
Relion® Protection and Control

# 630 series
# DNP3 Communication Protocol Manual

Power and productivity
for a better world™

ABB


Document ID: 1MRS756789
Issued: 2011-02-23
Revision: B
Product version: 1.1

# Copyright



## Trademarks



## Warranty

Please inquire about the terms of warranty from your nearest ABB representative.

ABB Oy

Distribution Automation

P.O. Box 699

FI-65101 Vaasa, Finland

Telephone: +358 10 2211

Facsimile: +358 10 22 41094

http://www.abb.com/substationautomation

# Disclaimer

The data, examples and diagrams in this manual are included solely for the concept or product description and are not to be deemed as a statement of guaranteed properties. All persons responsible for applying the equipment addressed in this manual must satisfy themselves that each intended application is suitable and acceptable, including that any applicable safety or other operational requirements are complied with. In particular, any risks in applications where a system failure and/or product failure would create a risk for harm to property or persons (including but not limited to personal injuries or death) shall be the sole responsibility of the person or entity applying the equipment, and those so responsible are hereby requested to ensure that all measures are taken to exclude or mitigate such risks.

This document has been carefully checked by ABB but deviations cannot be completely ruled out. In case any errors are detected, the reader is kindly requested to notify the manufacturer. Other than under explicit contractual commitments, in no event shall ABB be responsible or liable for any loss or damage resulting from the use of this manual or the application of the equipment.

# Conformity

This product complies with the directive of the Council of the European Communities on the approximation of the laws of the Member States relating to electromagnetic compatibility (EMC Directive 2004/108/EC) and concerning electrical equipment for use within specified voltage limits (Low-voltage directive 2006/95/EC). This conformity is the result of tests conducted by ABB in accordance with the product standards EN 50263 and EN 60255-26 for the EMC directive, and with the product standards EN 60255-1 and EN 60255-27 for the low voltage directive. The IED is designed in accordance with the international standards of the IEC 60255 series.

# Table of contents

# Section 1 Introduction

## 1.1 This manual

The communication protocol manual describes a communication protocol supported by the IED. The manual concentrates on vendor-specific implementations.

## 1.2 Intended audience

This manual addresses the communication system engineer or system integrator responsible for pre-engineering and engineering for communication setup in a substation from an IED perspective.

The system engineer or system integrator must have a basic knowledge of communication in protection and control systems and thorough knowledge of the specific communication protocol.

# 1.3 Product documentation

## 1.3.1 Product documentation set

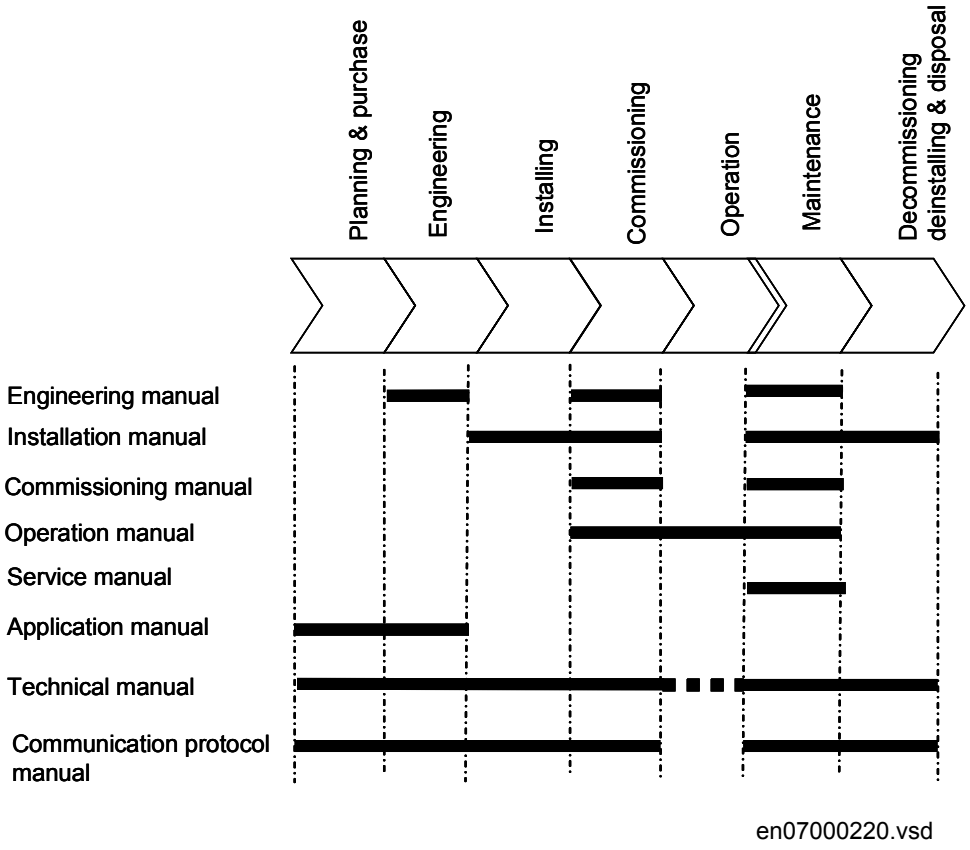


*Figure 1: The intended use of manuals in different lifecycles*

The engineering manual contains instructions on how to engineer the IEDs using the different tools in PCM600. The manual provides instructions on how to set up a PCM600 project and insert IEDs to the project structure. The manual also recommends a sequence for engineering of protection and control functions, LHMI functions as well as communication engineering for IEC 60870-5-103, IEC 61850 and DNP3.

The installation manual contains instructions on how to install the IED. The manual provides procedures for mechanical and electrical installation. The chapters are organized in chronological order in which the IED should be installed.

The commissioning manual contains instructions on how to commission the IED. The manual can also be used by system engineers and maintenance personnel for assistance during the testing phase. The manual provides procedures for checking of external circuitry and energizing the IED, parameter setting and configuration as

well as verifying settings by secondary injection. The manual describes the process of testing an IED in a substation which is not in service. The chapters are organized in chronological order in which the IED should be commissioned.

The operation manual contains instructions on how to operate the IED once it has been commissioned. The manual provides instructions for monitoring, controlling and setting the IED. The manual also describes how to identify disturbances and how to view calculated and measured power grid data to determine the cause of a fault.

The service manual contains instructions on how to service and maintain the IED. The manual also provides procedures for de-energizing, de-commissioning and disposal of the IED.

The application manual contains descriptions of preconfigurations. The manual can be used as a reference for configuring control, protection, measurement, recording and LED functions. The manual can also be used when creating configurations according to specific application requirements.

The technical manual contains application and functionality descriptions and lists function blocks, logic diagrams, input and output signals, setting parameters and technical data sorted per function. The manual can be used as a technical reference during the engineering phase, installation and commissioning phase, and during normal service.

The communication protocol manual describes a communication protocol supported by the IED. The manual concentrates on vendor-specific implementations.

The point list manual describes the outlook and properties of the data points specific to the IED. The manual should be used in conjunction with the corresponding communication protocol manual.

The service manual is not available yet.

## 1.3.2 Document revision history

| Document revision/date | Product series version | History |
|---|---|---|
| A/2009-09-15 | 1.0 | First release |
| B/2011-02-23 | 1.1 | Content updated to correspond to the product series version |

Download the latest documents from the ABB web site http://www.abb.com/substationautomation.

### 1.3.3 Related documentation

Product-specific point list manuals and other product series- and product-specific manuals can be downloaded from the ABB web site http://www.abb.com/substationautomation .

The purpose of this document is to describe specific configuration and interoperability information for an implementation of the Distributed Network Protocol, Version 3.0. This document, in conjunction with the DNP3 Basic 4 Document Set, and the DNP Subset Definitions Document, provides complete information on how to communicate via the DNP3 protocol.

## 1.4 Symbols and conventions

### 1.4.1 Safety indication symbols

The caution icon indicates important information or warning related to the concept discussed in the text. It might indicate the presence of a hazard which could result in corruption of software or damage to equipment or property.

The information icon alerts the reader of important facts and conditions.

The tip icon indicates advice on, for example, how to design your project or how to use a certain function.

Although warning hazards are related to personal injury, it is necessary to understand that under certain operational conditions, operation of damaged equipment may result in degraded process performance leading to personal injury or death. Therefore, comply fully with all warning and caution notices.

### 1.4.2 Manual conventions

Conventions used in IED manuals. A particular convention may not be used in this manual.

- Abbreviations and acronyms in this manual are spelled out in the glossary. The glossary also contains definitions of important terms.
- Push button navigation in the LHMI menu structure is presented by using the push button icons, for example:
  To navigate between the options, use  and  .
- HMI menu paths are presented in bold, for example:

Select **Main menu/Settings**.

- LHMI messages are shown in Courier font, for example:
  To save the changes in non-volatile memory, select `Yes` and press  .
- Parameter names are shown in italics, for example:
  The function can be enabled and disabled with the *Operation* setting.
- The ^ character in front of an input or output signal name in the function block symbol given for a function, indicates that the user can set an own signal name in PCM600.
- The * character after an input or output signal name in the function block symbol given for a function, indicates that the signal must be connected to another function block in the application configuration to achieve a valid application configuration.

# Section 2 DNP3 overview

DNP3 is a set of communications protocols used between components in process automation systems. Its main use is in utilities such as electric and water companies. Usage in other industries is not common, although technically possible. Specifically, it was developed to facilitate communications between various types of data acquisition and control equipment. It plays a crucial role in SCADA systems, where it is used by SCADA master stations (aka Control Centers), RTUs, and IEDs.

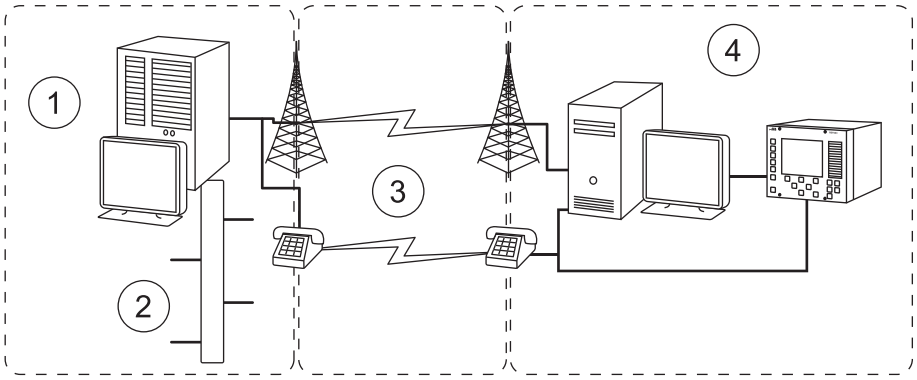


*Figure 2: DNP3 communication schematic representation*

1 SCADA master station / control center

2 External control points

3 Communication links (radio, microwave, spread-spectrum, twisted-pair, fibre-optics, dial-up, leased line)

4 Remote substation (station computer and IED)

## 2.1 DNP3 standard

The DNP3 protocol was developed by Westronic based on the early versions of the IEC 60870-5 standard telecontrol protocol specifications. Now the protocol specification is controlled by the DNP Users Group at www.dnp.org.

The protocol is based on the EPA, a simplified model of the ISO/OSI model. It specifies the data link layer, the application layer and a transport pseudo-layer. To support advanced RTU functions and messages larger than the maximum frame length as defined by the IEC document 60870-5-1, the DNP3 data link is intended to be used with the mentioned transport pseudo-layer. As a minimum, this transport layer implements message assembly and disassembly services.

## Physical layer

Even though the standard does not specify the physical layer, it does however specify how to operate in a networked environment and also suggests how to avoid collisions between simultaneously sending devices.

Many implementations use serial communication based on RS-232, RS-485 or even fibre optics.

DNP3 can also be used over packet-oriented networks such as TCP/IP and UDP in which, for example, Ethernet may be used. In this case DNP3 can be said to be tunneled over TCP/IP or UDP.

Additional information on the DNP3 physical layer is available at the DNP Users Group at www.dnp.org.

## Data link layer

The DNP3 data link layer is designed to operate with asynchronous or synchronous bit serial physical layers. Fully balanced transmission procedures were adopted to support spontaneous transmissions from outstations.

Data link functions include:

- Performing message data link retransmissions.
- Synchronizing and handling the FCB in the control octet.
- Setting and clearing the DFC bit based on buffer availability.
- Packing user data into the defined frame format includes CRC and transmitting the data to the physical layer.
- Unpacking the data link frame received from the physical layer into user data, checking and removing CRC.
- Controlling the physical layer.
- Responding to all valid frames received from the physical layer.

Data link responsibilities:

- Exchange of SDUs between peer DNP3 data links
- Error notification to data link user
- Sequencing of SDUs
- SDU delivery quality.

Link-layer confirm usage is not recommended and the implementation is optional. The IED does not request data-link layer confirmations for TCP/IP communication.

See the DNP technical bulletin TB1998-0402, section 3 for details at www.dnp.org.

### Transport pseudo-layer

To support advanced RTU functions and messages exceeding the maximum data link frame length, a transport pseudo-layer which implements message assembly and disassembly services was adopted.

Transport functions:

- Fragmenting user data into one or more data link frames and transmitting the data to the data link layer
- Assembling the data link frames received from the data link layer into user data
- Controlling all aspects of the data link excluding data link configuration

Transport responsibilities:

- Exchange of SDUs between peer DNP3 transport pseudo layers
- Error notification to transport users
- Sequencing of SDUs

### Application layer

The application layer is responsible for performing operations on data objects defined by the device or on the device itself. These operations include returning actual values (read function), assigning new values (write function) if the object represents control points, arming and energizing the output point (select, operate or direct operate functions) and if counters are used, reading actual values and clearing the counters. DNP3 uses the term point do identify an entity, and these entities can be categorized into point-types, such as analogs or binaries. Points are addressed by giving them an index number and an object is a formatted representation of data from a point. These objects can be assigned to classes in order to organize events and current values into categories. The DNP3 protocol defines four classes; 0 for static data (current value) and 1, 2 and 3 for event data.

### Communication modes

The IED supports four DNP3 communication modes.

- Quiescent operation
- Unsolicited report-by-exception operation
- Polled report-by-exception operation
- Polled static operation

## 2.2 Documentation

This implementation of DNP3 is fully compliant with DNP3 Subset Definition Level 2, and contains significant functionality beyond Subset Level 2. See the device profile for further information.

# Section 3 Vendor-specific implementation

## 3.1 DNP3 link modes

### 3.1.1 DNP3 TCP/IP mode

DNP3 TCP/IP link mode is supported by the IED. This implementation supports up to four different masters communicating simultaneously with the IED. The IED is a listening endpoint implementation and listens for connections from DNP3 masters on a configurable port, *TCPIPLisPort*. The IED does not connect to masters, meaning that it is not a dual-endpoint implementation.

It is possible to use both the connection establishment method based on the master IP address, and the connection establishment method based on the port number. The identification and association of the master is based both on the IP address of the master and the port number it connects to. It is essential to make sure that the parameters *TCPIPLisPort*, *MasterIP-Addr*, *MasterIPNetMask*, *SlaveAddress* and *MasterAddress* uniquely identifies one master from the other masters.

The above is an important concept to grasp during commissioning so that no conflicts occur. Therefore, it is strongly recommended not to change the *MasterIPNetMask* parameter to anything else than its default 255.255.255.255 unless necessary. The parameter should not be mixed up with the subnet mask of the IP configuration. The *MasterIPNetMask* can be used to allow to accept connections from masters that do have dynamic IP addresses within a known range.

For example, if a master changes its IP address dynamically in the range of 10.10.10.1 and 10.10.10.254, the *MasterIPNetMask* could be set to 255.255.255.0 to allow for connections from this range. If two masters share this dynamic range or share the same IP address, it is necessary to separate them by having them connect to separate ports, for example, 20000 and 20001 respectively.

Also, *SlaveAddress* and *MasterAddress* must be correctly configured for each master. Otherwise, the previously accepted connection is closed upon the reception of the first DNP3 message.

The IED supports the requirements of the standard to receive UDP broadcast messages on the ports configured by *UDPPortAccData*. When operating in UDP-only mode, *UDPPortInitNULL* and *UDPPortCliMast* need to be configured as well.

As a default, the IED sends a keep-alive message in every 10 seconds according to the value of the *tKeepAliveT* parameter. The time can be changed, and setting it to zero means that no keep-alive messages are sent. It is important to know the hazards of disabling the keep-alive, and it is not recommended to do so unless

necessary. If the keep-alive messages are unwanted, it is better to increase the value of *tKeepAliveT* so that it exceeds the master's poll rate.

If a master crashes or the communication links are broken and the master restarts, the TCP/IP makes the IED believe that the connection still exists. Since the IED conforms to the recommendations of the standard not to accept new connections when a connection already exists to the particular master, the master will never be allowed to connect again. Another parameter that concerns the TCP/IP connection status is *tBrokenConTout*. It determines how long a session is active after a TCP/IP connection has been broken. After the time period, the session becomes inactive and events are not stored. If the parameter is set to 0, events are stored until the sequential buffers overflow. Note that if the parameter is set to zero, all events from start-up until the sequential buffers overflow are saved even though no connection would have been established.

Further documentation concerning DNP3 TCP/IP communication is available in the IP Networking document Volume 7, from www.dnp.org.

## 3.2 Internal indications

Internal indications give information on certain status and error conditions within the outstation. They contain 2 octets of data and are found in the application layer on an outstation response.

See the DNP3 Specification Volume 3 Application Layer (Section 5 Detailed IIN Bit Descriptions) for more detailed descriptions of IIN bits.

*Table 1: Default class assignment for internal indications*

| Bit index | Descriptions and conditions | Writable |
|---|---|---|
| IIN1.0 | All stations – set after a broadcast message (any message using a destination address of 0xfff0 or above) has been received. Does not indicate an error condition | No |
| IIN1.1 | Class 1 event data available. Can be set at any time and does not indicate an error condition. | No |
| IIN1.2 | Class 2 event data available. Can be set at any time and does not indicate an error condition | No |
| IIN1.3 | Class 3 event data available. Can be set at any time and does not indicate an error condition | No |
| IIN1.4 | Time synchronization required from master. Can be set at any time and does not indicate an error condition. | No |
| IIN1.5 | Local mode. Set if some points are uncontrollable via DNP3. | No |
| IIN1.6 | Device trouble. Set if the IED has detected device problems. | No |
| IIN1.7 | Device restart. Set only under specific conditions. Does not indicate an error condition | Yes |
| IIN2.0 | Function unknown. Generally means that the function code (octet 2 of the request header) cannot be processed. | No |

Table continues on next page

| Bit index | Descriptions and conditions | Writable |
|---|---|---|
| IIN2.1 | Object unknown. Generally means that the function code could be processed but the object group / variation could not be processed | No |
| IIN2.2 | Parameter error. Generally indicates that both function code and object group / variation could be processed but that the qualifier / range field is in error. | No |
| IIN2.3 | Buffer overflow. Indicates that an event buffer has overflowed, and that change events, of at least one type, have been lost. Binary event buffer size is 1000. Counter event buffer size is 1000. Frozen event counter event are not supported. Analog event buffer size is 1000. | No |
| IIN2.4 | Requested operation is already executing. | No |
| IIN2.5 | Configuration corrupted. | No |
| IIN2.6 | Reserved. Always 0. | No |
| IIN2.7 | Reserved. Always 0. | No |

## 3.3 Event reporting

The IED supports spontaneous reporting, that is, unsolicited reporting, of events. Given the parameters *UREvCntThold1*, *tUREvBufTout1*, *UREvCntThold2*, *tUREvBufTout2*, *UREvCntThold3* and *tUREvBufTout3*, the IED can be configured to report events either after a number of events of a certain class have been generated or when at least one event of the class has been generated and the configured time-span has elapsed.

The event system has a rate limiter to reduce CPU load. Each channel has a quota of 10 events/second. If the quota is exceeded the event channel is blocked until the event changes is below the quota.

### 3.3.1 Event buffers

Binary input points, double-bit input points, counters and analog input points each have buffer sizes of 1000 events.

## 3.4 Command handling

DNP3 allows for operation on binary outputs via CROB. Direct Operate, Direct Operate with No Acknowledgement as well as Select/Operate pairs are allowed. The protocol requires that a pair of select- and operate-messages is completely alike and only one sequence number apart. This in turn requires masters not to send any requests between the select message and the operate message, otherwise the operate request will be denied.

Select and Operate requests may contain multiple objects. The select/control buffer size is large enough to hold 10 of the largest select requests possible.

### 3.4.1 Automation bits

Automation bit signals can be used to interpret and execute the count, on-time and off-time parameters of a CROB. Thereby pulse trains of different characteristics and lengths can be generated, and the outputs from the automation bits component can be connected to other function blocks in PCM600.

### 3.4.2 Apparatus control

Apparatuses can be controlled via DNP3. Open and close points to SCSWI are available for mapping in PCM600. These points can then be written to by as CROBs, thereby opening or closing the breaker. It is important to note that the control model, *ctlModel*, of the SCSWI is respected when set to *SBO Enh*. If *ctlModel* is set to *SBO Enh*, direct operate commands from DNP3 are not allowed. On the other hand, if *ctlModel* is set to *Dir Norm*, *SBO* commands from DNP3 are allowed.

Furthermore, the select timeout parameter *tSelectTimeout* in DNP3 should be set so that it harmonizes with the *tSelect* parameter of the SCSWI. The shortest of the two parameters dictates the timing of select/operate.

### 3.4.3 Binary output status points and control relay output blocks

While BOS points are included here for completeness, they are not often polled by DNP3 masters. BOS points represent the most recent value from a command operation for the corresponding CROB point. BOS points are not recommended to be included in class 0 polls.

As an alternative, it is recommended that actual status values affected by CROB points should be mapped as BI or DI. Requesting CROBs on the Open and Close points of SCSWI operate the breaker. The operation may take several seconds to complete. This means that a success response from the operate command may have been returned from the CROB even though the operation is still in progress. Therefore, the mentioned outputs from, for example, SCSWI need to be monitored as a complement.

This implies that the binary output object should not be assigned to classes 1, 2 or 3. A read of the binary outputs returns the last value written to that output.

## 3.5 Time synchronization

DNP3 supports time synchronization of the IED via object numbers 50...52. Time synchronization via DNP3 should only be used if time source with better accuracy is not available, for example, IRIG-B, GPS or SNTP. For TCP/IP channels, the LAN procedure should be used, in which two separate messages are transmitted

from the master, record current time and write, see DNP3 Specification Volume 5 for more information.

Parameters have to be set among the system wide time parameters as well as among the individual DNP3 masters.

DNP3 can be set for a coarse synchronization source under **Configuration/Time/Synchronisation/TimeSynch/CoarseSyncSrc** in the LHMI tree. Note that when DNP3 is set as coarse synchronization source, no fine synchronization source shall be configured. Otherwise, the time will jump between the fine and the coarse synchronization time sources.

Each DNP3 master configuration block has a number of parameters that affect the time synchronization. Only one master at a time is configured to set the time in the IED. Therefore, only one master configuration block enables the *DNPToSetTime* and *TSyncReqAfTout* parameter. That is, both parameters must have the same value and should only be set for one master at a time.

The *tSyncTimeout* parameter defines how long after a successful time synchronization the NeedTime IIN bit has to be set. The *tSyncReqAfTout* parameter defines if the *tSyncTimeout* should be used or not. Also, the IED supports both the new standard directive of use of UTC and local time for backward compatibility (*ExtTimeFormat*). If UTC is selected, the time in the time synchronization messages is expected to be in UTC, and vice versa.

## 3.6 Analog inputs

It is important to note that 16-bit and 32-bit variations of analog inputs are transmitted through DNP3 as signed numbers. The default analog input event buffer size is set 1000.

### 3.6.1 Analog data scaling

The four scaling options associated with analog input data reporting are None, Ratio, Multiplicative and Divisor. The selection *None* means that no scaling is performed on the source IEC 61850 value. The value is reported as such to DNP3.

#### Ratio, multiplicative and divisor scaling methods

The PCM600 tool contains four value arguments related to the scaling methods: *sourceMinVal*, *sourceMaxVal*, *destMinVal* and *destMaxVal*. The use of these arguments differs depending on the scaling method.

The ratio, multiplicative and divisor scaling methods use the first two arguments, *souceMinVal* and *sourceMaxVal*, to define the source value range inside which the object is to be used. The complete value range of the object is usually wanted even though the user could freely define the source range.

Arguments three and four, *destMinVal* and *destMaxVal*, define the destination value range. In ratio scaling, arguments *destMinVal* and *destMaxVal* define the corresponding range of the scaled, reported DNP3 value.

$$DNPvalue = (sourceValue - sourceMinVal) \left[ \frac{(destMaxVal - destMinVal)}{(sourceMaxVal - sourceMinVal)} \right] + destMinVal$$

(Equation 1)

In multiplicative scaling, argument four *destMaxVal* becomes a scale constant.

$$DNPvalue = sourceValue \times destMaxVal$$

(Equation 2)

In divisor scaling, argument four *destMaxVal* becomes a scale constant.

$$DNPvalue = \frac{sourceValue}{destMaxVal}$$

(Equation 3)

## 3.6.2 Analog input signal scaling for DNP3 master presentation

The presentation of an analog value in a telecontrol protocol varies between the different protocols and also with the age of the used protocol version. The range is from a simple 8 bit integer up to a double precision floating point. Internally in the IED many calculations are floating points.

PCM600 supports the re-scaling and the justification to the register presentation given by the project demands.

Figure 3 presents a typical example of a signal flow in the IED from the CTs, VTs to the DNP3 master. The CT, VT is connected to the IED by the transformer module TRM. The SMAI function block is a preprocessor to calculate, check the signals for further use in application function blocks of type MMXU. MMXU calculates the RMS values for the most used analog signals like, U, I, P, Q, for example. The RMS values are available in floating point presentation as output signals of function blocks of type MMXU.

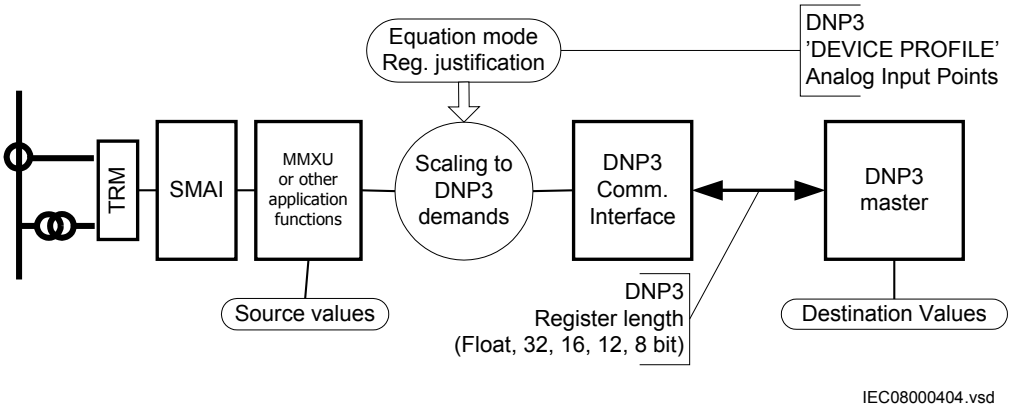


*Figure 3: PCM600: Typical example of DNP3 scaling*

The actual DNP3 specification defines 6 variations for the presentation of an analog value:

- Variation 1 - 32-bit with flag
- Variation 2 - 16-bit with flag
- Variation 3 - 32-bit without flag
- Variation 4 - 16-bit without flag
- Variation 5 - single-precision floating point with flag
- Variation 6 - double-precision floating point with flag

The IED supports all 32-bit and floating point variants without any additional scaling configuration. This is given as long as the *MaxSourceVal* (as it is given in the IED as floating point) is in the range of a 32-bit signed integer value (max. 32-bit = 2 147 483 648).

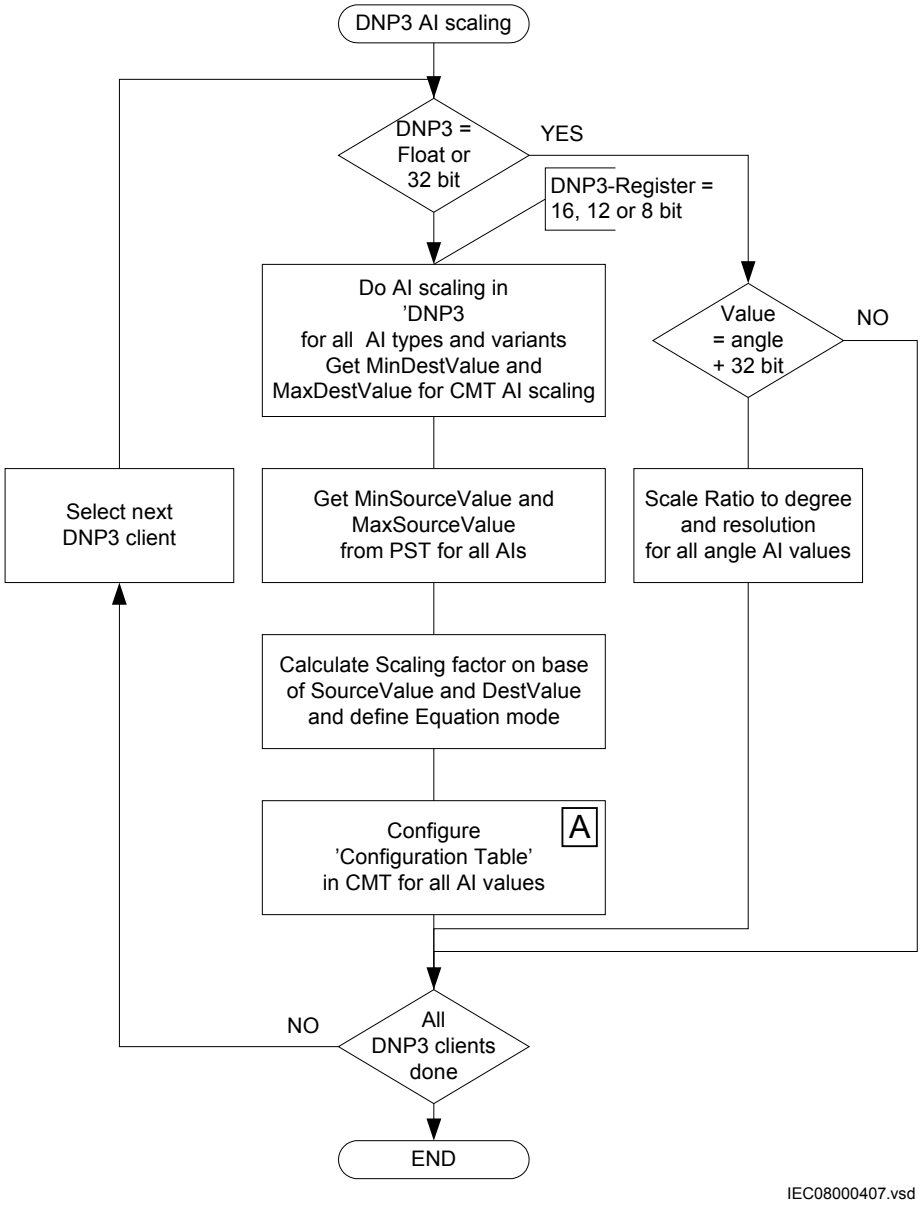


*Figure 4: CMT: Configuration Flowchart*

## 3.7 DNP3 points

See the engineering manual for instructions on how to configure DNP3 with PCM600.

### 3.7.1 Point configuration

The DNP3 point map is configurable in PCM600. All points in the IED are unmapped as default. In PCM600, the unmapped points in the variables list on the left may be inserted to the active point list on the right.

Point gaps may be inserted if wanted. Point gaps cannot be read by the client.

### 3.7.2 Class assignment

Class assignment allows the events generated in the IED to be reported as DNP3 events. Some configurations exceed the class assignment possibilities defined by the standard.

*Table 2: DNP3 point map configuration*

| Configuration | Description |
|---|---|
| None | Integrity class 0 scan returns gap.<br>Value is available only via static scan.<br>Point does not generate events. |
| Class 0 | Point is returned in the class 0 scan.<br>Point does not generate events. |
| Class 0 and any class 1,2,3 combination | Point is returned in the class 0 scan.<br>Point generates events for the selected class or classes. |
| Class 1, 2 or 3 combination | Point is not returned in the class 0 scan.<br>Point generates events for the selected class or classes. |

BOS points exist only if the corresponding CROB point has been inserted in the active point list.

## 3.8 Fault record

Fault record is a mechanism to browse through disturbance records. It shows a snapshot of important information from each existing disturbance record.

Fault record contains signals that provide information on the current disturbance that the user of the *FaultRecord* has selected. It provides signals that help the user to iterate and browse through the existing disturbances. All the signals that can be used to iterate the fault records can be mapped as binary outputs in PCM600 and operated on with CROBs. All signals that provide information on the current disturbance can be mapped as analog inputs and read by the master. The DNP3 master navigates through the *FaultRecord* using the three signals:

- *GetFirstRec* fetches the oldest record in the *FaultRecord*.
- *GetNextRec* fetches the next record in time in the *FaultRecord* relative to the previously fetched record. If the previously fetched record is the newest, no fetch is done.
- *GetPrevRec* fetches the previous record in time in the *FaultRecord* relative to the previously fetched record. If the previously fetched record is the oldest, no fetch is done.

When a new disturbance is recorded, and the outputs are mapped to one of the event classes, events are generated, but the navigation in the *FaultRecord* is not affected. Hence, when the next command is sent from the DNP3 master, the fetched position is relative to the last fetch done; the position in the *FaultRecord* before the new disturbance occurred.

The output signals provide the fault record number, which is the number of the disturbance in the LHMI or PCM600, the number of faults in the IED, the active setting group at the time of the disturbance recording, the trigger signal identity, the time stamp at the trigger time as well as the fault location and the fault type. In addition, the magnitude, angle, fault magnitude and fault angle are provided for up to 30 of the analog channels connected to the disturbance recorder, and for the last 10 analog channels, the calculated value at the trigger time is provided.

# Section 4 DNP3 parameters

## 4.1 Parameter descriptions

The DNP3 parameters for a specific IED can be accessed with PCM600 via **IED Configuration/Station Communication/DNP3.0**. There is one general setting for DNP3 (Disabled/Enabled) and specific settings for each communication TCP/IP channel, for example, CH1TCP and MST1TCP.

The DNP3 block contains one single parameter that controls the operation of DNP3. This parameter must be turned on for the other parameters to have effect. The channel blocks and the master blocks are separate but should be treated as pairs grouped together with a number. For example, CH1TCP and MST1TCP should be treated as an entity during engineering. The reason for this division is that it is conceptually possible to have multiple masters talking on the same channel, for example, a serial link, and it is also possible to imagine a single master switching between different channels, for example, different serial links.

### TCP/IP communication channels settings

*TCPIPLisPort* defines the listen port if the channel is configured for TCP/IP. Default is 20000.

*UDPPortAccData* defines the port on which the UDP datagrams should be accepted if the channel is configured for networking. Default is 20000.

*UDPPortInitNUL* defines the master's destination port to which the initial NULL response should be sent if the channel is configured for networking. Default is 20000.

*UDPPortCliMast* defines the master's destination port to which responses should be sent if the channel is configured for networking. If the parameter is set to 0, the port number is taken from the previous request. Default is 0. There are specific settings for the master sessions if the master session occurs on the serial channel or on the TCP/IP channels.

UDP is not supported. Do not use "UDP-only" for setting *Operation*.

### Master session settings for a specific communication channel

*Operation* determines the operation of the master session. 0 = Off. 1 = On.

*SlaveAddress* defines the DNP3 address of this master session.

*MasterAddress* defines the DNP3 address that this master session uses for communication.

ValMasterAddr determines if the stack should validate the source address in receive frames. DNP3 frames contain both a source address field and a destination address field. If this parameter is set to 0, the stack does not validate the source address and thus the frames whose destination address matches the configured slave session are accepted. If this parameter is set to 1, both the source and the destination addresses have to match before the frame is accepted.

*AddrQueryEnbl* determines whether to enable self-address functionality on this master session (slave) as specified by the DNP Technical Bulletin 2003-003. Self-Address Reservation. The master session (Slave) responds to the address 0xfffc as if it had received a request for its configured address. It responds with its own address so that the master can automatically discover the slave address.

*ApplConfTout* specifies how long the slave waits for the application layer confirmation from the master. This in combination with *unsolRetryDelay* or *unsolOfflineRetryDelay* determines how frequently an unsolicited response is resent.

*ApplMultFrgRes* determines if the application layer of this master session in the slave is allowed to send multi fragment responses.

*ConfMultFrag* determines if application layer confirmations are requested for non-final fragments of a multi-fragment response. Application layer confirmations are always requested for responses that contain events.

*UREnable* determines if unsolicited responses are allowed. If set to 0, no unsolicited responses are generated and requests to enable or disable unsolicited responses fail.

*UREvClassMask* specifies the initial or new state of the unsolicited event mask. This mask is used to determine which event class or classes generate unsolicited responses. According to the DNP3 specification, unsolicited responses should be disabled until an Enable Unsolicited Response request is received from the master. Thus, this value should generally be 0. However, some masters do not generate the Enable Unsolicited Response message, in which case they must be enabled here. Keep the value to 0 for all other purposes.

*UROfflineRetry* specifies the maximum number of unsolicited retries before changing to the offline retry period. Up to 65535 retries can be specified. Set *UROfflRetryDel* to the same value as *URRetryDelay* to define an infinite number of retries.

*tURRetryDelay* specifies in seconds the time to delay after an unsolicited confirm timeout before retrying the unsolicited response.

*tUROfflRetryDel* specifies in seconds the time to delay after an unsolicited timeout before retrying the unsolicited response if *UROfflineRetry* has been attempted. To disable retries after *UROfflineRetry*, set this value to the maximum value of a stack timer: 31 days. This limits the retries to one in every 31 days.

*UREvCntThold1* If unsolicited responses are enabled, this parameter specifies the maximum number of events in class 1 to be allowed before an unsolicited response is generated.

*tUREvBufTout1* If unsolicited responses are enabled (UREnable), this parameter specifies the maximum amount of time in seconds before an unsolicited response is generated after an event in class 1 has been received.

*UREvCntThold2* If unsolicited responses are enabled (UREnable), this parameter specifies the maximum number of allowed class 2 events before an unsolicited response is generated.

*tUREvBufTout2* If unsolicited responses are enabled (UREnable), this parameter specifies the maximum amount of time in seconds before an unsolicited response is generated after an event in class 2 has been received.

*UREvCntThold3* If unsolicited responses are enabled (UREnable), this parameter specifies the maximum number of allowed class 3 events before an unsolicited response will be generated.

*tUREvBufTout3* If unsolicited responses are enabled (UREnable), this parameter specifies the maximum amount of time in seconds before an unsolicited response is generated after an event in class 3 has been received .

*DelOldBufFull* If this parameter is set to 1, the event with the earliest timeStamp is deleted when a new event is added to the full event queue.

*ExtTimeFormat* 0 = LocalTime. 1 = UTC.

*DNPToSetTime* determines if time synch messages received for this master session (slave) are allowed to set the local time in the IED.

*tSynchTimeout* sets the periodicity for time requests. That is, it defines how long after a succeeded time synch message from the master, the IIN.4 bit should be set.

*TsyncReqAfTout* determines if the stack should start with the IIN.4 bit set.

*Averag3TimeReq* determines if the IED needs three time synch messages to set the time. If set, the IIN.4 bit is high until three time synch messages are received. The average of the two best messages are used to set the time.

*MasterIP-Addr* defines the master's IP address.

*MasterIPNetMsk* determines the subnet mask that should be used to mask with the IP address.

The master subnet mask must not be changed unless the master gets its IP-address dynamically assigned via, for example, DHCP. For details see, DNP3 TCP/IP mode

*Obj1DefVar* determines the default variation for Object 1, Binary Inputs.

*Obj2DefVar* determines the default variation for Object 2, Binary Input Change Events.

*Obj3DefVar* determines the default variation for Object 3, Double Bit Inputs.

*Obj4DefVar* determines the default variation for Object 4, Double Bit Input Change Events.

*Obj10DefVar* determines the default variation for Object 10, Binary Output Status.

*Obj20DefVar* determines the default variation for Object 20, Binary Counters.

*Obj22DefVar* determines the default variation for Object 22, Binary Counter Change Events.

*Obj30DefVar* determines the default variation for Object 30, Analog Inputs.

*Obj32DefVar* determines the default variation for Object 32, Analog Change Events.

*PairedPoint* enables the Object12 Close request on an even-index point to access the next-index point.

*tSelectTimeout* specifies the maximum amount of time that a select remains valid before the corresponding operate is received.

*tBrokenConTout* determines how long a session is active after a TCP/IP connection has been broken. After that time period the master session becomes inactive and events are not stored. If the parameter is set to 0, events are stored until the buffers overflow.

*tKeepAliveT* determines, in seconds, how often the DNP3 master session sends keep-alive messages. Default is 10s.

## 4.2 Parameter list

*Table 3: DNPGEN Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>On | - | - | Off | Operation mode Off / On |

*Table 4: CH1TCP Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>TCP/IP<br>UDP-Only[1)] | - | - | Off | Operation mode |
| TCPIPLisPort | 1 - 65535 | - | 1 | 20000 | TCP/IP listen port |
| UDPPortAccData | 1 - 65535 | - | 1 | 20000 | UDP port to accept UDP datagrams from master |
| UDPPortInitNUL | 1 - 65535 | - | 1 | 20000 | UDP port for initial NULL response |
| UDPPortCliMast | 0 - 65535 | - | 1 | 0 | UDP port to remote client/master |

1) Not supported

*Table 5: CH1TCP Non group settings (advanced)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| ApLayMaxRxSize | 20 - 2048 | - | 1 | 2048 | Application layer maximum Rx fragment size |
| ApLayMaxTxSize | 20 - 2048 | - | 1 | 2048 | Application layer maximum Tx fragment size |

*Table 6: CH2TCP Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>TCP/IP<br>UDP-Only[1)] | - | - | Off | Operation mode |
| TCPIPLisPort | 1 - 65535 | - | 1 | 20000 | TCP/IP listen port |
| UDPPortAccData | 1 - 65535 | - | 1 | 20000 | UDP port to accept UDP datagrams from master |
| UDPPortInitNUL | 1 - 65535 | - | 1 | 20000 | UDP port for initial NULL response |
| UDPPortCliMast | 0 - 65535 | - | 1 | 0 | UDP port to remote client/master |

1) Not supported

*Table 7: CH2TCP Non group settings (advanced)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| ApLayMaxRxSize | 20 - 2048 | - | 1 | 2048 | Application layer maximum Rx fragment size |
| ApLayMaxTxSize | 20 - 2048 | - | 1 | 2048 | Application layer maximum Tx fragment size |

*Table 8: CH3TCP Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>TCP/IP<br>UDP-Only[1)] | - | - | Off | Operation mode |
| TCPIPLisPort | 1 - 65535 | - | 1 | 20000 | TCP/IP listen port |
| UDPPortAccData | 1 - 65535 | - | 1 | 20000 | UDP port to accept UDP datagrams from master |
| UDPPortInitNUL | 1 - 65535 | - | 1 | 20000 | UDP port for initial NULL response |
| UDPPortCliMast | 0 - 65535 | - | 1 | 0 | UDP port to remote client/master |

1) Not supported

*Table 9: CH3TCP Non group settings (advanced)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| ApLayMaxRxSize | 20 - 2048 | - | 1 | 2048 | Application layer maximum Rx fragment size |
| ApLayMaxTxSize | 20 - 2048 | - | 1 | 2048 | Application layer maximum Tx fragment size |

*Table 10: CH4TCP Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>TCP/IP<br>UDP-Only[1)] | - | - | Off | Operation mode |
| TCPIPLisPort | 1 - 65535 | - | 1 | 20000 | TCP/IP listen port |
| UDPPortAccData | 1 - 65535 | - | 1 | 20000 | UDP port to accept UDP datagrams from master |
| UDPPortInitNUL | 1 - 65535 | - | 1 | 20000 | UDP port for initial NULL response |
| UDPPortCliMast | 0 - 65535 | - | 1 | 0 | UDP port to remote client/master |

1) Not supported

*Table 11: CH4TCP Non group settings (advanced)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| ApLayMaxRxSize | 20 - 2048 | - | 1 | 2048 | Application layer maximum Rx fragment size |
| ApLayMaxTxSize | 20 - 2048 | - | 1 | 2048 | Application layer maximum Tx fragment size |

*Table 12: MST1TCP Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>On | - | - | Off | Operation Off / On |
| SlaveAddress | 0 - 65519 | - | 1 | 1 | Slave address |
| MasterAddres | 0 - 65519 | - | 1 | 1 | Master address |
| ValMasterAddr | No<br>Yes | - | - | Yes | Validate source (master) address |
| MasterIP-Addr | 0 - 18 | IP Address | 1 | 0.0.0.0 | Master IP-address |
| MasterIPNetMsk | 0 - 18 | IP Address | 1 | 255.255.255.255 | Master IP net mask |
| Obj1DefVar | 1:BISingleBit<br>2:BIWithStatus | - | - | 1:BISingleBit | Object 1, default variation |
| Obj2DefVar | 1:BIChWithoutTime<br>2:BIChWithTime<br>3:BIChWithRelTime | - | - | 3:BIChWithRelTime | Object 2, default variation |
| Obj3DefVar | 1:DIWithoutFlag<br>2:DIWithFlag | - | - | 1:DIWithoutFlag | Object 3, default variation |
| Obj4DefVar | 1:DIChWithoutTime<br>2:DIChWithTime<br>3:DIChWithRelTime | - | - | 3:DIChWithRelTime | Object 4, default variation |
| Obj10DefVar | 1:BO<br>2:BOStatus | - | - | 2:BOStatus | Object 10, default variation |
| Obj20DefVar | 1:BinCnt32<br>2:BinCnt16<br>5:BinCnt32WoutF<br>6:BinCnt16WoutF | - | - | 5:BinCnt32WoutF | Object 20, default variation |
| Obj22DefVar | 1:BinCnt32EvWoutT<br>2:BinCnt16EvWoutT<br>5:BinCnt32EvWithT<br>6:BinCnt16EvWithT | - | - | 1:BinCnt32EvWoutT | Object 22, default variation |
| Obj30DefVar | 1:AI32Int<br>2:AI16Int<br>3:AI32IntWithoutF<br>4:AI16IntWithoutF<br>5:AI32FltWithF<br>6:AI64FltWithF | - | - | 3:AI32IntWithoutF | Object 30, default variation |
| Obj32DefVar | 1:AI32IntEvWoutF<br>2:AI16IntEvWoutF<br>3:AI32IntEvWithFT<br>4:AI16IntEvWithFT<br>5:AI32FltEvWithF<br>6:AI64FltEvWithF<br>7:AI32FltEvWithFT<br>8:AI64FltEvWithFT | - | - | 1:AI32IntEvWoutF | Object 32, default variation |

*Table 13: MST1TCP Non group settings (advanced)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| AddrQueryEnbl | No<br>Yes | - | - | Yes | Address query enable |
| tApplConfTout | 0.00 - 300.00 | s | 0.01 | 10.00 | Application layer confim timeout |
| ApplMultFrgRes | No<br>Yes | - | - | Yes | Enable application for multiple fragment response |
| ConfMultFrag | No<br>Yes | - | - | Yes | Confirm each multiple fragment |
| UREnable | No<br>Yes | - | - | Yes | Unsolicited response enabled |
| UREvClassMask | Off<br>Class 1<br>Class 2<br>Class 1 and 2<br>Class 3<br>Class 1 and 3<br>Class 2 and 3<br>Class 1, 2 and 3 | - | - | Off | Unsolicited response, event class mask |
| UROfflineRetry | 0 - 10 | - | 1 | 5 | Unsolicited response retries before off-line retry mode |
| tURRetryDelay | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response retry delay in s |
| tUROfflRtryDel | 0.00 - 60.00 | s | 0.01 | 30.00 | Unsolicited response off-line retry delay in s |
| UREvCntThold1 | 1 - 100 | - | 1 | 5 | Unsolicited response class 1 event count report treshold |
| tUREvBufTout1 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 1 event buffer timeout |
| UREvCntThold2 | 1 - 100 | - | 1 | 5 | Unsolicited response class 2 event count report treshold |
| tUREvBufTout2 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 2 event buffer timeout |
| UREvCntThold3 | 1 - 100 | - | 1 | 5 | Unsolicited response class 3 event count report treshold |
| tUREvBufTout3 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 3 event buffer timeout |
| DelOldBufFull | No<br>Yes | - | - | No | Delete oldest event when buffer is full |
| ExtTimeFormat | LocalTime<br>UTC | - | - | UTC | External time format |
| DNPToSetTime | No<br>Yes | - | - | No | Allow DNP to set time in IED |
| tSynchTimeout | 30 - 3600 | s | 1 | 1800 | Time synch timeout before error status is generated |
| TSyncReqAfTout | No<br>Yes | - | - | No | Time synchronization request after timeout |
| Averag3TimeReq | No<br>Yes | - | - | No | Use average of 3 time requests |
| PairedPoint | No<br>Yes | - | - | Yes | Enable paired point |

Table continues on next page

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| tSelectTimeout | 1.0 - 60.0 | s | 0.1 | 30.0 | Select timeout |
| tBrokenConTout | 0 - 3600 | s | 1 | 0 | Broken connection timeout |
| tKeepAliveT | 0 - 3600 | s | 1 | 10 | Keep-Alive timer |

*Table 14: MST2TCP Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>On | - | - | Off | Operation Off / On |
| SlaveAddress | 0 - 65519 | - | 1 | 1 | Slave address |
| MasterAddres | 0 - 65519 | - | 1 | 1 | Master address |
| ValMasterAddr | No<br>Yes | - | - | Yes | Validate source (master) address |
| MasterIP-Addr | 0 - 18 | IP Address | 1 | 0.0.0.0 | Master IP-address |
| MasterIPNetMsk | 0 - 18 | IP Address | 1 | 255.255.255.255 | Master IP net mask |
| Obj1DefVar | 1:BISingleBit<br>2:BIWithStatus | - | - | 1:BISingleBit | Object 1, default variation |
| Obj2DefVar | 1:BIChWithoutTime<br>2:BIChWithTime<br>3:BIChWithRelTime | - | - | 3:BIChWithRelTime | Object 2, default variation |
| Obj3DefVar | 1:DIWithoutFlag<br>2:DIWithFlag | - | - | 1:DIWithoutFlag | Object 3, default variation |
| Obj4DefVar | 1:DIChWithoutTime<br>2:DIChWithTime<br>3:DIChWithRelTime | - | - | 3:DIChWithRelTime | Object 4, default variation |
| Obj10DefVar | 1:BO<br>2:BOStatus | - | - | 2:BOStatus | Object 10, default variation |
| Obj20DefVar | 1:BinCnt32<br>2:BinCnt16<br>5:BinCnt32WoutF<br>6:BinCnt16WoutF | - | - | 5:BinCnt32WoutF | Object 20, default variation |
| Obj22DefVar | 1:BinCnt32EvWoutT<br>2:BinCnt16EvWoutT<br>5:BinCnt32EvWithT<br>6:BinCnt16EvWithT | - | - | 1:BinCnt32EvWoutT | Object 22, default variation |
| Obj30DefVar | 1:AI32Int<br>2:AI16Int<br>3:AI32IntWithoutF<br>4:AI16IntWithoutF<br>5:AI32FltWithF<br>6:AI64FltWithF | - | - | 3:AI32IntWithoutF | Object 30, default variation |
| Obj32DefVar | 1:AI32IntEvWoutF<br>2:AI16IntEvWoutF<br>3:AI32IntEvWithFT<br>4:AI16IntEvWithFT<br>5:AI32FltEvWithF<br>6:AI64FltEvWithF<br>7:AI32FltEvWithFT<br>8:AI64FltEvWithFT | - | - | 1:AI32IntEvWoutF | Object 32, default variation |

*Table 15: MST2TCP Non group settings (advanced)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| AddrQueryEnbl | No<br>Yes | - | - | Yes | Address query enable |
| tApplConfTout | 0.00 - 300.00 | s | 0.01 | 10.00 | Application layer confim timeout |
| ApplMultFrgRes | No<br>Yes | - | - | Yes | Enable application for multiple fragment response |
| ConfMultFrag | No<br>Yes | - | - | Yes | Confirm each multiple fragment |
| UREnable | No<br>Yes | - | - | Yes | Unsolicited response enabled |
| UREvClassMask | Off<br>Class 1<br>Class 2<br>Class 1 and 2<br>Class 3<br>Class 1 and 3<br>Class 2 and 3<br>Class 1, 2 and 3 | - | - | Off | Unsolicited response, event class mask |
| UROfflineRetry | 0 - 10 | - | 1 | 5 | Unsolicited response retries before off-line retry mode |
| tURRetryDelay | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response retry delay in s |
| tUROfflRtryDel | 0.00 - 60.00 | s | 0.01 | 30.00 | Unsolicited response off-line retry delay in s |
| UREvCntThold1 | 1 - 100 | - | 1 | 5 | Unsolicited response class 1 event count report treshold |
| tUREvBufTout1 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 1 event buffer timeout |
| UREvCntThold2 | 1 - 100 | - | 1 | 5 | Unsolicited response class 2 event count report treshold |
| tUREvBufTout2 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 2 event buffer timeout |
| UREvCntThold3 | 1 - 100 | - | 1 | 5 | Unsolicited response class 3 event count report treshold |
| tUREvBufTout3 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 3 event buffer timeout |
| DelOldBufFull | No<br>Yes | - | - | No | Delete oldest event when buffer is full |
| ExtTimeFormat | LocalTime<br>UTC | - | - | UTC | External time format |
| DNPToSetTime | No<br>Yes | - | - | No | Allow DNP to set time in IED |
| tSynchTimeout | 30 - 3600 | s | 1 | 1800 | Time synch timeout before error status is generated |
| TSyncReqAfTout | No<br>Yes | - | - | No | Time synchronization request after timeout |
| Averag3TimeReq | No<br>Yes | - | - | No | Use average of 3 time requests |
| PairedPoint | No<br>Yes | - | - | Yes | Enable paired point |

Table continues on next page

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| tSelectTimeout | 1.0 - 60.0 | s | 0.1 | 30.0 | Select timeout |
| tBrokenConTout | 0 - 3600 | s | 1 | 0 | Broken connection timeout |
| tKeepAliveT | 0 - 3600 | s | 1 | 10 | Keep-Alive timer |

*Table 16: MST3TCP Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>On | - | - | Off | Operation Off / On |
| SlaveAddress | 0 - 65519 | - | 1 | 1 | Slave address |
| MasterAddres | 0 - 65519 | - | 1 | 1 | Master address |
| ValMasterAddr | No<br>Yes | - | - | Yes | Validate source (master) address |
| MasterIP-Addr | 0 - 18 | IP Address | 1 | 0.0.0.0 | Master IP-address |
| MasterIPNetMsk | 0 - 18 | IP Address | 1 | 255.255.255.255 | Master IP net mask |
| Obj1DefVar | 1:BISingleBit<br>2:BIWithStatus | - | - | 1:BISingleBit | Object 1, default variation |
| Obj2DefVar | 1:BIChWithoutTime<br>2:BIChWithTime<br>3:BIChWithRelTime | - | - | 3:BIChWithRelTime | Object 2, default variation |
| Obj3DefVar | 1:DIWithoutFlag<br>2:DIWithFlag | - | - | 1:DIWithoutFlag | Object 3, default variation |
| Obj4DefVar | 1:DIChWithoutTime<br>2:DIChWithTime<br>3:DIChWithRelTime | - | - | 3:DIChWithRelTime | Object 4, default variation |
| Obj10DefVar | 1:BO<br>2:BOStatus | - | - | 2:BOStatus | Object 10, default variation |
| Obj20DefVar | 1:BinCnt32<br>2:BinCnt16<br>5:BinCnt32WoutF<br>6:BinCnt16WoutF | - | - | 5:BinCnt32WoutF | Object 20, default variation |
| Obj22DefVar | 1:BinCnt32EvWoutT<br>2:BinCnt16EvWoutT<br>5:BinCnt32EvWithT<br>6:BinCnt16EvWithT | - | - | 1:BinCnt32EvWoutT | Object 22, default variation |
| Obj30DefVar | 1:AI32Int<br>2:AI16Int<br>3:AI32IntWithoutF<br>4:AI16IntWithoutF<br>5:AI32FltWithF<br>6:AI64FltWithF | - | - | 3:AI32IntWithoutF | Object 30, default variation |
| Obj32DefVar | 1:AI32IntEvWoutF<br>2:AI16IntEvWoutF<br>3:AI32IntEvWithFT<br>4:AI16IntEvWithFT<br>5:AI32FltEvWithF<br>6:AI64FltEvWithF<br>7:AI32FltEvWithFT<br>8:AI64FltEvWithFT | - | - | 1:AI32IntEvWoutF | Object 32, default variation |

*Table 17: MST3TCP Non group settings (advanced)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| AddrQueryEnbl | No<br>Yes | - | - | Yes | Address query enable |
| tApplConfTout | 0.00 - 300.00 | s | 0.01 | 10.00 | Application layer confim timeout |
| ApplMultFrgRes | No<br>Yes | - | - | Yes | Enable application for multiple fragment response |
| ConfMultFrag | No<br>Yes | - | - | Yes | Confirm each multiple fragment |
| UREnable | No<br>Yes | - | - | Yes | Unsolicited response enabled |
| UREvClassMask | Off<br>Class 1<br>Class 2<br>Class 1 and 2<br>Class 3<br>Class 1 and 3<br>Class 2 and 3<br>Class 1, 2 and 3 | - | - | Off | Unsolicited response, event class mask |
| UROfflineRetry | 0 - 10 | - | 1 | 5 | Unsolicited response retries before off-line retry mode |
| tURRetryDelay | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response retry delay in s |
| tUROfflRtryDel | 0.00 - 60.00 | s | 0.01 | 30.00 | Unsolicited response off-line retry delay in s |
| UREvCntThold1 | 1 - 100 | - | 1 | 5 | Unsolicited response class 1 event count report treshold |
| tUREvBufTout1 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 1 event buffer timeout |
| UREvCntThold2 | 1 - 100 | - | 1 | 5 | Unsolicited response class 2 event count report treshold |
| tUREvBufTout2 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 2 event buffer timeout |
| UREvCntThold3 | 1 - 100 | - | 1 | 5 | Unsolicited response class 3 event count report treshold |
| tUREvBufTout3 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 3 event buffer timeout |
| DelOldBufFull | No<br>Yes | - | - | No | Delete oldest event when buffer is full |
| ExtTimeFormat | LocalTime<br>UTC | - | - | UTC | External time format |
| DNPToSetTime | No<br>Yes | - | - | No | Allow DNP to set time in IED |
| tSynchTimeout | 30 - 3600 | s | 1 | 1800 | Time synch timeout before error status is generated |
| TSyncReqAfTout | No<br>Yes | - | - | No | Time synchronization request after timeout |
| Averag3TimeReq | No<br>Yes | - | - | No | Use average of 3 time requests |
| PairedPoint | No<br>Yes | - | - | Yes | Enable paired point |

Table continues on next page

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| tSelectTimeout | 1.0 - 60.0 | s | 0.1 | 30.0 | Select timeout |
| tBrokenConTout | 0 - 3600 | s | 1 | 0 | Broken connection timeout |
| tKeepAliveT | 0 - 3600 | s | 1 | 10 | Keep-Alive timer |

*Table 18: MST4TCP Non group settings (basic)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| Operation | Off<br>On | - | - | Off | Operation Off / On |
| SlaveAddress | 0 - 65519 | - | 1 | 1 | Slave address |
| MasterAddres | 0 - 65519 | - | 1 | 1 | Master address |
| ValMasterAddr | No<br>Yes | - | - | Yes | Validate source (master) address |
| MasterIP-Addr | 0 - 18 | IP Address | 1 | 0.0.0.0 | Master IP-address |
| MasterIPNetMsk | 0 - 18 | IP Address | 1 | 255.255.255.255 | Master IP net mask |
| Obj1DefVar | 1:BISingleBit<br>2:BIWithStatus | - | - | 1:BISingleBit | Object 1, default variation |
| Obj2DefVar | 1:BIChWithoutTime<br>2:BIChWithTime<br>3:BIChWithRelTime | - | - | 3:BIChWithRelTime | Object 2, default variation |
| Obj3DefVar | 1:DIWithoutFlag<br>2:DIWithFlag | - | - | 1:DIWithoutFlag | Object 3, default variation |
| Obj4DefVar | 1:DIChWithoutTime<br>2:DIChWithTime<br>3:DIChWithRelTime | - | - | 3:DIChWithRelTime | Object 4, default variation |
| Obj10DefVar | 1:BO<br>2:BOStatus | - | - | 2:BOStatus | Object 10, default variation |
| Obj20DefVar | 1:BinCnt32<br>2:BinCnt16<br>5:BinCnt32WoutF<br>6:BinCnt16WoutF | - | - | 5:BinCnt32WoutF | Object 20, default variation |
| Obj22DefVar | 1:BinCnt32EvWoutT<br>2:BinCnt16EvWoutT<br>5:BinCnt32EvWithT<br>6:BinCnt16EvWithT | - | - | 1:BinCnt32EvWoutT | Object 22, default variation |
| Obj30DefVar | 1:AI32Int<br>2:AI16Int<br>3:AI32IntWithoutF<br>4:AI16IntWithoutF<br>5:AI32FltWithF<br>6:AI64FltWithF | - | - | 3:AI32IntWithoutF | Object 30, default variation |
| Obj32DefVar | 1:AI32IntEvWoutF<br>2:AI16IntEvWoutF<br>3:AI32IntEvWithFT<br>4:AI16IntEvWithFT<br>5:AI32FltEvWithF<br>6:AI64FltEvWithF<br>7:AI32FltEvWithFT<br>8:AI64FltEvWithFT | - | - | 1:AI32IntEvWoutF | Object 32, default variation |

*Table 19: MST4TCP Non group settings (advanced)*

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| AddrQueryEnbl | No<br>Yes | - | - | Yes | Address query enable |
| tApplConfTout | 0.00 - 300.00 | s | 0.01 | 10.00 | Application layer confim timeout |
| ApplMultFrgRes | No<br>Yes | - | - | Yes | Enable application for multiple fragment response |
| ConfMultFrag | No<br>Yes | - | - | Yes | Confirm each multiple fragment |
| UREnable | No<br>Yes | - | - | Yes | Unsolicited response enabled |
| UREvClassMask | Off<br>Class 1<br>Class 2<br>Class 1 and 2<br>Class 3<br>Class 1 and 3<br>Class 2 and 3<br>Class 1, 2 and 3 | - | - | Off | Unsolicited response, event class mask |
| UROfflineRetry | 0 - 10 | - | 1 | 5 | Unsolicited response retries before off-line retry mode |
| tURRetryDelay | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response retry delay in s |
| tUROfflRtryDel | 0.00 - 60.00 | s | 0.01 | 30.00 | Unsolicited response off-line retry delay in s |
| UREvCntThold1 | 1 - 100 | - | 1 | 5 | Unsolicited response class 1 event count report treshold |
| tUREvBufTout1 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 1 event buffer timeout |
| UREvCntThold2 | 1 - 100 | - | 1 | 5 | Unsolicited response class 2 event count report treshold |
| tUREvBufTout2 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 2 event buffer timeout |
| UREvCntThold3 | 1 - 100 | - | 1 | 5 | Unsolicited response class 3 event count report treshold |
| tUREvBufTout3 | 0.00 - 60.00 | s | 0.01 | 5.00 | Unsolicited response class 3 event buffer timeout |
| DelOldBufFull | No<br>Yes | - | - | No | Delete oldest event when buffer is full |
| ExtTimeFormat | LocalTime<br>UTC | - | - | UTC | External time format |
| DNPToSetTime | No<br>Yes | - | - | No | Allow DNP to set time in IED |
| tSynchTimeout | 30 - 3600 | s | 1 | 1800 | Time synch timeout before error status is generated |
| TSyncReqAfTout | No<br>Yes | - | - | No | Time synchronization request after timeout |
| Averag3TimeReq | No<br>Yes | - | - | No | Use average of 3 time requests |
| PairedPoint | No<br>Yes | - | - | Yes | Enable paired point |

Table continues on next page

| Name | Values (Range) | Unit | Step | Default | Description |
|---|---|---|---|---|---|
| tSelectTimeout | 1.0 - 60.0 | s | 0.1 | 30.0 | Select timeout |
| tBrokenConTout | 0 - 3600 | s | 1 | 0 | Broken connection timeout |
| tKeepAliveT | 0 - 3600 | s | 1 | 10 | Keep-Alive timer |

# Section 5 Glossary

| | |
|---|---|
| **BI** | Binary input |
| **BOS** | Binary output status |
| **CRC** | Cyclical redundancy check |
| **CROB** | Control relay output block |
| **CT** | Current transformer |
| **DFC** | Data flow control |
| **DI** | Digital input |
| **DNP3** | A distributed network protocol originally developed by Westronic. The DNP3 Users Group has the ownership of the protocol and assumes responsibility for its evolution. |
| **EMC** | Electromagnetic compatibility |
| **EPA** | Enhanced performance architecture |
| **Ethernet** | A standard for connecting a family of frame-based computer networking technologies into a LAN |
| **FCB** | Flow control bit; Frame count bit |
| **GPS** | Global Positioning System |
| **HMI** | Human-machine interface |
| **IEC** | International Electrotechnical Commission |
| **IEC 60870-5** | IEC standard for telecontrol equipment and systems. Part 5 defines transmission protocols. |
| **IEC 60870-5-103** | Communication standard for protective equipment; A serial master/slave protocol for point-to-point communication |
| **IEC 61850** | International standard for substation communication and modeling |
| **IED** | Intelligent electronic device |
| **IP** | Internet protocol |
| **IP address** | A set of four numbers between 0 and 255, separated by periods. Each server connected to the Internet is assigned a unique IP address that specifies the location for the TCP/IP protocol. |
| **IRIG-B** | Inter-Range Instrumentation Group's time code format B |
| **ISO** | International Standard Organization |

| | |
|---|---|
| LAN | Local area network |
| LED | Light-emitting diode |
| LHMI | Local human-machine interface |
| OSI | Open systems interconnection |
| PCM600 | Protection and Control IED Manager |
| RMS | Root-mean-square (value) |
| RS-232 | Serial interface standard |
| RS-485 | Serial link according to EIA standard RS485 |
| RTU | Remote terminal unit |
| SCADA | Supervision, control and data acquisition |
| SDU | Service data unit |
| SNTP | Simple Network Time Protocol |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| UDP | User datagram protocol |
| UTC | Coordinated universal time |
| VT | Voltage transformer |

# Contact us

**ABB Oy**
**Distribution Automation**
P.O. Box 699
FI-65101 VAASA, Finland
Phone +358 10 22 11
Fax +358 10 22 41094

**www.abb.com/substationautomation**



Power and productivity
for a better world™