

Atop Technologies, Inc.

Protocol Gateway DNP3.0 Client/Server

Protocol and eNode Designer configuration

eNode Configuration Manual

V1.3 September 29th, 2017

This PDF Document contains internal hyperlinks for ease of navigation. For example, click on any item listed in the <u>Table of Contents</u> to go to that page.

For example, click on any item listed in the <u>Lable of Contents</u> to go to the

- <u>Configuration Guide</u>
- Interoperability

Published by:

Atop Technologies, Inc.

2F, No. 146, Sec. 1, Tung-Hsing Rd, 30261 Chupei City, Hsinchu County Taiwan, R.O.C.

Tel: +886-3-550-8137 Fax: +886-3-550-8131 www.atoponline.com www.atop.com.tw

Important Announcement

The information contained in this document is the property of Atop technologies, Inc., and is supplied for the sole purpose of operation and maintenance of Atop Technologies, Inc., products.

No part of this publication is to be used for any other purposes, and it is not to be reproduced, copied, disclosed, transmitted, stored in a retrieval system, or translated into any human or computer language, in any form, by any means, in whole or in part, without the prior explicit written consent of Atop Technologies, Inc.,

Offenders will be held liable for damages and prosecution.

All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

Disclaimer

We have checked the contents of this manual for agreement with the hardware and the software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual is reviewed regularly and any necessary corrections will be included in subsequent editions.

Suggestions for improvement are welcome.

All other product's names referenced herein are registered trademarks of their respective companies.

Documentation Control

Author:	Matteo Tabarelli (Atop)
Revision:	1.3 Enhancement
Revision History:	eNode upgrade
Creation Date:	12 April 2017
Last Revision Date:	29 September 2017
Product Reference:	PG59XX Protocol Gateways
Document Status:	Released

Table of Contents

1	Intr	oduction	6
	1.1	Scope	6
		Document Reference	
	1.3	List of Abbreviations	6
2	Ger	neral Description	8
	2.1	Configuration Theory	8
		General Screen Description	
3	DN	P3 Configuration Guide	. 10
	3.1	Adding the Module in eNode Designer	10
		Server IED Properties	
		3.2.1.1 Slave Address	
		3.2.1.2 IP Address	
		3.2.1.3 IP Port	
	3.3	Client Configuration	
		3.3.1 Client Settings	
		3.3.1.1 Master Address	
		3.3.1.2 Communication Medium	
		3.3.1.3 Link Layer Timeout (ms)	
		3.3.1.4 Application Layer Timeout (ms)3.3.1.5 Class 0, 1, 2, 3 Poll Interval (ms)	
		3.3.1.6 Class 0 Poll Interval (ms)	
		3.3.2 Adding Data Points	
		3.3.3 Servers (Remote IEDs)	
	3.4	Server Configuration	
		3.4.1 Server Settings	
		3.4.1.1 Master Address	
		3.4.1.2 Communication Medium	
		3.4.1.3 Enable Self Address	
		3.4.1.4 Class {X} Event Buffer Size	
		3.4.1.5 Class {X} Event Buffer Overflow Percentage	
		3.4.1.6 Default Static Variations	
		3.4.1.7 Default Event Variations	
		3.4.2 Adding Data Point References	
	о F	3.4.3 Contiguous Server Object Addresses	
	3.5	Miscellaneous Common	
		3.5.1 Incomplete, Conflicting and not needed Information3.5.2 Modify Selected Points Window	
Δ	Cor	mmunication Port Properties	
Э	USI	ng Auto-increment Counters	. 25
		Automatic Increments in Constant Values	
6	Ref	ference Guide	. 27
		Table Buttons	
	6.2	Table Columns	
		6.2.1.1 Tag	
		6.2.1.2 Description	
		6.2.1.3 Groups	28

DNP3

	6.2.1.4	Start Address	
	6.2.1.5	Class	
	6.2.1.6	Analog Input Deadband	
	6.2.1.7	Command Type	
	6.2.1.8	SBO Timeout (ms)	
	6.3 DNP3's Rela	ated ADH Types	
7	DNP3 Client P	Properties	30
	7.1 Device Profi	ile	
	7.2 Implementa	ition Table	
8	DNP3 Server F	Properties	
		ile	
	8.2 Implementa	ition Table	52

Table of Figures

Figure 2-1 - Example Screen	9
Figure 3-1 - Adding the module in eNode Designer	10
Figure 3-2 - Client settings panel (serial above; Ethernet below).	
Figure 3-3 - Client IED panel (Ethernet).	
Figure 3-4 - Add data points window.	
Figure 3-5 - Multiple connected servers example.	15
Figure 3-6 - Add a connected server.	
Figure 3-7 – Remove a connected server.	16
Figure 3-8 - Server settings panel extract.	16
Figure 3-9 – Server IED panel.	17
Figure 3-10 - Add new references window	
Figure 3-11 - Data point references added.	20
Figure 3-12 - Changing the address of a server point.	21
Figure 3-13 - Modify data points window example.	
Figure 4-1 - Serial port properties.	
Figure 4-2 – Ethernet port properties	

1 Introduction

Thank you for Buying Atop's Protocol Gateway.

The product is bundled with the following three user manuals:

- 1) Hardware specific installation user manual, **not covered in this document.** It covers Atop's hardware installation procedure, wiring, power connection etc.
- 2) Getting started with Atop's Protocol Gateway user manual configuration tool introduction, web configuration, software architecture introduction not covered in this document. This manual covers the introduction, installation, network set-up maintenance and using of the configuration tool software, including the procedure to be followed for uploading new configurations to Atop's device.
- 3) Protocol specific user manual (This Manual). One protocol-specific manual will be provided for each protocol installed on the device. This manual covers:
 - a. Basic device network configuration
 - b. Step-by-step protocol set-up for in eNode designer
 - c. Description of the protocol-specific software features, the device profile and the implementation table of supported functionalities.

This manual is for **DNP3.0 Client/Server** and describes how to use the **DNP3 eNode Designer Module** to configure Atop's *DNP3 ADH Application* within the eNode Designer configuration tool.

1.1 Scope

This document is divided into 3 major sections:

- General Description;
- Configuration Guide; and
- DNP3.0 Client Device Profile/ Implementation table and
- DNP3.0 Server Device Profile/ Implementation table

1.2 Document Reference

- [1] Document Title: Getting started User Manual: 197-0100 Revision: Version 1.00 or higher
- [2] Document Title: DNP3 Specification Revision: 2.05, 24th June 2009

1.3 *List of Abbreviations*

ADH	= Application Data Hub
DNP3	= Distributed Network Protocol 3
IED	= Intelligent Electronic Device

IP = Internet Protocol

TCP = Transmission Control Protocol

2 General Description

The DNP3 eNode Module can be used to configure the DNP3 ADH Application as a master or slave. For naming consistency across eNode Designer, the master is called a client, and the slave is called a server.

The DNP3 client can communicate with many DNP3 server IEDs, all of whose data point details can be configured using this module.

2.1 Configuration Theory

Most configuration properties describe a *server*. When configuring the ADH application server, you are configuring the properties of the server itself. Atop protocol gateway supports one server application per protocol per port. When configuring the ADH Application client, you are describing the properties of all the remote servers with whom the client is communicating.

Configuring the protocol-specific information, such as object addresses, is handled in the module. This is explained in this document.

Communication port properties (such as Baud Rate) are configured on the communication port itself. The Device module handles the communication port properties, so heavy details are outside the scope of this document. Port configuration instructions are provide in the eNode Designer general user manual.

Screenshots of the typical configuration method are anyway shown in section 4. The relevant properties of the communication ports automatically apply to the application. For example, in a DNP3 server application, the IP Address the application binds to is taken from the parent Ethernet port.

2.2 General Screen Description

A small configuration example is shown below to better help describe the layout of the screen.

<mark>と</mark> eNode Design		—		×
File Edit View Tools Settings Project	Help 1 SETTINGS IED [2] × + Slave Address: 2 - Link Layer Timeout: 1000 - Class123 poll Interval: 1000 - Class0123 poll Interval: 1000 -	s: 192 . 168	2	
COM1	Data Reference Description Group Data Type Starting Index Addr Range 3	Class	CmdT	ýpe
	Commands Reference Description Group Data Type Starting Index Ad Range Class 4 4 Add Delete Modify Selected Points Move Up Move Down	CmdTy	Select Be	fore Op

Figure 2-1 - Example Screen

1) Tabs –

Server IED Properties – Describes the protocol-specific properties of the server IED.

Data Table and buttons – Shows all (information) data associated with the IED, and shows the buttons to be used to modify them.

Commands Table and buttons – Shows all commands associated with the IED, and shows the buttons to be used to modify them.

The user is able edit contents of the data and commands tables freely.

3 DNP3 Configuration Guide

3.1 Adding the Module in eNode Designer

The DNP3 module can be added to both Ethernet and Serial ports.

The application can be set up as a Client or a Server. The choice will be presented when adding it to the project.

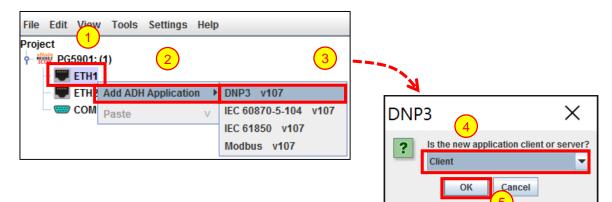


Figure 3-1 - Adding the module in eNode Designer.

- Right click the desired communication port.
- 2) Open the Add ADH Application menu.
- 3 Select DNP3.
- 4 Select **Client** or **Server** from the drop-down menu.
- 5 Click OK.

3.2 Server IED Properties

The server IED properties are at the top of the module screen. The options available are limited to what is relevant for the communication port.

Serial port example:		
	Slave Address: 2	
Ethernet port example:		
Slave Address: 2	IP Address: 192 . 168 . 1 . 123	IP Port: 20000 -

Each property is described in detail below.

3.2.1.1 Slave Address

Description	The slave address of the server IED. For servers it describes its own slave address. For clients, it describes the slave address of the remote server.
Data Entry	Integer
Range	0 to 65519
Input Option	Mandatory

3.2.1.2 IP Address

Ethernet and Client only

Description	The IP Address of the remote server IED. This option is only available in clients, since in servers, the IP Address is taken from the Ethernet port.
Data Entry	IP Address String
Range	Valid IPv4 Addresses (0.0.0.0 to 255.255.255.255)
Input Option	Mandatory

3.2.1.3 IP Port

Ethernet only

Description	The IP Port used by the server IED.
Data Entry	Integer
Range	1 <i>to 65535. Default: 20000</i>
Input Option	Mandatory

3.3 *Client Configuration*

Adding a DNP3 client application will immediately show the following figure. The first tab shows the settings that apply to the whole client application. Each tab after this shows a single DNP3 server with which the client is communicating. Each tab is named "IED [$\{X\}$: $\{Y\}$]" where $\{X\}$ is the master address and $\{Y\}$ is the slave address.

SETTINGS IED [1] × +	
Protocol Settings Master Address	2
Communications Medium	Serial 💌
SETTINGS IED [1] × +	
SETTINGS IED [1] × + Protocol Settings	
	1

Figure 3-2 - Client settings panel (serial above; Ethernet below).

Selecting the IED tab will show the following view.

SETTINGS IED[1] × +							
Slave Address: [Class123 poll Inte		ayer Timeout: Class0123 pol		ation Layer Timeout:		P Address: 192 Port: 20000	. 168 . 1 . 1
Data							
Refernece	Description	Group	Data Type	Starting Index Add	Range	Class	CmdType
Commands							
Refernece	Description	Group	Data Type Startin	g Index A Range	Class	CmdType	Select Before O
Add Dele	ete Modify S	elected Points	Move Up	Move Down			

Figure 3-3 - Client IED panel (Ethernet).

Here the "Add" and "+1" button can be used to add data points. Adding data points is explained in the "Add data points" section, and the other buttons are described in section 6: Reference Guide.

3.3.1 Client Settings

3.3.1.1 Master Address

Description	The DNP3 master address to use.
Data Entry	Integer
Range	0 to 65519
Input Option	Mandatory

3.3.1.2 Communication Medium

Description	The communication method to use. The list will automatically be restricted based on the parent's port type.
Data Entry	Drop down menu
Options	Serial, TCP, UDP
Input Option	Mandatory

3.3.1.3 Link Layer Timeout (ms)

Description	Description The timeout for a data link layer confirmation in milliseconds.	
Data Entry	Integer	
Range	100 to 65535	
Input Option	Mandatory	

3.3.1.4 Application Layer Timeout (ms)

Description	Description The timeout for an application layer confirmation in milliseconds.	
Data Entry	Integer	
Range	2000 to 65535	
Input Option	Mandatory	

3.3.1.5 Class 0, 1, 2, 3 Poll Interval (ms)

Description	The poll interval of the event classes: class 1, 2 and 3. Every poll interval all event classes are polled. Measured in milliseconds.
Data Entry	Integer
Range	1000 to 65535
Input Option	Mandatory

3.3.1.6 Class 0 Poll Interval (ms)

Description	The poll interval of static data. Every poll interval all data is polled for its present value: class 0, 1, 2 and 3. Measured in milliseconds.
Data Entry	Integer
Range	1000 to 65535
Input Option	Mandatory

3.3.2 Adding Data Points

To add data points, left click the "**Add**" button beneath the tables in the main view. Doing so will show the following window. The window is used to add many data points at once with the specified values. For details on the meaning of each column, see section 6.2.

	New Data						
Preview							
Reference	Description	Group	Data Type	Starting Index Ad	Range	Class	CmdType
Enable Input1	Description of it	BINARY_INPUT	Double Point	0	1	ASSIGN_TO_CL	. STATUS_ONLY
Enable Input2	Description of it	BINARY_INPUT	Double Point	0	1	ASSIGN_TO_CL	. STATUS_ONLY
Enable Input3	Description of it	BINARY_INPUT	Double Point	0	1	ASSIGN_TO_CL	. STATUS_ONLY
Enable Input4	Description of it	BINARY_INPUT	Double Point	0	1	ASSIGN_TO_CL	STATUS_ONLY
Enable Input5	Description of it	BINARY_INPUT	Double Point	0	1	ASSIGN_TO_CL	STATUS_ONLY
Enable Input6	Description of it	BINARY_INPUT	Double Point	0	1	ASSIGN_TO_CL	STATUS_ONLY
New values				2			
	Description	Group	Data Type	2 Starting Index Ad	Peange	Class	CmdType
Reference	Description Description of it	Group	Data Type Double Point	2 Starting Index Ad 0	Reange 1	Class ASSIGN_TO_CL	CmdType .STATUS_ONLY
	Description of it				Reange 1		

Figure 3-4 - Add data points window.

Preview Area – Shows the preview of the data points that will be added.

New values – This area is used to enter values. Tag, description, Starting Index Address and Range use manual data entry (click the box and type new values). Data type, CmdType and class use drop-down menus. Entering an integer into the address column will start at that number and automatically increment in each successive point.

Number of rows – This counter can be used to add many data points at once.

Automatic Counters – These counters can be used to add many data points at once. The starting values and step values can be changed in this area. See also 5 Using Auto-increment Counters.

OK button – to accept the new data points.

1

5

DNP3

3.3.3 Servers (Remote IEDs)

Each slave IED is represented by a single tab and a tree node in the eNode Designer project tree.

File Edit View Tools Settings Help						
Project	SETTINGS IED [1:	2] × IED [16] ×	IED [17] ×	+		
• ₩₩ PG5901: (1)						
	Slave Address: Class123 poll Inter		yer Timeout: Class0123 poll In	1000 <u>*</u> terval:		
COM1	Refernece	Description	Group	Data		
	Enable Input1	Description of ite	BINARY_INPUT	Double Po		
	Enable Input2	Description of ite	BINARY_INPUT	Double Po		
	Enable Input3	Description of ite	BINARY_INPUT	Double Po		
	Enable Input4	Description of ite	BINARY_INPUT	Double Po		
	Enable Input5	Description of ite	BINARY_INPUT	Double Po		
	Enable Input6	Description of ite	BINARY_INPUT	Double Po		

Figure 3-5 - Multiple connected servers example.

To modify the connected IEDs list follow the instructions below:

To **add** a new remote IED, click the **"+"** tab at the end of the existing remote server(s) list.

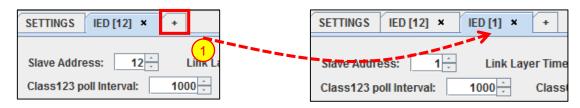


Figure 3-6 - Add a connected server.

To remove a remote IED, it must have no data points specified. If there are data points in the table and you still wish to remove the IED, you will have to remove such data points first.

To remove a remote IED, click the cross on the right side of the tab of the IED you wish to remove.



Figure 3-7 – Remove a connected server.

3.4 Server Configuration

A DNP3 server application outputs data from the ADH database, receives commands and passes them into the ADH system to command another application to perform the operation. Therefore, all server operations use data point *references* to already existing data points that have been created by other application clients or client-servers. Atop protocol gateway supports one server application per protocol per device.

The options describe the local server itself. The settings tab can be used to set the local settings, while the IED tab is similar to the client-tab. Example figures of both are shown below.

Settings Server	
Protocol Settings	
Name	
Slave Address	1
Master Address	2
Enable Self Address	r
Command Timeout (ms)	3000 *
Class 1 Event Buffer Size	1000 -
Class 1 Event Buffer Overflow Percentage	90 -
Class 2 Event Buffer Size	1000 -
Class 2 Event Buffer Overflow Percentage	90 -
Class 3 Event Buffer Size	1000 -
Class 3 Event Buffer Overflow Percentage	90 -
Time Sync Interval (s)	300 -
Ethernet Port Number	20000 *

Figure 3-8 - Server settings panel extract.

Sett	ings	Server						
Data	Data							
	Re	ference	Description	Group	Range	Class	CmdType	Analog Input Deadband
Con	nmand	Is						
	Re	ference	Description	Group	Range	Class	CmdType	Select Before Operate
4	Add Re	ference	Delete Modify	Selected Points	Move Up Move	Down		

Figure 3-9 – Server IED panel.

Here the "Add Reference" button can be used to add data point references. The procedure is explained fully in the next section. The other buttons are described in section 6: Reference Guide.

3.4.1 Server Settings

All server settings are explained in the headings below.

3.4.1.1 Master Address

Description The DNP3 master address to be used.	
Data Entry	Integer
Range	0 to 65519
Input Option	Mandatory

3.4.1.2 Communication Medium

Description	The communication method to be used. The list will automatically be restricted based on the parent's port type.
Data Entry	Drop down menu
Options	Serial, TCP, UDP
Input Option	Mandatory

3.4.1.3 Enable Self Address

Description	The value of the "Enable self address" flag when the application is started. When unsolicited responses are enabled, unsolicited messages may be sent.
-------------	---

Data Entry	Check box
Range	🗆 disabled, 🗹 enabled
Input Option	Mandatory

3.4.1.4 Class {X} Event Buffer Size

Description	The buffer size for class {X}: the maximum number of events to store.
Data Entry	Integer
Range	10 - 65535
Input Option	Mandatory

3.4.1.5 Class {X} Event Buffer Overflow Percentage

Description	If the buffer for class {X} fills to this percentage, a buffer overflow event is sent to the master station.
Data Entry	Integer
Range	25-100. Recommended 50-95. Default: 90
Input Option	Mandatory

3.4.1.6 Default Static Variations

Description	The default static va	ariations.
Data Entry	Drop down menus	
	Binary Input	Packed, With Flags
	Double Input	Packed, With Flags
	Counter Input	32bit with flags, 16 bit with flags, 32 bit (no flags), 16 bit (no flags)
	Counter Input Frozen	32bit with flags, 16 bit with flags, 32bit with flags and time, 16bit with flags and time, 32 bit (no flags), 16 bit (no flags)
Options	Analog Input	32bit with flags, 16 bit with flags, 32 bit (no flags), 16 bit (no flags), Single precision with flags and time
	Analog Input Deadband	16bit, 32bit, Single precision float
	Binary Output	Packed, With Flags
	Analog Output	32bit with flags, 16bit with flags, Single precision with flags
Input Option	Mandatory	

3.4.1.7 Default Event Variations

Description	The default event variations.
Data Entry	Drop down menus

	Binary Input	No Time, Absolute Time, Relative Time
	Double Input	No Time, Absolute Time, Relative Time
Options	Counter Input	32bit with flags, 16bit with flags, 32bit with flags and time, 16bit with flags and time
optione	Counter Input Frozen	32bit with flags, 16bit with flags, 32bit with flags and time, 16bit with flags and time
	Analog Input	32bit (no time), 16bit (no time), 32bit with time, 16bit with time, Single precision (no time), Single precision with time

DNP3

3.4.2 Adding Data Point References

To add new data point references, left click the "Add Reference" button underneath the tables in the main view. This will bring up the Add References window defined by the eNode Designer main application. It should appear similar to the following figure. Here we are adding references to data points created by an IEC 60870-5-104 client.

Project 9- 1000 PG5901: (1)	Add nev	v referen	ces to which points?				
	#	Map	Application	Tag	Exchange Type	Data Type	Map Count
- 💏 IEC 60870-5-104 (C)	2	~	PG5901: (1) / ETH1 / IEC 60870-5-104 (C): (1)	Example DI1	Data	Single Point	
- IED [1: 1]	3	~	PG5901: (1) / ETH1 / IEC 60870-5-104 (C): (1)	Example DI2	Data	Single Point	
	4		PG5901: (1) / ETH1 / IEC 60870-5-104 (C): (1)	Example DI3	Data	Single Point	
- 10 DNP3 (S): (1)	5		PG5901: (1) / ETH1 / IEC 60870-5-104 (C): (1)	Example DI4	Data	Single Point	
	6	~	PG5901: (1) / ETH1 / IEC 60870-5-104 (C): (1)	Example output 1	Command (Single Stage)	Single Point	
- 📟 COM1	7		PG5901: (1) / ETH1 / IEC 60870-5-104 (C): (1)	Example output 2	Command (Single Stage)	Single Point	
	8		P 1: (1) / ETH1 / IEC 60870-5-104 (C): (1)	Example output 3	Command (Single Stage)	Single Point	
	9	V	A 1): (1) / ETH1 / IEC 60870-5-104 (C): (1)	Example output 4	Command (Single Stage)	Single Point	
	Mapped	l with:					
	#	:		Application			Map Count
						,	
						2 ок	Cancel

Figure 3-10 - Add new references window.

Select Data Points – Adding a reference to a point creates a "mapping" to that point. Select which data points the server application is interested in using.

Left Click **OK** when done to accept the new references.

The data points that will appear in the list and that will be available for mapping are those whose data point type is compatible with the DNP3 application. For the table matching DNP3 data types to ADH types, see section 0.

Reference	Description	Group	Range	Class	CmdType	Analog Input Deadband
Example DI1	Digital input 001	BINARY_INPUT	1	NO_CLASS_ASSIGN	STATUS_ONLY	0
Example DI2	Digital input 002	BINARY_INPUT	1	NO_CLASS_ASSIGN	STATUS_ONLY	0
Commands						
Commands Reference	Description	Group	Range	Class	CmdType	Select Before Operate
	Description Output 001	Group BINARY_OUTPUT	Range	Class NO_CLASS_ASSIGN	CmdType DIRECT_OPERATION	Select Before Operate

Figure 3-11 - Data point references added.

All properties will be automatically defined based on the data point's values.

3.4.3 Contiguous Server Object Addresses

The DNP3 protocol specifies that all addresses (indices) must be contiguous for each point type. For example if there are 10 binary inputs, they must have addresses 0 through 9. The data points in server's table are always automatically sorted so that each point type is listed consecutively, with incremental addresses.

For this reason, in the server, the **address values cannot be changed directly** by the user. In order to change the address of a data point, use the "Move Up" and "Move Down" buttons within the point group (Binary Input / ...).

Tag	Description	Data Type	Address	Class	Analog Input Deadba.
Example DI0	Example DI0	Binary Input	0	Class 1	
Example DI1	Example DI1	Binary Input	1	Class 1	
Example DI2	Example DI2	Binary Input	2	Class 1	
Example DI3	Example DI3	Binary Input	3	Class 1	
Example DPI 1	Example DPI 1	Double Input	0	Class 1	
Example DPI 2	Example DPI 2	Double Input	1	Class 1	
Add Reference	Delete	Modify Selected Points	s Move Up	2 Move Down	

Data					
Tag	Description	Data Type	Address	Class	Analog Input Deadba
Example DI0	Example DI0	Binary Input	0	Class 1	
Example DI1	Example DI1	Binary Input	1	Class 1	
Example DI3	Example DI3	Binary Input	2	Class 1	
Example DI2	Example DI2	Binary Input	3	Class 1	
Example DPI 1	Example DPI 1	Double Input	0	Class 1	
Example DPI 2	Example DPI 2	Double Input	1	Class 1	

Figure 3-12 - Changing the address of a server point.

- Select the row(s) you want to move (to change the address).
- Use the **Move Up** or **Move Down** button. In this example, *Move Down* was used.
- 3 The row(s) have been moved and the addresses have been updated accordingly. Note that this is the same data point as selected in (1).

Move down will by nature increase addresses by one (per click) and move up will reduce addresses by one (per click).

3.5 Miscellaneous Common

3.5.1 Incomplete, Conflicting and not needed Information

Incomplete or conflicting information is shown in red, and will cause warning symbols on the tab and in the project tree. Hovering over the warning icons will show further details about the cause of the warning. This allows the user to quickly fix invalid information.

🔥 eNode Designe	r											Х
File Edit View Tools Settings	Help											
Project	SETTINGS 🚹 IED ['	I] X +										
• EtH1 1	Class123 poll Interval:	1 Link Laye	r Timeout: 1000 Class0123 poll Interv		ayer Timeo.	out: 10		P Addres Port:	ss: 192 . 168	1]	
P ETH2 Address Invalid								2				
COM1	Refernece	Description	Group	Data	Туре	Starting In	dex Address	\sim	Range	Class	CmdTy	pe
	Example_BI_1	Example binary in				0		5	N	_CLASS_ASSIGN		
	Example_BI_2	Example binary in				0		5	N	_CLASS_ASSIGN		
	Example_BI_3	Example binary in						5	N	_CLASS_ASSIGN		
	Example_BI_4	Example binary in						5	N	_CLASS_ASSIGN		
	Example DI_1		put DOUBLE_INPL			6		5	N	CLASS_ASSIGN		
	Example DI_2	Example double in	·			6		5	N	_CLASS_ASSIGN		
	Example_Al_1		nput ANALOG_INPU			2		5	N	_CLASS_ASSIGN		
	Example_AI_2		put ANALOG_INPU			2		0	N	_CLASS_ASSIGN		
	Example_AI_3	Example analog in	nput ANALOG_INPU	T Signed Wor	d	2		0	N	_CLASS_ASSIGN	STATUS_ONL	(
	Commands				L					3		
	Refernece	Description	Group	Data Type	Starting In	ndex Addr	Rang	е	Class	Cmaryer	Select Befo	re Oper
		Direct output	BINARY OUTPUT	Single Point	4		5		NO CLASS ASSIG	DIRECT_OPERAT	ГІ 0	
	Direct BO 2	Direct output	BINARY_OUTPUT	Single Point	4		5			DIRECT_OPERAT		
	Direct_BO_3	Direct output	BINARY_OUTPUT	Single Point	4		5		NO_CLASS_ASSIC	DIRECT_OPER AT	ГІ 0	
	SBO_1	Select before oper	BINARY_OUTPUT	Single Point	3		3		NO_CLASS_ASSIG	S SELECT_BEFOR	E 400	
												_
	Add Delete	Modify Sele	cted Points	Nove Up Move	Down							

- Mouse-over a warning to show a tooltip explaining the warning.
- **Invalid Data** shows in red. The darker red means the data is invalid, and the lighter red means there is an address conflict.
- **Unneeded Data** is hidden and not editable. For example, the SBO timeout is not required for direct operates. These table cells have a grey background, no contents and cannot be edited.

1

3

DNP3

3.5.2 Modify Selected Points Window

The "Modify Selected Points" window is used to change many row properties in one single step.

Select the data points you want to change, and then press the "Modify Selected Points" button beneath the tables. It will generate the following window.

💦 Renar	ne Points			(1)			×
Original							
Reference	Description	Group	Data Type	Starting Index Add	Range	Class	CmdType
Example_BI_2	Example binary in	. BINARY_INPUT	Single Point	0	5	NO_CLASS_ASSI	STATUS_ONLY
Example_BI_3	Example binary in	. BINARY_INPUT	Single Point		5	NO_CLASS_ASSI	
Example_BI_4	Example binary in	. BINARY_INPUT	Single Point		5	NO_CLASS_ASSI	STATUS_ONLY
Example DI_1	Example double i	DOUBLE_INPUT	Signed Word	6	5	NO_CLASS_ASSI	STATUS_ONLY
Preview				2		0	0.7
Reference	Description	Group	Data Type	Starting Index Addr	ress Range	Class	CmdType
Example_BI_2	Example binary i	_	Double Point	5	1	NO_CLASS_AS	STATUS_ONLY
Example_BI_3	Example binary i E	-	Double Point	5	1	NO_CLASS_AS	STATUS_ONLY
Example_BI_4	Example binary i E	_	Double Point	5	1	NO_CLASS_AS	
Example DI_1	Example double E	DINART_INFUT	Double Point	5	1	NO_CLASS_AS	STATUS_ONLY
New values							
Tag	Description	Group	Data Type	tarting Index Addr	ress Range	Class	CmdType
[N]	[N] E	BINARY_INPUT	Double Point	5	1	NO_CLASS_AS	STATUS_ONLY
You can use "[N]" as the original text of the cell. You can use "[X]" as an auto increment counter. [Y] and [Z] work similarly. Use [3X] to produce values like "001". Define counter [X] Define counter [X] Start At: 1 + Step by: 1 + Start At: 1 + Step by: 1 + Start At: 1 + Step by: 1 +							
					5	ОК Car	icel

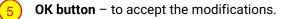
Figure 3-13 - Modify data points window example.

1) Original data table – Shows the original data table.

Preview – Shows the new data table that will be used if the modifications are accepted. These fields update automatically according to the contents of (3).

New values – Contain the new values for the table cells. "[N]" can be used to maintain the original value of the cell, and the auto-incrementing counters [X], [Y] and [Z] can be used to add numbers. For details, see Using Auto-increment.

Counter properties – Sets the initial values and step amounts of the counters [X], [Y] and [Z].



Data point references always use the *tag* and *description* of the "real" point. Consequently, these values will not be changed by a server application. The object addresses in the server are required to be contiguous, so the addresses cannot be changed using this window either. See server configuration for details.

2

4 Communication Port Properties

The device module handles the communication port properties. The typical method is described briefly below.

File Edit View Tools Settings			- D	×
Project PG5901: (1) Project PG5901: (1) ETH1 ETH2 COM1 PG5901: (1) PG5901:	Operate mode: Baud rate: Parity: Data Bits: Stop Bits:	RS232 ▼ 9600 ▼ None ▼ 8 ▼ 1 ▼	Pre transmission delay (ms): Post transmission delay (ms): Inter-character delay: Message timeout (ms): Message retries: Character timeout (ms): Character retries:	$ \begin{array}{c} 0 \\ $

Figure 4-1 - Serial port properties.

Select the communication port in the project tree – This will typically cause the central panel to show the port's properties.

Properties – The communication port's properties can be set.

Similarly, the Ethernet properties are shown below.

💦 eNode Design	_	×		
File Edit View Tools Settings	Help			
Project 	IP Address: Subnet mask: Gateway:	192 168 1 2 255 255 255 0 192 168 1 1		

Figure 4-2 – Ethernet port properties.

5 Using Auto-increment Counters

The following is a full example showing how auto-increment works. The example given shows the IEC 60870-5-104 window, however the DNP3 auto-increment works in the same way.

	Add Poir	its				^
	Preview					
	Tag	Description	Data Type	Time Starr	IDA	COT
	example_tag_0	example_desc_0000	Single Point	CP24Time2a	1	General Interrogation
	example_tag_2	example_desc_0010	Single Point	CP24Time2a	2	General Interrogation
	example_tag_4	example_desc_0020	Single Point	CP24Time2a	3	General Interrogation
	example_tag_6	example_desc_0030	Single Point	CP24Time2a	4	General Interrogation
	example_tag_9	example_desc_0040	Single Point	CP24Time2a	5	General Interrogation
1	\sim \mathbf{T}					
	2 w values:	Description	Data Type	Time Starr		СОТ
	Z)w values: Tag example_tag_[X]	Description example_desc_[4Y]	Data Type ingle Point	Time Stam CP24Time2a		COT General Interrogation

Figure 5-1 – Using Auto Increment when adding Data Points or Commands.

The *Number of Rows* can be modified to set the number of data points or commands created from the *New values* section. As shown in the example above, five data points/commands are created and shown in the preview section as the *Number of Rows* is set to 5.

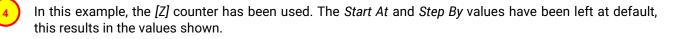
When using the auto increment counters by default, they will start at one and increment by one. Anyway autoincrement value has its own section for configuration. Adjusting *Start At* will change the value that the first data point/command receives. Adjusting *Step By* will change the value that the second and subsequent values will be incremented by.



In this example, the [X] counter is used. The *Start At* value has been set to 0 and the *Step By* value has been set to 2. This results in the values seen in the preview section.

It is also possible to include a number within the square brackets and before the X, Y or Z while using auto increment. This will produce values that contain the entered number of digits. Any digit that is not taken up by the value determined by the *Start At* and *Step By* values will be shown as zeros.

In this example, the [Y] counter has been used with the integer 4 to indicate the number structure. This results in the values shown in the preview section.



If no auto increment value is entered in any field, each data point/command field value will be created the same with the exception of *Tag* and *IOA*. The first new data point/command's *Tag* value will represent what was entered in the *New value* section. However, the subsequent data points/commands will contain the initial *Tag* value followed by an underscore and a number incrementing by one from 1 onwards. (Example: tag, tag_1, tag_2 etc.). This is an artefact of eNode Designer ensuring all data point tag names are unique.

5.1 Automatic Increments in Constant Values

In the DNP3 eNode module, the following fields will be automatically increased by one for each row, even if a constant value is entered in the "New value" field.

Address

6 Reference Guide

6.1 Table Buttons

Client Options: Add +1 Delete Server Options:	Modify Selected Points Move Up Move Down Sort
	elete Modify Selected Points Move Up Move Down
Add +1 Add Reference Delete Modify Selected Points Move Up Move Down Sort	Adds new data points in the client. See section 3.3.2. Adds a single new data point in the client. See section 3.3.2. Adds a new data point reference in the server. See section 3.4.2. Deletes the selected data points Modify the properties of the selected data points. See section 3.5.2. Moves the selected data points up one row in the table Moves the selected data points down one row in the table Sorts the table: groups by data type, then by address order.

6.2 Table Columns

6.2.1.1 Tag

Description	A unique Tag name for each data point.
Data Entry	String
Min Length	1
Max Length	N/A
Input Option	Mandatory

6.2.1.2 Description

Description	User defined description for each data point.
Data Entry	String
Min Length	1
Max Length	N/A
Input Option	Mandatory

6.2.1.3 Groups

Description	The DNP3 point type.
Data Entry	Drop Down Menu
Groups	Binary Input, Double Input, Counter Input, Analog Input, Binary Output, Analog Output
Input Option	Mandatory

6.2.1.4 Start Address

Description	The DNP3 starting <i>index</i> in the DNP3 point type.
Data Entry	Integer
Start Address	n/a
Count	0 to 65535 (the max Address starting from "start addr"+"count" can't exceed 65535)
Input Option	Mandatory

6.2.1.5 Class

Server Only

Description	The DNP3 class. Using <i>No class</i> means that no events are raised for the data point.
Data Entry	Drop down menu
Options	No class, Class 1, Class 2, Class 3
Input Option	Mandatory

6.2.1.6 Analog Input Deadband

Server Data Only

Description	The dead-band of the Analog input. An event is triggered when the Analog input changes an amount greater than the deadband value.
Data Entry	Floating point number
Range	0.0 or greater
Input Option	Mandatory for Analog inputs

6.2.1.7 Command Type

Description	The command security level: direct or SBO.
Data Entry	Drop down menu
RangeDirect Operate, Select Before Operate	
Input Option	Mandatory for commands

6.2.1.8 SBO Timeout (ms)

Server commands only

Description	The timeout to wait for a SBO (Select before operate) to complete in milliseconds.
Data Entry	Integer
Range	0 to 65000
Input Option	Mandatory if command type is Select Before Operate

6.3 DNP3's Related ADH Types

The DNP3 data types correspond to the ADH types given in the table below.

DNP3 Point Type	ADH Data Type	ADH Exchange Type
Binary Input	Single Point	Data
Double Input	Double Point	Data
Counter Input	Unsigned 32	Data
Analog Input	Float 32	Data
Binary Output	Single Point	Command
Analog Output	Float 32	Command

Table 6-1 – DNP3 data types relation to ADH data point types.

7 DNP3 Client Properties

7.1 Device Profile

Vendor Name: Atop Technologies, Inc.	
Device Name: PG59XX Series DNP3 Server o	ver Ethernet or Serial
Device Function: ■ Master □ Outstation	
DNP Levels Supported for: Request and Response None Level 1 Level 2 Level 3 Level 4	Supported Function Blocks: Self-Address Support Data Sets File Transfer Virtual Terminals Mapping to IEC61850 Object Models defined in a DNP3 XML file Function code 31, activate configuration Authentication
Connections Supported: Serial IP Networking Serial Connections:	
Serial Connection Parameters:	Bit 1 Stop Bit No Parity
	Bit, 1 Stop Bit, No Parity
Serial Connection Parameters: ■ Asynchronous – 8 Data Bits, 1 Start Baud Rate: □ fixed	Bit, 1 Stop Bit, No Parity
Serial Connection Parameters: Asynchronous – 8 Data Bits, 1 Start Baud Rate: fixed Configurable – 110 to 115200 Flow Control: None Hardware flow control	Bit, 1 Stop Bit, No Parity

Receiver Inter-character Timeout:	
■ Not checked	
□ No gap permitted	
□ Fixed a bit times	
□ Fixed a ms	
Inter-character Gap in Transmission:	
■ None	
Maximum bit times	
Maximum ms	
ID Naturaling	
IP Networking: Type of End Point:	
■ TCP Initiating	
■ TCP Listening	
■ UDP datagram	
TCP Listen Port Number:	
□ Fixed at 20000	
■ Configurable, range 1 to 65535 (default 2000)	0)
TCP Keep-alive timer:	
■ Fixed at 19000 ms	
□ Configurable, range to ms	
_ coga.az.o, .ago to tra	
Local UDP Port:	
☐ Fixed at 20000	
■ Configurable, range 1 to 65535 (default 2000)	0)
Multiple Outstation Connections:	
□ Not supported	
• •	1)
Supports multiple outstations (maximum is 64)	•)
Time Synchronization Support:	
□ Not supported	
DNP3 LAN Procedure	
■ DNP3 Write Time	
Data Link Address:	Self-Address Support using address 0xFFFC:
□ Fixed at 292	■ Yes
■ Configurable, range 0 to 65519 (default 2)	□ No
Sends Confirmed User Data Frames:	Data Link Layer Confirmation Timeout:
■ Never	■ None
□ Sometimes, explain	□ Fixed at 2000 ms
	□ Configurable, range to ms
Maximum Data Link Retries:	
Never Retries	
□ Fixed at 3	
□ Configurable, range to	

 Maximum number of octets Transmitted in a Data Link Frame: ■ Fixed at 292 □ Configurable, range to 	Maximum number of octets that can be Received in a Data Link Frame: ■ Fixed at 292 □ Configurable, range to
 Maximum number of octets Transmitted in an Application Layer Fragment: ■ Fixed at 2048 □ Configurable, range to 	Maximum number of octets that can be Received in an Application Layer Fragment: ■ Fixed at 249 □ Configurable, range to
Timeout waiting for Complete Application Layer Fragmer None Fixed at 6000 ms Configurable, range _1000_ to _65535_ ms Control Status Codes Supported:	nt:
 1 - TIMEOUT 2 - NO_SELECT 3 - FORMAT_ERROR 4 - NOT_SUPPORTED 5 - ALREADY_ACTIVE 6 - HARDWARE_ERROR 7 - LOCAL 8 - TOO_MANY_OBJS 9 - NOT_AUTHORIZED 10 - AUTOMATION_INHIBIT 	 11 - PROCESSING_LIMITED 12 - OUT_OF_RANGE 13 - DOWNSTREAM_LOCAL 14 - ALREADY_COMPLETE 15 - BLOCKED 16 - CANCELLED 17 - BLOCKED_OTHER_MASTER 18 - DOWNSTREAM_FAIL 126 - RESERVED 127 - UNDEFINED

DNP3 Client

Properties

7.2 Implementation Table

						, ,
2	2	<u>→</u>	<u>→</u>	<u>→</u>	Obj ect	
	0	2	<u>→</u>	0	Var iati on	
Binary Input Change without Time	Binary Input Change - All Variations (Default variation)	Binary Input with Status	Binary Input - Packed Format	Binary Input - All Variations (Variation 0 is used to request default variation)	Description	OBJECT
	1	-		1(Read), 22(Assign Class)	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
06,07,08	06,07,08	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	Qualifier Codes (hex)	irse)
129, 130 (Unsolicited	129	129	129,	129(Response)	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
17, 28	17, 28	00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	Qualifier Codes (hex)	vse pond with)

4	ω	ω	ω	2	2		Obj ect	
0	2	<u> </u>	0	ω	2		Var iati on	
Double-bit Binary Input Change - All Variations (Default variation)	Double-bit Binary Input	Double-bit Binary Input – Packed Format	Double-bit Binary Input - All Variations (Variation 0 is used to request default variation)	Binary Input Change with Relative Time	Binary Input Change with Time		Description	OBJECT
	4	-	1(Read), 22(Assign Class)				Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
06,07,08	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	06,07,08	06,07,08		Qualifier Codes (hex)	rse)
129	129	129,	129(Response)	129, 130	129, 130	Response)	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
17, 28	00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	17, 28	17, 28		Qualifier Codes (hex)	ond with)

		OBJECT	REQUEST (Library will parse)	r Inse)	RESPONSE (Library will respond with)
Obj ect	Var iati on	Description	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)	
4		Double-bit Binary Input Change without Time	1	06,07,08	
4	2	Double-bit Binary Input Change with Time	-	06,07,08	
4	ω	Double-bit Binary Input Change with Relative Time	1	06,07,08	
10	0	Binary Output - All Variations	1(Read), 22(Assign Class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	top) or all) 1 qty)
10	-	Binary Output	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	op) or all) I qty)
10	N	Binary Output Status	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	stop) , or all) ed qty)

20	20	20	12	12	Obj ect	
2	<u>ــ</u>	0	-	0	Var iati on	
16-Bit Binary Counter	32-Bit Binary Counter	Binary Counter - All Variations	Control Relay Output Block	Control Block - All Variations	Description	OBJECT
, ,	<u>,</u>	1(Read), 22(Assign Class) 7(Immediate Freeze - No Response), 9 (Freeze and Clear), 10 (Freeze and Clear – No Response)	3(Select), 4(Operate), 5 (Direct Operate), 6 (Direct Operate NR)		Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	17, 28		Qualifier Codes (hex)	rse)
129,	129,	129	129		Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	echo of request		Qualifier Codes (hex)	NSE pond with)

г

N	N	N	N		
21	21	20	20	Obj ect	
	0	6	Сл	Var on	
32-Bit Frozen Counter	Frozen Counters - All Variations	16-Bit Binary Counter without Flag	32-Bit Binary Counter without Flag	Description	OBJECT
	1(Read), 22(Assign Class)	<u>~</u>	-	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	Qualifier Codes (hex)	r irse)
129,	129(Response)	129,	129,	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	Qualifier Codes (hex)	VSE pond with)

21	21	21	21		@ O	
					ect	
9	ດ	5	2		Var iati on	
32-Bit Frozen Counter without Flag	16-Bit Frozen Counter with Time of Freeze	32-Bit Frozen Counter with Time of Freeze	16-Bit Frozen Counter		Description	OBJECT
-	<u>→</u>	4	-		Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	17, 28 (index)	Qualifier Codes (hex)	rse)
129,	129,	129,	129,		Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	00, 01 17, 28		Qualifier Codes (hex)	vSE pond with)

		OBJECT	REQUEST (Library will parse)	rse)	RESPONSE (Library will respond with)	ISE vond with)
Obj ect	Var iati on	Description	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)
				07, 08 ,(limited qty) 17, 28 (index)		
21	10	16-Bit Frozen Counter without Flag		00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	129,	00, 01 17, 28
22	0	Counter Change Event - All Variations	1	06,07,08	129	17, 28
22	<u> </u>	32-Bit Counter Change Event without Time	→	06,07,08	129, 130	17, 28
22	2	16-Bit Counter Change Event without Time	_	06,07,08	129, 130	17, 28
22	J	32-Bit Counter Change Event with Time	_	06,07,08	129, 130	17, 28
22	6	16-Bit Counter Change Event with Time	_	06,07,08	129, 130	17, 28
23	0	Frozen Counter Events - All Variations	1	06,07,08	129	17, 28
23	-	32-Bit Frozen Counter Event without Time	1	06,07,08	129, 130	17, 28

30	30	Obj 23 23 ect j 30 23 30					
N		0	6	ъ	N	iati	
16-Bit Analog Input	32-Bit Analog Input	Analog Input - All Variations	16-Bit Frozen Counter Event with Time	32-Bit Frozen Counter Event with Time	16-Bit Frozen Counter Event without Time	nr Description ti	OBJECT
_		1(Read), 22(Assign Class) 7(Immediate Freeze), 8 (Immediate Freeze - No Response), 9 (Freeze and Clear), 10 (Freeze and Clear – No Response)			-	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	06,07,08	06,07,08	06,07,08	Qualifier Codes (hex)	rse)
129,	129,	129, 130 1 129, 130 1 129, 130 1 129(Response) 1 129(Response) 1				Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	17, 28	17, 28	17, 28	Qualifier Codes (hex)	JSE pond with)

				· · · ·	
31	30	30	30	ect Obj	
0	Cī	4	ယ	Var iati on	
Frozen Analog Input - All Variations	Single-precision float –point with flag	16-Bit Analog Input without flag	32-Bit Analog Input without flag	Description	OBJECT
1(Read), 22(Assign Class)	_			(Library will parse) Application Layer Function Codes (Decimal) 07 17	
00, 01 (start-stop)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	Qualifier Codes (hex) 07, 08 ,(limited qty) 17, 28 (index)	
129(Response)	129,	129,	129,	(Library Will respond With) Application Layer Qualifie Function Codes (h (Decimal)	RESPONSE
00, 01,	00, 01, 17, 28	00, 01, 17, 28	00, 01 17, 28	Qualifier Codes (hex)	VSE

31	3	31	3		Obj ect	
4	ω	2	-		Var iati on	
16-Bit Frozen Analog Input with Time of	32-Bit Frozen Analog Input with Time of Freeze	16-Bit Frozen Analog Input	32-Bit Frozen Analog Input		Description	OBJECT
	<u>ب</u>	<u>ــ</u>	<u>ب</u>		Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	Qualifier Codes (hex)	rse)
129,	129,	129,	129,		Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
	00, 01, 17, 28	00, 01, 17, 28	00, 01, 17, 28	17, 28	Qualifier Codes (hex)	VSE pond with)

1					
<u>3</u>	31	31		Obj ect	
7	σ	J		Var iati on	
Single-precision floatpoint with flag	16-Bit Frozen Analog Input without Flag	32-Bit Frozen Analog Input without Flag	Freeze	Description	OBJECT
-	<u>→</u>	-		Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	Qualifier Codes (hex)	rse)
129,	129,	129,		Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01, 17, 28	00, 01, 17, 28	00, 01, 17, 28	00, 01, 17, 28	Qualifier Codes (hex)	VSE pond with)

	06,07,08	32-Bit Frozen Analog Event without Time	1 3
ō	06,07,08	Frozen Analog Event - All Variations	0 F
8	06,07,08	Single-precision float point Analog	7 8
00	06,07,08	Single-precision, float –point Analog Change	5
80	80,70,80	16-Bit Analog Change Event with Time	4 1
80	06,07,08	32-Bit Analog Change Event with Time	ය კ
,08	06,07,08	16-Bit Analog Change Event without Time	2 1
80,	06,07,08	32-Bit Analog Change Event without Time	1 3
80	06,07,08	Analog Change Event - All Variations	0 A
Qualifier Codes (hex)	Application Layer Function Qua Codes (Decimal)	Description	Var iati on
	REQUEST (Library will parse)	OBJECT	-

40	40	33	33	33	33	33	• •	
							ect	
	0	7	ъ	4	З	2	Var iati on	
32-Bit Analog Output Status	Analog Output Status - All Variations	Single-precision_float -point_Frozen Analog Change Event with Time	Single-precision_float -point Frozen Analog Change Event with out Time	16-Bit Frozen Analog Event with Time	32-Bit Frozen Analog Event with Time	16-Bit Frozen Analog Event without Time	Description	OBJECT
	1(Read), 22(Assign Class)	-	-	1	1	-	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	06,07,08	06,07,08	06,07,08	06,07,08	06,07,08	Qualifier Codes (hex)	irse)
129,	129(Response)	129, 130	129, 130	129, 130	129, 130	129, 130	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01, 17, 28	00, 01, 17, 28	17, 28	17, 28	17, 28	17, 28	17, 28	Qualifier Codes (hex)	vSE pond with)

50	41	41	41	40	40		Obj ect	
_	ω	N		ω	N		Var iati on	
Time and Date	Analog Output - Single-precision float -	16-Bit Analog Output Block	32-Bit Analog Output Block	Single-precision float –point Analog Output	16-Bit Analog Output Status		Description	OBJECT
2(Write)	3, 4, 5, 6	3, 4, 5, 6	3, 4, 5, 6				Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
07 (Quantity = 1)	17, 28	17, 28	17, 28	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	17, 28 (index)	Qualifier Codes (hex)	rse)
129	129	129	129	129,	129,		RESPONSE (Library will respond with) Application Layer Function Codes (Decimal)	
07 (quantity = 1)	echo of request	echo of request	echo of request	00, 01, 17, 28	00, 01, 17, 28		Qualifier Codes (hex)	vSE pond with)

	80		60		60		60	60	Obj ect	
	-		4		ω		N		Var iati on	
	Internal Indications		Class 3 Data		Class 2 Data		Class 1 Data	Class 0 Data	Description	OBJECT
2(write)		20 (Enable Unsolicited), 21,(Disable Unsolicited)	_	20 (Enable Unsolicited), 21,(Disable Unsolicited)	_	20 (Enable Unsolicited), 21,(Disable Unsolicited)			Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
index=7	00	06	06,07,08	06	06,07,08	06	06,07,08	06	Qualifier Codes (hex)	rse)
			129		129		129	129	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
									Qualifier Codes (hex)	ISE pond with)

8 DNP3 Server Properties

8.1 Device Profile

DNP3 Device Profile

This document defines the options of the DNP3 protocol used by Atop DNP3 devices and is accompanied by an implementation table.

Atop Technologies, Inc.	
Device Name: PG59XX Series DNP3 Server over Ethernet or S	erial
Device Function: □ Master ■ Outstation	
DNP Levels Supported for: Request and Response None Level 1 Level 2 Level 3 Level 4	Supported Function Blocks: Self-Address Support Data Sets File Transfer Virtual Terminals Mapping to IEC61850 Object Models defined in a DNP3 XML file Function code 31, activate configuration Authentication
Connections Supported: ■ Serial ■ IP Networking	i
Serial Connections: Serial Connection Parameters: ■ Asynchronous – 8 Data Bits, 1 Start Bit	t, 1 Stop Bit, No Parity
Baud Rate: ☐ fixed ■ Configurable – 110 to 115200	
Flow Control: None Hardware flow control Software flow control	
Interval to Request Link Status: ■ Not supported □ Fixed at seconds □ Configurable – 0 to 2147483647	
Supports DNP3 Collision Avoidance: ■ No	

□ Yes, using back-off time = (Min + Random) m	nethod
Receiver Inter-character Timeout:	
 No gap permitted Fixed a bit times 	
\Box Fixed a bit times \Box Fixed a ms	
■ Configurable– 0 to 60000	
Inter-character Gap in Transmission:	
■ None	
□ Maximum bit times	
Maximum ms	
IP Networking:	
Type of End Point:	
TCP Listening	
■ UDP datagram	
TCP Listen Port Number:	
□ Fixed at 20000	
■ Configurable, range 1 to 65535 (default 20000)	
TCP Keep-alive timer:	
■ Fixed at 19000 ms	
□ Configurable, range to ms	
Local UDP Port:	
□ Fixed at 20000	
Configurable, range 1 to 65535 (default 20000)	
Multiple Master Connections:	
□ Not supported	
Supports multiple masters (maximum is 5)	
Time Synchronization Support:	
□ Not supported	
DNP3 LAN Procedure	
■ DNP3 Write Time	
Data Link Address:	Self-Address Support using address 0xFFFC:
□ Fixed at ■ Coofigurable, range 0 to 65510 (default 1)	
Configurable, range 0 to 65519 (default 1)	
Sends Confirmed User Data Frames:	Data Link Layer Confirmation Timeout:
■ Never	■ None
Sometimes, explain	□ Fixed at 2000 ms
□ Always	Configurable, range to ms
Maximum Data Link Retries:	
■ Never Retries	
□ Fixed at 3	
□ Configurable, range to	
Maximum much as for the Tree in the Disc	Maximum mumber of a table to the top of the top of the
Maximum number of octets Transmitted in a Data	Maximum number of octets that can be Received in a
Link Frame:	Data Link Frame:
■ Fixed at 292	■ Fixed at 292
Configurable, range to	□ Configurable, range to

Maximum number of octets Transmitted in an Application Layer Fragment: Fixed at 2048 Configurable, range to	Maximum number of octets that can be Received in an Application Layer Fragment: Fixed at 249 Configurable, range to
Timeout waiting for Complete Application Layer Fragm None Fixed at 6000 ms Configurable, range to ms	ent:
Timeout waiting for Application Confirm of solicited re ■ None □ Fixed at 6000 ms □ Configurable, range 0 to 2147483647 ms (defau	
Requests Application Confirmation for event response Yes No Configurable	and non-final fragments:
Sends Multi-Fragment Responses: ■ Yes □ No	Last Fragment Confirmation: ☐ Always ■ Sometimes, Only when it contains events ☐ Never
Maximum number of objects allowed in a single contro ■ <i>Fixed at 16</i> □ <i>Configurable, range to</i>	I request for CROB (group 12):
Maximum number of objects allowed in a single control Fixed at 16 Configurable, range to 	I request for Analog Outputs (group 41):
Control Status Codes Supported:	
 1 – TIMEOUT 2 – NO_SELECT 3 – FORMAT_ERROR 4 – NOT_SUPPORTED 5 – ALREADY_ACTIVE 6 – HARDWARE_ERROR 7 – LOCAL 8 – TOO_MANY_OBJS 9 – NOT_AUTHORIZED 10 – AUTOMATION_INHIBIT 	 11 - PROCESSING_LIMITED 12 - OUT_OF_RANGE 13 - DOWNSTREAM_LOCAL 14 - ALREADY_COMPLETE 15 - BLOCKED 16 - CANCELLED 17 - BLOCKED_OTHER_MASTER 18 - DOWNSTREAM_FAIL 126 - RESERVED 127 - UNDEFINED
Supports Unsolicited Reporting: Not Supported Configurable, selectable from On and Off	
Unsolicited Response Confirmation Timeout: □ Fixed at ms □ Configurable, range 0 to 4294967295 ms	Number of Unsolicited Retries: □ Fixed at 5 □ Configurable, range to
Event Buffer Overflow Behavior: Discard the oldest event Discard the newest event	

□ Other, explain
Event Buffer Organization:
Per Object
Per Class
Class 1:
Class 2:
Class 3:
□ Single Buffer
□ Fixed at
□ Configurable, range to
Outstation Unselicited Deserves Trigger Conditions:
Outstation Unsolicited Response Trigger Conditions:
(Number of events)
class 1: ■ Not used to trigger Unsolicited Response □ Fixed at □ Configurable
class 2: ■ Not used to trigger Unsolicited Response □ Fixed at □ Configurable
class 3: ■ Not used to trigger Unsolicited Response □ Fixed at □ Configurable

8.2 Implementation Table

0	Obj Var ect iati on		1 0 (Variation 0 is used to request default variation)		
OBJECT	Description	Binary Input - All Variations (Variation 0 is used to		Binary Input - Packed Format	Binary Input - Packed Format Binary Input with Status
REQUEST (Library will parse)	Application Layer Function Codes (Decimal)	1(Read), 22(Assign Class)			
r ırse)	Qualifier Codes (hex)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)		00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtX</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>qty</u>) 17, 28 (index) 00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>qty</u>) 17, 28 (index)
RESPONSE (Library will respond with)	Application Layer Function Codes (Decimal)	129(Response)		129,	129,
NSE spond with)	Qualifier Codes (hex)	00, 01 17, 28		00, 01 17, 28	00, 01 17, 28 00, 01 17, 28

4	ω	ω	ω	Ν	2	2	• •	
							Obj ect	-
0	N	ل ـ	0	ω	2	<u> </u>	Var iati on	
Double-bit Binary Input Change - All Variations (Default variation)	Double-bit Binary Input	Double-bit Binary Input – Packed Format	Double-bit Binary Input - All Variations (Variation 0 is used to request default variation)	Binary Input Change with Relative Time	Binary Input Change with Time	Binary Input Change without Time	Description	OBJECT
	-	-	1(Read), 22(Assign Class)			-	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
06,07,08	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>aty</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	06,07,08	06,07,08	06,07,08	Qualifier Codes (hex)	rse)
129	129	129,	129(Response)	129, 130	129, 130	129, 130 (Unsolicited Response)	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
17, 28	00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	17, 28	٦٢, 28	17, 28	Qualifier Codes (hex)	NSE pond with)

10	10	10	4	4	4	Obj ect	
N	-	0	ω	2	-	Var iati on	
Binary Output Status	Binary Output	Binary Output - All Variations	Double-bit Binary Input Change with Relative Time	Double-bit Binary Input Change with Time	Double-bit Binary Input Change without Time	Description	OBJECT
_	-	1(Read), 22(Assign Class)				Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	06,07,08	06,07,08	06,07,08	Qualifier Codes (hex)	r arse)
129,	129,	129	129, 130	129, 130	129, 130 (Unsolicited Response)	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	17, 28	17,20	17, 28	Qualifier Codes (hex)	pond with)

DNP3 Server
Properties

20	20	20	12	12	Obj ect	
Ν		0		0	iati on	
16-Bit Binary Counter	32-Bit Binary Counter	Binary Counter - All Variations	Control Relay Output Block	Control Block - All Variations	Description	OBJECT
<u>, </u>	ب	1(Read), 22(Assign Class) 7(Immediate Freeze), 8 (Immediate Freeze - No Response), 9 (Freeze and Clear), 10 (Freeze and Clear – No Response(3(Select), 4(Operate), 5 (Direct Operate), 6 (Direct Operate NR)		Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	17, 28		Qualifier Codes (hex)	r irse)
129,	129,	129	129		Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	echo of request		Qualifier Codes (hex)	NSE spond with)

-	OBJECT	REQUEST (Library will parse)	arse)	RESPONSE (Library will respond with)
Obj Var ect iati on	Description	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)	Application Layer Function Codes (Decimal)
20 5	32-Bit Binary Counter without Flag	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtv</u>) 17, 28 (index)	129,
20 6	16-Bit Binary Counter without Flag	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtv</u>) 17, 28 (index)	129,
21 0	Frozen Counters - All Variations	1(Read), 22(Assign Class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtx</u>) 17, 28 (index)	129(Response)
21 1	32-Bit Frozen Counter	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtx</u>) 17, 28 (index)	129,

DNP3 Server
Properties

N	N	N	N		
21	21	21	21	Obj ect	
Q	6	CJ	Ν	Var iati on	
32-Bit Frozen Counter without Flag	16-Bit Frozen Counter with Time of Freeze	32-Bit Frozen Counter with Time of Freeze	16-Bit Frozen Counter	Description	OBJECT
	L	L	-	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtt</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtx</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtx</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	Qualifier Codes (hex)	rse)
129,	129,	129,	129,	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	Qualifier Codes (hex)	NSE pond with)

		OBJECT	REQUEST (Library will parse)	r ırse)	RESPONSE (Library will respond with)	vse pond with)
Obj ect	Var iati on	Description	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)
21	10	16-Bit Frozen Counter without Flag	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtv</u>) 17, 28 (index)	129,	00, 01 17, 28
22	0	Counter Change Event - All Variations	-	06,07,08	129	17, 28
22	-	32-Bit Counter Change Event without Time		06,07,08	129, 130	17, 28
22	2	16-Bit Counter Change Event without Time	-1	06,07,08	129, 130	17, 28
22	5	32-Bit Counter Change Event with Time		06,07,08	129, 130	17, 28
22	ര	16-Bit Counter Change Event with Time	-	06,07,08	129, 130	17, 28
23	0	Frozen Counter Events - All Variations	-	06,07,08	129	17, 28
23	-	32-Bit Frozen Counter Event without Time		06,07,08	129, 130	17, 28
23	N	16-Bit Frozen Counter Event without Time	-	06,07,08	129, 130	17, 28
23	J	32-Bit Frozen Counter Event with Time		06,07,08	129, 130	17, 28

30	30	30	23	Obj ect	
N	-	0	6	Var iati on	
16-Bit Analog Input	32-Bit Analog Input	Analog Input - All Variations	16-Bit Frozen Counter Event with Time	Description	OBJECT
-	1	1(Read), 22(Assign Class) 7(Immediate Freeze), 8 (Immediate Freeze - No Response), 9 (Freeze and Clear), 10 (Freeze and Clear – No Response)	-	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	06,07,08	Qualifier Codes (hex)	rse)
129,	129,	129(Response)	129, 130	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
00, 01 17, 28	00, 01 17, 28	00, 01 17, 28	17, 28	Qualifier Codes (hex)	pond with)

Obj ect	Var iati	OBJECT Description	Applicati	REQUEST (Library will pa Application Layer Function Codes (Decimal)	par
30	ω 3	32-Bit Analog Input without flag			00, 01 06 (no 07, 08 17, 28
30	4	16-Bit Analog Input without flag			00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtx</u>) 17, 28 (index)
ას	U	Single-precision Tloat -point with Tlag	1		00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtv</u>) 17, 28 (index)
31	0	Frozen Analog Input - All Variations	1(Read), 22(Assign Class)	lass)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)

	1	OBJECT	REQUEST (Library will parse)	rse)	RESPONSE (Library will respond with)	VSE pond with)
Obj ect	Var iati on	Description	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)
<u>3</u>	-	32-Bit Frozen Analog Input		00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited gtx) 17, 28 (index)	129,	00, 01, 17, 28
31	2	16-Bit Frozen Analog Input		00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>atx</u>) 17, 28 (index)	129,	00, 01, 17, 28
31	ω	32-Bit Frozen Analog Input with Time of Freeze	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gty</u>) 17, 28 (index)	129,	00, 01, 17, 28
31	4	16-Bit Frozen Analog Input with Time of Freeze		00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtv</u>) 17, 28 (index)	129,	00, 01, 17, 28

						1	,
	32	32	3 T	31	31	Obj ect	
		0	~	თ	Сī	Var iati on	
	32-Bit Analog Change Event without Time	Analog Change Event - All Variations	Single-precision Tloat -point with Tlag	16-Bit Frozen Analog Input without Flag	32-Bit Frozen Analog Input without Flag	Description	OBJECT
	1	1	-	-	-	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
	06,07,08	06,07,08	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtx</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtx</u>) 17, 28 (index)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited <u>gtx</u>) 17, 28 (index)	Qualifier Codes (hex)	r irse)
	129, 130	129	129,	129,	129,	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
17, 28	17, 28	17, 28	00, 01, 17, 28	00, 01, 17, 28	00, 01, 17, 28	Qualifier Codes (hex)	NSE spond with)

	Obj ect	32	32	32	32	33	33	33	33	33	33	သ
_	Var iati on	ы	4	Сī	7	0	-1	2	ω	4	5	7
OBJECT	Description	32-Bit Analog Change Event with Time	16-Bit Analog Change Event with Time	Single-precision float –point Analog Change Event without Time	Single-precision float -point Analog Change Event with Time	Frozen Analog Event - All Variations	32-Bit Frozen Analog Event without Time	16-Bit Frozen Analog Event without Time	32-Bit Frozen Analog Event with Time	16-Bit Frozen Analog Event with Time	Single-precision float -point Frozen Analog Change Event without Time	Single-precision float –point Frozen Analog Change Event with Time
REQUEST (Library will parse)	Application Layer Function Codes (Decimal)				→	_	1	1	_	-1	1	
Irse)	Qualifier Codes (hex)	06,07,08	06,07,08	06,07,08	06,07,08	06,07,08	06,07,08	06,07,08	06,07,08	06,07,08	06,07,08	06,07,08
RESPONSE (Library will respond with)	Application Layer Function Codes (Decimal)	129, 130	129, 130	129, 130	129, 130	129	129, 130	129, 130	129, 130	129, 130	129, 130	129, 130
pond with)	Qualifier Codes (hex)	17, 28	17, 28	17, 28	17, 28	17, 28	17, 28	17, 28	17, 28	17, 28	17, 28	17, 28

2	Var	OBJECT	REQUEST (Library will parse)		(Library will respond with)
Obj ect	iati on	Description	Application Layer Function Codes (Decimal)	Qualifier Codes (hex)	
40	0	Analog Output Status - All Variations	1(Read), 22(Assign Class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	
40	→	32-Bit Analog Output Status		00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	
40	N	16-Bit Analog Output Status	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	
40	ω	Single-precision float –point Analog Output Status	-	00, 01 (start-stop) 06 (no range, or all) 07, 08 ,(limited qty) 17, 28 (index)	p) (ty)

80		60		60		60	60	50	41	41	41	• •	
0								0				Obj ect	
~		4		ω		2	<u> </u>	<u> </u>	ω	2		Var iati on	
Internal Indications		Class 3 Data		Class 2 Data		Class 1 Data	Class 0 Data	Time and Date	Analog Output – Single-precision float – point	16-Bit Analog Output Block	32-Bit Analog Output Block	Description	OBJECT
	20 (Enable Unsolicited), 21,(Disable Unsolicited)		20 (Enable Unsolicited), 21,(Disable Unsolicited)		20 (Enable Unsolicited), 21,(Disable Unsolicited)			2(Write)	3, 4, 5, 6	3, 4, 5, 6	3, 4, 5, 6	Application Layer Function Codes (Decimal)	REQUEST (Library will parse)
00 index=7	06	06,07,08	06	06,07,08	06	06,07,08	06	07 (Quantity = 1)	17, 28	17, 28	17, 28	Qualifier Codes (hex)	rse)
		129		129		129	129	129	129	129	129	Application Layer Function Codes (Decimal)	RESPONSE (Library will respond with)
								07 (quantity = 1)	echo of request	echo of request	echo of request	Qualifier Codes (hex)	NSE spond with)



Atop Technologies, Inc.

www.atoponline.com www.atop.com.tw

TAIWAN HEADQUARTER:

2F, No. 146, Sec. 1, Tung-Hsing Rd, 30261 Chupei City, Hsinchu County Taiwan, R.O.C. Tel: +886-3-550-8137 Fax: +886-3-550-8131

ATOP INDIA OFFICE:

Abhishek Srivastava Head of India Sales Atop Communication Solution(P) Ltd. No. 22, Kensington Terrace, Kensington Rd, Bangalore, 560008, India Tel: +91-80-4920-6363 E-mail: Abhishek.S@atop.in

ATOP EMEA OFFICE:

Bhaskar Kailas (BK) Vice President (Business Development) Atop Communication Solution(P) Ltd. No. 22, Kensington Terrace, Kensington Rd, Bangalore, 560008, India Tel: +91-988-0788-559 E-mail: Bhaskar.k@atop.in

ATOP CHINA BRANCH:

3F, 75th, No. 1066 Building, Qingzhou North Road, Shanghai, China Tel: +86-21-64956231

ATOP INDONESIA BRANCH:

Jopson Li Branch Director Wisma Lampung Jl. No. 40, Tomang Raya Jakarta, Barat, 11430, Indonesia Tel: +62-857-10595775 E-mail: jopsonli@atop.com.tw

ATOP AMERICAS OFFICE:

Venke Char Sr. Vice President & Head of Business 11811 North Tatum Blvd, Suite 3031 Phoenix, AZ 85028, United States Tel: +1-602-953-7669 E-mail: venke@atop.in