	Rename DOnames within LTIM	NA

Note: Tissues 675, 735, 772, 775, 776, 878 are not relevant for conformance testing.

<b>Part 8-1 Tissue</b>	<b>Description</b>	<b>Implemented Y/NA</b>
784	Tracking of control (CTS)	NA
817	Fixed-length GOOSE float encoding	NA
834	File dir name length 64	Y
951	Encoding of Owner attribute	Y
1040	More associate error codes	Y
1178	Select Response+ is non-null value	Y
1285	ICD file must have a communication section for 8-1 implementation	Y





