Wonderware Operations Integration – Supervisory BACnet/IP BACLITE Server (G-1.2 Series)

Contents

WONDERWARE OPERATIONS INTEGRATION - SUPERVISORY BACNET/IP BACLITE SERVER (G-1.2 SERIES).. 3

Legal Information..4
Technical Support.. 5
Introduction to the BACLITE OI Server.. 6
Configuring the BACLITE OI Server...7
  Configuring a Channel's Communication Settings.. 7
  Setting a Device's Station ID..7
BACLITE OI Server Reference...9
  Item Reference Syntax.. 9
  Address Descriptions...9
  Supported Data Types.. 43
  Examples of Item References..49
BACLITE OI Server Error Codes...50

Wonderware Operations Integration - Supervisory BACnet/IP BACLITE Server (G-1.2 Series)

This document describes the technical specifications and configuration options for the Wonderware® Operations Integration - Supervisory BACnet/IP BACLITE Server (or BACLITE OI Server, for short).

Last revision 2016-03-25

Legal Information

Although precaution has been taken in the preparation of this documentation, Schneider Electric Software, LLC assumes no responsibility for errors or omissions. The information in this documentation is subject to change without notice and does not represent a commitment on the part of Schneider Electric Software, LLC. The software described in this documentation is furnished under a license agreement. This software may be used or copied only in accordance with the terms of such license agreement.

Schneider Electric Software, LLC
26561 Rancho Parkway South
Lake Forest, CA 92630 U.S.A.
(949) 727-3200

http://software.invensys.com/

For comments or suggestions about the product documentation, send an e-mail message to ProductDocumentationComments@schneider-electric.com.

Technical Support

Wonderware Technical Support offers a variety of support options to answer any questions on Wonderware products and their implementation.

Before you contact Technical Support, refer to the relevant section(s) in this documentation for a possible solution to the problem. If you need to contact technical support for help, have the following information ready:

- The type and version of the operating system you are using.
- Details of how to recreate the problem.
- The exact wording of the error messages you saw.
- Any relevant output listing from the Log Viewer or any other diagnostic applications.
- Details of what you did to try to solve the problem(s) and your results.
- If known, the Wonderware Technical Support case number assigned to your problem, if this is an ongoing problem.

Introduction to the BACLITE OI Server

These are the technical specifications for Wonderware Operations Integration - Supervisory BACnet/IP BACLITE Server.

Requirements

BACLITE OI Server requires Wonderware Operations Integration – Core G-1.2 or later.

Supported hardware and software

This OI Server connects over UDP/IP to all devices that are fully compatible with the BACnet/IP protocol. Also, it supports the following application services:

Application Service	Initiate Requests
Read-Property-Multiple	#
Read-Property-Single	#
Write-Property-Single	#

Conformance

The following hardware and software was used for conformance testing of this OI Server:

- Equipment:
  - Metasys Trane BCU
  - NAE (Johnson Controls)
  - SCADA Engine BACnet Simulator
- Cable: Ethernet

> **Note:** Once the registers on the BACnet devices are writable and the testing procedures are based on a device simulator which does not supports writing all the addresses we cannot guarantee writing support to all the BACnet objects.

Configuring the BACLITE OI Server

Each server instance has its own hierarchy of objects, and each object has parameters that you need to configure in order to establish communication between the OI Server and individual devices on the network.

You can view a BACLITE server instance's configuration hierarchy under its **Configuration** node.

This section only describes how to configure object parameters for a BACLITE server instance. For more general information about adding and configuring objects, see "Configuring Your OI Server" in the *Operations Integration Server Manager Help*.

Configuring a Channel's Communication Settings

Configure the communication settings for a selected channel to ensure uninterrupted communication with the device network.

Link Type

The specific interface or protocol to be used. The only option at this time is **BACnet/IP**.

Device UDP Port

The specific UDP/IP port to be used. The valid values range from 1 to 65535. If no port is specified, it will default to port 47808 (BAC0h).

Null Placeholder:Block Size

This is actually two settings separated by a colon (`:`).

The first setting, **Null Placeholder**, is the value that you use to represent NULL in write operations. For some properties like PRESENT-VALUE and DESCRIPTION, when this value is encountered in a write operation, a special NULL marker is sent to the device. For example, if you use **`-333`** for NULL in your BACnet application, specify that value. There is no default value for this setting, so if you do not specify a value, the NULL marker will never be sent. (An empty string is not an acceptable placeholder value.)

The second setting, **Block Size**, is the number of objects that can be requested in each message. It is a way to limit the size of the messages transmitted. If you do not specify a value, the default value is 30.

Advanced

Click this button to open the *Advanced Settings* dialog box, which provides access to additional communication settings such as timeouts, retries, and buffer sizes. You might need to change these settings if the OI Server behaves unexpectedly during run time, but the default settings should work for most network configurations. For more information about these settings, see "Advanced Settings" in *Operations Integration Server Manager Help*.

Setting a Device's Station ID

Set the station ID for a selected device so that the OI Server can identify and communicate with it on the network.

Syntax

In most cases, if you are using the standard BACnet/IP protocol, the station ID for a target device should use the following syntax:

```
<subnet>:<device ID>
```

If you are on a restricted network or if the target device cannot be discovered on the network, the station ID for the device should use one of the following syntaxes:

<IP address>:<subnet>:<device ID>

<IP address>:<subnet>:<device ID>:<MAC address>

When the OI Server sends a broadcast message to discover BACnet devices on the network, it uses the subnet mask 255.255.255.0. If you have devices that are not on the same network using this mask, the OI Server will not be able to discover them. To work around this issue, include the device's MAC address in the station ID.

The following syntax diagram shows all of the possible options:
{ | *IP address*: }*subnet*: *device ID*{ | : *MAC address* }

IP address

The specific IP address of the target device.

subnet

The BACnet/IP virtual network number. This should be a decimal value.

device ID

The specific ID number of the target device. This should be a decimal value.

MAC address

The specific MAC address of the target device. This should be a hexadecimal value of either one byte (e.g., `B8`) or six bytes (e.g., `B8098AC14A1F`) in length, depending on how your BACnet program and network gateway are configured. For the BACnet/IP protocol, the MAC address is typically six bytes.

If *MAC address* is not specified, *device ID* is automatically encoded as an equivalent, six-byte hexadecimal value and included in the network protocol data unit (NPDU). This value is low-bytes first and padded with zeroes. For example, if *device ID* is `1024` (0x400), the equivalent hexadecimal value is `000400000000`.

If *MAC address* is specified, make sure the number of bytes (1 or 6) matches the Destination MAC Layer Address Length (i.e., the **`bacnet.dlen`** field) in your BACnet program.

Examples

Examples of valid station IDs:

```
12:2

1111:12

1234:32

192.168.110.101:1111:12

192.168.110.101:1234:32

192.168.110.101:1111:12:B8098AC14A1F
```

BACLITE OI Server Reference

Use item references to access data stored in memory registers in connected devices, as well as to access standard system items in the OI Server itself.

This section only describes the item reference syntax and options for the BACLITE server. For more general information about item references, see "Managing Device Items" and "Item Reference Descriptions" in the *Operations Integration Server Manager Help*.

Item Reference Syntax

Item references in this OI Server use the following syntax.

We can communicate with a specific PLC object instance and property using the following syntax:

```
<type>:<instance>:<property>:[priority array]
```

The following syntax diagram shows address options:
register type :*instance* : *property* { | :*priority array* }

register type

The register type. Valid types are AI, AO, AV, BI, BO, BV, DEV, MSI, MSO, MSV, SCH, LSP, LSZ, ACC, AVR, CAL, CMD, EVT, LOP, NOT, PRG, PC, GRP, FO and TL.

instance

The instance of the specified register type.

property

The property of the specified register type and instance.

priority array

The priority for writing. If no value is specified, the default value is 16.

Address Descriptions

The address descriptions consist of the register type, its item name and the allowable range of values, the default data type, allowable suffixes (if any), and allowable access methods.

Analog Input (AI)

Supported properties of the Analog Input object (AI)

Property	Value Format	Example

OBJECT-IDENTIFIER	String
	`[ObjectNumber]:`
	`[Instance Number]`
		1:0

OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	Integer	5

Property	Value Format	Example
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Integer	0
UPDATE-INTERNAL	Integer	0
UNITS	Integer	0
MIN-PRES-VALUE	Integer	0
MAX-PRES-VALUE	Integer	0
RESOLUTION	Float	0.1
COV-INCREMENT	Integer	0
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0
HIGH-LIMIT	Integer	0
LOW-LIMIT	Integer	0
DEADBAND	Integer	0
LIMIT-ENABLE	2 Boolean values (1 = true, 0 = false) encoded as a string `{lowLimitEnable, highLimitEnable}`	01
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	123-AI

Analog Output (AO)

Supported properties of the Analog Output object (AO)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	Integer	5
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as a string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Integer	0
UNITS	Integer	0
MIN-PRES-VALUE	Integer	0
MAX-PRES-VALUE	Integer	0
RESOLUTION	Real	0.1
PRIORITY-ARRAY	16 values separated by pipes (|) and encoded as string. Each value can be one of the following data types: NULL, real, binary, integer. For more information, see BACnetPriorityArray.	NULL|NULL|NULL|NULL|NULL|NULL|12|NULL| NULL|NULL|NULL|NULL|NULL|NULL|NULL|25.000000
RELINQUISH-DEFAULT	Real	12.3
COV-INCREMENT	Integer	0
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0
HIGH-LIMIT	Integer	0
LOW-LIMIT	Integer	0
DEADBAND	Integer	0
LIMIT-ENABLE	2 Boolean values (1 = true, 0 = false) encoded as a string `{lowLimitEnable, highLimitEnable}`	01
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101

Property	Value Format	Example
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Test

Analog Value (AV)

Supported properties of the Analog Value object (AV)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
DESCRIPTION	String	This sensor monitors the core activity
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Integer	0
UNITS	Integer	0
PRIORITY-ARRAY	Sequence of 16 values separated by pipes (|) and encoded as a string. The type of these values can be one of the following (Null, real, binary, integer) . For more information, see BACnetPriorityArray.	NULL|NULL|NULL|NULL|NULL|NULL|12|NULL| NULL|NULL|NULL|NULL|NULL|NULL|NULL|25.000000
RELINQUISH-DEFAULT	Real	12.3
COV-INCREMENT	Integer	0
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0
HIGH-LIMIT	Integer	0
LOW-LIMIT	Integer	0
DEADBAND	Integer	0

Property	Value Format	Example
LIMIT-ENABLE	2 Boolean values (1 = true, 0 = false) encoded as a string `{lowLimitEnable, highLimitEnable}`	01
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Test

Binary Input (BI)

Supported properties of the Binary Input object (BI)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Integer	123
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	Integer	5
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Integer	0
POLARITY	Integer (normal = 0, reverse = 1)	1
INACTIVE-TEXT	String	test
ACTIVE-TEXT	String	Test

Property	Value Format	Example
CHANGE-OF-STATE-TIME	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
CHANGE-OF-STATE-COUNT	Unsigned Integer	33
TIME-OF-STATE-COUNT-RESET	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
ELAPSED-ACTIVE-TIME	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
TIME-OF-ACTIVE-TIME-RESET	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
TIME-DELAY	Unsigned Integer	10
NOTIFICATION-CLASS	Integer	0
ALARM-VALUE	Integer	1
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Test

Binary Output (BO)

Supported properties of the Binary Output object (BO)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Integer	123
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	Integer	5

Property	Value Format	Example
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Integer	0
POLARITY	Integer (normal = 0, reverse = 1)	1
INACTIVE-TEXT	String	test
ACTIVE-TEXT	String	Test
CHANGE-OF-STATE-TIME	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
CHANGE-OF-STATE-COUNT	Unsigned Integer	33
TIME-OF-STATE-COUNT-RESET	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
ELAPSED-ACTIVE-TIME	Unsigned Integer	12
TIME-OF-ACTIVE-TIME-RESET	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
MINIMUN-OFF-TIME	Integer	0
MINIMUN-ON-TIME	Integer	0
PRIORITY-ARRAY	Sequence of 16 values separated by pipes (|) and encoded as a string. The type of these values can be one of the following (Null, real, binary, integer). For more information, see BACnetPriorityArray.	NULL|NULL|NULL|NULL|NULL|NULL|12|NULL| NULL|NULL|NULL|NULL|NULL|NULL|NULL|25.000000
RELINQUISH-DEFAULT	Real	12.3
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0
FEEDBACK-VALUE	Integer (inactive = 0, active = 1)	1
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Test

Binary Value (BV)

Supported properties of the Binary Value object (BV)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]: [Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Integer	123
DESCRIPTION	String	This sensor monitors the core activity
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Integer	0
INACTIVE-TEXT	String	test
ACTIVE-TEXT	String	Test
CHANGE-OF-STATE-TIME	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
CHANGE-OF-STATE-COUNT	Unsigned Integer	33
TIME-OF-STATE-COUNT-RESET	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
ELAPSED-ACTIVE-TIME	Unsigned Integer	12
TIME-OF-ACTIVE-TIME-RESET	Date and Time separated by a space character and encoded as a string. For more information look at BACnetDateTime type in the Data Type section.	23-8-2011 9:42:44.64
MINIMUN-OFF-TIME	Integer	0
MINIMUN-ON-TIME	Integer	0
PRIORITY-ARRAY	Sequence of 16 values separated by pipes (|) and encoded as a string. The type of these values can be one of the following (Null, real, binary, integer) . For more information, see BACnetPriorityArray.	NULL|NULL|NULL|NULL|NULL|NULL|12|NULL| NULL|NULL|NULL|NULL|NULL|NULL|NULL|25.000000
RELINQUISH-DEFAULT	Real	12.3
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0
ALARM-VALUE	Integer	1

Property	Value Format	Example
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Test

Multi-State Input (MSI)

Supported properties of the Multi-State Input object (MSI)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	Integer	5
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Boolean	0
NUMBER-OF-STATES	Unsigned Integer	2
STATE-TEXT	String	test
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0
ALARM-VALUES	List of unsigned integers separated by pipes (|) and encoded as a string	1|1|2
FAULT-VALUES	List of unsigned integers separated by pipes (|) and encoded as a string	1|3

Property	Value Format	Example
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Test

Multi-State Output (MSO)

Supported properties of the Multi-State Output object (MSO)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	Integer	5
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Boolean	0
NUMBER-OF-STATES	Unsigned Integer	2
STATE-TEXT	String	test
PRIORITY-ARRAY	Sequence of 16 values separated by pipes (|) and encoded as a string. The type of these values can be one of the following (Null, real, binary, integer) . For more information, see BACnetPriorityArray.	NULL|NULL|NULL|NULL|NULL|NULL|12|NULL| NULL|NULL|NULL|NULL|NULL|NULL|NULL|25.000000
RELINQUISH-DEFAULT	Real	12.3
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0

Property	Value Format	Example
FEEDBACK-VALUE	Integer (inactive = 0, active = 1)	1
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Test

Multi-State Value (MSV)

Supported properties of the Multi-State Value object (MSV)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
DESCRIPTION	String	This sensor monitors the core activity
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Boolean	0
NUMBER-OF-STATES	Unsigned Integer	2
STATE-TEXT	String	test
PRIORITY-ARRAY	Sequence of 16 values separated by pipes (|)and encoded as a string. The type of these values can be one of the following (Null, real, binary, integer). For more information, see BACnetPriorityArray.	NULL|NULL|NULL|NULL|NULL|NULL|12|NULL| NULL|NULL|NULL|NULL|NULL|NULL|NULL|25.000000
RELINQUISH-DEFAULT	Real	12.3
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0

Property	Value Format	Example
ALARM-VALUES	List of unsigned integers separated by pipes (|) and encoded as a string	1|1|2
FAULT-VALUES	List of unsigned integers separated by pipes (|) and encoded as a string	1|3
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Test

Schedule (SCH)

Supported properties of the Schedule object (SCH)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
DESCRIPTION	String	This sensor monitors the core activity
EFFECTIVE-PERIOD	String	
WEEKLY-SCHEDULE	String	
EXCEPTION-SCHEDULE	String	
SCHEDULE-DEFAULT	String Bacnet Type Any	2:12 (See Dataype table for more information)
LIST-OF-OBJECT-PROPERTY-REFERENCES	String `[OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX (Optional)] | [DEVICE-IDENTIFIER (Optional)]; [OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX`	This property specifies the Device Identifiers, Object Identifiers and Property Identifiers of the properties to be written with specific values at specific times on specific days. For example: 29,4194303|29|786|30,4194303; 29,4194303|29|786|30,4194303…

Property	Value Format	Example
	`(Optional)] | [DEVICE-IDENTIFIER (Optional)];`...	
PRIORITY-FOR-WRITING	Unsigned Integer	
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
RELIABILITY	Integer	0
OUT-OF-SERVICE	Boolean	0
PROFILE-NAME	String	Test

Device (DEV)

Supported properties of the Device object (DEV)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
SYSTEM-STATUS	Integer `{operational (0), operational-read-only (1), download-required (2), download-in-progress (3), non-operational (4), backup-in-progress (5)}`	4
VENDOR-NAME	String	TEST
VENDOR-IDENTIFIER	Integer	12
MODEL-NAME	String	TEST
FIRMWARE-REVISION	String	TEST
APLICATION-SOFTWARE-VERSION	String	test
LOCATION	String	test
DESCRIPTION	String	This sensor monitors the core activity
PROTOCOL-VERSION	Unsigned Integer	4
PROTOCOL-REVISION	Unsigned Integer	4
PROTOCOL-SERVICES-SUPPORTED	See BacnetServicesSupported	
PROTOCOL-OBJECT-TYPES-SUPPORTED	See BacnetServicesSupported	
PROFILE-NAME	String	Test

Life Safety Point (LSP)

Supported properties of the Life Safety Point object (LSP)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
TRACKING-VALUE	Float (0–23)	Reflects the non-latched state of the Life Safety Point object.
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	String	This is a text description of the physical device that the Life Safety Point object represents.
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Boolean	0
MODE	Unsigned Integer (0–14)	Shall convey the desired operating mode for the Life Safety Point object.
ACCEPTED-MODES	String	Shall specify all values the Mode property accepts when written to using BACnet services. For example: 10|20|30|40|…
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0
LIFE-SAFETY-ALARM-VALUES	String	This property is required if intrinsic reporting is supported by this object.
ALARM-VALUES	List of unsigned integers separated by pipes (|) and encoded as a string	1|2|3
FAULT-VALUES	List of unsigned integers separated by pipes (|) and encoded as a string	1|2|3
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101

Property	Value Format	Example
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
SILENCED	Unsigned Integer (0–3)	Indicates the silenced state.
OPERATION-EXPECTED	Unsigned Integer (0–9)	Specifies the next operation expected by this object to handle a specific life safety situation.
MAINTENANCE-REQUIRED	Unsigned Integer (0–3)	Indicates the type of maintenance required for the life safety point.
SETTING	Unsigned Integer	Shall be used to convey the desired setting of the input(s) or process used to determine the logical state of the Present Value.
DIRECT-READING	Float	Indicates an analog quantity that reflects the measured or calculated reading from an initiating device.
UNITS	Integer	0
MEMBER-OF	List separated by pipes (|) and encoded as a string	Shall indicate those Life Safety Zone objects of which this Life Safety Point object is considered to be a zone member. For example: 0:1,0:2|0:3,0:4|0:5,0:6|…
PROFILE-NAME	String	test

Life Safety Zone (LSZ)

Supported properties of the Life Safety Zone object (LSZ)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
TRACKING-VALUE	Float (0–23)	Reflects the non-latched state of the Life Safety Point object.
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	String	This is a text description of the physical device that the Life Safety Point object represents.
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010

Property	Value Format	Example
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Boolean	0
MODE	Integer (0–14)	Shall convey the desired operating mode for the Life Safety Point object.
ACCEPTED-MODES	String	Shall specify all values the Mode property accepts when written to using BACnet services. For example: 10|20|30|40|…
TIME-DELAY	Integer	0
NOTIFICATION-CLASS	Integer	0
LIFE-SAFETY-ALARM-VALUES	String	This property is required if intrinsic reporting is supported by this object.
ALARM-VALUES	List of unsigned integers separated by pipes (|) and encoded as a string	1|2|3
FAULT-VALUES	List of unsigned integers separated by pipes (|) and encoded as a string	1|2|3
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
SILENCED	Unsigned Integer (0–3)	Indicates the silenced state.
OPERATION-EXPECTED	Unsigned Integer (0–9)	Specifies the next operation expected by this object to handle a specific life safety situation.
MAINTENANCE-REQUIRED	Unsigned Integer (0–3)	Indicates the type of maintenance required for the life safety point.
ZONE-MEMBERS	List separated by pipes (|) and encoded as a string	(0:1,0:2|0:3,0:4|0:5,0:6|...)
MEMBER-OF	List separated by pipes (|) and encoded as a string	Shall indicate those Life Safety Zone objects of which this Life Safety Point object is considered to be a zone member. For example: 0:1,0:2|0:3,0:4|0:5,0:6|…
PROFILE-NAME	String	test

Accumulator (ACC)

Supported properties of the Accumulator object (ACC)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	temperatureSensor_core
OBJECT-TYPE	Integer	0
PRESENT-VALUE	Float	123.2
DESCRIPTION	String	This sensor monitors the core activity
DEVICE-TYPE	String	This property, of type CharacterString, is a text description of the physical device represented by the Accumulator object.
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	1010
EVENT-STATE	Integer	0
RELIABILITY	Integer	0
OUT-OF-SERVICE	Boolean	0
SCALE	Integer/Float/String	Indicates the conversion factor to be multiplied with the value of the Present Value property to provide a value in the units indicated by Units. For example: 3.14
UNITS	String	0
PRESCALE	List of integers separated by semicolons (;) and encoded as a string	Presents the coefficients that are used for converting the pulse signals generated by the measuring instrument into the value displayed by Present Value. For example: -3413;-1119
MAX-PRES-VALUE	Integer	0
VALUE-CHANGE-TIME	Timestamp encoded as a string	This read-only property, of type BACnetDateTime, shall be present if the Present_Value property is adjustable by writing to the Value Before Change or Value Set properties. For example: 25-10-2013 15:39:23.27
VALUE-BEFORE-CHANGE	Integer	This property, of type Unsigned, indicates the value of the Present_Value property just prior to the most recent write to the Value Set or Value Before Change properties. For example: 100
VALUE-SET	Integer	This property, of type Unsigned, indicates the value of the Present_Value property after the most recent write to the Value Set or Value Before Change properties. For example: 786

Property	Value Format	Example
PULSE-RATE	Integer	This property, of type Unsigned, shall indicate the number of input pulses received during the most recent period specified by Limit Monitoring Interval. For example: 0
HIGH-LIMIT	Integer	0
LIMIT-MONITORING-INTERVAL	Integer	This property, of type Unsigned, specifies the monitoring period in seconds for determining the value of Pulse Rate. For example: 500
NOTIFICATION-CLASS	Integer	0
TIME-DELAY	Integer	0
LIMIT-ENABLE	2 Boolean values (1 = true, 0 = false) encoded as a string `{lowLimitEnable, highLimitEnable}`	01
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	101
NOTIFY-TYPE	Integer	1
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	test

Averaging (AVR)

Supported properties of the Averaging object (AVR)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	1:0
OBJECT-NAME	String	Averaging Object
OBJECT-TYPE	Integer	18
MINIMUM-VALUE	Float	Shall reflect the lowest value contained within the buffer window for the most recent 'Window Samples' samples. For example: 1.0

Property	Value Format	Example
MINIMUM-VALUE-TIMESTAMP	String `[Date] | [Time]`	Indicates the date and time at which the value stored in Minimum Value was sampled. For example: [1-1-1970] [11:0:0.0]
AVERAGE-VALUE	Float	Shall reflect the average value contained within the buffer window for the most recent 'Window Samples' samples. For example: 1.0
VARIANCE-VALUE	Float	Shall reflect the variance value contained within the buffer window for the most recent 'Window Samples' samples. For example: 1.0
MAXIMUM-VALUE	Float	Shall reflect the highest value contained within the buffer window for the most recent 'Window Samples' samples. For example: 1.0
MAXIMUM-VALUE-TIMESTAMP	String `[Date] | [Time]`	Indicates the date and time at which the value stored in Maximum Value was sampled. For example: [1-1-1970] [11:0:0.0]
DESCRIPTION	String	This sensor monitors the core activity For example: Averaging Object
ATTEMPTED-SAMPLES	Integer	Indicates the number of samples that have been attempted to be collected for the current window. For example: 0
VALID-SAMPLES	Integer	Indicates the number of samples that have been successfully collected for the current window. For example: 0
OBJECT-PROPERTY-REFERENCE	String `[OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX (Optional)] | [DEVICE-IDENTIFIER (Optional)]`	Shall identify the object and property whose value is to be sampled during the 'Window Interval'. For example: 29,4194303|29|786|30,4194303
WINDOW-INTERVAL	Integer	Shall indicate the period of time in seconds over which the minimum, maximum and average values are calculated. For example: 50
WINDOW-SAMPLES	Integer	Shall indicate the number of samples to be taken during the period of time specified by the 'Window Interval' property. For example: 10
PROFILE-NAME	String	Averaging Profile

Calendar (CAL)

Supported properties of the Calendar object (CAL)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]: [Instance Number]`	1:0
OBJECT-NAME	String	Calendar Object
OBJECT-TYPE	Integer	6
DESCRIPTION	String	This sensor monitors the core activity For example: Calendar Object
PRESENT-VALUE	Boolean	Indicates the current value of the calendar. For example: 1 (TRUE) if the current date is in the Date List and 0 (FALSE) if it is not.
DATE-LIST	String `[DATE]|[DATE-RANGE]|[MONTH/ WEEK-OF-MOMTH/DAY-OF-WEEK]`...	This property is a List of BACnetCalendarEntry, each of which is either an individual date (Date), range of dates (BACnetDateRange), or month/week-of-month/day-of-week specification (BACnetWeekNDay). For example: 11-2-1981|30-1-2013,11-2-2013|12#4#21…
PROFILE-NAME	String	Calendar Profile

Command Object (CMD)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]: [Instance Number]`	1:0
OBJECT-NAME	String	Object: Command
OBJECT-TYPE	Integer	6
DESCRIPTION	String	This sensor monitors the core activity For example: Description: Command
PRESENT-VALUE	Unsigned Integer	Indicates which action the Command object is to take or has already taken. For example: 3.
IN-PROCESS	Boolean	Shall be set to TRUE when a value is written to the Present Value property. This TRUE value indicates that the Command object has begun processing one of a set of action sequences. Once all of the writes have been attempted by the Command object, the In_Process property shall be set back to FALSE. For example: 1/0.
ALL-WRITES-SUCCESSFUL	Boolean	Indicates the success or failure of the sequence of actions that are triggered when the Present Value property is written to. For example: 1/0.

Property	Value Format	Example
ACTION	String `[Device_Identifier(Optional)], [Object_Identifier], [Property_Identifier], [Property_Array_Index(Optional)], [Property_Value], [Priority(1..16)(Optional)], [Post_Delay(Optional)], [Quit_On_Failure], [Write_Successful]|[Device_Identifier(Optional)], [Object_Identifier], [Property_Identifier], [Property_Array_Index(Optional)], [Property_Value], [Priority(1..16)(Optional)], [Post_Delay(Optional)], [Quit_On_Failure], [Write_Successful];[Device_Identifier(Optional)], [Object_Identifier], [Property_Identifier], [Property_Array_Index(Optional)], [Property_Value], [Priority(1..16)(Optional)], [Post_Delay(Optional)], [Quit_On_Failure], [Write_Successful]|`...	This property, of type BACnetARRAY of BACnetActionList, specifies an array of "action lists." For example: ,5:0,85,,0,,,1,0|8:1,5:2,85,0,5,4,3,0,1;,5:0,85,,0,,,1,0|…
ACTION-TEXT	String `[CharacterString]|[CharacterString]|...`	This property, of type BACnetARRAY of CharacterString, shall be used to indicate a text string description for each of the possible values of the Present Value property. For example: Item 0|Item 1|...
PROFILE-NAME	String	Profile: Command

Event Enrollment (EVT)

Supported properties of the Event Enrollment object (EVT)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]: [Instance Number]`	1:0
OBJECT-NAME	String	Object: Event Enrollment
OBJECT-TYPE	Integer	9
DESCRIPTION	String	This sensor monitors the core activity For example: Description: Event Enrollment
EVENT-TYPE	Integer `[Enumerated - {CHANGE_OF_BITSTRING, CHANGE_OF_STATE,`	This read only property, of type BACnetEventType, indicates the type of event algorithm that is to be used to detect the occurrence of events and report to enrolled devices. For example: 0

Property	Value Format	Example
	CHANGE_OF_VALUE, COMMAND_FAILURE, FLOATING_LIMIT, OUT_OF_RANGE, BUFFER_READY, CHANGE_OF_LIFE_SAFETY, EXTENDED}].	
NOTIFY-TYPE	Integer	This property, of type BACnetNotifyType, shall convey whether the notifications generated by the monitoring algorithm specified by the Event_Type property should be Events or Alarms. For example: 2
EVENT-PARAMETERS	String [Choice]|[Time-Delay]|[Bitmask]|[List-Of-Bitstring-Values]	The Event_Parameters property, of type BACnetEventParameter, determines the algorithm used to monitor the referenced object and provides the parameter values needed for this algorithm. For example: COB|10|10101010|10101010;10101010;10101010…
OBJECT-PROPERTY-REFERENCE	String [OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX (Optional)] | [DEVICE-IDENTIFIER (Optional)]	This property, of type BACnetDeviceObjectPropertyReference, designates the particular object and property referenced by this Event Enrollment object. The algorithm specified by the Event_Type property is applied to the referenced property in order to determine the Event_State of the event. For example: 29,4194303|29|786|30,4194303
EVENT-STATE	Integer	This property, of type BACnetEventState, contains the current state of the event. For example: 0
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string {to-offnormal, to-fault, to-normal}	This property, of type BACnetEventTransitionBits, conveys three flags that determine whether notifications are enabled for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL transitions. For example: 101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string {to-offnormal, to-fault, to-normal}	This property, of type BACnetEventTransitionBits, shall convey three separate flags that each indicate whether the most recent TO-OFFNORMAL, TO-FAULT, or TO-NORMAL event transitions have been acknowledged, if acknowledgment is required for that transition. For example: 111
NOTIFICATION-CLASS	Integer	This property, of type Unsigned, implicitly references a Notification Class object in the device containing the Event Enrollment object. For example: 0
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	This property, of type BACnetARRAY[3] of BACnetTimeStamp, shall convey the times of the last event notifications for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events, respectively. For example: 2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Profile: Event Enrollment

Loop (LOP)

Supported properties of the Loop object (LOP)

Property	Value Format	Example
OBJECT-IDENTIFIER	String **[ObjectNumber]: [Instance Number]**	12:0
OBJECT-NAME	String	Object: Loop
OBJECT-TYPE	Integer	12
PRESENT-VALUE	Float	This property indicates the current output value of the loop algorithm in units of the Output Units property. For example: 100
DESCRIPTION	String	This sensor monitors the core activity For example: Description: Loop
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string **{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}**	This property, of type BACnetStatusFlags, represents four Boolean flags that indicate the general "health" of the loop. For example: 1101
EVENT-STATE	String	The Event State property, of type BACnetEventState, is included in order to provide a way to determine if this object has an active event state associated with it. For example: 1.0
RELIABILITY	Integer	The Reliability property, of type BACnetReliability, provides an indication of whether the Present Value of the loop in question is reliable as far as the BACnet Device or operator can determine and, if not, why. For example: 1
OUT-OF-SERVICE	Boolean	The Out Of Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the algorithm this object represents is or is not in service. For example: 1
UPDATE-INTERVAL	Integer	This property, of type Unsigned, indicates the interval in milliseconds at which the loop algorithm updates the output (Present Value property). For example: 150
OUTPUT-UNITS	Integer	This property, of type BACnetEngineeringUnits, indicates the engineering units for the output (Present Value property) of this control loop. For example: 13
MANIPULATED-VARIABLE-REFERENCE	String **[OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX (Optional)]**	This property is of type BACnetObjectPropertyReference. The output (Present Value) of the control loop is written to the object and property designated by the Manipulated Variable Reference. For example: 29,4194303|29|786

Property	Value Format	Example
CONTROLLED-VARIABLE-REFERENCE	String `[OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX (Optional)]`	This property is of type BACnetObjectPropertyReference. The Controlled Variable Reference identifies the property used to set the Controlled Variable Value property of the Loop object. For example: 29,4194303|29|786
CONTROLLED-VARIABLE-VALUE	Float	This property, of type REAL, is the value of the property of the object referenced by the Controlled Variable Reference property. This control loop compares the Controlled Variable Value with the Setpoint to calculate the error. For example: 0
CONTROLLED-VARIABLE-UNITS	Integer	This property, of type BACnetEngineeringUnits, indicates the engineering units for the Controlled Variable Value property of this object. For example: 2
SETPOINT-REFERENCE	String `[OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX (Optional)] | [DEVICE-IDENTIFIER (Optional)]`	This property, of type BACnetSetpointReference, is a list of references that has a length of zero or one. A length of zero indicates that the setpoint for this control loop is fixed and is contained in the Setpoint property. For example: 8,0|85|15
SETPOINT	Float	This property, of type REAL, is the value of the loop setpoint or of the property of the object referenced by the Setpoint Reference, expressed in units of the Controlled Variable Units property. For example: 10
ACTION	Integer	This property, of type BACnetAction, defines whether the loop is DIRECT or REVERSE acting. For example: 1
PROPORTIONAL-CONSTANT	Float	This property, of type REAL, is the value of the proportional gain parameter used by the loop algorithm. It may be used to represent any of the various forms of gain for the proportional control mode, such as overall gain, throttling range, or proportional band. For example: 5
PROPORTIONAL-CONSTANT-UNITS	Integer	This property, of type BACnetEngineeringUnits, indicates the engineering units of the Proportional Constant property of this object. For example: 2
INTEGRAL-CONSTANT	Float	This property, of type REAL, is the value of the integral gain parameter used by the loop algorithm. For example: 1.2
INTEGRAL-CONSTANT-UNITS	Integer	This property, of type BACnetEngineeringUnits, indicates the engineering units of the Integral Constant property of this object. For example: 3

Property	Value Format	Example
DERIVATIVE-CONSTANT	Float	This property, of type REAL, is the value of the derivative gain parameter used by the loop algorithm. For example: 4.5
DERIVATIVE-CONSTANT-UNITS	Integer	This property, of type BACnetEngineeringUnits, indicates the engineering units of the Derivative Constant property of this object. For example: 7
BIAS	Float	This property, of type REAL, is the bias value used by the loop algorithm expressed in units of the Output Units property. For example: 1.43
MAXIMUM-OUTPUT	Float	This property, of type REAL, is the maximum value of the Present Value property as limited by the PID loop algorithm. For example: 100
MINIMUM-OUTPUT	Float	This property, of type REAL, is the minimum value of the Present Value property as limited by the loop algorithm. For example: 50
PRIORITY-FOR-WRITING	Integer	Loop objects may be used to control the commandable property of an object. This property, of type Unsigned, provides a priority to be used by the command prioritization mechanism. It identifies the particular priority slot in the Priority Array of the Controlled Variable Reference that is controlled by this loop. It shall have a value in the range 1-16. For example: 27
COV-INCREMENT	Float	This property, of type REAL, shall specify the minimum change in Present Value that will cause a COVNotification to be issued to subscriber COV-clients. This property is required if COV reporting is supported by this object. For example: 1.44
TIME-DELAY	Integer	This property, of type Unsigned, shall specify the minimum period of time in seconds that the difference between the Setpoint and the Controlled Variable Value (the Error) must remain outside the band defined by the Error Limit property before a TO-OFFNORMAL event is generated or within the same band before a TO-NORMAL event is generated. This property is required if intrinsic reporting is supported by this object. For example: 17
NOTIFICATION-CLASS	Integer	This property, of type Unsigned, shall specify the notification class to be used when handling and generating event notifications for this object. For example: 0
ERROR-LIMIT	Float	This property, of type REAL, shall convey the absolute magnitude that the difference between the Setpoint and Controlled Variable Value (the Error) must exceed before a TO-OFFNORMAL event is generated. For example: 5

Property	Value Format	Example
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	This property, of type BACnetEventTransitionBits, shall convey three flags that separately enable and disable reporting of TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. This property is required if intrinsic reporting is supported by this object. For example: 101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	This property, of type BACnetEventTransitionBits, shall convey three flags that separately indicate the receipt of acknowledgments for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. For example: 111
NOTIFY-TYPE	Integer	This property, of type BACnetNotifyType, shall convey whether the notifications generated by the object should be Events or Alarms. This property is required if intrinsic reporting is supported by this object. For example: 21
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	This optional property, of type BACnetARRAY[3] of BACnetTimeStamp, shall convey the times of the last event notifications for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events, respectively. For example: 2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Profile: Loop

Notification (NOT)

Supported properties of the Notification object (NOT)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	15:0
OBJECT-NAME	String	Object Name: Notification Class
OBJECT-TYPE	Integer	15
DESCRIPTION	String	This sensor monitors the core activity For example: Description: Notification Class
NOTIFICATION-CLASS	Integer	This property, of type Unsigned, shall indicate the numeric value of this notification class and shall be equal to the instance number of the Notification Class object. For example: 0
PRIORITY	String	This property, of type BACnetARRAY[3] of Unsigned, shall convey the priority to be used for event notifications for TOOFFNORMAL, TO-FAULT, and TO-NORMAL events, respectively. For example: 1|0|1
ACK-REQUIRED	String	This property, of type BACnetEventTransitionBits, shall convey three separate flags that represent whether acknowledgment shall be required in

<table>
<tr><th>Property</th><th>Value Format</th><th>Example</th></tr>
<tr><td></td><td></td><td>notifications generated for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL event transitions, respectively.
For example: 110</td></tr>
<tr><td>RECIPIENT-LIST</td><td>String
<code>[VALID_DAYS]|[FROM_TIME TO_TIME]|[RECIPIENT (Device or Address)]|[PROCESS_IDENTIFIER]|[ISSUE_CONFIRMED_NOTIFICATIONS]|[TRANSITIONS];…</code></td><td>This property, of type List of BACnetDestination, shall convey a list of one or more recipient destinations to which notifications shall be sent when event-initiating objects using this class detect the occurrence of an event.
For example: 1111000|15:21:35.0 0:0:0.0|1,0,88 |0|0|010;1111000|0:0:0.0 0:0:0.0|0,0,0|0|0|111;1111000|0:0:0.0 0:0:0.0|1,1,99 |0|0|111…</td></tr>
<tr><td>PROFILE-NAME</td><td>String</td><td>Profile Name: Notification Class</td></tr>
</table>

Program (PRG)

Supported properties of the Program object (PRG)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	16:0
OBJECT-NAME	String	Object Name: Program Object Type (PRG)
OBJECT-TYPE	Integer	16
PROGRAM-STATE	Integer • 0 - IDLE • 1 - LOADING • 2 - RUNNING • 3 - WAITING • 4 - HALTED • 5 - UNLOADING	This property, of type BACnetProgramState, reflects the current logical state of the process executing the application program this object represents. This property is Read-Only. For example: 3
PROGRAM-CHANGE	Integer • 0 - READY • 1 - LOAD • 2 - RUN • 3 - HALT • 4 - RESTART • 5 - UNLOAD	This property, of type BACnetProgramRequest, is used to request changes to the operating state of the process this object represents. The Program Change property provides one means for changing the operating state of this process. For example: 4
REASON-FOR-HALT	Integer • 0 - NORMAL • 1 - LOAD FAILED • 2 - INTERNAL • 3 - PROGRAM • 4 - OTHER	If the process executing the application program this object represents encounters any type of error that causes process execution to be halted, then this property shall reflect the reason why the process was halted. The Reason For Halt property shall be an enumerated type called BACnetProgramError. For example: 1

Property	Value Format	Example
DESCRIPTION-OF-HALT	String	This property is a character string that may be used to describe the reason why a program has been halted. For example: ProgramA Halt
PROGRAM-LOCATION	String	This property is a character string that may be used by the application program to indicate its location within the program code, for example, a line number or program label or section name. The content of this string is a local matter. For example: ProgramA
DESCRIPTION	String	This property is a string of printable characters that may be used to describe the application being carried out by this process or other locally desired descriptive information. For example: Description: Program Object Type (PRG)
INSTANCE-OF	String	This property is a character string that is the local name of the application program being executed by this process. The content of this string is a local matter. For example: Program Object Type (PRG) Instance
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	This property, of type BACnetStatusFlags, represents four Boolean flags that indicate the general "health" of the program. For example: 1010
RELIABILITY	Integer	The Reliability property, of type BACnetReliability, provides an indication of whether the application-specific properties of the program object or the process executing the application program are "reliable" as far as the BACnet Device can determine and, if not, why. For example: 9
OUT-OF-SERVICE	Boolean	The Out Of Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the process this object represents is not in service. For example: 1
PROFILE-NAME	String	Profile Name: Program Object Type (PRG)

Pulse Converter (PC)

Supported properties of the Pulse Converter object (PC)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]: [Instance Number]`	24:0
OBJECT-NAME	String	Object Name: Pulse Converter Object Type (PC)
OBJECT-TYPE	Integer	24
DESCRIPTION	String	Description: Pulse Converter Object Type (PC)
PRESENT-VALUE	Float	This property, of type REAL, indicates the accumulated value of the input being measured. It is computed by multiplying the current value of the Count property by the value of the Scale Factor property.

Property	Value Format	Example
		For example: 100
INPUT-REFERENCE	String `[OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX (Optional)]`	This optional property, of type BACnetObjectPropertyReference, indicates the object and property (typically an Accumulator object's Present Value property) representing the actual physical input that is to be measured and presented by the Pulse Converter object. For example: 29,4194303|29|786
STATUS-FLAGS	4 Boolean values (1 = true, 0 = false) encoded as string `{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}`	This property, of type BACnetStatusFlags, represents four Boolean flags that indicate the general "health" of a Pulse Converter. For example: 1101
EVENT-STATE	Integer	The Event State property, of type BACnetEventState, is included in order to provide a way to determine if this object has an active event state associated with it. If the object supports intrinsic reporting, then the Event State property shall indicate the event state of the object. For example: 1
RELIABILITY	Integer	The Reliability property, of type BACnetReliability, provides an indication of whether the Present Value and/or Count properties or the operation of the physical input in question is "reliable" as far as the BACnet Device or operator can determine and, if not, why. For example: 5
OUT-OF-SERVICE	Boolean	The Out Of Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the input that the object directly represents, if any, is not in service. For example: 1
UNITS	Integer	This property, of type BACnetEngineeringUnits, indicates the measurement units of the Present Value property. See the BACnetEngineeringUnits ASN.1 production in Clause 21 for a list of engineering units defined by this standard. For example: 7
SCALE-FACTOR	Float	This property, of type REAL, provides the conversion factor for computing Present Value. It represents the change in Present Value resulting from changing the value of Count by one. For example: 70
ADJUST-VALUE	Float	This property, of type REAL, is written to adjust the Present Value property (and thus the Count property also) by the amount written to Adjust Value. For example: 10
COUNT	Integer	This read-only property, of type Unsigned, indicates the count of the input pulses as acquired from the physical input or the property referenced by the Input Reference property. For example: 20
UPDATE-TIME	String	This read-only property, of type BACnetDateTime, reflects the date and time of the most recent change to the Count property as a result of input pulse accumulation and is updated atomically with the Count property.

Property	Value Format	Example
		If no such change has yet occurred, this property shall have wildcard values for all date and time fields. For example: 29-5-2014 17:6:24.78
COUNT-CHANGE-TIME	String	This read-only property, of type BACnetDateTime, represents the date and time of the most recent occurrence of a write to the Adjust Value property. If no such write has yet occurred, this property shall have wildcard values for all date and time fields. For example: 29-5-2014 17:6:24.78
COUNT-BEFORE-CHANGE	Integer	This property, of type Unsigned, indicates the value of the Count property just prior to the most recent write to the Adjust Value properties. If no such write has yet occurred, this property shall have the value zero. For example: 0
COV-INCREMENT	Float	This property, of type REAL, shall specify the minimum change in Present Value that will cause a COV notification to be issued to subscriber COV-clients. This property is required if COV reporting is supported by this object. For example: 30
NOTIFICATION-CLASS	Integer	This property, of type Unsigned, shall specify the notification class to be used when handling and generating event notifications for this object. For example: 25
TIME-DELAY	Integer	This property, of type Unsigned, shall specify the minimum period of time in seconds that the Present Value must remain outside the band defined by the High Limit and Low Limit properties before a TO-OFFNORMAL event is generated or remain within the same band, including the Deadband property, before a TO-NORMAL event is generated. This property is required if intrinsic reporting is supported by this object. For example: 80
HIGH-LIMIT	Float	This property, of type REAL, shall specify a limit that the Present Value must exceed before an event is generated. This property is required if intrinsic reporting is supported by this object. For example: 50
LOW-LIMIT	Float	This property, of type REAL, shall specify a limit below which the Present Value must fall before an event is generated. This property is required if intrinsic reporting is supported by this object. For example: 60
DEADBAND	Float	This property, of type REAL, shall specify a range between the High Limit and Low Limit properties, which the Present Value must remain within for a TO-NORMAL event to be generated under these conditions: • the Present Value must fall below the High Limit minus Deadband, and • the Present Value must exceed the Low Limit plus the Deadband, and • the Present Value must remain within this range for a minimum period of time, specified in the Time Delay property, and

Property	Value Format	Example
		• either the HighLimitEnable or LowLimitEnable flag must be set in the Limit Enable property, and • the TO-NORMAL flag must be set in the Event Enable property This property is required if intrinsic reporting is supported by this object. For example: 40
LIMIT-ENABLE	2 Boolean values (1 = true, 0 = false) encoded as a string `{lowLimitEnable, highLimitEnable}`	This property, of type BACnetLimitEnable, shall convey two flags that separately enable and disable reporting of high limit and low limit offnormal events and their return to normal. This property is required if intrinsic reporting is supported by this object. For example: 10
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	This property, of type BACnetEventTransitionBits, shall convey three flags that separately enable and disable reporting of TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. In the context of Pulse Converter objects, transitions to the High Limit or Low Limit Event States are considered to be "offnormal" events. This property is required if intrinsic reporting is supported by this object. For example: 101
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	This property, of type BACnetEventTransitionBits, shall convey three flags that separately indicate the receipt of acknowledgements for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. For example: 111
NOTIFY-TYPE	Integer	This property, of type BACnetNotifyType, shall convey whether the notifications generated by the object should be Events or Alarms. This property is required if intrinsic reporting is supported by this object. For example: 2
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	This optional property, of type BACnetARRAY[3] of BACnetTimeStamp, shall convey the times of the last event notifications for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events, respectively. For example: 2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Profile Name: Pulse Converter Object Type (PC)

Group (GRP)

Supported properties of the Group object (GRP)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	11:0
OBJECT-NAME	String	Object Name: Group Object Type (GRP)
OBJECT-TYPE	Integer	11

Property	Value Format	Example
DESCRIPTION	String	Description: Group Object Type (GRP)
LIST-OF-GROUP-MEMBERS	String `[OBJECT-IDENTIFIER]-[LIST-OF-PROPERTY-REFERENCES (Optional)]|...`	This property is a list of one or more read access specifications, which defines the members of the group that shall be referenced when this object is specified in a protocol transaction. Each read access specification shall consist of two parts: 1) an Object Identifier and 2) a List Of Property References. For example: 0:0-0:10,0,0|0:0|0:0| 0:0-0:10,5:25,0|0:0|0:0-2:30,3:30
PRESENT-VALUE	String `[OBJECT-IDENTIFIER]-[LIST-OF-RESULTS (Optional)]|...`	This property is a list that contains the values of all the properties specified in the List Of Group Members. This is a "read only" property; it cannot be used to write a set of values to the members of the group. For example: 0:0-0:10'2'50,0:111,0:111|0:0|0:0| 0:0-0:10'2'50,5:25'2'32,0:111|0:0|0:0-2:30'2'32,3:30'2'32
PROFILE-NAME	String	Profile Name: Group Object Type (GRP)

File (FO)

Supported properties of the File object (FO)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	10:0
OBJECT-NAME	String	Object Name: File Object Type (FO)
OBJECT-TYPE	Integer	10
DESCRIPTION	String	Description: File Object Type (FO)
FILE-TYPE	String	This property, of type CharacterString, identifies the intended use of this file. For example: JPEG
FILE-SIZE	Integer	This property, of type Unsigned, indicates the size of the file data in octets. For example: -1
MODIFICATION-DATE	String	This property, of type BACnetDateTime, indicates the last time this object was modified. A File object shall be considered modified when it is created or written to. For example: 29-5-2014 17:6:24.78
ARCHIVE	Boolean	This property, of type BOOLEAN, indicates whether the File object has been saved for historical or backup purposes. For example: 1
READ-ONLY	Boolean	This property, of type BOOLEAN, indicates whether (FALSE) or not (TRUE) the file data may be changed through the use of a BACnet AtomicWriteFile service. For example: 0

Property	Value Format	Example
FILE-ACCESS-METHOD	Integer	This property, of type BACnetFileAccessMethod, indicates the type(s) of file access supported for this object. The possible values for File_Access_Method are: {RECORD_ACCESS, STREAM_ACCESS}. For example: 1
RECORD-COUNT	Integer	This property, of type Unsigned, indicates the size of the file data in records. For example: 0
PROFILE-NAME	String	Profile Name: File Object Type (FO)

Trend Log (TL)

Supported properties of the Trend Log object (TL)

Property	Value Format	Example
OBJECT-IDENTIFIER	String `[ObjectNumber]:[Instance Number]`	20:4194302
OBJECT-NAME	String	Object Name: Trend Log Object Type (TL)
OBJECT-TYPE	Integer	20
DESCRIPTION	String	Description: Trend Log Object Type (TL)
LOG-ENABLE	Boolean	This property, of type BOOLEAN, indicates and controls whether (TRUE) or not (FALSE) logging is enabled. For example: 1
START-TIME	String	This property, of type BACnetDateTime, specifies the date and time at or after which logging shall be enabled by this property. For example: 29-5-2014 17:6:24.78
STOP-TIME	String	This property, of type BACnetDateTime, specifies the date and time at or after which logging shall be disabled by this property. For example: 29-5-2014 17:6:24.78
LOG-DEVICE-OBJECT-PROPERTY	String `[OBJECT-IDENTIFIER] | [PROPERTY-IDENTIFIER] | [PROPERTY-ARRAY-INDEX (Optional)] | [DEVICE-IDENTIFIER (Optional)]`	This property, of type BACnetDeviceObjectPropertyReference, specifies the Device Identifier, Object Identifier and Property Identifier of the property to be trend logged. For example: 29,4194303|29|786|30,4194303
LOG-INTERVAL	Unsigned Integer	This property, of type Unsigned, specifies the periodic interval in hundredths of seconds for which the referenced property is to be logged. For example: 100
COV-RESUBSCRIPTION-INTERVAL	Unsigned Integer	If the Trend Log is acquiring data from a remote device by COV subscription, this property, of type Unsigned, specifies the number of seconds between COV resubscriptions, provided that COV subscription is in effect.

Property	Value Format	Example
		For example: 300
CLIENT-COV-INCREMENT	Float	If the Trend Log is acquiring COV data, this property, of type BACnetClientCOV, specifies the increment to be used in determining that a change of value has occurred. For example: 1.44
STOP-WHEN-FULL	Boolean	This property, of type BOOLEAN, specifies whether (TRUE) or not (FALSE) logging should cease when the buffer is full. When logging ceases, Log Enable shall be set FALSE. For example: 1
BUFFER-SIZE	Unsigned Integer32	This property, of type Unsigned32, shall specify the maximum number of records the buffer may hold. If writable, it may not be written when Log Enable is TRUE. The disposition of existing records when Buffer Size is written is a local matter. For example: 20
RECORD-COUNT	Unsigned Integer32	This property, of type Unsigned32, shall represent the number of records currently resident in the log buffer. A write of the value zero to this property shall cause all records in the log buffer to be deleted and Records Since Notification to be reset to zero. Upon completion, this event shall be reported in the log as the initial entry. For example: 20
TOTAL-RECORD-COUNT	Unsigned Integer32	This property, of type Unsigned32, shall represent the total number of records collected by the Trend Log object since creation. For example: 72
NOTIFICATION-THRESHOLD	Unsigned Integer32	This property, of type Unsigned32, shall specify the value of Records Since Notification at which notification occurs. This property is required if intrinsic reporting is supported by this object. For example: 100
RECORDS-SINCE-NOTIFICATION	Unsigned Integer32	This property, of type Unsigned32, represents the number of records collected since the previous notification, or since the beginning of logging if no previous notification has occurred. This property is required if intrinsic reporting is supported by this object. For example: 72
LAST-NOTIFY-RECORD	Unsigned Integer32	This property, of type Unsigned32, represents the SequenceNumber associated with the most recently collected record whose collection triggered a notification. For example: 0
EVENT-STATE	Integer	The Event State property, of type BACnetEventState, is included in order to provide a way to determine if this object has an active event state associated with it. For example: 0
NOTIFICATION-CLASS	Unsigned Integer	This property, of type Unsigned, shall specify the notification class to be used when handling and generating event notifications for this object. For example: 0

Property	Value Format	Example
EVENT-ENABLE	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	This property, of type BACnetEventTransitionBits, shall convey three flags that separately enable and disable reporting of TO-FAULT and TO-NORMAL events. For example: 010
ACKED-TRANSITIONS	3 Boolean values (1 = true, 0 = false) encoded as a string `{to-offnormal, to-fault, to-normal}`	This property, of type BACnetEventTransitionBits, shall convey three flags that separately indicate the receipt of acknowledgments for TO-OFFNORMAL, TO-FAULT and TO-NORMAL events. For example: 111
NOTIFY-TYPE	Integer	This property, of type BACnetNotifyType, shall convey whether the notifications generated by the object should be Events or Alarms. This property is required if intrinsic reporting is supported by this object. For example: 2
EVENT-TIME-STAMPS	Sequence of time stamps encoded as a string	This optional property, of type BACnetARRAY [3] of BACnetTimeStamp, shall convey the times of the last event notifications for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events, respectively. For example: 2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0;2|1-1-1970 11:0:0.0
PROFILE-NAME	String	Profile Name: Trend Log Object Type (TL)

Notes

This OI Server supports Read-Property-Multiple, Read-Property-Single and Write-Property-Single application services.

Supported Data Types

The data type is specified as a suffix in the item syntax. This OI Server supports the following data types.

Enumerations

The enumerations are associated with textual values on the properties that use this data type. This OI Server returns the numeric value of these enumerations on the tags used to read them. The following enumerations are currently used. The BACnetEngineeringUnits enumeration is also used, on the UNITS property.

Enumeration	Values
BACnetEventState	• normal (0) • fault (1) • offnormal (2) • high-limit (3) • low-limit (4) • life-safety-alarm (5)
BACnetNotifyType	• alarm (0) • event (1) • ack-notification (2)
BACnetPolarity	• normal (0) • reverse (1)

Enumeration	Values
BACnetBinaryPV	• inactive (0) • active (1)
Tag Number	• NULL (0) • Boolean (1) • Unsigned Integer (2) • Integer (3) • Real (4) • String (7) • Enumerated (9)

Dates

On this OI Server, single dates are always represented as a 3-part string with the following format:

DD-MM-YYYY

Where...

DD

is the number of the day (1 to 31)

MM

is the month number (1 to 12)

YYYY

is the year (1900 to 2154)

Dates might be composed to form ranges, and/or with times form a specific moment or range. Using values outside of the ranges specified here will produce unspecified behavior.

Times

On this OI Server, single times are always represented as a 4-part string with the following format:

HH:MM:SS.HS

Where...

HH

is the hour (0 to 23)

MM

is the minute (0 to 59)

SS

is the second (0 to 59)

HS

is the hundredth of second (0 to 99)

Dates might be composed to form ranges, and/or with times form a specific moment or range. Using values outside of the ranges specified here will produce unspecified behavior.

Schedule Object

The schedule object has special properties that demand further explanations on its syntax. The effective period, weekly schedule and exception schedule are these properties, combining dates and times on ranges and periods.

Effective Period:

The tag must have a value complying to the following syntax:

```
<start date>|<end date>
```

Where…
<start date>|<end date>

is a date as specified before, on the data types section. As the effective period property implies a period, two dates must be supplied, separated by the | (pipe) character.

For example a period starting at Apr/12/2010 and ending at May/14/2010 shall be:

```
12-4-2010|14-5-2010
```

Weekly Schedule:

The weekly schedule is used to specify times and values for specific days. This property combines times with enumerations for types and values, onto an array of 7 elements (corresponding to the 7 days of a week). Each element of the array may also have multiple times. Starting with a single time instance, the following syntax is used:

```
<time instance> = <time>,<type>,<value>
```

Where…
<time>

is a time as specified on data types section

<type>

is one of the elements of the Tag Number enumeration

<value>

is the value of *<type>* associated with this time

A single day may contain zero or more time instances. Those are divided by a semi-colon character, forming a daily schedule element:

```
<daily schedule> = [<time instance>;<time instance>;...]
```

This syntax means that each day might have any number of time configurations on the schedule, using a semi-colon to separate them. The following example clarifies this syntax:

```
Time 12h 25m 32s 500ms       Type: Boolean(1)           Value: True (1)
Value for the day: 12:25:32.50,1,1

Time 13h 40m 3s 20ms         Type: Boolean (1)          Value: False(0)
Value for the day: 13:40:3.2,1,0
```

It is important to notice that the above example was only for one day but it is necessary to specify the seven days even though some of them may not be defined. So, the complete syntax of a weekly-schedule is:

```
<monday>|<tuesday>|<wednesday>|<thursday>|<friday>|<saturday>|<sunday>
```

Each of the days of the week is a <daily schedule>, which means it does not need to have a value.

Example:

The complete and valid value for the weekly-schedule:

```
12:34:56.99,4,5.12;22:10:0.0,2,135|||4:5:12.0,1,1|||1:2:3.4,7,sample string
```

In this example we have the seven days, with schedules for Monday, Thursday and Sunday, summarized on the table:

Day of Week	Time of Day	Type	Value
Monday	12:34:56.99	4 (Real)	5.12
Monday	22:10:0.0	2 (Uint)	135
Thursday	4:5:12.0	1 (Boolean)	TRUE
Sunday	1:2:3.4	7 (String)	sample string

The value configured on the schedule will be read on the PRESENT-VALUE address of the object on its time of activity.

Exception Scheduler:

The Exception Scheduler values are built with a sequence of groups. Each group is composed of a date, a list of time and values and a priority value. The list of time_value are separated by a comma (,), whereas the parts of the group are separated by a semi-colon (;) character:

```
<group> = <date>;[<time_value>,< time_value>,...];<priority>
```

Notice that <date> syntax is specified on the data types section and the priority is an unsigned integer number. The <time_value> is composed of a time and a value separated by (/) . The value is composed of a datatype number (see datatype table) and a value separated by (:).

Example of a time 10:15:30.50.

Example of a value like unsigned integer 10 will be 2:10 (2 is the datatype number of the unsigned integer)Now the time_value for this will be 10:15:30.50/2:10

The first group differs from the others only by the date part:

```
<first group> = <start date>,<end date>;[<time_value>,<time_value>,...];<priority>
```

The complete syntax of the exception scheduler is composed of the first group and zero or more groups, divided by the | (pipe) character:

```
<first group>[|<group>|<group>|<group>...]
```

Example of an exception scheduler with one group:

```
26-02-2013;12:1:2.0/2:10;0
```

Example of a more complex exception scheduler with two groups:

```
26-2-2013;12:1:2.0/2:10;0|0-0-1900;0:0:0.0/0:NULL,0:0:0.0/1:0,0:0:0.0/7:BacneTest;0
```

Example of an exception scheduler with starting & ending date with one group:

```
16-11-2013,29-11-2013;3:10:0.0/1:1,11:0:0.0/1:0,14:50:0.0/1:1,23:59:59.0/1:0;2|
```

DataType Table

DataType Number	DataType Description	Example of usage with value
0	NULL	0:NULL
1	Boolean	1:1 (1 is true, 0 is false)
2	Unsigned	2:45
3	Signed	3:-34
4	Real	4:5.6
5	Double	5:5666
6	Octect String	6:12 23 34
7	String	7:Test
8	Bit String	8:10101010
9	Enumerated	9:1
10	Date	10:21-01-2013
11	Time	11:12:12:12.5
12	Object Identifier	12:1:1 1:1 (analog output,instance 1) See table ObjectIdentifier definitions.

Object Identifier Definitions Reference

Object	Identifier
ACCUMULATOR	23
ANALOG_INPUT	0
ANALOG_OUTPUT	1
ANALOG_VALUE	2
AVERAGING	18
BINARY_INPUT	3
BINARY_OUTPUT	4
BINARY_VALUE	5
CALENDAR	6
COMMAND	7
DEVICE	8
EVENT_ENROLLMENT	9
FILE	10
GROUP	11
LIFE_SAFETY_POINT	21
LIFE_SAFETY_ZONE	22
LOOP	12
MULTI_STATE_INPUT	13
MULTI_STATE_OUTPUT	14
MULTI_STATE_VALUE	19

Object	Identifier
NOTIFICATION_CLASS	15
PROGRAM	16
PULSE_CONVERTER	24
SCHEDULE	17
TREND_LOG	20

BACnetPriorityArray

The BacnetPriorityArray data type is represented as an array of 16 priority values. This OI Server supports NULL values and unsigned values on these fields.To write on a priority array data type you must write a Present-Value specifying the priority, please look at the page 28 for more information.The read operations can be performed directly on the property of this type.

The format for displaying this datatype is:

```
<Value for Priority1>|< Value for Priority2>|< Value for Priority3>|. . .|< Value for
 Priority16>
Example:
          5|1|30|0|0|0|0|0|0|0|0|0|0|0|0|0
```

The meaning of each priority level is described on the table below.

Priority Level	Application
1	Manual-Life Safety
2	Automatic-Life Safety
3	Available
4	Available
5	Critical Equipment Control
6	Minimum On/Off
7	Available
8	Manual Operator
9	Available
10	Available
11	Available
12	Available
13	Available
14	Available
15	Available
16	Available

BACnetDateTime

The BacnetDateTime is represented as an date and a time, following this format:

```
dd-mm-yyyy hh:mm:ss.ms
Example:
        12-12-2012 12:12:12.4
```

Examples of Item References

These are examples of valid item references for this OI Server. For more information about the referenced addresses, see the manufacturer's documentation for your device.

Device Address	Item Reference
Analog Input 0, Object Name	`AI:0:OBJECT-NAME`
Analog Input 0, Present Value	`AI:0:PRESENT-VALUE`
Analog Input 5, Present Value	`AI:5:PRESENT-VALUE`
Analog Value 10, Description	`AV:10:DESCRIPTION`
Schedule 1, Weekly	`SCH:1:WEEKLY-SCHEDULE`

BACLITE OI Server Error Codes

The following tables describe the additional error codes that you might receive when poll/poke requests and operations fail.

Code	Description	Possible Causes	Solution
0	OK	Communication without problems.	None.
1	Invalid operation	Invalid item reference.	Check the item reference.
2	Malformed packet	• Disconnected cables. • PLC is turned off, in stop mode, or in error mode. • Wrong station number.	• Check cable wiring. • Check PLC state. It must be RUN. • Check station number.
3	Wrong link	• Wrong link setting on the driver settings station.	Check the driver settings station.
4	Could not create receiver sink	Restart the driver.	Restart the driver.
5	Termination error	• Wrong station number. • Invalid value in **Station** field.	• Check the station number. • Check the **Station** field configuration.
6	Unsupported BVLL function	• Disconnected cables. • PLC turned off, or in Stop or Error mode. • Wrong station number.	• Check the cable wiring • Check the PLC state (it must be RUN) • Check the station number
8	Unknown APDU type	Invalid request.	Use a different BACnet application service.
9	Object not found	The requested object is not found on the device.	Check on the device if the object exists.
10	Invalid station on driver settings	The format of the station is incorrect.	See the section of station to learn how to validate the station.
11	Invalid MAC on driver settings	The MAC specified on the driver settings is invalid.	
12	Invalid UDP Port on driver settings	Port number is invalid.	Check the port number of the driver.
13	Device is unavailable	Device is not accessible.	Check the device status.
14	Unknown object	The object does not exists.	• Check the objects on the device.
15	Unknown property	The property does not exists.	Check the properties on the device.
16	Write denied	You are not allowed to write on the property.	Check the property is it is really writeable.
17	Write error	Error while writing.	Check the device status.
18	Invalid format	The value typed is invalid.	Check the format of the value typed.

Code	Description	Possible Causes	Solution
0	OK	Communicating without error.	None required.
-15	Timeout waiting for message to start	• Disconnected cables. • PLC is turned off, in stop mode, or in error mode. • Wrong station number. • Wrong parity (for serial communication).	• Check cable wiring. • Check the PLC mode — it must be RUN. • Check the station number. • Increase the timeout in the driver's advanced settings.

Code	Description	Possible Causes	Solution
		• Wrong RTS/CTS configuration (for serial communication).	• Check the RTS/CTS configuration (for serial communication).