# NB-ASC(e)
# User Manual

AMERICAN AUTO-MATRIX
SMART BUILDING SOLUTIONS®

*NB-ASC(e) User Manual*

Part Number 1E-04-00-0114



## World Headquarters

American Auto-Matrix
One Technology Lane
Export, Pennsylvania 15632-8903  USA
Tel (1) 724-733-2000
Fax (1) 724-327-6124
Email aam@aamatrix.com
www.aamatrix.com

***5/15/2013***

- Corresponds with v6.10 Firmware Release
- Added STATbus Reset Count property to Device Object of Object and Properties Appendix Sections

***4/10/2013***

- Corresponds with v6.09 Firmware Release
- Updated various portions Sections 3 through 6, and Appendix A through C to reflect support for new deadband and min run/off timer properties added as part of v6.09 firmware.
- Updated Section 2.4 - Changed name to SBC-STAT Support. All references for wiring and installation are now referred to the STAT User Manual (1E-04-00-0103).
- Updated overall document to support PDF bookmarks with hyperlinks to topics, as well as Table of Contents hyperlinks.

***4/4/2012***

- Updated Section 1 - Product Overview. Includes information regarding support for single SBC-RHT sensor applications.
- Updated Section 2 - Wiring & Installation. Includes information regarding support for single SBC-RHT sensor applications.
- Updated Section 5 - Reversing Valve object. Includes new properties corresponding to Defrost Cycle application introduced with v6.08 firmware release.
- Updated Appendix A - Objects & Properties. Includes property information for SBC-RHT support.
- Updated Appendix B - Objects & Properties. Includes property information for SBC-RHT support, and Defrost Cycle application notes.
- Updates to Appendices A, B, C to include Zone Temperature Object
- Added Present Occupancy Status property

***10/20/2009***

Manual Relaunch - Coincides with BTL listing of the product. All previous revisions of documentation are discontinued.

This manual describes the installation and operation of the NB-ASC(e). This document is divided into the following sections, each beginning with a table of contents for the section:

- One: Product Overview, describing the features of the NB-ASC(e) and presenting the specifications for the controller.
- Two: Wiring & Installation, detailing the wiring and installation procedures.
- Three: Fundamental Concepts, listing rudimentary concepts of American Auto-Matrix technology.
- Four: Product Configuration for Rooftop Mode.
- Five: Product Configuration for Heatpump Mode.
- Six: Product Configuration for Fancoil Mode.
- Appendix A - Rooftop Object and Property Reference.
- Appendix B - Heatpump Object and Property Reference.
- Appendix C - Fancoil Object and Property Reference.

**NOTE**

Notes indicate noteworthy information and appear in boxes with this format separated from the running text.

**CAUTION**

*Cautions indicate information that may prevent serious system or user problems and appear in boxes with this format separated from the running text.*

**WARNING**

***Warnings indicate inforamtion that prevent personal injury or equipment damage and appear in boxes with this format separated from the running text.***

1.1 Description .......... 1-3
1.2 Inputs .......... 1-7
1.2.1 UIs Configured as Digital Inputs .......... 1-7
1.2.2 UIs Configured for Linear Sensors .......... 1-7
1.2.3 UIs Configured for Thermistor Sensors .......... 1-7
1.3 Outputs .......... 1-8
1.3.1 Binary Outputs .......... 1-8
1.3.2 Analog Output .......... 1-8
1.4 STATbus .......... 1-9
1.5 Specifications .......... 1-10
1.5.1 Networking .......... 1-10
1.5.2 Terminations .......... 1-10
1.5.3 Input Supply .......... 1-10
1.5.4 Operating Environment .......... 1-10
1.5.5 Dimensions .......... 1-10
1.5.6 Agency Approvals .......... 1-10
2.1 Installation .......... 2-3
2.2 Mounting the Controller .......... 2-4
2.3 Wiring Requirements .......... 2-5
2.3.1 Connecting Power .......... 2-5
2.3.2 Universal Input Wiring .......... 2-6
2.3.2.1 Connecting Voltage Inputs to UIs .......... 2-6
2.3.2.2 Connecting 4-20mA Inputs to UIs .......... 2-6
2.3.2.3 Connecting Status Inputs to UIs .......... 2-7
2.3.3 Relay Output Wiring .......... 2-7
2.3.4 Analog Output Wiring .......... 2-7
2.4 SBC-STAT Support .......... 2-9
3.1 Fundamental Concepts .......... 3-3
3.1.1 BACnet MS/TP Protocol Information .......... 3-3
3.1.2 MS/TP Token Passing .......... 3-3
3.1.3 MS/TP LAN Wiring .......... 3-4
3.1.4 Device Addressing .......... 3-5
3.1.5 Communication Rates .......... 3-5
3.1.6 Network Optimization .......... 3-5
3.2 Command Prioritization .......... 3-6
4.1 Rooftop Sequence Overview .......... 4-3
4.2 Selecting Your Application Mode .......... 4-4
4.3 Universal Input Configuration .......... 4-5
4.3.1 Sensor Configuration .......... 4-5
4.3.1.1 Digital Inputs .......... 4-5
4.3.1.2 Linear Sensors (0-10VDC) .......... 4-5
4.3.1.3 4-20mA Sensors .......... 4-6
4.3.1.4 Thermistor Input .......... 4-6
4.3.1.5 Analog SBC-STAT1 .......... 4-7
4.4 Supply & Outside Temperature Configuration .......... 4-8

4.4.1 Supply/Discharge Temperature .................................................. 4-8
4.4.2 Outside Air Temperature .................................................. 4-8
4.4.3 Outside Air Temperature Broadcasts .................................................. 4-8
4.5 Output Configuration .................................................. 4-9
4.6 Fan Control .................................................. 4-10
4.6.1 Fan Properties .................................................. 4-10
4.6.2 Fan Status .................................................. 4-11
4.7 Cool 1 and 2 .................................................. 4-12
4.7.1 Lockout Configuration .................................................. 4-12
4.7.2 Cool 1 Properties .................................................. 4-12
4.7.3 Cool 2 Properties .................................................. 4-13
4.8 Heat 1 and 2 .................................................. 4-15
4.8.1 Lockout Configuration .................................................. 4-15
4.8.2 Heat 1 Properties .................................................. 4-15
4.8.3 Heat 2 Properties .................................................. 4-16
4.9 Economizer .................................................. 4-18
4.10 PID Control .................................................. 4-19
4.11 Scheduling .................................................. 4-21
4.11.1 Weekly Scheduling .................................................. 4-21
4.11.2 Broadcast Schedule .................................................. 4-21
4.11.3 Power-up State .................................................. 4-21
4.11.4 Host Override .................................................. 4-22
4.12 SBC-STAT Configuration .................................................. 4-23
4.12.1 Setpoint Adjustment Configuration .................................................. 4-23
4.12.2 User Override .................................................. 4-23
4.13 Occupancy Detection .................................................. 4-24
4.14 Pulse Counting .................................................. 4-25
5.1 Heatpump Sequence Overview .................................................. 5-3
5.2 Selecting Your Application Mode .................................................. 5-4
5.2.1 Configuring the Reversing Delay .................................................. 5-4
5.3 Universal Input Configuration .................................................. 5-5
5.3.1 Sensor Configuration .................................................. 5-5
5.3.1.1 Digital Inputs .................................................. 5-5
5.3.1.2 Linear Sensors (0-10VDC) .................................................. 5-5
5.3.1.3 4-20mA Sensors .................................................. 5-6
5.3.1.4 Thermistor Input .................................................. 5-6
5.3.1.5 Analog SBC-STAT1 .................................................. 5-7
5.4 Supply & Outside Temperature Configuration .................................................. 5-8
5.4.1 Supply/Discharge Temperature .................................................. 5-8
5.4.2 Outside Air Temperature .................................................. 5-8
5.4.3 Outside Air Temperature Broadcasts .................................................. 5-8
5.5 Output Configuration .................................................. 5-9
5.6 Fan Control .................................................. 5-10
5.6.1 Fan Properties .................................................. 5-10
5.6.2 Fan Status .................................................. 5-11

5.7 Reversing Valve and Defrost Cycle ... 5-12
5.7.1 Reversing Valve Properties ... 5-12
5.7.2 Defrost Cycle Properties ... 5-12
5.8 Stage 1 and 2 ... 5-14
5.8.1 Cooling Lockout Configuration ... 5-14
5.8.2 Heating Lockout Configuration ... 5-14
5.8.3 Stage 1 Properties ... 5-14
5.8.4 Stage 2 Properties ... 5-15
5.9 Auxiliary Heat ... 5-17
5.9.1 Auxiliary Heat Properties ... 5-17
5.10 Economizer ... 5-18
5.11 PID Control ... 5-19
5.12 Scheduling ... 5-21
5.12.1 Weekly Scheduling ... 5-21
5.12.2 Broadcast Schedule ... 5-21
5.12.3 Power-up State ... 5-21
5.12.4 Host Override ... 5-22
5.13 SBC-STAT Configuration ... 5-23
5.13.1 Setpoint Adjustment Configuration ... 5-23
5.13.2 User Override ... 5-23
5.14 Occupancy Detection ... 5-24
5.15 Pulse Counting ... 5-25
6.1 Fancoil Sequence Overview ... 6-3
6.2 Selecting Your Application Mode ... 6-4
6.3 Universal Input Configuration ... 6-5
6.3.1 Sensor Configuration ... 6-5
6.3.1.1 Digital Inputs ... 6-5
6.3.1.2 Linear Sensors (0-10VDC) ... 6-5
6.3.1.3 4-20mA Sensors ... 6-6
6.3.1.4 Thermistor Input ... 6-6
6.3.1.5 Analog SBC-STAT1 ... 6-7
6.4 Supply & Outside Temperature Configuration ... 6-8
6.4.1 Supply/Discharge Temperature ... 6-8
6.4.2 Outside Air Temperature ... 6-8
6.4.3 Outside Air Temperature Broadcasts ... 6-8
6.5 Output Configuration ... 6-9
6.6 Fan Control ... 6-10
6.6.1 Fan Properties ... 6-10
6.6.2 Fan Status ... 6-11
6.7 Electric Reheat ... 6-12
6.7.1 Electric Reheat Properties ... 6-12
6.8 Valve Control ... 6-13
6.8.1 Valve Control Properties ... 6-13
6.9 Economizer ... 6-15
6.10 PID Control ... 6-16

6.11 Scheduling ........ 6-18
6.11.1 Weekly Scheduling........ 6-18
6.11.2 Broadcast Schedule ........ 6-18
6.11.3 Power-up State ........ 6-18
6.11.4 Host Override ........ 6-19
6.12 SBC-STAT Configuration........ 6-20
6.12.1 Setpoint Adjustment Configuration........ 6-20
6.12.2 User Override........ 6-20
6.13 Occupancy Detection........ 6-21
6.14 Pulse Counting........ 6-22
A.1 Device........ A-3
A.2 Zone Temperature........ A-7
A.3 Universal Inputs 1-5........ A-12
A.4 Supply Temperature ........ A-13
A.5 Outside Air Temperature ........ A-14
A.6 Relative Humidity........ A-15
A.7 Analog Outputs 1-4........ A-16
A.8 Analog Values (Setpoints) 1-7........ A-17
A.9 Pulse Input........ A-18
A.10 BO01 ........ A-19
A.11 BO02 ........ A-21
A.12 BO03 ........ A-23
A.13 BO04 ........ A-24
A.14 BO05 ........ A-26
A.15 Schedule........ A-27
A.16 Economizer........ A-28
A.17 PID Control 1 - 4........ A-30
A.18 Occupancy Detection ........ A-33
A.19 Proof of Flow ........ A-34
A.20 Outside Air Temp. Broadcast ........ A-35
A.21 Broadcast Schedule ........ A-36
B.1 Device........ B-3
B.2 Zone Temperature........ B-7
B.3 Universal Inputs 1-5........ B-11
B.4 Supply Temperature ........ B-12
B.5 Outside Air Temperature ........ B-13
B.6 Relative Humidity........ B-14
B.7 Analog Outputs 1-4........ B-15
B.8 Analog Values (Setpoints) 1-7........ B-16
B.9 Pulse Input........ B-17
B.10 BO01 ........ B-18
B.11 BO02 ........ B-20
B.12 BO03 ........ B-22
B.13 BO04 ........ B-24
B.14 BO05 ........ B-26

B.15 Schedule .......... B-27
B.16 Economizer .......... B-28
B.17 Aux Heat .......... B-30
B.18 PID Control 1 - 4 .......... B-32
B.19 Occupancy Detection .......... B-35
B.20 Proof of Flow .......... B-36
B.21 Outside Air Temp. Broadcast .......... B-37
B.22 Broadcast Schedule .......... B-38
C.1 Device .......... C-3
C.2 Zone Temperature .......... C-7
C.3 Universal Inputs 1-5 .......... C-11
C.4 Supply Temperature .......... C-12
C.5 Outside Air Temperature .......... C-13
C.6 Analog Outputs 1-4 .......... C-14
C.7 Analog Values (Setpoints) 1-7 .......... C-15
C.8 Pulse Input .......... C-16
C.9 BO01 .......... C-17
C.10 BO02 .......... C-19
C.11 BO03 .......... C-20
C.12 BO04 .......... C-21
C.13 BO05 .......... C-22
C.14 Fan Speed .......... C-23
C.15 Schedule .......... C-24
C.16 Economizer .......... C-25
C.17 Reheat Ctrl .......... C-27
C.18 Valve Ctrl 1 and 2 .......... C-28
C.19 PID Control 1 - 4 .......... C-30
C.20 Occupancy Detection .......... C-32
C.21 Proof of Flow .......... C-33
C.22 Outside Air Temp. Broadcast .......... C-34
C.23 Broadcast Schedule .......... C-35

# SECTION 1: OVERVIEW

*This section provides a complete overview of the NB-ASC(e) controller product manufactured by American Auto-Matrix. Detailed functionality descriptions of each key feature of the NB-ASC(e) (inputs, outputs, etc.) is provided within this section.*

## IN THIS SECTION


Description ........ 1-3
Inputs ........ 1-7
UIs Configured as Digital Inputs ........ 1-7
UIs Configured for Linear Sensors ........ 1-7
UIs Configured for Thermistor Sensors ........ 1-7
Outputs ........ 1-8
Binary Outputs ........ 1-8
Analog Output ........ 1-8
STATbus ........ 1-9
Specifications ........ 1-10
Networking ........ 1-10
Terminations ........ 1-10
Input Supply ........ 1-10
Operating Environment ........ 1-10
Dimensions ........ 1-10
Agency Approvals ........ 1-10

# 1.1 DESCRIPTION

The Native Series (NB) BACnet unitary control modules are fundamental control devices in the American Auto-Matrix (AAM) System Architecture.

The NB-ASC(e) is a unitary controller designed for application selectable configuration through the use of flash technology. The NB-ASC(e) provides selectable applications for Rooftop, Heatpump, and Fancoil operations and has built-in control functions that correlate with the control applications including PID control, valve control, outside air temperature based economizer control, as well as many other functions that are reviewed in later sections of this manual.

Network communications occurs over the BACnet MS/TP protocol layer. The NB-ASC(e) is designed to operate in a stand-alone environment, or interact with other inter-networked devices. The device is configured using NB-Pro, a commissioning environment and engineering tool used to configure Native Series devices. This device can be programmed over the network, or in a direct-connect scenario.

The hardware layout of the NB-ASC(e) consists of a main module with removable terminal blocks to connect universal inputs, analog outputs, digital outputs, an optically isolated digital input for status or pulse counting, and a dedicated input for connecting a single or networked set of *SBC*-STAT family devices using AAM's revolutionary sensor network known as STATbus. Other key components include diagnostic LEDs for network indication and digital output status, configurable jumpers for configuring universal inputs, and a real-time clock module for stand-alone or time-master applications.

The NB-ASC(e) is available in two model variants:

*Table 1-1: NB-ASC(e) Models*

| Product Designation | BOs | AO | OI | UI | Real-time Clock |
|---|---|---|---|---|---|
| *NB-ASC* | 5 Relays | 2 | 0 | 2 | Optional - Field Installable and sold separately |
| *NB-ASC(e)* | 5 Relays | 4 | 1 | 5 | Integrated |

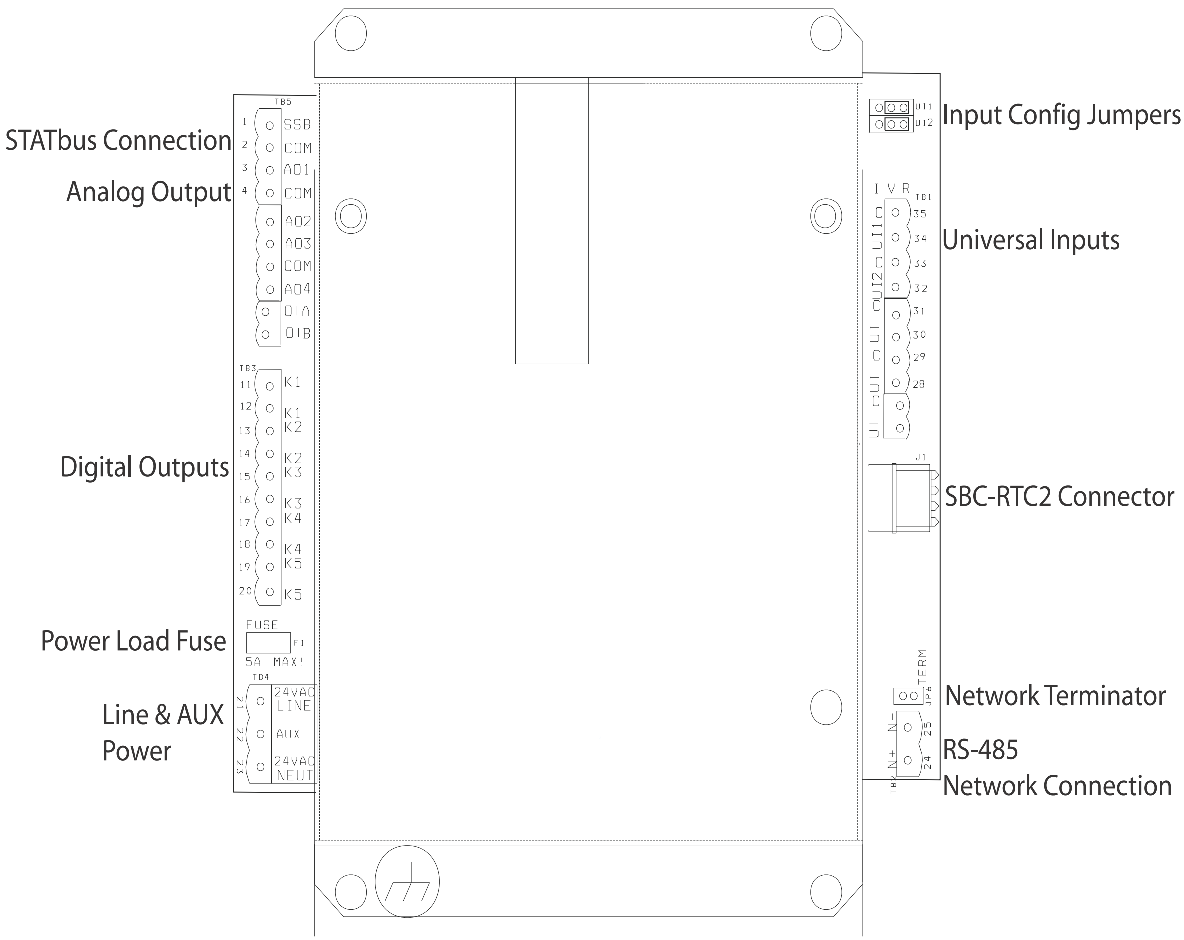


*Figure 1-1 NB-ASC Hardware Layout*

*Table 1-2: NB-ASC(e) Input/Output Board Assignments*

| Terminal | I/O | Description |
|---|---|---|
| 1 (TB5) | SSB | STATbus Network Port Lead |
| 2 (TB5) | COM | STATbus Network Port Common |
| 3 (TB5) | AO1 | Analog Output Channel 1 |
| 4 (TB5) | COM | AO Shared Common |
| 5 (TB5) | AO2 | Analog Output Channel 2 |
| 6 (TB5) | AO3 | Analog Output Channel 3 |
| 7 (TB5) | COM | AO Shared Common |
| 8 (TB5) | AO4 | Analog Output Channel 4 |
| 9(TB5) | OIA | Optically Isolated (Digital Input) Side A |
| 10 (TB5) | OIB | Optically Isolated (Digital Input) Side B |
| 11 (TB3) | K1 | Relay 1 Common |
| 12 (TB3) | K1 | Relay 1 Normally Open |
| 13 (TB3) | K2 | Relay 2 Common |
| 14 (TB3) | K2 | Relay 2 Normally Open |
| 15 (TB3) | K3 | Relay 3 Common |
| 16 (TB3) | K3 | Relay 3 Normally Open |
| 17 (TB3) | K4 | Relay 4 Common |
| 18 (TB3) | K4 | Relay 4 Normally Open |
| 19 (TB3) | K5 | Relay 5 Common |
| 20 (TB3) | K5 | Relay 5 Normally Open |
| 21 (TB4) | 24 VAC Line | 24 Volt AC Control Power Input |
| 22 (TB4) | AUX | 24 Volt AC Auxiliary Output for DC Loads (5A Fuse) |
| 23 (TB4) | 24 VAC NEUT | 24 Volt AC Neutral |
| 24 (TB2) | N+ | Positive 485 Network Communication Line |
| 25 (TB2) | N- | Negative 485 Network Communication Line |

*Table 1-2: NB-ASC(e) Input/Output Board Assignments*

| Terminal | I/O | Description |
|---|---|---|
| 26 (TB1) | UI5 | Universal Input 5 |
| 27 (TB1) | COM | Common |
| 28 (TB1) | UI4 | Universal Input 4 |
| 29 (TB1) | COM | Common |
| 30 (TB1) | UI3 | Universal Input 3 |
| 31 (TB1) | COM | Input Common |
| 32 (TB1) | UI2 | Universal Input 2 |
| 33 (TB1) | COM | Input Common |
| 34 (TB1) | UI1 | Universal Input 1 |
| 35 (TB1) | COM | Input Common |

# 1.2 INPUTS

Universal Inputs are high resolution (15-bit) UIs that can accept 0-20mA when in current mode, 0-1MΩ inputs when in resistance mode, or 0-10VDC when in voltage mode. A capacitor in the circuit provides a 10Hz low-pass filter. Over-range protection is provided to clamp normal over-range conditions and to protect against damage from electrostatic discharge (ESD). The UIs can be configured for alarming, setup/ setback, filtering, and input polarity.

All of the UIs have associated selection jumpers for selecting current, resistance and voltage modes. The procedure for configuring these jumpers in explained in further detail in Section 2: Wiring and Installation.

## 1.2.1 UIS CONFIGURED AS DIGITAL INPUTS

Each Universal Input may be programmed to behave as a digital (on/off) input. UIs configured as digital can be defined for contact closures or any two-state inputs that are needed in a particular application. Typical digital input applications that are supported by the device are fan status, proof of flow, and occupancy detection (dependent on sensor type).

Digital configured UIs support two different types of contact alarming (changes from 0 to1 or 1 to 0). Filtering options are also provided to program the amount of time required for input stability prior to alarming. The polarity of the digital input signal is also programmable within the input object.

## 1.2.2 UIS CONFIGURED FOR LINEAR SENSORS

Each Universal Input may be programmed as a linear scaled input for 0-10VDC input devices or 4-20mA input devices. Each of these types of devices may be programmed within minimum and maximum Engineering Units which define the end points of the analog values after they are read and converted to scaled Engineering Units.

For example, if you are using a linear, 4-20mA input device to measure carbon dioxide content (CO2) in the air, you must configure the minimum and maximum scaled values for that input as 0 ppm (parts-per-million) and 5,000 ppm.

Linearly scaled analog inputs have associated high and low alarm limit properties that may be setup and setback during unoccupied periods (programmable times when zone control can be less stringent), the result is a wider operating range during unoccupied periods. In addition, a hysteresis can be applied to each alarm definition to control nuisance alarms that may occur between transitions.

High and low limit alarms may be disabled for a programmable length of time after the controller has been reset or after power has been cycled.

## 1.2.3 UIS CONFIGURED FOR THERMISTOR SENSORS

Each Universal Input may be programmed to accept a Precon, type-3, 10k ohm thermistor. A built-in table is provide for approximate linearization. Inputs using Precon, type 3 thermistors will use temperature ranges of -30.0° to 230.0° F (-35.0° to 110.0° C). The temperature units (Fahrenheit or Celsius) can be configured in the FF00;**EM**.

Thermistor inputs can have associated high and low alarm properties that may be setup and setback during unoccupied periods (programmable times when zone control can be less stringent), The result of a wider operating range during unoccupied periods. In addition, a hysteresis can be applied to each alarm definition to control nuisance alarms that may occur between transitions.

High and low limit alarms may be disabled for a programmable length of time after the controller has been reset of after power has been cycled.

## 1.3 OUTPUTS

The NB-ASC(e) provides five (5) relays and up to four (4) Analog Outputs, dependent on model.

### 1.3.1 BINARY OUTPUTS

Binary Outputs provide On/Off control of output devices such as fans, valves, or cooling/heating stages. All BOs enforce minimum cycle time operation, determine the polarity, and provide a runtime alarm limit for the output. Relays have a 1A, 24VAC/DC rated load, normally open, non-polar contact. A tranzorb protection device is provided to suppress transients and contact arcing. Pulse width modulation (PWM) operation of floating valves or other devices is not recommended with relays.

**CAUTION**

Triacs are capable of switching a 1A, 24VDC load, but they will not turn off until the power load is removed.

**NOTE**

AAM recommends that output loads be wired so that one side of the load is grounded when possible.

### 1.3.2 ANALOG OUTPUT

The NB-ASC(e) includes up to four (4) Analog Outputs, capable of outputting a 0-10VDC signal at an 8-bit resolution. The Analog Output may be programmed to operate in either manual mode or automatic mode in association with a PID Control Loop, Economizer Control, or other program block in the controller.

## 1.4 STATBUS

This product includes one STATbus (SSB) port for connecting a single or a network of up to 4 digital *SBC*-STAT room sensors. STATbus is a polarity insensitive, two wire sensor bus that can be used in either analog or digital mode. In analog mode, NB-ASC(e) supports *SBC*-STAT1 and *SBC*-STAT2.

In digital mode, NB-ASC(e) supports digital *SBC*-STAT1D, *SBC*-STAT2D, and *SBC*-STAT3, as well as the ability to network up to four digital *SBC*-STAT sensors together on the same physical sensor bus. In networked *SBC*-STAT applications, the NB-ASC(e) has the option to control based on a collective average, highest reading, lowest reading, or one specific *SBC*-STAT.

For single thermostat applications, the Rooftop and Heatpump profiles of the NB-ASC(e) can utilize the SBC-RHT for temperature and relative humidity monitoring. Only a single SBC-RHT sensor may be connected to a single NB-ASC(e) deployed for Rooftop and Heatpump applications.

**NOTE**

Use of the SBC-RHT with this product is valid for single thermostat applications only. The NB-ASC(e) does not support the ability to network multiple SBC-RHT sensors, or a mixture of digital SBC-STAT sensors and SBC-RHT sensors.

For information on *SBC*-STAT products, reference the STAT User Manual for complete details.

**NOTE**

Analog *SBC*-STAT sensors cannot be networked together on the same sensor bus. If your application requires multiple zone sensors, you must use digital *SBC*-STAT senosrs.

*Table 1-3: SBC-STAT Sensors Supported by NB-ASC(e)*

| Analog *SBC*-STAT | Digital *SBC*-STAT |
| --- | --- |
| SBC-STAT1 | SBC-STAT1D |
| SBC-STAT2 | SBC-STAT2D |
| | SBC-STAT3D |
| | SBC-RHT<br>**(Rooftop and Heatpump only)** |

# 1.5 SPECIFICATIONS

## 1.5.1 NETWORKING

The following specifications are necessary for networking of the controller:

- **line signaling:** EIA-485
- **wiring:** shielded, twisted pair 18-22 AWG
- **network protection:** dual tranzorbs, Hi ESD driver
- **network configuration:** multidrop to 4,000ft. (1.5km) total
- **protocol:** BACnet MS/TP over EIA-485

## 1.5.2 TERMINATIONS

- Pluggable terminal blocks for inputs, outputs, power, and network connection.

## 1.5.3 INPUT SUPPLY

- NEC class 2 transformer (customer-supplied).
- 22-26VAC, 50/60Hz, 10VA maximum, 5VA typical.
- 5A fuse load protection.
- PTC control electronics protection.

## 1.5.4 OPERATING ENVIRONMENT

- **Temperature range:** 0–50°C.
- **Humidity range:** 0–95% RH non-condensing.

## 1.5.5 DIMENSIONS

- **Size:** 8.5 in. (21.6 cm) × 4.75 in. (14.6 cm) × 2.63 in. (6.7 cm) and 5.5 in (16.51 cm) × 4.75 in (14.6 cm) × 1.5 in. (3.81 cm)
- **Shipping weight:** 2.04 lbs (.93 kg) and 1.86 lbs (.84 kg)

## 1.5.6 AGENCY APPROVALS

- UL listed 916, Management Equipment, Energy (PAZX).
- UL 873 Component-Temperature-Indicating and Regulating Equipment (XAPX2).
- Complies with FCC rules Part 15, Class B Computing Device.
- Complies with CE directives and standards.
- BTL Listed - BACnet-Application Specific Controller (B-ASC), conforming to 135-2004 standard.

# SECTION 2: WIRING & INSTALLATION

*This section reviews general wiring and installation practices. Detailed information is given to many areas including wiring for power, communications, inputs/outputs, and important safety requirements.*

## IN THIS SECTION


Installation ........ 2-3
Mounting the Controller ........ 2-4
Wiring Requirements ........ 2-5
- Connecting Power ........ 2-5
- Universal Input Wiring ........ 2-6
  - Connecting Voltage Inputs to UIs ........ 2-6
  - Connecting 4-20mA Inputs to UIs ........ 2-6
  - Connecting Status Inputs to UIs ........ 2-7
- Relay Output Wiring ........ 2-7
- Analog Output Wiring ........ 2-7

SBC-STAT Support ........ 2-9

## 2.1 INSTALLATION

The installation of the controller involves mounting the controller, supplying power, connecting to the communications network, and connecting input and output devices. All wiring connections to the device are made with the use of plug (female) & socket (male) terminal blocks (TB). The plug consists of terminal ports and adjustment screws. Input/output device, network, and power wires enter the terminal ports and are secured to the assembly with the adjustment screws. The socket consists of a row of pins and is permanently mounted to the printed circuit board (PCB).

When connecting/disconnecting the two parts of the TB, align the holes on the plug with the pins on the socket and avoid twisting, thus damaging the assembly. Such damage will void the product warranty.

| WARNING |
| --- |
| The sockets to which the terminal block plugs connect are permanently attached to the PCB. Twisting or applying torque when connecting/ disconnecting will result in damage that will void the product warranty. |

## 2.2 MOUNTING THE CONTROLLER

To mount the controller, perform the following steps:

| CAUTION |
| --- |
| The area in which the controller is being mounted should be free from moisture, to prevent against short-circuiting of on-board components. |

1. If you are not using self-drilling mounting screws, use the controller backing as a template, mark the mounting holes on the mounting location. Remove the controller, then drill pilot holes in the mounting location. AAM recommends that at least two (2) screws be used to secure the NB-ASC(e) controller to the mounting location.
2. Align the mounting holes of the NB-ASC(e) controller with the pilot holes drilled in Step 1 and secure the controller to the mounting location using sheet metal screws. Sheet metal screws used to secure the controller to the mounting location are not supplied with the NB-ASC(e) controllers.

# 2.3 WIRING REQUIREMENTS

**WARNING**

For your safety, power should be removed when performing any type of wiring to the controller.

Follow the recommended wiring guidelines to reduce the chance of operation and communication errors. If you do not use proper wiring techniques, your site may not meet Federal Communications Commission (FCC) Class A regulations for radio frequency interference (RFI) emissions. All EIA-485 communications networks should employ shielded, twisted pair wiring. Each twisted pair must be individually shielded. Unshielded cables must be placed in solid metal conduit alone. Communications wiring (as well as *SBC*-STAT and other input wiring) **should not** be routed together with—or close to—other wiring carrying DC switching, AC lines, fluorescent lighting or any other RFI/electromagnetic interference (EMI)-emitting source. Failure to use these types of conductors may result in various system communications problems such as excessive network retries, noise susceptibility, and loss of communication.

## 2.3.1 CONNECTING POWER

You must use a 24VAC 50/60Hz NEC class II transformer rated at 10VA maximum (5VA typical) for power supply to the controller. AAM recommends that at least 18AWG wiring be used, but the terminals can accommodate 14–22AWG. Connect power to the 24VAC LINE (TB4: Terminal 21) and 24VAC NEUT (TB4: Terminal 23) of the controller.

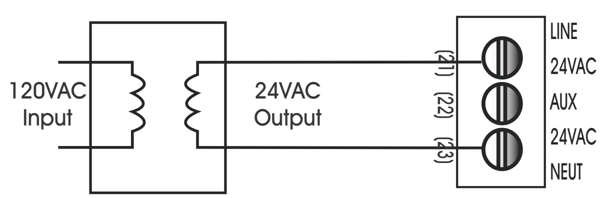


*Figure 2-1 Connecting Power*

**CAUTION**

AAM does not recommend sharing power transformers between devices. If this technique is used, AC polarity must be maintained throughout the power network. Product damage due to misapplication of power will void any warranty in place.

### 2.3.2 UNIVERSAL INPUT WIRING

The following section discusses common wiring applications for use with the controller. Deviations from the following examples should be discussed with AAM Technical Services prior to making modifications to a controller. Any modification other than those supported by AAM may void product warranty.

#### 2.3.2.1 CONNECTING VOLTAGE INPUTS TO UIS

To connect a voltage input (0 - 10VDC analog input device) to a UI, perform the following:

1. Remove the IVR jumper for the corresponding Universal Input.
2. Wire voltage input device to UI and COM terminals accordingly.

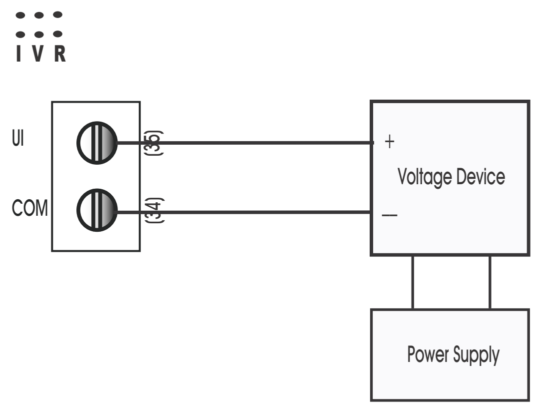


*Figure 2-2 Wiring a Voltage Input to a Universal Input*

#### 2.3.2.2 CONNECTING 4-20MA INPUTS TO UIS

To connect a 4-20mA input to a UI, perform the following:

1. Jumper the I and V pins for the corresponding Universal Input on the IVR jumper block.
2. Wire 4-20mA device to UI and COM terminals accordingly.

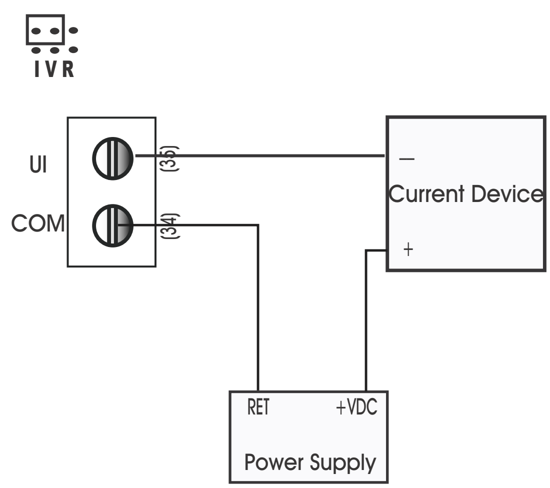


*Figure 2-3 Wiring a 4-20mA Input to a Universal Input*

#### 2.3.2.3 CONNECTING STATUS INPUTS TO UIS

To connect a status input (digital input) to a UI, perform the following:

1. Jumper the V and R pins for the corresponding Universal Input on the IVR jumper block.
2. Write digital input to UI and COM terminals accordingly.

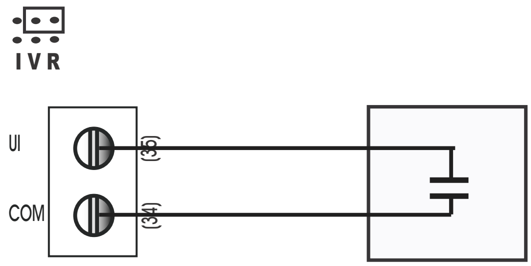


*Figure 2-4 Wiring a Digital Input to a Universal Input*

## 2.3.3 RELAY OUTPUT WIRING

This product uses relay outputs. Specifications of the outputs include:

- 24VAC/VDC, 1A rated load
- Isolated, normally open (Form 1A) contact
- Non-Polar

When wiring, connect one of the output load wires to either one of the relay output terminals, connect the remaining output load wire to a power source wire, then connect the other power source wire to the other remaining relay output terminal.

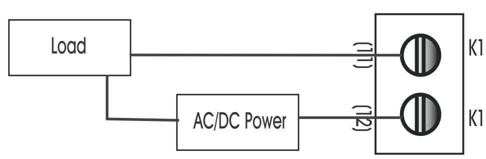


*Figure 2-5 Wiring Relay Outputs*

## 2.3.4 ANALOG OUTPUT WIRING

The analog output is a 0-10VDC output (8-bit resolution) which is typically used for PID Loop applications. Wiring the analog output to a device is straight-forward, by wiring the AO terminal to the positive side of the end device, followed by the COM terminal to the negative side of the end device.


*Figure 2-6 Analog Output Wired for a 0-10VDC Output*

## 2.4 SBC-STAT SUPPORT

The NB-ASC(e) supports the following SBC-STAT models:

- SBC-STAT1
- SBC-STAT1d
- SBC-STAT2
- SBC-STAT2d
- SBC-STAT3
- SBC-RHT (Rooftop and Heatpump applications only)

For information on wiring and installation of SBC-STAT products, please refer to the STAT User Manual (Part # 1E-04-00-0103).

**NOTE**

The designation of 'd' in the SBC-STAT model indicates that the unit uses a digital sensor, and can support networking of multiple SBC-STAT devices.

SBC-STAT models with no 'd' designation cannot be networked and are intended for a one-to-one application between in and the controller.

**NOTE**

Use of the SBC-RHT with this product is valid for single thermostat applications only. The NB-ASC(e) does not support the ability to network multiple SBC-RHT sensors, or a mixture of digital SBC-STAT sensors and SBC-RHT sensors.

**NOTE**

This product supports the ability to network SBC-STAT1D, SBC-STAT2d, and SBC-STAT3. SBC-RHT sensors cannot be used in networked applications.

# SECTION 3: FUNDAMENTAL CONCEPTS

*This section provides information on general concepts and theory that must be understood prior to setup and configuration of the NB-ASC(e).*

## IN THIS SECTION

Fundamental Concepts ........ 3-3
Command Prioritization ........ 3-6
BACnet MS/TP Protocol Information ........ 3-3
MS/TP Token Passing ........ 3-3
MS/TP LAN Wiring ........ 3-4
Device Addressing ........ 3-5
Communication Rates ........ 3-5
Network Optimization ........ 3-5

# 3.1 FUNDAMENTAL CONCEPTS

This section of the user manual reviews standard fundamental concepts and provides an explanation of the prerequisite information necessary to know prior to installing American Auto-Matrix SBC-Series products.

## 3.1.1 BACNET MS/TP PROTOCOL INFORMATION

BACnet MS/TP (**M**aster **S**lave **T**oken **P**assing) is an EIA-485 network layer intended for use with lower-level devices such as Unitary Controllers. In comparison to BACnet/IP and BACnet/Ethernet, MS/TP is more cost-effective to implement due to lower cost of wiring. Given the MS/TP network is a serial-based network, devices may be configured to communicate at different baud rates specified by BACnet. Therefore it is essential to know information regarding the BACnet network you are connecting to prior to installing and implementing the NB-ASC(e).

## 3.1.2 MS/TP TOKEN PASSING

BACnet MS/TP uses token passing to allow devices to communicate on the network. Token passing is controlled by each device, which contains an internal memory list of other MS/TP peers connected to the network. The token is passed in order of the MAC Address (Unit ID) from lowest to highest. In most MS/TP networks, each device is configured to be a master. Given all devices may be a master, MS/TP may appear and react slower than traditional building automation protocols. However, configuring your network for faster baud rates will help provide better bandwidth and transport speed of network messaging.

Token passing is a communications scheme that allows connected devices connected to inter-communicate with one another. A network “token” is passed from unit to unit on the network in a round-robin fashion by order of the MAC Address (lowest to highest) to provide a transport to access the network. When a unit possesses the token, it may perform any network activity for which it is responsible. When finished, the token is then passed onto the next device. At any time, the unit that possesses the token is the only device permitted to initiate communications with another device on the network or to request information from it. A device that receives the token may or may not need to perform network functions (e.g. read values from a remote device, broadcast information, etc.). If not, it will simply pass the token along the network.

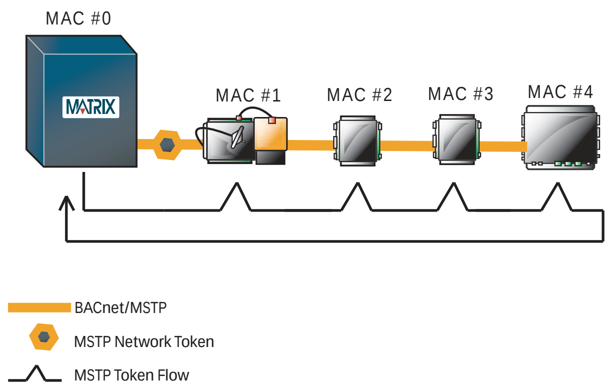


*Figure 3-1 MS/TP Token Passing Example*

Because each device can be an MS/TP master, it is important to realize that each MS/TP network should be optimized. Later sub-sections of this manual explain this process.

### 3.1.3 MS/TP LAN WIRING

Similar to EIA-485 standards, BACnet MS/TP networks support a maximum network distance of 4000 feet maximum with 18-AWG, 2-wire, shielded-twisted-pair cabling. American Auto-Matrix devices are designed with half-watt serial drivers, allowing up to a maximum of 64 devices to be connected to a single MS/TP network bus.

If you are connecting the NB-ASC(e) to an existing MS/TP network consisting of third-party devices, consult third-party vendor documentation regarding MS/TP network considerations.

### 3.1.4 DEVICE ADDRESSING

BACnet MS/TP devices contain two device addresses. One device address is known as a Device Instance, and the other is a MAC Address.

The Device Instance is an address assignment that is used to identify the BACnet device on a global BACnet network. When a device is connected to a global BACnet network consisting of multiple data layers joined together using routers, the Device Instance is used to uniquely identify the device on a global basis. The valid range for the device instance in a BACnet device is 0 to 4,194,302. The NB-ASC(e) must be configured for a unique, non-conflicting Device Instance. In the event that multiple devices are assigned the same Device Instance, both devices will simply not communicate on the BACnet network, or could be subject to mis-directed messaging (a message intended for Device-A may be routed to Device-B)

The MAC Address is an address assignment used within the BACnet MS/TP segment to permit a device to actively communicate on the BACnet MS/TP network. Valid MAC Address assignments range from 0 to 127 and are typically assigned in a logical and incremental order to permit faster token passing between devices. The MAC Address of a BACnet MS/TP device must be a unique, non-conflicting value that exists on the local MS/TP network. In the event that multiple devices are assigned with the same MAC Address, the effects can be far detrimental than that of a conflicting Device Instance; potentially resulting in a failure of the entire local MS/TP network. In the event that the NB-ASC(e) encounters its MAC Address may be a duplicate, the NB-ASC(e) will inform the user that a duplicate MAC Address has been detected and will not perform client communications until resolved.

### 3.1.5 COMMUNICATION RATES

As a serial based protocol, BACnet MS/TP supports the following four baud rates: 9.6kbps, 19.2kbps, 38.4kbps, and 76.8kbps. The NB-ASC(e) can be configured for any of the these baud rates, as well as native PC baud rates 57.6kbps and 115.2kbps which are currently not supported by the BACnet standard.

Each device communicating on an MS/TP network must be configured for the same baud rate at all times. In the event that the NB-ASC(e)'s communication baud rate is incorrect for the network it is connected to, the NB-ASC(e) will inform the user that a different baud rate has been detected and will not perform client communications until resolved.

### 3.1.6 NETWORK OPTIMIZATION

In BACnet MS/TP devices, specific device properties are available to permit optimization. network communications. By adjusting Device properties max-master and max-info-frames, users can adjust the token passing abilities of devices. The functionality of these two properties is described as follows:

- ***Max-Master*** - defines the highest unit ID of a MSTP master that is connected to the network. This value specifies to what address extent a token may pass. For example if you have 64 devices addressed in logical order, this value would be assigned to 64. This value should be set to the same value across all devices connected to an MSTP network.
- ***Max-Info-Frames*** - defines the amount of data frames that a MSTP master can use the token before passing onto the next device. This value is typically set by the factory, but can be modified if necessary. In the event a device does not need to keep the token for the amount of frames specified, AAM devices will automatically pass the token onto the next device.

## 3.2 COMMAND PRIORITIZATION

BACnet uses a command prioritization scheme for objects that control equipment or software parameters that affect the operation of equipment connected to devices. Through the use of this command prioritization scheme (commonly referred to as Priority Array), a method is provided that allows a device to determine the order in which an object is controlled. Command Prioritization assigns unique levels of priority to the different types of devices that could write values to a device. There are 16 prioritization levels with Level 1 being highest and Level 16 the lowest. A complete list of BACnet Priority Array Levels and their uses is given in Table 3-1.

*Table 3-1 : Command Prioritization Levels*

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

BACnet defines the types of objects that are either required or may optionally support the command prioritization scheme. While many factors depend on whether an object may support the feature, Table 3-2 provides a list of objects that are subject to Command Prioritization.

*Table 3-2: Objects Subject to Command Prioritization Support*

| Object Type | Support Required? | Notes |
|---|---|---|
| Analog Output | Yes | n/a |
| Analog Value | No | An Analog Value which is “writable” is not required to support Command Prioritization, rather the out-of-service property must be set to TRUE in order for the object to accept write commands. The ability to set an Analog Value into an out-of-service mode may be limited by the manufacturer of the device, as the functionality of the object may be intended to be read-only. |
| Binary Output | Yes | n/a |

*Table 3-2: Objects Subject to Command Prioritization Support*

| Object Type | Support Required? | Notes |
|---|---|---|
| Binary Value | No | A Binary Value which is "writable" is not required to support Command Prioritization, rather the out-of-service property must be set to TRUE in order for the object to accept write commands. The ability to set an Binary Value into an out-of-service mode may be limited by the manufacturer of the device, as the functionality of the object may be intended to be read-only. |
| Multi-State Output | Yes | n/a |
| Multi-State Value | No | A Multi-State Value which is "writable" is not required to support Command Prioritization, rather the out-of-service property must be set to TRUE in order for the object to accept write commands. The ability to set an Multi-State into an out-of-service mode may be limited by the manufacturer of the device, as the functionality of the object may be intended to be read-only. |

# SECTION 4: ROOFTOP CONFIGURATION

*The following section provides configuration details regarding the NB-ASC(e) Rooftop application. Please review the following sections carefully prior to configuring the controller.*

## IN THIS SECTION


Rooftop Sequence Overview ........ 4-3
Selecting Your Application Mode ........ 4-4
Universal Input Configuration ........ 4-5
  Sensor Configuration ........ 4-5
    Digital Inputs ........ 4-5
    Linear Sensors (0-10VDC) ........ 4-5
    4-20mA Sensors ........ 4-6
    Thermistor Input ........ 4-6
    Analog SBC-STAT1 ........ 4-7
Supply & Outside Temperature Configuration ........ 4-8
  Supply/Discharge Temperature ........ 4-8
  Outside Air Temperature ........ 4-8
  Outside Air Temperature Broadcasts ........ 4-8
Output Configuration ........ 4-9
Fan Control ........ 4-10
  Fan Properties ........ 4-10
  Fan Status ........ 4-11
Cool 1 and 2 ........ 4-12
  Lockout Configuration ........ 4-12
  Cool 1 Properties ........ 4-12
  Cool 2 Properties ........ 4-13
Heat 1 and 2 ........ 4-15
  Lockout Configuration ........ 4-15
  Heat 1 Properties ........ 4-15
  Heat 2 Properties ........ 4-16
Economizer ........ 4-18
PID Control ........ 4-19
Scheduling ........ 4-21
  Weekly Scheduling ........ 4-21
  Broadcast Schedule ........ 4-21
  Power-up State ........ 4-21
  Host Override ........ 4-22
SBC-STAT Configuration ........ 4-23
  Setpoint Adjustment Configuration ........ 4-23
  User Override ........ 4-23
Occupancy Detection ........ 4-24
Pulse Counting ........ 4-25

## 4.1 ROOFTOP SEQUENCE OVERVIEW

The Rooftop application of the NB-ASC(e) provides standard rooftop control utilizing zone temperature control with integrated (optional) outside air temperature and supply temperature (discharge temp) mediation. The Rooftop application contains universal inputs and static output assignments for fan, heating stages and cooling stages to provide simple configuration.

- Five (5) Universal Inputs - configurable for:
  - Outside Air Temp & Discharge Air Temp Sensors
  - Occupancy Detection Sensor
  - Proof of Flow Sensing
  - ‘General Purpose Input Sensing
- One (1) dedicated fan output
  - Assignable by Universal Input or Opto-Isolated Digital Input selection
  - Schedule control options to allow auto control (off in deadband) or run continuously
  - Failure interlocking based on a specific input
- Two (2) dedicated cooling (Outputs 2 and 3) and heating (Outputs 4 and 5) stages; includes the following features:
  - Temperature offset control of stages - based on zone temperature
  - Minimum runtime and off-time control
  - Optional Discharge Temperature and Outside Air Temperature lockout of stages
  - Stage deadband for stage disengaging -based on zone temperature
- Outside Air Temperature-based economizer control
  - Based on the input selected for Outside Air Temperature
  - Can utilize one of the four available PID Control Loops
  - Can utilize an unused 2nd stage of heating or cooling
  - Provides minimum position control, staging delay, lockouts, and reset setpoint and limiting
- Up to Four (4) dedicated PID Control sequences for proportional control
  - Each PID Control Loop is internally connected to a respective Analog Output (PID1 -> AO1, etc.)
  - PID Control Loops have several input variables including all inputs, and user adjust setpoints
  - PID Control Loops have several reset variables including all inputs, and user adjust setpoints
  - Interlock failure positioning

The NB-ASC(e) also provides the following features:

- Configurable Heating and Cooling Setpoints
  - Includes defined setbacks for unoccupied and night setback schedule modes
- Weekly Schedule
  - Performs weekly-scheduling using a BACnet Schedule object.
  - Two-mode (Occupied/Unoccupied) or Four-Mode scheduling options
  - Host Override of Schedule control - allowing global controllers to force the unit into a mode
- Occupancy Detection
  - Assignable by Universal Input selection
  - Definable delays and duration timing for occupancy controlled units.
- Pulse Counting
  - Utilizes the opto-isolated digital input
  - Can be configured to count rising edges, falling edges, or both
  - Configurable factor and scale values.

## 4.2 SELECTING YOUR APPLICATION MODE

Prior to configuring equipment such as heating and cooling stages, you should first define the application mode that the Rooftop will conform to. There are three different application modes that the Rooftop can perform. They are as followed:

- Cooling Only - utilizes cooling stages and optional economizer control. Optionally uses PID Control for other miscellaneous control applications carried out.
- Heating Only - utilizes heating stages and optional economizer control. Optionally uses PID Control for other miscellaneous control applications carried out.
- Supply Dependent (VST) - performs both heating and cooling with optional economizer control. Optically uses PID Control for other miscellaneous control applications carried out.

To configure your application mode, perform the following steps:

1. Using NB-Pro, access *Zone Temperature*
2. Locate **(BT) Control Mode**
3. For Cooling Only, set **(BT) Control Mode** = 1 (Cooling Only)
4. For Heating Only, set **(BT) Control Mode** = 2 (Heating Only)
5. For Heating & Cooling, set **(BT) Control Mode** = 3 (Supply Dependent)

## 4.3 UNIVERSAL INPUT CONFIGURATION

Universal Inputs permit the configuration of multiple sensor types, dependent on your application. The NB-ASC(e) also supports alarm capabilities within its Universal Inputs. The following section provides a quick-start reference for initial configuration of inputs, as well as alarming. Complete information regarding each property available within Universal Inputs can be located in Section 5.

### 4.3.1 SENSOR CONFIGURATION

The following section discusses how to configure a Universal Input for a specific sensor type. Universal Input configuration includes modifying control logic programming inside the NB-ASC(e) using NB-Pro and IVR hardware jumper configuration. The IVR jumpers are located above the Universal Input terminal blocks on the top right-hand side of the unitary controller, as illustrated in Figure 4-1.

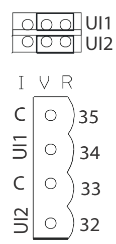


*Figure 4-1 Universal Input IVR Jumper Location*

#### 4.3.1.1 DIGITAL INPUTS

To setup an input as a digital sensor, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 0 (Digital). Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than (0) No Fault Detected, verify input wiring.

*Table 4-1: Summary of Digital Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 0 | Digital sensor configuration |
| | reliability | 0 (No Fault Detected) | Input set to automatic mode |

#### 4.3.1.2 LINEAR SENSORS (0-10VDC)

To setup an input for a 0-10VDC sensor, completely remove the IVR jumper to configure the UI for voltage mode.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 2 (Linear). Configure **min-pres-value** and **max-pres-value** to the minimum and maximum scaled values for **present-value** is

the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 4-2: Summary of an Example Linear Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 2 (Linear) | Linear sensor configuration |
| | min-pres-value | 0 | Lowest present-value scale reading for the target sensor |
| | max-pres-value | 100 | Highest present-value scale reading for the target sensor |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 4.3.1.3 4-20mA SENSORS

To setup an input as a 4-20mA sensor, you must first configure the IVR jumper to IV, which places the input into a voltage setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 3 (4-20mA). Configure **min-pres-value** and **max-pres-value** to the minimum and maximum scaled values for **present-value** is the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 4-3: Summary of an Example 4-20mA Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 3 | 4-20mA sensor configuration |
| | min-pres-value | 0 | Lowest present-value scale reading for the target sensor |
| | max-pres-value | 100 | Highest present-value scale reading for the target sensor |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 4.3.1.4 THERMISTOR INPUT

To setup an input for thermistor readings, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 7 (Thermistor). Configure the minimum and maximum scaled values for **present-value** is the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 4-4: Summary of Thermistor Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) sensor type | 7 (Thermistor) | Thermistor sensor configuration |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 4.3.1.5 ANALOG SBC-STAT1

To setup an input for temperature readings from an Analog SBC-STAT1, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 8 (Analog STAT1). Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than (0) Reliable, verify input wiring.

*Table 4-5: Summary of Analog STAT1 Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) sensor type | 8 (STAT1) | Analog STAT1 sensor configuration |
| | out-of-service | 0 (False) | Input set to automatic mode |

## 4.4 SUPPLY & OUTSIDE TEMPERATURE CONFIGURATION

The following section reviews configuration of the Supply and Outside Air Temperature sensors within the Rooftop application.

### 4.4.1 SUPPLY/DISCHARGE TEMPERATURE

The Rooftop application utilizes supply temperature for lockout application, as well as integrated supply control. Configuring this area is mandatory to make certain that stages of heating and cooling are or are not locked out. If your application does not require this temperature sensor, set **out-of-service** = 1 (Yes), located in *Supply Temperature*.

If your application does involve integration of supply sensing, perform the following steps:

1. In NB-Pro, access *Supply Temperature*
2. Verify **out-of-service** = 0 (False)
3. Set **(IC) Input Object Select** to the Universal Input your Supply Temperature sensor is connected to.
4. Set **(DD) Auto Mode Deadband** to a realistic value. The Auto Mode Deadband specifies the temperature difference by which the supply air must either exceed the heating setpoint to engage heating mode or fall below the cooling setpoint to engage cooling mode. A value of 0.0 will disable supply deadband control over heating and cooling modes. This property is commonly used when your application has been configured for Supply Dependent mode.

### 4.4.2 OUTSIDE AIR TEMPERATURE

The Rooftop application utilizes outside air temperature for locking out heating and cooling stages, as well as for operating the Economizer routine. Configuring this area is mandatory to make certain that stages of heating and cooling are or are not locked out. If your application does not require this temperature sensor, set **out-of-service** = 1 (True) located in *Outside Air Temperature*

If your application does involve integration of outside air temperature sensing, perform the following steps:

1. In NB-Pro, access *Outside Air Temperature*
2. Verify **out-of-service** = 0 (No)
3. Set **(IC) Input Object Select** to the Universal Input your Outside Air Temperature sensor is connected to.

### 4.4.3 OUTSIDE AIR TEMPERATURE BROADCASTS

The Rooftop application can receive an Outside Air Temperature broadcast from another unit that is configured to perform broadcasting (such as an NB-GPC). Additionally, the unit can also send a broadcast if desired. To receive or send Outside Air Temperature, perform the following steps:

1. In NB-Pro, access *Outside Air Temp*. *Broadcast*
2. To receive a broadcast Outside Air Temperature, set **(RB) Receive Broadcast** = 1 (Yes).
3. To send Outside Air Temperature broadcasts, set **(BE) Broadcast Enable** = 1 (Yes).

## 4.5 OUTPUT CONFIGURATION

In order for outputs to be manipulated by any of the control strategies in the controller, all of the outputs (both Analog and Binary) must be configures for automatic control.

To configure Analog Outputs, perform the following steps:

1. Using NB-Pro, access *AO0x(where x = Analog Output termination).*
2. For each Analog Output, verify **out-of-service** = 0 (False).
3. Minimum and Maximum Scaled Voltages and Engineering Units can be assigned here as well.

When Analog Outputs are configured for manual mode, they cannot be manipulated by any of the internal control processes. However, the Analog Output can be manually written to.

To configure Binary Outputs, perform the following steps:

1. Using NB-Pro, access *B00x (where x = Binary Output termination).*
2. For each Binary Output, verify **out-of-service** = 0 (False).
3. Output polarity may also be configured in this section.

## 4.6 FAN CONTROL

The Rooftop application accommodates a fan which is connected to Binary Output 1 and is controlled by the settings in *BO01*. Setup involves configuring the following items:

. Fan Properties
. Fan Status

### 4.6.1 FAN PROPERTIES

To configure the Fan Properties, perform the following steps:

1. In NB-Pro, access *BO01*
2. Configure all properties accordingly.

*Table 4-6: Fan Property Details*

| Property | Value and Interpretation |
|---|---|
| (FR) Minimum Run Time | Minimum amount of time, in minutes, the fan will stay on - prevents short-cycling |
| (FS) Minimum Off Time | Minimum amount of time, in minutes, the fan will stay off - prevents short-cycling. |
| (FX) Staging Delay | Defaults to 0.1. Can be assigned to any value greater than 0.1. A value of 0.0 will disable the fan, and is not recommended. |
| (FD) Shutoff Delay | The amount of time in seconds that the fan will remain on after all stages of cooling have been de-energized. |
| (FO) Occupied Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>1 (On) - Fan is always on. |
| (FU) Unoccupied Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>1 (On) - Fan is always on. |
| (FN) Night Setback Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>. 1 (On) - Fan is always on. |
| (DB) Fan Auto Mode Deadband | This property is used exclusively in conjunction with fan modes (FO, FU, FN) that are configured for Auto modes. When the device is in a schedule state where the fan operates in auto mode, the current zone temperature must exceed setpoint plus or minus the deadband temperature value in order for the fan to activate, followed by stages of heating or cooling. This property is useful for preventing the fan from turning on in situations where the zone temperature could possibly drift closely near mode changes. |

*Table 4-6: Fan Property Details*

| Property | Value and Interpretation |
|---|---|
| (MT) Min Run/Off Countdown Timer | Reflects the amount of time remaining (in seconds) if the fan output is currently respecting minimum-on or minimum-off time.<br><br>Examples:<br><br>If (FR) = 1.0 and the Fan Turns on, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 60 seconds would be reflected by this property.<br><br>If (FS) = 2.0 and the fan turns off, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 120 seconds would be reflected by this property. |

### 4.6.2 FAN STATUS

Fan Status is used to track the actual status of the fan output through use of a proof of flow sensor connected to a universal input or even the opto-isolated digital input. If you use a proof of flow sensor and the flow status does not match the output status, stages of heating and cooling will be locked out.

To configure Fan Status, perform the following steps:

1. Using NB-Pro, access *Proof of Flow*
2. Select the input where your proof of flow sensor it attached using **(IC) Status Input**.
3. Enter a delay value in **(PD) Delay**. This will impose a delay before considering a positive flow indication.

## 4.7 COOL 1 AND 2

The two cooling stages, controlled by BO02 and BO03, are cycled on and off to maintain the zone within a programmable deadband around a programmable setpoint. Setup involves configuring the following items:

- Lockout Configuration
- Cool 1 Properties
- Cool 2 Properties

### 4.7.1 LOCKOUT CONFIGURATION

You can configure the cooling stages to be locked out based on a temperature lockout setpoint for both outside air temperature and/or discharge air temperature. If you do not plan to have lockouts, you must still configure the properties accordingly. To configure, perform the following steps:

1. In NB-Pro, access *BO02*
2. **(TL) DAT Low Temp Lockout** defines the lowest sensed temperature allowed before cooling is interrupted or not permitted to stage on. Enter your desired lockout value. If you do not plan to use a lockout baed on Discharge Temperature, set **(TL) DAT Low Temp Lockout** = 999.0
3. **(CL) OAT Cooling Lockout** defines the lowest sensed temperature allowed before cooling is interrupted or not permitted to stage on. Enter your desired lockout value. If you do not plan to use a lockout based on Outside Air Temperature, set **(CL) OAT Cooling Lockout** = 999.0

### 4.7.2 COOL 1 PROPERTIES

To configure Cool 1 properties, perform the following steps:

1. In NB-Pro, access *BO02*
2. Configure your properties accordingly.

*Table 4-7: Cool 1 Property Details*

| Property | Value and Interpretation |
|---|---|
| (TO) Temp Offset | Indicates the temperature offset from your cooling setpoint before Cool 1 is engaged. Note that the stage may engage if the Staging Delay expires prior to the temperature offset. |
| (MX) Staging Delay | Indicates the amount of time, in minutes, before Cool 2 is engaged. Note that the stage may engage if the Temp Offset is exceeded prior to the Staging Delay. |
| (MR) Minimum Run Time | Minimum amount of time, in minutes, the stage will stay on - prevents short-cycling |
| (MS) Minimum Off Time | Minimum amount of time, in minutes, the stage will stay off - prevents short-cycling. |
| (DB) Deadband | The stage will not dis-engage until the Zone Temperature passes setpoint by the number of degrees specified here. |

*Table 4-7: Cool 1 Property Details*

| Property | Value and Interpretation |
|---|---|
| (MT) Min Run/Off Countdown Timer | Reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time.<br><br>Examples:<br><br>If (FR) = 1.0 and the output turns on, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 60 seconds would be reflected by this property.<br><br>If (FS) = 2.0 and the output turns off, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 120 seconds would be reflected by this property. |

### 4.7.3 COOL 2 PROPERTIES

To configure Cool 2 properties, perform the following steps:

1. In NB-Pro, access *BO03*
2. Configure your properties accordingly.

*Table 4-8: Cool 2 Property Details*

| Property | Value and Interpretation |
|---|---|
| (TO) Temp Offset | Indicates the temperature offset from your cooling setpoint before Cool 2 is engaged. Note that the stage may engage if the Staging Delay expires prior to the temperature offset. |
| (MR) Minimum Run Time | Minimum amount of time, in minutes, the stage will stay on - prevents short-cycling |
| (MS) Minimum Off Time | Minimum amount of time, in minutes, the stage will stay off - prevents short-cycling. |
| (DB) Deadband | The stage will not dis-engage until the Zone Temperature passes setpoint by the number of degrees specified here. |

*Table 4-8: Cool 2 Property Details*

| Property | Value and Interpretation |
|---|---|
| (MT) Min Run/Off Countdown Timer | Reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time.<br><br>Examples:<br><br>If (FR) = 1.0 and the output turns on, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 60 seconds would be reflected by this property.<br><br>If (FS) = 2.0 and the output turns off, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 120 seconds would be reflected by this property. |

## 4.8 HEAT 1 AND 2

The two heating stages, controlled by BO04 and BO05, are cycled on and off to maintain the zone within a programmable deadband around a programmable setpoint. Setup involves configuring the following items:

- Lockout Configuration
- Heat 1 Properties
- Heat 2 Properties

### 4.8.1 LOCKOUT CONFIGURATION

You can configure the heating stages to be locked out based on a temperature lockout setpoint for both outside air temperature and/or discharge air temperature. If you do not plan to have lockouts, you must still configure the properties accordingly. To configure, perform the following steps:

1. In NB-Pro, access *BO04*
2. **(TH) DAT High Temp Lockout** defines the highest sensed temperature allowed before heating is interrupted or not permitted to stage on. Enter your desired lockout value. If you do not plan to use a lockout baed on Discharge Temperature, set **(TH) DAT Low Temp Lockout** = 999.0
3. **(HL) OAT Heating Lockout** defines the highest sensed temperature allowed before heating is interrupted or not permitted to stage on. Enter your desired lockout value. If you do not plan to use a lockout based on Outside Air Temperature, set **(CH) OAT Cooling Lockout** = 999.0

### 4.8.2 HEAT 1 PROPERTIES

To configure Heat 1 properties, perform the following steps:

1. In NB-Pro, access *BO04*
2. Configure your properties accordingly.

*Table 4-9: Heat 1 Property Details*

| Property | Value and Interpretation |
|---|---|
| (TO) Temp Offset | Indicates the temperature offset from your cooling setpoint before Heat 1 is engaged. Note that the stage may engage if the Staging Delay expires prior to the temperature offset. |
| (MX) Staging Delay | Indicates the amount of time, in minutes, before Heat 2 is engaged. Note that the stage may engage if the Temp Offset is exceeded prior to the Staging Delay. |
| (MR) Minimum Run Time | Minimum amount of time, in minutes, the stage will stay on - prevents short-cycling |
| (MS) Minimum Off Time | Minimum amount of time, in minutes, the stage will stay off - prevents short-cycling. |
| (DB) Deadband | The stage will not dis-engage until the Zone Temperature passes setpoint by the number of degrees specified here. |

*Table 4-9: Heat 1 Property Details*

| Property | Value and Interpretation |
|---|---|
| (MT) Minimum Run/Off Timer | Reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time.<br><br>Examples:<br><br>If (FR) = 1.0 and the output turns on, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 60 seconds would be reflected by this property.<br><br>If (FS) = 2.0 and the output turns off, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 120 seconds would be reflected by this property. |

### 4.8.3 HEAT 2 PROPERTIES

To configure Heat 2 properties, perform the following steps:

1. In NB-Pro, access *BO05*
2. Configure your properties accordingly.

*Table 4-10: Heat 1 Property Details*

| Property | Value and Interpretation |
|---|---|
| (TO) Temp Offset | Indicates the temperature offset from your cooling setpoint before Heat 1 is engaged. Note that the stage may engage if the Staging Delay expires prior to the temperature offset. |
| (MX) Staging Delay | Indicates the amount of time, in minutes, before Heat 2 is engaged. Note that the stage may engage if the Temp Offset is exceeded prior to the Staging Delay. |
| (MR) Minimum Run Time | Minimum amount of time, in minutes, the stage will stay on - prevents short-cycling. |
| (MS) Minimum Off Time | Minimum amount of time, in minutes, the stage will stay off - prevents short-cycling. |
| (DB) Deadband | The stage will not dis-engage until the Zone Temperature passes setpoint by the number of degrees specified here. |

*Table 4-10: Heat 1 Property Details*

| Property | Value and Interpretation |
|---|---|
| (MT) Minimum Run/Off Timer | Reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time.<br><br>Examples:<br><br>If (FR) = 1.0 and the output turns on, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 60 seconds would be reflected by this property.<br><br>If (FS) = 2.0 and the output turns off, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 120 seconds would be reflected by this property. |

## 4.9 ECONOMIZER

The Rooftop application includes an Economizer. Based on outside air temperature, the economizer can be configured to open a damper during calls for cooling to provide free cooling, allowing the space to cool down using cooler outside air as an alternative to running all stages of cooling. The economizer can utilize an available second stage of heating or cooling, or use one of the four PID control loops to control a proportional motor.

To use the economizer, your Outside Air Temperature sensor must be configured (see previous steps). To configure the economizer, perform the following steps:

1. Using NB-Pro, access *Economizer*
2. Configure your properties accordingly.

*Table 4-11: Economizer Property Details*

| Property | Value and Interpretation |
|---|---|
| (EE) Economizer Enable | Specifies the PID Loop or Binary Output to be used for Economizer control.<br>0 = Disabled<br>1 = PID 1<br>2 = PID 2<br>3 = PID 3<br>4 = PID 4<br>5 = BO3<br>6 = BO5<br><br>**Note** - Using BO3 or BO5 will cancel out the second stage of heating or cooling respectively. |
| (OH) OAT High Limit | If the OAT rises above this setpoint, the economizer output will be set to the defined minimum position. |
| (OL) OAT Low Limit | If the OAT falls below this setpoint, the economizer output will be set to the defined minimum position. |
| (EM) Minimum Position | Specifies this minimum position for the economizer output when OAT rises or falls below the high and low limits |
| (ED) Economizer Staging Delay | Specifies how many minutes the controller will wait before using additional cooling stages after the economizer damper reaches 100% |
| (CM) Calculated Minimum Position | Displays the actual minimum position as calculated by the Economizer's internal control loop. |
| (MV) Reset Variable | Specifies an input for reset applications. |
| (MP) Reset Setpoint | Specifies the setpoint at which reset action occurs. |
| (MR) Maximum Reset | Specifies the maximum amount to reset the minimum position by when the variable reaches setpoint. |
| (ML) Reset Limit | Applied to determine the minimum position when (ML) is equal to the reset variable's input value. |

## 4.10 PID CONTROL

The NB-ASC(e) provides up to four (4) PID Control Loops for proportional control strategies. While some of these loops may be utilized by other strategies (such as Economizer or Valve Control - application dependent), the PID Control Loops can be used to control proportional equipment in an efficient manner.

Each PID Control is statically linked to a specific Analog Output. For example, PID Control 1 is linked to Analog Output 1, PID Control 2 is linked to Analog Output 2, etc. PID Loops use command prioritization to control the linked Analog Output using Priority 11.

The following section discusses the operation and configuration of PID Control in the NB-ASC(e).

*Table 4-12: PID Control Property Details*

| Property | Value and Interpretation |
|---|---|
| (SP) Loop Setpoint | Specifies the setpoint for loop control. The setpoint corresponds to the input variable specified in **(IC) Input Object** |
| (CS) Control Setpoint | Indicates the current setpoint used as part of the control loop, displays the calculated setpoint with any setup/ setback, or setpoint adjustment from a connected SBC-STAT that may be applied. |
| (PO) Percent Output | Indicates the scaled output as configured in the actual Analog Output (FD0x). |
| (IN) Input Object Value | Indicates the current value of the input variable specified in **(IC) Input Object**. |
| (IC) Input Object | Specifies the input reference that the loop will proportionally control by. Valid options include:<br><br>0 = Disable<br>1 = Zone Temperature<br>2 = Supply Temperature<br>4 = UI1<br>5 = UI2<br>6 = UI3<br>7 = UI4<br>8 = UI5<br>9 = Zone Heating (follows adjusted heating setpoint)<br>10 = Zone Cooling (follows adjusted cooling setpoint)<br>11 = Outside Air Temperature |
| (MR) Maximum Reset | Specifies the maximum amount needed to reset to e loop setpoint based on when reset is being used. |
| (RC) Reset Variable Value | Indicates the current value of the reset variable specified in **(RV) Reset Variable**. |

*Table 4-12: PID Control Property Details*

| Property | Value and Interpretation |
|---|---|
| (RV) Reset Variable | Specifies the reset input reference that the loop will use to perform reset control. Valid options for reset include:<br><br>0 = Disable<br>1 = Zone Temperature<br>2 = Supply Temperature<br>4 = UI1<br>5 = UI2<br>6 = UI3<br>7 = UI4<br>8 = UI5<br>11 = Outside Air Temperature |
| (RS) Reset Setpoint | Specifies the setpoint at which reset action begins. When the reset variable's value exceeds this setpoint, reset action will be used to determine the **(CS) Control Setpoint**. |
| (RL) Reset Limit | Specifies the value at which maximum reset is applied, based on the reset variable's value. |
| (DB) Deadband | Specifies the deadband for proportional control. The deadband straddles the setpoint. For example, a value of 2.0 would be applied to both the left and right side of the action. |
| (PB) Proportional Band | Specifies the variable range over which the output is changed based on input variable changes. The proportional band is centered around the setpoint for the loop. |
| (RP) Reset Period | Specifies the amount of time, in seconds, over which error history is accumulated for reset control. |
| (RT) Rate | Specifies the derivative rate for the loop |
| (CS) Control Sign | Specifies control action of the loop 0 = Normal/Direct, 1 = Reverse. |
| (SU) Setup/Setback | Specifies how much the setpoint should be lowered during unoccupied periods. |
| (ID) Interlock Enable/Disable | Enabled/Disables interlock failure modes for the control loop. |
| (P1) Interlock 1 Position | Fail position for the control loop when Interlock 1 occurs |
| (P2) Interlock 2 Position | Fail position for the control loop when Interlock 2 occurs |
| (P3) No Flow Position | Fail position if no flow or fan operation occurs. |
| (CE) Control Enable | Activation point for the control loop. To enable the loop, set to a value of 1(Yes). |

# 4.11 SCHEDULING

Scheduling controls the current temperature setpoint of the NB-ASC(e). There are multiple ways scheduling can be performed in the controller. Internal schedules can be defined by the user. The user can determine when and in which schedule mode (or state) the *NB*-ASC will operate; including occupied, warm-up, unoccupied, or night setback. The NB-ASC(e) accommodates four-mode schedules or two-mode schedules (utilizing Occupied and Unoccupied modes only). All schedule configuration is performed through the *Schedule* object.

## 4.11.1 WEEKLY SCHEDULING

The NB-ASC(e) contains a BACnet Schedule object, which is capable of performing 4-mode scheduling (Occupied, Unoccupied, Warm up, and Night Setback) for all days of the week.

To configure a Schedule Object, perform the following steps:

1. Using NB-Pro, access *Schedules.*
2. Determine if you wish to use two-mode scheduling or four-mode scheduling. To use two-mode scheduling, you must write an ENUM data type value to schedule-default. A value of zero (0) indicates Unoccupied, where as a value of one (1) indicates Occupied mode. Otherwise, write an unsigned value (0 = Unoccupied, 1 = Warm up, 2 = Occupied, 3 = Night Setback).
3. Configure time,value entries for each day of the week in **weekly-schedule**. A maximum of 6 time,value entries can be associated to one day of the week. If no values are entered for a specific day, the schedule will operate using the las time,value entry made in the previous day. No entries in any day of the week will result in the Schedule operating off the **schedule-default** value.

## 4.11.2 BROADCAST SCHEDULE

The Broadcast Schedule is a schedule sent out over the network by another controller such as an NB-GPC. The active internal schedule will be overridden if the NB-ASC(e) is configured to receive network broadcast schedules. If the **(RB) Receive Broadcast** property is enabled, the current schedule will reflect the **(CV) Current Value** property. To configure the NB-ASC(e) to receive network broadcast schedules:

1. Using NB-Pro, access *Broadcast Schedule*
2. Set **(RB) Receive Broadcast** = 1 (Yes)

## 4.11.3 POWER-UP STATE

If an unscheduled power loss occurs and power is restored, or if a soft reset of the controller is performed (Device;**(RS) Reset Controller** = 1), the controller will operate in the schedule mode defined by the user in the **Power-up State** (Device; **(PS) Power Up State**) property until a time synchronization received by the device from a time master. To set the schedule mode in which you want the device to operate upon power restoration or after a soft reset has occurred, select the value that corresponds to the desired power-up state. The possible states are listed in Table 4-13.

*Table 4-13 : Power-up States*

| Value | Power-Up State |
|---|---|
| 0 | Unoccupied |
| 1 | Warm-up |
| 2 | Occupied |
| 3 | Night Setback |

### 4.11.4 HOST OVERRIDE

In multi-device or zone situations, it may be advantageous to have a host or other peer device directly control the schedule state of the controller without broadcasts. In this case, the controller has a Host Override function in the Schedule object that can be utilized.

To configure the device to have its schedule controlled by an external source, set Schedule; **(HE) Host Override Local Schedules** = 1 (Yes). Once set, the schedule of the device is then controlled through writes to the Schedule;**(HO) Host Schedule Setting** property.

The schedule mode set in **HO** will be the active mode unless:

- a broadcast is received
- an occupancy sensor is properly configured and occupancy is detected
- user override occurs

When host override is used, the true state of the schedule can be monitored through Zone Temperature; **(PO) Present Occupancy status**. In previous iterations of NB-ASC(e) firmware, the present-value of the schedule could be monitored, but cannot in v6.03 or later due to BTL requirements.

# 4.12 SBC-STAT CONFIGURATION

The NB-ASC(e) supports SBC-STAT model devices, as referenced in Section 2. There are a few configuration options available for a connected SBC-STAT, which this section reviews

## 4.12.1 SETPOINT ADJUSTMENT CONFIGURATION

Setpoint adjustment configuration can be achieved by accessing *Zone Temperature* using NB-Pro. Options found in this section include the following in the table below:

*Table 4-14: Setpoint Adjustment Properties*

| Property | Value and Interpretation |
|---|---|
| (ZS) Zone Midpoint | Specifies the comfort level for the zone. |
| (TS) User Setpoint Offset | Specifies an offset to apply to Zone Heating and Zone Cooling for PID Control. |
| (TM) User Adjust Increment | Specifies the magnitude of increment/decrement changes made to the setpoint. |
| (TT) User Adjust Duration | Specifies how much time, in minutes, a setpoint change is applied to the controller. |
| (SD) Calculated Setpoint Display | Specifies whether the offset, zone midpoint, heating setpoint, or cooling setpoint is displayed when a user performs setpoint adjustment. |

## 4.12.2 USER OVERRIDE

If the active schedule controlling the NB-ASC(e) is in unoccupied or night setback mode, user override is possible. If the user presses the up or down arrow push-button on the SBC-STAT2, SBC-STAT2D, or SBC-STAT3 and the **(SE) Override Enable/Disable** property is Enabled (value of 1), the unit will go into occupied mode.

The duration of this mode, which is also called extended occupancy, can be set by using the **(ED) Extended Occupancy Duration** property.

To configure the NB-ASC(e) for user override ability via an *SBC*-STAT, perform the following steps:

1. Using NB-Pro, access *Zone Temperature*
2. Set **(SE) User Override** = 1 (Enabled)
3. Set **(ED) Extended Occupancy Duration** to however many minutes you wish for user override mode to occur.

When the unit's schedule is in unoccupied mode and the user enables override from a connected STAT, occupancy will occur for the amount of minutes specified in **(ED) Extended Occupancy Duration**. Once the time has elapsed, the unit will revert back to its configured schedule mode.

## 4.13 OCCUPANCY DETECTION

The occupancy detection feature enables the NB-ASC(e) to automatically go to occupied mode, (also called extended occupancy) when a dedicated occupancy sensor indicates the monitored zone is occupied. The length of time that the controller will operate in extended occupancy is defined by the user in the Occupancy Detection; **(MT) Extended Occupancy Duration** property. To configure the controller for occupancy detection capability, perform the following steps:

1. Using NB-Pro, access *Occupancy Detection*
2. Set **(IC) Status Input** to the input that the occupancy detector is connected to.
3. Set **(MD) Extended Occupancy Delay** to the desired number of seconds the detector must indicate that occupancy is detected before overriding the zone. This prevents false triggering of the occupancy detection in the event someone or something quickly passes through the zone.
4. Set **(MT) Extended Occupancy Duration** to the desired number of minutes the controller is to remain in occupied mode once the zone has been occupied.

**NOTE**

If **(MT) Extended Occupancy Duration** is not set to a value greater than zero, the controller will not enter extended occupancy when it is detected that the zone is occupied.

**NOTE**

The input selected for the Occupancy Detection application must be configured as a digital input.

## 4.14 PULSE COUNTING

The NB-ASC(e) provides an optically isolated digital input, which can be used for pulse counting. A Pulse application is provided, allowing users to perform count applications. To setup Pulse Counting, perform the following:

1. Using NB-Pro, access *Pulse Input*
2. Select a valid count mode from **(MD) Counter Mode**.
3. Enter your scale factor into **(SF) Pulse Scale Factor**

Your scaled value based on the number of pulses collected will appear in **(SV) Pulse Scale Value**.

# Section 5: Heatpump Configuration

*The following section provides configuration details regarding the NB-ASC(e) Heatpump application. Please review the following sections carefully prior to configuring the controller.*

## In this section


Heatpump Sequence Overview ..... 5-3
Selecting Your Application Mode ..... 5-4
Configuring the Reversing Delay ..... 5-4
Universal Input Configuration..... 5-5
Sensor Configuration ..... 5-5
Digital Inputs ..... 5-5
Linear Sensors (0-10VDC)..... 5-5
4-20mA Sensors ..... 5-6
Thermistor Input..... 5-6
Analog SBC-STAT1..... 5-7
Supply & Outside Temperature Configuration..... 5-8
Supply/Discharge Temperature..... 5-8
Outside Air Temperature ..... 5-8
Outside Air Temperature Broadcasts ..... 5-8
Output Configuration..... 5-9
Fan Control ..... 5-10
Fan Properties ..... 5-10
Fan Status.....5-11
Reversing Valve and Defrost Cycle..... 5-12
Reversing Valve Properties..... 5-12
Defrost Cycle Properties ..... 5-12
Stage 1 and 2..... 5-14
Cooling Lockout Configuration..... 5-14
Heating Lockout Configuration..... 5-14
Stage 1 Properties..... 5-14
Stage 2 Properties..... 5-15
Auxiliary Heat..... 5-17
Auxiliary Heat Properties..... 5-17
Economizer ..... 5-18
PID Control ..... 5-19
Scheduling ..... 5-21
Weekly Scheduling..... 5-21
Broadcast Schedule..... 5-21
Power-up State..... 5-21
Host Override..... 5-22
SBC-STAT Configuration ..... 5-23
Setpoint Adjustment Configuration ..... 5-23
User Override..... 5-23
Occupancy Detection..... 5-24
Pulse Counting..... 5-25

## 5.1 HEATPUMP SEQUENCE OVERVIEW

The Heatpump application of the NB-ASC(e) provides standard heat pump control utilizing zone temperature control with integrated (optional) outside air temperature and supply temperature (discharge temp) mediation. The Heatpump application contains universal inputs and static output assignments for fan, heating stages and cooling stages to provide simple configuration.

- Five (5) Universal Inputs - configurable for:
  - Outside Air Temp & Discharge Air Temp Sensors
  - Occupancy Detection Sensor
  - Proof of Flow Sensing
  - ‘General Purpose Input Sensing
- Fan Output - Binary Output 1
  - Assignable by Universal Input or Opto-Isolated Digital Input selection
  - Schedule control options to allow auto control (off in deadband) or run continuously
  - Failure interlocking based on a specific input
- Reversing Valve - Binary Output 2
  - Configurable Settling Delay
  - Configurable Defrost Cycle Application
- Up to two (2) stages of heating and cooling.
  - Temperature offset control of stages - based on zone temperature
  - Minimum runtime and off-time control
  - Optional Discharge Temperature and Outside Air Temperature lockout of stages
  - Stage deadband for stage dis-engagement, based on zone temperature
- Auxiliary Heat Application
  - Configurable for Single Stage or Dual Stage Auxiliary Heat
  - Optional Stage Balancing with Outside Air and Supply Air lockouts
- Up to Four (4) dedicated PID Control sequences for proportional control
  - Each PID Control Loop is internally connected to a respective Analog Output (PID1 -> AO1, etc.)
  - PID Control Loops have several input variables including all inputs, and user adjust setpoints
  - PID Control Loops have several reset variables including all inputs, and user adjust setpoints
  - Interlock failure positioning
- Outside Air Temperature-based economizer control
  - Based on the input selected for Outside Air Temperature
  - Can utilize one of the four available PID Control Loops
  - Can utilize an unused 2nd stage of heating or cooling
  - Provides minimum position control, staging delay, lockouts, and reset setpoint and limiting

The NB-ASC(e) also provides the following features:

- Configurable Heating and Cooling Setpoints
  - Includes defined setbacks for unoccupied and night setback schedule modes
- Weekly Schedule
  - Performs weekly-scheduling using a BACnet Schedule object.
  - Two-mode (Occupied/Unoccupied) or Four-Mode scheduling options
  - Host Override of Schedule control - allowing global controllers to force the unit into a mode
- Occupancy Detection
  - Assignable by Universal Input selection
  - Definable delays and duration timing for occupancy controlled units.
- Pulse Counting
  - Utilizes the opto-isolated digital input
  - Can be configured to count rising edges, falling edges, or both
  - Configurable factor and scale values.

## 5.2 SELECTING YOUR APPLICATION MODE

Prior to configuring equipment such as heating and cooling stages, you should first define the application mode that the Heatpump will conform to. There are three different application modes that the Heatpump can perform. They are as followed:

- Cooling Only - utilizes cooling only routines and optional economizer control. Optionally uses PID Control for other miscellaneous control applications carried out.
- Heating Only - utilizes heating only routines and optional economizer control. Optionally uses PID Control for other miscellaneous control applications carried out.
- Supply Dependent (VST) - performs both heating and cooling. Optionally uses PID Control for other miscellaneous control applications carried out.

To configure your application mode, perform the following steps:

1. Using NB-Pro, access *Zone Temperature*
2. Locate **(BT) Control Mode**
3. For Cooling Only, set **(BT) Control Mode** = 1 (Cooling Only)
4. For Heating Only, set **(BT) Control Mode** = 2 (Heating Only)
5. For Heating & Cooling, set **(BT) Control Mode** = 3 (Supply Dependent)

### 5.2.1 CONFIGURING THE REVERSING DELAY

In the Heatpump application, there is a configurable reversing delay that is imposed before a zone can call for heating after a period of cooling, or a call for cooling after a period of heating. By default, this parameter defaults to 15 minutes.

To modify this parameter, perform the following steps:

1. Using NB-Pro, access *Zone Temperature*
2. Locate **(RD) Reversing Delay**. Configure the value for a desired reversing delay.

## 5.3 Universal Input Configuration

Universal Inputs permit the configuration of multiple sensor types, dependent on your application. The NB-ASC(e) also supports alarm capabilities within its Universal Inputs. The following section provides a quick-start reference for initial configuration of inputs, as well as alarming. Complete information regarding each property available within Universal Inputs can be located in Section 5.

### 5.3.1 Sensor Configuration

The following section discusses how to configure a Universal Input for a specific sensor type. Universal Input configuration includes modifying control logic programming inside the NB-ASC(e) using NB-Pro and IVR hardware jumper configuration. The IVR jumpers are located above the Universal Input terminal blocks on the top right-hand side of the unitary controller, as illustrated in Figure 5-1.

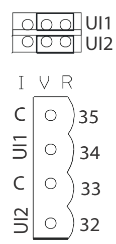


*Figure 5-1 Universal Input IVR Jumper Location*

#### 5.3.1.1 Digital Inputs

To setup an input as a digital sensor, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 0 (Digital). Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than (0) No Fault Detected, verify input wiring.

*Table 5-1: Summary of Digital Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 0 | Digital sensor configuration |
| | reliability | 0 (No Fault Detected) | Input set to automatic mode |

#### 5.3.1.2 Linear Sensors (0-10VDC)

To setup an input for a 0-10VDC sensor, completely remove the IVR jumper to configure the UI for voltage mode.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 2 (Linear). Configure **min-pres-value** and **max-pres-value** to the minimum and maximum scaled values for **present-value** is

the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 5-2: Summary of an Example Linear Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 2 (Linear) | Linear sensor configuration |
| | min-pres-value | 0 | Lowest present-value scale reading for the target sensor |
| | max-pres-value | 100 | Highest present-value scale reading for the target sensor |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 5.3.1.3 4-20mA SENSORS

To setup an input as a 4-20mA sensor, you must first configure the IVR jumper to IV, which places the input into a voltage setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 3 (4-20mA). Configure **min-pres-value** and **max-pres-value** to the minimum and maximum scaled values for **present-value** is the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 5-3: Summary of an Example 4-20mA Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 3 | 4-20mA sensor configuration |
| | min-pres-value | 0 | Lowest present-value scale reading for the target sensor |
| | max-pres-value | 100 | Highest present-value scale reading for the target sensor |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 5.3.1.4 THERMISTOR INPUT

To setup an input for thermistor readings, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 7 (Thermistor). Configure the minimum and maximum scaled values for **present-value** is the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 5-4: Summary of Thermistor Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) sensor type | 7 (Thermistor) | Thermistor sensor configuration |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 5.3.1.5 ANALOG SBC-STAT1

To setup an input for temperature readings from an Analog SBC-STAT1, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 8 (Analog STAT1). Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than (0) Reliable, verify input wiring.

*Table 5-5: Summary of Analog STAT1 Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) sensor type | 8 (STAT1) | Analog STAT1 sensor configuration |
| | out-of-service | 0 (False) | Input set to automatic mode |

## 5.4 SUPPLY & OUTSIDE TEMPERATURE CONFIGURATION

The following section reviews configuration of the Supply and Outside Air Temperature sensors within the Heatpump application.

### 5.4.1 SUPPLY/DISCHARGE TEMPERATURE

The Heatpump application utilizes supply temperature for lockout application, as well as integrated supply control. Configuring this area is mandatory to make certain that stages of heating and cooling are or are not locked out. If your application does not require this temperature sensor, set **out-of-service** = 1 (Yes), located in *Supply Temperature*.

If your application does involve integration of supply sensing, perform the following steps:

1. In NB-Pro, access *Supply Temperature*
2. Verify **out-of-service** = 0 (False)
3. Set **(IC) Input Object Select** to the Universal Input your Supply Temperature sensor is connected to.
4. Set **(DD) Auto Mode Deadband** to a realistic value. The Auto Mode Deadband specifies the temperature difference by which the supply air must either exceed the heating setpoint to engage heating mode or fall below the cooling setpoint to engage cooling mode. A value of 0.0 will disable supply deadband control over heating and cooling modes. This property is commonly used when your application has been configured for Supply Dependent mode.

### 5.4.2 OUTSIDE AIR TEMPERATURE

The Heatpump application utilizes outside air temperature for locking out heating and cooling stages, as well as for operating the Economizer routine. Configuring this area is mandatory to make certain that stages of heating and cooling are or are not locked out. If your application does not require this temperature sensor, set **out-of-service** = 1 (True) located in *Outside Air Temperature*

If your application does involve integration of outside air temperature sensing, perform the following steps:

1. In NB-Pro, access *Outside Air Temperature*
2. Verify **out-of-service** = 0 (No)
3. Set **(IC) Input Object Select** to the Universal Input your Outside Air Temperature sensor is connected to.

### 5.4.3 OUTSIDE AIR TEMPERATURE BROADCASTS

The Heatpump application can receive an Outside Air Temperature broadcast from another unit that is configured to perform broadcasting (such as an NB-GPC). Additionally, the unit can also send a broadcast if desired. To receive or send Outside Air Temperature, perform the following steps:

1. In NB-Pro, access *Outside Air Temp*. *Broadcast*
2. To receive a broadcast Outside Air Temperature, set **(RB) Receive Broadcast** = 1 (Yes).
3. To send Outside Air Temperature broadcasts, set **(BE) Broadcast Enable** = 1 (Yes).

## 5.5 OUTPUT CONFIGURATION

In order for outputs to be manipulated by any of the control strategies in the controller, all of the outputs (both Analog and Binary) must be configures for automatic control.

To configure Analog Outputs, perform the following steps:

1. Using NB-Pro, access *AO0x(where x = Analog Output termination).*
2. For each Analog Output, verify **out-of-service** = 0 (False).
3. Minimum and Maximum Scaled Voltages and Engineering Units can be assigned here as well.

When Analog Outputs are configured for manual mode, they cannot be manipulated by any of the internal control processes. However, the Analog Output can be manually written to.

To configure Binary Outputs, perform the following steps:

1. Using NB-Pro, access *B00x (where x = Binary Output termination).*
2. For each Binary Output, verify **out-of-service** = 0 (False).
3. Output polarity may also be configured in this section.

## 5.6 FAN CONTROL

The Heatpump application accommodates a fan which is connected to Binary Output 1 and is controlled by the settings in the BO01 object. Setup involves configuring the following items:

- Fan Properties
- Fan Status

### 5.6.1 FAN PROPERTIES

To configure the Fan Properties, perform the following steps:

1. In NB-Pro, access *BO01*
2. Configure all properties accordingly.

*Table 5-6: Fan Properties Details*

| Property | Value and Interpretation |
|---|---|
| (FR) Minimum Run Time | Minimum amount of time, in minutes, the fan will stay on - prevents short-cycling |
| (FS) Minimum Off Time | Minimum amount of time, in minutes, the fan will stay off - prevents short-cycling. |
| (FX) Staging Delay | Defaults to 0.1. Can be assigned to any value greater than 0.1. A value of 0.0 will disable the fan, and is not recommended. |
| (FD) Shutoff Delay | The amount of time in seconds that the fan will remain on after all stages of cooling have been de-energized. |
| (FO) Occupied Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>1 (On) - Fan is always on. |
| (FU) Unoccupied Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>1 (On) - Fan is always on. |
| (FN) Night Setback Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>1 (On) - Fan is always on. |
| (DB) Fan Auto Mode Deadband | This property is used exclusively in conjunction with fan modes (FO, FU, FN) that are configured for Auto modes. When the device is in a schedule state where the fan operates in auto mode, the current zone temperature must exceed setpoint plus or minus the deadband temperature value in order for the fan to activate, followed by stages of heating or cooling. This property is useful for preventing the fan from turning on in situations where the zone temperature could possibly drift closely near mode changes. |

*Table 5-6: Fan Properties Details*

| Property | Value and Interpretation |
|---|---|
| (MT) Min Run/Off Countdown Timer | Reflects the amount of time remaining (in seconds) if the fan output is currently respecting minimum-on or minimum-off time.<br><br>Examples:<br><br>If (FR) = 1.0 and the Fan Turns on, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 60 seconds would be reflected by this property.<br><br>If (FS) = 2.0 and the fan turns off, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 120 seconds would be reflected by this property. |

### 5.6.2 FAN STATUS

Fan Status is used to track the actual status of the fan output through use of a proof of flow sensor connected to a universal input or even the opto-isolated digital input. If you use a proof of flow sensor and the flow status does not match the output status, stages of heating and cooling will be locked out.

To configure Fan Status, perform the following steps:

1. Using NB-Pro, access *Proof of Flow*
2. Select the input where your proof of flow sensor it attached using **(IC) Status Input**.
3. Enter a delay value **(PD) Delay**. This will impose a delay before considering a positive flow indication.

## 5.7 REVERSING VALVE AND DEFROST CYCLE

The Reversing Valve output (Binary Output 2) is used to switch between heating and cooling modes based on zone demand, and also controls an optional defrost cycle for heatpump applications.

When heating or cooling is required, the reversing valve will open/close and then allow for a single or dual stage of heating or cooling to energize.

The optional defrost cycle application provides a defrost mode for Heatpump. This application assumes that a valid temperature sensor (commonly installed along the coil) has been connected to an available universal input. The application also permits Outside Air Temperature to be used if necessary. When the measured variable drops below the **Enter Defrost Setpoint** parameter, the reversing valve will switch to cooling mode, and then enable heating stages to defrost the coils. The controller will remain in this mode for the amount of time based on the **Maximum Defrost Cycle Time** parameter. If the measured variable exceeds the **Exit Defrost Setpoint** parameter or if the maximum amount of time expires, the heating stages will disable and the reversing valve will return to heating mode. The next defrost cycle will then occur based on the **Programmed Time Between Defrost Cycles** parameter.

Setup involves configuring the following items:

- Reversing Valve Properties
- Defrost Cycle Properties

### 5.7.1 REVERSING VALVE PROPERTIES

Property configuration of the Reversing Valve is quite simple and involved two properties. They can be located in NB-Pro by accessing *BO02*

- **(SD) Settling Delay** - specifies how many seconds that is imposed both before and after the reversing valve state is changed. The delay begins once all stages have been shut down. Stages will not energize for this period after the valve state is changed.
- **polarity** - permits configuration of the output polarity. This property is set based on the valve status used for cooling states.

### 5.7.2 DEFROST CYCLE PROPERTIES

Property configuration for defrost cycle applications involves the following properties located under *BO02.* Select properties are used to provide feedback status relative to the operation of the defrost cycle application.

- **(FB) Defrost Status** - read-only feedback property indicating if defrost logic is active, and if a defrost cycle is in progress. By default, the device is not configured for defrost cycle applications.
- **(IN) Defrost Input Select** - specified which input to use as the measured variable for defrost cycles. This input will commonly be set to the corresponding universal input which has a coil temperature sensor connected. However, the application permits any universal input to be used.
- **(ES) Enter Defrost Setpoint** - specified the value that the measured variable must drop below in order for defrost control to occur.
- **(XS) Exit Defrost Setpoint** - specifies the value that the measured variable must rises above in order for defrost control to terminate.
- **(PT) Programmed Time Between Defrost Cycles** - specifies the initial time period, in minutes, from the end of one cycle to the start of the next. The actual time may be adjusted automatically based on the length of time that an initial defrost cycle may have taken to complete.
- **(MC) Maximum Defrost Cycle Time** - specifies the maximum amount of time, in minutes, that a defrost cycle will last.
- **(AT) Adjusted Time Between Defrost Cycles** - read-only feedback property indicating an adjustment made to the **Programmed Time Between Defrost Cycle Times** based on previous cycles. If the previous cycle does not raise the temperature up to the value defined in the **Exit Defrost Setpoint**

parameter, then the controller will reduce the time before the next cycle. If the previous cycle completed quickly (faster than 25% of **Maximum Defrost Cycle Time**), then the time in increased.

. **(LT) Defrost Action Time Remaining** - read-only feedback property indicating how much time, in minutes, remain before the defrost logic will look to perform the next action.

## 5.8 STAGE 1 AND 2

Stages 1 and 2 are used to provide compressor driven heating and cooling based on the directional status of the Reversing Valve. The two cooling stages, controlled by BO3 and BO4, are cycled on and off to maintain the zone within a programmable deadband around a programmable setpoint. Setup involves configuring the following items:

- Cooling Lockout Configuration
- Heating Lockout Configuration
- Stage 1 Properties
- Stage 2 Properties

### 5.8.1 COOLING LOCKOUT CONFIGURATION

You can configure the cooling stages to be locked out based on a temperature lockout setpoint for both outside air temperature and/or discharge air temperature. If you do not plan to have lockouts, you must still configure the properties accordingly. To configure, perform the following steps:

1. In NB-Pro, access *BO03*
2. **(TL) DAT Low Temp Lockout** defines the lowest sensed temperature allowed before cooling is interrupted or not permitted to stage on. Enter your desired lockout value. If you do not plan to use a lockout baed on Discharge Temperature, set **(TL) DAT Low Temp Lockout** = 999.0
3. **(CL) OAT Cooling Lockout** defines the lowest sensed temperature allowed before cooling is interrupted or not permitted to stage on. Enter your desired lockout value. If you do not plan to use a lockout based on Outside Air Temperature, set **(CL) OAT Cooling Lockout** = 999.0

### 5.8.2 HEATING LOCKOUT CONFIGURATION

You can configure the heating stages to be locked out based on a temperature lockout setpoint for both outside air temperature and/or discharge air temperature. If you do not plan to have lockouts, you must still configure the properties accordingly. To configure, perform the following steps:

1. In NB-Pro, access *BO03*
2. **(TH) DAT High Temp Lockout** defines the highest sensed temperature allowed before heating is interrupted or not permitted to stage on. Enter your desired lockout value. If you do not plan to use a lockout baed on Discharge Temperature, set **(TH) DAT Low Temp Lockout** = 999.0
3. **(HL) OAT Heating Lockout** defines the highest sensed temperature allowed before heating is interrupted or not permitted to stage on. Enter your desired lockout value. If you do not plan to use a lockout based on Outside Air Temperature, set **(CH) OAT Cooling Lockout** = 999.0

### 5.8.3 STAGE 1 PROPERTIES

To configure Cool 1 properties, perform the following steps:

1. In NB-Pro, access *BO03*
2. Configure your properties accordingly.

*Table 5-7: Stage 1 Property Details*

| Property | Value and Interpretation |
|---|---|
| (CO) Cooling: Temp Offset | Indicates the temperature offset from your cooling setpoint before cooling stage 1 is engaged. Note that the stage may engage if the Staging Delay expires prior to the temperature offset. |
| (CR) Cooling: Min Run Time | Minimum amount of time, in minutes, the stage will stay on - prevents short-cycling |

*Table 5-7: Stage 1 Property Details*

| Property | Value and Interpretation |
|---|---|
| (CS) Cooling: Min Off Time | Minimum amount of time, in minutes, the stage will stay off - prevents short-cycling. |
| (CX) Cooling: Staging Delay | Indicates the amount of time, in minutes, before cooling stage 1 is engaged. Note that the stage may engage if the Temp Offset is exceeded prior to the Staging Delay. |
| (CB) Cooling: Deadband | The stage will not dis-engage until the Zone Temperature falls below the cooling setpoint by the number of degrees specified here. |
| (HO) Heating: Temp Offset | Indicates the temperature offset from your cooling setpoint before heating stage 1 is engaged. Note that the stage may engage if the Staging Delay expires prior to the temperature offset. |
| (HR) Heating: Min Run Time | Minimum amount of time, in minutes, the stage will stay on - prevents short-cycling |
| (HS) Heating: Min Off Time | Minimum amount of time, in minutes, the stage will stay off - prevents short-cycling. |
| (HX) Heating: Staging Delay | Indicates the amount of time, in minutes, before heating stage 1 is engaged. Note that the stage may engage if the Temp Offset is exceeded prior to the Staging Delay. |
| (HB) Heating: Deadband | The stage will not dis-engage until the Zone Temperature rises above the heating setpoint by the number of degrees specified here. |
| (AL) Heating: Low OAT Aux Limit | Defines an outside air temperature setpoint below which the compressor is left off and only Auxiliary Heat is used to provide heat. This has no effect if no Aux Heat stages are defined. |

### 5.8.4 STAGE 2 PROPERTIES

o configure Stage 2 properties, perform the following steps:

1. In NB-Pro, access *BO04*
2. Configure your properties accordingly.

*Table 5-8: Stage 2 Property Details*

| Property | Value and Interpretation |
|---|---|
| (CO) Cooling: Temp Offset | Indicates the temperature offset from your cooling setpoint before cooling stage 2 is engaged. Note that the stage may engage if the Staging Delay expires prior to the temperature offset. |
| (CR) Cooling: Min Run Time | Minimum amount of time, in minutes, the stage will stay on - prevents short-cycling |

*Table 5-8: Stage 2 Property Details*

| Property | Value and Interpretation |
|---|---|
| (CS) Cooling: Min Off Time | Minimum amount of time, in minutes, the stage will stay off - prevents short-cycling. |
| (CB) Cooling: Deadband | The stage will not dis-engage until the Zone Temperature falls below the cooling setpoint by the number of degrees specified here. |
| (HO) Heating: Temp Offset | Indicates the temperature offset from your cooling setpoint before heating stage 1 is engaged. Note that the stage may engage if the Staging Delay expires prior to the temperature offset. |
| (HR) Heating: Min Run Time | Minimum amount of time, in minutes, the stage will stay on - prevents short-cycling |
| (HS) Heating: Min Off Time | Minimum amount of time, in minutes, the stage will stay off - prevents short-cycling. |
| (HB) Heating: Deadband | The stage will not dis-engage until the Zone Temperature rises above the heating setpoint by the number of degrees specified here. |

# 5.9 AUXILIARY HEAT

The Auxiliary Heat application is used by the Heatpump to provide auxiliary heat in the event that normal compressor based stages have been unable to satisfy the zone by providing supply temperature air higher than a specific setpoint defined in the Aux Heat object. Auxiliary Heat can be configured for a single-stage - utilizing BO5; or dual-stage - utilizing BO4 and BO5 on single-compressor units that need both a stage of auxiliary heat and an additional stage of auxiliary heat for emergency situations. Configuration involves the following steps:

. Auxiliary Heat Properties

## 5.9.1 AUXILIARY HEAT PROPERTIES

To configure the Auxiliary Heat properties, perform the following steps:

1. In NB-Pro, access *Aux Heat*
2. Configure properties accordingly.

*Table 5-9: Auxiliary Heat Property Details*

| Property | Value and Interpretation |
|---|---|
| (RO) Aux Heat Application | 0 = None<br>1 = Single Stage (K5)<br>2 = Dual Stage (K4 and K5)<br><br>Note - If Dual Stage has been selected, K4 will become unavailable as a second stage compressor, where K4 will function as the auxiliary stage of heat, and K5 will function as an emergency auxiliary stage of heat. |
| (FR) Fan Flow Required | Specifies whether flow is requires for auxiliary heat stages to operate |
| (BA) Balancing Enabled | Enables runtime-based balancing of stages as they are enabled (dual-stage only) |
| (SS) Aux Heat Setpoint | Supply Temperature Setpoint at which auxiliary heat is enabled |
| (EM) Emergency Aux Heat Setpoint | Supply Temperature Setpoint at which second stage of auxiliary heat is enabled |
| (ID) Interstage Delay | Specifies the time delay, in minutes, that is imposed between compressor stage enablement and auxiliary heat stage enablement. |
| (HL) OAT Lockout | If the outside air temperature rises above the value specified in this property, auxiliary heat will not energize. |
| (TH) Supply Temp Lockout | If the supply temperature rises above the temperature setpoint specified in this property, auxiliary heat will not energize. |

## 5.10 ECONOMIZER

The Heatpump application includes an Economizer. Based on outside air temperature, the economizer can be configured to open a damper during calls for cooling to provide free cooling, allowing the space to cool down using cooler outside air as an alternative to running all stages of cooling. The economizer can utilize an available second stage of heating or cooling, or use one of the four PID control loops to control a proportional motor.

To use the economizer, your Outside Air Temperature sensor must be configured (see previous steps). To configure the economizer, perform the following steps:

1. Using NB-Pro, access *Economizer*
2. Configure your properties accordingly.

*Table 5-10: Economizer Property Details*

| Property | Value and Interpretation |
|---|---|
| (EE) Economizer Enable | Specifies the PID Loop or Binary Output to be used for Economizer control.<br>0 = Disabled<br>1 = PID 1<br>2 = PID 2<br>3 = PID 3<br>4 = PID 4<br>5 = BO3<br>6 = BO5<br><br>**Note** - Using BO3 or BO5 will cancel out the second stage of heating or cooling respectively. |
| (OH) OAT High Limit | If the OAT rises above this setpoint, the economizer output will be set to the defined minimum position. |
| (OL) OAT Low Limit | If the OAT falls below this setpoint, the economizer output will be set to the defined minimum position. |
| (EM) Minimum Position | Specifies this minimum position for the economizer output when OAT rises or falls below the high and low limits |
| (ED) Economizer Staging Delay | Specifies how many minutes the controller will wait before using additional cooling stages after the economizer damper reaches 100% |
| (CM) Calculated Minimum Position | Displays the actual minimum position as calculated by the Economizer's internal control loop. |
| (MV) Reset Variable | Specifies an input for reset applications. |
| (MP) Reset Setpoint | Specifies the setpoint at which reset action occurs. |
| (MR) Maximum Reset | Specifies the maximum amount to reset the minimum position by when the variable reaches setpoint. |
| (ML) Reset Limit | Applied to determine the minimum position when (ML) is equal to the reset variable's input value. |

# 5.11 PID CONTROL

The NB-ASC(e) provides up to four (4) PID Control Loops for proportional control strategies. While some of these loops may be utilized by other strategies (such as Economizer or Valve Control - application dependent), the PID Control Loops can be used to control proportional equipment in an efficient manner.

Each PID Control is statically linked to a specific Analog Output. For example, PID Control 1 is linked to Analog Output 1, PID Control 2 is linked to Analog Output 2, etc.

The following section discusses the operation and configuration of PID Control in the NB-ASC(e).

*Table 5-11: PID Control Property Details*

| Property | Value and Interpretation |
|---|---|
| (SP) Loop Setpoint | Specifies the setpoint for loop control. The setpoint corresponds to the input variable specified in **(IC) Input Object** |
| (CS) Control Setpoint | Indicates the current setpoint used as part of the control loop, displays the calculated setpoint with any setup/ setback, or setpoint adjustment from a connected SBC-STAT that may be applied. |
| (PO) Percent Output | Indicates the scaled output as configured in the actual Analog Output. |
| (IN) Input Object Value | Indicates the current value of the input variable specified in **(IC) Input Object**. |
| (IC) Input Object | Specifies the input reference that the loop will proportionally control by. Valid options include:<br><br>0 = Disable<br>1 = Zone Temperature<br>2 = Supply Temperature<br>4 = UI1<br>5 = UI2<br>6 = UI3<br>7 = UI4<br>8 = UI5<br>9 = Zone Heating (follows adjusted heating setpoint)<br>10 = Zone Cooling (follows adjusted cooling setpoint)<br>11 = Outside Air Temperature |
| (MR) Maximum Reset | Specifies the maximum amount needed to reset to e loop setpoint based on when reset is being used. |
| (RC) Reset Variable Value | Indicates the current value of the reset variable specified in **(RV) Reset Variable**. |

*Table 5-11: PID Control Property Details*

| Property | Value and Interpretation |
|---|---|
| (RV) Reset Variable | Specifies the reset input reference that the loop will use to perform reset control. Valid options for reset include:<br><br>0 = Disable<br>1 = Zone Temperature<br>2 = Supply Temperature<br>4 = UI1<br>5 = UI2<br>6 = UI3<br>7 = UI4<br>8 = UI5<br>11 = Outside Air Temperature |
| (RS) Reset Setpoint | Specifies the setpoint at which reset action begins. When the reset variable's value exceeds this setpoint, reset action will be used to determine the **(CS) Control Setpoint**. |
| (RL) Reset Limit | Specifies the value at which maximum reset is applied, based on the reset variable's value. |
| (DB) Deadband | Specifies the deadband for proportional control. The deadband straddles the setpoint. For example, a value of 2.0 would be applied to both the left and right side of the action. |
| (PB) Proportional Band | Specifies the variable range over which the output is changed based on input variable changes. The proportional band is centered around the setpoint for the loop. |
| (RP) Reset Period | Specifies the amount of time, in seconds, over which error history is accumulated for reset control. |
| (RT) Rate | Specifies the derivative rate for the loop |
| (CS) Control Sign | Specifies control action of the loop 0 = Normal/Direct, 1 = Reverse. |
| (SU) Setup/Setback | Specifies how much the setpoint should be lowered during unoccupied periods. |
| (ID) Interlock Enable/Disable | Enabled/Disables interlock failure modes for the control loop. |
| (P1) Interlock 1 Position | Fail position for the control loop when Interlock 1 occurs |
| (P2) Interlock 2 Position | Fail position for the control loop when Interlock 2 occurs |
| (P3) No Flow Position | Fail position if no flow or fan operation occurs. |
| (CE) Control Enable | Activation point for the control loop. To enable the loop, set to a value of 1(Yes). |

# 5.12 SCHEDULING

Scheduling controls the current temperature setpoint of the NB-ASC(e). There are multiple ways scheduling can be performed in the controller. Internal schedules can be defined by the user. The user can determine when and in which schedule mode (or state) the *NB*-ASC will operate—occupied, warm-up, unoccupied, or night setback. The NB-ASC(e) accommodates four-mode schedules or two-mode schedules (utilizing Occupied and Unoccupied modes only). All schedule configuration is performed through the *Schedule* object.

## 5.12.1 WEEKLY SCHEDULING

The NB-ASC(e) contains a BACnet Schedule object, which is capable of performing 4-mode scheduling (Occupied, Unoccupied, Warm-up, and Night Setback) for all days of the week.

To configure a Schedule Object, perform the following steps:

1. Using NB-Pro, access *Schedules.*
2. Determine if you wish to use two-mode scheduling or four-mode scheduling. To use two-mode scheduling, you must write an ENUM data type value to schedule-default. A value of zero (0) indicates Unoccupied, where as a value of one (1) indicates Occupied mode. Otherwise, write an unsigned value (0 = Unoccupied, 1 = Warm-up, 2 = Occupied, 3 = Night Setback).
3. Configure time,value entries for each day of the week in **weekly-schedule**. A maximum of 6 time,value entries can be associated to one day of the week. If no values are entered for a specific day, the schedule will operate using the las time,value entry made in the previous day. No entries in any day of the week will result in the Schedule operating off the **schedule-default** value.

## 5.12.2 BROADCAST SCHEDULE

The Broadcast Schedule is a schedule sent out over the network by another controller such as an NB-GPC. The active internal schedule will be overridden if the NB-ASC(e) is configured to receive network broadcast schedules. If the **(RB) Receive Broadcast** property is enabled, the current schedule will reflect the **(CV) Current Value** property. To configure the NB-ASC(e) to receive network broadcast schedules:

1. Using NB-Pro, access *Broadcast Schedule*
2. Set **(RB) Receive Broadcast** = 1 (Yes)

## 5.12.3 POWER-UP STATE

If an unscheduled power loss occurs and power is restored, or if a soft reset of the controller is performed (Device;**(RS) Reset Controller** = 1), the controller will operate in the schedule mode defined by the user in the **Power-up State** (Device; **(PS) Power Up State**) property until a time synchronization received by the device from a time master. To set the schedule mode in which you want the device to operate upon power restoration or after a soft reset has occurred, select the value that corresponds to the desired power-up state. The possible states are listed in Table 5-12.

*Table 5-12 : Power-up States*

| value | Power-Up State |
|---|---|
| 0 | Unoccupied |
| 1 | Warm-up |
| 2 | Occupied |
| 3 | Night Setback |

### 5.12.4 HOST OVERRIDE

In multi-device or zone situations, it may be advantageous to have a host or other peer device directly control the schedule state of the controller without broadcasts. In this case, the controller has a Host Override function in the Schedule object that can be utilized.

To configure the device to have its schedule controlled by an external source, set Schedule; **(HE) Host Override Local Schedules** = 1 (Yes). Once set, the schedule of the device is then controlled through writes to the Schedule;**(HO) Host Schedule Setting** property.

The schedule mode set in **HO** will be the active mode unless:

- a broadcast is received
- an occupancy sensor is properly configured and occupancy is detected
- user override occurs

When host override is used, the true state of the schedule can be monitored through Zone Temperature; **(PO) Actual Schedule Status**. In previous iterations of NB-ASC(e) firmware, the present-value of the schedule could be monitored, but cannot in v6.03 or later due to BTL requirements.

## 5.13 SBC-STAT CONFIGURATION

The NB-ASC(e) supports SBC-STAT model devices, as referenced in Section 2. There are a few configuration options available for a connected SBC-STAT, which this section reviews

### 5.13.1 SETPOINT ADJUSTMENT CONFIGURATION

Setpoint adjustment configuration can be achieved by accessing *Zone Temperature* using NB-Pro. Options found in this section include the following in the table below:

*Table 5-13: Setpoint Adjustment Property Details*

| Property | Value and Interpretation |
|---|---|
| (ZS) Zone Midpoint | Specifies the comfort level for the zone. |
| (TS) User Setpoint Offset | Specifies an offset to apply to Zone Heating and Zone Cooling for PID Control. |
| (TM) User Adjust Increment | Specifies the magnitude of increment/decrement changes made to the setpoint. |
| (TT) User Adjust Duration | Specifies how much time, in minutes, a setpoint change is applied to the controller. |
| (SD) Calculated Setpoint Display | Specifies whether the offset, zone midpoint, heating setpoint, or cooling setpoint is displayed when a user performs setpoint adjustment. |

### 5.13.2 USER OVERRIDE

If the active schedule controlling the NB-ASC(e) is in unoccupied or night setback mode, user override is possible. If the user presses the up or down arrow push-button on the SBC-STAT2, SBC-STAT2D, or SBC-STAT3 and the **(SE) Override Enable/Disable** property is Enabled (value of 1), the unit will go into occupied mode.

The duration of this mode, which is also called extended occupancy, can be set by using the **(ED) Extended Occupancy Duration** property.

To configure the NB-ASC(e) for user override ability via an *SBC*-STAT, perform the following steps:

1. Using NB-Pro, access *Zone Temperature*
2. Set **(SE) User Override** = 1 (Enabled)
3. Set **(ED) Extended Occupancy Duration** to however many minutes you wish for user override mode to occur.

When the unit's schedule is in unoccupied mode and the user enables override from a connect STAT, occupancy will occur for the amount of minutes specified in **(ED) Extended Occupancy Duration**. Once the time has elapsed, the unit will revert back to its configured schedule mode.

## 5.14 OCCUPANCY DETECTION

The occupancy detection feature enables the NB-ASC(e) to automatically go to occupied mode, (also called extended occupancy) when a dedicated occupancy sensor indicates the monitored zone is occupied. The length of time that the controller will operate in extended occupancy is defined by the user in the FC01; **(MT) Extended Occupancy Duration** property. To configure the controller for occupancy detection capability, perform the following steps:

1. Using NB-Pro, access *Occupancy Detection*
2. Set **(IC) Status Input** to the input that the occupancy detector is connected to.
3. Set **(MD) Extended Occupancy Delay** to the desired number of seconds the detector must indicate that occupancy is detected before overriding the zone. This prevents false triggering of the occupancy detection in the event someone or something quickly passes through the zone.
4. Set **(MT) Extended Occupancy Duration** to the desired number of minutes the controller is to remain in occupied mode once the zone has been occupied.

**NOTE**

If **(MT) Extended Occupancy Duration** is not set to a value greater than zero, the controller will not enter extended occupancy when it is detected that the zone is occupied.

**NOTE**

The input selected for the Occupancy Detection application must be configured as a digital input.

## 5.15 PULSE COUNTING

The NB-ASC(e) provides an optically isolated digital input, which can be used for pulse counting. A Pulse application is provided, allowing users to perform count applications. To setup Pulse Counting, perform the following:

1. Using NB-Pro, access *Pulse Input*
2. Select a valid count mode from **(MD) Counter Mode**.
3. Enter your scale factor into **(SF) Pulse Scale Factor**

Your scaled value based on the number of pulses collected will appear in **(SV) Pulse Scale Value**.

# SECTION 6: FANCOIL CONFIGURATION

*The following section provides configuration details regarding the NB-ASC(e) Fancoil application. Please review the following sections carefully prior to configuring the controller.*

## IN THIS SECTION


Fancoil Sequence Overview ........ 6-3
Selecting Your Application Mode ........ 6-4
Universal Input Configuration........ 6-5
Sensor Configuration ........ 6-5
Digital Inputs ........ 6-5
Linear Sensors (0-10VDC)........ 6-5
4-20mA Sensors ........ 6-6
Thermistor Input ........ 6-6
Analog SBC-STAT1........ 6-7
Supply & Outside Temperature Configuration........ 6-8
Supply/Discharge Temperature........ 6-8
Outside Air Temperature ........ 6-8
Outside Air Temperature Broadcasts ........ 6-8
Output Configuration........ 6-9
Fan Control ........ 6-10
Fan Properties ........ 6-10
Fan Status........ 6-11
Electric Reheat........ 6-12
Electric Reheat Properties ........ 6-12
Valve Control........ 6-13
Valve Control Properties ........ 6-13
Economizer........ 6-15
PID Control ........ 6-16
Scheduling ........ 6-18
Weekly Scheduling........ 6-18
Broadcast Schedule........ 6-18
Power-up State........ 6-18
Host Override........ 6-19
SBC-STAT Configuration ........ 6-20
Setpoint Adjustment Configuration ........ 6-20
User Override........ 6-20
Occupancy Detection........ 6-21
Pulse Counting........ 6-22

# 6.1 FANCOIL SEQUENCE OVERVIEW

The Fancoil application of the NB-ASC(e) provides standard fancoil control utilizing zone temperature control with integrated (optional) outside air temperature and supply temperature (discharge temp) mediation. The fan can be configured for single speed or multi-speed (up to 3 speeds max), include stages of electric reheat, as well as valve control options for additional heating and cooling with integrated changeover.

- Five (5) Universal Inputs - configurable for:
  - Outside Air Temp & Discharge Air Temp Sensors
  - Occupancy Detection Sensor
  - Proof of Flow Sensing
  - ‘General Purpose Input Sensing
- Five (5) Binary Outputs
  - Fan configurable for up to three fan speeds (BO1 = Low, BO2 = Medium, BO3 = High)
  - Electric reheat application with supply temperature lockout capabilities
- Valve Control
  - Can utilize un-used pairs of binary outputs (BO2/BO3 or BO4/BO5)
  - Can utilize an available PID Control Loop
  - Provides changeover control for heating and cooling with setpoints and definable input
  - Outside Air Temperature and Discharge Air Temperature Lockout
  - Configurable loop parameters
- Up to Four (4) dedicated PID Control sequences for proportional control
  - Each PID Control Loop is internally connected to a respective Analog Output (PID1 -> AO1, etc.)
  - PID Control Loops have several input variables including all inputs, and user adjust setpoints
  - PID Control Loops have several reset variables including all inputs, and user adjust setpoints
  - Interlock failure positioning
- Outside Air Temperature-based economizer control
  - Based on the input selected for Outside Air Temperature
  - Utilizes an available PID Control Loop
  - Provides minimum position control, staging delay, lockouts, and reset setpoint and limiting

The NB-ASC(e) also provides the following features:

- Configurable Heating and Cooling Setpoints
  - Includes defined setbacks for unoccupied and night setback schedule modes
- Weekly Schedule
  - Performs weekly-scheduling using a BACnet Schedule object.
  - Two-mode (Occupied/Unoccupied) or Four-Mode scheduling options
  - Host Override of Schedule control - allowing global controllers to force the unit into a mode
- Occupancy Detection
  - Assignable by Universal Input selection
  - Definable delays and duration timing for occupancy controlled units.
- Pulse Counting
  - Utilizes the opto-isolated digital input
  - Can be configured to count rising edges, falling edges, or both
  - Configurable factor and scale values.

## 6.2 SELECTING YOUR APPLICATION MODE

Prior to configuring equipment such as heating and cooling stages, you should first define the application mode that the Fancoil will conform to. There are three different application modes that the Fancoil can perform. They are as followed:

- Cooling Only - utilizes cooling only routines and optional economizer control. Optionally uses PID Control for other miscellaneous control applications carried out.
- Heating Only - utilizes heating only routines and optional economizer control. Optionally uses PID Control for other miscellaneous control applications carried out.
- Supply Dependent (VST) - performs both heating and cooling. Optionally uses PID Control for other miscellaneous control applications carried out.
- Cooling with Reheat - performs cooling with electric reheat

To configure your application mode, perform the following steps:

1. Using NB-Pro, access *Zone Temperature*
2. Locate **(BT) Control Mode**
3. For Cooling Only, set **(BT) Control Mode** = 1 (Cooling Only)
4. For Heating Only, set **(BT) Control Mode** = 2 (Heating Only)
5. For Heating & Cooling, set **(BT) Control Mode** = 3 (Supply Dependent)
6. For Cooling w/ Reheat, set **(BT) Control Mode** = 4 (Cooling w/Reheat)

# 6.3 UNIVERSAL INPUT CONFIGURATION

Universal Inputs permit the configuration of multiple sensor types, dependent on your application. The NB-ASC(e) also supports alarm capabilities within its Universal Inputs. The following section provides a quick-start reference for initial configuration of inputs. Complete information regarding each property available within Universal Inputs can be located in Section 5.

## 6.3.1 SENSOR CONFIGURATION

The following section discusses how to configure a Universal Input for a specific sensor type. Universal Input configuration includes modifying control logic programming inside the NB-ASC(e) using NB-Pro and IVR hardware jumper configuration. The IVR jumpers are located above the Universal Input terminal blocks on the top right-hand side of the unitary controller, as illustrated in Figure 6-1.

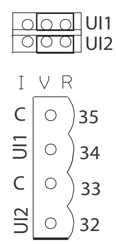


*Figure 6-1 Universal Input IVR Jumper Location*

### 6.3.1.1 DIGITAL INPUTS

To setup an input as a digital sensor, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 0 (Digital). Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than (0) No Fault Detected, verify input wiring.

*Table 6-1: Summary of Digital Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 0 | Digital sensor configuration |
| | reliability | 0 (No Fault Detected) | Input set to automatic mode |

### 6.3.1.2 LINEAR SENSORS (0-10VDC)

To setup an input for a 0-10VDC sensor, completely remove the IVR jumper to configure the UI for voltage mode.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 2 (Linear). Configure **min-pres-value** and **max-pres-value** to the minimum and maximum scaled values for **present-value** is

the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 6-2: Summary of an Example Linear Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 2 (Linear) | Linear sensor configuration |
| | min-pres-value | 0 | Lowest present-value scale reading for the target sensor |
| | max-pres-value | 100 | Highest present-value scale reading for the target sensor |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 6.3.1.3 4-20mA Sensors

To setup an input as a 4-20mA sensor, you must first configure the IVR jumper to IV, which places the input into a voltage setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 3 (4-20mA). Configure **min-pres-value** and **max-pres-value** to the minimum and maximum scaled values for **present-value** is the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 6-3: Summary of an Example 4-20mA Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) Sensor Type | 3 | 4-20mA sensor configuration |
| | min-pres-value | 0 | Lowest present-value scale reading for the target sensor |
| | max-pres-value | 100 | Highest present-value scale reading for the target sensor |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 6.3.1.4 Thermistor Input

To setup an input for thermistor readings, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 7 (Thermistor). Configure the minimum and maximum scaled values for **present-value** is the actual reading from the sensor. Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than 0 (No Fault Detected), verify input wiring.

*Table 6-4: Summary of Thermistor Input Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) sensor type | 7 (Thermistor) | Thermistor sensor configuration |
| | out-of-service | 0 (False) | Input set to automatic mode |

#### 6.3.1.5 ANALOG SBC-STAT1

To setup an input for temperature readings from an Analog SBC-STAT1, you must first configure the IVR jumper to VR, which places the input into a resistance setting.

In NB-Pro, access *UI0x* (*x* = Universal Input number), and set **(ST) Sensor Type** = 8 (Analog STAT1). Verify that **out-of-service** = 0 (False) to assure that the value displayed in **present-value** is the actual reading from the sensor. If **reliability** displays a value other than (0) Reliable, verify input wiring.

*Table 6-5: Summary of Analog STAT1 Configuration*

| Object | Property | Value | Description |
|---|---|---|---|
| UI0x | (ST) sensor type | 8 (STAT1) | Analog STAT1 sensor configuration |
| | out-of-service | 0 (False) | Input set to automatic mode |

## 6.4 SUPPLY & OUTSIDE TEMPERATURE CONFIGURATION

The following section reviews configuration of the Supply and Outside Air Temperature sensors within the Fancoil application.

### 6.4.1 SUPPLY/DISCHARGE TEMPERATURE

The Fancoil application can utilize supply temperature for heating/cooling lockout, as well as integrated supply control. Configuring this area is mandatory to make certain that stages of heating and cooling are or are not locked out. If your application does not require this temperature sensor, set **out-of-service** = 1 (Yes), located in *Supply Temperature*.

If your application does involve integration of supply sensing, perform the following steps:

1. In NB-Pro, access *Supply Temperature*
2. Verify **out-of-service** = 0 (False)
3. Set **(IC) Input Object Select** to the Universal Input your Supply Temperature sensor is connected to.
4. Set **(DD) Auto Mode Deadband** to a realistic value. The Auto Mode Deadband specifies the temperature difference by which the supply air must either exceed the heating setpoint to engage heating mode or fall below the cooling setpoint to engage cooling mode. A value of 0.0 will disable supply deadband control over heating and cooling modes. This property is commonly used when your application has been configured for Supply Dependent mode.

### 6.4.2 OUTSIDE AIR TEMPERATURE

The Fancoil application utilizes outside air temperature for locking out heating and cooling stages, as well as for operating the Economizer routine. Configuring this area is mandatory to make certain that stages of heating and cooling are or are not locked out. If your application does not require this temperature sensor, set **out-of-service** = 1 (True) located in *Outside Air Temperature.*

If your application does involve integration of outside air temperature sensing, perform the following steps:

1. In NB-Pro, access *Outside Air Temperature*
2. Verify **out-of-service** = 0 (No)
3. Set **(IC) Input Object Select** to the Universal Input your Outside Air Temperature sensor is connected to.

### 6.4.3 OUTSIDE AIR TEMPERATURE BROADCASTS

The Fancoil application can receive an Outside Air Temperature broadcast from another unit that is configured to perform broadcasting (such as an SBC-GPC). To receive or send Outside Air Temperature, perform the following steps:

1. In NB-Pro, access *Outside Air Temp*. *Broadcast*
2. To receive a broadcast Outside Air Temperature, set **(RB) Receive Broadcast** = 1 (Yes).
3. To send Outside Air Temperature broadcasts, set **(BE) Broadcast Enable** = 1 (Yes).

## 6.5 OUTPUT CONFIGURATION

In order for outputs to be manipulated by any of the control strategies in the controller, all of the outputs (both Analog and Binary) must be configures for automatic control.

To configure Analog Outputs, perform the following steps:

1. Using NB-Pro, access *AO0x(where x = Analog Output termination).*
2. For each Analog Output, verify **out-of-service** = 0 (False).
3. Minimum and Maximum Scaled Voltages and Engineering Units can be assigned here as well.

When Analog Outputs are configured for manual mode, they cannot be manipulated by any of the internal control processes. However, the Analog Output can be manually written to.

To configure Binary Outputs, perform the following steps:

1. Using NB-Pro, access *B00x (where x = Binary Output termination).*
2. For each Binary Output, verify **out-of-service** = 0 (False).
3. Output polarity may also be configured in this section.

## 6.6 FAN CONTROL

The Fancoil application can be configured for single speed, 2-speed, or 3-speed fan control based on the type of unit you are working with. The following section reviews configuration for the following items:

- Fan Properties
- Fan Status

### 6.6.1 FAN PROPERTIES

To configure the Fan Properties, perform the following steps:

1. In NB-Pro, access objects *Fan Speed* and *BO01.*
2. Configure all properties accordingly.

*Table 6-6: Fan Property Details*

| Object | Property | Value and Interpretation |
|---|---|---|
| Fan Speed | (SP) Fan Speeds | Specifies the number of fan speeds which will be utilized (1, 2, or 3) |
| Fan Speed | present-value | network point that allows network users to command the fan speed. |
| BO01 | (FD) Shutoff Delay | The amount of time in seconds that the fan will remain on after all stages of cooling have been de-energized. |
| BO01 | (FO) Occupied Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>1 (On) - Fan is always on. |
| BO01 | (FU) Unoccupied Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>1 (On) - Fan is always on. |
| BO01 | (FN) Night Setback Mode | 0 (Auto) - Fan turns off and off only when needed as defined by heating and cooling setpoints<br><br>1 (On) - Fan is always on. |
| BO01 | (DB) Fan Auto Mode Deadband | The property is used exclusively in conjunction with fan modes (FO, FU, FN) that are configured for Auto modes. When the device is in a schedule state where the fan operates in auto mode, the current zone temperature must exceed setpoint plus or minus the deadband temperature value in order for the fan to activate, followed by stages of heating or cooling. This property is useful for preventing the fan from turning on in situations where the zone temperature could possibly drift closely near mode changes. |

Table 6-6: Fan Property Details

| Object | Property | Value and Interpretation |
|---|---|---|
| BO01 | (MT) Min Run/Off Countdown Timer | Reflects the amount of time remaining (in seconds) if the fan output is currently respecting minimum-on or minimum-off time.<br><br>Examples:<br><br>If (FR) = 1.0 and the Fan Turns on, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 60 seconds would be reflected by this property.<br><br>If (FS) = 2.0 and the fan turns off, (MT) will provide a countdown equal to the amount of time remaining for the minimum period being respected. In this case, a countdown from 120 seconds would be reflected by this property. |

## 6.6.2 FAN STATUS

Fan Status is used to track the actual status of the fan output through use of a proof of flow sensor connected to a universal input or even the opto-isolated digital input. If you use a proof of flow sensor and the flow status does not match the output status, stages of heating and cooling will be locked out.

To configure Fan Status, perform the following steps:

1. Using NB-Pro, access *Proof of Flow*
2. Select the input where your proof of flow sensor it attached using **(IC) Status Input**.
3. Enter a delay value **(PD) Delay**. This will impose a delay before considering a positive flow indication.

## 6.7 ELECTRIC REHEAT

The Fancoil application can be configured to utilize stages of electric reheat. Dependent on your fan speed configuration, the controller can utilize two-stage electric reheat using output pairs BO2/BO3 and BO4/BO5. For single speed fan application, you can also utilize four-stage electric reheat using Binary Outputs 2 through 5. The following section reviews configuration for the following items:

. Electric Reheat Properties

### 6.7.1 ELECTRIC REHEAT PROPERTIES

To configure the Electric Reheat Properties, perform the following steps:

1. In NB-Pro, access *Reheat Ctrl*
2. Configure all properties accordingly.

*Table 6-7: Electric Reheat Properties Details*

| Property | Value and Interpretation |
|---|---|
| (RO) Reheat Mode | Used to configure the electric reheat application for use with the controller.<br>0 = Disabled<br>1 = 2-stage (K2/K3)<br>2 = 2-stage (K4/K5)<br>3 = 4-stage (K2/K3/K4/K5) |
| (FR) Stages Requiring Flow | Specifies which stages require a proof of flow indication prior to engaging. |
| (AF) Require Max Airflow | Typically set for No |
| (MX) Max Supply Temp | Specifies the maximum supply temperature above which reheats will de-energize/lockout |
| (BA) Balance Stage Usage | When set to 1 (Yes), the controller will balance stage usage by comparing run hour totals from each output. Stages with lower run hours will be energized first. |
| (OF) Reheat Offset | Specifies, in degrees, the offset from the calculated heating and cooling setpoints that determine when staging occurs. |
| (ID) Stage Delay | Specifies the amount of time, in minutes, between stage energizing. |

# 6.8 VALVE CONTROL

Valve Control can be used to provide additional heating and cooling capabilities with changeover control for heating and cooling. Valve control can be configured to use paired binary outputs, or utilize an available PID Control Loop.

*Table 6-8: Valve Control Output Assignment Configuration*

| (VM) Valve Mode | Output Strategy |
|---|---|
| Pulse Width Modulation<br><br>or<br><br>Floating Point Motor Control | Valve Control 1 -<br>K2 (close) / K3 (open)<br><br>Valve Control 2 -<br>K4 (close) / K5 (open) |
| PID 1 | PID Control 1 / Analog Output 1 |
| PID 2 | PID Control 2 / Analog Output 2 |
| PID 3 | PID Control 3 / Analog Output 3 |
| PID 4 | PID Control 4 / Analog Output 4 |

The following section reviews configuration for the following items:

. Valve Control Properties

## 6.8.1 VALVE CONTROL PROPERTIES

To configure Valve Control Properties, perform the following:
To configure the Electric Reheat Properties, perform the following steps:

1. In NB-Pro, access *Valve Control x (*where *x* = object number*).*
2. Configure all properties accordingly.

*Table 6-9: Valve Control Properties Details*

| Property | Value and Interpretation |
|---|---|
| (VU) Valve Use | Specifies the use for the valve<br>0 = Disabled<br>1 = Cooling<br>2 = Heating |
| (VM) Valve Mode | Defines how the valve is physically controlled.<br>0 = Pulse With Modulation<br>1 = Floating Point Motor Control<br>2 = PID 1<br>3 = PID 2<br>4 = PID 3<br>5 = PID 4 |
| (UT) Update Threshold | Defines how often the desired valve position is updated and is used to minimize the actuation of the valve for insignificant changes. |

*Table 6-9: Valve Control Properties Details*

| Property | Value and Interpretation |
|---|---|
| (RI) Re-calibration Interval | Specifies the amount of time, in hours, between valve re-calibration periods. The valve is re-calibrated by driving the valve in the closed direction for the full travel time, then restoring the desired position. |
| (VO) Valve Offset | Added/Subtracted from the heating and cooling setpoints to determine the setpoint for the loop. |
| (VP) Valve Proportional Band | Specifies the amount of degrees over which the output valve is proportional to the error. The proportional band is offset from the setpoint (determined by VO) |
| (VI) Valve Integration Constant | Specifies the amount of error history (0 to 25.5%) used to calculate the desired position of the valve |
| (VT) Valve Travel Time | Specifies the amount of time, in seconds, it takes the valve motor to travel from fully closed to fully open. |
| (PP) Pulse Duration Period | Specifies the amount of time, in seconds, the valve is pulsed on when the loop is using Pulse Width Modulation as the configured Valve Mode. |
| (CD) Change Valve Direction | Specifies the direction of the valve. |
| (TL) DAT Low Temp Lockout | Specifies the lowest supply temperature reading permitted before cooling control is locked out. |
| (CL) OAT Cooling Lockout | Specifies the lowest outside air temperature reading permitted before cooling control is locked out. |
| (TH) DAT High Temp Lockout | Specifies the highest supply temperature reading permitted before heating control is locked out. |
| (HL) OAT Heating Lockout | Specifies the highest outside air temperature reading permitted before heating control is locked out. |
| (AM) Auto/Manual | Used to enable or disable control |
| (CC) Changeover Control Input | Indicates the input monitored for heating/cooling changeover |
| (CS) Changeover Cool Setpoint | Specifies a temperature below which the valve operation will switch to cooling mode |
| (HS) Changeover Heat Setpoint | Specifies a temperature above which the valve operation will switch to heating mode |

## 6.9 ECONOMIZER

The Fancoil application includes an Economizer. Based on outside air temperature, the economizer can be configured to open a damper during calls for cooling to provide free cooling, allowing the space to cool down using cooler outside air as an alternative to running all stages of cooling. The economizer can utilize an available second stage of heating or cooling, or use one of the four PID control loops to control a proportional motor.

To use the economizer, your Outside Air Temperature sensor must be configured (see previous steps). To configure the economizer, perform the following steps:

1. Using NB-Pro, access *Economizer*
2. Configure your properties accordingly.

*Table 6-10: Economizer Property Details*

| Property | Value and Interpretation |
|---|---|
| (EE) Economizer Enable | Specifies the PID Loop or Binary Output to be used for Economizer control.<br>0 = Disabled<br>1 = PID 1<br>2 = PID 2<br>3 = PID 3<br>4 = PID 4<br>5 = BO3<br>6 = BO5<br><br>**Note** - Using BO3 or BO5 will cancel out the second stage of heating or cooling respectively. |
| (OH) OAT High Limit | If the OAT rises above this setpoint, the economizer output will be set to the defined minimum position. |
| (OL) OAT Low Limit | If the OAT falls below this setpoint, the economizer output will be set to the defined minimum position. |
| (EM) Minimum Position | Specifies this minimum position for the economizer output when OAT rises or falls below the high and low limits |
| (ED) Economizer Staging Delay | Specifies how many minutes the controller will wait before using additional cooling stages after the economizer damper reaches 100% |
| (CM) Calculated Minimum Position | Displays the actual minimum position as calculated by the Economizer's internal control loop. |
| (MV) Reset Variable | Specifies an input for reset applications. |
| (MP) Reset Setpoint | Specifies the setpoint at which reset action occurs. |
| (MR) Maximum Reset | Specifies the maximum amount to reset the minimum position by when the variable reaches setpoint. |
| (ML) Reset Limit | Applied to determine the minimum position when (ML) is equal to the reset variable's input value. |

## 6.10 PID CONTROL

The NB-ASC(e) provides up to four (4) PID Control Loops for proportional control strategies. While some of these loops may be utilized by other strategies (such as Economizer or Valve Control - application dependent), the PID Control Loops can be used to control proportional equipment in an efficient manner.

Each PID Control is statically linked to a specific Analog Output. For example, PID Control 1 is linked to Analog Output 1, PID Control 2 is linked to Analog Output 2, etc.

The following section discusses the operation and configuration of PID Control in the NB-ASC(e).

*Table 6-11: PID Control Property Details*

| Property | Value and Interpretation |
|---|---|
| (SP) Loop Setpoint | Specifies the setpoint for loop control. The setpoint corresponds to the input variable specified in **(IC) Input Object** |
| (CS) Control Setpoint | Indicates the current setpoint used as part of the control loop, displays the calculated setpoint with any setup/ setback, or setpoint adjustment from a connected SBC-STAT that may be applied. |
| (PO) Percent Output | Indicates the scaled output as configured in the actual Analog Output (FD0x). |
| (IN) Input Object Value | Indicates the current value of the input variable specified in **(IC) Input Object**. |
| (IC) Input Object | Specifies the input reference that the loop will proportionally control by. Valid options include:<br><br>0 = Disable<br>1 = Zone Temperature<br>2 = Supply Temperature<br>4 = UI1<br>5 = UI2<br>6 = UI3<br>7 = UI4<br>8 = UI5<br>9 = Zone Heating (follows adjusted heating setpoint)<br>10 = Zone Cooling (follows adjusted cooling setpoint)<br>11 = Outside Air Temperature |
| (MR) Maximum Reset | Specifies the maximum amount needed to reset to e loop setpoint based on when reset is being used. |
| (RC) Reset Variable Value | Indicates the current value of the reset variable specified in **(RV) Reset Variable**. |

*Table 6-11: PID Control Property Details*

| Property | Value and Interpretation |
|---|---|
| (RV) Reset Variable | Specifies the reset input reference that the loop will use to perform reset control. Valid options for reset include:<br><br>0 = Disable<br>1 = Zone Temperature<br>2 = Supply Temperature<br>4 = UI1<br>5 = UI2<br>6 = UI3<br>7 = UI4<br>8 = UI5<br>11 = Outside Air Temperature |
| (RS) Reset Setpoint | Specifies the setpoint at which reset action begins. When the reset variable's value exceeds this setpoint, reset action will be used to determine the **(CS) Control Setpoint**. |
| (RL) Reset Limit | Specifies the value at which maximum reset is applied, based on the reset variable's value. |
| (DB) Deadband | Specifies the deadband for proportional control. The deadband straddles the setpoint. For example, a value of 2.0 would be applied to both the left and right side of the action. |
| (PB) Proportional Band | Specifies the variable range over which the output is changed based on input variable changes. The proportional band is centered around the setpoint for the loop. |
| (RP) Reset Period | Specifies the amount of time, in seconds, over which error history is accumulated for reset control. |
| (RT) Rate | Specifies the derivative rate for the loop |
| (CS) Control Sign | Specifies control action of the loop 0 = Normal/Direct, 1 = Reverse. |
| (SU) Setup/Setback | Specifies how much the setpoint should be lowered during unoccupied periods. |
| (ID) Interlock Enable/Disable | Enabled/Disables interlock failure modes for the control loop. |
| (P1) Interlock 1 Position | Fail position for the control loop when Interlock 1 occurs |
| (P2) Interlock 2 Position | Fail position for the control loop when Interlock 2 occurs |
| (P3) No Flow Position | Fail position if no flow or fan operation occurs. |
| (CE) Control Enable | Activation point for the control loop. To enable the loop, set to a value of 1(Yes). |

## 6.11 SCHEDULING

Scheduling controls the current temperature setpoint of the NB-ASC(e). There are multiple ways scheduling can be performed in the controller. Internal schedules can be defined by the user. The user can determine when and in which schedule mode (or state) the *NB*-ASC will operate—occupied, warm-up, unoccupied, or night setback. The NB-ASC(e) accommodates four-mode schedules or two-mode schedules (utilizing Occupied and Unoccupied modes only). All schedule configuration is performed through the *Schedule* object.

### 6.11.1 WEEKLY SCHEDULING

The NB-ASC(e) contains a BACnet Schedule object, which is capable of performing 4-mode scheduling (Occupied, Unoccupied, Warm-up, and Night Setback) for all days of the week.

To configure a Schedule Object, perform the following steps:

1. Using NB-Pro, access *Schedules.*
2. Determine if you wish to use two-mode scheduling or four-mode scheduling. To use two-mode scheduling, you must write an ENUM data type value to schedule-default. A value of zero (0) indicates Unoccupied, where as a value of one (1) indicates Occupied mode. Otherwise, write an unsigned value (0 = Unoccupied, 1 = Warm-up, 2 = Occupied, 3 = Night Setback).
3. Configure time,value entries for each day of the week in **weekly-schedule**. A maximum of 6 time,value entries can be associated to one day of the week. If no values are entered for a specific day, the schedule will operate using the las time,value entry made in the previous day. No entries in any day of the week will result in the Schedule operating off the **schedule-default** value.

### 6.11.2 BROADCAST SCHEDULE

The Broadcast Schedule is a schedule sent out over the network by another controller such as an NB-GPC. The active internal schedule will be overridden if the NB-ASC(e) is configured to receive network broadcast schedules. If the **(RB) Receive Broadcast** property is enabled, the current schedule will reflect the **(CV) Current Value** property. To configure the NB-ASC(e) to receive network broadcast schedules:

1. Using NB-Pro, access *Broadcast Schedule*
2. Set **(RB) Receive Broadcast** = 1 (Yes)

### 6.11.3 POWER-UP STATE

If an unscheduled power loss occurs and power is restored, or if a soft reset of the controller is performed (Device;**(RS) Reset Controller** = 1), the controller will operate in the schedule mode defined by the user in the **Power-up State** (Device; **(PS) Power Up State**) property until a time synchronization received by the device from a time master. To set the schedule mode in which you want the device to operate upon power restoration or after a soft reset has occurred, select the value that corresponds to the desired power-up state. The possible states are listed in Table 6-12.

*Table 6-12 : Power-up States*

| value | Power-Up State |
|---|---|
| 0 | Unoccupied |
| 1 | Warm-up |
| 2 | Occupied |
| 3 | Night Setback |

### 6.11.4 HOST OVERRIDE

In multi-device or zone situations, it may be advantageous to have a host or other peer device directly control the schedule state of the controller without broadcasts. In this case, the controller has a Host Override function in the Schedule object that can be utilized.

To configure the device to have its schedule controlled by an external source, set Schedule; **(HE) Host Override Local Schedules** = 1 (Yes). Once set, the schedule of the device is then controlled through writes to the Schedule;**(HO) Host Schedule Setting** property.

The schedule mode set in **HO** will be the active mode unless:

- a broadcast is received
- an occupancy sensor is properly configured and occupancy is detected
- user override occurs

When host override is used, the true state of the schedule can be monitored through Zone Temperature; **(PO) Actual Schedule Status**. In previous iterations of NB-ASC(e) firmware, the present-value of the schedule could be monitored, but cannot in v6.03 or later due to BTL requirements.

## 6.12 SBC-STAT CONFIGURATION

The NB-ASC(e) supports SBC-STAT model devices, as referenced in Section 2. There are a few configuration options available for a connected SBC-STAT, which this section reviews

### 6.12.1 SETPOINT ADJUSTMENT CONFIGURATION

Setpoint adjustment configuration can be achieved by accessing *Zone Temperature* using NB-Pro. Options found in this section include the following in the table below:

*Table 6-13: Setpoint Adjustment Properties*

| Property | Value and Interpretation |
|---|---|
| (ZS) Zone Midpoint | Specifies the comfort level for the zone. |
| (TS) User Setpoint Offset | Specifies an offset to apply to Zone Heating and Zone Cooling for PID Control. |
| (TM) User Adjust Increment | Specifies the magnitude of increment/decrement changes made to the setpoint. |
| (TT) User Adjust Duration | Specifies how much time, in minutes, a setpoint change is applied to the controller. |
| (SD) Calculated Setpoint Display | Specifies whether the offset, zone midpoint, heating setpoint, or cooling setpoint is displayed when a user performs setpoint adjustment. |

### 6.12.2 USER OVERRIDE

If the active schedule controlling the NB-ASC(e) is in unoccupied or night setback mode, user override is possible. If the user presses the up or down arrow push-button on the SBC-STAT2, SBC-STAT2D, or SBC-STAT3 and the **(SE) Override Enable/Disabl**e property is Enabled (value of 1), the unit will go into occupied mode.

The duration of this mode, which is also called extended occupancy, can be set by using the **(ED) Extended Occupancy Duration** property.

To configure the NB-ASC(e) for user override ability via an *SBC*-STAT, perform the following steps:

1. Using NB-Pro, access *Zone Temperature*
2. Set **(SE) User Override** = 1 (Enabled)
3. Set **(ED) Extended Occupancy Duration** to however many minutes you wish for user override mode to occur.

When the unit's schedule is in unoccupied mode and the user enables override from a connect STAT, occupancy will occur for the amount of minutes specified in **(ED) Extended Occupancy Duration**. Once the time has elapsed, the unit will revert back to its configured schedule mode.

## 6.13 OCCUPANCY DETECTION

The occupancy detection feature enables the NB-ASC(e) to automatically go to occupied mode, (also called extended occupancy) when a dedicated occupancy sensor indicates the monitored zone is occupied. The length of time that the controller will operate in extended occupancy is defined by the user in the FC01; **(MT) Extended Occupancy Duration** property. To configure the controller for occupancy detection capability, perform the following steps:

1. Using NB-Pro, access *Occupancy Detection*
2. Set **(IC) Status Input** to the input that the occupancy detector is connected to.
3. Set **(MD) Extended Occupancy Delay** to the desired number of seconds the detector must indicate that occupancy is detected before overriding the zone. This prevents false triggering of the occupancy detection in the event someone or something quickly passes through the zone.
4. Set **(MT) Extended Occupancy Duration** to the desired number of minutes the controller is to remain in occupied mode once the zone has been occupied.

**NOTE**

If **(MT) Extended Occupancy Duration** is not set to a value greater than zero, the controller will not enter extended occupancy when it is detected that the zone is occupied.

**NOTE**

The input selected for the Occupancy Detection application must be configured as a digital input.

## 6.14 PULSE COUNTING

The NB-ASC(e) provides an optically isolated digital input, which can be used for pulse counting. A Pulse application is provided, allowing users to perform count applications. To setup Pulse Counting, perform the following:

1. Using NB-Pro, access Pulse Input
2. Select a valid count mode from **(MD) Counter Mode**.
3. Enter your scale factor into **(SF) Pulse Scale Factor**

Your scaled value based on the number of pulses collected will appear in **(SV) Pulse Scale Value**.

# Rooftop Objects & Properties

*The following tables contain listings of the BACnet objects and property assignments. Each property is listed with its identifier number, data type, access code, storage, default value (if any) and a brief description of its functionality.*

## In this Appendix


Device ... A-3
Zone Temperature ... A-7
Universal Inputs 1-5 ... A-12
Supply Temperature ... A-13
Outside Air Temperature ... A-14
Relative Humidity ... A-15
Analog Outputs 1-4 ... A-16
Analog Values (Setpoints) 1-7 ... A-17
Pulse Input ... A-18
BO01 ... A-19
BO02 ... A-21
BO03 ... A-23
BO04 ... A-24
BO05 ... A-26
Schedule ... A-27
Economizer ... A-28
PID Control 1 - 4 ... A-30
Occupancy Detection ... A-33
Proof of Flow ... A-34
Outside Air Temp. Broadcast ... A-35
Broadcast Schedule ... A-36

# A.1 DEVICE

**NOTE**

The Device object is represented in *NB*-Pro as follows:
**AAM Rooftop xxxxxxxxxx**
(where xxxxxxxxxx is the Unitary Controller serial number)

The instance must be a unique number from 0 to 4194302. By default, AAM sets the value in such a way that it is unique to AAM products based off the unit's serial number, however the user must ensure the device instance is unique on the job site's network.

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_ identifier** | 75 | BACnet ObjID | RO | EEPROM<br>Device (8), Instance *serial number* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>AAM NB-GPC *serial number* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Device (8) | indicates membership in a particular object type class. |
| **system_status** | 112 | BACnet ObjID | RO | -<br>0 | indicates the current physical and logical status of the BACnet Device. |
| **vendor_name** | 121 | CharStr | RO | NRAM<br>American Auto-Matrix | identifies the manufacturer of the BACnet Device. |
| **vendor_ identifier** | 120 | Unsigned | RO | -<br>6 | a unique vendor identification code, assigned by ASHRAE, which is used to distinguish proprietary extensions to the protocol. |
| **model_name** | 70 | CharStr | RO | NRAM<br>NB-GPC1 | indicates the vendor’s name used to represent the model of the device. |
| **firmware_ revision** | 44 | CharStr | RO | NRAM<br>*revision number* | indicates the level of firmware installed in the device. |
| **application_ software_ version** | 12 | CharStr | RO | NRAM<br>*version number* | identifies the version of application software installed in the device. |
| **protocol_ version** | 98 | Unsigned | RO | -<br>1 | indicates the version of the BACnet protocol supported by this BACnet Device. |
| **protocol_ revision** | 139 | Unsigned | RO | -<br>2 | indicates the minor revision level of the BACnet standard. |
| **protocol_ services_ supported** | 97 | BACnet Services Supported | RO | - | indicates which standardized protocol services are supported by this device's protocol implementation. |
| **protocol_ object_ types_ supported** | 96 | BACnet Object Types Supported | RO | - | indicates which standardized object types are supported by this device's protocol implementation. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_list** | 76 | BACnet Array | RO | - | a list of each object within the device that is accessible through BACnet services. |
| **max_apdu_ length_ accepted** | 62 | Unsigned | RO | NRAM 480 | specifies the maximum number of information frames the node may send before it must pass the token. |
| **segmentation_ supported** | 107 | BACnet Segment ation | RO | - 3 | indicates whether the device supports segmentation of messages and, if so, whether it supports segmented transmission, reception, or both. |
| **local_time** | 57 | Time | RW | - | indicates the time of day to the best of the device's knowledge. |
| **local_date** | 56 | Date | RW | - | indicates the date to the best of the device's knowledge. |
| **apdu_timeout** | 11 | Unsigned | RW | NRAM 3000 | indicates the amount of time, in milliseconds, between retransmissions of an APDU requiring acknowledgment for which no acknowledgment has been received. |
| **number_of_ apdu_retries** | 73 | Unsigned | RW | NRAM 1 | indicates the maximum number of times that an APDU shall be retransmitted. |
| **time_ synchronization _recipients** | 116 | List of BACnet recipients | RW | NRAM {} | a list of one device to which the device may automatically send a Time Synchronization request. |
| **max_master** | 64 | Unsigned | RW | EEPROM 127 | specifies the highest possible address for master nodes and shall be less than or equal to 127. |
| **max_info_ frames** | 63 | Unsigned | RW | NRAM 10 | specifies the maximum number of information frames the node may send before it must pass the token. |
| **device_ address_ binding** | 30 | List | RW | - 30 | a list of the device addresses that will be used when the remote device must be accessed via a BACnet service request. |
| **BU** | 16758 | Bool | RW | RAM 0 | **Backup Control**<br>**BU** = 1 forces backup of **AE** and digital outputs 1-5 **RH** to EEPROM. |
| **CC** | 16770 | UInt | RW | EE 0 | **Count of Clock Fails**<br>This counter increments upon hardware failure but can also be advanced during the removal of power. |
| **CM** | 16779 | UInt | R | RAM Flash 255 | **Controller Manufacturer Code**<br>This property identifies the Manufacturer of the device. For American Auto-Matrix products this code is 255. This property is read-only. |
| **CP** | 16781 | UInt | RW | EE 0 | **Network Baud Rate**<br>0=9600<br>6=38.4K<br>7=19.2K<br>9=57.6K<br>10=76.8k |
| **CT** | 16784 | UInt | R | RAM Flash 201 | **Controller Type**<br>This property identified the type of device relative to firmware. |
| **DE** | 16795 | UInt | RW | RAM 0 | **Default Enable Command**<br>This property is used to restore configuration settings to factory defaults. To default the settings enter a value of 197 (a value which is unlikely to occur randomly). It may take several seconds to complete the reset. Note that this will not alter the unit ID or selected communications baud rate, but will erase all application configurations you may have performed. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **EM** | 16813 | Bool | RW | EE<br>0 | **English/Metric**<br>This property specifies the type of engineering units to be used for temperatures. If EM is set to zero, degrees are specified in Fahrenheit, if EM is set to one, degrees are specified in Celsius. Note that a change in this property automatically converts setpoints to the appropriate units. The display mode for digital thermostats is also changed but can be also be set separately.<br><br>0 = English<br>1 = Metric Units |
| **F1** | 16820 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 1 will not trip the fan.<br>If 1 and Interlock 1 is active, the fan is shut down. |
| **F2** | 16821 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 2 will not trip the fan.<br>If 1 and Interlock 2 is active, the fan is shut down. |
| **F3** | 16822 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 3 will not trip the fan.<br>If 1 and Interlock 3 is active, the fan is shut down. |
| **FT** | 16834 | UInt | R | RAM<br>Flash<br>4 | **Firmware Type**<br>Defines the class of firmware operating system used in this controller. Only flash updates of matching firmware type will be accepted. Upgrades and conversions to other classes of firmware will require special handling. Contact the factory for further information. This property is read-only. |
| **I1** | 16868 | UInt | RW | EE<br>0 | **Interlock 1 Input Channel**<br>0=Disabled, 1=UI1, 2=UI2, 3=UI3, 4=UI4, 5=UI5, 6=OIA/B |
| **I2** | 16869 | UInt | RW | EE<br>0 | **Interlock 2 Input Channel**<br>0=Disabled, 1=UI1, 2=UI2, 3=UI3, 4=UI4, 5=UI5, 6=OIA/B |
| **I3** | 16870 | UIntBool | RW | EE<br>0 | **Fan Failure Interlock**<br>0=Disabled<br>1=Enable |
| **IC** | 16876 | UInt | R | EE<br>0 | **EEPROM Default Count**<br>This counter increments whenever the EEPROM is restored to factory default settings (see Device;DE Default Enable). |
| **ID** | 16877 | UInt | RW | EE<br>Factory Set | **Unit ID**<br>assignable between 0 and 127. |
| **IS** | 16882 | bitstring | R | RAM<br>N/A | **Interlock Status**<br>0=Interlock 1, bit 1=Interlock 2, bit 2=Interlock 3 |
| **MS** | 16902 | Bool | RW | EE<br>0 | **Master/Slave Mode**<br>This defines the MS/TP mode of the device. When configured as Master, the device will be capable of being auto-discovered as well as pass network tokens and solicit requests to the network (such as time synchronizations). When configured as Slave,<br>the device must be manually addressed and cannot solicit requests to the network. |
| **OC** | 16917 | UInt | RW | EE<br>0 | **Count of Illegal Opcodes**<br>This counter increments upon firmware failure but can also be advanced during the removal of power. |
| **OS** | 16925 | Real | R | N/A | **Kernel Version**<br>Specifies the version number of the currently active Kernel Boot Block. This property is read-only. |
| **PD** | 16942 | UInt | RW | EE<br>5 | **Power-on Delay**<br>This property determines how long, in seconds, the device waits before energizing its outputs after a power loss or soft reset. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **PS** | 16951 | UInt | RW | EE<br>2 | **Power-up State**<br>This property determines which schedule state the device uses after a power loss and before its time is synchronized by a host. The selections are:<br><br>0=unoccupied<br>1=warmup<br>2=occupied<br>3=night setback |
| **RC** | 16963 | UInt | RW | EE<br>0 | **Count of Resets**<br>This counter increments each time power is applied to the controller. This counts power outages and noise related resets as well as resets initiated through Device;RS. |
| **RE** | 16964 | UInt | RO | RAM | **STATbus Reset Count**<br>This read only counter increments each time a diagnostic STATbus reset occurs and is used for diagnostic and troubleshooting purposes. |
| **RI** | 16967 | Bool | RW | EE<br>0 | **Reset Fan Failure Interlock**<br>This property resets Fan Failure Interlock. When Fan Failure Interlock is enabled to shut down the fan (Device;F3=1), and Fan Failure Interlock is active, setting Reset Fan Failure Interlock (Device;RI=1) allows the fan to restart. |
| **RS** | 16972 | Bool | RW | RAM<br>0 | **Reset**<br>0 = disabled (default),<br>1 = reset controller |
| **SN** | 16991 | UInt | R | EE<br>factory set | **Serial Number**<br>Displays the Serial Number of the controller. This property is read-only. |
| **SR** | 16994 | UInt | R | RAM<br>Flash | **Software Time Stamp**<br>This uniquely defines each flash firmware image. |
| **UP** | 17030 | UInt | R | EE<br>0 | **Flash Update Count**<br>This counter increment each time a new flash firmware image is accepted by the controller. |
| **VE** | 17043 | Real | R | RAM<br>Flash | **Software Version**<br>Specifies the version number of the currently active firmware. This property is read-only. |
| **WC** | 17050 | UInt | RW | EE<br>0 | **Count of Watchdog COP**<br>This counter increments upon firmware failure but can also be advanced during the removal of power. |
| **ZN** | 17084 | UInt | RW | EE<br>0 | **Zone Number**<br>Defines the zone number that this device is assigned to for receiving network broadcasts from other devices (e.g. schedule broadcast, OAT broadcast, etc) |
| **ZP** | 17085 | UInt | R | RAM<br>0 | **Count of High Current Pulses**<br>This counter increment when high pulses in the pulse input are received. |

# A.2 ZONE TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (0), Instance *N* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Universal Input N | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (0) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **BM** | 16754 | UInt | RW | EE 0 | **SSB Bus Mode** by default this should be set to Master (**BM**=0) unless multiple controllers are wired onto a single Sensor Bus (SSB). All additional controllers on the SSB must be configured as Slaves (**BM**=1). |
| **BT** | 16757 | UInt | RW | EE 0 | **Application (Box Type)** none (**BT**=0), cooling only (**BT**=1), heating only (**BT**=2), supply dependant (**BT**=3), and cooling w/reheat (**BT**=4). **BT** defaults to cooling only (**BT**=1). |
| **CC** | 16770 | Real | RO | RAM | **Current Cooling Setpoint** shows the current cooling temperature control setpoint. This will depend on setbacks and user adjustments. The attribute is read-only. |
| **CH** | 16775 | Real | RO | RAM | **Current Heating Setpoint** shows the current heating temperature control setpoint. this will depend on setbacks and user adjustments. The point is read-only. |

| | | | | | |
|---|---|---|---|---|---|
| **DF** | 16796 | UInt | RW | EE<br>0 | **Thermostat Display Format**<br>defines the format used to display the current temperature on the digital thermostat. The display of the tenths digit and the Fahrenheit/Celsius character are options. Also, the display may be eliminated.<br>0=##° (Default)<br>1=##.#°<br>2=##°F (or C)<br>3=##.#°F (or C)<br>4=No Temp Display |
| **DL** | 16798 | Real | RO | RAM | **Total Zone Demand Load**<br>indicates the heating/cooling demand for the zone in terms of temperature separation from setpoints. |
| **DM** | 16799 | UInt | RO | RAM | **Demand Mode**<br>indicates the demand for the zone. A satisfied zone will indicate “vent” (**DM**=0). If the *NB*-ASCe is in cooling mode and the zone temperature exceeds the cooling setpoint, “cool” is indicated (**DM**=1). If the controller is in heating mode and the zone temperature falls below the heating setpoint, “heat” is indicated (**DM**=2). |
| **DS** | 16803 | UInt | RW | EE<br>0 | **Thermostat Display Mode**<br>specifies whether English or Metric units are to be used for digital thermostat display on the *SBC*-STAT3. This mode is automatically altered as appropriate when the system Engineering Units property is set but may be modified later if required to display the alternate units.<br>0=Fahrenheit (Default)<br>1=Celsius |
| **DV** | 16805 | UInt | RW | EE<br>0 | **Thermostat Display Value**<br>by default (**DV**=0) each digital thermostat will display the identical temperature value (**ZT**) which is the average of each. With DV=1, each thermostat will display its own temperature (including offset)**.** |
| **ED** | 16808 | UInt | RW | EE<br>60 | **Extended Occupancy Time Duration**<br>specifies the amount of time in minutes to extend occupancy. **ED** has a default value of 60. |
| **ER** | 16816 | UInt | RW | RAM | **Extended Occupancy Time Remaining**<br>shows the amount of time remaining in extended occupancy. This value is set to the Extended Occupancy Duration (**ET**) when either push button on an analog thermostat is pressed. The *SBC*-STAT3 digital thermostat employs its User Menu for this function. **ER** is a read-only property that cannot be changed directly. |
| **G0** | 16837 | UInt | RO | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G1** | 16838 | UInt | RO | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G2** | 16839 | UInt | RO | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G3** | 16840 | UInt | RO | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **OF** | 16919 | Real | RW | EE<br>0 | **Temperature Adjustment**<br>defines an optional correction that may be required as an adjustment for the thermostat location and the possible measurement errors. |

| | | | | | |
|---|---|---|---|---|---|
| **PB** | 16940 | UInt | RW | EE<br>2200 | **Balance PIN**<br>this Personal Identification Number controls access to the Balance Menu. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access to the menu. A matching number must be entered by the Balancer. Values of 10,000 or greater will hide the menu. Entered P.I.N. numbers remain valid for only four (4) minutes after their use**. PB** has a default value of 2200. |
| **PG** | 16945 | UInt | RW | EE<br>0 | **Primary GID**<br>specifies the GID of the Primary thermostat in Primary GID mode (**RM**=8). If this thermostat is not available, then the Average temperature mode (**RM**=0) is used. |
| **PI** | 16947 | UInt | RW | EE<br>3300 | **Installer PIN**<br>this Personal Identification Number controls access to all menus. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access. A matching number must be entered by the Installer. Values of 10,000 or greater will hide the Install Menu. An authenticated Installer can access all menus. Entered P.I.N.s remain valid for only four minutes after the last button press. **PI** has a default value of 3300. |
| **PO** | 16949 | UInt | RO | RAM | **Present Occupancy Status**<br>shows the current occupancy status for the schedule and reflects any schedule overrides made.<br>0=Unoccupied<br>1=Warmup<br>2=Occupied<br>3=Night Setback |
| **PS** | 16951 | UInt | RW | EE<br>1100 | **Service PIN**<br>this Personal Identification Number controls access to the Service Menu. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access to the menu. A matching number must be entered by the service technician. Values of 10,000 or greater will hide the menu. Entered P.I.N.s remain valid for only four (4) minutes after their use. **PS** has a default value of 1100. |
| **PU** | 16952 | UInt | RW | EE<br>0000 | **User PIN**<br>this Personal Identification Number controls access to the User Menu. A value of 0 makes the menu always accessible. Values from 0001 to 9999 inclusive are used to control access to the menu. A matching number must be entered by the User. Values of 10,000 or greater will hide the menu. Entered P.I.N.s remain valid for only four (4) minutes after their use. **PU** has a default value of 0. |
| **RM** | 16969 | UInt | RW | EE<br>0 | **Reading Mode**<br>indicates the current reading mode. This would be either Cooling or Heating as specifies by the system box type (**BT**). If **BT** is set to supply dependant, the point will indicate the current mode as determined by the source/duct temperature.<br>0=Average<br>1=Highest<br>2=Lowest<br>3=Hi/Lo VST mode<br>4=Device 0<br>5=Device 1<br>6=Device 2<br>7=Device 3<br>8=Primary GID |

| | | | | | |
|---|---|---|---|---|---|
| **SD** | 16983 | UInt | RW | EE<br>0 | **Calculated Setpoint Display**<br>specifies what method is used to display setpoint adjustments on an SBC-STAT3 LCD screen.<br>0 = Disable (+/-2.5)<br>1 = Zone Midpoint (Zone Temperature: (ZS) Zone Midpoint)<br>2 = Heating Setpoint (Zone Temperature: (CH) Heating Setpoint)<br>3 - Cooling Setpoint (Zone Temperature: (CC) Cooling Setpoint) |
| **SE** | 16984 | Boolean | RW | EE<br>1 | **Override Disabled/Enabled**<br>enables or disables the user's ability to enter extended occupancy override.<br>0=Disabled<br>1=Enabled (Default) |
| **T0** | 17002 | Real | R | RAM | **Thermostat Reading for G0**<br>indicates the current temperature sensed by G0. |
| **T1** | 17003 | Real | R | RAM | **Thermostat Reading for G1**<br>indicates the current temperature sensed by G1. |
| **T2** | 17004 | Real | R | RAM | **Thermostat Reading for G2**<br>indicates the current temperature sensed by G2. |
| **T3** | 17005 | Real | R | RAM | **Thermostat Reading for G3**<br>indicates the current temperature sensed by G3. |
| **TM** | 17011 | Real | RW | EE<br>0.5 | **Offset Increment**<br>specifies the magnitude of incremental changes to the User Setpoint Offset (TS). The User Adjust Position (TP) is multiplied by TM to determine the User Setpoint Offset (TS) value. If the User Adjust Increment is 0, you will not be able to alter the setpoint. |
| **TP** | 17013 | UInt | RW | RAM<br>0 | **User Adjust Position**<br>the User Setpoint Offset (TS) can be raised or lowered in integral steps. This property tracks the current step. It can be set to any signed integer but will be constrained to +/-2 when adjusted by an analog thermostat or to +/-5 when set through a digital thermostat. The point is used in combination with the User Adjust Increment (TM) to calculate the User Setpoint Offset. |
| **TR** | 17014 | UInt | RO | RAM<br>0 | **User Adjust Remaining**<br>displays the time remaining before the User Setpoint Offset (TS) setting is reset. |
| **TS** | 17015 | Real | RW | RAM<br>0 | **Setpoint Offset**<br>defines an offset for application to PID setpoints. This point shows the current value calculated when you multiply the User Adjust Position (TM) by the User Adjust Increment (TP). This setting is temporary and is valid only for TT minutes unless TT=0. |
| **TT** | 17016 | UInt | RW | EE<br>120 | **User Adjust Duration**<br>the User Setpoint Offset (TS) is a temporary setting. The TT property defines in minutes the duration for which the setting applies. After that time, the User Adjust Position and User Adjust Offset are reset to 0 degrees. If the User Adjust Duration is 0, then setpoint changes remain in effect until modified. The default value for TT is 120. |
| **ZS** | 17087 | Real | RW | RAM<br>70 | **Heating/Cooling Setpoint**<br>displays the midpoint between the current cooling and heating setpoints. This property reflects changes in both setpoints. A change in ZS results in the appropriate shift of both the cooling and heating setpoint maintaining the effective deadband. |

# A.3 UNIVERSAL INPUTS 1-5

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (0), Instance *N* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Universal Input *N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (0) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general "health" of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is "reliable" as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **IF** | 16878 | UInt | RW | EE 0.0 | **Input Filtering** specifies the amount of time in tenths of seconds during which an input configured as a digital input must remain stable for the value to be considered reliable. This is also the weighted gain if the input is configured as analog. |
| **IP** | 16881 | Bool | RW | EE 0 | **Input Polarity**<br>0=normal<br>1=reverse |
| **OF** | 16919 | Real | RW | EE 0 | **UI Offset** specifies an offset value to be added to the Universal Input's present-value. |
| **ST** | 16996 | UInt | RW | EE 7 | **Sensor Type**<br>0= digital<br>2= full scale, linear from min_pres_value to max_pres_value<br>3= 4–20mA linear scaled from min_pres_value to max_pres_value<br>7= −22.0 to 122.0°F thermistor<br>1,4,5, and 6 are unused |

# A.4 SUPPLY TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (0), Instance *8* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM Supply Temperature | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (0) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **DD** | 16794 | Real | RW | EE 2.5°F | **Auto Duct Delta Temperature**<br>This defines the temperature difference by which the supply air must either exceed the heating setpoint to switch to 'heating mode', or fall below the cooling setpoint to engage 'cooling' mode. |
| **IC** | 16875 | UInt | RW | EE 0 | **Input Channel Select**<br>1=UI01<br>2=UI02<br>3=UI03<br>4=UI04<br>5=UI05 |
| **OF** | 16919 | Real | RW | EE 0 | **Supply/Duct Temperature Adjustment**<br>Defines an offset used to adjust the current present-value. |
| **SM** | 16990 | UInt | R | RAM NA | **Cooling/Heating Supply Mode**<br>0=Cooling<br>1=Heating |

# A.5 OUTSIDE AIR TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (0), Instance *9* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Outside Air Temperature | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (0) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **IC** | 16875 | UInt | RW | EE 0 | **Input Channel Select**<br>1=UI01<br>2=UI02<br>3=UI03<br>4=UI04<br>5=UI05 |
| **OF** | 16919 | Real | RW | EE 0 | **Outside Air Temperature Adjustment**<br>Defines an offset used to adjust the current present-value. |

# A.6 RELATIVE HUMIDITY

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (0), Instance *10* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Relative Humidity | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (0) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **DS** | 16803 | UInt | RW | EE 0 | **Display Relative Humidity?**<br>0 = No<br>1 = Yes |
| **OF** | 16919 | Real | RW | EE 0 | **Relative Humidity Adjustment**<br>Defines an offset used to adjust the current present-value. |

# A.7 ANALOG OUTPUTS 1-4

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Output (1), Instance 1-12 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Analog Output *N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Output (1) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM 0.0 | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM 100.0 | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **relinquish_ default** | 104 | Real | RW | NRAM 0.0 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **DT** | 16804 | UInt | RW | EE 252 | **Data type** This property specifies the precision of present-value. |
| **HS** | 16863 | Real | RW | EE 100.0 | **Maximum Scaled Voltage** This property specifies the actual analog output value for a present-value of max-pres-value. |
| **LS** | 16894 | Real | RW | EE 0.00 | **Minimum Scaled Voltage** This property specifies the actual analog output value for a present-value of min-pres-value. |

# A.8 ANALOG VALUES (SETPOINTS) 1-7

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Analog Value (2), Instance 1-7 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>*N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Analog Value (2) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM<br>0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM<br>0 | four flags that indicate the general “health” of the object. |
| **event_state** | 36 | BACnet Event State | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. Setting this value to True will allow for the present-value to become writable. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM<br>95 | indicates the measurement units of this object. |

# A.9 PULSE INPUT

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Input (3), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM *Pulse Input* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Binary Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM 0 | four flags that indicate the general “health” of the object. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. Setting this value to True will allow for the present-value to become writable. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the polarity for the input. |
| **MD** | 16896 | UInt | RW | EE 0 | **Pulse Counter Mode**<br>0=Off<br>1=Falling edge<br>2=Rising edge<br>3=Both edges |
| **NP** | 16910 | UInt | RW | RAM 0 | **Pulse Count Total**<br>This property indicates the total amount of pulses accumulated by the device. |
| **SF** | 16985 | Real | RW | EE 0.00 | **Pulse Multiplier**<br>This property specifies the multiplier to apply against (NP) Pulse Count Total to generate the value in (SV) Scaled Pulse Count. |
| **SV** | 16998 | Real | R | RAM 0.00 | **Scaled Pulse Value**<br>This property indicates the scaled pulse value, which is (NP) x (SF). |

# A.10 BO01

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO01 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **FD** | 16825 | UInt | RW | EE 30 | **Shutoff Delay (seconds)**<br>This property shows the amount of time, in seconds, the fan output will stay energized once the zone temperature reaches the deadband. |
| **FN** | 16829 | UInt | RW | EE 0 | **Night Setback Fan Mode**<br>1 = fan runs for the entire period<br>0 = fan shuts off when zone temp is within the deadband. |
| **FO** | 16830 | UInt | RW | EE 0 | **Occupied Fan Mode**<br>1 = fan runs for the entire period<br>0 = fan shuts off when zone temp is within the deadband. |
| **FR** | 16832 | Real | RW | EE 0.5 | **Minimum Run Time**<br>This property shows the minimum amount of time, in minutes, the fan output will stay energized. This prevents short cycling of the fan output. |
| **FS** | 16833 | Real | RW | EE 1.0 | **Minimum Off Time**<br>This property shows the minimum amount of time, in minutes, the fan output will stay de-energized. This prevents short cycling of the fan output. |
| **FU** | 16835 | UInt | RW | EE 0 | **Unoccupied Fan Mode**<br>1 = fan runs for the entire period<br>0 = fan shuts off when zone temp is within the deadband. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **DB** | 16792 | Real | RW | EE<br>0.0 | **Fan Auto Mode Deadband**<br>used exclusively in conjunction with fan modes (FO, FU, FN) that are configured for Auto modes. When the device is in a schedule state where the fan operates in auto mode, the current zone temperature must exceed setpoint plus or minus the deadband temperature value in order for the fan to activate, followed by stages of heating or cooling. This property is useful for preventing the fan from turning on in situations where the zone temperature could possibly drift closely near mode changes. |
| **MT** | 16903 | UInt | RO | RAM<br>- | **Min Run/Off Countdown Timer**<br>reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours**<br>This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE<br>0.0 | **Run Limit**<br>This property specifies a run time limit in hours for the output. |

# A.11 BO02

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 2 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO02 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **CL** | 16778 | Real | RW | EE 55.0 | **Cooling OAT Lockout** Defines the minimum Outside Air Temperature below which cooling will be locked out. Please note that stages will not be de-energized should the Outside Air Temperature fall below this temperature during an active cycle. |
| **DB** | 16792 | Real | RW | EE 0.0 | **Deadband** This stage will not be disengaged until the Zone Temperature falls below the cooling setpoint by the number of degrees specified by this property. |
| **MR** | 16901 | Real | RW | EE 3.0 | **Minimum Run Time** This property shows the minimum amount of time, in minutes, the stage will stay energized. |
| **MS** | 16902 | Real | RW | EE 7.0 | **Minimum Off Time** This property shows the minimum amount of time, in minutes, the stage will stay de-energized. |
| **MX** | 16905 | Real | RW | EE 20.0 | **Staging Delay (minutes)** This property indicates the maximum amount of time, in minutes, that the stage will operate before energizing the next stage of cooling. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours** This property shows the total amount of time, in hours, the output has been energized. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **RL** | 16968 | Real | RW | EE<br>0.0 | **Run Limit**<br>This property specifies a run time limit in hours for the output. |
| **TL** | 17010 | Real | RW | EE<br>45.0 | **Low Temperature Lockout**<br>Defines the minimum Supply Air Temperature below which cooling will be locked out. This offers protection against freeze up. |
| **TO** | 17012 | Real | RW | EE<br>0.0 | **Stage Temperature Offset**<br>This property indicates the temperature offset from setpoint required before engaging the stage. Note that the stage may also engage if the Staging Time of the prior stage expires. |
| **MT** | 16903 | UInt | RO | RAM<br>- | **Min Run/Off Countdown Timer**<br>reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time. |

# A.12 BO03

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 3 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO03 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **DB** | 16792 | Real | RW | EE 0.0 | **Deadband** This stage will not be disengaged until the Zone Temperature falls below the cooling setpoint by the number of degrees specified by this property. |
| **MR** | 16901 | Real | RW | EE 3.0 | **Minimum Run Time** This property shows the minimum amount of time, in minutes, the stage will stay energized. |
| **MS** | 16902 | Real | RW | EE 7.0 | **Minimum Off Time** This property shows the minimum amount of time, in minutes, the stage will stay de-energized. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours** This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE 0.0 | **Run Limit** This property specifies a run time limit in hours for the output. |
| **TO** | 17012 | Real | RW | EE 2.0 | **Stage Temperature Offset** This property indicates the temperature offset from setpoint required before engaging the stage. Note that the stage may also engage if the Staging Time of the prior stage expires. |
| **MT** | 16903 | UInt | RO | RAM - | **Min Run/Off Countdown Timer** reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time. |

# A.13 BO04

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 4 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO04 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general "health" of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **DB** | 16792 | Real | RW | EE 0.0 | **Deadband** This stage will not be disengaged until the Zone Temperature rises above the heating setpoint by the number of degrees specified by this property. |
| **HL** | 16858 | Real | RW | EE 80.0 | **Heating OAT Lockout** Heating stages will be locked out if the Outside Air Temperature rises above this setting. Please note that stages will not be de-energized should the OAT rise above this temperature during an active cycle. |
| **MR** | 16901 | Real | RW | EE 3.0 | **Minimum Run Time** This property shows the minimum amount of time, in minutes, the stage will stay energized. |
| **MS** | 16902 | Real | RW | EE 7.0 | **Minimum Off Time** This property shows the minimum amount of time, in minutes, the stage will stay de-energized. |
| **MX** | 16905 | Real | RW | EE 20.0 | **Staging Delay (minutes)** This property indicates the maximum amount of time, in minutes, that the stage will operate before energizing the next stage of heating. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours** This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE 0.0 | **Run Limit** This property specifies a run time limit in hours for the output. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **TH** | 17009 | Real | RW | EE<br>105.0 | **High Temp Lockout**<br>Heating stages will be locked out if the Supply Temperature rises above this setting. |
| **TO** | 17012 | Real | RW | EE<br>0.0 | **Stage Temp Offset**<br>This property indicates the temperature offset from setpoint required before engaging the stage. Note that the stage may also engage if the Staging Time of the prior stage expires. |
| **MT** | 16903 | UInt | RO | RAM<br>- | **Min Run/Off Countdown Timer**<br>reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time. |

# A.14 BO05

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 5 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO05 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **DB** | 16792 | Real | RW | EE 0.0 | **Deadband** This stage will not be disengaged until the Zone Temperature rises above the heating setpoint by the number of degrees specified by this property. |
| **MR** | 16901 | Real | RW | EE 3.0 | **Minimum Run Time** This property shows the minimum amount of time, in minutes, the stage will stay energized. |
| **MS** | 16902 | Real | RW | EE 7.0 | **Minimum Off Time** This property shows the minimum amount of time, in minutes, the stage will stay de-energized. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours** This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE 0.0 | **Run Limit** This property specifies a run time limit in hours for the output. |
| **TO** | 17012 | Real | RW | EE 2.0 | **Stage Temp Offset** This property indicates the temperature offset from setpoint required before engaging the stage. Note that the stage may also engage if the Staging Time of the prior stage expires. |
| **MT** | 16903 | UInt | RO | RAM - | **Min Run/Off Countdown Timer** reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time. |

# A.15 SCHEDULE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Schedule (17), Instance *1* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Schedule | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Schedule (17) | indicates membership in a particular object type class. |
| **present-value** | 85 | Unsigned or Enum | RO | RAM 0 | indicates the current schedule value |
| **effective-period** | 32 | Date Range | RO | RAM 0 | specifies the date rage in which the schedule is active. |
| **schedule-default** | 174 | Unsigned or Enum | RW | RAM 0 | specifies the datatype and value used for partial time-of-day scheduling. |
| **weekly-schedule** | 123 | Array List | RW | RAM 0 | contains time,value pair operations for each day of the week. |
| **list-of-object-property-references** | 54 | List | RO | NRAM 0 | contains the list of object properties that the schedule writes to. |
| **status-flags** | 111 | Bitstring | RO | NRAM 0 | indicates the general health of the object. |
| **reliability** | 103 | Enum | RO | NRAM 0 | indicates the reliability of the object. |
| **out-of-service** | 81 | Boolean | RO | NRAM 0 | indicates the service status of the object. |
| **priority-for-writing** | 88 | Unsigned | RW | NRAM 0 | indicates the priority which the Schedule will write with. |
| **HE** | 16853 | Bool | RW | EE 0 | H**ost Overrides**<br>0=disabled<br>1=enabled |
| **HO** | 16860 | UInt | RW | RAM 0 | **Host Schedule**<br>0 = unoccupied<br>1 = warm-up<br>2 = occupied<br>3 = night setback |
| **IS** | 16882 | UInt | RW | EE 3 | **Inactive Schedule State**<br>0 = unoccupied<br>1 = warm-up<br>2 = occupied<br>3 = night setback |
| **ZE** | 17081 | Bool | RW | EE 0 | **Receive Schedule**<br>0=No<br>1=Yes |

# A.16 ECONOMIZER

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (133), Instance *1* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Economizer | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **CM** | 16779 | Real | R | RAM N/A | **Calculated Minimum Position** This property displays the actual minimum position. |
| **ED** | 16808 | Real | RW | EE 1.0 | **Economizer Staging Delay (minutes)** This property specifies how many minutes the controller waits before using additional cooling stages after the economizer damper reaches 100. |
| **EE** | 16809 | UInt | RW | EE | **Economizer Enable** 0=Off 1=PID 1 2=PID 2 3=PID 3 4=PID 4 5=DO 3 6=DO 5 |
| **EM** | 16813 | Real | RW | EE 10.0 | **Economizer Minimum Position (%)** This property specifies the PID minimum position in percent for the economizer damper. |
| **ML** | 16090 | Real | RW | EE 0.0 | **Reset Limit for Economizer Minimum Protection** This property specifies the value at which maximum reset is used. When the value of the reset variable is equal to ML, the (MR) Maximum Reset is used in determining the calculated minimum position. |
| **MP** | 16900 | Real | RW | EE 0.0 | **Reset Setpoint for Economizer Minimum Position** This property specifies the value at which the reset action begins. When the value of the reset variable exceeds MP, reset action will be used in determining the economizer minimum position. |
| **MR** | 16901 | Real | RW | EE 0.0 | **Maximum Reset for Economizer Minimum Position** This property specifies the maximum amount to reset the minimum position setpoint (EM) by when reset is being used. |
| **MV** | 16904 | UInt | RW | EE 0.0 | **Reset Variable for Economizer Minimum Position** 0=Disabled 1=Zone Temperature 2=Supply Temperature 4=UI01 5=UI02 6=UI03 7=UI04 8=UI05 11=OAT |
| **OH** | 16920 | Real | RW | EE 60.0 | **OAT High Limit** This property specifies the Outside Air Temperature (OAT) high limit. If the OAT is above the high limit, the PID is set to the Economizer Minimum Position (EM). |
| **OL** | 16922 | Real | RW | EE 45.0 | **OAT Low Limit** This property specifies the Outside Air Temperature (OAT) low limit. If the OAT is below the low limit, the PID is set to the Economizer Minimum Position (EM). |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **CM** | 16779 | Real | R | RAM<br>N/A | **Calculated Minimum Position**<br>This property displays the actual minimum position. |
| **ED** | 16808 | Real | RW | EE<br>1.0 | **Economizer Staging Delay (minutes)**<br>This property specifies how many minutes the controller waits before using additional cooling stages after the economizer damper reaches 100. |

# A.17 PID CONTROL 1 - 4

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (133), Instance 11-14 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM PID Control 1-4 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **AO** | 16746 | UInt | RW | RAM NA | **Analog Output** This property shows the scaled output value used by the analog output. AO is the PO value scaled to min-pres-value and max-pres-value of the corresponding analog output. |
| **CE** | 16772 | Bool | RW | EE 0 | **Control Enable** 0=No 1=Yes |
| **CS** | 16783 | Real | R | RAM NA | **Control Setpoint** This property shows the calculated, actual loop control setpoint. CS reflects the unoccupied setup/setback as well as any reset and/or thermostat setpoint adjustment. |
| **DB** | 16792 | Real | RW | EE 0 | **Deadband** This property specifies the deadband around the setpoint for the control loop. When the value of the measured variable is within this deadband, the output signal remains constant at the midpoint of the throttling range (halfway between OH and OL). |
| **I1** | 16868 | Bool | RW | EE 0 | **Interlock 1 Enable** 0=Disabled 1=Enabled |
| **I2** | 16869 | Bool | RW | EE 0 | **Interlock 2 Enable** 0=Disabled 1=Enabled |
| **I3** | 16870 | Bool | RW | EE 0 | **No Flow Interlock Enable** 0=Disabled 1=Enable |
| **IC** | 16876 | UInt | RW | EE 0 | **Input Select** 0=disabled 1=Zone Temp 2=Supply Temp 4=UI1 5=UI2 6 =UI3 7 =UI4 8=UI5 9=Zone Heating 10=Zone Cooling 11=OAT 13=Relative Humidity |
| **IN** | 16880 | Real | R | RAM 0 | **Input Value** This property displays the value of the input selected in IC. |
| **MR** | 16901 | Real | RW | EE 0 | **Maximum Reset** This property specifies the maximum amount to reset the loop setpoint (SP) by when reset is being used. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **P1** | 16930 | Real | RW | EE<br>0.0 | **Interlock 1 Position**<br>This property specifies the PID output value when the Interlock 1 is enabled. |
| **P2** | 16931 | Real | RW | EE<br>0.0 | **Interlock 2 Position**<br>This property specifies the PID output value when the Interlock 2 is enabled. |
| **P3** | 16932 | Real | RW | EE<br>0.0 | **No Flow Interlock Position**<br>This property specifies the PID output value when the value of Fan Status is 0 (No Flow). |
| **PB** | 16940 | Real | RW | EE<br>0 | **Proportional Band**<br>This property specifies the input variable range over which the output value is proportional to the error value. The proportional band is centered around setpoint for the loop. |
| **PO** | 16949 | Real | RW | RAM<br>NA | **Percent Output**<br>This property shows the output value in hundredths of a percent. This value is calculated based on the error, change in error and past error for the control loop. PO is then scaled to the selected engineering units of the analog output and stuffed into the AO property, as well as the CV of the analog output. |
| **RC** | 16963 | Real | R | RAM<br>NA | **Reset Value**<br>This property displays the value of the input selected in RV. |
| **RL** | 16968 | Real | RW | EE<br>0 | **Reset Limit**<br>This property specifies the value at which maximum reset is used. When the value of the reset variable is equal to RL, the maximum reset (MR) is used in determining the calculated setpoint. |
| **RP** | 16971 | UInt | RW | EE<br>0 | **Reset Period**<br>This property specifies the reset period (in seconds) over which the error history is accumulated. If RP=10 seconds with a constant error of 2.0, then every second the error history would increase by 0.2. In five seconds, the error history would be 1.0 and at the end of ten seconds the error history would be 2.0. RP=0 disables integral action. |
| **RS** | 16972 | Real | RW | EE<br>0 | **Reset Setpoint**<br>This property specifies the value at which the reset action begins. When the value of the reset variable exceeds RS, reset action will be used in determining the calculated setpoint. |
| **RT** | 16973 | Real | RW | EE<br>0 | **Rate**<br>This property is the derivative rate and specifies a percentage of change in error that is to be used in calculating PO. The value is specified in percent-per-second. |
| **RV** | 16974 | UInt | RW | EE<br>0 | **Reset Variable**<br>0= disabled<br>1=Zone Temp<br>2=Supply Temp<br>4=UI1<br>5=UI2<br>6 =UI3<br>7 =UI4<br>8=UI5<br>11=OAT<br>13=Relative Humidity |
| **SG** | 16986 | UInt | RW | EE<br>0 | **Action**<br>0=normal (positive error causes an increase in output).<br>1=reverse (positive error causes a decrease in output) |
| **SP** | 16993 | Real | RW | EE<br>0 | **Loop Setpoint**<br>This property specifies the desired loop setpoint. This value is used with the unoccupied setup/setback and the reset to calculate the CS control setpoint. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **SU** | 16997 | Real | RW | EE<br>0 | **Setup/Setback**<br>This property specifies the amount to add (if SG=0) or subtract (if SG=1) from the setpoint during an unoccupied period. The adjusted setpoint will be displayed in CS. |

# A.18 OCCUPANCY DETECTION

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (131), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM Occupancy Detection | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Proprietary (254) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **IC** | 16876 | UInt | RW | EE 0 | **Input Select**<br>0 = None<br>1 = UI1<br>2 = UI2<br>3 = UI3<br>4 = UI4<br>5 = UI5<br>6 = OIA/B |
| **MD** | 16896 | UInt | RW | EE 30 | **Extended Occupancy Delay**<br>This property sets the amount of time, in seconds, the occupancy detector must remain on before the occupancy detector will override the zone. This prevents false triggers that might occur as others pass quickly through the zone. |
| **MR** | 16901 | UInt | R | RAM NA | **Extended Occupancy Remaining**<br>This property displays the time remaining for occupancy detector override. |
| **MS** | 16902 | UInt | R | RAM NA | **Occupancy Status**<br>0=No Detection<br>1=Detection |
| **MT** | 16903 | UInt | RW | EE 0 | **Extended Occupancy Duration**<br>This property defines, in minutes, the length of time to override the zone whenever occupancy is detected. |

# A.19 PROOF OF FLOW

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (131), Instance 2 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>Proof of Flow | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (131) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **PF** | 16944 | UInt | R | RAM<br>0 | **Proof of Flow Indication**<br>This property shows the status of the fan for Proof of Flow. |
| **IC** | 16876 | UInt | RW | EE | **Input Select**<br>0 =None<br>1 = UI1<br>2 = UI2<br>3 = UI3<br>4 = UI4<br>5 = UI5<br>6 = OIA/B |
| **PD** | 16942 | UInt | RW | EE<br>60 | **Proof of Flow Delay (seconds)**<br>This property shows the amount of time, in seconds, imposed before enabling a positive flow indication. |

# A.20 OUTSIDE AIR TEMP. BROADCAST

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (143), Instance 0 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>Outside Air Temp. Broadcast | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (143) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **CV** | 16785 | Real | R | RAM<br>0.0 | **Current Value**<br>shows the current value of the network broadcast schedule values received by the *NB*-ASC |
| **RB** | 16962 | Bool | RW | EE<br>0 | **Receive Broadcasts?**<br>0=No<br>1=Yes |
| **BE** | 16752 | Bool | RW | 0 | **OAT Broadcast Enable**<br>0=No<br>1=Yes |

# A.21 BROADCAST SCHEDULE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (143), Instance 5 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>Broadcast Schedule | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (143) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **CV** | 16785 | UInt | R | RAM<br>0 | **Current Value**<br>indicates the current schedule value being received. |
| **RB** | 16962 | Bool | RW | EE<br>0 | **Receive Broadcasts?**<br>0=No<br>1=Yes |

# HEATPUMP OBJECTS & PROPERTIES

*The following tables contain listings of the BACnet objects and property assignments. Each property is listed with its identifier number, data type, access code, storage, default value (if any) and a brief description of its functionality.*

## IN THIS APPENDIX


Device ..... B-3
Zone Temperature ..... B-7
Universal Inputs 1-5 ..... B-11
Supply Temperature ..... B-12
Outside Air Temperature ..... B-13
Relative Humidity ..... B-14
Analog Outputs 1-4 ..... B-15
Analog Values (Setpoints) 1-7 ..... B-16
Pulse Input ..... B-17
BO01 ..... B-18
BO02 ..... B-20
BO03 ..... B-22
BO04 ..... B-24
BO05 ..... B-26
Schedule ..... B-27
Economizer ..... B-28
Aux Heat ..... B-30
PID Control 1 - 4 ..... B-32
Occupancy Detection ..... B-35
Proof of Flow ..... B-36
Outside Air Temp. Broadcast ..... B-37
Broadcast Schedule ..... B-38

# B.1 DEVICE

**NOTE**

The Device object is represented in *NB*-Pro as follows:
**AAM Heatpump xxxxxxxxxx**
(where xxxxxxxxxx is the Unitary Controller serial number)

The instance must be a unique number from 0 to 4194302. By default, AAM sets the value in such a way that it is unique to AAM products based off the unit's serial number, however the user must ensure the device instance is unique on the job site's network.

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_ identifier** | 75 | BACnet ObjID | RO | EEPROM<br>Device (8), Instance *serial number* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>AAM NB-GPC *serial number* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Device (8) | indicates membership in a particular object type class. |
| **system_status** | 112 | BACnet ObjID | RO | -<br>0 | indicates the current physical and logical status of the BACnet Device. |
| **vendor_name** | 121 | CharStr | RO | NRAM<br>American Auto-Matrix | identifies the manufacturer of the BACnet Device. |
| **vendor_ identifier** | 120 | Unsigned | RO | -<br>6 | a unique vendor identification code, assigned by ASHRAE, which is used to distinguish proprietary extensions to the protocol. |
| **model_name** | 70 | CharStr | RO | NRAM<br>NB-GPC1 | indicates the vendor's name used to represent the model of the device. |
| **firmware_ revision** | 44 | CharStr | RO | NRAM<br>*revision number* | indicates the level of firmware installed in the device. |
| **application_ software_ version** | 12 | CharStr | RO | NRAM<br>*version number* | identifies the version of application software installed in the device. |
| **protocol_ version** | 98 | Unsigned | RO | -<br>1 | indicates the version of the BACnet protocol supported by this BACnet Device. |
| **protocol_ revision** | 139 | Unsigned | RO | -<br>2 | indicates the minor revision level of the BACnet standard. |
| **protocol_ services_ supported** | 97 | BACnet Services Supported | RO | - | indicates which standardized protocol services are supported by this device's protocol implementation. |
| **protocol_ object_ types_ supported** | 96 | BACnet Object Types Supported | RO | - | indicates which standardized object types are supported by this device's protocol implementation. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_list** | 76 | BACnet Array | RO | - | a list of each object within the device that is accessible through BACnet services. |
| **max_apdu_ length_ accepted** | 62 | Unsigned | RO | NRAM<br>480 | specifies the maximum number of information frames the node may send before it must pass the token. |
| **segmentation_ supported** | 107 | BACnet Segment ation | RO | -<br>3 | indicates whether the device supports segmentation of messages and, if so, whether it supports segmented transmission, reception, or both. |
| **local_time** | 57 | Time | RW | - | indicates the time of day to the best of the device's knowledge. |
| **local_date** | 56 | Date | RW | - | indicates the date to the best of the device's knowledge. |
| **apdu_timeout** | 11 | Unsigned | RW | NRAM<br>3000 | indicates the amount of time, in milliseconds, between retransmissions of an APDU requiring acknowledgment for which no acknowledgment has been received. |
| **number_of_ apdu_retries** | 73 | Unsigned | RW | NRAM<br>1 | indicates the maximum number of times that an APDU shall be retransmitted. |
| **time_ synchronization _recipients** | 116 | List of BACnet recipients | RW | NRAM<br>{} | a list of one device to which the device may automatically send a Time Synchronization request. |
| **max_master** | 64 | Unsigned | RW | EEPROM<br>127 | specifies the highest possible address for master nodes and shall be less than or equal to 127. |
| **max_info_ frames** | 63 | Unsigned | RW | NRAM<br>10 | specifies the maximum number of information frames the node may send before it must pass the token. |
| **device_ address_ binding** | 30 | List | RW | -<br>30 | a list of the device addresses that will be used when the remote device must be accessed via a BACnet service request. |
| **BU** | 16758 | Bool | RW | RAM<br>0 | **Backup Control**<br>**BU** = 1 forces backup of **AE** and digital outputs 1-5 **RH** to EEPROM. |
| **CC** | 16770 | UInt | RW | EE<br>0 | **Count of Clock Fails**<br>This counter increments upon hardware failure but can also be advanced during the removal of power. |
| **CM** | 16779 | UInt | R | RAM<br>Flash<br>255 | **Controller Manufacturer Code**<br>This property identifies the Manufacturer of the device. For American Auto-Matrix products this code is 255. This property is read-only. |
| **CP** | 16781 | UInt | RW | EE<br>0 | **Network Baud Rate**<br>0=9600<br>6=38.4K<br>7=19.2K<br>9=57.6K<br>10=76.8k |
| **CT** | 16784 | UInt | R | RAM<br>Flash<br>201 | **Controller Type**<br>This property identified the type of device relative to firmware. |
| **DE** | 16795 | UInt | RW | RAM<br>0 | **Default Enable Command**<br>This property is used to restore configuration settings to factory defaults. To default the settings enter a value of 197 (a value which is unlikely to occur randomly). It may take several seconds to complete the reset. Note that this will not alter the unit ID or selected communications baud rate, but will erase all application configurations you may have performed. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **EM** | 16813 | Bool | RW | EE<br>0 | **English/Metric**<br>This property specifies the type of engineering units to be used for temperatures. If EM is set to zero, degrees are specified in Fahrenheit, if EM is set to one, degrees are specified in Celsius. Note that a change in this property automatically converts setpoints to the appropriate units. The display mode for digital thermostats is also changed but can be also be set separately.<br><br>0 = English<br>1 = Metric Units<br>0 = English<br>1 = Metric Units |
| **F1** | 16820 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 1 will not trip the fan.<br>If 1 and Interlock 1 is active, the fan is shut down. |
| **F2** | 16821 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 2 will not trip the fan.<br>If 1 and Interlock 2 is active, the fan is shut down. |
| **F3** | 16822 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 3 will not trip the fan.<br>If 1 and Interlock 3 is active, the fan is shut down. |
| **FT** | 16834 | UInt | R | RAM<br>Flash<br>4 | **Firmware Type**<br>Defines the class of firmware operating system used in this controller. Only flash updates of matching firmware type will be accepted. Upgrades and conversions to other classes of firmware will require special handling. Contact the factory for further information. This property is read-only. |
| **I1** | 16868 | UInt | RW | EE<br>0 | **Interlock 1 Input Channel**<br>0=Disabled, 1=UI1, 2=UI2, 3=UI3, 4=UI4, 5=UI5, 6=OIA/B |
| **I2** | 16869 | UInt | RW | EE<br>0 | **Interlock 2 Input Channel**<br>0=Disabled, 1=UI1, 2=UI2, 3=UI3, 4=UI4, 5=UI5, 6=OIA/B |
| **I3** | 16870 | UIntBool | RW | EE<br>0 | **Fan Failure Interlock**<br>0=Disabled<br>1=Enable |
| **IC** | 16876 | UInt | R | EE<br>0 | **EEPROM Default Count**<br>This counter increments whenever the EEPROM is restored to factory default settings (see Device;DE Default Enable). |
| **ID** | 16877 | UInt | RW | EE<br>Factory Set | **Unit ID**<br>assignable between 0 and 127. |
| **IS** | 16882 | bitstring | R | RAM<br>N/A | **Interlock Status**<br>0=Interlock 1, bit 1=Interlock 2, bit 2=Interlock 3 |
| **MS** | 16902 | Bool | RW | EE<br>0 | **Master/Slave Mode**<br>This defines the MS/TP mode of the device. When configured as Master, the device will be capable of being auto-discovered as well as pass network tokens and solicit requests to the network (such as time synchronizations). When configured as Slave,<br>the device must be manually addressed and cannot solicit requests to the network. |
| **OC** | 16917 | UInt | RW | EE<br>0 | **Count of Illegal Opcodes**<br>This counter increments upon firmware failure but can also be advanced during the removal of power. |
| **OS** | 16925 | Real | R | N/A | **Kernel Version**<br>Specifies the version number of the currently active Kernel Boot Block. This property is read-only. |
| **PD** | 16942 | UInt | RW | EE<br>5 | **Power-on Delay**<br>This property determines how long, in seconds, the device waits before energizing its outputs after a power loss or soft reset. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **PS** | 16951 | UInt | RW | EE<br>2 | **Power-up State**<br>This property determines which schedule state the device uses after a power loss and before its time is synchronized by a host. The selections are:<br><br>0=unoccupied<br>1=warmup<br>2=occupied<br>3=night setback |
| **RC** | 16963 | UInt | RW | EE<br>0 | **Count of Resets**<br>This counter increment each time power is applied to the controller. This counts power outages and noise related resets as well as resets initiated through Device;RS. |
| **RE** | 16964 | UInt | RO | RAM | **STATbus Reset Count**<br>This read only counter increments each time a diagnostic STATbus reset occurs and is used for diagnostic and troubleshooting purposes. |
| **RI** | 16967 | Bool | RW | EE<br>0 | **Reset Fan Failure Interlock**<br>This property resets Fan Failure Interlock. When Fan Failure Interlock is enabled to shut down the fan (Device;F3=1), and Fan Failure Interlock is active, setting Reset Fan Failure Interlock (Device;RI=1) allows the fan to restart. |
| **RS** | 16972 | Bool | RW | RAM<br>0 | **Reset**<br>0 = disabled (default),<br>1 = reset controller |
| **SN** | 16991 | UInt | R | EE<br>factory set | **Serial Number**<br>Displays the Serial Number of the controller. This property is read-only. |
| **SR** | 16994 | UInt | R | RAM<br>Flash | **Software Time Stamp**<br>This uniquely defines each flash firmware image. |
| **UP** | 17030 | UInt | R | EE<br>0 | **Flash Update Count**<br>This counter increment each time a new flash firmware image is accepted by the controller. |
| **VE** | 17043 | Real | R | RAM<br>Flash | **Software Version**<br>Specifies the version number of the currently active firmware. This property is read-only. |
| **WC** | 17050 | UInt | RW | EE<br>0 | **Count of Watchdog COP**<br>This counter increments upon firmware failure but can also be advanced during the removal of power. |
| **ZN** | 17084 | UInt | RW | EE<br>0 | **Zone Number**<br>Defines the zone number that this device is assigned to for receiving network broadcasts from other devices (e.g. schedule broadcast, OAT broadcast, etc) |
| **ZP** | 17085 | UInt | R | RAM<br>0 | **Count of High Current Pulses**<br>This counter increment when high pulses in the pulse input are received. |

# B.2 ZONE TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Analog Input (0), Instance *N* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>Universal Input N | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Analog Input (0) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM<br>0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM<br>0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM<br>0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM<br>95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **BM** | 16754 | UInt | RW | EE<br>0 | **SSB Bus Mode**<br>by default this should be set to Master (**BM**=0) unless multiple controllers are wired onto a single Sensor Bus (SSB). All additional controllers on the SSB must be configured as Slaves (**BM**=1). |
| **BT** | 16757 | UInt | RW | EE<br>0 | **Application (Box Type)**<br>none (**BT**=0), cooling only (**BT**=1), heating only (**BT**=2), supply dependant (**BT**=3), and cooling w/reheat (**BT**=4). **BT** defaults to cooling only (**BT**=1). |
| **CC** | 16770 | Real | R | RAM | **Current Cooling Setpoint**<br>shows the current cooling temperature control setpoint. This will depend on setbacks and user adjustments. The attribute is read-only. |
| **CH** | 16775 | Real | R | RAM | **Current Heating Setpoint**<br>shows the current heating temperature control setpoint. this will depend on setbacks and user adjustments. The point is read-only. |

| | | | | | |
|---|---|---|---|---|---|
| **DF** | 16796 | UInt | RW | EE<br>0 | **Thermostat Display Format**<br>defines the format used to display the current temperature on the digital thermostat. The display of the tenths digit and the Fahrenheit/Celsius character are options. Also, the display may be eliminated.<br>0=##° (Default)<br>1=##.#°<br>2=##°F (or C)<br>3=##.#°F (or C)<br>4=No Temp Display |
| **DL** | 16798 | Real | R | RAM | **Total Zone Demand Load**<br>indicates the heating/cooling demand for the zone in terms of temperature separation from setpoints. |
| **DM** | 16799 | UInt | R | RAM | **Demand Mode**<br>indicates the demand for the zone. A satisfied zone will indicate “vent” (**DM**=0). If the *NB*-ASCe is in cooling mode and the zone temperature exceeds the cooling setpoint, “cool” is indicated (**DM**=1). If the controller is in heating mode and the zone temperature falls below the heating setpoint, “heat” is indicated (**DM**=2). |
| **DS** | 16803 | UInt | RW | EE<br>0 | **Thermostat Display Mode**<br>specifies whether English or Metric units are to be used for digital thermostat display on the *SBC*-STAT3. This mode is automatically altered as appropriate when the system Engineering Units property is set but may be modified later if required to display the alternate units.<br>0=Fahrenheit (Default)<br>1=Celsius |
| **DV** | 16805 | UInt | RW | EE<br>0 | **Thermostat Display Value**<br>by default (**DV**=0) each digital thermostat will display the identical temperature value (**ZT**) which is the average of each. With DV=1, each thermostat will display its own temperature (including offset)**.** |
| **ED** | 16808 | UInt | RW | EE<br>60 | **Extended Occupancy Time Duration**<br>specifies the amount of time in minutes to extend occupancy. **ED** has a default value of 60. |
| **ER** | 16816 | UInt | RW | RAM | **Extended Occupancy Time Remaining**<br>shows the amount of time remaining in extended occupancy. This value is set to the Extended Occupancy Duration (**ET**) when either push button on an analog thermostat is pressed. The *SBC*-STAT3 digital thermostat employs its User Menu for this function. **ER** is a read-only property that cannot be changed directly. |
| **G0** | 16837 | UInt | R | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G1** | 16838 | UInt | R | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G2** | 16839 | UInt | R | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G3** | 16840 | UInt | R | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **OF** | 16919 | Real | RW | EE<br>0 | **Temperature Adjustment**<br>defines an optional correction that may be required as an adjustment for the thermostat location and the possible measurement errors. |

| | | | | | |
|---|---|---|---|---|---|
| **PB** | 16940 | UInt | RW | EE<br>2200 | **Balance PIN**<br>this Personal Identification Number controls access to the Balance Menu. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access to the menu. A matching number must be entered by the Balancer. Values of 10,000 or greater will hide the menu. Entered P.I.N. numbers remain valid for only four (4) minutes after their use**. PB** has a default value of 2200. |
| **PG** | 16945 | UInt | RW | EE<br>0 | **Primary GID**<br>specifies the GID of the Primary thermostat in Primary GID mode (**RM**=8). If this thermostat is not available, then the Average temperature mode (**RM**=0) is used. |
| **PI** | 16947 | UInt | RW | EE<br>3300 | **Installer PIN**<br>this Personal Identification Number controls access to all menus. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access. A matching number must be entered by the Installer. Values of 10,000 or greater will hide the Install Menu. An authenticated Installer can access all menus. Entered P.I.N.s remain valid for only four minutes after the last button press. **PI** has a default value of 3300. |
| **PO** | 16949 | UInt | RO | RAM | **Present Occupancy Status**<br>shows the current occupancy status for the schedule and reflects any schedule overrides made.<br>0=Unoccupied<br>1=Warmup<br>2=Occupied<br>3=Night Setback |
| **PS** | 16951 | UInt | RW | EE<br>1100 | **Service PIN**<br>this Personal Identification Number controls access to the Service Menu. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access to the menu. A matching number must be entered by the Servicer. Values of 10,000 or greater will hide the menu. Entered P.I.N.s remain valid for only four (4) minutes after their use. **PS** has a default value of 1100. |
| **PU** | 16952 | UInt | RW | EE<br>0000 | **User PIN**<br>this Personal Identification Number controls access to the User Menu. A value of 0 makes the menu always accessible. Values from 0001 to 9999 inclusive are used to control access to the menu. A matching number must be entered by the User. Values of 10,000 or greater will hide the menu. Entered P.I.N.s remain valid for only four (4) minutes after their use. **PU** has a default value of 0. |
| **RM** | 16969 | UInt | RW | EE<br>0 | **Reading Mode**<br>indicates the current reading mode. This would be either Cooling or Heating as specifies by the system box type (**BT**). If **BT** is set to supply dependant, the point will indicate the current mode as determined by the source/duct temperature.<br>0=Average<br>1=Highest<br>2=Lowest<br>3=Hi/Lo VST mode<br>4=Device 0<br>5=Device 1<br>6=Device 2<br>7=Device 3<br>8=Primary GID |

| | | | | | |
|---|---|---|---|---|---|
| **SD** | 16983 | UInt | RW | EE<br>0 | **Calculated Setpoint Display**<br>specifies what method is used to display setpoint adjustments on an SBC-STAT3 LCD screen.<br>0 = Disable (+/-2.5)<br>1 = Zone Midpoint (Zone Temperature: (ZS) Zone Midpoint)<br>2 = Heating Setpoint (Zone Temperature: (CH) Heating Setpoint)<br>3 - Cooling Setpoint (Zone Temperature: (CC) Cooling Setpoint) |
| **SE** | 16984 | Boolean | RW | EE<br>1 | **Override Disabled/Enabled**<br>enables or disables the user's ability to enter extended occupancy override.<br>0=Disabled<br>1=Enabled (Default) |
| **T0** | 17002 | Real | R | RAM | **Thermostat Reading for G0**<br>indicates the current temperature sensed by G0. |
| **T1** | 17003 | Real | R | RAM | **Thermostat Reading for G1**<br>indicates the current temperature sensed by G1. |
| **T2** | 17004 | Real | R | RAM | **Thermostat Reading for G2**<br>indicates the current temperature sensed by G2. |
| **T3** | 17005 | Real | R | RAM | **Thermostat Reading for G3**<br>indicates the current temperature sensed by G3. |
| **TM** | 17011 | Real | RW | EE<br>0.5 | **Offset Increment**<br>specifies the magnitude of incremental changes to the User Setpoint Offset (TS). The User Adjust Position (TP) is multiplied by TM to determine the User Setpoint Offset (TS) value. If the User Adjust Increment is 0, you will not be able to alter the setpoint. |
| **TP** | 17013 | UInt | RW | RAM<br>0 | **User Adjust Position**<br>the User Setpoint Offset (TS) can be raised or lowered in integral steps. This property tracks the current step. It can be set to any signed integer but will be constrained to +/-2 when adjusted by an analog thermostat or to +/-5 when set through a digital thermostat. The point is used in combination with the User Adjust Increment (TM) to calculate the User Setpoint Offset. |
| **TR** | 17014 | UInt | | RAM<br>0 | **User Adjust Remaining**<br>displays the time remaining before the User Setpoint Offset (TS) setting is reset. |
| **TS** | 17015 | Real | RW | RAM<br>0 | **Setpoint Offset**<br>defines an offset for application to PID setpoints. This point shows the current value calculated when you multiply the User Adjust Position (TM) by the User Adjust Increment (TP). This setting is temporary and is valid only for TT minutes unless TT=0. |
| **TT** | 17016 | UInt | RW | EE<br>120 | **User Adjust Duration**<br>the User Setpoint Offset (TS) is a temporary setting. The TT property defines in minutes the duration for which the setting applies. After that time, the User Adjust Position and User Adjust Offset are reset to 0 degrees. If the User Adjust Duration is 0, then setpoint changes remain in effect until modified. The default value for TT is 120. |
| **ZS** | 17087 | Real | RW | RAM<br>70 | **Heating/Cooling Setpoint**<br>displays the midpoint between the current cooling and heating setpoints. This property reflects changes in both setpoints. A change in ZS results in the appropriate shift of both the cooling and heating setpoint maintaining the effective deadband. |

# B.3 UNIVERSAL INPUTS 1-5

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (3), Instance *N* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Universal Input *N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **IF** | 16878 | UInt | RW | EE 0.0 | **Input Filtering**<br>specifies the amount of time in tenths of seconds during which an input configured as a digital input must remain stable for the value to be considered reliable. This is also the weighted gain if the input is configured as analog. |
| **IP** | 16881 | Bool | RW | EE 0 | **Input Polarity**<br>0=normal<br>1=reverse |
| **OF** | 16919 | Real | RW | EE 0 | **UI Offset**<br>specifies an offset value to be added to the Universal Input's present-value. |
| **ST** | 16996 | UInt | RW | EE 7 | **Sensor Type**<br>0= digital<br>2= full scale, linear from min_pres_value to max_pres_value<br>3= 4–20mA linear scaled from min_pres_value to max_pres_value<br>7= −22.0 to 122.0°F thermistor<br>1,4,5, and 6 are unused |

# B.4 SUPPLY TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Analog Input (3), Instance *8* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>Supply Temperature | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Analog Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM<br>0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM<br>0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM<br>0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM<br>95 | indicates the measurement units of this object. |
| **DD** | 16794 | Real | RW | EE<br>2.5°F | **Auto Duct Delta Temperature**<br>This defines the temperature difference by which the supply air must either exceed the heating setpoint to switch to 'heating mode', or fall below the cooling setpoint to engage 'cooling' mode. |
| **IC** | 16875 | UInt | RW | EE<br>0 | **Input Channel Select**<br>1=UI01<br>2=UI02<br>3=UI03<br>4=UI04<br>5=UI05 |
| **OF** | 16919 | Real | RW | EE<br>0 | **Supply/Duct Temperature Adjustment**<br>Defines an offset used to adjust the current present-value. |
| **SM** | 16990 | UInt | R | RAM<br>NA | **Cooling/Heating Supply Mode**<br>0=Cooling<br>1=Heating |

# B.5 OUTSIDE AIR TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (3), Instance *9* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Outside Air Temperature | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **IC** | 16875 | UInt | RW | EE 0 | **Input Channel Select**<br>1=UI01<br>2=UI02<br>3=UI03<br>4=UI04<br>5=UI05 |
| **OF** | 16919 | Real | RW | EE 0 | **Outside Air Temperature Adjustment**<br>Defines an offset used to adjust the current present-value. |

# B.6 RELATIVE HUMIDITY

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (0), Instance *10* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Relative Humidity | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (0) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **DS** | 16803 | UInt | RW | EE 0 | **Display Relative Humidity?**<br>0 = No<br>1 = Yes |
| **OF** | 16919 | Real | RW | EE 0 | **Relative Humidity Adjustment**<br>Defines an offset used to adjust the current present-value. |

# B.7 ANALOG OUTPUTS 1-4

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Output (1), Instance 1-12 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Analog Output *N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Output (1) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM 0.0 | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM 100.0 | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **relinquish_ default** | 104 | Real | RW | NRAM 0.0 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **DT** | 16804 | UInt | RW | EE 252 | **Data type** This property specifies the precision of present-value. |
| **HS** | 16863 | Real | RW | EE 100.0 | **Maximum Scaled Voltage** This property specifies the actual analog output value for a present-value of max-pres-value. |
| **LS** | 16894 | Real | RW | EE 0.00 | **Minimum Scaled Voltage** This property specifies the actual analog output value for a present-value of min-pres-value. |

# B.8 ANALOG VALUES (SETPOINTS) 1-7

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Analog Value (2), Instance 1-7 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>*N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Analog Value (2) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM<br>0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM<br>0 | four flags that indicate the general “health” of the object. |
| **event_state** | 36 | BACnet Event State | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. Setting this value to True will allow for the present-value to become writable. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM<br>95 | indicates the measurement units of this object. |

# B.9 PULSE INPUT

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Binary Input (3), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>*Pulse Input* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Binary Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM<br>0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM<br>0 | four flags that indicate the general “health” of the object. |
| **event_state** | 36 | BACnet Event State | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. Setting this value to True will allow for the present-value to become writable. |
| **polarity** | 84 | Enum | RW | NRAM<br>0 | indicates the polarity for the input. |
| **MD** | 16896 | UInt | RW | EE<br>0 | **Pulse Counter Mode**<br>0=Off<br>1=Falling edge<br>2=Rising edge<br>3=Both edges |
| **NP** | 16910 | UInt | RW | RAM<br>0 | **Pulse Count Total**<br>This property indicates the total amount of pulses accumulated by the device. |
| **SF** | 16985 | Real | RW | EE<br>0.00 | **Pulse Multiplier**<br>This property specifies the multiplier to apply against (NP) Pulse Count Total to generate the value in (SV) Scaled Pulse Count. |
| **SV** | 16998 | Real | R | RAM<br>0.00 | **Scaled Pulse Value**<br>This property indicates the scaled pulse value, which is (NP) x (SF). |

# B.10 BO01

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO01 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **FD** | 16825 | UInt | RW | EE 30 | **Shutoff Delay (seconds)** This property shows the amount of time, in seconds, the fan output will stay energized once the zone temperature reaches the deadband. |
| **FN** | 16829 | UInt | RW | EE 0 | **Night Setback Fan Mode** 1 = fan runs for the entire period 0 = fan shuts off when zone temp is within the deadband. |
| **FO** | 16830 | UInt | RW | EE 0 | **Occupied Fan Mode** 1 = fan runs for the entire period 0 = fan shuts off when zone temp is within the deadband. |
| **FR** | 16832 | Real | RW | EE 0.5 | **Minimum Run Time** This property shows the minimum amount of time, in minutes, the fan output will stay energized. This prevents short cycling of the fan output. |
| **FS** | 16833 | Real | RW | EE 1.0 | **Minimum Off Time** This property shows the minimum amount of time, in minutes, the fan output will stay de-energized. This prevents short cycling of the fan output. |
| **FU** | 16835 | UInt | RW | EE 0 | **Unoccupied Fan Mode** 1 = fan runs for the entire period 0 = fan shuts off when zone temp is within the deadband. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **DB** | 16792 | Real | RW | EE<br>0.0 | **Fan Auto Mode Deadband**<br>used exclusively in conjunction with fan modes (FO, FU, FN) that are configured for Auto modes. When the device is in a schedule state where the fan operates in auto mode, the current zone temperature must exceed setpoint plus or minus the deadband temperature value in order for the fan to activate, followed by stages of heating or cooling. This property is useful for preventing the fan from turning on in situations where the zone temperature could possibly drift closely near mode changes. |
| **MT** | 16903 | UInt | RO | RAM<br>- | **Min Run/Off Countdown Timer**<br>reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours**<br>This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE<br>0.0 | **Run Limit**<br>This property specifies a run time limit in hours for the output. |

# B.11 BO02

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 2 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO02 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours** This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE 0.0 | **Run Limit** This property specifies a run time limit in hours for the output. |
| **SD** | 16983 | Unsigned | RW | EE 180 | **Settling Delay** This property specifies a delay period in seconds that is imposed both before and after the reversing valve state is changed. The delay begins once all stages have been shut down. Stages will not energize for this period after the valve state is changed. |
| **FB** | 17094 | Unsigned | RO | EE 0 | **Defrost Status** Indicates if the defrost logic is active, and if a defrost cycle is in progress. |
| **IN** | 16880 | Unsigned | RW | EE 0 | **Defrost Input Select** This property selects what input to use to control the defrost cycle. If set to “Unused”, no active defrost control capabilities will occur. |
| **ES** | 16817 | Real | RW | EE 0 | **Enter Defrost Setpoint** This property defines the temperature that the coils must be cooled to for the defrost cycle to engage. |
| **XS** | 17095 | Real | RW | EE 0 | **Exit Defrost Setpoint** This property defines the temperature at which point the defrost cycle will terminate. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **PT** | 17096 | Real | RW | EE<br>0 | **Programmed Time Between Defrost Cycles**<br>This property defines the initial time period (in minutes) from the end of one cycle to the start of the next. The actual time may be adjusted automatically based on the length of time that an initial defrost cycle may have taken. |
| **MC** | 17097 | Real | RW | EE<br>0 | **Maximum Defrost Cycle Time**<br>This property defines the maximum time (in minutes) that a defrost cycle will last. |
| **AT** | 16748 | Real | RO | EE<br>0 | **Adjusted Time Between Defrost Cycles**<br>This property indicates an adjustment made to the PT based on previous cycles. If the previous cycle does not raise the temperature up to the value defines in XS, then the controller will reduce the time before the next cycle. If the previous cycle completed quickly (faster than 25% of MC) then the time is increased. |
| **LT** | 17098 | Real | RO | EE<br>0 | **Defrost Action Time Remaining**<br>This property indicates how many minutes are left before the defrost logic will look to perform the next action. |

# B.12 BO03

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 3 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO03 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **CL** | 16778 | Real | RW | 55.0 EE | **Cooling OAT Lockout** Defines the Outside Air Temperature setpoint below which cooling is locked out. Stages will not be de-energized should the OAT fall below this temperature during an active cycle. |
| **CB** | 16769 | Real | RW | EE 0.0 | **Cooling Stage Deadband** This stage will not be disengaged until the Zone Temperature falls below the cooling setpoint by the number of degrees specified by this property. |
| **CO** | 16780 | Real | RW | 0.0 EE | **Cooling Stage Temperature Offset** This property indicates the temperature offset from setpoint required before engaging the stage for Cooling. Note that the stage may also engage if the Staging Delay for the prior stage expires. |
| **CR** | 16782 | Real | RW | 3.0 EE | **Cooling Minimum Run Time** This property shows the minimum amount of time, in minutes, the stage will stay energized when cooling. |
| **CS** | 16783 | Real | RW | 7.0 EE | **Cooling Minimum Off Time** This property shows the minimum amount of time, in minutes, the stage will remain de-energized when cooling. |
| **CX** | 16786 | Real | RW | 20.0 EE | **Cooling Staging Delay** This property indicates the maximum amount of time, in minutes, that the cooling stage will operate before energizing the next stage of cooling. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **HB** | 17092 | Real | RW | EE<br>0.0 | **Heating Stage Deadband**<br>This stage will not be disengaged until the Zone Temperature rises above the heating setpoint by the number of degrees specified by this property. |
| **HL** | 16858 | Real | RW | 80.0 | **Heating OAT Lockout**<br>Defines the Supply Temperature setpoint above which heating is locked out. Stages will not be de-energized should the OAT rise above this temperature during an active cycle. |
| **HO** | 16860 | Real | RW | 0.0<br>EE | **Heating Stage Temperature Offset**<br>This property indicates the temperature offset from setpoint required before engaging the stage for Heating. Note that the stage may also engage if the Staging Delay for the prior stage expires. |
| **HR** | 16862 | Real | RW | 3.0<br>EE | **Heating Minimum Run Time**<br>This property shows the minimum amount of time, in minutes, the stage will stay energized when heating. |
| **HS** | 16863 | Real | RW | 7.0<br>EE | **Heating Minimum Off Time**<br>This property shows the minimum amount of time, in minutes, the stage will remain de-energized when heating. |
| **HX** | 16866 | Real | RW | 20.0<br>EE | **Heating Staging Delay**<br>This property indicates the maximum amount of time, in minutes, that the heating stage will operate before energizing the next stage of heating. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours**<br>This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE<br>0.0 | **Run Limit**<br>This property specifies a run time limit in hours for the output. |
| **TH** | 17009 | Real | RW | 105.0<br>EE | **High Temperature Lockout**<br>Defines the Supply Temperature setpoint above which heating is locked out. |
| **TL** | 17010 | Real | RW | 45.0<br>EE | **Low Temperature Lockout**<br>Defines the Supply Temperature setpoint below which cooling is locked out. This offers protection against freeze up. |

# B.13 BO04

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Binary Output (4), Instance 4 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>BO04 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | -<br>Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM<br>0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM<br>0 | four flags that indicate the general "health" of the program. |
| **event_state** | 36 | Enum | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM<br>0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM<br>7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM<br>NULL | contains prioritized commands that are in effect for this object. |
| **CB** | 16769 | Real | RW | EE<br>0.0 | **Cooling Stage Deadband**<br>This stage will not be disengaged until the Zone Temperature falls below the cooling setpoint by the number of degrees specified by this property. |
| **CO** | 16780 | Real | RW | 2.0<br>EE | **Cool Stage Temp Offset**<br>This property indicates the temperature offset from setpoint required before engaging the stage for Cooling. Note that the stage may also engage if the Staging Delay for the prior stage expires. |
| **CR** | 16782 | Real | RW | 3.0<br>EE | **Cool Minimum Run Time**<br>This property shows the minimum amount of time, in minutes, the stage will stay energized when cooling. |
| **CS** | 16783 | Real | RW | 7.0<br>EE | **Cool Minimum Off Time**<br>This property shows the minimum amount of time, in minutes, the stage will remain de-energized when cooling. |
| **HB** | 17092 | Real | RW | EE<br>0.0 | **Heating Stage Deadband**<br>This stage will not be disengaged until the Zone Temperature rises above the heating setpoint by the number of degrees specified by this property. |
| **HO** | 16860 | Real | RW | 2.0<br>EE | **Heat Stage Temp Offset**<br>This property indicates the temperature offset from setpoint required before engaging the stage for Heating. Note that the stage may also engage if the Staging Delay for the prior stage expires. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **HR** | 16862 | Real | RW | 3.0<br>EE | **Heat Minimum Run Time**<br>This property shows the minimum amount of time, in minutes, the stage will stay energized when heating. |
| **HS** | 16863 | Real | RW | 7.0<br>EE | **Heat Minimum Off Time**<br>This property shows the minimum amount of time, in minutes, the stage will remain de-energized when heating. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours**<br>This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE<br>0.0 | **Run Limit**<br>This property specifies a run time limit in hours for the output. |

# B.14 BO05

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 5 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO05 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general "health" of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours** This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE 0.0 | **Run Limit** This property specifies a run time limit in hours for the output. |

# B.15 SCHEDULE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Schedule (17), Instance *1* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Schedule | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Schedule (17) | indicates membership in a particular object type class. |
| **present-value** | 85 | Unsigned or Enum | RO | RAM 0 | indicates the current schedule value |
| **effective-period** | 32 | Date Range | RO | RAM 0 | specifies the date rage in which the schedule is active. |
| **schedule-default** | 174 | Unsigned or Enum | RW | RAM 0 | specifies the datatype and value used for partial time-of-day scheduling. |
| **weekly-schedule** | 123 | Array List | RW | RAM 0 | contains time,value pair operations for each day of the week. |
| **list-of-object-property-references** | 54 | List | RO | NRAM 0 | contains the list of object properties that the schedule writes to. |
| **status-flags** | 111 | Bitstring | RO | NRAM 0 | indicates the general health of the object. |
| **reliability** | 103 | Enum | RO | NRAM 0 | indicates the reliability of the object. |
| **out-of-service** | 81 | Boolean | RO | NRAM 0 | indicates the service status of the object. |
| **priority-for-writing** | 88 | Unsigned | RW | NRAM 0 | indicates the priority which the Schedule will write with. |
| **HE** | 16853 | Bool | RW | EE 0 | H**ost Overrides**<br>0=disabled<br>1=enabled |
| **HO** | 16860 | UInt | RW | RAM 0 | **Host Schedule**<br>0 = unoccupied<br>1 = warm-up<br>2 = occupied<br>3 = night setback |
| **IS** | 16882 | UInt | RW | EE 3 | **Inactive Schedule State**<br>0 = unoccupied<br>1 = warm-up<br>2 = occupied<br>3 = night setback |
| **ZE** | 17081 | Bool | RW | EE 0 | **Receive Schedule**<br>0=No<br>1=Yes |

# B.16 ECONOMIZER

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (133), Instance *1* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Economizer | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **CM** | 16779 | Real | R | RAM N/A | **Calculated Minimum Position** This property displays the actual minimum position. |
| **ED** | 16808 | Real | RW | EE 1.0 | **Economizer Staging Delay (minutes)** This property specifies how many minutes the controller waits before using additional cooling stages after the economizer damper reaches 100. |
| **EE** | 16809 | UInt | RW | EE | **Economizer Enable** 0=Off<br>1=PID 1<br>2=PID 2<br>3=PID 3<br>4=PID 4<br>5=DO 3<br>6=DO 5 |
| **EM** | 16813 | Real | RW | EE 10.0 | **Economizer Minimum Position (%)** This property specifies the PID minimum position in percent for the economizer damper. |
| **ML** | 16090 | Real | RW | EE 0.0 | **Reset Limit for Economizer Minimum Protection** This property specifies the value at which maximum reset is used. When the value of the reset variable is equal to ML, the (MR) Maximum Reset is used in determining the calculated minimum position. |
| **MP** | 16900 | Real | RW | EE 0.0 | **Reset Setpoint for Economizer Minimum Position** This property specifies the value at which the reset action begins. When the value of the reset variable exceeds MP, reset action will be used in determining the economizer minimum position. |
| **MR** | 16901 | Real | RW | EE 0.0 | **Maximum Reset for Economizer Minimum Position** This property specifies the maximum amount to reset the minimum position setpoint (EM) by when reset is being used. |
| **MV** | 16904 | UInt | RW | EE 0.0 | **Reset Variable for Economizer Minimum Position** 0=Disabled<br>1=Zone Temperature<br>2=Supply Temperature<br>4=UI01<br>5=UI02<br>6=UI03<br>7=UI04<br>8=UI05<br>11=OAT |
| **OH** | 16920 | Real | RW | EE 60.0 | **OAT High Limit** This property specifies the Outside Air Temperature (OAT) high limit. If the OAT is above the high limit, the PID is set to the Economizer Minimum Position (EM). |
| **OL** | 16922 | Real | RW | EE 45.0 | **OAT Low Limit** This property specifies the Outside Air Temperature (OAT) low limit. If the OAT is below the low limit, the PID is set to the Economizer Minimum Position (EM). |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **CM** | 16779 | Real | R | RAM<br>N/A | **Calculated Minimum Position**<br>This property displays the actual minimum position. |
| **ED** | 16808 | Real | RW | EE<br>1.0 | **Economizer Staging Delay (minutes)**<br>This property specifies how many minutes the controller waits before using additional cooling stages after the economizer damper reaches 100. |

# B.17 AUX HEAT

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (133), Instance *3* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Aux Heat | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **RO** | 16970 | Unsigned | R | RAM 0 | **Stages Available**<br>This property specifies the number of stages of Aux Heat. For Single Stage operation, relay K5 is used for Aux Heat. For Dual Stage both K5 and K4 are used for Aux Heat, in which case Stage 2 Heat/Cool is not available for use.<br><br>1 = K4<br>2 = K4/K5 |
| **EN** | 16814 | Bitstring | RW | EE 1.0 | **Stages Energized**<br>This property shows which Aux Heat stages are presently engaged. |
| **FR** | 16832 | Bitstring | RW | EE | **Fan Required**<br>Defines by the appropriate checkbox which of the available Aux Heat stages requires a positive flow indication from the Proof of Flow object. Not all sources of auxiliary heat (perimeter heating for instance) will require verification of air flow. By default, all stages require a positive flow indication.<br><br>0=Off<br>1=PID 1<br>2=PID 2<br>3=PID 3<br>4=PID 4<br>5=DO 3<br>6=DO 5 |
| **BA** | 16751 | Boolean | | | **Use Balancing?**<br>If set to “Enabled” this considers the Run Hour Totals for each Binary Output for the individual relay outputs in energizing Aux Heat stages. Stages with lower usage will be energized first. The result is the balanced use of these stages. For best operation, all stages should be nearly equivalent. The default setting does not perform balancing. |
| **ID** | 16877 | Real | RW | EE 10.0 | **Staging Delay**<br>This property defines the time, in minutes, before a stage of Aux Heat is energized once compressor stages of heating have been energized. The timer starts when heating stages are active and the supply temp is lower than the defined setpoint. Once stage 1 of Aux Heat engages, the timer is restarted for the possible second stage. |
| **SS** | 16995 | Real | RW | EE 0.0 | **Aux Heat Setpoint**<br>This defines the supply duct temperature below which Aux Heat may be required. While in heat mode, once the temp. has stayed below this for the time specified in Interstage Delay the Aux Heat is energized. Once on, Aux Heat stages remain on until demand is met, regardless of supply duct temperature. |
| **EM** | 16813 | Real | RW | EE 0.0 | **Emergency Aux Heat Setpoint**<br>This defines a minimum supply duct temperature below which Aux Heat is available even when the outside temperature is warmer than the OAT lockout. The intent of this is to automatically allow for Aux Heat when the compressor is not functioning. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **HL** | 16858 | Real | RW | EE<br>0.0 | **OAT Lockout**<br>Aux Heat stages will be locked out should the Outside Air Temperature exceed the value specified in this property. |
| **TH** | 17009 | Real | RW | EE<br>0.0 | **Supply Temp High Lockout**<br>Aux Heat stages will be locked out should the Supply Temperature exceed the value specified in this property. |

# B.18 PID CONTROL 1 - 4

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (133), Instance 11-14 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>PID Control 1-4 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **AO** | 16746 | UInt | RW | RAM<br>NA | **Analog Output**<br>This property shows the scaled output value used by the analog output. AO is the PO value scaled to min-pres-value and max-pres-value of the corresponding analog output. |
| **CE** | 16772 | Bool | RW | EE<br>0 | **Control Enable**<br>0=No<br>1=Yes |
| **CS** | 16783 | Real | R | RAM<br>NA | **Control Setpoint**<br>This property shows the calculated, actual loop control setpoint. CS reflects the unoccupied setup/setback as well as any reset and/or thermostat setpoint adjustment. |
| **DB** | 16792 | Real | RW | EE<br>0 | **Deadband**<br>This property specifies the deadband around the setpoint for the control loop. When the value of the measured variable is within this deadband, the output signal remains constant at the midpoint of the throttling range (halfway between OH and OL). |
| **I1** | 16868 | Bool | RW | EE<br>0 | **Interlock 1 Enable**<br>0=Disabled<br>1=Enabled |
| **I2** | 16869 | Bool | RW | EE<br>0 | **Interlock 2 Enable**<br>0=Disabled<br>1=Enabled |
| **I3** | 16870 | Bool | RW | EE<br>0 | **No Flow Interlock Enable**<br>0=Disabled<br>1=Enable |
| **IC** | 16876 | UInt | RW | EE<br>0 | **Input Select**<br>0= disabled<br>1=Zone Temp<br>2=Supply Temp<br>4=UI1<br>5=UI2<br>6 =UI3<br>7 =UI4<br>8=UI5<br>9=Zone Heating<br>10=Zone Cooling<br>11=OAT<br>13=Relative Humidity |
| **IN** | 16880 | Real | R | RAM<br>0 | **Input Value**<br>This property displays the value of the input selected in IC. |
| **MR** | 16901 | Real | RW | EE<br>0 | **Maximum Reset**<br>This property specifies the maximum amount to reset the loop setpoint (SP) by when reset is being used. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **P1** | 16930 | Real | RW | EE<br>0.0 | **Interlock 1 Position**<br>This property specifies the PID output value when the Interlock 1 is enabled. |
| **P2** | 16931 | Real | RW | EE<br>0.0 | **Interlock 2 Position**<br>This property specifies the PID output value when the Interlock 2 is enabled. |
| **P3** | 16932 | Real | RW | EE<br>0.0 | **No Flow Interlock Position**<br>This property specifies the PID output value when the value of Fan Status is 0 (No Flow). |
| **PB** | 16940 | Real | RW | EE<br>0 | **Proportional Band**<br>This property specifies the input variable range over which the output value is proportional to the error value. The proportional band is centered around setpoint for the loop. |
| **PO** | 16949 | Real | RW | RAM<br>NA | **Percent Output**<br>This property shows the output value in hundredths of a percent. This value is calculated based on the error, change in error and past error for the control loop. PO is then scaled to the selected engineering units of the analog output and stuffed into the AO property, as well as the CV of the analog output. |
| **RC** | 16963 | Real | R | RAM<br>NA | **Reset Value**<br>This property displays the value of the input selected in RV. |
| **RL** | 16968 | Real | RW | EE<br>0 | **Reset Limit**<br>This property specifies the value at which maximum reset is used. When the value of the reset variable is equal to RL, the maximum reset (MR) is used in determining the calculated setpoint. |
| **RP** | 16971 | UInt | RW | EE<br>0 | **Reset Period**<br>This property specifies the reset period (in seconds) over which the error history is accumulated. If RP=10 seconds with a constant error of 2.0, then every second the error history would increase by 0.2. In five seconds, the error history would be 1.0 and at the end of ten seconds the error history would be 2.0. RP=0 disables integral action. |
| **RS** | 16972 | Real | RW | EE<br>0 | **Reset Setpoint**<br>This property specifies the value at which the reset action begins. When the value of the reset variable exceeds RS, reset action will be used in determining the calculated setpoint. |
| **RT** | 16973 | Real | RW | EE<br>0 | **Rate**<br>This property is the derivative rate and specifies a percentage of change in error that is to be used in calculating PO. The value is specified in percent-per-second. |
| **RV** | 16974 | UInt | RW | EE<br>0 | **Reset Variable**<br>0= disabled<br>1=Zone Temp<br>2=Supply Temp<br>4=UI1<br>5=UI2<br>6 =UI3<br>7 =UI4<br>8=UI5<br>11=OAT<br>13=Relative Humidity |
| **SG** | 16986 | UInt | RW | EE<br>0 | **Action**<br>0=normal (positive error causes an increase in output).<br>1=reverse (positive error causes a decrease in output) |
| **SP** | 16993 | Real | RW | EE<br>0 | **Loop Setpoint**<br>This property specifies the desired loop setpoint. This value is used with the unoccupied setup/setback and the reset to calculate the CS control setpoint. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **SU** | 16997 | Real | RW | EE<br>0 | **Setup/Setback**<br>This property specifies the amount to add (if SG=0) or subtract (if SG=1) from the setpoint during an unoccupied period. The adjusted setpoint will be displayed in CS. |

# B.19 OCCUPANCY DETECTION

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (131), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>Occupancy Detection | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (254) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **IC** | 16876 | UInt | RW | EE<br>0 | **Input Select**<br>0 = None<br>1 = UI1<br>2 = UI2<br>3 = UI3<br>4 = UI4<br>5 = UI5<br>6 = OIA/B |
| **MD** | 16896 | UInt | RW | EE<br>30 | **Extended Occupancy Delay**<br>This property sets the amount of time, in seconds, the occupancy detector must remain on before the occupancy detector will override the zone. This prevents false triggers that might occur as others pass quickly through the zone. |
| **MR** | 16901 | UInt | R | RAM<br>NA | **Extended Occupancy Remaining**<br>This property displays the time remaining for occupancy detector override. |
| **MS** | 16902 | UInt | R | RAM<br>NA | **Occupancy Status**<br>0=No Detection<br>1=Detection |
| **MT** | 16903 | UInt | RW | EE<br>0 | **Extended Occupancy Duration**<br>This property defines, in minutes, the length of time to override the zone whenever occupancy is detected. |

# B.20 PROOF OF FLOW

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (131), Instance 2 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Proof of Flow | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Proprietary (131) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **PF** | 16944 | UInt | R | RAM 0 | **Proof of Flow Indication** This property shows the status of the fan for Proof of Flow. |
| **IC** | 16876 | UInt | RW | EE | **Input Select** 0 =None 1 = UI1 2 = UI2 3 = UI3 4 = UI4 5 = UI5 6 = OIA/B |
| **PD** | 16942 | UInt | RW | EE 60 | **Proof of Flow Delay (seconds)** This property shows the amount of time, in seconds, imposed before enabling a positive flow indication. |

# B.21 OUTSIDE AIR TEMP. BROADCAST

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (143), Instance 0 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Outside Air Temp. Broadcast | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Proprietary (143) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **CV** | 16785 | Real | R | RAM 0.0 | **Current Value** shows the current value of the network broadcast schedule values received by the *NB*-ASC |
| **RB** | 16962 | Bool | RW | EE 0 | **Receive Broadcasts?** 0=No 1=Yes |
| **BE** | 16752 | Bool | RW | 0 | **OAT Broadcast Enable** 0=No 1=Yes |

# B.22 BROADCAST SCHEDULE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (143), Instance 5 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>Broadcast Schedule | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (143) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **CV** | 16785 | UInt | R | RAM<br>0 | **Current Value**<br>indicates the current schedule value being received. |
| **RB** | 16962 | Bool | RW | EE<br>0 | **Receive Broadcasts?**<br>0=No<br>1=Yes |

# Fancoil Objects & Properties

*The following tables contain listings of the BACnet objects and property assignments. Each property is listed with its identifier number, data type, access code, storage, default value (if any) and a brief description of its functionality.*

## In this Appendix


Device ........ C-3
Zone Temperature ........ C-7
Universal Inputs 1-5 ........ C-11
Supply Temperature ........ C-12
Outside Air Temperature ........ C-13
Analog Outputs 1-4 ........ C-14
Analog Values (Setpoints) 1-7 ........ C-15
Pulse Input ........ C-16
BO01 ........ C-17
BO02 ........ C-19
BO03 ........ C-20
BO04 ........ C-21
BO05 ........ C-22
Fan Speed ........ C-23
Schedule ........ C-24
Economizer ........ C-25
Reheat Ctrl ........ C-27
Valve Ctrl 1 and 2 ........ C-28
PID Control 1 - 4 ........ C-30
Occupancy Detection ........ C-32
Proof of Flow ........ C-33
Outside Air Temp. Broadcast ........ C-34
Broadcast Schedule ........ C-35

# C.1 DEVICE

**NOTE**

The Device object is represented in *NB*-Pro as follows:
**AAM Fancoil xxxxxxxxxx**
(where xxxxxxxxxx is the Unitary Controller serial number)

The instance must be a unique number from 0 to 4194302. By default, AAM sets the value in such a way that it is unique to AAM products based off the unit's serial number, however the user must ensure the device instance is unique on the job site's network.

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_ identifier** | 75 | BACnet ObjID | RO | EEPROM<br>Device (8), Instance *serial number* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>AAM NB-GPC *serial number* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Device (8) | indicates membership in a particular object type class. |
| **system_status** | 112 | BACnet ObjID | RO | -<br>0 | indicates the current physical and logical status of the BACnet Device. |
| **vendor_name** | 121 | CharStr | RO | NRAM<br>American Auto-Matrix | identifies the manufacturer of the BACnet Device. |
| **vendor_ identifier** | 120 | Unsigned | RO | -<br>6 | a unique vendor identification code, assigned by ASHRAE, which is used to distinguish proprietary extensions to the protocol. |
| **model_name** | 70 | CharStr | RO | NRAM<br>NB-GPC1 | indicates the vendor's name used to represent the model of the device. |
| **firmware_ revision** | 44 | CharStr | RO | NRAM<br>*revision number* | indicates the level of firmware installed in the device. |
| **application_ software_ version** | 12 | CharStr | RO | NRAM<br>*version number* | identifies the version of application software installed in the device. |
| **protocol_ version** | 98 | Unsigned | RO | -<br>1 | indicates the version of the BACnet protocol supported by this BACnet Device. |
| **protocol_ revision** | 139 | Unsigned | RO | -<br>2 | indicates the minor revision level of the BACnet standard. |
| **protocol_ services_ supported** | 97 | BACnet Services Supported | RO | - | indicates which standardized protocol services are supported by this device's protocol implementation. |
| **protocol_ object_ types_ supported** | 96 | BACnet Object Types Supported | RO | - | indicates which standardized object types are supported by this device's protocol implementation. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_list** | 76 | BACnet Array | RO | - | a list of each object within the device that is accessible through BACnet services. |
| **max_apdu_ length_ accepted** | 62 | Unsigned | RO | NRAM 480 | specifies the maximum number of information frames the node may send before it must pass the token. |
| **segmentation_ supported** | 107 | BACnet Segment ation | RO | - 3 | indicates whether the device supports segmentation of messages and, if so, whether it supports segmented transmission, reception, or both. |
| **local_time** | 57 | Time | RW | - | indicates the time of day to the best of the device's knowledge. |
| **local_date** | 56 | Date | RW | - | indicates the date to the best of the device's knowledge. |
| **apdu_timeout** | 11 | Unsigned | RW | NRAM 3000 | indicates the amount of time, in milliseconds, between retransmissions of an APDU requiring acknowledgment for which no acknowledgment has been received. |
| **number_of_ apdu_retries** | 73 | Unsigned | RW | NRAM 1 | indicates the maximum number of times that an APDU shall be retransmitted. |
| **time_ synchronization _recipients** | 116 | List of BACnet recipients | RW | NRAM {} | a list of one device to which the device may automatically send a Time Synchronization request. |
| **max_master** | 64 | Unsigned | RW | EEPROM 127 | specifies the highest possible address for master nodes and shall be less than or equal to 127. |
| **max_info_ frames** | 63 | Unsigned | RW | NRAM 10 | specifies the maximum number of information frames the node may send before it must pass the token. |
| **device_ address_ binding** | 30 | List | RW | - 30 | a list of the device addresses that will be used when the remote device must be accessed via a BACnet service request. |
| **BU** | 16758 | Bool | RW | RAM 0 | **Backup Control**<br>**BU** = 1 forces backup of **AE** and digital outputs 1-5 **RH** to EEPROM. |
| **CC** | 16770 | UInt | RW | EE 0 | **Count of Clock Fails**<br>This counter increments upon hardware failure but can also be advanced during the removal of power. |
| **CM** | 16779 | UInt | R | RAM Flash 255 | **Controller Manufacturer Code**<br>This property identifies the Manufacturer of the device. For American Auto-Matrix products this code is 255. This property is read-only. |
| **CP** | 16781 | UInt | RW | EE 0 | **Network Baud Rate**<br>0=9600<br>6=38.4K<br>7=19.2K<br>9=57.6K<br>10=76.8k |
| **CT** | 16784 | UInt | R | RAM Flash 201 | **Controller Type**<br>This property identified the type of device relative to firmware. |
| **DE** | 16795 | UInt | RW | RAM 0 | **Default Enable Command**<br>This property is used to restore configuration settings to factory defaults. To default the settings enter a value of 197 (a value which is unlikely to occur randomly). It may take several seconds to complete the reset. Note that this will not alter the unit ID or selected communications baud rate, but will erase all application configurations you may have performed. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **EM** | 16813 | Bool | RW | EE<br>0 | **English/Metric**<br>This property specifies the type of engineering units to be used for temperatures. If EM is set to zero, degrees are specified in Fahrenheit, if EM is set to one, degrees are specified in Celsius. Note that a change in this property automatically converts setpoints to the appropriate units. The display mode for digital thermostats is also changed but can be also be set separately.<br><br>0 = English<br>1 = Metric Units<br>0 = English<br>1 = Metric Units |
| **F1** | 16820 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 1 will not trip the fan.<br>If 1 and Interlock 1 is active, the fan is shut down. |
| **F2** | 16821 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 2 will not trip the fan.<br>If 1 and Interlock 2 is active, the fan is shut down. |
| **F3** | 16822 | Bool | RW | EE<br>0 | **Fan Failure Interlock Trips Fan?**<br>If 0, Interlock 3 will not trip the fan.<br>If 1 and Interlock 3 is active, the fan is shut down. |
| **FT** | 16834 | UInt | R | RAM<br>Flash<br>4 | **Firmware Type**<br>Defines the class of firmware operating system used in this controller. Only flash updates of matching firmware type will be accepted. Upgrades and conversions to other classes of firmware will require special handling. Contact the factory for further information. This property is read-only. |
| **I1** | 16868 | UInt | RW | EE<br>0 | **Interlock 1 Input Channel**<br>0=Disabled, 1=UI1, 2=UI2, 3=UI3, 4=UI4, 5=UI5, 6=OIA/B |
| **I2** | 16869 | UInt | RW | EE<br>0 | **Interlock 2 Input Channel**<br>0=Disabled, 1=UI1, 2=UI2, 3=UI3, 4=UI4, 5=UI5, 6=OIA/B |
| **I3** | 16870 | UIntBool | RW | EE<br>0 | **Fan Failure Interlock**<br>0=Disabled<br>1=Enable |
| **IC** | 16876 | UInt | R | EE<br>0 | **EEPROM Default Count**<br>This counter increments whenever the EEPROM is restored to factory default settings (see Device;DE Default Enable). |
| **ID** | 16877 | UInt | RW | EE<br>Factory Set | **Unit ID**<br>assignable between 0 and 127. |
| **IS** | 16882 | bitstring | R | RAM<br>N/A | **Interlock Status**<br>0=Interlock 1, bit 1=Interlock 2, bit 2=Interlock 3 |
| **MS** | 16902 | Bool | RW | EE<br>0 | **Master/Slave Mode**<br>This defines the MS/TP mode of the device. When configured as Master, the device will be capable of being auto-discovered as well as pass network tokens and solicit requests to the network (such as time synchronizations). When configured as Slave,<br>the device must be manually addressed and cannot solicit requests to the network. |
| **OC** | 16917 | UInt | RW | EE<br>0 | **Count of Illegal Opcodes**<br>This counter increments upon firmware failure but can also be advanced during the removal of power. |
| **OS** | 16925 | Real | R | N/A | **Kernel Version**<br>Specifies the version number of the currently active Kernel Boot Block. This property is read-only. |
| **PD** | 16942 | UInt | RW | EE<br>5 | **Power-on Delay**<br>This property determines how long, in seconds, the device waits before energizing its outputs after a power loss or soft reset. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **PS** | 16951 | UInt | RW | EE<br>2 | **Power-up State**<br>This property determines which schedule state the device uses after a power loss and before its time is synchronized by a host. The selections are:<br><br>0=unoccupied<br>1=warmup<br>2=occupied<br>3=night setback |
| **RC** | 16963 | UInt | RW | EE<br>0 | **Count of Resets**<br>This counter increment each time power is applied to the controller. This counts power outages and noise related resets as well as resets initiated through Device;RS. |
| **RE** | 16964 | UInt | RO | RAM | **STATbus Reset Count**<br>This read only counter increments each time a diagnostic STATbus reset occurs and is used for diagnostic and troubleshooting purposes. |
| **RI** | 16967 | Bool | RW | EE<br>0 | **Reset Fan Failure Interlock**<br>This property resets Fan Failure Interlock. When Fan Failure Interlock is enabled to shut down the fan (Device;F3=1), and Fan Failure Interlock is active, setting Reset Fan Failure Interlock (Device;RI=1) allows the fan to restart. |
| **RS** | 16972 | Bool | RW | RAM<br>0 | **Reset**<br>0 = disabled (default),<br>1 = reset controller |
| **SN** | 16991 | UInt | R | EE<br>factory set | **Serial Number**<br>Displays the Serial Number of the controller. This property is read-only. |
| **SR** | 16994 | UInt | R | RAM<br>Flash | **Software Time Stamp**<br>This uniquely defines each flash firmware image. |
| **UP** | 17030 | UInt | R | EE<br>0 | **Flash Update Count**<br>This counter increment each time a new flash firmware image is accepted by the controller. |
| **VE** | 17043 | Real | R | RAM<br>Flash | **Software Version**<br>Specifies the version number of the currently active firmware. This property is read-only. |
| **WC** | 17050 | UInt | RW | EE<br>0 | **Count of Watchdog COP**<br>This counter increments upon firmware failure but can also be advanced during the removal of power. |
| **ZN** | 17084 | UInt | RW | EE<br>0 | **Zone Number**<br>Defines the zone number that this device is assigned to for receiving network broadcasts from other devices (e.g. schedule broadcast, OAT broadcast, etc) |
| **ZP** | 17085 | UInt | R | RAM<br>0 | **Count of High Current Pulses**<br>This counter increment when high pulses in the pulse input are received. |

# C.2 ZONE TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (0), Instance *N* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Universal Input N | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (0) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **BM** | 16754 | UInt | RW | EE 0 | **SSB Bus Mode** by default this should be set to Master (**BM**=0) unless multiple controllers are wired onto a single Sensor Bus (SSB). All additional controllers on the SSB must be configured as Slaves (**BM**=1). |
| **BT** | 16757 | UInt | RW | EE 0 | **Application (Box Type)** none (**BT**=0), cooling only (**BT**=1), heating only (**BT**=2), supply dependant (**BT**=3), and cooling w/reheat (**BT**=4). **BT** defaults to cooling only (**BT**=1). |
| **CC** | 16770 | Real | RO | RAM | **Current Cooling Setpoint** shows the current cooling temperature control setpoint. This will depend on setbacks and user adjustments. The attribute is read-only. |
| **CH** | 16775 | Real | RO | RAM | **Current Heating Setpoint** shows the current heating temperature control setpoint. this will depend on setbacks and user adjustments. The point is read-only. |

| | | | | | |
|---|---|---|---|---|---|
| **DF** | 16796 | UInt | RW | EE<br>0 | **Thermostat Display Format**<br>defines the format used to display the current temperature on the digital thermostat. The display of the tenths digit and the Fahrenheit/Celsius character are options. Also, the display may be eliminated.<br>0=##° (Default)<br>1=##.#°<br>2=##°F (or C)<br>3=##.#°F (or C)<br>4=No Temp Display |
| **DL** | 16798 | Real | RO | RAM | **Total Zone Demand Load**<br>indicates the heating/cooling demand for the zone in terms of temperature separation from setpoints. |
| **DM** | 16799 | UInt | RO | RAM | **Demand Mode**<br>indicates the demand for the zone. A satisfied zone will indicate “vent” (**DM**=0). If the *NB*-ASCe is in cooling mode and the zone temperature exceeds the cooling setpoint, “cool” is indicated (**DM**=1). If the controller is in heating mode and the zone temperature falls below the heating setpoint, “heat” is indicated (**DM**=2). |
| **DS** | 16803 | UInt | RW | EE<br>0 | **Thermostat Display Mode**<br>specifies whether English or Metric units are to be used for digital thermostat display on the *SBC*-STAT3. This mode is automatically altered as appropriate when the system Engineering Units property is set but may be modified later if required to display the alternate units.<br>0=Fahrenheit (Default)<br>1=Celsius |
| **DV** | 16805 | UInt | RW | EE<br>0 | **Thermostat Display Value**<br>by default (**DV**=0) each digital thermostat will display the identical temperature value (**ZT**) which is the average of each. With DV=1, each thermostat will display its own temperature (including offset). |
| **ED** | 16808 | UInt | RW | EE<br>60 | **Extended Occupancy Time Duration**<br>specifies the amount of time in minutes to extend occupancy. **ED** has a default value of 60. |
| **ER** | 16816 | UInt | RW | RAM | **Extended Occupancy Time Remaining**<br>shows the amount of time remaining in extended occupancy. This value is set to the Extended Occupancy Duration (**ET**) when either push button on an analog thermostat is pressed. The *SBC*-STAT3 digital thermostat employs its User Menu for this function. **ER** is a read-only property that cannot be changed directly. |
| **G0** | 16837 | UInt | RO | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G1** | 16838 | UInt | RO | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G2** | 16839 | UInt | RO | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **G3** | 16840 | UInt | RO | RAM | **Global ID for Device**<br>the Global Identification for the Sensor Bus device. |
| **OF** | 16919 | Real | RW | EE<br>0 | **Temperature Adjustment**<br>defines an optional correction that may be required as an adjustment for the thermostat location and the possible measurement errors. |

| | | | | | |
|---|---|---|---|---|---|
| **PB** | 16940 | UInt | RW | EE<br>2200 | **Balance PIN**<br>this Personal Identification Number controls access to the Balance Menu. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access to the menu. A matching number must be entered by the Balancer. Values of 10,000 or greater will hide the menu. Entered P.I.N. numbers remain valid for only four (4) minutes after their use**. PB** has a default value of 2200. |
| **PG** | 16945 | UInt | RW | EE<br>0 | **Primary GID**<br>specifies the GID of the Primary thermostat in Primary GID mode (**RM**=8). If this thermostat is not available, then the Average temperature mode (**RM**=0) is used. |
| **PI** | 16947 | UInt | RW | EE<br>3300 | **Installer PIN**<br>this Personal Identification Number controls access to all menus. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access. A matching number must be entered by the Installer. Values of 10,000 or greater will hide the Install Menu. An authenticated Installer can access all menus. Entered P.I.N.s remain valid for only four minutes after the last button press. **PI** has a default value of 3300. |
| **PO** | 16949 | UInt | RO | RAM | **Present Occupancy Status**<br>shows the current occupancy status for the schedule and reflects any schedule overrides made.<br>0=Unoccupied<br>1=Warmup<br>2=Occupied<br>3=Night Setback |
| **PS** | 16951 | UInt | RW | EE<br>1100 | **Service PIN**<br>this Personal Identification Number controls access to the Service Menu. A value of 0 makes the menu always accessible. Values inclusively from 1 to 9,999 are used to control access to the menu. A matching number must be entered by the service technician. Values of 10,000 or greater will hide the menu. Entered P.I.N.s remain valid for only four (4) minutes after their use. **PS** has a default value of 1100. |
| **PU** | 16952 | UInt | RW | EE<br>0000 | **User PIN**<br>this Personal Identification Number controls access to the User Menu. A value of 0 makes the menu always accessible. Values from 0001 to 9999 inclusive are used to control access to the menu. A matching number must be entered by the User. Values of 10,000 or greater will hide the menu. Entered P.I.N.s remain valid for only four (4) minutes after their use. **PU** has a default value of 0. |
| **RM** | 16969 | UInt | RW | EE<br>0 | **Reading Mode**<br>indicates the current reading mode. This would be either Cooling or Heating as specifies by the system box type (**BT**). If **BT** is set to supply dependant, the point will indicate the current mode as determined by the source/duct temperature.<br>0=Average<br>1=Highest<br>2=Lowest<br>3=Hi/Lo VST mode<br>4=Device 0<br>5=Device 1<br>6=Device 2<br>7=Device 3<br>8=Primary GID |

| | | | | | |
|---|---|---|---|---|---|
| **SD** | 16983 | UInt | RW | EE<br>0 | **Calculated Setpoint Display**<br>specifies what method is used to display setpoint adjustments on an SBC-STAT3 LCD screen.<br>0 = Disable (+/-2.5)<br>1 = Zone Midpoint (Zone Temperature: (ZS) Zone Midpoint)<br>2 = Heating Setpoint (Zone Temperature: (CH) Heating Setpoint)<br>3 - Cooling Setpoint (Zone Temperature: (CC) Cooling Setpoint) |
| **SE** | 16984 | Boolean | RW | EE<br>1 | **Override Disabled/Enabled**<br>enables or disables the user's ability to enter extended occupancy override.<br>0=Disabled<br>1=Enabled (Default) |
| **T0** | 17002 | Real | R | RAM | **Thermostat Reading for G0**<br>indicates the current temperature sensed by G0. |
| **T1** | 17003 | Real | R | RAM | **Thermostat Reading for G1**<br>indicates the current temperature sensed by G1. |
| **T2** | 17004 | Real | R | RAM | **Thermostat Reading for G2**<br>indicates the current temperature sensed by G2. |
| **T3** | 17005 | Real | R | RAM | **Thermostat Reading for G3**<br>indicates the current temperature sensed by G3. |
| **TM** | 17011 | Real | RW | EE<br>0.5 | **Offset Increment**<br>specifies the magnitude of incremental changes to the User Setpoint Offset (TS). The User Adjust Position (TP) is multiplied by TM to determine the User Setpoint Offset (TS) value. If the User Adjust Increment is 0, you will not be able to alter the setpoint. |
| **TP** | 17013 | UInt | RW | RAM<br>0 | **User Adjust Position**<br>the User Setpoint Offset (TS) can be raised or lowered in integral steps. This property tracks the current step. It can be set to any signed integer but will be constrained to +/-2 when adjusted by an analog thermostat or to +/-5 when set through a digital thermostat. The point is used in combination with the User Adjust Increment (TM) to calculate the User Setpoint Offset. |
| **TR** | 17014 | UInt | | RAM<br>0 | **User Adjust Remaining**<br>displays the time remaining before the User Setpoint Offset (TS) setting is reset. |
| **TS** | 17015 | Real | RW | RAM<br>0 | **Setpoint Offset**<br>defines an offset for application to PID setpoints. This point shows the current value calculated when you multiply the User Adjust Position (TM) by the User Adjust Increment (TP). This setting is temporary and is valid only for TT minutes unless TT=0. |
| **TT** | 17016 | UInt | RW | EE<br>120 | **User Adjust Duration**<br>the User Setpoint Offset (TS) is a temporary setting. The TT property defines in minutes the duration for which the setting applies. After that time, the User Adjust Position and User Adjust Offset are reset to 0 degrees. If the User Adjust Duration is 0, then setpoint changes remain in effect until modified. The default value for TT is 120. |
| **ZS** | 17087 | Real | RW | RAM<br>70 | **Heating/Cooling Setpoint**<br>displays the midpoint between the current cooling and heating setpoints. This property reflects changes in both setpoints. A change in ZS results in the appropriate shift of both the cooling and heating setpoint maintaining the effective deadband. |

# C.3 UNIVERSAL INPUTS 1-5

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (3), Instance *N* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Universal Input *N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **IF** | 16878 | UInt | RW | EE 0.0 | **Input Filtering**<br>specifies the amount of time in tenths of seconds during which an input configured as a digital input must remain stable for the value to be considered reliable. This is also the weighted gain if the input is configured as analog. |
| **IP** | 16881 | Bool | RW | EE 0 | **Input Polarity**<br>0=normal<br>1=reverse |
| **OF** | 16919 | Real | RW | EE 0 | **UI Offset**<br>specifies an offset value to be added to the Universal Input's present-value. |
| **ST** | 16996 | UInt | RW | EE 7 | **Sensor Type**<br>0= digital<br>2= full scale, linear from min_pres_value to max_pres_value<br>3= 4–20mA linear scaled from min_pres_value to max_pres_value<br>7= –22.0 to 122.0°F thermistor<br>1,4,5, and 6 are unused |

# C.4 SUPPLY TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (3), Instance *8* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM Supply Temperature | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **DD** | 16794 | Real | RW | EE 2.5°F | **Auto Duct Delta Temperature**<br>This defines the temperature difference by which the supply air must either exceed the heating setpoint to switch to 'heating mode', or fall below the cooling setpoint to engage 'cooling' mode. |
| **IC** | 16875 | UInt | RW | EE 0 | **Input Channel Select**<br>1=UI01<br>2=UI02<br>3=UI03<br>4=UI04<br>5=UI05 |
| **OF** | 16919 | Real | RW | EE 0 | **Supply/Duct Temperature Adjustment**<br>Defines an offset used to adjust the current present-value. |
| **SM** | 16990 | UInt | R | RAM NA | **Cooling/Heating Supply Mode**<br>0=Cooling<br>1=Heating |

# C.5 OUTSIDE AIR TEMPERATURE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Input (3), Instance *9* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Outside Air Temperature | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | RAM 0 | four flags that indicate the general "health" of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is "reliable" as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **IC** | 16875 | UInt | RW | EE 0 | **Input Channel Select**<br>1=UI01<br>2=UI02<br>3=UI03<br>4=UI04<br>5=UI05 |
| **OF** | 16919 | Real | RW | EE 0 | **Outside Air Temperature Adjustment**<br>Defines an offset used to adjust the current present-value. |

# C.6 ANALOG OUTPUTS 1-4

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Output (1), Instance 1-12 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Analog Output *N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Output (1) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **reliability** | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the **present_value** is “reliable” as far as the device or operator can determine. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |
| **min_pres_value** | 69 | Real | RW | NRAM 0.0 | indicates the lowest number that can be reliably used for the **present_value** property of this object. |
| **max_pres_value** | 65 | Real | RW | NRAM 100.0 | indicates the highest number that can be reliably used for the **present_value** property of this object. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **relinquish_ default** | 104 | Real | RW | NRAM 0.0 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **DT** | 16804 | UInt | RW | EE 252 | **Data type** This property specifies the precision of present-value. |
| **HS** | 16863 | Real | RW | EE 100.0 | **Maximum Scaled Voltage** This property specifies the actual analog output value for a present-value of max-pres-value. |
| **LS** | 16894 | Real | RW | EE 0.00 | **Minimum Scaled Voltage** This property specifies the actual analog output value for a present-value of min-pres-value. |

# C.7 ANALOG VALUES (SETPOINTS) 1-7

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Analog Value (2), Instance 1-7 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM *N* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Analog Value (2) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM 0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM 0 | four flags that indicate the general “health” of the object. |
| **event_state** | 36 | BACnet Event State | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. Setting this value to True will allow for the present-value to become writable. |
| **units** | 117 | BACnet Eng. Units | RW | NRAM 95 | indicates the measurement units of this object. |

# C.8 PULSE INPUT

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Binary Input (3), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>*Pulse Input* | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Binary Input (3) | indicates membership in a particular object type class. |
| **present_value** | 85 | Real | RW | RAM<br>0.0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | BACnet Status Flags | RO | NRAM<br>0 | four flags that indicate the general “health” of the object. |
| **event_state** | 36 | BACnet Event State | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. Setting this value to True will allow for the present-value to become writable. |
| **polarity** | 84 | Enum | RW | NRAM<br>0 | indicates the polarity for the input. |
| **MD** | 16896 | UInt | RW | EE<br>0 | **Pulse Counter Mode**<br>0=Off<br>1=Falling edge<br>2=Rising edge<br>3=Both edges |
| **NP** | 16910 | UInt | RW | RAM<br>0 | **Pulse Count Total**<br>This property indicates the total amount of pulses accumulated by the device. |
| **SF** | 16985 | Real | RW | EE<br>0.00 | **Pulse Multiplier**<br>This property specifies the multiplier to apply against (NP) Pulse Count Total to generate the value in (SV) Scaled Pulse Count. |
| **SV** | 16998 | Real | R | RAM<br>0.00 | **Scaled Pulse Value**<br>This property indicates the scaled pulse value, which is (NP) x (SF). |

# C.9 BO01

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO01 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **FD** | 16825 | UInt | RW | EE 30 | **Shutoff Delay (seconds)**<br>This property shows the amount of time, in seconds, the fan output will stay energized once the zone temperature reaches the deadband. |
| **FN** | 16829 | UInt | RW | EE 0 | **Night Setback Fan Mode**<br>1 = fan runs for the entire period<br>0 = fan shuts off when zone temp is within the deadband. |
| **FO** | 16830 | UInt | RW | EE 0 | **Occupied Fan Mode**<br>1 = fan runs for the entire period<br>0 = fan shuts off when zone temp is within the deadband. |
| **FR** | 16832 | Real | RW | EE 0.5 | **Minimum Run Time**<br>This property shows the minimum amount of time, in minutes, the fan output will stay energized. This prevents short cycling of the fan output. |
| **FS** | 16833 | Real | RW | EE 1.0 | **Minimum Off Time**<br>This property shows the minimum amount of time, in minutes, the fan output will stay de-energized. This prevents short cycling of the fan output. |
| **FU** | 16835 | UInt | RW | EE 0 | **Unoccupied Fan Mode**<br>1 = fan runs for the entire period<br>0 = fan shuts off when zone temp is within the deadband. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **DB** | 16792 | Real | RW | EE<br>0.0 | **Fan Auto Mode Deadband**<br>used exclusively in conjunction with fan modes (FO, FU, FN) that are configured for Auto modes. When the device is in a schedule state where the fan operates in auto mode, the current zone temperature must exceed setpoint plus or minus the deadband temperature value in order for the fan to activate, followed by stages of heating or cooling. This property is useful for preventing the fan from turning on in situations where the zone temperature could possibly drift closely near mode changes. |
| **MT** | 16903 | UInt | RO | RAM<br>- | **Min Run/Off Countdown Timer**<br>reflects the amount of time remaining (in seconds) if the output is currently respecting minimum-on or minimum-off time. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours**<br>This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE<br>0.0 | **Run Limit**<br>This property specifies a run time limit in hours for the output. |

# C.10 BO02

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Binary Output (4), Instance 2 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>BO02 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | -<br>Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM<br>0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM<br>0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM<br>0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM<br>7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM<br>NULL | contains prioritized commands that are in effect for this object. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours**<br>This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE<br>0.0 | **Run Limit**<br>This property specifies a run time limit in hours for the output. |

# C.11 BO03

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Binary Output (4), Instance 3 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>BO03 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | -<br>Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM<br>0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM<br>0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM<br>0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM<br>7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM<br>NULL | contains prioritized commands that are in effect for this object. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours**<br>This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE<br>0.0 | **Run Limit**<br>This property specifies a run time limit in hours for the output. |

# C.12 BO04

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 4 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO04 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours** This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE 0.0 | **Run Limit** This property specifies a run time limit in hours for the output. |

# C.13 BO05

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Binary Output (4), Instance 5 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM BO05 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | - Binary Output (4) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM 0 | indicates the current value, in engineering units, of the object. |
| **status_flags** | 111 | Bit Str | RO | NRAM 0 | four flags that indicate the general "health" of the program. |
| **event_state** | 36 | Enum | RO | RAM 0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| **polarity** | 84 | Enum | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the **present_value**e property. If the **polarity** property is NORMAL, then the ACTIVE state of the **present_value** property is also the ACTIVE or ON state of the physical output as long as **out_of_service** is FALSE. If the Polarity property is REVERSE, then the ACTIVE state of the **present_value** property is the INACTIVE or OFF state of the physical output as long as **out_of_service** is FALSE. |
| **relinquish_ default** | 104 | Real | RW | NRAM 7 | the default value to be used for the **present_value** property when all command priority values in the **priority_array** property have a NULL value. |
| **priority_array** | 87 | BACnet Array | RO | RAM NULL | contains prioritized commands that are in effect for this object. |
| **RH** | 16966 | Real | RW | RAM | **Run Hours** This property shows the total amount of time, in hours, the output has been energized. |
| **RL** | 16968 | Real | RW | EE 0.0 | **Run Limit** This property specifies a run time limit in hours for the output. |

# C.14 FAN SPEED

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | Multi-StateValue(19), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>Fan Speed | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | Enum | RO | -<br>Multi-State Value (19) | indicates membership in a particular object type class. |
| **present_value** | 85 | Enum | RW | RAM<br>0 | indicates the current value. |
| **status_flags** | 111 | Bit Str | RO | NRAM<br>0 | four flags that indicate the general “health” of the program. |
| **event_state** | 36 | Enum | RO | RAM<br>0 | provides a way to determine if this object has an active event state associated with it. |
| **out_of_service** | 81 | Boolean | RW | NRAM<br>0 | indicates whether or not the process this object represents is not in service. |
| **number-of-states** | 84 | Enum | RW | NRAM<br>0 | indicates the number of states tracked by the object. |
| **RS** | 16972 | Unsigned | RW | RAM | **Fan Running Speed**<br>defines the status of the fan output |
| **SP** | 16993 | Unsigned | RW | EE<br>0.0 | **Fan Speed Application**<br>This property defines now many fan speeds are available. For a 3 speed fan, K1 enables low, K3 enables high and K2 enables medium speed. For 2 speed fans only low (K1) and high (K3) are available. For single speed fans, only low (K1) is available. |

# C.15 SCHEDULE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Schedule (17), Instance *1* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Schedule | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Schedule (17) | indicates membership in a particular object type class. |
| **present-value** | 85 | Unsigned or Enum | RO | RAM 0 | indicates the current schedule value |
| **effective-period** | 32 | Date Range | RO | RAM 0 | specifies the date rage in which the schedule is active. |
| **schedule-default** | 174 | Unsigned or Enum | RW | RAM 0 | specifies the datatype and value used for partial time-of-day scheduling. |
| **weekly-schedule** | 123 | Array List | RW | RAM 0 | contains time,value pair operations for each day of the week. |
| **list-of-object-property-references** | 54 | List | RO | NRAM 0 | contains the list of object properties that the schedule writes to. |
| **status-flags** | 111 | Bitstring | RO | NRAM 0 | indicates the general health of the object. |
| **reliability** | 103 | Enum | RO | NRAM 0 | indicates the reliability of the object. |
| **out-of-service** | 81 | Boolean | RO | NRAM 0 | indicates the service status of the object. |
| **priority-for-writing** | 88 | Unsigned | RW | NRAM 0 | indicates the priority which the Schedule will write with. |
| **HE** | 16853 | Bool | RW | EE 0 | H**ost Overrides**<br>0=disabled<br>1=enabled |
| **HO** | 16860 | UInt | RW | RAM 0 | **Host Schedule**<br>0 = unoccupied<br>1 = warm-up<br>2 = occupied<br>3 = night setback |
| **IS** | 16882 | UInt | RW | EE 3 | **Inactive Schedule State**<br>0 = unoccupied<br>1 = warm-up<br>2 = occupied<br>3 = night setback |
| **ZE** | 17081 | Bool | RW | EE 0 | **Receive Schedule**<br>0=No<br>1=Yes |

# C.16 ECONOMIZER

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (133), Instance *1* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Economizer | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **CM** | 16779 | Real | R | RAM N/A | **Calculated Minimum Position** This property displays the actual minimum position. |
| **ED** | 16808 | Real | RW | EE 1.0 | **Economizer Staging Delay (minutes)** This property specifies how many minutes the controller waits before using additional cooling stages after the economizer damper reaches 100. |
| **EE** | 16809 | UInt | RW | EE | **Economizer Enable**<br>0=Off<br>1=PID 1<br>2=PID 2<br>3=PID 3<br>4=PID 4<br>5=DO 3<br>6=DO 5 |
| **EM** | 16813 | Real | RW | EE 10.0 | **Economizer Minimum Position (%)** This property specifies the PID minimum position in percent for the economizer damper. |
| **ML** | 16090 | Real | RW | EE 0.0 | **Reset Limit for Economizer Minimum Protection** This property specifies the value at which maximum reset is used. When the value of the reset variable is equal to ML, the (MR) Maximum Reset is used in determining the calculated minimum position. |
| **MP** | 16900 | Real | RW | EE 0.0 | **Reset Setpoint for Economizer Minimum Position** This property specifies the value at which the reset action begins. When the value of the reset variable exceeds MP, reset action will be used in determining the economizer minimum position. |
| **MR** | 16901 | Real | RW | EE 0.0 | **Maximum Reset for Economizer Minimum Position** This property specifies the maximum amount to reset the minimum position setpoint (EM) by when reset is being used. |
| **MV** | 16904 | UInt | RW | EE 0.0 | **Reset Variable for Economizer Minimum Position**<br>0=Disabled<br>1=Zone Temperature<br>2=Supply Temperature<br>4=UI01<br>5=UI02<br>6=UI03<br>7=UI04<br>8=UI05<br>11=OAT |
| **OH** | 16920 | Real | RW | EE 60.0 | **OAT High Limit** This property specifies the Outside Air Temperature (OAT) high limit. If the OAT is above the high limit, the PID is set to the Economizer Minimum Position (EM). |
| **OL** | 16922 | Real | RW | EE 45.0 | **OAT Low Limit** This property specifies the Outside Air Temperature (OAT) low limit. If the OAT is below the low limit, the PID is set to the Economizer Minimum Position (EM). |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **CM** | 16779 | Real | R | RAM<br>N/A | **Calculated Minimum Position**<br>This property displays the actual minimum position. |
| **ED** | 16808 | Real | RW | EE<br>1.0 | **Economizer Staging Delay (minutes)**<br>This property specifies how many minutes the controller waits before using additional cooling stages after the economizer damper reaches 100. |

# C.17 REHEAT CTRL

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (133), Instance *3* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Reheat Ctrl | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **AF** | 16744 | UInt | RW | EE 0 | **Require Max Airflow?**<br>0=No<br>1=Yes |
| **AV** | 16750 | BitStr | RO | RAM | **Stages Available for Use**<br>This displays the stages that are currently available for use. This depends on the current options (indicated in (RO)) and the presence of a positive flow indication. Note that other uses of the relay outputs (A/C actuator, for instance) may remove those stages from reheat availability. This property is read-only. |
| **BA** | 16751 | UInt | RW | EE 0 | **Reheat Balance Mode**<br>0=No<br>1=Yes |
| **EN** | 16814 | BitStr | RO | RAM | **Stages Energized**<br>This displays those reheat stages that are currently energized. This property is read-only. |
| **FR** | 16832 | BitStr | RW | EE 15 | **Stages Requiring Flow**<br>Defines by the appropriate bit setting which of the available reheat stages requires a positive flow indication (see Proof of Flow). Not all sources of auxiliary heat (perimeter heating for instance) will require verification of air flow. By default, all stages require a positive flow indication. |
| **ID** | 16877 | Real | RW | EE 4.0 | **Interstage Delay Time**<br>This property shows the minimum amount of time, in minutes, before the next reheat stage will be energized. Stages are energized at this interval until the zone temperature rises to within the Reheat Offset of the Heating Setpoint. |
| **MX** | 16905 | Real | RW | EE 105.0 | **Maximum Supply Temperature**<br>This property establishes a maximum supply duct temperature above which the reheats will de-energize to become locked out. |
| **OF** | 16919 | Real | RW | EE 1.5 | **Reheat Offset**<br>This property specifies, in degrees, the offset from the calculated heating control setpoint that determines the temperature below which additional reheat stages can be energized. Note that in addition to this, the (AF) Require Max Airflow property may be set to hold off the addition of stages. For temperatures below the heating setpoint, but within the Reheat Offset of it, reheat stages are de-energized at 30 second intervals. |
| **RO** | 16970 | UInt | RW | EE 0 | **Reheat Application**<br>0=Disabled<br>1=2-Stage (K2-3)<br>2=2-Stage (K4-5)<br>3=4-Stage (K2-K5) |

# C.18 VALVE CTRL 1 AND 2

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (133), Instance *8 - 9* | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Valve Control x | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **AM** | 16745 | Bool | RW | EE 0 | **Auto/Manual**<br>0=Manual<br>1=Automatic |
| **CD** | 16771 | Bool | RW | EE 0 | **Change Valve Direction**<br>This property is used to set the direction of the Valve outputs. When the property is set to 0, the direction is normal, increase signal on BO2, and decrease signal on BO3. With the property set to 1, the output channels are reversed.<br><br>0=Normal<br>1=Reverse |
| **PP** | 16950 | UInt | RW | EE 0 | **PWM Period**<br>This property shows the amount of time, in seconds, the valve is to be pulsed ON, when PE=1 or pulse width modulation is enabled. |
| **RI** | 16967 | UInt | RW | EE 0 | **Recalibrate Int. Check**<br>This property shows the amount of time, in hours, between valve recalibration cycles. The valve position is recalibrated by driving the valve in the closed direction for the full travel time, then restoring the desired position. |
| **ST** | 16996 | UInt | RO | RAM | **Valve Status**<br>Indicates the status of the valve channel. |
| **TH** | 17009 | Real | RW | EE 105.0 | **High Temp Lockout**<br>Defines the highest Supply Temperature permitted before control is locked out. |
| **TL** | 17010 | Real | RW | EE 45.0 | **Low Temp Lockout**<br>Defines the lowest Supply Temperature permitted before control is locked out. |
| **UT** | 17032 | UInt | RW | EE 5 | **Update Threshold**<br>The Desired Valve Position is not updated until it differs from the actual valve position by at least the amount specified by Update Threshold. The desired positions of 0 and 100 are not subject to the threshold requirement. This property is used to minimize the actuation of the value for insignificant changes. |
| **VA** | 17041 | UInt | RO | RAM | **Valve Actual Position**<br>This property shows the actual valve position based on travel time. |
| **VD** | 17042 | UInt | RW | RAM | **Valve Desired Position**<br>This property shows the desired valve position to which the loop should control the valve to in order to bring the measured input variable closer to the setpoint. If AM=0, then VD can be set manually by a host. |
| **VI** | 17044 | Real | RW | EE 5.0 | **Valve Integration Constant**<br>This property shows the amount of proportional error history (0 to 25.5) used to calculate the desired position for the valve. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **VM** | 17045 | UInt | RW | EE<br>0 | **Valve Mode**<br>This property defines how the valve is physically controlled. Floating point motor control uses K2 and K3 so is not available in 2 or 3-speed fan mode. Pulse Width Modulation control uses K2 and is not available in 3-speed fan mode. Analog outputs may be used via a PID.<br><br>0=Pulse Width Modulation<br>1=Realing Point Motor Control |
| **VO** | 17046 | Real | RW | EE<br>0.0 | **Valve Temp Offset**<br>This property is added to Cooling or Heating Setpoint to determine the setpoint for the loop. If the controller is configured for hot water control, then the value is subtracted from calculated heating setpoint. If the controller is configured for chilled water control, then the value is added to calculated cooling setpoint. |
| **VP** | 17047 | Real | RW | EE<br>5.0 | **Valve Proportional Band**<br>This property specifies the input variable range, in degrees (0.0 to 100.0), over which the output value is proportional to the error value. The proportional band is offset from the setpoint for the loop. |
| **VT** | 17048 | UInt | RW | EE<br>180 | **Recalb with New Value**<br>This property shows the amount of time, in seconds, it takes the valve motor to travel from a fully closed position to a full open position. This action applied only to applications utilizing a valve on Binary Outputs. |
| **VU** | 17049 | UInt | RW | EE<br>0 | **Valve Use**<br>This property configures the valve for heating or cooling mode. If using Changeover Control it will be set automatically.<br><br>0=Disabled<br>1=Cooling<br>2=Heating |
| **CC** | 16780 | Unsigned | RW | EE<br>0 | **Valve Changeover Control Input**<br>This property allows for automatic changeover between heat and cool mode of valve operation. If enabled, this selects what input is used to monitor the water temperature. For analog inputs, the temperature is compared to the CC Heat Setpoint and CC Cool Setpoint to see if a changeover is required. If the input is digital then ON/1 = cooling and OFF/0 = heating. |
| **CS** | 16783 | Real | RW | EE<br>0.0 | **Changeover Cooling Setpoint**<br>For Automatic Changeover Control, this property defines a temperature below which the valve operation will switch to cooling mode. |
| **HS** | 16863 | Real | RW | EE<br>0.0 | **Changeover Heating Setpoint**<br>For Automatic Changeover Control, this property defines a temperature above which the valve operation will switch to heating mode. |

# C.19 PID CONTROL 1 - 4

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (133), Instance 11-14 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>PID Control 1-4 | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (133) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **AO** | 16746 | UInt | RW | RAM<br>NA | **Analog Output**<br>This property shows the scaled output value used by the analog output. AO is the PO value scaled to min-pres-value and max-pres-value of the corresponding analog output. |
| **CE** | 16772 | Bool | RW | EE<br>0 | **Control Enable**<br>0=No<br>1=Yes |
| **CS** | 16783 | Real | R | RAM<br>NA | **Control Setpoint**<br>This property shows the calculated, actual loop control setpoint. CS reflects the unoccupied setup/setback as well as any reset and/or thermostat setpoint adjustment. |
| **DB** | 16792 | Real | RW | EE<br>0 | **Deadband**<br>This property specifies the deadband around the setpoint for the control loop. When the value of the measured variable is within this deadband, the output signal remains constant at the midpoint of the throttling range (halfway between OH and OL). |
| **I1** | 16868 | Bool | RW | EE<br>0 | **Interlock 1 Enable**<br>0=Disabled<br>1=Enabled |
| **I2** | 16869 | Bool | RW | EE<br>0 | **Interlock 2 Enable**<br>0=Disabled<br>1=Enabled |
| **I3** | 16870 | Bool | RW | EE<br>0 | **No Flow Interlock Enable**<br>0=Disabled<br>1=Enable |
| **IC** | 16876 | UInt | RW | EE<br>0 | **Input Select**<br>0= disabled<br>1=Zone Temp<br>2=Supply Temp<br>4=UI1<br>5=UI2<br>6 =UI3<br>7 =UI4<br>8=UI5<br>9=Zone Heating<br>10=Zone Cooling<br>11=OAT |
| **IN** | 16880 | Real | R | RAM<br>0 | **Input Value**<br>This property displays the value of the input selected in IC. |
| **MR** | 16901 | Real | RW | EE<br>0 | **Maximum Reset**<br>This property specifies the maximum amount to reset the loop setpoint (SP) by when reset is being used. |
| **P1** | 16930 | Real | RW | EE<br>0.0 | **Interlock 1 Position**<br>This property specifies the PID output value when the Interlock 1 is enabled. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **P2** | 16931 | Real | RW | EE<br>0.0 | **Interlock 2 Position**<br>This property specifies the PID output value when the Interlock 2 is enabled. |
| **P3** | 16932 | Real | RW | EE<br>0.0 | **No Flow Interlock Position**<br>This property specifies the PID output value when the value of Fan Status is 0 (No Flow). |
| **PB** | 16940 | Real | RW | EE<br>0 | **Proportional Band**<br>This property specifies the input variable range over which the output value is proportional to the error value. The proportional band is centered around setpoint for the loop. |
| **PO** | 16949 | Real | RW | RAM<br>NA | **Percent Output**<br>This property shows the output value in hundredths of a percent. This value is calculated based on the error, change in error and past error for the control loop. PO is then scaled to the selected engineering units of the analog output and stuffed into the AO property, as well as the CV of the analog output. |
| **RC** | 16963 | Real | R | RAM<br>NA | **Reset Value**<br>This property displays the value of the input selected in RV. |
| **RL** | 16968 | Real | RW | EE<br>0 | **Reset Limit**<br>This property specifies the value at which maximum reset is used. When the value of the reset variable is equal to RL, the maximum reset (MR) is used in determining the calculated setpoint. |
| **RP** | 16971 | UInt | RW | EE<br>0 | **Reset Period**<br>This property specifies the reset period (in seconds) over which the error history is accumulated. If RP=10 seconds with a constant error of 2.0, then every second the error history would increase by 0.2. In five seconds, the error history would be 1.0 and at the end of ten seconds the error history would be 2.0. RP=0 disables integral action. |
| **RS** | 16972 | Real | RW | EE<br>0 | **Reset Setpoint**<br>This property specifies the value at which the reset action begins. When the value of the reset variable exceeds RS, reset action will be used in determining the calculated setpoint. |
| **RT** | 16973 | Real | RW | EE<br>0 | **Rate**<br>This property is the derivative rate and specifies a percentage of change in error that is to be used in calculating PO. The value is specified in percent-per-second. |
| **RV** | 16974 | UInt | RW | EE<br>0 | **Reset Variable**<br>0= disabled<br>1=Zone Temp<br>2=Supply Temp<br>4=UI1<br>5=UI2<br>6 =UI3<br>7 =UI4<br>8=UI5<br>11=OAT |
| **SG** | 16986 | UInt | RW | EE<br>0 | **Action**<br>0=normal (positive error causes an increase in output).<br>1=reverse (positive error causes a decrease in output) |
| **SP** | 16993 | Real | RW | EE<br>0 | **Loop Setpoint**<br>This property specifies the desired loop setpoint. This value is used with the unoccupied setup/setback and the reset to calculate the CS control setpoint. |
| **SU** | 16997 | Real | RW | EE<br>0 | **Setup/Setback**<br>This property specifies the amount to add (if SG=0) or subtract (if SG=1) from the setpoint during an unoccupied period. The adjusted setpoint will be displayed in CS. |

# C.20 OCCUPANCY DETECTION

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (131), Instance 1 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RO | NRAM<br>Occupancy Detection | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (254) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **IC** | 16876 | UInt | RW | EE<br>0 | **Input Select**<br>0 = None<br>1 = UI1<br>2 = UI2<br>3 = UI3<br>4 = UI4<br>5 = UI5<br>6 = OIA/B |
| **MD** | 16896 | UInt | RW | EE<br>30 | **Extended Occupancy Delay**<br>This property sets the amount of time, in seconds, the occupancy detector must remain on before the occupancy detector will override the zone. This prevents false triggers that might occur as others pass quickly through the zone. |
| **MR** | 16901 | UInt | R | RAM<br>NA | **Extended Occupancy Remaining**<br>This property displays the time remaining for occupancy detector override. |
| **MS** | 16902 | UInt | R | RAM<br>NA | **Occupancy Status**<br>0=No Detection<br>1=Detection |
| **MT** | 16903 | UInt | RW | EE<br>0 | **Extended Occupancy Duration**<br>This property defines, in minutes, the length of time to override the zone whenever occupancy is detected. |

# C.21 PROOF OF FLOW

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (131), Instance 2 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>Proof of Flow | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (131) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **PF** | 16944 | UInt | R | RAM<br>0 | **Proof of Flow Indication**<br>This property shows the status of the fan for Proof of Flow. |
| **IC** | 16876 | UInt | RW | EE | **Input Select**<br>0 =None<br>1 = UI1<br>2 = UI2<br>3 = UI3<br>4 = UI4<br>5 = UI5<br>6 = OIA/B |
| **PD** | 16942 | UInt | RW | EE<br>60 | **Proof of Flow Delay (seconds)**<br>This property shows the amount of time, in seconds, imposed before enabling a positive flow indication. |

# C.22 OUTSIDE AIR TEMP. BROADCAST

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | -<br>Proprietary (143), Instance 0 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM<br>Outside Air Temp. Broadcast | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | -<br>Proprietary (143) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM<br>6-NB-Rooftop-R1 | indicates the current schedule value |
| **CV** | 16785 | Real | R | RAM<br>0.0 | **Current Value**<br>shows the current value of the network broadcast schedule values received by the *NB*-ASC |
| **RB** | 16962 | Bool | RW | EE<br>0 | **Receive Broadcasts?**<br>0=No<br>1=Yes |
| **BE** | 16752 | Bool | RW | 0 | **OAT Broadcast Enable**<br>0=No<br>1=Yes |

# C.23 BROADCAST SCHEDULE

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---|---|---|---|---|---|
| **object_identifier** | 75 | BACnet ObjID | RO | - Proprietary (143), Instance 5 | a numeric code that is used to identify the object. |
| **object_name** | 77 | CharStr | RW | NRAM Broadcast Schedule | represents a name for the object that is unique internetwork-wide. |
| **object_type** | 79 | BACnet ObjType | RO | - Proprietary (143) | indicates membership in a particular object type class. |
| **profile-name** | 168 | CharStr | RO | RAM 6-NB-Rooftop-R1 | indicates the current schedule value |
| **CV** | 16785 | UInt | R | RAM 0 | **Current Value** indicates the current schedule value being received. |
| **RB** | 16962 | Bool | RW | EE 0 | **Receive Broadcasts?** 0=No 1=Yes |