

NB-FHC
USER MANUAL



© 2012 American Auto-Matrix™

This document is protected by copyright and is the attribute of American Auto-Matrix. It may not be used or copied in whole or in part for any purpose other than that for which it is supplied without authorization. This document does not constitute any warranty, expressed or implied.

Every effort has been made to ensure that all information was correct at the time of publication. American Auto-Matrix reserves the right to alter specifications, performance, capabilities and presentation of this product at any time.

American Auto-Matrix and Auto-Matrix are trademarks of American Auto-Matrix and are not to be used for publication without the written consent of American Auto-Matrix.

All other brand names or product names are trademarks or registered trademarks of their respective companies or organizations.

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

4/26/2012

Corresponds with v2.02 firmware.

This manual describes the installation and operation of the NB-FHC and FHC-SD. 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-FHC and FHC-SD products, and presenting the specifications for the controller.
- . Two: Wiring & Installation, detailing the wiring and installation procedures.
- . Three: Application Configuration, providing a step-by-step sequence for configuring the NB-FHC.
- . Four: Objects and Properties, listing all objects and properties within the device.
- . Five: Using the FHC-SD, provides an overview of the operator interface for the FHC-SD.
- . Appendix A: Channels & Attributes, enumerating the attributes in every channel along with their datatype, access, storage and default values.

This document contains certain style and formatting conventions for conveying information in a clear and concise manner:

- . Attribute names are shown in **bold**. For example: **present-value**.
- . Menu commands appear with a ">" symbol between levels. For example: File>Open.
- . *Italics* indicate a section of this manual or another publication.
- . The following formats are used to highlight important information:

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

SECTION 1: PRODUCT OVERVIEW

This section describes the NB-FHC and FHC-SD and provides operating specifications for the controller.

IN THIS SECTION

| | |
|---------------------------------------|-----|
| Product Overview..... | 1-3 |
| Features of the NB-FHC | 1-3 |
| Features of the FHC-SD | 1-3 |
| Product Specifications - NB-FHC..... | 1-4 |
| Networking | 1-4 |
| Product Setup and Configuration..... | 1-4 |
| Terminations..... | 1-4 |
| Power Requirements..... | 1-4 |
| Operating Environment | 1-4 |
| Battery Backup..... | 1-4 |
| Dimensions | 1-4 |
| Agency Approvals | 1-4 |
| Product Specifications - FHC-SD | 1-5 |
| Networking | 1-5 |
| Hardware | 1-5 |
| Power Requirements..... | 1-5 |
| Operating Conditions | 1-5 |
| Dimensions | 1-5 |
| Agency Approvals | 1-5 |

1.1 PRODUCT OVERVIEW

The NB-FHC is a native BACnet fume hood controller that offers unprecedented technology combined with flexibility of control found nowhere else in the industry today. By offering multiple methods for fume hood control in a single controller, fume hoods can be controlled using actual Face Velocity readings from any fume hood equipped with Auto-Flow components, through Sash Position Derived control, or a combination of both face velocity and sash position control methods for exceptionally responsive control of the fume hood. Because the NB-FHC is built to utilize the BACnet communications protocol, it enables straightforward integration to laboratory and building automation systems without the need for additional communication gateways or expensive custom drivers.

The NB-FHC offers a dedicated sensor bus input for the FHC-SD, a color touch screen local user interface. The local display provides user feedback information regarding the current status of the fume hood, including alarm notification, setpoint control and monitoring, as well as many other unique features. The FHC-SD also provides the ability for the fume hood commissioning agent to configure various NB-FHC set-up properties without having to use engineering tools.

1.1.1 FEATURES OF THE NB-FHC

- . Standalone or networked operation
- . Face Velocity Measurement or Sash Position derived methods of control in a single controller
- . Optimized PID control, executed every 50ms
- . Dedicated inputs and outputs for ease of set-up
- . Configurable setpoints for flow, face velocity and alarm parameters
- . Dedicated Cascade Flow Network capabilities for multi-fume hood installations
- . Occupancy detection option, allowing for shutdown or setback of the hood in unoccupied conditions
- . Easy configuration and firmware updates via *NB-Pro*
- . BACnet MS/TP Network Communications, supporting all standard baud rates
- . Battery-backed real time clock and memory
- . Self-diagnostic circuits and LED indicators for power, I/O, network activity, processor and programs.
- . Lockable enclosure with face velocity transmitter mounting option

1.1.2 FEATURES OF THE FHC-SD

- . Full color, 12-bit, 480x272 pixel TFT-based touch screen display
- . Powered by NB-FHC controller using a dedicated connection port, requiring no extra power transformer
- . Easy to use menu and icon driven navigation
- . Provides access to control setpoints, alarm limits and other controller configuration parameters - reducing the need to use commissioning software to make operational changes to the NB-FHC.
- . Full operator password protection
- . Flexible hood emergency mode functions

1.2 PRODUCT SPECIFICATIONS - NB-FHC

1.2.1 NETWORKING

- . **line signaling:** RS-485
- . **wiring:** shielded, twisted pair
- . **network protection:** dual tranzorbs, PTC
- . **communications speed:** 9.6kbps, 19.2kbps, 38.4kbps, 57.6kbps, 76.8kbps, *115.2kbps
- . **network configuration:** multidrop bus, per RS-485 specification and practice
- . **communications protocol:** BACnet MS/TP

1.2.2 PRODUCT SETUP AND CONFIGURATION

- . **Software:** NB-Pro v2.8 or later

1.2.3 TERMINATIONS

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

1.2.4 POWER REQUIREMENTS

- . **Power Source:** 22-29 VAC, 50/60 Hz, 4.16A max
- . **Line Protection:** 1.85A PTC protection

1.2.5 OPERATING ENVIRONMENT

- . **temperature range:** 32-122°F (0-50°C)
- . **humidity range:** 0-80% RH, non-condensing
- . **altitude:** up to 2000m

1.2.6 BATTERY BACKUP

- . Only Sanyo CR-1/3N or Duracell CL-1/3N battery rated 3.0 VDC is to be used.

1.2.7 DIMENSIONS

- . **size:** 8.407 x 6.5 x 1.25 in. (20.83x16.51x3.18 cm)
- . **shipping weight:** 3 lb. (1.36 kg)

1.2.8 AGENCY APPROVALS

- . UL Listed Management Equipment, Energy (PAZX) UL standard 916
- . UL Recognized Temperature-Indicating and Regulating Equipment – Component (XAPX2) UL standard 873
- . Complies with FCC Part 15, subpart B, for Class B Computing Device
- . Complies with CE Directive and Standards
- . BTL Listed - Advanced Application Controller (B-AAC Profile) - 135-2004 Standard

1.3 PRODUCT SPECIFICATIONS - FHC-SD

1.3.1 NETWORKING

- . **Line signaling:** EIA-485
- . **Data isolation:** GMR Data Isolated
- . **Protocol:** STATbus

1.3.2 HARDWARE

- . **Processor:** ARM7-TDMI, 86Mhz
- . **Memory:** 8MB-flash
- . **Expansion slot:** SD card socket for applying firmware updates
- . **LCD:** 4.3 in, 480x720 - 16:9 color TFT with white LED backlight
- . **Touch screen:** 4-wire analog resistive with protective flex window
- . **Beeper:** 4Khz

1.3.3 POWER REQUIREMENTS

- . **Power source:** 24 VAC/VDC, 50/60Hz
- . **Consumption:** 2.5W, 4W maximum
- . **Line protection:** 0.5A PTC, self-resetting fuse

1.3.4 OPERATING CONDITIONS

- . **Temperature:** 41-122° F (5 - 50° C) ambient
- . **Humidity:** 0-90%, non-condensing
- . **Altitude:** -2000 to + 10,000 ft. (-650 to + 3500m)

1.3.5 DIMENSIONS

- . **Size:** 3.400 in. (8.65cm) x 6.000 in. (15.24cm) x 1.019 in. (2.59cm)

1.3.6 AGENCY APPROVALS

- . CE Approved
- . FCC, Class A Computing Device
- . UL listed 916, Management Equipment, Energy (PAZX)
- . UL Listed 873, Component Temperature-Indicating and Regulating Equipment (XAPX2)
- . UL Listed, Component Temperature-Indicating and Regulation Equipment for Canada (XAPX8)

SECTION 2: WIRING AND INSTALLATION

This section describes how to make the necessary connections for power and network communications to the NB-FHC. Connection of input and output devices, as well as the use of remote I/O on the STATbus is covered as well.

IN THIS SECTION

| | |
|---|------|
| NB-FHC Installation | 2-3 |
| Mounting the Controller and Enclosure..... | 2-3 |
| Mounting the Face Velocity Transmitter to the Enclosure | 2-4 |
| Powering the NB-FHC..... | 2-5 |
| Connecting the RS-485 (BACnet MS/TP) Network..... | 2-7 |
| Universal Inputs | 2-11 |
| IVR Jumpers for Universal Inputs | 2-12 |
| Loop Powered Universal Inputs | 2-13 |
| Current Inputs | 2-13 |
| Voltage Inputs | 2-14 |
| Thermistor Inputs | 2-14 |
| Non-Linear Inputs..... | 2-14 |
| Analog Outputs | 2-16 |
| Connecting Analog Outputs | 2-16 |
| VI Jumper for Analog Outputs..... | 2-16 |
| Wiring the Auto-Flow (AF-ACT) Actuator | 2-17 |
| Configuring Analog Outputs | 2-17 |
| Digital Outputs..... | 2-20 |
| Connecting the Digital Output | 2-20 |
| Cascade Flow Setup..... | 2-21 |
| FHC-SD Touch Screen Display..... | 2-22 |
| Mounting the FHC-SD..... | 2-22 |
| Disassemble the FHC-SD..... | 2-23 |
| Mounting the Backplate | 2-24 |

2.1 NB-FHC INSTALLATION

This section will explain the steps necessary to mount the FHC, supply power to the controller, connect the controller to the communications network and connect and configure the controller's inputs and outputs

WARNING



This product should only be used in a manner specified by American Auto-Matrix. No operator replaceable parts exist on this product. Only trained personnel should service equipment.

There are three main steps to installing the FHC. First the controller must be physically mounted. Second, the controller must be connected to a suitable power supply. Third, the controller must be connected to the RS-485 communications network.

NOTE



All equipment must be installed by a licensed electrician in accordance with the NEC/CEC and local codes.

2.1.1 MOUNTING THE CONTROLLER AND ENCLOSURE

The FHC product consists of the main unitary controller mounted inside of a rugged metal enclosure. The metal enclosure has a hinged or removable door for ease of access in the event that the unit is mounted in tight quarters. The enclosure is equipped with a lock to prevent unauthorized accessed to the unit.

Because of the application specific functionality of fume hood controllers, it is recommended that it be installed on top of the fume hood or a ceiling compartment above the fume hood itself. Mounting close proximity above the fume hood will enable less problematic connections of peripheral devices such as the face velocity transmitter/sensor, sash position potentiometer, exhaust flow transmitter, and the FHC-SD.

The mounting area should be flat and unobstructed by other equipment or machinery, free of moisture, and located away from potential leakage. The product should be mounted to a site which provides access to a stable 24 VAC power transformer from which the controller will draw its power. The temperature of the mounting location must be between 32° F and 122° F (0° C to 50° C) with a relative humidity of 0-80% non-condensing.

The mounting dimensions for the NB-FHC are shown in Figure 2-1.

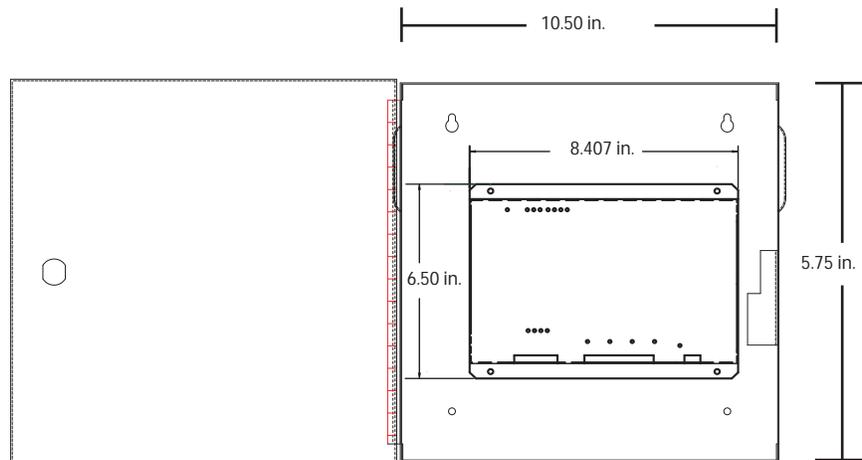


Figure 2-1: Mounting Dimensions for the NB-FHC (all dimensions in inches)

NOTE



When installing the NB-FHC, make sure that there is sufficient room to allow easy insertion and removal of terminal block plugs.

2.1.2 MOUNTING THE FACE VELOCITY TRANSMITTER TO THE ENCLOSURE

The backside of the product enclosure provides mounting holes to accommodate mounting of the Auto-Flow Face Velocity Transmitter (FVR-1A or FVR-1B). Please refer to the installation instructions that are included with referenced peripherals for proper wiring and air flow tubing connections. If you elect to not mount the transmitter on the product enclosure, please refer to the installation instructions accompanying the transmitter.

To mount the Face Velocity Transmitter:

1. Remove the four mounting screws from the rear side of the enclosure using a Phillips screwdriver.
2. Align the screw holes of the Face Velocity Transmitter to the enclosure. The result of this alignment will mount the transmitter facing in a horizontal direction. The transmitter may be mounted with the airflow pickup connections to either the left or right (depending on entry of flow tubes).
3. Place the mounting screws through the mounting holes of the transmitter to make contact with the screwholes of the enclosure. Tighten each screw.
4. Once the transmitter is securely mounted to the enclosure, refer to the installation instructions included with the transmitter to install the high and low pickup tubes of the face velocity sensor.
5. Wire the transmitter to the controller in accordance to the instructions included with the transmitter, and controller wiring instructions (Section 2-2 of this manual).

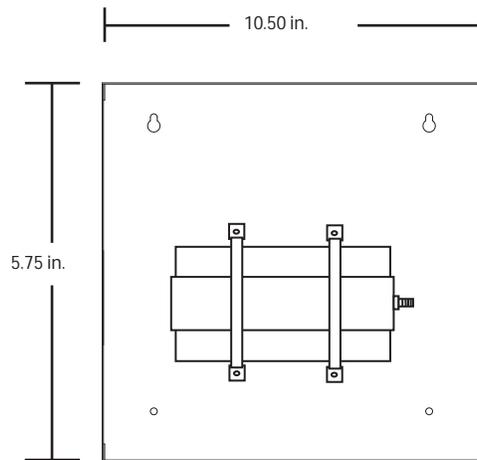


Figure 2-2 Mounted Face Velocity Transmitter

NOTE



Ensure there are no kinks or sharp bends in the airflow tubing that would prevent the flow of air between the transmitter and the face velocity sensor.

NOTE



Allow a minimum of 12' of airflow tubing between the transmitter and face velocity sensor for optimal performance and results.

2.1.3 POWERING THE NB-FHC

The NB-FHC can be used in a variety of configurations. Because of this, the power requirements of the controller can vary greatly and are very much application specific. There are three main parts of the NB-FHC which require power: the on-board I/O, the 24 VAC courtesy power output, and the power outputs on the STATbus ports.

WARNING



AAM recommends the use of a single transformer to power a single controller. Do not tie the DC and AC grounds together.

The NB-FHC requires a 22-29 VAC, 50/60 Hz NEC Class 2 transformer capable of providing a minimum of 1.85 A. The transformer is connected to the NB-FHC at pins 43 and 44 on terminal block TB12 as shown in Figure 2-3.

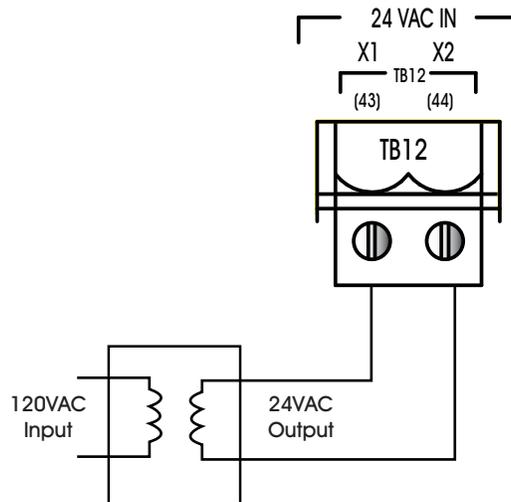


Figure 2-3: Connecting Power to the NB-FHC

Once power is connected to the NB-FHC, the Main Power light, located to the left of TB12 as shown in Figure 2-4, should be lit.

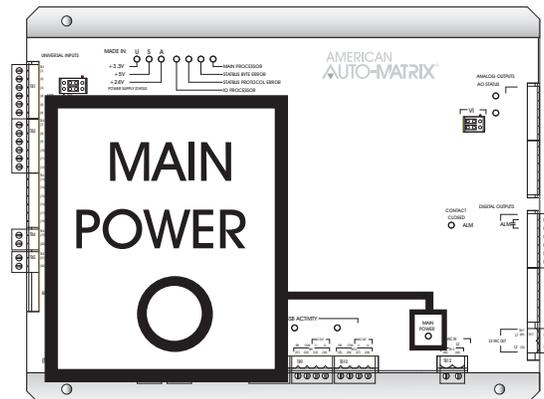


Figure 2-4: Location of the Main Power indicator

The Power Supply Status lights, shown in Figure 2-5, indicate that the 3.3 V, 5 V, and 24 V on-board power supplies are functioning properly. The Main Processor light will begin to blink. When the processor is functioning normally, this light will cycle on and off approximately every half second.

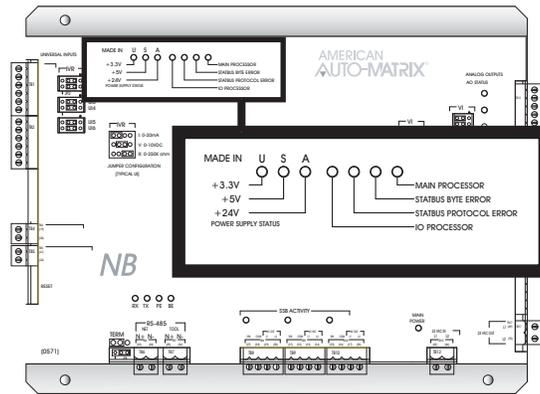


Figure 2-5: Location of the Power Supply Status Indicators and the Main Processor Indicator

2.1.4 CONNECTING THE RS-485 (BACNET MS/TP) NETWORK

The NB-FHC communicates via RS-485 via BACnet MS/TP protocol. As shown in Figure 2-6, the NB-FHC has two network connections at terminal blocks TB6 and TB7. The BACnet MS/TP network is connected to terminal block TB6 using 14-22 AWG, shielded, twisted pair wiring making sure to maintain polarity between devices.

The shield wire run to the first controller in the network should be connected to a reliable earth ground. At each connection along the network, the shield wires should be connected together and taped back. At the last controller, the shield wire should simply be taped back.

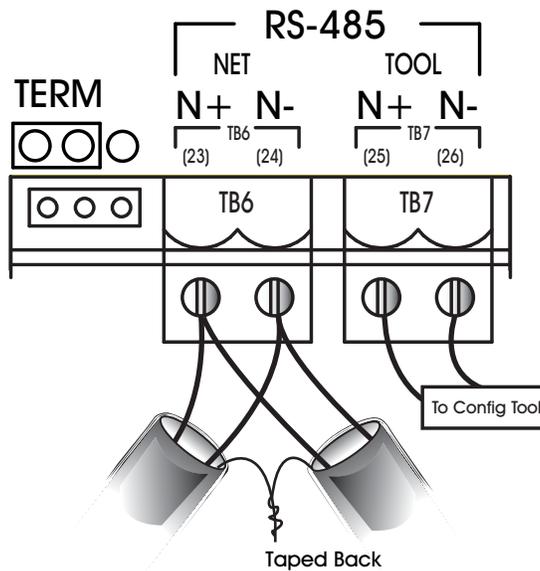


Figure 2-6: Proper Connection of the NB-FHC to an RS-485 Network

CAUTION

BACnet MS/TP communication networks requires shielded, twisted pair cabling. Do not attempt to network an MS/TP communication bus without shielded, twisted pair cabling. Failure to meet this specification may result in poor network performance.

The second network port on the NB-FHC, terminal block TB7, is intended for the temporary connection of a configuration device. Do not use this terminal block to establish a regular connection between devices. this terminal block The connection at TB7 is **NOT** intended for connection of devices that are part of the building RS-485 network.

2.1.4.1 RS-485 SHIELD GROUNDING

Proper shielding of the RS-485 cabling can help reduce the risk of communication problems and damage to equipment because of transient voltage spikes (e.g. lightning). The following guidelines should be followed at all times when grounding RS-485 cable shields:

- . Each MS/TP network segment should have a single point of shield ground.
- . Do not ground the RS-485 network shield using a controller terminal.
- . Never ground both ends of a shield because differences in potential between the grounds may include a current on the shield, causing network interference.
- . At ungrounded, exposed shield points (end of a segment between devices), tape back the shield to the wire jacket.

2.1.4.2 RS-485 NETWORK TERMINATION

For NB-FHC units that are electrically the last device on the network, it may be necessary to connect a termination resistor across the N+ and N- terminals to minimize unwanted communications line effects, such as signal reflection. Only the end units on the multi-drop network should be terminated. The NB-FHC provides the option of connecting an internal, jumper-selectable, 249 Ω termination resistor across the N+ and N- terminals. To connect the termination resistor, the TERM jumper, located to the left of TB6, should be moved to the left two pins as shown in Figure 2-7a. To disconnect the internal termination resistor, move the TERM jumper to the right two pins as shown in Figure 2-7b. If necessary, additional termination resistance can be added by connecting a resistor between the N+ and N- pins. Network termination should also be applied at the opposite end of the network at the first device.

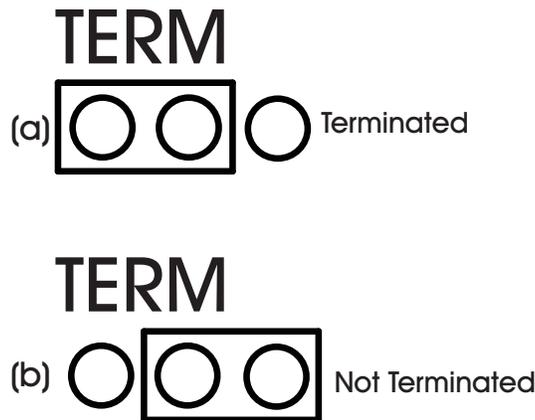


Figure 2-7: Configuring the TERM Jumper for RS-485 Communications Termination

CAUTION



Do not mis-match terminating resistors on the RS-485 network. Ensure that both resistors on a segment are rated for the same value.

Once power is supplied to the NB-FHC and the unit is connected to the RS-485 network, the RX and TX lights, located just above terminal blocks TB6 and TB7, shown in Figure 2-8, should blink rapidly, indicating that the controller is communicating over the network.

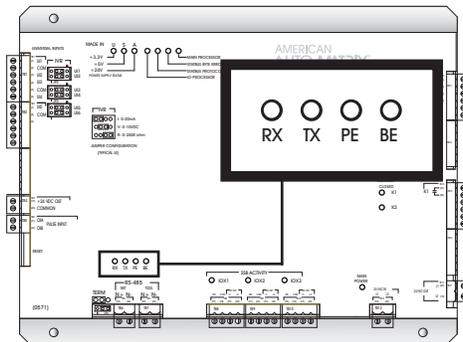


Figure 2-8: Network Status Indicator Lights

The BE light indicates that a Byte Error has occurred. This type of error occurs when information is being transferred over the network and data is lost. A Byte Error occurs if a problem is detected in the start, stop, or data bit timings. This type of error would also be generated if communication baud rates do not match. In this case, the NB-FHC detects that the error has occurred, lights the BE light and sends a request over the network for the information to be resent.

A packet of data is a number of bytes of data that are transmitted over the network together. Each packet of data includes a checksum value used to validate the data. A Packet Error occurs when NB-FHC detects

a mismatch between the calculated checksum and the value received with the packet. Packet errors can also be caused by faulty transmission or reception of a data packet. When this happens, the PE light will be lit, indicating that such an error has occurred. A request will then be sent for the data to be retransmitted.

2.2 UNIVERSAL INPUTS

The NB-FHC provides on-board universal inputs (UIs) for monitoring end devices to carry out control sequences. Each universal input is dedicated to a specific function for fume hood applications. Dependent on the application mode that the NB-FHC is configured for, some inputs may or may not be used. Devices are connected to the on-board universal input points on the NB-FHC via terminal blocks TB1, and TB2. There are fewer common terminals than universal input points, so adjacent points must share common terminals. For example, UI1 and UI2, located at terminals 1 and 3 respectively, would both be connected using the common ground on terminal 2.

Table 2-1 Universal Input Function Assignments

| Input | Designation | Function |
|---------|-------------|--------------------|
| Input 1 | FV | Face Velocity |
| Input 2 | EXF | Exhaust Flow |
| Input 3 | AVP | Aux. Digital Input |
| Input 4 | EMG | Emergency Input |
| Input 5 | S1 | Sash Position |

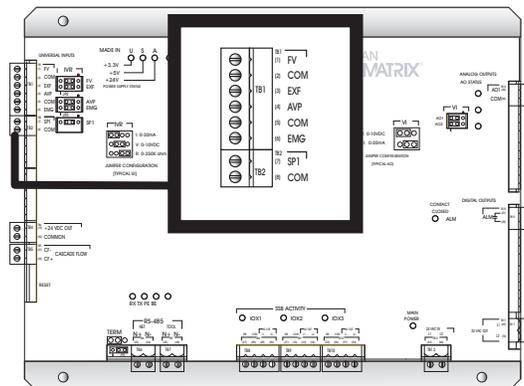


Figure 2-9: Location of the Universal Inputs on the NB-FHC

2.2.1 IVR JUMPERS FOR UNIVERSAL INPUTS

The IVR jumpers are located next to terminal block TB2 as shown in Figure 2-10.

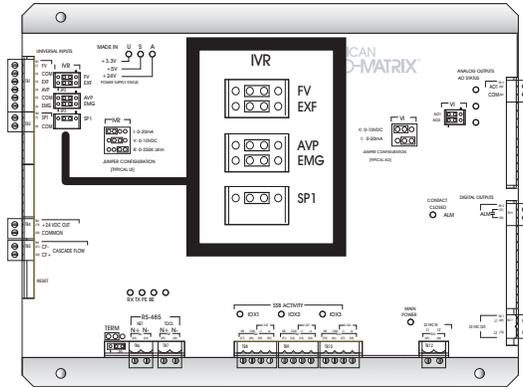


Figure 2-10: Location of the IVR Jumpers for Universal Inputs

The IVR jumpers are used to configure the NB-FHC for the different types of signals that can be connected to the associated universal input point. By moving the jumper to different positions the associated point can be selected as either a current, voltage, or resistance input. The possible positions for the IVR jumper are show in Figure 2-11.

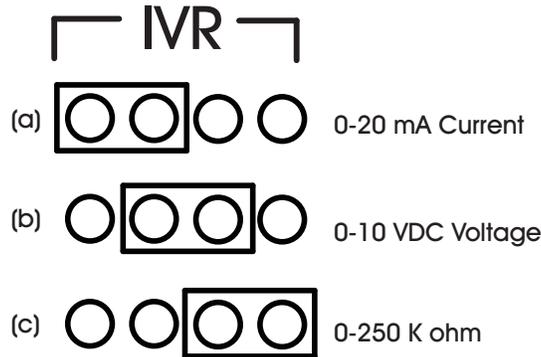


Figure 2-11: IVR Jumper Positions

Table 2-2 Jumper Positions Configuration Table

| Jumper Configuration (corresponds to Figure 2-11) | Input Configuration |
|---|---------------------|
| “I” Position - Figure 2-11 item-a | 0-20mA Sensors |
| “V” Position - Figure 2-11 item-b | 0-10V Sensors |
| “R” Position - Figure 2-11 item-c | Resistive Sensors |

2.2.2 LOOP POWERED UNIVERSAL INPUTS

For inputs that require power, the *NB-FHC* provides a 24 VDC power connection capable of providing up to 150 mA via terminals 19 and 20 on terminal block TB4. This is enough current to power a few devices. An example of a loop powered input connected to the *NB-FHC* is shown in Figure 2-12.

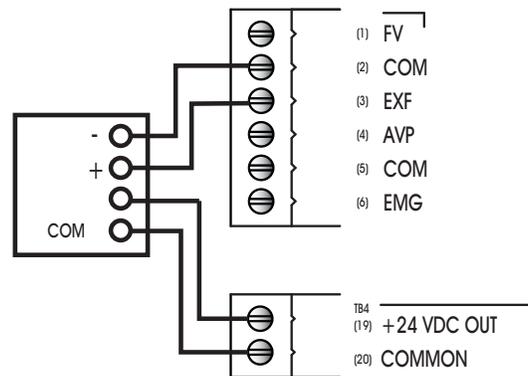


Figure 2-12: Using Loop Powered Universal Inputs

2.2.3 CURRENT INPUTS

Any sensor which generates a signal in the form of a current is classified as a current sensor. Ranges of 0-20 mA and 4-20 mA are common in sensors. The current produced by these sensors is directly proportional to the value being measured. For example, if a pressure sensor measured 0 to 5 inches of water gauge and had an output range of 0 to 20 mA, then a reading of 10 mA would correspond to a pressure of 2.5 inches of water gauge.

The first thing that needs to be done so that a universal input can be used as a current input is to make sure that the IVR jumper is in the correct position (See “IVR Jumpers for Universal Inputs”, Section 2.2.1). In this case, you would make sure that the IVR jumper for the input was set to the “I” position.

Once the jumper has been set, you need to use *NB-Pro* to configure the input object. Configuration entails specifying what type of sensor is connected to an input and then specifying the range of values over which that sensors operates.

To specify the sensor type, access the Universal Input object corresponding to the input you are configuring. Set the **(ST) Sensor Type** property to a value of 3, for 4-20 mA inputs, or 8, for 0-20 mA inputs.

Some Universal Inputs in the *NB-FHC* are hardcoded to a particular sensor type and will not allow modification.

You also need to specify the range over which the sensor operates. This is necessary so that the *NB-FHC* can calculate the measured value from the input signal. The **min_pres_value** property should be set to the lowest value that your sensor can measure. The **max_pres_value** property should be set to the maximum scaled value for your input.

2.2.4 VOLTAGE INPUTS

Any sensor which outputs a voltage in response to a measured value is classified as a voltage sensor. Voltage sensors behave in very much the same way as current sensors. The primary differences have to do with the internal circuitry of the *NB-FHC* and how the signal is read.

The first thing that needs to be done so that a universal input can be used as a voltage input is to make sure that the IVR jumper is in the correct position. In this case, you would make sure that the IVR jumper for the input was set to the “V” position.

Once the jumper has been set, you need to use *NB-Pro* to configure the input object. Configuration entails specifying what type of sensor is connected to an input and then specifying the range of values over which that sensors operates.

To specify the sensor type, access the Universal Input object corresponding to the input you are configuring. Set the **(ST) Sensor Type** property to a value of 2, for 0-5 V inputs, or 6, for 0-10 V inputs.

Some Universal Inputs in the *NB-FHC* are hardcoded to a particular sensor type and will not allow modification.

You also need to specify the range over which the sensor operates. This is necessary so that the *NB-FHC* can calculate the measured value from the input signal. The **min_pres_value** property should be set to the lowest value that your sensor can measure. This would correspond to the reading at zero volts. The **max_pres_value** property should be set to the maximum scaled value for your input. For example, if a 0-10 V carbon dioxide sensor measuring from 0-5000 PM would have **min_pres_value** would be set to 0 and **max_pres_value** would be set to 5000.

2.2.5 THERMISTOR INPUTS

The thermistor is one of the most common types of resistive sensors for temperature measurement. The thermistor’s combination of high accuracy over a wide range coupled with its low cost makes it one of the most popular temperature sensors used. Because of the thermistor’s popularity, the *NB-FHC* has a pre-programmed response curve for a **Precon Type III**.

The first thing that needs to be done to configure a universal input to be used as a thermistor input is to make sure that the IVR jumper is in the correct position. In this example, you would make sure that the IVR jumper to the “R” position.

Once the jumper has been set, you need to use *NB-Pro* to configure the input object. Configuration entails specifying what type of sensor is connected to an input and then specifying the range of values over which that sensors operates. Access the Universal Input object and set the **(ST) Sensor Type** property equal to a value of 7, which corresponds to Thermistor. The *NB-FHC* will then automatically set the **min_pres_value** to -30.0 and the **max_pres_value** to 230.0, the minimum and maximum values that can be read by this type of sensor. The temperature will now be displayed in the **present_value** property of this object.

Some Universal Inputs in the *NB-FHC* are hardcoded to a particular sensor type and will not allow modification.

2.2.6 NON-LINEAR INPUTS

The thermistors discussed previously are just one example of a sensor who’s output characteristics are well defined. Because Precon type III thermistors are so prevalent, the output response curve is included with the controller. However, there are a number of other common sensors who’s response is non-linear.

For these types of inputs, the NB-FHC provides the option of using one of the two Piecewise Curve objects to specify the response of the input device.

The Piecewise Curves can be used with current, voltage, or resistive inputs. Before configuring the Piecewise Curve, make sure that the IVR jumper is in the correct position for the type of sensor being used.

To use an input with a non-linear response, you must define the Piecewise Curve and then set the input to use the curve you have defined to scale its readings. The response data necessary to construct the curve will usually be available in the catalog from which the sensor was ordered or on the data sheet accompanying the sensor.

To set the input to use the Piecewise Curve, set the **(ST) Sensor Type** property for the input equal to 4 or 5. This tells the NB-FHC to use Piecewise Curve 1 or 2 respectively. The next step is to set the minimum and maximum scaled values using the **min_pres_value** and **max_pres_value** properties. The **min_pres_value**, should be set to the lowest scaled value from your piecewise curve. For most application this will correspond to the value of **Y1** in the Piecewise Curve object. Similarly, **max_pres_value** should be set to the maximum scaled value for the input. This will most likely be the value of **YB** in the Piecewise Curve object. Some Universal Inputs in the NB-FHC are hardcoded to a particular sensor type and will not allow modification.

2.3 ANALOG OUTPUTS

2.3.1 CONNECTING ANALOG OUTPUTS

The NB-FHC has one (1) Analog Output, which is used to control the Auto-Flow AF-ACT Exhaust Damper actuator for fume hood applications. A second Analog Output is used exclusively for Cascade Flow network applications, explained later in this section.

| NOTE | |
|---|---|
|  | <p>Analog Output 2 is designed exclusively for Cascade Flow network applications and cannot be used to control equipment or end-devices associated with laboratory and fume hood control.</p> |

Outputs are connected to the on-board analog output points on the NB-FHC via terminal blocks TB13. Each terminal block has connections for three analog outputs. Each output also has a common terminal associated with it. The common terminal provides a ground connection for the output.

Before the output is physically connected, the VI jumper should be set to determine how the output is to function. Output devices are connected to an analog output point by connecting the positive wire to an odd numbered terminal and the negative to the associated even numbered common terminal, labeled COM, directly below it.

2.3.2 VI JUMPER FOR ANALOG OUTPUTS

Analog outputs may be configured via a jumper to operate either as a 0-10 V voltage output or a 0-20 mA current output. Each analog output has a VI jumper associated with it which is used to select the output as either a voltage or current output.

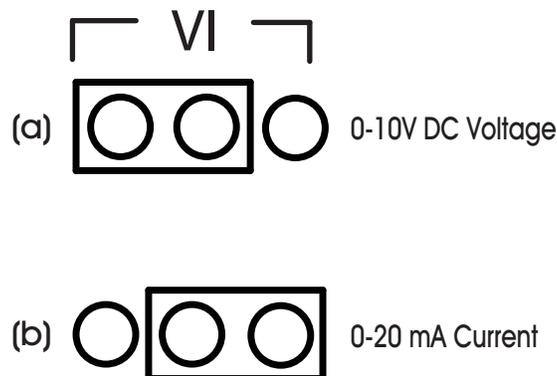


Figure 2-13: VI Jumpers for Analog Outputs

This jumper selects whether the output will operate as a 0-10 V voltage output or as a 0-20 mA current output. For a voltage output, you would move the jumper to the left, to the “V” position. For a current output, you would move the jumper to the Right, to the “I” position. For example, if you want to configure AO1 to operate as a current output, you would move the corresponding VI jumper to the right-most position. This is shown in Figure 2-13.

Each analog output point has an AO Status light associated with it. This light indicates when the output is powered and the approximate output level. It will light green if the associated output is between 0% and 33% of full scale, orange between 34% and 66% of full scale, and red between 67% and 100% of full scale.

2.3.3 WIRING THE AUTO-FLOW (AF-ACT) ACTUATOR

The AF-ACT Actuator is a high-speed response actuator designed exclusively for Auto-Flow fume hood control applications. The actuator is wired to Analog Output 1. Power for the actuator can be supplied through the use of the courtesy 24VAC output located on the bottom right side of the *NB-FHC* unitary controller.

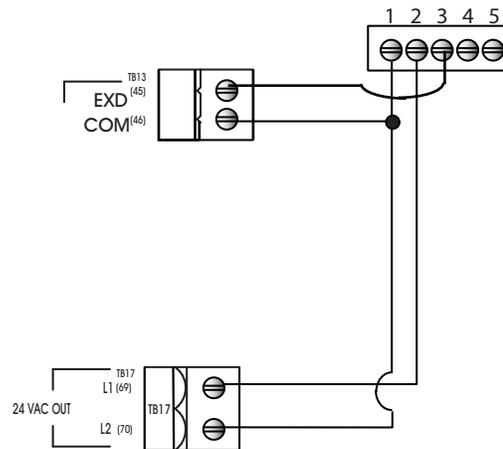


Figure 2-14 AF-ACT Example Wiring Diagram

NOTE



Careful consideration should be taken if you use the courtesy 24VAC output to power the AF-ACT actuator and a binary output source for external alarm. Please check required power ratings of the binary output device you are connecting to the *NB-FHC* if you intend to power both simultaneously. If you are unsure, please contact AAM Technical Services for additional assistance.

2.3.4 CONFIGURING ANALOG OUTPUTS

Both voltage and current outputs can be configured to work as either normal or reverse-acting. The analog output on the *NB-FHC* can operate in automatic mode, where the output is controlled by the *NB-FHC*'s control loop. For emergency situations or manual override purposes, the Analog Output can be commanded using BACnet's Command Prioritization feature (also known as Priority Array). The minimum and maximum desired positions can be specified and the output can be interlocked for emergency situations.

The first thing that needs to be done is to make sure that the VI jumper is in the correct position. For a voltage output, you want to set the jumper to the “V” position and, for a current output, you want the jumper in the “I” position.

Properties **(LS)** and **(HS)** specify the minimum and maximum scaled voltage for the outputs, expressed as a percentage of the full scale. **min_pres_value** and **max_pres_value** are used to specify the display range for the **present_value**. **min_pres_value** is the value of **present_value** that corresponds to an output of **(LS)**. Similarly, **max_pres_value** is the value of **present_value** that will result in an output of **(HS)**.

For example, if the **present_value** is to be displayed as a percentage (0-100%) of a 10 VDC output range, set **min_pres_value** to 0 and **max_pres_value** to 100 (a display range of 0%-100% of full scale). Then set **LS=0.0** and **HS=100.0**, so that **present_value=0** represents 0.0% of the output range and **present_value=100** represents 100.0% of the output range. This is shown in Figure 2-15.

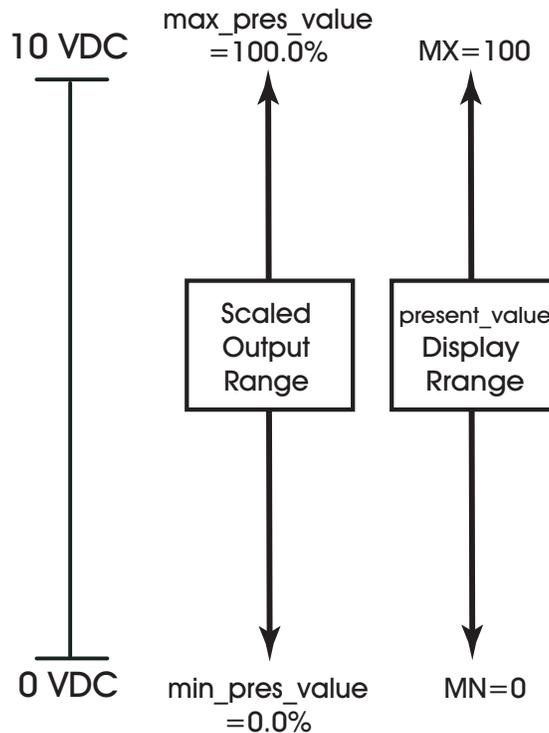


Figure 2-15: 0 to 10 V Displayed as a Percentage

If the output device in the previous example only operated from 2-10 V instead of 0-10 V, you would simply change the value of **(LS)** to be 20.0 because 2 V is 20% of the 10 V maximum. Everything else from the previous example would remain the same. This is the default setting for the standard AutoFlow actuator using a voltage output. This can be seen in Figure 2-16.

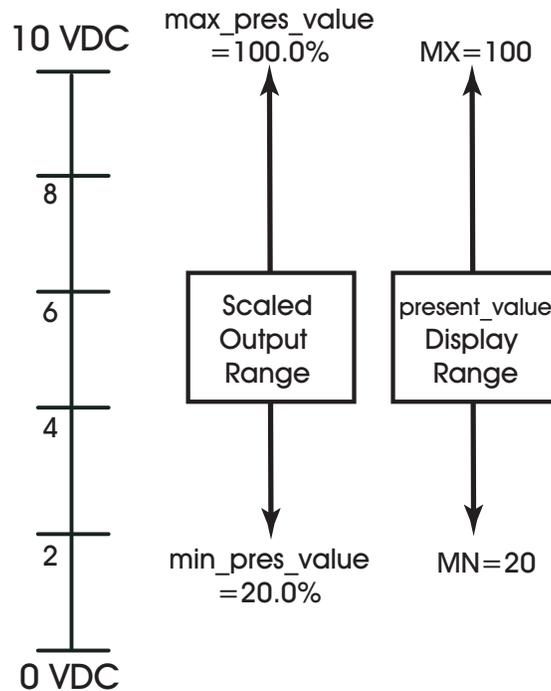


Figure 2-16: 2 to 10 V Displayed as a Percentage

2.4 DIGITAL OUTPUTS

The NB-FHC has one digital output object intended for use with an external alarm. The digital output can be controlled automatically by the NB-FHC or can have its output state set manually, either by the user or by a remote host.

2.4.1 CONNECTING THE DIGITAL OUTPUT

Outputs are connected to the on-board digital output points on the NB-FHC via terminal blocks TB15 and TB16. Each terminal block has connections for three digital outputs. Output devices are connected to a digital output point by connecting the wires to the terminal on the connector.

Whenever a digital output is energized, the associated red Contact Closed light will be lit. This light will stay lit until the output is turned off.

The NB-FHC can be used to power digital outputs using the courtesy 24 VAC voltage output connection located at terminals 69 and 70 on terminal block TB17. This power output is capable of providing 24 VAC at up to 1 A. Figure 2-17 shows how this output can be used to power a digital output.

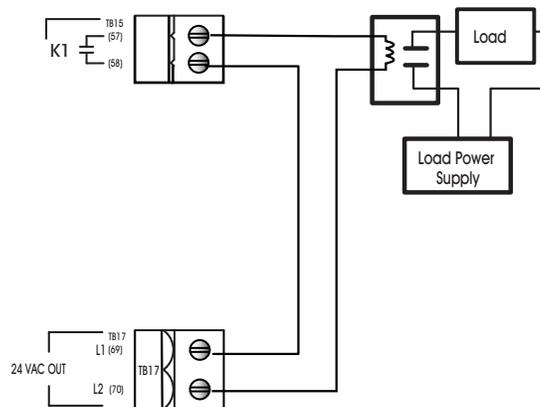


Figure 2-17: Digital Output Using the Power Output

2.4.2 OPTIONAL LIGHTING OUTPUT

The FHC's second digital output can be used as an optional lighting output utilizing an SSB-DO1 module. When present, the FHC will automatically enumerate the SSB-DO1 module and the output will then be available for control. The Show Lighting Output Button on Display (LB) property will expose the feature on the display and allow the output to be commanded On/Off. Additionally, the output can be controlled automatically by the FHC through the use of the Auto Stuff feature in the Digital Output object.

2.5 CASCADE FLOW SETUP

The *NB-FHC* provides terminal connections for a cascade flow network, a dedicated serial communication network designed to send high speed updates of exhaust flow to multiple *NB-FHC* controllers. The receiving unit reads the signal, adding its own exhaust flow (read from the exhaust flow transmitter), then transmitting onto the next successive *NB-FHC* controller. The final flow can be sent to an *NB-GPC-LC1* or approved laboratory fume hood controller. This specialized polarity insensitive system is independent of the standard EIA-485 communications network, and is intended to provide critical accumulated total exhaust flow information much more rapidly than typical EIA-485 network communications. If the accumulated total exhaust flow information from multiple *NB-FHC* controllers is not required, then use of this function is optional.

On each *NB-FHC*, an additional Analog Output designated with the termination labels of CFO and COM is used to transmit Cascade Flow. In turn, each *NB-FHC* contains an optically isolated digital Input, marked CF- and CF+ is used to receive the totalized cascade flow from other *NB-FHC* networked in series. An example network displayed below provides a clear visual representation of how Cascade Flow is networked between *NB-FHC* controllers.

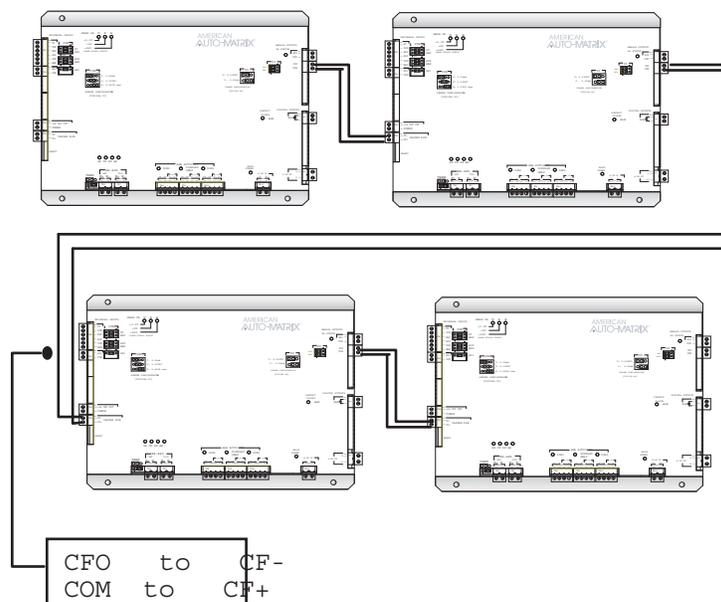


Figure 2-18 Cascade Flow Network Example

2.6 FHC-SD TOUCH SCREEN DISPLAY

The FHC-SD or Fume Hood Controller Small Display is the color touch screen display user interface to the FHC. It is typically mounted on the mullion of the fume hood. It communicates directly with the FHC, displaying system values and allows for operators of the fume hood to change various parameters (fume hood set points, emergency mode, alarm ranges, sensor constraints, etc.) of the FHC and the fume hood environment as needed.

NOTE



The FHC-SD is a one-to-one display, meaning each fumehood is required to have its own separate FHC and FHC-SD.

The installation of the FHC-SD involves mounting the product, supplying power, and connecting to the communications network. All wiring connections to the FHC-SD are made with the use of plug (female) & socket (male) terminal blocks (TB). The plug consists of terminal ports and 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 terminal block, 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 of the terminal block plugs are permanently attached to the PCB. Twisting or applying torque during connection revolutions may result in damage, voiding product warranty.

2.6.1 TOOLS REQUIRED

The following tools and supplied may be requires for installation of the FHC-SD:

- Torx screw driver: used to disassemble T-10 Torx screws from the FHC-SD housing
- Small flat-blade screw driver: used for EIA-485 and Power connections
- Wire strippers: for wiring purposes
- A meter: for verifying power connections

2.6.2 MOUNTING THE FHC-SD

The FHC-SD can be installed near the equipment it is monitoring. The installation site must provide 24VAC supply power. The temperature range for operation must be between 32 F and 104 F (0 C and 40 C). In addition, the mounting area should be free from moisture and/or leakage, and should be unobstructed by equipment or machinery. Before mounting the product, be sure to determine exactly where you wish to install the product. Figure 2-19 provides full dimension details of the unit.

The housing back-plate of the FHC-SD provides pre-drilled areas for different methods of mounting the product. Using the mounting plate, the FHC-SD can be mounted simply on a wall, on top of a US standard 2x4 junction box, or UK standard junction box.

The middle of the back plate contains an open area to accommodate the EIA-485 networking port and power port for the FHC-SD. If you are going to mount the FHC-SD onto a wall surface, be sure to have the appropriate tools in order to do so (such as a 2.5 inch hole saw).

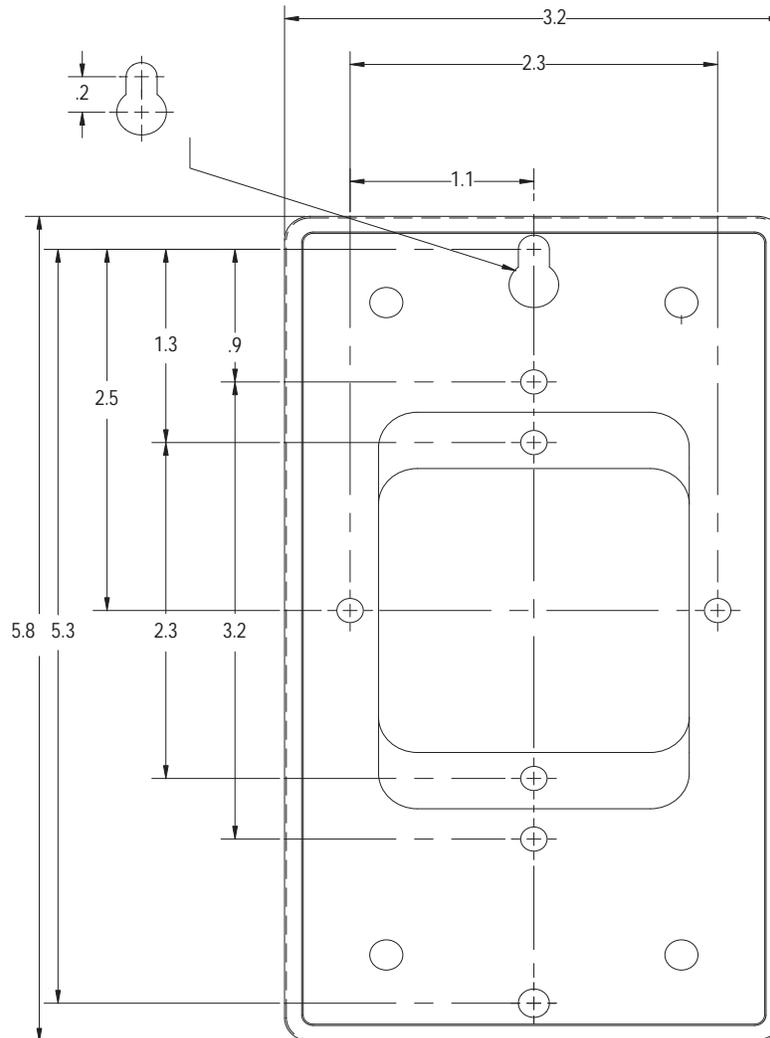


Figure 2-19 Mounting Dimensions of the FHC-SD

2.6.3 DISASSEMBLE THE FHC-SD

Prior to mounting, you will need to disassemble the FHC-SD. To do this, you will need a Torx screw driver that accommodates size T-10 screws. To disassemble the FHC-SD, perform the following steps:

1. Carefully loosen the four (4) T-10 Torx Drive screws located at the front of the FHC-SD and remove the cover.

WARNING

The printed circuit board and touch-screen hardware is assembled in a unconnected (“floating”) manner. Use care during assembly cycles.

2. Remove PCB component set of the FHC-SD from housing (if still installed). Older units may require screw posts to be removed from the top of the PCB in order to achieve this. Screw posts can be loosened using an appropriate tool.
3. Retain all screws and housing components for later reassembly.

2.6.4 MOUNTING THE BACKPLATE

The backplate of the FHC-SD’s housing can accommodate direct wall mounting, or mounting applications on electrical junction boxes. The backplate includes pre-formed screw holes that will align with drill holes located on the junction box. These screw holes are located near the middle of the backplate, near the cut out areas for power and network wiring.

For junction box mounting, use included screws from the junction box to mount the backplate.

For direct wall mounting, a keyhole-style screw hole is located on the outer left side of the backplate. The outer right side of the backplate contains a standard screw hole. The middle of the backplate contains an open area that accommodates network and power connections. To mount the backplate onto a wall, you may draw trace lines of the backplate prior to drilling.

2.6.5 CONNECTING THE FHC-SD TO THE NB-FHC

Connecting the FHC-SD to the NB-FHC requires a cable run between the two devices. The cable specification requires 4-wire, 18ga. twisted pair. The connection will carry both power and communications from the NB-FHC to the FHC-SD.

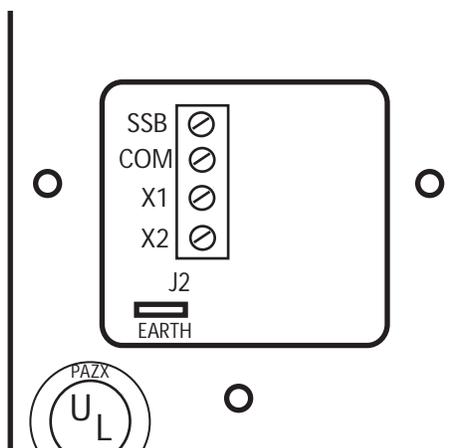


Figure 2-20 STATbus Network connection of FHC-SD

SECTION 3: APPLICATION CONFIGURATION

This section describes the setup for each application type that the NB-FHC supports, as well as extended features of the NB-FHC controller. Detailed programming notes are provided in this section for each application type.

IN THIS SECTION

| | |
|--|------|
| Control Mode Overview..... | 3-3 |
| Control Mode 20: AutoFlow With PID/Fuzzy Control | 3-5 |
| Control Mode 21: AutoFlow With PID/Fuzzy Control & Exhaust Flow Feedback | 3-7 |
| Control Mode 22: AutoFlow with Feed Forward Control | 3-9 |
| Control Mode 23: AutoFlow With Feed Forward Control & Exhaust Flow Feedback..... | 3-11 |
| Control Mode 30: Sash Position (Open Loop) | 3-14 |
| Control Mode 31: Sash Position With Shaft Pot Feedback..... | 3-16 |
| Emergency Mode | 3-20 |
| Cascade Flow | 3-21 |
| Cascade Flow Reliability | 3-22 |
| Occupancy Control..... | 3-23 |
| Alarm and Event Notifications | 3-24 |
| Advanced Loop Tuning | 3-25 |

3.1 CONTROL MODE OVERVIEW

There are two major control sequences in the NB-FHC, Auto-Flow method and Sash Position Control method.

The Auto-Flow method is based off measuring the actual face velocity across the plane of a sash using a pitot tube sensor, then controlling the exhaust output damper to regulate airflow for the fume hood. The Auto-Flow method has four different variations for control:

- . Auto-Flow with PID/Fuzzy Control
- . Auto-Flow with PID/Fuzzy Control and Exhaust Flow Feedback
- . Auto-Flow with Feed Forward Control
- . Auto-Flow with Feed Forward Control and Exhaust Flow Feedback

The Sash Position Method differs from Auto-Flow, as control of the exhaust output damper is achieved through measuring the position of sash opening. This method also has three different variations for control:

- . Sash Position Open Loop
- . Sash Position with Shaft Position Feedback
- . Sash Position with Exhaust Flow Feedback

3.2 CONTROL SEQUENCE CONFIGURATION

Various control methods are available for any given hood, based on the hardware compliment that is physically attached. Most control methods are closed loop, meaning that they actively sense the face velocity and make changes based on changes in the system. Several control methods require a calibrated venture valve and allow for open loop control if that is what is requested.

3.2.1 CONTROL METHOD OVERVIEW

Deciding on a control method can be made easier by looking at the following chart. Once the method has been chosen and the hardware has been attached, the FHC can be set up to make use of that particular I/O compliment in a matter of minutes.

| Name | Mode # | AutoFlow Sensor (FVR 1A or FVR 1B) | Sash String Pot. | Damper | Flow Switch (AVP) | Hood Presence Input (STATbus) | Exhaust Pitot Input |
|---|--------|------------------------------------|------------------|--------------------------------------|-------------------|-------------------------------|---------------------|
| AutoFlow With PID/Fuzzy Control | 20 | Yes | No | Any | Any | No | No |
| AutoFlow With PID/Fuzzy Control & Exhaust Flow Feedback | 21 | Yes | No | Any | Any | No | Yes |
| AutoFlow With Feed Forward Control | 22 | Yes | Yes | Any | Any | No | No |
| AutoFlow With Feed Forward Control & Exhaust Flow Feedback. | 23 | Yes | Yes | Any | Any | No | Yes |
| Sash Position (Open Loop) | 30 | No | Yes | Venturi Valve | Yes | No | No |
| Sash Position with Shaft Pot Feedback | 31 | No | Yes | Venturi Valve -w- Shaft Pot Feedback | Yes | No | No |
| Sash Position with Exhaust Flow Feedback | 32 | No | Yes | Any | Any | No | Yes |

Figure 3-1 FHC Control Modes

3.3 CONTROL MODE 20: AUTOFLOW WITH PID/FUZZY CONTROL

The standard Auto-Flow control methodology is designed to operate using traditional closed PID loop control, measured by the actual face velocity. The following table provides setup parameter information for configuring your unit for the Auto-Flow application profile.

| | |
|----------------------|-----|
| Auto Flow Sensor: | Yes |
| String Pot (Sash): | No |
| Damper: | Any |
| Flow Switch (AVP): | Any |
| Hood Presence Input: | No |
| Exhaust Pitot Input: | No |

3.3.1 SETUP HARDWARE

Attach the AutoFlow Sensor to 24Vdc (TB4-19) and attach the return line to input #1, (TB1-1). Set the IVR jumper to I for current.

Attach Actuator to EXD output and set the jumper appropriately. (Voltage most likely).

3.3.2 SETUP FIRMWARE

Default the controller into mode 20 using the settings listed above. This requires setting the “Modify Hardware Configuration” variable to “Allow Changes Below.” Then once the appropriate hardware compliment has been set the variable needs to be modified once more to “Commit To Changes.” This will save the information to the FHC, default its values in the new mode and cause it and the FHC-SD to reboot.

Object Path: Initial Configuration -> Hardware Configuration

Once the Unit has been rebooted, make sure to re-discover the object list of the controller in NB-Pro. This can be accomplished by restarting NB-Pro or by clicking on “Discovery”, then “Discover Object List” followed by selecting the FHC on question. Failure to rediscover the unit in NB-Pro will result in objects being invisible to NB-Pro.

Type in the appropriate occupied setpoint and unoccupied setpoint information in the Setpoint object. Defaults are 100 and 60 meaning that when occupied the hood will target 100 feet per minute and when unoccupied it will target 60 feet per minute.

Object Path: Control -> Face Velocity Setpoint

In the Face Velocity Sensor Curve Object choose FVR-1A, FVR-1B or HWA-1 in the “(AC) Automatic Stuffing” property to fill in the curve. If a different sensor is being used its curve can be manually entered into the X & Y points.

Object Path: Sensor Management-> Face Velocity Sensor Curve

The Exhaust Damper is set up to use a linear output configuration.

By default the PID loop is turned on and stuffed with some generic values.

Channel Path: Damper Output ->EXD

3.3.3 TUNING & CALIBRATION

Follow standard PID or Fuzzy logic tuning procedures to get the hood operating as desired.

Follow standard Face Velocity Calibration techniques to measure the face velocity and type the coefficient into AV-5, Sensor Coefficient.

The optional digital input (AVP) can be mapped in to disable/enable control or to set the occupancy for the unit. It is important to note that both of these functions can be set to receive broadcasts from other FHC's in order to share inputs. If this unit is one with a sensor attached to the AVP input terminal, the meaning of this switch can be mapped to either purpose. If you want the hood to be disabled when this switch is occupied, you can map it into the "Disable Control" object via the property "TI". If instead you choose to use this switch to determine occupancy, map the switch into the "Hood Presence Detector" via the "TI" property found there.

Object Path: Control -> Disable Control OR Occupancy -> Hood Presence Detector

Face Velocity alarms can be configured following the procedures indicated in the Alarming section of this manual.

3.4 CONTROL MODE 21: AUTOFLOW WITH PID/FUZZY CONTROL & EXHAUST FLOW FEEDBACK

This mode uses active sensor measurements to determine the hood's face velocity. The exhaust damper is modulated based on the target face velocity using either a PID loop or a Fuzzy-PID algorithm. Exhaust Flow Data is also collected which can be used for alarming purposes and room pressurization purposes.

| | |
|----------------------|-----|
| Auto Flow Sensor: | Yes |
| String Pot (Sash): | No |
| Damper: | Any |
| Flow Switch (AVP): | Any |
| Hood Presence Input: | No |
| Exhaust Pitot Input: | Yes |

3.4.1 SETUP HARDWARE

Attach the AutoFlow sensor to 24Vdc (TB4-19) and attach the return line to the FV input, (TB1-1). Set the IVR jumper to I for current.

Attach Actuator to EXD output and set the jumper appropriately. (Voltage most likely).

Attach the exhaust flow sensor to 24Vdc (TB4-19) and attach the return line to the EXF input, (TB1-3). Set the IVR jumper to I for current.

3.4.2 SETUP FIRMWARE

Default the controller into mode 21 using the settings listed above. This requires setting the "Modify Hardware Configuration" variable to "Allow Changes Below." Then once the appropriate hardware compliment has been set the variable needs to be modified once more to "Commit To Changes." This will save the information to the FHC, default its values in the new mode and cause it and the FHC-SD to reboot.

Object Path: Initial Configuration -> Hardware Configuration

Once the Unit has been rebooted, make sure to re-discover the object list of the controller in NB-Pro. This can be accomplished by restarting NB-Pro or by clicking on "Discovery", then "Discover Object List" followed by selecting the FHC on question. Failure to rediscover the unit in NB-Pro will result in objects being invisible to NB-Pro.

Type in the appropriate occupied setpoint and unoccupied setpoint information in the Setpoint Object. Defaults are 100 and 60 meaning that when occupied the hood will target 100 feet per minute and when unoccupied it will target 60 feet per minute.

Object Path: Control -> Face Velocity Setpoint

In the Face Velocity Sensor Curve Object choose FVR-1A, FVR-1B or HWA-1 in the "(AC) Automatic Stuffing" property to fill in the curve. If a different sensor is being used its curve can be manually entered into the X & Y points.

CONTROL MODE 21: AUTOFLOW WITH PID/FUZZY CONTROL & EXHAUST FLOW

Object Path: Sensor Management -> Face Velocity Sensor Curve

The Exhaust Flow sensor curve needs to be set up for the size and type of sensor you have selected. If using a cross pitot with a differential pressure transmitter there are a number of automatic configuration options in the "(AC) Automatic Stuffing" properties drop down box. If a different sensor is being used its curve can be manually entered into the X & Y points.

Object Path: Sensor Management -> Exhaust Flow Sensor Curve

The Exhaust Damper is set up to use a linear output configuration. In the event that you need to limit the damper or if you choose to change its output characteristics, the scaling type can be changed from linear to "Curve 3." Once this has been set, the non-linear data can be entered into curve 3 for the damper actuator.

Object Path: Outputs -> Exhaust Damper Output EXD

By default the PID loop is turned on and stuffed with some generic values.

Object Path: Outputs -> Exhaust Damper Output EXD

Tuning & Calibration:

Follow standard PID or Fuzzy logic tuning procedures to get the hood operating as desired.

Follow standard Face Velocity Calibration techniques to measure the face velocity and type the coefficient into AV-5, Sensor Coefficient.

The optional digital input (AVP) is automatically mapped in when present to disable/enable control or to set the occupancy for the unit. It is important to note that both of these functions can be set to receive broadcasts from other FHC's in order to share inputs. If this unit is one with a sensor attached to the AVP input terminal, the meaning of this switch can be mapped to either purpose. If you want the hood to be disabled when this switch is occupied, you can map it into the "Disable Control" object via the property "TI." If instead you choose to use this switch to determine occupancy, map the switch into the "Hood Presence Detector" via the "TI" property found there.

Object Path: Control -> Disable Control OR Occupancy -> Hood Presence Detector

Face Velocity alarms can be configured following the procedures indicated in the Alarming section of this manual.

3.5 CONTROL MODE 22: AUTOFLOW WITH FEED FORWARD CONTROL

This mode uses active sensor measurements to determine the hood's face velocity. The exhaust damper is modulated based on the target face velocity using a PID control loop. In addition, a potentiometer is attached to the hood's sash allowing the control algorithm to use sash area data to speed up the response of the PID loop.

| | |
|----------------------|-----|
| Auto Flow Sensor: | Yes |
| String Pot (Sash): | Yes |
| Damper: | Any |
| Flow Switch (AVP): | Any |
| Hood Presence Input: | No |
| Exhaust Pitot Input: | No |

3.5.1 SETUP HARDWARE:

Attach the AutoFlow Sensor to 24Vdc (TB4-19) and attach the return line to input #1, (TB1-1). Set the IVR jumper to I for current.

Attach Actuator to EXD output and set the jumper appropriately. (Voltage most likely).

Attach the sash position potentiometer between input SP1 and common, (TB1-7 & TB1-8). Set the IVR jumper to R for resistance.

3.5.2 SETUP FIRMWARE

Default the controller into mode 22 using the settings listed above. This requires setting the "Modify Hardware Configuration" variable to "Allow Changes Below." Then once the appropriate hardware compliment has been set the variable needs to be modified once more to "Commit To Changes." This will save the information to the FHC, default its values in the new mode and cause it and the FHC-SD to reboot.

Object Path: Initial Configuration -> Hardware Configuration

Once the Unit has been rebooted, make sure to re-discover the object list of the controller in NB-Pro. This can be accomplished by restarting NB-Pro or by clicking on "Discovery", then "Discover Object List" followed by selecting the FHC on question. Failure to rediscover the unit in NB-Pro will result in objects being invisible to NB-Pro.

Type in the appropriate occupied setpoint and unoccupied setpoint information in the Setpoint Channel. Defaults are 100 and 60 meaning that when occupied the hood will target 100 feet per minute and when unoccupied it will target 60 feet per minute.

Object Path: Control -> Face Velocity Setpoint

In the Face Velocity Sensor Curve Object choose FVR-1A, FVR-1B or HWA-1 in the "(AC) Automatic Stuffing" property to fill in the curve. If a different sensor is being used its curve can be manually entered into the X & Y points.

Object Path: Sensor Management -> face Velocity Sensor Curve

The Exhaust Damper is set up to use a linear output configuration. In the event that you need to limit the damper or if you choose to change its output characteristics, the scaling type can be changed from linear to "Curve 3." Once this has been set, the non-linear data can be entered into curve 3 for the damper actuator.

Object Path: Outputs -> Exhaust Damper Output EXD

Set up the Sash Geometry Object, either via the FHC-SD display or via the PUP network.

Object Path: Initial Configuration -> Sash Geometry

By default the PID loop is turned on and stuffed with some generic values.

Object Path: Outputs -> Exhaust Damper Output EXD

Tuning & Calibration:

The feed forward portion of the control is disabled until a variable is written to the PID loop's "Feed-Forward Gain" property. This value is calculated using data from Sash Geometry and Exhaust Valve Output objects. Specifically, the hood needs to be at steady state at two points (likely with the sash fully closed and again with it fully open). The Feed Forward Gain is calculated using the following equation: $FF = ((\text{Percent Damper Output @ Max Open} - \text{Percent Damper Output @ Min Open}) / (\text{Sash Area in Sq Feet/Meters @ Max Open} - \text{Sash Area in Sq Feet/Meters @ Min Open}))$. Once this variable has been set, the PID loop will use the feed-forward control when it is in Either PID or Fuzzy Logic mode.

Follow standard PID or Fuzzy logic tuning procedures to get the hood operating as desired.

Follow standard Face Velocity Calibration techniques to measure the face velocity and type the coefficient into AV-5, Sensor Coefficient.

The optional digital input (AVP) is automatically mapped in when present to disable/enable control or to set the occupancy for the unit. It is important to note that both of these functions can be set to receive broadcasts from other FHC's in order to share inputs. If this unit is one with a sensor attached to the AVP input terminal, the meaning of this switch can be mapped to either purpose. If you want the hood to be disabled when this switch is occupied, you can map it into the "Disable Control" channel via the attribute "TI." If instead you choose to use this switch to determine occupancy, map the switch into the "Hood Presence Detector" via the "TI" attribute found there.

Channel Path: Occupancy-> Disable Control OR Occupancy-> Hood Presence Detector

Face Velocity alarms can be configured following the procedures indicated in the Alarming section of this manual.

3.6 CONTROL MODE 23: AUTOFLOW WITH FEED FORWARD CONTROL & EXHAUST FLOW FEEDBACK

This mode uses active sensor measurements to determine the hood's face velocity. The exhaust damper is modulated based on the target face velocity using a PID control loop. In addition, a potentiometer is attached to the hood's sash allowing the control algorithm to use sash area data to speed up the response of the PID loop. Exhaust Flow Data is also collected which can be used for alarming purposes and room pressurization purposes.

| | |
|----------------------|-----|
| Auto Flow Sensor: | Yes |
| String Pot (Sash): | Yes |
| Damper: | Any |
| Flow Switch (AVP): | Any |
| Hood Presence Input: | No |
| Exhaust Pitot Input: | Yes |

3.6.1 SETUP HARDWARE

Attach the AutoFlow Sensor to 24Vdc (TB4-19) and attach the return line to input #1, (TB1-1). Set the IVR jumper to I for current.

Attach Actuator to EXD output and set the jumper appropriately. (Voltage most likely).

Attach the sash position potentiometer between input SP1 and common, (TB1-7 & TB1-8). Set the IVR jumper to R for resistance.

Attach the exhaust flow sensor to 24Vdc (TB4-19) and attach the return line to the EXF input, (TB1-3). Set the IVR jumper to I for current.

3.6.2 SETUP FIRMWARE

Default the controller into mode 23 using the settings listed above. This requires setting the "Modify Hardware Configuration" variable to "Allow Changes Below." Then once the appropriate hardware compliment has been set the variable needs to be modified once more to "Commit To Changes." This will save the information to the FHC, default its values in the new mode and cause it and the FHC-SD to reboot.

Object Path: Initial Configuration -> Hardware Configuration

Once the Unit has been rebooted, make sure to re-discover the object list of the controller in NB-Pro. This can be accomplished by restarting NB-Pro or by clicking on "Discovery", then "Discover Object List" followed by selecting the FHC on question. Failure to rediscover the unit in NB-Pro will result in objects being invisible to NB-Pro.

Type in the appropriate occupied setpoint and unoccupied setpoint information in the Setpoint Object. Defaults are 100 and 60 meaning that when occupied the hood will target 100 feet per minute and when unoccupied it will target 60 feet per minute.

CONTROL MODE 23: AUTOFLOW WITH FEED FORWARD CONTROL & EXHAUST

Object Path: Control -> Face Velocity Setpoint

In the Face Velocity Sensor Curve Object choose FVR-1A, FVR-1B or HWA-1 in the "(AC) Automatic Stuffing" property to fill in the curve. If a different sensor is being used its curve can be manually entered into the X & Y points.

Object Path: Sensor Management-> Face Velocity Sensor Curve

The Exhaust Flow sensor curve needs to be set up for the size and type of sensor you have selected. If using a cross pitot with a differential pressure transmitter there are a number of automatic configuration options in the "(AC) Automatic Stuffing" property's drop down box. If a different sensor is being used its curve can be manually entered into the X & Y points.

Object Path: Sensor Management -> Exhaust Flow Sensor Curve

The Exhaust Damper is set up to use a linear output configuration. In the event that you need to limit the damper or if you choose to change its output characteristics, the scaling type can be changed from linear to "Curve 3." Once this has been set, the non-linear data can be entered into curve 3 for the damper actuator.

Object Path: Outputs -> Exhaust Damper Output EXD

Set up the Sash Geometry Object, either via the FHC-SD display or via the MS/TP network.

Object Path: Initial Configuration ->Sash Geometry

By default the PID loop is turned on and stuffed with some generic values.

Object Path: Outputs-> Exhaust Damper Output EXD

3.6.3 TUNING & CALIBRATION

The feed forward portion of the control is disabled until a variable is written to the PID loop's "Feed-Forward Gain" property. This value is calculated using data from Sash Geometry and Exhaust Valve Output objects. Specifically, the hood needs to be at steady state at two points (likely with the sash fully closed and again with it fully open). The Feed Forward Gain is calculated using the following equation: $FF = ((\text{Percent Damper Output @ Max Open} - \text{Percent Damper Output @ Min Open}) / (\text{Sash Area in Sq Feet/ Meters @ Max Open} - \text{Sash Area in Sq Feet/Meters @ Min Open}))$. Once this variable has been set, the PID loop will use the feed-forward control when it is in Either PID or Fuzzy Logic mode.

Follow standard PID or Fuzzy logic tuning procedures to get the hood operating as desired.

Follow standard Face Velocity Calibration techniques to measure the face velocity and type the coefficient into AV-5, Sensor Coefficient.

The optional digital input (AVP) is automatically mapped in when present to disable/enable control or to set the occupancy for the unit. It is important to note that both of these functions can be set to receive broadcasts from other FHC's in order to share inputs. If this unit is one with a sensor attached to the AVP

input terminal, the meaning of this switch can be mapped to either purpose. If you want the hood to be disabled when this switch is occupied, you can map it into the “Disable Control” object via the property “TI”. If instead you choose to use this switch to determine occupancy, map the switch into the “Hood Presence Detector” object via the “TI” property found there.

Object Path: Control> Disable Control OR Occupancy -> Hood Presence Detector

Face Velocity alarms can be configured following the procedures indicated in the Alarming section of this manual.

3.7 CONTROL MODE 30: SASH POSITION (OPEN LOOP)

This mode uses sash position to calculate the open area of the hood. This information is used to determine the appropriate target exhaust flow which is then sent to the calibrated Venturi valve. The AVP input is used to ensure that the exhaust flow is great enough to support the Venturi valve's operation. When the AVP flow switch indicates low or no flow, the hood disables its control.

| | |
|----------------------|---------------|
| Auto Flow Sensor: | No |
| String Pot (Sash): | Yes |
| Damper: | Venturi valve |
| Flow Switch (AVP): | Yes |
| Hood Presence Input: | No |
| Exhaust Pitot Input: | No |

3.7.1 SETUP HARDWARE

Attach the sash position potentiometer between input SP1 and common, (TB1-7 & TB1-8). Set the IVR jumper to R for resistance.

Attach Actuator to EXD output and set the jumper appropriately. (Voltage most likely).

Attach the flow proving switch to the AVP input (TB1-4 & TB1-5) and set the IVR jumper to R for resistance.

3.7.2 SETUP FIRMWARE

Default the controller into mode 30 using the settings listed above. This requires setting the "Modify Hardware Configuration" variable to "Allow Changes Below." Then once the appropriate hardware compliment has been set the variable needs to be modified once more to "Commit To Changes." This will save the information to the FHC, default its values in the new mode and cause it and the FHC-SD to reboot.

Object Path: Setup -> Hardware Configuration

Once the Unit has been rebooted, make sure to re-discover the object list of the controller in NB-Pro. This can be accomplished by restarting NB-Pro or by clicking on "Discovery", then "Discover Object List" followed by selecting the FHC on question. Failure to rediscover the unit in NB-Pro will result in objects being invisible to NB-Pro.

Type in the appropriate occupied setpoint and unoccupied setpoint information in the Setpoint Object. Defaults are 100 and 60 meaning that when occupied the hood will target 100 feet per minute and when unoccupied it will target 60 feet per minute.

Object Path: Control -> Face Velocity Setpoint

The Exhaust Damper is set up by default to use curve 3. For this required curve, one must put the CFM of the actuator as the engineering units and the % of actuator travel for the full scale output. In this way, a

linear damper calibrated from 500 to 1500 CFM would have 0 to 100 in the X values and 500 to 1500 spread out evenly for the Y values. Non-linear actuators can also be set up in the very same way.

Object Path: Sensor Management -> Exhaust Damper Output Curve

Set up the Sash Geometry Object, either via the FHC-SD display or via the MS/TP network.

Object Path: Initial Configuration -> Sash Geometry

By default the PID loop is turned on and stuffed with some generic values.

Object Path: Outputs -> Exhaust Damper Output EXD

3.7.3 TUNING & CALIBRATION

Follow standard Face Velocity Calibration techniques to measure the face velocity and type the coefficient into AV-5, Sensor Coefficient.

In order for the hood to be turned off when the lab is not in use, the “disable control” input can be set up to track a broadcast from another FHC, it can be manually controlled via PUP or it can track the AVP input. The drop-down options in the property “TI” will allow each hood to select how this is controlled. Please note that in this control mode, the actual face velocity is not being measured so the AVP input must be used to signal that there is enough pressure across the valve for it to operate properly.

Object Path: Control -> Disable Control

3.8 CONTROL MODE 31: SASH POSITION WITH SHAFT POT FEEDBACK

This mode uses sash position to calculate the open area of the hood. This information is used to determine the appropriate exhaust flow which is then sent to the calibrated Venturi valve. Feedback from the valve's shaft position is used to determine the current exhaust flow. The AVP input is used to ensure that the exhaust flow is great enough to support the Venturi valve's operation. When the AVP flow switch indicates low or no flow, the hood disables its control.

| | |
|----------------------|--------------------------------------|
| Auto Flow Sensor: | No |
| String Pot (Sash): | Yes |
| Damper: | Venturi valve -w- Shaft Pot Feedback |
| Flow Switch (AVP): | Yes |
| Hood Presence Input: | No |
| Exhaust Pitot Input: | No |

3.8.1 SETUP HARDWARE

Attach the sash position potentiometer between input SP1 and common, (TB1-7 & TB1-8). Set the IVR jumper to R for resistance.

Attach the Damper's Shaft Pot to the EXF input (TB1-2 & TB1-3) and set the IVR jumper to R for resistance.

Attach Actuator to EXD output and set the jumper appropriately. (Voltage most likely).

Attach the flow proving switch to the AVP input (TB1-4 & TB1-5) and set the IVR jumper to R for resistance.

3.8.2 SETUP FIRMWARE

Default the controller into mode 31 using the settings listed above. This requires setting the "Modify Hardware Configuration" variable to "Allow Changes Below." Then once the appropriate hardware compliment has been set the variable needs to be modified once more to "Commit To Changes." This will save the information to the FHC, default its values in the new mode and cause it and the FHC-SD to reboot.

Object Path: Initial Configuration -> Hardware Configuration

Once the Unit has been rebooted, make sure to re-discover the object list of the controller in NB-Pro. This can be accomplished by restarting NB-Pro or by clicking on "Discovery", then "Discover Object List" followed by selecting the FHC on question. Failure to rediscover the unit in NB-Pro will result in objects being invisible to NB-Pro.

Type in the appropriate occupied setpoint and unoccupied setpoint information in the Setpoint Object. Defaults are 100 and 60 meaning that when occupied the hood will target 100 feet per minute and when unoccupied it target 60 feet per minute.

Object Path: Control -> Face Velocity Setpoint

The Exhaust Flow is determined by the feedback from the shaft potentiometer. This uses Piecewise Curve 2, "Exhaust Valve Shaft Pot Curve" to determine the exhaust flow. This curve should be set up to indicate the flow vs. resistance for the shaft pot on hand.

The "Exhaust Valve Shaft Pot Curve" needs to be filled in with data specifying the relationship between the shaft pot resistance and the associated exhaust flow. The actual data entered is the percent of full scale resistance and the corresponding exhaust flow in CFM. Calculating the "full scale resistance value is accomplished via the following equation:

$$\% \text{ Full Scale Resistance} = 25 * ((\text{Resistance } (\Omega)) / (\text{Resistance } (\Omega) + 20000))$$

As an example, if a resistance of 67,600 Ω corresponded to 465 CFM, the "Engineering Units" entry would be 465 and the "% Full Scale" value would be 19.292 which is the result of placing the value 67,600 into the equation above.

Object Path: Sensor Management -> Exhaust Valve Shaft Pot Curve

Set up the Sash Geometry Object, either via the FHC-SD display or via the PUP network.

Object Path: Initial Configuration -> Sash Geometry

By default the PID loop is turned on and stuffed with some generic values.

Channel Path: Outputs -> Exhaust Damper Output EXD

Tuning & Calibration:

Follow standard PID or Fuzzy logic tuning procedures to get the hood operating as desired.

Follow Standard Calibration techniques to measure the face velocity and type the coefficient into AV-5, Sensor Coefficient.

In order for the hood to be turned off when the lab is not in use, the "Disable Control" input can be set up to track a broadcast from another FHC, it can be manually controlled via PUP or it can track the AVP input. The drop-down options in the property "TI" will allow each hood to select how this is controlled. Please note that in this control mode, the actual face velocity is not being measured so the AVP input must be used to signal that there is enough pressure across the valve for it to operate properly.

Object Path: Control -> Disable Control

3.9 CONTROL MODE 32: SASH POSITION WITH EXHAUST FLOW FEEDBACK

This mode uses sash position to calculate the open area of the hood. This information is used to determine the appropriate exhaust flow which is then sent to the exhaust damper. Feedback from the measured exhaust flow is used to close the control loop.

| | |
|----------------------|-----|
| Auto Flow Sensor: | No |
| String Pot (Sash): | Yes |
| Damper: | Any |
| Flow Switch (AVP): | Any |
| Hood Presence Input: | No |
| Exhaust Pitot Input: | Yes |

3.9.1 SETUP HARDWARE:

Attach the sash position potentiometer between input SP1 and common, (TB1-7 & TB1-8). Set the IVR jumper to R for resistance.

Attach the exhaust flow sensor to 24Vdc (TB4-19) and attach the return line to the EXF input, (TB1-3). Set the IVR jumper to I for current.

Attach Actuator to EXD output and set the jumper appropriately. (Voltage most likely).

3.9.2 SETUP FIRMWARE

Default the controller into mode 32 using the settings listed above. This requires setting the "Modify Hardware Configuration" variable to "Allow Changes Below." Then once the appropriate hardware compliment has been set the variable needs to be modified once more to "Commit To Changes." This will save the information to the FHC, default its values in the new mode and cause it and the FHC-SD to reboot.

Object Path: Initial Configuration -> Hardware Configuration

Once the Unit has been rebooted, make sure to re-discover the object list of the controller in NB-Pro. This can be accomplished by restarting NB-Pro or by clicking on "Discovery", then "Discover Object List" followed by selecting the FHC on question. Failure to rediscover the unit in NB-Pro will result in objects being invisible to NB-Pro.

Type in the appropriate occupied setpoint and unoccupied setpoint information in the Setpoint Object. Defaults are 100 and 60 meaning that when occupied the hood will target 100 feet per minute and when unoccupied it will target 60 feet per minute.

Object Path: Control -> Face Velocity Setpoint

The Exhaust Flow sensor curve needs to be set up for the size and type of sensor you have selected. If using a cross pitot with a differential pressure transmitter there are a number of automatic configuration options in the "(AC) Automatic Stuffing" property's drop down box. If a different sensor is being used its curve can be manually entered into the X & Y points.

Object Path: Sensor Management -> Exhaust Flow Sensor Curve

Set up the Sash Geometry Object, either via the FHC-SD display or via the MS/TP network.

Object Path: Initial Configuration -> Sash Geometry

By default the PID loop is turned on and stuffed with some generic values.

Object Path: Outputs -> Exhaust Damper Output EXD

3.9.3 TUNING & CALIBRATION

Follow standard PID or Fuzzy logic tuning procedures to get the hood operating as desired.

Follow Standard Face Velocity Calibration techniques to measure the face velocity and type the coefficient into the Setpoint Channel's "Sensor Coefficient" attribute.

3.10 EMERGENCY MODE

The Emergency Mode capability of the NB-FHC is used to purge air in a hood in the event of a critical situation or event. Emergency Mode is triggered by any one of the following actions:

- . User touches the Emergency Icon on the FHC-SD touch screen display
- . An Emergency Input sensor connected to Universal Input 4 is triggered
- . The NB-FHC receives an emergency broadcast via the object Broadcast 7
- . Low Air Differential Pressure connected to Digital Input 3 (Sash Position applications only)

Emergency mode operation ceases when a user writes to the Reset Emergency Mode (RE) property, located in the Touchscreen Display object.

Table 3-1 Emergency Mode Configuration Setup

| Step | Action | Procedure Details |
|------|---|---|
| 1 | Configure Emergency Input - Universal Input 4 | <ol style="list-style-type: none"> 1. Set (ST) Sensor Type = Digital (value of 0) - this should be the default sensor type for Universal Input 4. If it is currently set for an Analog sensor type, change it to Digital, then reset the NB-FHC either manually or through software and rediscover its objects. 2. Configure the appropriate polarity mode for your input switch sensor through polarity. 3. Verify out-of-service = False (value of 0) |
| 2 | Configure Exhaust Damper Output - Analog Output 1 | <ol style="list-style-type: none"> 1. Verify that Priority Array Index 2 is currently set to NULL to allow Emergency Mode services to function properly. 2. Configure (EP) Emergency Position to the desired emergency position for the output. The default value is 100.00 for 100% open. 3. Verify out-of-service = False (value of 0) |
| 3 | Configure External Alarm - Digital Output 1 | <ol style="list-style-type: none"> 1. Verify that Priority Array Index 2 is currently set to NULL to allow Emergency Mode services to function properly. 2. Verify out-of-service = False (value of 0) |

3.11 CASCADE FLOW

The NB-FHC provides terminal connections for a cascade flow network. The cascade flow network permits a series of NB-FHC controllers to be connected together along a single pulse-width modulated network. The network initially begins with a unit transmitting exhaust flow. The next unit networked, receives the flow, then adds its own exhaust flow, then transmitting onto the next successive NB-FHC controller. The final flow can be sent to a main NB-FHC or approved laboratory fume hood controller for lab control applications. This specialized system is independent of the standard EIA-485 communications network, and is intended to provide critical accumulated total exhaust flow information much more rapidly than typical EIA-485 network communications. If the accumulated total exhaust flow information from multiple NB-FHC controllers is not request, then use of this function is optional. Network wiring details for this function can be found in Section 2 of this manual.

On the NB-FHC controller, the Cascade Flow transmitter is designated as Analog Output 2 and is located directly beneath the designated Analog Output for Exhaust Damper Control. The transmitted flow is received in by another NB-FHC at the Cascade Flow network receiver, an on-board Digital Input located directly beneath the NB-FHC's courtesy 24VDC power output.

Table 3-2 Cascade Flow Configuration Setup

| Step | Action | Procedure Details |
|------|---|--|
| 1 | Configure Analog Output 2 (Cascade Exhaust Flow) Cascade Flow Transmitter | <ol style="list-style-type: none"> 1. Verify that the jumper configuration for the Cascade Flow transmitter located on Analog Output 2 is configured for voltage mode (0-10VDC). 2. Configure (MN) Minimum Scaled Value and (MX) Maximum Scaled Value for the anticipated minimum and maximum totalized flow readings for your laboratory environment. The total flow value is scaled between the configured (MN) Minimum Scaled Value and (MX) Maximum Scaled Value parameters. (MN) can be as low a zero (0). (MX) can be as high as 32,767. 3. Verify out-of-service = False (value of 0) <p>Analog Output 2 will always transmit total exhaust flow, and display the current reading in its present-value property. The value transmitted will be the sum of flow measured on Universal Input 2 and the totalized cascade flow received on Digital Input 1.</p> |
| 2 | Configure Digital Input 1 (Cascade Flow) Cascade Flow Receiver | <ol style="list-style-type: none"> 1. Verify out-of-service = false (value of 0). 2. Set (MD) Pulse Counter Mode = Cascade Flow (value of 3). <p>Careful consideration must be taken to ensure that the first NB-FHC electrically networked in the Cascade Flow network not have it's Digital Input configured to receive a totalized flow reading. This ensures that the NB-FHC is not expecting to receive a cascade flow transmission from another NB-FHC controller.</p> <p>The total flow value received at Digital Input 1 will be stored in (SV) Scaled Pulse Count Value property.</p> |

3.11.1 CASCADE FLOW RELIABILITY

When using the Cascade Flow network, there are a some conditions that can make cascade flow values unreliable. The following table outlines these conditions and notes for the purpose of troubleshooting.

Table 3-3 Cascade Flow Conditions of Unreliability

| Condition | Cause | Notes |
|-----------|---------------------------------------|---|
| 1 | Bad Totalized Flow | The <i>NB-FHC</i> will indicate that a previous controller does not have a reliable value to transmit if a transmitted total flow equal to 32,767 has been received by a <i>NB-FHC</i> in series. This can occur if the exhaust flow reading from a previous <i>NB-FHC</i> is unreliable, or when then totalized flow that an <i>NB-FHC</i> has received from another device is unreliable. |
| 2 | Exhaust Flow Input is Unreliable | If the Exhaust Flow input value is unreliable, then the Cascade Flow output for the device will be marked as unreliable, causing the value of 32,767 to be sent to the next <i>NB-FHC</i> in series on the Cascade Flow network. |
| 3 | Cascade Network Communication Failure | If the <i>NB-FHC</i> is configured to receive a cascade flow and does not receive a value, then the Digital Input configured to receive Cascade Flow will be marked as unreliable, and a value of 32,767 will be sent to the next <i>NB-FHC</i> in series on the Cascade Flow network. |
| 4 | Cascade Network Errors | If the <i>NB-FHC</i> is receiving cascade flow from another controller and communication value is affected by errors due to improper wiring such as a short, the Digital Input will be marked as unreliable. This may only be a temporary condition and once a valid value has been receive, the Digital Input will then be marked as reliable. |

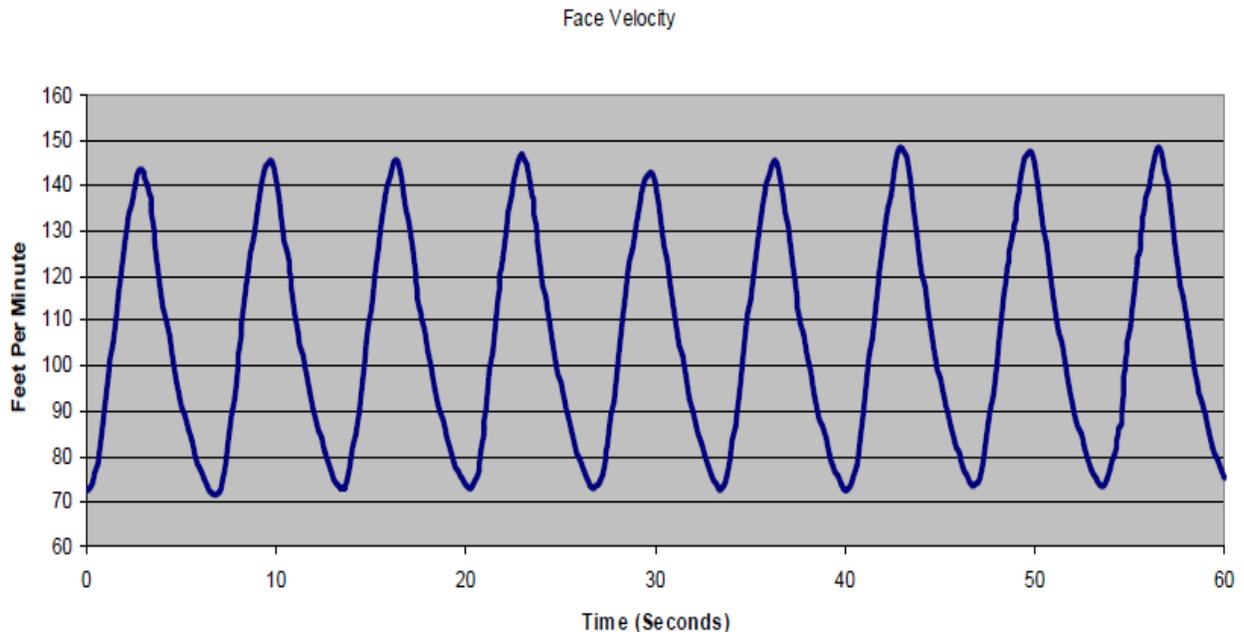
3.12 OCCUPANCY CONTROL

Occupancy Control is used to signal active occupancy of a fume hood. When the Hood Presence Detector Input Status reads a active (true) condition, occupancy will be assumed and the controller will operate within normal control parameters. When an inactive (false) condition is detected, the controller will operate in an unoccupied manner, where setup/setback parameters may be applied to alarm parameters and control setpoints, thereby reducing energy consumption.

By default, the Enable/Disable Hood Occupancy property “TI” is set to Forced to Occupied (1) allowing the unit to operate in a full-time occupied mode. The following provides configuration steps to configure Occupancy Control within the NB-FHC.

Table 3-4 Hood Presence Configuration Setup

| Step | Action | Procedure Notes |
|------|--|---|
| 1 | Configure Occupancy Control (Binary Value [5], Instance 1) | <p>Set (TI) Enable/Disable Hood Occupancy to the intended occupancy control mode. The following options are available:</p> <ul style="list-style-type: none"> • (0) Forced to unoccupied • (1) Forced to occupied • (2) Track STATbus Expansion Input • (3) Track AVP Input • (4) Track Broadcast Object • (5) Track Schedule Object |



3.13 ALARM AND EVENT NOTIFICATIONS

The NB-FHC contains one Notification Class object. The Notification Class object defines where alarm and event notifications are sent. Prior to configuring each object for a specific alarm application, the Notification Class object must be configured. The Notification Class object provides functionality for one recipient, defined by the Device Instance or MAC Address. Valid schedule times, process IDs, etc. are configured from the **recipient-list** property of this object.

Table 3-5 Notification Class Setup Configuration

| Step | Action | Procedure Notes |
|------|--|---|
| 1 | Defined Requirement of Acknowledgments - ack-required property | This property conveys three separate flags which represent whether an acknowledgment from a supervisory controller or operator workstation shall be required in notification generated. By default, the NB-FHC will require acknowledgments for alarms and events that transition TO-OFFNORMAL and TO-NORMAL. TO-FAULT conditions can require acknowledgements if desired. |
| 2 | Configure Recipient- recipient-list property | <ol style="list-style-type: none"> 1. Verify that the Recipient 1 check box has a check mark, thus enabling the recipient. 2. Select valid days in which notifications should be sent to the recipient, based on each day of the week. 3. Specify the from time and to time parameters of when notifications should be sent in correlation to the valid days selected. For 24 hour operation set from time = 12:00:00 AM, set to time = 23:59:59 4. Specify the process ID for events generated by the NB-FHC. This can be any number between 0 and 4,194,303. 5. Configure the recipient address as either an Object ID or MAC Address. 6. Specified whether notifications should be issued as confirmed by placing a check in the Issue Confirmed check box. 7. Specify the transitions that should be sent by placing a check in the appropriate transition boxes (TO-OFFNORMAL, TO-NORMAL, TO-FAULT). |
| 3 | Configure Object for Alarm Application - (AE) Alarm Enable property | <ol style="list-style-type: none"> 1. Configure your selected object for a valid alarm state by configuring (AE) Alarm Enable to a valid alarm application. 2. Depending on the alarm application selected, you will need to configure properties such as low-limit, high-limit, (LX) Low Extreme Limit, (HX) High Extreme Limit, and alarm-value. 3. Configure the event-enable flags to select which events you wish to enable the NB-FHC to send notifications for. |

3.14 ADVANCED LOOP TUNING

Fume hoods are generally required to have an airflow of 100 feet per minute across their open faces to safely contain air-borne chemicals, gases, particulates or pathogens. Changes to the system can immediately effect the amount of air flowing through the hood requiring a very fast response from the controller. Opening or closing the sash(es), walking or standing in front of the hood, or any change to the Lab's HVAC system, such as opening or closing doors or externally-caused changes in exhaust flows all directly effect the "face velocity" of the airflow.

Precisely controlling the face velocity is important not only for safety reasons but for cost considerations as well. In most applications 100% of the air through the fume hood has to be directed outside, meaning that every cubic foot of conditioned air through the hood needs to be replaced by fresh (newly conditioned) air. Quickly and accurately measuring the face velocity coupled with very fast changes to the fans or dampers allows for safe laboratory environments, less wasted air and lower energy costs.

3.14.1 SPEED CONSIDERATIONS

The speed at which a fume hood reacts to its settings is based entirely upon the I/O complement of the system. Some sensing technologies are faster at sending feedback to the controller than others. In addition, some methods of controlling the airflow are faster than others. It is for this reason that every fume hood will likely need to be properly tuned to operate as safely and efficiently as possible.

Things that affect the speed of a fume hood system:

- . The speed at which the sensor can measure changes to the system
- . The diameter of the duct work (more air mass means more inertia)
- . The distance between the damper (or fan) and the hood
- . The speed at which the output (damper or VFD) can make changes to the system

The ramification of a slow-to-react fume hood is that it will take longer to achieve the setpoint after a disturbance in the system. As an example, if there is a VFD controlling fans on the output it takes more time to register the change than it would if one was to use a fast-moving damper. This means that the loop will need to be tuned to be far slower to react to changes in the sash height or people moving in front of the hood. It would still be possible to tune the hood to operate at a very fast speed but the inertia of the fans makes them change speed fairly slowly, meaning that it would be common to overshoot the target unless the loop was modified to not drive the output as hard/fast.

The fastest control mode would use an FVR slack-membrane transducer and a very fast acting damper, with feed-forward control enabled. It is not unreasonable with this setup to see the hood react and stabilize to almost any change in the system within 3 seconds. Using a system with slower sensors such as hot wire anemometers and slower dampers such as venture valves might make the same system react within 5 or so seconds. The slowest systems would utilize a VFD to control fan speed (instead of using a damper) and a slow sensor like a hot wire anemometer. The speed of these systems varies greatly with the size of the fans but would react in closer to 15 seconds from a very large disturbance in the system.

Regardless of the hardware associated with the fume hood, tuning the controller is achieved in exactly the same way-the end result will simply be slower or faster based on the equipment used.

3.14.2 CONTROL LOOP OVERVIEW

In the most general terms, control loops are used to achieve a target input value by changing one or more outputs. In the case of a fume hood the face velocity is the input variable that is being controlled. The output in this case is either a damper, a VFD, a venturi valve or a wave chopper circuit driving an exhaust fan. By modifying the output we attempt to influence the face velocity reading in the hopes of having it achieve the target setpoint.

PID loops are the baseline standard when it comes to control loops. This is in part to the fact that have

been around for so long and part because so many people are familiar with how they operate and how to tune them to achieve a different outcome. The first PID loops were mechanical/pneumatic devices that contained a lever a spring and a known mass. Eventually these mechanical devices were replaced by op-amps, capacitors and resistors that mimicked the response curves of the initial mechanical devices. Eventually direct digital control (DDC) became the norm and from that point on PID loops were generally designed in software only, making their creation and modification far easier than it had been with discreet components.

Of course, PID-based control loops are no longer the only game in town. Over the past several decades fuzzy logic loops have grown to rival the popularity of the traditional PID circuit. The term “fuzzy logic” is a much more broad term than PID as there are literally thousands of different control strategies that all fall under the “fuzzy” umbrella. The common feature that makes a control scheme a fuzzy logic scheme is that there is a fuzzification process followed by the application of a rule base and then finally a defuzzification process. Any system that utilizes those three steps can be considered a fuzzy logic system, although the differences between any two fuzzy systems preclude a common method for tuning.

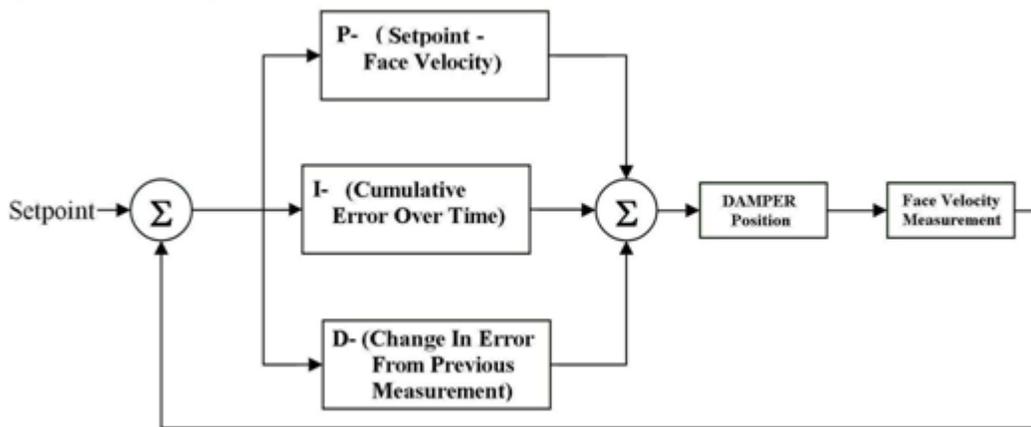
3.14.2.1 BEFORE YOU BEGIN

Prior to tuning any control loops, it is very important that you do three important things:

- . Calibrate the face velocity sensor. If the face velocity reading is inaccurate, you could be forced to reconfigure loop parameters once the hood has been properly calibrated.
- . Set the “Face Velocity Averaging Index” on the display to a value of 1. If it is left at the default value of 10, the readout on the display will be averaged. Averaging the LCD readout is appropriate once the hood has been tuned, but leaving it set above 1 will make it harder for you to interpret the results.
- . Set your copy of NB-Pro to graph the data on the screen at its fastest trend viewer interval, 2 seconds. You can step through all of the PID/Fuzzy parameters via the display screen, so your copy of NB-Pro is most useful as a visual feedback tool when tuning these loops.

3.14.3 PID LOOPS

PID loops are a collection of three influences, Proportional, Integral and Derivative each working with or against each other to direct the change in output of the loop.



Each of the three variables (P,I,D) are constants that are used to specify the power or influence of each of the three parts of the equation.

The “P” portion of a PID loop is the proportional aspect of the loop. This is the difference between the setpoint and the measured value. In our case, this would be the target face velocity of 100 fpm and the measured face velocity.

The proportional gain is the main driving force behind the PID loop and is the variable that determines the response time for the entire system. It is for that reason that the P variable is set first, and then the I and D variables are set to mitigate the overshoot inherent in a P-only system.

If you consider the face velocity at 130fpm and the desired velocity is 100fpm, the instantaneous error in the system would be negative 30fpm. The P-portion of the PID loop would then take that value, multiply it by the user settable P constant and use that to change the damper output.

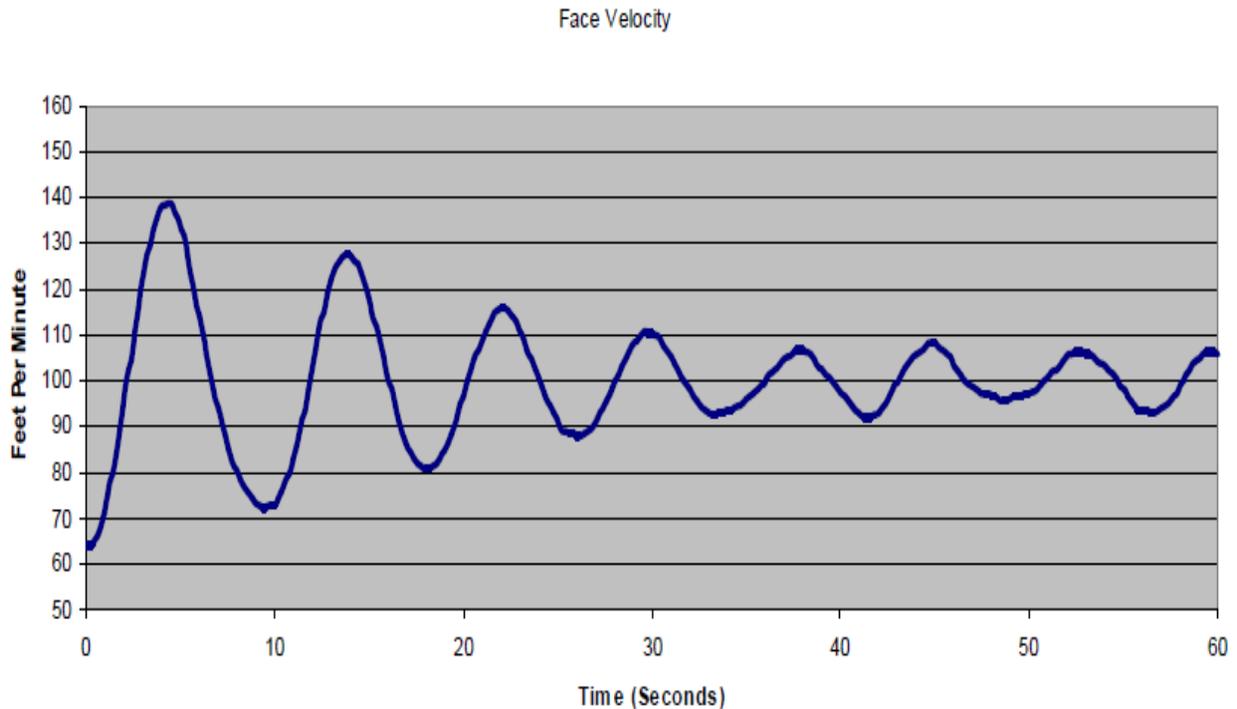
The default proportional gain is set at a value of 6.2. This means that if the error is 30fpm, the proportional portion of the PID loop would be (6.2×-30) or -186. This value is then divided by 100 and added to the present output of the exhaust damper output. In this case, the damper would close 1.86% every time through the loop.

The result of a strictly proportional loop is easy to understand/predict. If the PID loop was tuned so that “P” was the only non-zero value, the result would be oscillations around the setpoint as there would be no other variables to correct for the overshoot. Picture the previous example, where the output was being closed 1.86% every time through the loop. As the damper closes and the face velocity drops, the damper will continue to close every time through the loop until such time as the face velocity is slightly below the setpoint, then it will begin to move the damper more and more open until the face velocity swings back above the setpoint again.

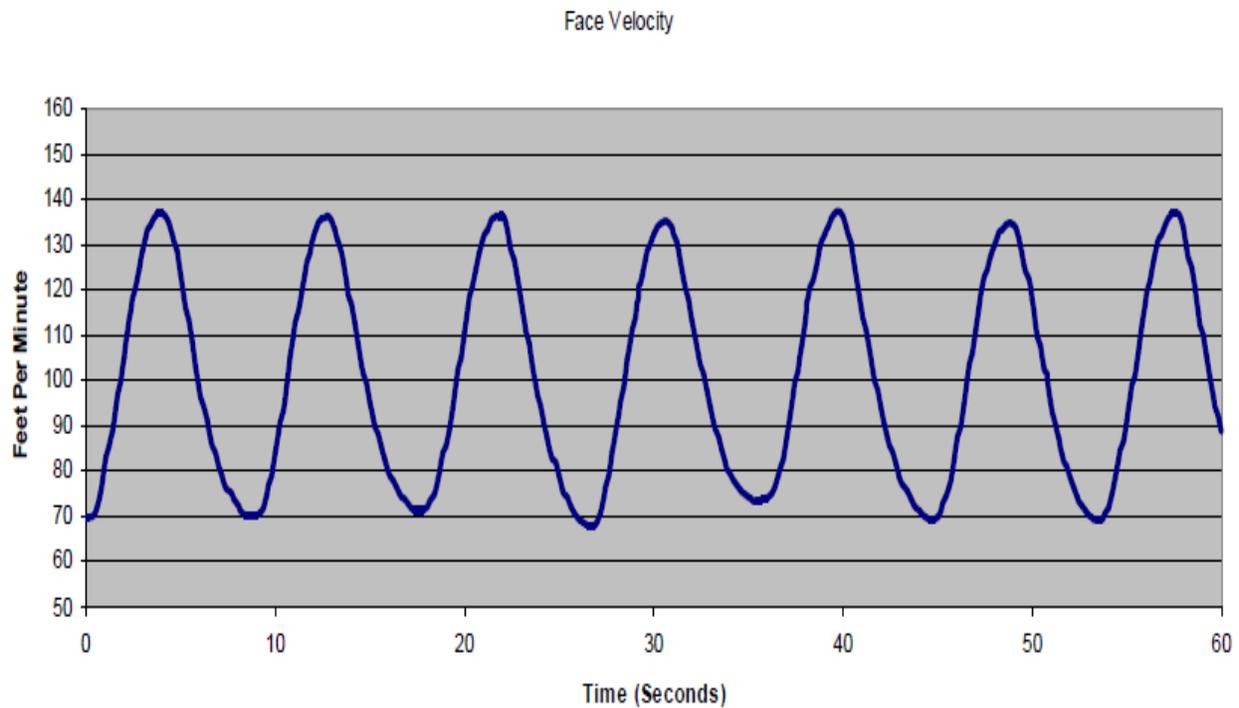
Below are three graphs from the same fume hood. In each the PID loop was set with 0's for the I and D variables meaning that the value of P was the only point set up in the loop. Looking at these graphs, one is able to see the effect of the proportional gain on the PID loop.

In the first graph, the P value is set very low. As you can see, even though there are no I or D values set, the loop will still achieve a fairly steady state over time. This is because the proportional value is set low enough that the overshoot gets less after each consecutive crossover.

Even though this loop is eventually able to stabilize near steady state, it is still undesirable from a control standpoint. At best this particular graph shows the oscillations are able to be capped at +8 fpm, something that is still unacceptable.



By increasing the proportional gain, one is able to get the system into a state of stable oscillations. This state is distinguishable from the previous too-low-setting because over time the oscillations neither taper off to the steady state, nor do they grow to what looks like a saw-tooth (saturated) waveform. In theory this wave should be almost a perfect sine wave, but in practice it might have some very slight variations as seen below.

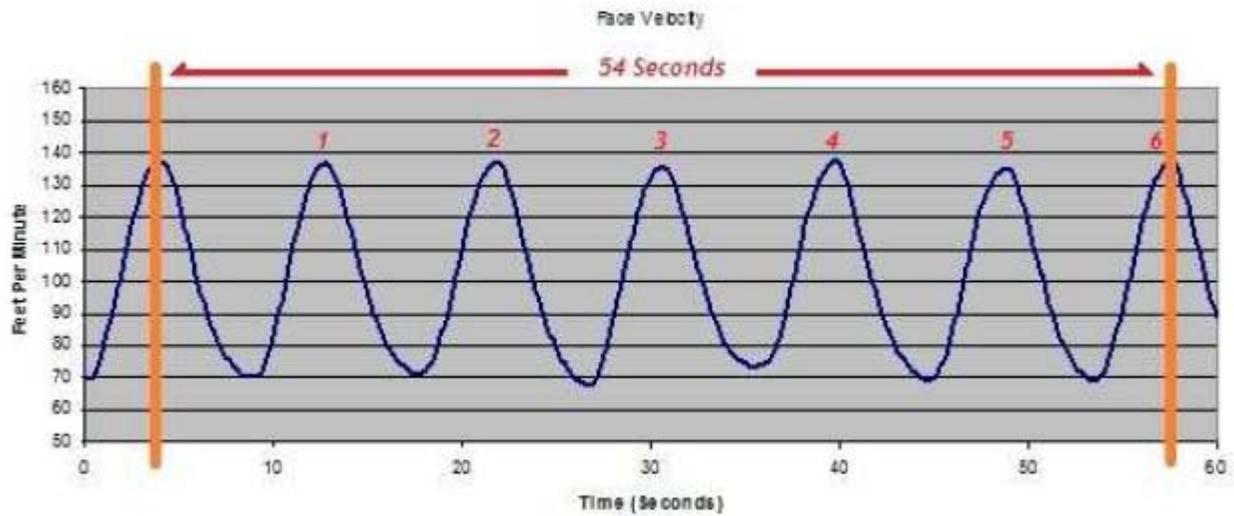


If one were to increase the proportional gain past the steady oscillation state the result is a tighter, more saw-toothed waveform. This state is difficult to distinguish from the stable oscillation waveform shown above, except that it has more jagged points to the curves, the actuator spends more/most of its time at full open or full closed, and this system will be unstable all the time, regardless if there was an initial disturbance in the system.

In order to tune the FHC's PID loop, a variation on the Ziegler-Nichols method is used. The beauty of this method is that once the system is put into a state of steady oscillations, several calculations are made and all three of the PID constants are easily derived.

Recall the graph above where the PID loop was put into a state of steady oscillations by setting the values of I and D to zero and stepping the P value up until the oscillations were permanent and did not decrease over time. This was achieved in this system by setting P to a value of 13.

Once this variable has been determined, the period of the oscillations needs to be calculated. The graph below is the same as the one Steady State oscillation graph found above, except that the timing characteristics have been overlaid:



Using the Ziegler Nichols method, the PID parameters should be set to be the following:

- . $P = (P_steady_state_oscillations / 2)$
- . $I = (Period / 2)$
- . $D = (Period / 8)$

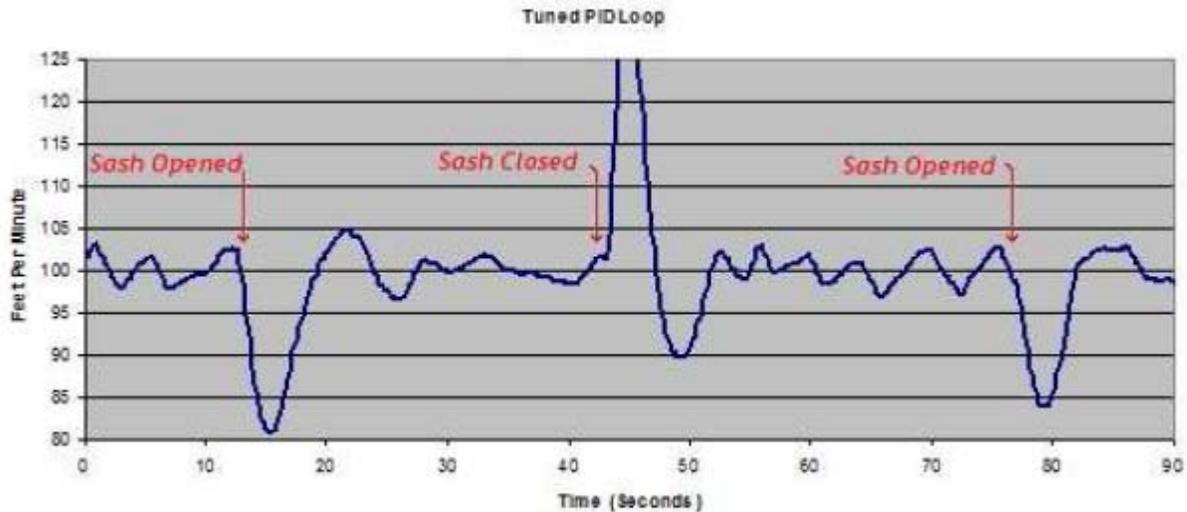
Using this simple equation, the optimal PID settings can be determined to be:

- . $P = 13/2$
- . $I = ((54\text{ Seconds} / 6) / 2)$
- . $D = ((54\text{ Seconds} / 6) / 8)$

The result of these calculations yields the following settings:

- . $P = 6.5$
- . $I = 4.5$
- . $D = 1.125$

Plugging these variables into the controller yields the following response curve:



3.14.4 FUZZY LOGIC

AAM's alternative to the PID loop is a custom fuzzy logic loop. A properly tuned PID loop can be considered the best option for control, both because it is fast to achieve the setpoint and because it is understood by so many people. The case for the fuzzy logic system is simple-it is better at responding to changes to the overall system.

Take for example a laboratory where there are 20 hoods side by side, all fed into the same exhaust duct. If the exhaust fan is not set up (or is unable) to respond differently depending on the number of hoods in the system, you end up with a system where the basic parameters of the system are different depending on the number of hoods in use. This is not a perfect situation for a PID loop, as you will find that when one hood is running you will have a different response curve than when all 20 are running. This would be apparent if you tuned the hood twice, once with only one hood running and again with all 20 running. The resulting PID parameters might be different depending on what the system was doing when you tuned it each time.

To counter this situation, it is common for the installer to "detune" the PID loop so that it is not as fast as it could be but it is more forgiving when there are changes to the system. Detuning the loop can be an appropriate approach, but there are settings where this would not be appropriate or acceptable to the customer. The simplest solution for a very dynamic system is to switch from using the PID loop and instead making use of the fuzzy logic loop.

AAM's fuzzy logic loop has been designed with a high speed fume hood system in mind, meaning that it is a strictly proportional derivative (PD) system. Although very different from a PID loop, it still requires three variables to be properly tuned. These variables are two proportional gains, one for large changes and another one for minor variations around the setpoint, and one derivative (rate-specific) variable.

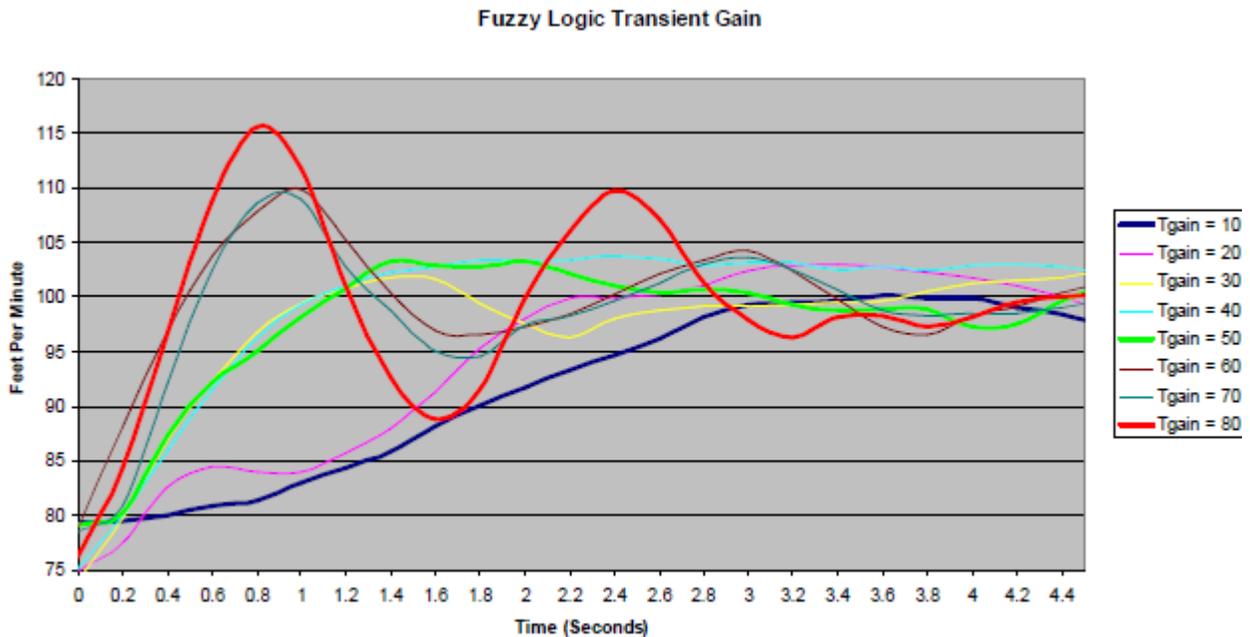
The three tuning parameters are as follows:

1. Transient Gain-Determines how hard the loop will drive the damper when the error is large
2. Steady State Gain-Specifies how hard to drive the damper when the face velocity is very near to the setpoint
3. Rate Sensitivity-Specifies how important the rate of change is to the calculation

The transient gain is the first variable to be tuned in the fuzzy logic routine. This variable dictates the

overall response time of the loop, much like the P variable does in a PID loop. By increasing this number, the loop will drive harder/faster.

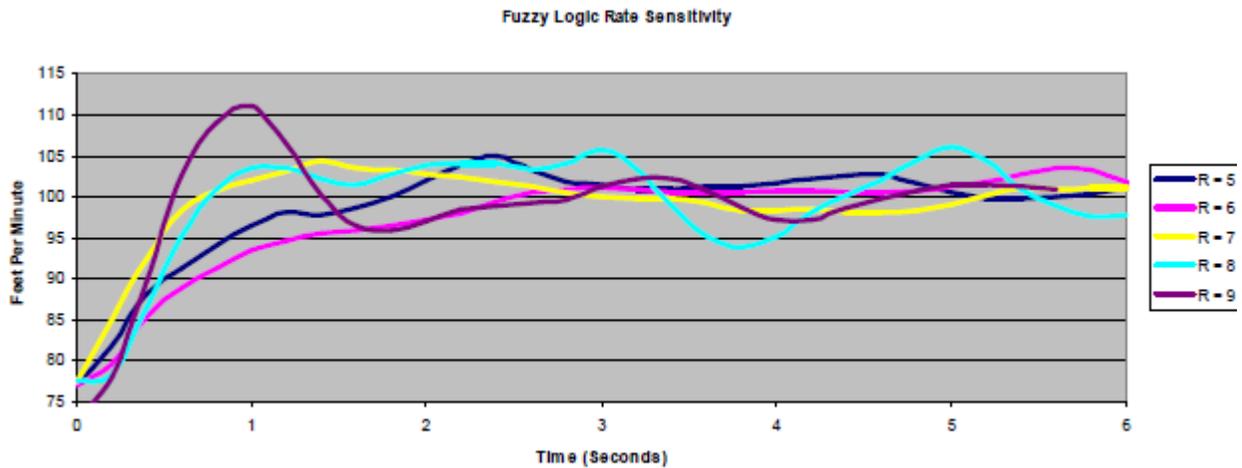
Unlike PID loop tuning, watching a careful graph of the data is not required, instead all that is needed is a bit of trial and error. The below graph shows many step-changes to the transient gain, all with default settings for the steady-state gain and the rate sensitivity. Please note that graphing is not required for tuning a fuzzy logic loop, although the provided graph will help explain the tuning process:



The main point of interest when choosing an appropriate Tgain is how fast it responds and how much overshoot there is. Allowing some overshoot is acceptable as that can be reduced by changing other parameters in the next steps, but it should not have much overshoot at this stage of the tuning process.

Looking at the graph above, it is clear that setting the Tgain to a value between 30 and 50 results in a recovery time of roughly 1.2 seconds before the face velocity is within the target zone of 95 to 105 fpm. For this reason, a value of 40 was selected for this hood. Please note that the fuzzy logic routine is incredibly flexible and will work fairly well without worrying too much about fine tuning parameters. If it seems fast and stable, set the value and move on.

Now that the Transient gain has been set, it is time to set the rate sensitivity, (R). This value determines how much weight to give the change in speed (the differential) when calculating the appropriate change in damper position. Setting this value is accomplished by stepping the value downwards until there is little to no overshoot.

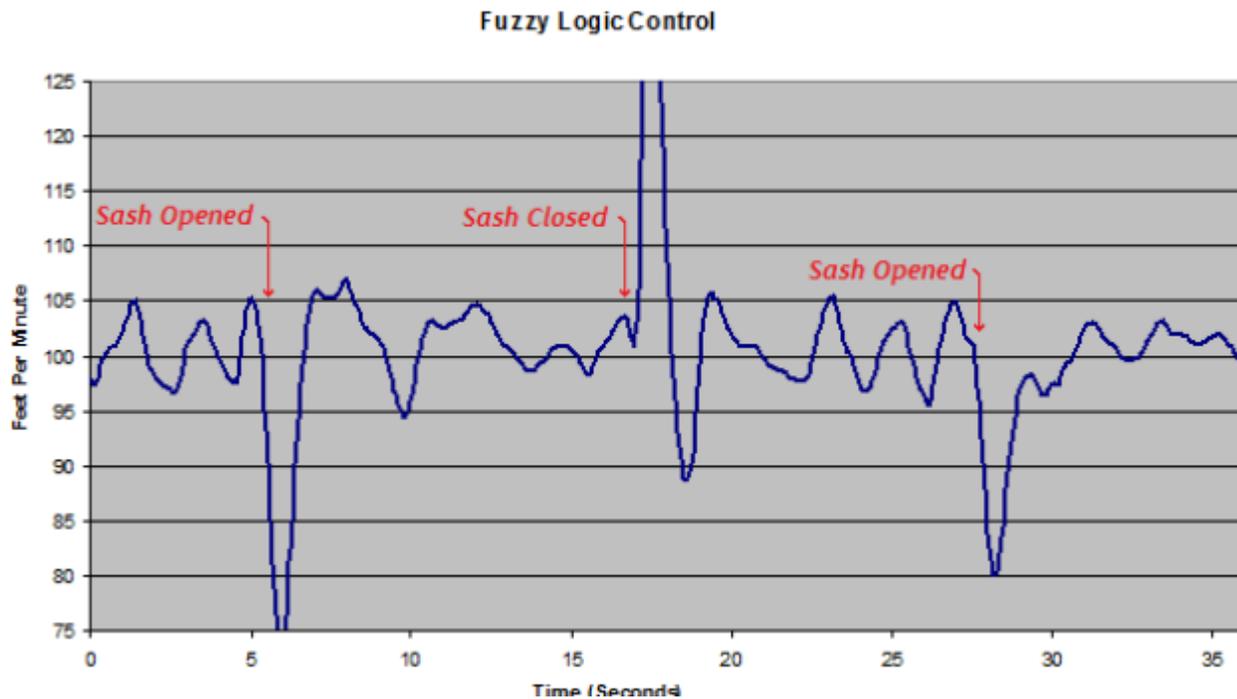


As you can see in the chart above, the overshoot is reduced to a maximum of 4 feet per minute when the rate Sensitivity is stepped back to 7. This is the value that will be used for the hood going forward.

The only other parameter is the steady state gain. This is set by stepping through the various steady state gain settings until the unit is stable when the system is at steady state (meaning when no one is moving and there are no step-changes to the system).

This part of the puzzle is easiest to accomplish standing motionless in front of the hood, varying the SS-Gain via the Touchscreen display. When the steady state gain is too low, the output will tend to drift from the setpoint as it is not being driven hard enough. When it is set too high the output will drive very hard across the setpoint, resulting in miniature sinusoidal oscillations. Choosing an appropriate value for this variable should take no more than a minute as you step through the values.

Once the Fuzzy Logic routine has been appropriately tuned, you should find that the hood is very stable and very resilient to any sorts of changes to the system. Below is a graph of a fairly well tuned steady state system. You can see three instances where the sash was moved during this test.



3.14.5 FEED FORWARD CONTROL

Feed-Forward is a method of control where the output is driven from one position to another based not on the actual feedback from the face velocity sensor, but on other changes to the system. In this way the loop can make changes to the output not based on velocity feedback, but rather from its ability to account for a change in the system that is about to affect the face velocity. This sort of preemptive move is considered “feed forward” precisely because it reacts to the system in advance of any change in face velocity.

In the case of AAM's fume hoods, feed forward control is used to remove changes in sash height from the control loops. By adding a string-potentiometer to any single sash hood, the FHC is able to continuously and instantaneously calculate the open area of the hood. As the sash is thrown open or closed (or quickly/slowly moved anywhere in between) the feed-forward algorithm takes over from the PID/fuzzy loop, allowing the damper to move to a new target position before releasing control back to the control loop.

If the fume hood has a single sash, it is highly recommended that a sash-pot be included to utilize feed forward. The reason for this recommendation is simple: when feed forward is set up, the hood can react virtually instantaneously to big step changes to the system resulting from sash movement. The secondary ramification of using feed-forward is that the control loop can be tuned to be more steady/responsive at near steady state instead of having to be able to take both the macro (sash movement) and micro (human movement in front of the hood) into account. This means that in general terms, the proportional aspect of the control can be toned down which results in incredibly steady airflow at steady state.

Feed forward can be used in either PID or Fuzzy logic routines in the FHC. Because the feed forward variable is a ratio of the open area of the sash to the damper output it is universally applied, regardless of which style of control you are using. This means that once you have configured the feed forward variable

you can switch between fuzzy and PID routines without having to reset or recalculate any of the feed forward variables.

Feed forward control can be set up prior to tuning the loop, however there is one consideration to take into account. Specifically, tuning the loop is achieved by introducing a disturbance to the system and then monitoring the response from the controller. This is most easily achieved by “throwing the sash” from one position to another. In the case of feed-forward, moving the sash should have very little effect on the system as the output will be driven proportionally to the change in open area. For this reason, if feed-forward is set up prior to tuning the loops a different type of disturbance should be introduced, typically this would be a setpoint step-change triggered by putting the unit in or out of service.

SECTION 4: USING THE FHC-SD

This section describes operation of the FHC-SD Touch-Screen Display, detailing usage information regarding the user interface and general instructions.

IN THIS SECTION

| | |
|-----------------------------------|------|
| The Operator Interface..... | 4-3 |
| The Home Screen..... | 4-5 |
| Main Menu..... | 4-7 |
| Operator Password Protection..... | 4-7 |
| Data Entry Overview..... | 4-8 |
| Alarms Menu..... | 4-9 |
| Hood Setpoints..... | 4-10 |
| Inputs..... | 4-11 |
| Configure..... | 4-13 |
| Calibrate Screen..... | 4-13 |
| Product Information..... | 4-13 |
| Session Timer..... | 4-13 |
| Operators..... | 4-14 |
| Modify Programming..... | 4-14 |
| Store Debug Information..... | 4-15 |
| Hood Setup..... | 4-16 |
| Diagnostics..... | 4-17 |

4.1 THE OPERATOR INTERFACE

The FHC-SD provides hood operators with status information regarding the operation of the NB-FHC. The color touch-screen interface consists of dynamic fields that can be used to reference and modify pertinent data points in the Fume Hood Controller. Other aspects of the operator interface may appear with additional buttons that are used to configure the FHC-SD, enter pin-passwords, penetrate menus, change system variables, monitor and modify network attributes, scroll through attribute and alarm lists, perform tests, and signal emergency override conditions. Buttons displayed on the FHC-SD will perform different functions depending on the situation.

4.1.1 CONFIGURATION VIA BACNET INTERFACE

There is an object in the Fume Hood Controller that pertains solely to the display. This object is named “Touchscreen Display (Port 2)” and it is used to set up some of the parameters that govern the look, feel and access levels of the display.

| | |
|---|------------------------------|
| ✓ (CA) FHC Display Comm Status (49985) | Online (1) |
| ✓ (ST) Session Timer (54100) | 10 |
| ✓ (O1) Operator 1's Password (53041) | 11111 |
| ✓ (P1) Operator 1's Privilege Level (53297) | 2 |
| ✓ (O2) Operator 2's Password (53042) | 0 |
| ✓ (P2) Operator 2's Privilege Level (53298) | 0 |
| ✓ (O3) Operator 3's Password (53043) | 0 |
| ✓ (P3) Operator 3's Privilege Level (53299) | 0 |
| ✓ (O4) Operator 4's Password (53044) | 10000 |
| ✓ (P4) Operator 4's Privilege Level (53300) | 2 |
| ✓ (AL) Silence Time In Minutes (When Silence Button Is Pressed) (49484) | 15 |
| ✓ (HS) Home Screen Layout (51283) | Face Velocity + Setpoint (1) |
| ✓ (HT) Optional Home Screen Text (51284) | Add Text... |
| ✓ (HB) Background Color (51266) | White (0) |
| ✓ (AN) Face Velocity Display Averagin Index (1-15 Samples) (49486) | 10 |
| ✓ (SN) FHC Display Serial Number (54094) | 131594 |
| ✓ (VE) FHC Display Firmware Version (54853) | 1.030000 |
| ✓ (SR) FHC Display Flash Release Time-Stamp (54098) | 6192311 |
| ✓ (RE) Reset Emergency Mode (53829) | No (0) |
| ✓ (EE) Show Press For Emergency Button on Display? (50501) | Yes (1) |
| ✓ (LB) Show Lighting Output Button on Display? (52290) | Yes (1) |

Figure 4-1 Viewing the Display Configuration in NB-Pro

4.1.1.1 (CA) FHC DISPLAY COMM STATUS

This read-only property describes the current communication status of the display. IF the display is connected, powered up and communicating, this value will be set to “Online.” If the display has not been attached or is not communicating with the hood, this variable will indicate “offline.”

4.1.1.2 (ST) SESSION TIMER

This property indicates the time in minutes that a user can be logged into the display. Of course a user can also manually log out of the display at any time. Regardless of whether the user is still near the hood, after X minutes they will automatically be logged out. The default for this value is 10 minutes.

4.1.1.3 OPERATOR PASSWORDS/PRIVILEGES

Some configuration parameters and display screens are meant to be viewed by all parties but some have been reserved for only a select few people. As an example, you would not want an end user standing in front of a hood to be able to accidentally change the balancing data, the type of hood, or the sensor management data. On the other hand there is good reason to allow these users access to the I/O data.

You can assign up to 4 different passwords to the display, each of which can have a unique level of privileges. There are three privilege levels available, 0, 1 and 2. Level 2 access is the highest level and it allows access to all screens and abilities. Level 1 access has many but not all of the privileges of level 2 and a “privilege level” of 0 indicates that the user cannot log into the system.

By default, User 1’s password is “11111” and this user has been granted level 2 access.

4.1.1.4 (AL) ALARM SILENCE TIMER

When the FHC is in an emergency or alarm condition, the alarm buzzer in the display will sound. When this buzzer is active a button will appear on the home screen that allows the user to temporarily silence the alarm.

The **(AL)** property indicates the time in minutes that the buzzer in the display can be temporarily silenced. By default this time is set to 15 minutes.

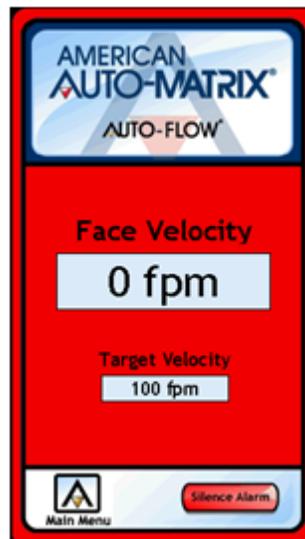


Figure 4-2 Home Screen In Alarm with “Silence Alarm” Button Pressed

4.1.1.5 HOME SCREEN LAYOUT

There are 4 different layouts available for the Home Screen on the FHC-SD. The options include the following:

- . Displaying the Face Velocity
- . Displaying the Face Velocity and the current Face Velocity setpoint
- . Displaying the Face Velocity with small custom text above
- . Displaying the Face Velocity with large custom text above

The third and fourth options require that you fill out the next variable, “Home Screen Text.” By default it states “Add Text...” but it is generally assumed that it would indicate the number of the hood, the

chemicals used in it, etc. Depending on how long a message you want to display you may find either the large or small text options to be best for your particular application.

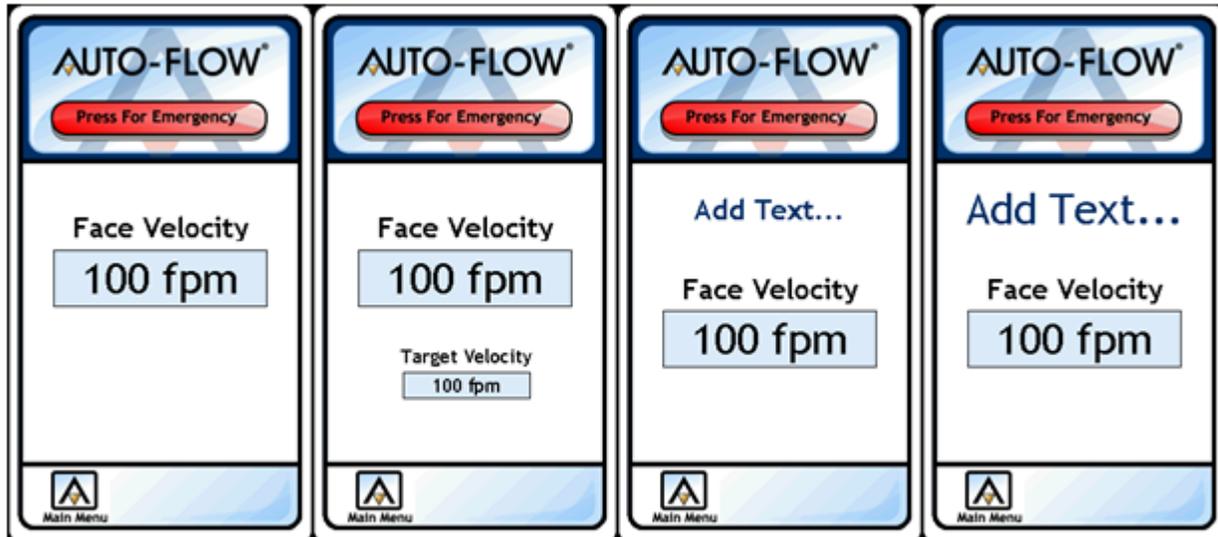


Figure 4-3 Possible Home Screen Configurations

4.1.1.6 (HT) HOME SCREEN TEXT

If you choose to have custom text on your home screen you will need to configure the text in this variable. By default it says “Add Text” but it can be configured to display up to 16 characters.

NOTE



Some characters take up more space than others. If your message cannot fit across the screen with the large text option, selecting the small text option should allow the message to appear properly.

4.1.1.7 (HB) HOME SCREEN BACKGROUND CO

4.1.2 THE HOME SCREEN

Upon start-up, the FHC-SD will display the Home Screen, which displays the current measured face velocity. The Home Screen can be configured to also display the current control setpoint, or custom text in small or large font size (consisting of about 12 to 15 characters, size dependent). When enabled, a software emergency push button can be made available to command a fume hood into emergency mode.

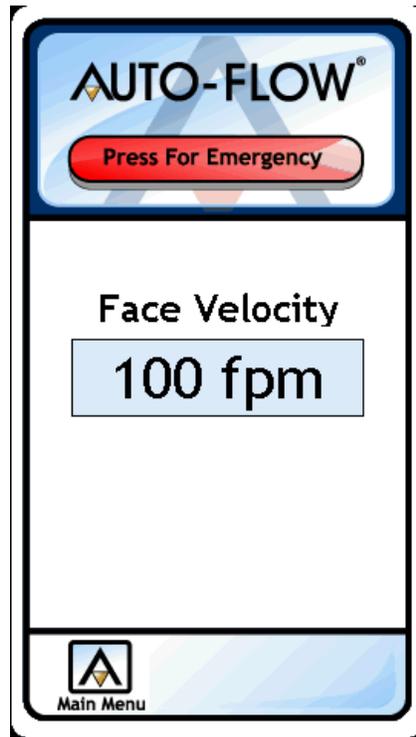


Figure 4-4 FHC-SD Home Screen

4.1.3 MAIN MENU



The Main Menu button provides access to the main menu of the system. Depending on privilege level configurations, an operator may access specific functions with the FHC-SD. The Main Menu provides navigational access to many topical areas regarding the functionality of the NB-FHC including alarm information, input and output monitoring, and setpoint adjustment information to name just a few.

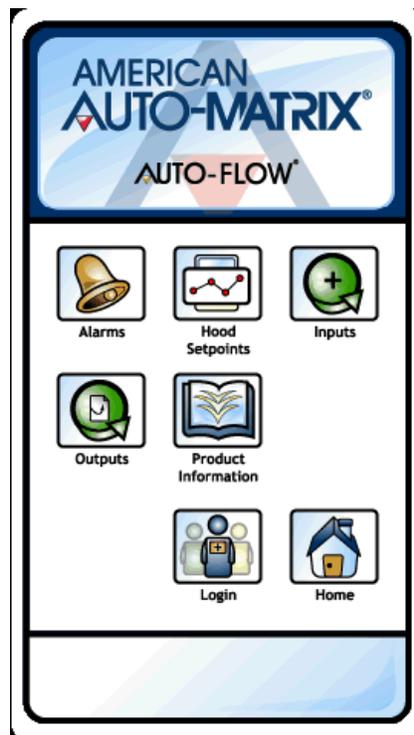


Figure 4-5 FHC-SD Main Menu

4.1.4 OPERATOR PASSWORD PROTECTION

The NB-FHC accommodates up to a maximum of four operator passwords, providing access to different areas of the touch-screen display.

To login as an operator, touch the Login button. The operator is prompted to enter an operator password to gain access to additional menu capabilities. From the password prompt screen, you must enter a five-digit password using the direction keys on the touch-screen display. When shipped from the factory, the NB-FHC includes four default operator passwords, all of which are configured for Level 1 permissions. Operator passwords should already be configured, as discussed in Section 3 of this manual.

If the five-digit password that you enter matches one of the programmable operator passwords, the corresponding operator will be logged in. If multiple operators have the same password, the higher access level will be used when this password has been used for login.

4.1.5 DATA ENTRY OVERVIEW

Modifying values through the FHC-SD is accomplished through the use of the data entry panel. The data entry panel provides up, down, left, and right buttons. Using these buttons, modifications to numeric values can occur by using the left and right arrows to navigate to the placeholder desired, followed by touching the up or down buttons to increment or decrement the value respectively. Holding any of these buttons longer than one second will result in faster, continuous increments or decrements of the desired value. In the example below, the Low Extreme Alarm Limit is being edited. The edit cursor can be moved to any placeholder of the value, and edited in incremental or decremental steps.

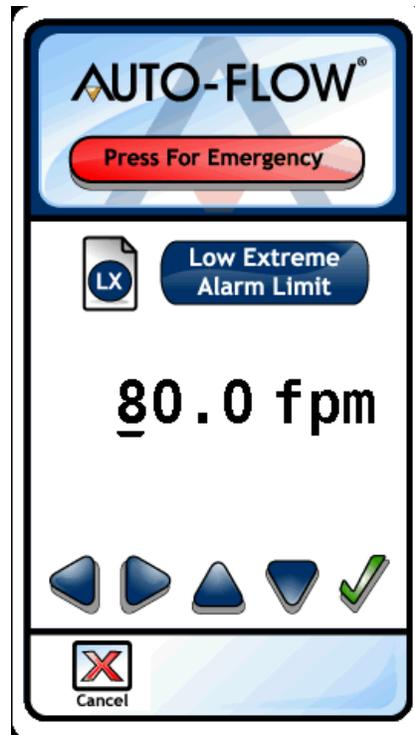


Figure 4-6 Example of Editing a Value

4.2 ALARMS MENU



The Alarms menu button provides access to alarm index of the FHC-SD. The Alarm index provides access to the following features:

- .Monitor Alarm Statuses of Inputs and PID Control
- .Adjust the Alarm Limits of Face Velocity and Exhaust Flow, including
 - .Low Limit Alarm Setpoint
 - .Low Extreme Limit Alarm Setpoint
 - .High Limit Alarm Setpoint
 - .High Extreme Alarm Setpoint

To adjust alarm limits, an operator with at least Level 1 Permissions must be logged into the display.

From this menu, users may also enable, cancel, and disable emergency mode. While operators with Level 1 Permissions can cancel emergency mode, the ability to enable and disable emergency mode is available exclusively to operators with Level 2 permissions.

Table 4-1 Alarms Permission Notes

| Feature | Permission Level Required |
|------------------------|---------------------------|
| Alarm Status | Does not require login |
| Alarm Limits | Level 1 or greater |
| Audio Controls | Level 1 or greater |
| Cancel Emergency Mode | Level 1 or greater |
| Enable Emergency Mode | Level 2 |
| Disable Emergency Mode | Level 2 |

4.3 HOOD SETPOINTS



The Hood Setpoints menu button provides access to adjust setpoint parameters used by the NB-FHC to perform fume hood control. Within this menu, operators can monitor and modify the current Face Velocity setpoint, directly affecting PID Control performance of the Exhaust Damper Output. Any user (logged in or not) has the ability to monitor the Face Velocity Setpoint.

In order to have access to modify the parameter, an operator with at least Level 1 permissions must be logged in to the display.

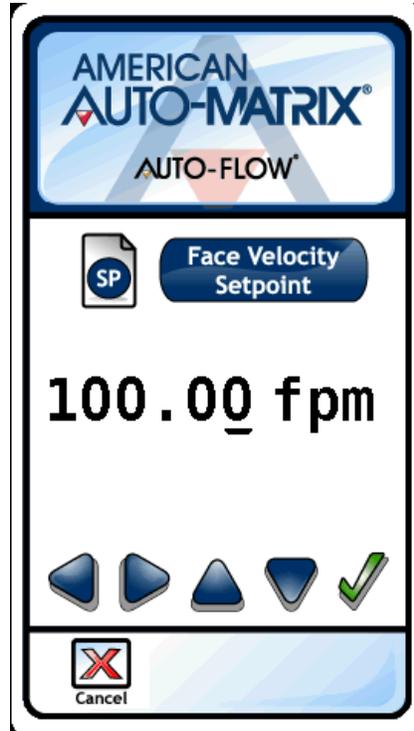


Figure 4-7 Adjusting the Face Velocity Setpoint

4.4 INPUTS



The Inputs menu button provides access to the Input Values screen, which provides an overview of input values. From this display, users can monitor the following input values:

- .Face Velocity
- .Exhaust Flow
- .Emergency Input Status
- .Sash Position Resistance
- .Cascade Flow
- .Hood Presence Status

The ability to monitor input values does not require any specific operator login. However, it is important to understand that this screen provides a read-only summary of input values.

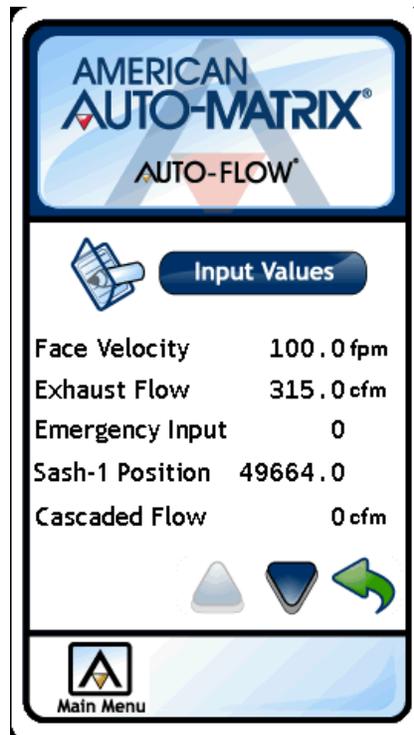


Figure 4-8 Input Values Screen

4.4.1 OUTPUTS



The Outputs menu button provides access to a read-only display of on-board Analog and Digital output statuses. The values monitored from this screen provide real-time updates for monitoring and troubleshooting purposes. From this display, users can monitor:

- .Exhaust Damper Output
- .Cascade Flow
- .External Alarm

The ability to monitor output statuses does not require any specific operator login. However, it is important to understand that this screen provides a read-only summary of output statuses.

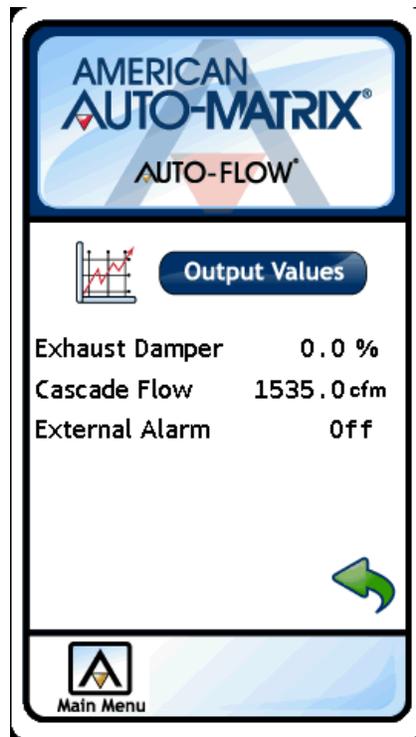


Figure 4-9 Output Values Screen

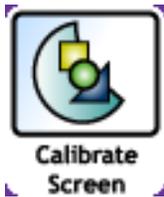
4.5 CONFIGURE



The Configure menu button provides access to extended configuration options of the FHC-SD. From this area, users can configure the following areas:

- .Calibrate Screen
- .Product Information
- .Session Timer
- .Operators
- .Modify Programming (Performing Firmware Updates)
- .Store Debug Information

4.5.1 CALIBRATE SCREEN



The Calibrate Screen option provides single point access to calibrate the parameters of the FHC-SD's touch-screen display. When selected, a target button will display on select areas of the touch-screen display. To calibrate, simply touch the target button as displayed on each area of the screen.

To access this feature, an operator must have (at least) Level 1 permissions.

4.5.2 PRODUCT INFORMATION



The Product Information option provides a basic overview of read-only details regarding the NB-FHC controller and the FHC-SD hardware unit. Information included in this screen includes:

- .Serial Number Information
- .Current and Original Factory Firmware
- .Unit ID Information
- .Original Date of Manufacturing.

To access this feature, an operator must have (at least) Level 1 permissions.

4.5.3 SESSION TIMER



The Session Timer option provides the ability to monitor and adjust the current log-in session timer. The Session Timer can be set as low as five (5) minutes to up a maximum of one-hundred-twenty (120) minutes.

This area also provides an option to view the session time remaining for the operator logged into the FHC-SD.

To access this feature, an operator must have (at least) Level 1 permissions.

4.5.4 OPERATORS



The Operators option provides access to setup and modify Operator Passwords and Privilege levels for the FHC-SD. Changes made through the FHC-SD to these parameters take effect immediately throughout the system. However, if you are modifying your own operator password and or privilege level, the change will not take effect until your login session has expired or manually logout.

To access this area of the FHC-SD, an operator with Level 2 privileges is required.

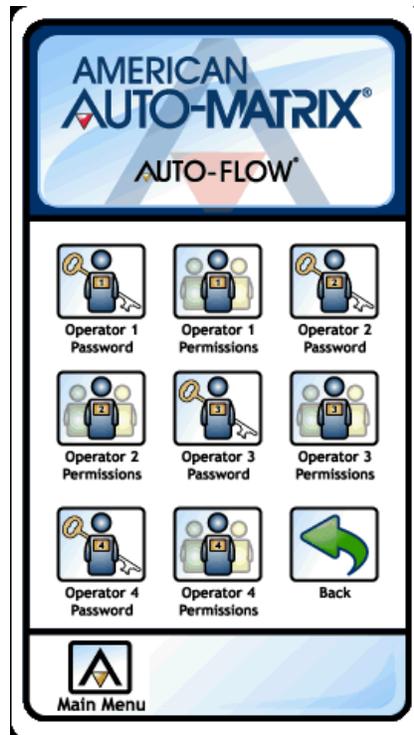
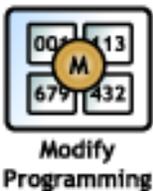


Figure 4-10 Operators Screen

4.5.5 MODIFY PROGRAMMING



The Modify Programming button option is used to perform firmware upgrades to the FHC-SD. From the Modify Programming option, two sub-options, Update Firmware and Enter Postdates, can be selected from this area.

The Update Firmware option allows users to update the product firmware to new revisions that are made available or purchased. The process to update firmware requires use of the SD Flash Memory card slot. A firmware upgrade procedure for each update is provided by American Auto-Matrix Technical Services upon release of a new firmware enhancement.

The Enter Passcodes option is to be used when directed to by American Auto-Matrix Technical Services.

To access this area of the FHC-SD, an operator with Level 2 privileges is required.

4.5.6 STORE DEBUG INFORMATION



**Store Debug
Information**

The Store Debug Information option provides the ability to store pertinent system debug information onto an SD Memory Card when troubleshooting issues in the field. While this feature may be used in rare circumstances, it is recommended to only access this feature when suggested to do so by AAM Technical Service.

To access this area of the FHC-SD, an operator with Level 2 privileges is required.

4.6 HOOD SETUP



The Hood Setup menu button provides the ability to modify key setup parameters of the NB-FHC for fume hood control applications. From the menu, users have the ability to:

- .Modify the Engineering Unit Configuration of the NB-FHC
- .View Unit Setup Data - provides a read-only display of the NB-FHC Unit Setup
- .Unit Setup Wizard - provides a guided wizard setup method to configuring sash position-based applications.
- .Sensor Coefficient - provides a method to adjust the sensor coefficient

Operators with Level 1 Permissions have the ability to modify the engineering unit configuration of the NB-FHC. All other features of this menu require an operator with Level 2 permissions.

Table 4-2 Hood Setup Permission Notes

| Feature | Permission Level Required |
|-------------------------------|---------------------------|
| Modify Engineering Unit Setup | Level 1 or greater |
| View Unit Setup Data | Level 2 |
| Unit Setup Wizard | Level 2 |
| Sensor Coefficient | Level 2 |

4.7 DIAGNOSTICS



Diagnostics

The Diagnostics menu is also commonly referred to as the Super User Menu. In this menu, users can view various diagnostic information regarding various operational parameters of the FHC-SD. Items available for viewing in this area include:

- .STATbus Communication Statistics
- .CPU Usage
- .Individual CPU Task Usage
- .Task Timing
- .Task Stack Usage

To view each parameter, simply press the up and down arrows to scroll through the list of values. For general performance, these variables have no effect on day-to-day operations, as this feature is typically used for troubleshooting with AAM Technical Services.

To access this area of the FHC-SD, an operator with Level 2 privileges is required.

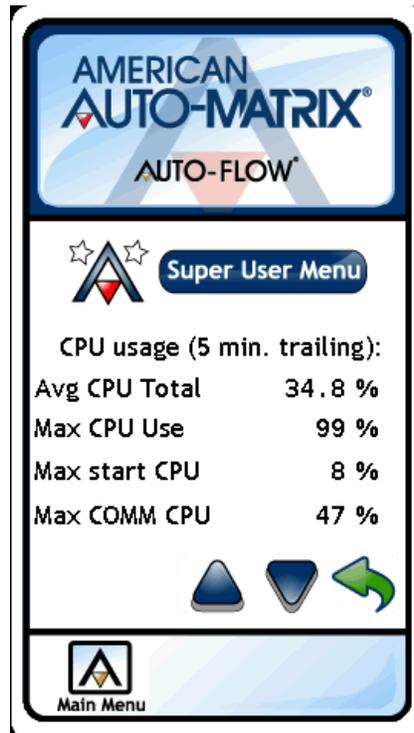


Figure 4-11 Super User Menu Statistics

APPENDIX A: OBJECTS & PROPERTIES

The following tables contain listings of the BACnet objects and property assignments for the NB-FHC. 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 SECTION

| | |
|-------------------------------|------|
| Device | A-2 |
| NOTIFICATIONCLASS1 | A-8 |
| FILE0 | A-9 |
| Universal Inputs 1-12 | A-10 |
| Digital Inputs 1-8 | A-13 |
| Analog Outputs 1-2 | A-15 |
| External Alarm | A-17 |
| Analog Values | A-19 |
| Binary Values | A-20 |
| PID Control 1 | A-21 |
| STATbus 1, 3 and 4 | A-24 |
| Small Display Config | A-26 |
| Universal Input Summary | A-28 |
| Digital Input Summary | A-30 |
| Analog Output Summary | A-31 |
| Hood Presence Detector | A-32 |
| Digital Output Summary | A-33 |
| Control Summary | A-34 |
| Piecewise Curves 1-4 | A-35 |
| Sash Area | A-37 |
| Broadcast 0 - 7 | A-38 |

A.1 DEVICE

NOTE

The Device object is represented in *NB-Pro* as follows:

AAM NB-FHC 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 | RW | EEPROM Device (8), Instance <i>serial number</i> | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM AAM NB-FHC <i>serial number</i> | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Device (8) | indicates membership in a particular object type class. |
| system_status | 112 | BACnet ObjID | RO | - 0 | indicates the current physical and logical status of the BACnet Device. |
| vendor_name | 121 | CharStr | RO | NRAM American Auto- Matrix | identifies the manufacturer of the BACnet Device. |
| vendor_Identifier | 120 | Unsigned | RO | - 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 NB-FHC1 | indicates the vendor's name used to represent the model of the device. |
| firmware_revision | 44 | CharStr | RO | NRAM <i>revision number</i> | indicates the level of firmware installed in the device. |
| application_software_version | 12 | CharStr | RO | NRAM <i>version number</i> | identifies the version of application software installed in the device. |
| location | 58 | CharStr | RW | NRAM " " | user defined location of the BACnet device. |
| description | 28 | CharStr | RW | NRAM " " | user defined description of the BACnet device. |
| protocol_version | 98 | Unsigned | RO | - 1 | indicates the version of the BACnet protocol supported by this BACnet Device. |
| protocol_revision | 139 | Unsigned | RO | - 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. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--|--------------|-------------------------------|--------|-------------------|---|
| protocol_object_types_supported | 96 | BACnet Object Types Supported | RO | - | indicates which standardized object types are supported by this device's protocol implementation. |
| object_list | 76 | BACnet Array | RO | - | a list of objects 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 Segmentation | 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. |
| utc_offset | 119 | Integer | RW | NRAM -300 | indicates the number of minutes (-780 to +780) offset between local standard time and Universal Time Coordinated. |
| daylight_savings_status | 24 | Boolean | RO | NRAM 0 | indicates whether daylight savings time is in effect or not. |
| 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. |
| BF | 49734 | Unsigned | RW | NRAM 0 | Broadcast Fire? specifies whether the "change to fire mode" command should be broadcast 0=No 1=Yes |
| BS | 49747 | Unsigned | RW | NRAM 0 | Broadcast Sync Time? specifies whether to broadcast time synchronization information 0=No 1=Yes |
| BT | 49748 | Unsigned | RW | NRAM 255 | Broadcast Time Interval specifies the time (in minutes) between network broadcasts 1-254=time in minutes 255=disabled |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|--------------------|--|
| CF | 49990 | Unsigned | RW | NRAM 0 | Communication Failure Timeout specifies the amount of time, in seconds, that the controller will wait to be polled before entering a communication fault state. |
| CM | 49997 | Unsigned | RO | RAM 255 | Manufacturer indicates the factory-set manufacturer number for the controller. (CM for American Auto-Matrix controllers is always 255) |
| CP | 50000 | Unsigned | RW | NRAM 6 | Communication Speed the rate at which the controller communicates. 0=9600 6=38400 (default) 7=19200 8=115.2k 9=57.6k |
| CR | 50002 | Unsigned | RW | NRAM 0 | Configure Remote I/O specifies options for the configuration of remote I/O device located on the STATbus. 0=Normal 1=GPC to bus 2=Edit I/O GIDs 3=GPC from bus |
| CT | 50004 | Unsigned | RO | RAM 105 | Controller Type factory-set controller type number for the controller. CT for the NB-FHC1 is 205. |
| DA | 50241 | Unsigned | RW | RAM follows RTC | Day of Week specifies the current day of the week. 0=Monday 1=Tuesday 2=Wednesday 3=Thursday 4=Friday 5=Saturday 6=Sunday |
| DD | 50244 | Unsigned | RW | NRAM | Device Instance To Send Alarms To specifies the MSTP address or device instance to which alarm messages will be sent when IA is set to "3=Unconfirmed to Device" or "4=Confirmed to Device". |
| DE | 50245 | Unsigned | RW | RAM 0 | Default Enable used to return all properties in the controller to their default values. 0=Normal operation 197=set properties to their default values. |
| EM | 50509 | Boolean | RW | NRAM 0 | Engineering Units specifies the units to be used when returning values 0=English 1=Metric |
| ET | 50516 | Time | RW | NRAM | Daylight Saving End Time specifies the time at which daylight saving time ends. |
| FA | 50753 | BitStr | RO | NRAM | Faults Detected a bitmap indicating the current fault conditions. bit 0=Fire bit 1=Emergency bit 6=Communications failure 0=Normal 1=Fault condition |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|---|
| FC | 50755 | Unsigned | RO | NRAM | Flash update count indicates the number of times the controller has been flashed. |
| FT | 50772 | Unsigned | RO | RAM | Firmware Type indicates which firmware is installed on the controller 7=FHC1 8=GPC2 9=GPC3 |
| ID | 51524 | Unsigned | RW | NRAM - | Unit Number (ID) specifies the controller's identification number. The value of ID defaults to the last two digits of the unit's serial number. |
| ND | 50500 | Unsigned | RW | NRAM 0 | Daylight Saving End Day specifies the day on which daylight savings time ends 0 = None 1 = First Sunday 2 = First Friday 3 = Second Saturday 4 = Third Sunday 5 = Third Saturday 6 = Last Sunday 7 = Last Thursday 8 = Last Friday |
| NM | 52813 | Unsigned | RW | NRAM 0 | Daylight Saving End Month specifies the month in which daylight savings time ends 0 = None 1 = January 2 = February 3 = March, 4 = April 5 = May 6 = June 7 = July 8 = August 9 = September 10 = October 11 = November 12 = December |
| OS | 53075 | Real | RO | RAM | Kernel Version indicates the version number of the kernel. |
| PD | 53316 | Unsigned | RW | NRAM 15 | Power-Up Delay time delay (in seconds) that must elapse after the controller is reset before it begins control and alarming functions. 0=No delay 1-255=# of seconds |
| PF | 53318 | Unsigned | RW | NRAM 3 | Priority for Fault Alarms specifies where in the priority array to-fault alarms will be listed. |
| PI | 53321 | Unsigned | RW | NRAM | Process ID for Alarms |
| PN | 53326 | Unsigned | RW | NRAM 4 | Priority for Normal Alarms specifies where in the priority array to-normal alarms will be listed. |
| PO | 53327 | Unsigned | RW | NRAM 2 | Priority for Off-normal Alarms specifies where in the priority array to-offnormal alarms will be listed. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|---|
| PS | 53331 | Unsigned | RW | NRAM 0 | Power-up state schedule state that the controller will operate in when it is first powered up or after power is restored following a power failure. 0=unoccupied 1=warmup 2=occupied 3=night setback |
| RD | 53828 | Unsigned | RW | NRAM 0 | Daylight Savings Start Day specifies the day on which daylight saving time begins 0 = None 1 = First Sunday 2 = First Friday 3 = Second Saturday 4 = Third Sunday 5 = Third Saturday 6 = Last Sunday 7 = Last Thursday 8 = Last Friday |
| RM | 53837 | Unsigned | RW | NRAM 0 | Daylight Savings Start Month specifies the month in which daylight savings time begins 0 = None 1 = January 2 = February 3 = March 4 = April 5 = May 6 = June 7 = July 8 = August 9 = September 10 = October 11 = November 12 = December |
| RS | 53843 | Unsigned | RW | RAM 0 | Reset the Controller? used to reset the controller. Setting RS to 1 resets the controller. 0=No 1=Yes |
| SN | 54094 | Unsigned | RO | NRAM - | Serial Number the factory-set serial number. |
| SR | 54098 | Unsigned | RO | RAM xxxxxxxx | Flash Release Code the release code of the firmware currently flashed on the controller, used primarily for technical support purposes. |
| ST | 54100 | Time | RW | NRAM | Daylight Saving Start Time specifies the time at which daylight savings time begins. |
| TF | 54342 | Unsigned | RO | NRAM 0 | Time to Remain in Fire Mode The time (in minutes) that the controller will remain in fire mode. |
| UD | 54596 | Boolean | RW | NRAM | Use Device specifies whether the address specified in DD to which alarm messages will be sent is an MSTP address of a device instance. 0=MSTP Address 1=Device Instance |
| VE | 54853 | Real | RO | RAM | Firmware Version contains the version of the controller's firmware. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|--|
| ZN | 55886 | Unsigned | RW | NRAM 0 | Zone Number specifies the zone number for the controller. All zone broadcasts will only be sent to controllers with matching zone numbers. |

A.2 NOTIFICATIONCLASS1

| Property | Identifier # | Data Type | Access | Default | Description |
|---------------------------|--------------|--------------------------|--------|--|---|
| object_identifier | 75 | BACnet ObjID | RO | - Notification Class (15), Instance (1) | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM NOTIFICATIONCLASS1 | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Notification Class (15) | indicates membership in a particular object type class. |
| description | 28 | CharStr | RO | NRAM "NOTIFICATIONCLASS1" | user defined description of the object and it's functionality. This property is read only for this object type. |
| notification_class | 17 | Unsigned | RO | - 1 | specifies the notification class to be used when handling and generating event notifications for this object. |
| priority | 86 | BACnet Array | RW | NRAM 2 | specifies the priority to be used for event notifications for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events, respectively. |
| ack_required | 1 | BACnet Event Trans. Bits | RW | NRAM 1, 0, 1 | three separate flags that indicate whether acknowledgment shall be required in notifications generated for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL event transitions, respectively. |
| recipient_list | 102 | BACnet List | RO | | a list of one or more recipient destinations to which notifications shall be sent when event-initiating objects using this class detect the occurrence of an event. |

A.3 FILE0

| Property | Identifier # | Data Type | Access | Default | Description |
|---------------------------|--------------|------------------------------------|--------|----------------------------|---|
| object_identifier | 75 | BACnet ObjID | RO | - File (10), Instance 0 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM FILE0 | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - File (10) | indicates membership in a particular object type class. |
| description | 28 | CharStr | RO | - Flash Upgrade | user defined description of the object and it's functionality. This property is read only for this object type. |
| file_type | 43 | CharStr | RO | - Flash Upgrade | identifies the intended use of this file. |
| file_size | 42 | Unsigned | RO | NRAM 327680 | indicates the size of the file data. |
| modification_date | 71 | BACnet Date Time | RO | NRAM NULL | indicates the last time this object was modified. |
| archive | 13 | Boolean | RW | NRAM 1 | indicates whether the File object has been saved for historical or backup purposes. |
| read_only | 99 | Boolean | RO | - 0 | indicates whether or not the file data may be changed through the use of a BACnet Atomic Write File service. |
| file_access_method | 41 | BACnet File Access Method | RO | - 1 | indicates the type(s) of file access supported for this object. |

A.4 UNIVERSAL INPUTS 1-12

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---------------------------|--------------|---------------------|--------|---|---|
| object_identifier | 75 | BACnet ObjID | RO | - Analog Input (3), Instance <i>N</i> or Binary Input (3), Instance <i>N</i> | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Universal Input <i>N</i> | represents a name for the object that is unique internetnetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Analog Input (3) or Binary 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. |
| description | 28 | CharStr | RW | NRAM .. | user defined description of the object and it's functionality. |
| 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. |
| polarity | 84 | BACnet Polarity | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the present_value 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. |
| 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. |
| resolution | 106 | Real | RO | - 0.001525 | indicates the smallest recognizable change in present_value in engineering units (read-only). |
| time_delay | 113 | Unsigned | RW | NRAM 0 | specifies the minimum period of time in seconds during which the present_value must be different from the alarm_value property before a TO-OFFNORMAL event is generated or must remain equal to the alarm_value property before a TO-NORMAL event is generated. |
| notification_class | 17 | Unsigned | RO | - 1 | specifies the notification class to be used when handling and generating event notifications for this object. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|-----------------------------|--------|-------------------|---|
| high_limit | 45 | Real | RW | NRAM 0.0 | specifies a limit that the present_value must exceed before an event is generated. |
| low_limit | 59 | Real | RW | NRAM 0.0 | specifies a limit below which the present_value must fall before an event is generated. |
| deadband | 25 | Real | RW | NRAM 0.0 | specifies a range between the high_limit and low_limit properties within which the present_value must remain for a TO-NORMAL event to be generated |
| limit_enable | 52 | BACnet Limit Enable | RW | NRAM 0 | enables and disables reporting of HighLimit and LowLimit offnormal events and their return to normal. |
| alarm_value | 6 | BACnet BinaryPV | - | - | specifies the value that the Present_Value property must have before a TO-OFFNORMAL event is generated. |
| event_enable | 35 | BACnet Event Trans. Bits | RW | NRAM 0 | three flags that separately enable and disable reporting of TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. |
| acked_transitions | 0 | BACnet Event Trans. Bits | RW | NRAM 7 | three flags that separately indicate the receipt of acknowledgments for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. |
| notify_type | 72 | BACnet Notify Type | RW | NRAM 0 | specifies whether the notifications generated by the object should be Events or Alarms. |
| event_time_stamps | 130 | Array of BACnet Time Stamps | RO | NRAM | specifies the last time each event transition (TO-OFFNORMAL, TO-FAULT, and TO-NORMAL) occurred. |
| AE | 49477 | Unsigned | RW | NRAM 0 | Alarm Enable specifies the type of alarming to be used by the input. 0=Disabled 1=Contact 0>1 2=Contact 1>0 3=Change of State 4=Low Limit 5=High Limit 6=Low & Hi Limit 7=Multi-Limit 8=Fire |
| ST | 54100 | Unsigned | RW | NRAM 0 | Sensor Type specifies the type of sensor connected to the input. 0=Digital 2=MN..MX 0 to 5V 3=MN..MX 4 to 20mA 4=Curve 1 5=Curve 2 6=MN..MX 0 to 10V 7=Thermistor -30.0 to 230.0 8=MN..MX 0 to 20mA 9=SMARTStat Temperature 10=SMARTStat Humidity |
| HX | 51288 | Real | RW | NRAM 0.0 | High Extreme Alarm Limit specifies an extreme high value at which an input reading must measure above in order for a high extreme alarm condition to occur. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|----------|--------------|-----------|--------|-------------------|--|
| HL | | Real | RW | NRAM 0.0 | High Alarm Limit specifies the value at which an input reading must measure above in order for a high limit alarm condition to occur. |
| LL | | Real | RW | NRAM 0.0 | Low Alarm Limit specifies the value at which an input reading must measure below in order for a low limit alarm condition to occur. |
| LX | | Real | RW | NRAM 0.0 | Low Extreme Alarm Limit specifies an extreme low value at which an input reading must measure below in order for low extreme alarm condition to occur. |
| IF | 51526 | Real | RW | NRAM 0 | Input Filter Delay or Weighted Gain the debounce time (in seconds) during which the input must remain stable to avoid the signal being read as a digital bounce. In the case of a bounce, the object reliability is set to 1. For analog inputs, IF specifies a weighted gain used to smooth the values from a fluctuating input. |
| OF | 53062 | Real | RW | NRAM 0.0 | Input Offset specifies an offset amount to be added to the current value. |
| SU | 54101 | Real | RW | NRAM 0.0 | Amount to Setup/Setback Alarm Limit specifies a value (0.0 to 25.5) which is added to high_limit and subtracted from low_limit during scheduled unoccupied periods. |
| SW | | Real | RW | NRAM 0.0 | Sash Width specifies the total width of the fume hood sash door. |
| SH | | Real | RW | NRAM 0.0 | Sash Height specifies the total height of the fume hood sash door. |
| SZ | | Real | RW | NRAM 0.0 | Sash Zero specifies the offset from a zero reading for the sash position sensor. |
| SP | | Real | RW | NRAM 0.0 | Sash Span indicates the raw resistance for the entire sash span when the fume hood sash door has been opened fully. |
| RD | | Real | RO | RAM 0.0 | Raw Resistance Data provides resistance data for configuring the sash span. |
| SA | | Real | RO | RAM 0.0 | Calculated Sash Area provide the calculated area of the sash area. |
| GI | 51017 | Unsigned | RW | NRAM 0 | GID of I/O Device indicates the global identification number of the STATbus device associated with the universal input. If the input does not have a STATbus device mapped to it, GI will be 0. |

A.5 DIGITAL INPUTS 1-8

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---------------------------|--------------|--------------------------|--------|--|---|
| object_identifier | 75 | BACnet ObjID | RO | - Binary Input (3), Instance 49-56 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Digital Input <i>N</i> | 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 | indicates the current value, in engineering units, of the object. |
| description | 28 | CharStr | RW | NRAM "" | user defined description of the object and it's functionality. |
| 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. |
| polarity | 84 | BACnet Polarity | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the present_value 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. |
| time_delay | 113 | Unsigned | RW | NRAM 0 | specifies the minimum period of time in seconds during which the present_value must be different from the alarm_value property before a TO-OFFNORMAL event is generated or must remain equal to the alarm_value property before a TO-NORMAL event is generated. |
| notification_class | 17 | Unsigned | RO | - 1 | specifies the notification class to be used when handling and generating event notifications for this object. |
| alarm_value | 6 | BACnet BinaryPV | - | - | specifies the value that the Present_Value property must have before a TO-OFFNORMAL event is generated. |
| event_enable | 35 | BACnet Event Trans. Bits | RW | NRAM 0 | three flags that separately enable and disable reporting of TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. |
| acked_transitions | 0 | BACnet Event Trans. Bits | RW | NRAM 0 | three flags that separately indicate the receipt of acknowledgments for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. |
| notify_type | 72 | BACnet Notify Type | RW | NRAM 0 | specifies whether the notifications generated by the object should be Events or Alarms. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|---|
| AE | 49477 | Unsigned | RW | NRAM 0 | Alarm Enable specifies the type of alarming to be used by the input. 0=Disabled 1=Contact 0>1 2=Contact 1>0 3=Change Of State 8=Fire |
| RH | 53832 | Real | RW | RAM | Run Hours indicates the number of hours present_value =1 for the input. |
| RL | 53836 | Real | RW | NRAM 0 | Run Limit specifies a number of hours that present_value =1 after which a run limit alarm is generated. Setting RL =0.0 disabled run limit alarms for the input. |
| IF | 51526 | Real | RW | NRAM 0 | Input Filter Delay the debounce time (in seconds) during which the input must remain stable to avoid the signal being read as a digital bounce. In the case of a bounce, the object reliability is set to 1. |
| MD | 52548 | Unsigned | RW | NRAM | Pulse Counter Mode defines how the <i>NB-FHC1</i> detects a pulse. 0=Rising edge 1=Falling edge 2=Both 3=Receive Cascade Flow |
| NP | 52816 | Unsigned | RW | NRAM | Number of Pulses Accumulated displays the current number of pulses detected on the digital input selected in IC . |
| SF | 54086 | Real | RW | NRAM | Pulse Multiplier specifies a scaling factor (0.000 to 65.535) that is multiplied by NP to obtain a scaled count. |
| SV | 54102 | Real | RO | RAM | Scaled Pulse Count displays a scaled version of NP using the formula SV = NP x SF |
| GI | 51017 | Unsigned | RW | NRAM 0 | GID of I/O Device indicates the global identification number of the STATbus device associated with the output. If the input does not have a STATbus device mapped to it, GI will be 0. |

A.6 ANALOG OUTPUTS 1-2

| 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. |
| description | 28 | CharStr | RW | NRAM "" | user defined description of the object and it's functionality. |
| 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. |
| resolution | 106 | Real | RO | - 0.024414 | indicates the smallest recognizable change in present_value in engineering units (read-only). |
| 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. |
| acked_transitions | 0 | BACnet Event Trans. Bits | RW | NRAM 7 | three flags that separately indicate the receipt of acknowledgments for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. |
| OU | 53077 | Boolean | RO | RAM | Actual Output Value specifies the actual output state of the output. This may differ from the current value because of delays and other effects. |
| ST | | Unsigned | RW | NRAM | Scaling Type specifies the type of curve used to scale the Analog Output. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|---|
| MN | 52307 | Real | RW | NRAM 0 | Minimum Scaled Voltage specifies the percentage of the total output for present_value=min_pres_value . |
| MX | 51283 | Real | RW | NRAM 100 | Maximum Scaled Voltage specifies the percentage of the total output for present_value=max_pres_value . |
| EP | 50512 | Real | RW | NRAM 100.0 | Emergency Position specifies the value at which Analog Output; present-value should be commanded to when Emergency Mode has been entered. |
| GI | 51017 | Unsigned | RW | NRAM 0 | GID of I/O Device indicates the global identification number of the STATbus device associated with the analog output. If the output does not have a STATbus device mapped to it, GI will be 0. |

A.7 EXTERNAL ALARM

| 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 | RW | NRAM Digital Output <i>N</i> | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Binary Output (4) | 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. |
| description | 28 | CharStr | RW | NRAM " | user defined description of the object and it's functionality. |
| 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. |
| out_of_service | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |
| polarity | 84 | BACnet Polarity | RW | NRAM 0 | indicates the relationship between the physical state of the output and the logical state represented by the present_value 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. |
| minimum_off_time | 66 | Unsigned | RW | NRAM 0 | specifies the minimum number of seconds that the present_value shall remain in the INACTIVE state after a write to the present_value property causes that property to assume the INACTIVE state. |
| minimum_on_time | 67 | Unsigned | RW | NRAM 0 | indicates the minimum number of seconds that the present_value shall remain in the ACTIVE state after a write to the present_value property causes that property to assume the ACTIVE state. |
| reliability | 103 | BACnet Reliability | RO | RAM 0 | indicates whether the present_value is "reliable" as far as the device or operator can determine. |
| time_delay | 113 | Unsigned | RW | NRAM 0 | specifies the minimum period of time in seconds during which the present_value must be different from the alarm_value property before a TO-OFFNORMAL event is generated or must remain equal to the alarm_value property before a TO-NORMAL event is generated. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|--|
| AE | 49477 | Unsigned | RW | NRAM 0 | Alarm Function indicates the alarm application applied to the Digital Output. 0 = Disabled 8 = Fire |
| OU | 53077 | Boolean | RO | RAM | Actual Output State specifies the actual output state of the output. This may differ from the current value because of delays and other effects. |
| PW | 53335 | Real | RW | NRAM 0 | Pulse Width when Output is On specifies the "on" time (present_value =1) in seconds (0.0 to 25.5) that the output should remain on after a transition from the off to on state. 0=Disabled 0.1-25.5=pulse "on" duration in seconds |
| RH | 53832 | Real | RW | NRAM 0 | Run Hours indicates the number of hours present_value =1 for the input. |
| RL | 53836 | Real | RW | NRAM 0 | Run Hours Limit specifies a number of hours that present_value =1 after which a run limit alarm is generated. Setting RL =0.0 disabled run limit alarms for the input. |
| GI | 51017 | Unsigned | RW | NRAM 0 | GID of I/O Device indicates the global identification number of the STATbus device associated with the digital output. If the output does not have a STATbus device mapped to it, GI will be 0. |

A.8 ANALOG VALUES

| 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 | RW | NRAM Digital Output <i>N</i> | 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 | indicates the current value, in engineering units, of the object. |
| description | 28 | CharStr | RW | NRAM " | user defined description of the object and it's functionality. |
| 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. |
| 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. |

A.9 BINARY VALUES

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|---------------------|--------|--|--|
| object_identifier | 75 | BACnet ObjID | RO | - Binary Value (5), Instance 1-4 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM N | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Binary Value (5) | indicates membership in a particular object type class. |
| present_value | 85 | BACnet Binary PV | RW | RAM 0 | indicates the current value of the object. |
| description | 28 | CharStr | RW | NRAM “ ” | user defined description of the object and it's functionality. |
| 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. |
| out_of_service | 81 | Boolean | RW | NRAM 0 | indicates whether or not the process this object represents is not in service. |

A.10 PID CONTROL 1

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|---------------------------------------|--------------|----------------------------------|--------|---|--|
| object_identifier | 75 | BACnet ObjID | RO | - Loop (12), Instance 1 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM N | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Loop (12) | indicates membership in a particular object type class. |
| present_value | 85 | Real | RW | RAM 0 | indicates the current value of the object. |
| description | 28 | CharStr | RW | NRAM " | user defined description of the object and it's functionality. |
| 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. |
| 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 True | indicates whether or not the process this object represents is not in service. A value of <i>false</i> indicates that the Loop object is in active control, whereas a value of <i>true</i> indicates the Loop object is not performing active control. |
| update_interval | 118 | Unsigned | RO | RAM | indicates the interval in milliseconds at which the loop algorithm updates the output (present_value). |
| output_units | 82 | BACnet Eng. Units | RO | NRAM | indicates the engineering unit for the output of the object. |
| manipulated_variable_reference | 60 | BACnet Object Property Reference | RO | NRAM Analog Output 1; present-value | indicates the object controlled or manipulated by the Loop object. This value will always indicate Analog Output 1; present-value. |
| controlled_variable_reference | 19 | BACnet Object Property Reference | RO | NRAM Analog Input 1; present-value | indicates the object used as the control variable for the Loop object. This value will always indicate Analog Input 1, present-value. |
| controlled_variable_value | 21 | Real | RO | NRAM | indicates the value of the property defined in the controlled_variable_reference property. |
| controlled_variable_units | 20 | BACnet Eng. Units | RO | NRAM | indicates the engineering units for the controlled_variable_value . |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|------------------------------------|--------------|--------------------------|--------|-------------------|--|
| setpoint_reference | 109 | BACnet SP Ref. | RO | NRAM | indicates the setpoint object reference for the control loop. |
| setpoint | 108 | Real | RW | NRAM | specifies the loop setpoint used by the Loop object to perform PID Control. |
| action | 2 | BACnet Action | RW | NRAM | defines whether or not the Loop object operates in a DIRECT or REVERSE acting manner. 0 = Direct 1 = Reverse |
| proportional_constant | 93 | Real | RW | NRAM | defines the value of proportional gain used by the loop. |
| proportional_constant_units | 94 | BACnet Eng. Units | RO | NRAM | defines the units of measure for the value in proportional-constant. |
| integral_constant | 49 | Real | RW | NRAM | defines the value of integral gain used by the loop. |
| integral_constant_units | 50 | BACnet Eng. Units | RO | NRAM | defines the unit of measure for the value in integral-constant. |
| derivative_constant | 26 | Real | RW | NRAM | defines the value of derivative gain used by the loop. |
| derivative_constant_units | 27 | BACnet Eng. Units | RO | NRAM | defines the unit of measure for the value in derivative-constant. |
| bias | 14 | Real | RW | NRAM | defines the amount of biasing used by the loop algorithm for startup control sequences. |
| maximum_output | 61 | Real | RO | RAM | defines the maximum value of the present-value as limited by the PID loop algorithm. |
| minimum_output | 68 | Real | RO | RAM | defines the minimum value of the present-value as limited by the PID loop algorithm. |
| priority_for_writing | 88 | Unsigned | RO | NRAM | defines the BACnet Priority Level used to control the object referenced in controlled-variable-reference, provided that the object is commandable and contains a priority array. |
| time_delay | 113 | Unsigned | RW | NRAM | defines the minimum time period, in seconds, that the different between the setpoint and the controlled-variable-value must remain outside of the error band before a notification is generated. |
| notification_classes | 17 | Unsigned | RO | NRAM | defines the Notification Class object instance in which notifications for the Loop object are processed |
| error_limit | 34 | Real | RW | NRAM | defines the absolute limit of error difference that may exist between the setpoint and controlled-variable-value properties before a notification is generated. |
| event_enable | 35 | BACnet Event Trans. Bits | RW | NRAM | three flags that separately enable and disable reporting of TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|--------------------------|--------|-------------------|--|
| acked_transitions | 0 | BACnet Event Trans. Bits | RO | RAM | three flags that separately indicate the receipt of acknowledgments for TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events. |
| notify_type | 72 | BACnet Notify Type | RW | NRAM | specifies whether the notifications generated by the object should be Events or Alarms. |
| PO | 53327 | Real | RO | NRAM | Percent Out specifies the current percent output of the PID Control Loop. |
| CS | 50003 | Real | RO | NRAM | Calculated Control Setpoint specifies the current control setpoint (after any applied setback) used by the PID Control Loop. |
| F1 | 50737 | Real | RW | RW | FeedForward Gain 1 specifies the calculation value for FeedForward Gain 1 for Sash Position Applications. |
| F2 | 50738 | Real | RW | RW | FeedForward Gain 2 specifies the calculation value for FeedForward Gain 2 for Sash Position Applications. |
| FB | 50754 | Real | RW | RW | FeedForward Bias specifies the amount of biasing used between the two gain calculations for sash positioning. |
| SU | 54101 | Real | RW | RW | Unoccupied Setup/Setback specifies the amount of setback applied to the setpoint during states when the fume hood is not occupied. |
| ER | 50514 | Real | RO | RO | Absolute Error provides the calculated absolute error between the setpoint and controlled-variable-value. |
| DL | 50252 | Real | RO | RO | Demand Load provides the calculated demand/different between the controlled-variable-value and setpoint. |

A.11 STATBUS 1, 3 AND 4

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|----------------|--------|---|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (255), Instance 1, 3, and 4 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM STATbus N | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Proprietary (255) | indicates membership in a particular object type class. |
| DN | 50254 | CharStr | RW | NRAM " " | Description user defined description of the object and it's functionality. |
| BS | 49747 | Unsigned | RO | RAM 0 | Bus Status indicates the status of the SMARTStat Bus. 0=Ok 1=Needs configured 2=Error |
| CD | 49988 | Unsigned | RW | RAM 0 | Configure Device (GID) specifies the device to be configured. |
| CF | 49990 | Unsigned | RW | RAM 0 | Configure Function enables the blinking LED on the device specified in CD . 0=Disabled 1=Blink LED |
| G1 | 50993 | Unsigned | RW | NRAM 0 | GID Device 1 indicates the global identification number of device 1. |
| G2 | 50994 | Unsigned | RW | NRAM 0 | GID Device 2 indicates the global identification number of device 2. |
| G3 | 50995 | Unsigned | RW | NRAM 0 | GID Device 3 indicates the global identification number of device 3. |
| G4 | 50996 | Unsigned | RW | NRAM 0 | GID Device 4 indicates the global identification number of device 4. |
| G5 | 50997 | Unsigned | RW | NRAM 0 | GID Device 5 indicates the global identification number of device 5. |
| G6 | 50998 | Unsigned | RW | NRAM 0 | GID Device 6 indicates the global identification number of device 6. |
| G7 | 50999 | Unsigned | RW | NRAM 0 | GID Device 7 indicates the global identification number of device 7. |
| G8 | 51000 | Unsigned | RW | NRAM 0 | GID Device 8 indicates the global identification number of device 8. |
| G9 | 51001 | Unsigned | RW | NRAM 0 | GID Device 9 indicates the global identification number of device 9. |
| GA | 51009 | Unsigned | RW | NRAM 0 | GID Device 10 indicates the global identification number of device 10. |
| GB | 510010 | Unsigned | RW | NRAM 0 | GID Device 11 indicates the global identification number of device 11. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|--|
| GC | 51011 | Unsigned | RW | NRAM 0 | GID Device 12 indicates the global identification number of device 12. |
| GD | 51012 | Unsigned | RW | NRAM 0 | GID Device 13 indicates the global identification number of device 13. |
| SM | 54093 | BitStr | RO | RAM 0 | Status Map 1=unconfigured 2=duplicate |

A.12 SMALL DISPLAY CONFIG

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|----------------|--------|--|--|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (255), Instance 16 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Small Display Config | 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. |
| DN | 50254 | CharStr | RW | NRAM " " | Description user defined description of the object and it's functionality. |
| PE | 53317 | Unsigned | RW | NRAM | Password Enabled |
| O1 | 53041 | Unsigned | RW | NRAM | Opr 1 Password specifies the password for Operator 1 |
| O2 | 53042 | Unsigned | RW | NRAM | Opr 2 Password specifies the password for Operator 2 |
| O3 | 53043 | Unsigned | RW | NRAM | Opr 3 Password specifies the password for Operator 3 |
| O4 | 53044 | Unsigned | RW | NRAM | Opr 4 Password specifies the password for Operator 4 |
| P1 | 53297 | Unsigned | RW | NRAM | Opr 1 Privileges specifies the privilege level for Operator 1 0 = No Privileges 1 = Level 1 2 = Level 2 |
| P2 | 53298 | Unsigned | RW | NRAM | Opr 2 Privileges specifies the privilege level for Operator 2 0 = No Privileges 1 = Level 1 2 = Level 2 |
| P3 | 53299 | Unsigned | RW | NRAM | Opr 3 Privileges specifies the privilege level for Operator 3 0 = No Privileges 1 = Level 1 2 = Level 2 |
| P4 | 53300 | Unsigned | RW | NRAM | Opr 4 Privileges specifies the privilege level for Operator 4 0 = No Privileges 1 = Level 1 2 = Level 2 |
| AL | 49484 | Unsigned | RW | NRAM | Silence Time Specifies the time in which alarms remain silent when a user presses the silence soft button. |
| AS | 49485 | Unsigned | RW | NRAM | Alarm Silence |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|--------------------------|---|
| HS | 51283 | Unsigned | RW | NRAM | Default Home Screen specifies the appearance for the home screen of the FHC-SD. 0 = Default 1 = With Setpoint 2 = With Small Text 3 = With Large Text |
| HT | 51284 | CharStr | RW | NRAM | Home Screen Text provides a custom text string, displayed on the Home Screen when the HS is configured to display text. |
| ST | 54100 | Unsigned | RW | NRAM | Session Timer indicates the amount of time, in minutes, an operator session can occur. Once the time elapses, the user will be automatically logged off. |
| SN | 54094 | Unsigned | RO | RAM .. | Small Display Serial Number indicates the factory serial number of the connected FHC-SD unit. |
| CT | 50004 | Unsigned | RO | RAM 108 | Small Display Controller Type indicates the controller type identification number of the FHC-SD unit connected to the NB-FHC controller. |
| FT | 50772 | Unsigned | RO | RAM 40 | Small Display Firmware Type indicates the current firmware type of the FHC-SD unit connected to the NB-FHC controller. |
| OS | 53075 | Real | RO | NRAM x.xx | Small Display Kernel (boot block) indicates the current kernel operating system revision used within the FHC-SD unit. |
| VE | 54853 | Real | RO | NRAM x.xx | Small Display Firmware Version indicates the current firmware revision of the FHC-SD unit connected to the NB-FHC controller. |
| SR | 54098 | Unsigned | RO | NRAM xxxxxxxxx | Small Display Software Time Stamp indicates the software time stamp for the firmware of the FHC-SD. |
| FC | 50755 | Unsigned | RO | NRAM 0 (when shipped) | Small Display Flash Update Counter indicates how many times the FHC-SD has been flash updates with a different revision of firmware. |
| WC | 55107 | Unsigned | RO | NRAM 0 | Small Display Watchdog Counter a troubleshooting parameter used to detect network communication problems between the FHC-SD and the NB-FHC. |
| OC | 53059 | Unsigned | RO | NRAM 0 | Small Display Illegal Opcode Counter a troubleshooting parameter used to detect network communication problems between the FHC-SD and the NB-FHC. |
| IC | 51523 | Unsigned | RO | NRAM 0 | Small Display Initialization Counter indicates how many times that the FHC-SD has been commanded back to factory default values by user intervention. |
| CF | 49990 | BitStr | RO | NRAM | Small Display Control Flags a bitstring value indicating what modes or actions have occurred in the FHC-SD touch-screen display. |

A.13 UNIVERSAL INPUT SUMMARY

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|----------------|--------|---------------------------------------|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (254), Instance 0 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Universal Input Summary | 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. |
| DN | 50254 | CharStr | RW | NRAM " | Description user defined description of the object and it's functionality. |
| AT | 49492 | Unsigned | RW | NRAM 0 | Alarm Limit setup/setback time delay the time (in minutes) that must elapse after transitions from occupied to unoccupied or from unoccupied to occupied before alarm limits return to their previous (pre-setup/setback) values. |
| IP | 51536 | BitStr | RW | NRAM 0 | Input Polarities specifies how the controller represents the input internally. 0=Normal polarity. Low Voltage is displayed as present_value =0 and High Voltage is displayed as present_value =1. 1=Reverse polarity. Low Voltage is displayed as present_value =1 and High Voltage is displayed as present_value =0. IP is a bitmap with bit 0 corresponding to UI1, bit 1=UI2, etc. |
| OI | E9 | RW | NRAM | 0 | Overridden Inputs allows the current value of the corresponding input, V1 through VO, to be overridden. 0=Normal 1=Override OI is a bitmap with bit 0 corresponding to UI1, bit 1=UI2, etc. |
| RE | 53829 | BitStr | RO | RAM | Inputs with Unreliable objects indicates whether the reading from the corresponding input is considered reliable. 0=Reliable 1=Unreliable RE is a bitmap with bit 0 corresponding to UI1, bit 1=UI2, etc. |
| V1 | 54833 | Real | RW | RAM | Current Measured Input 1 Value indicates the present_value of Universal Input 1. |
| V2 | 54834 | Real | RW | RAM | Current Measured Input 2 Value indicates the present_value of Universal Input 2. |
| V3 | 54835 | Real | RW | RAM | Current Measured Input 3 Value indicates the present_value of Universal Input 3. |
| V4 | 54836 | Real | RW | RAM | Current Measured Input 4 Value indicates the present_value of Universal Input 4. |
| V5 | 54837 | Real | RW | RAM | Current Measured Input 5 Value indicates the present_value of Universal Input 5. |
| V6 | 54838 | Real | RW | RAM | Current Measured Input 6 Value indicates the present_value of Universal Input 6. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|---|
| V7 | 54839 | Real | RW | RAM | Current Measured Input 7 Value indicates the present_value of Universal Input 7. |
| V8 | 54840 | Real | RW | RAM | Current Measured Input 8 Value indicates the present_value of Universal Input 8. |
| SE | 54085 | Boolean | RW | NRAM 0 | User Occupancy Override Enable specifies whether or not the user may override the schedule occupancy state. |
| ED | 50500 | Unsigned | RW | NRAM 0 | Extended Occupancy Duration specifies the time, in minutes, occupancy is extended when the stat indicates an override |
| ER | 50514 | Unsigned | RW | RAM | Extended Occupancy Remaining indicates the amount of time remaining before the controller reverts to unoccupied mode. |
| PI | 53321 | Unsigned | RW | NRAM 0 | Installer P.I.N. specifies the P.I.N. code used to enter the Installer menu. |
| PS | 53331 | Unsigned | RW | NRAM 1100 | Service P.I.N. specifies the P.I.N. code used to enter the Service menu. |
| PU | 53333 | Unsigned | RW | NRAM 0 | User P.I.N. specifies the P.I.N. code used to enter the User menu. |

A.14 DIGITAL INPUT SUMMARY

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|----------------|--------|--|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (254), Instance 48 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Digital Input Summary | 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. |
| DN | 50254 | CharStr | RW | NRAM " | Description user defined description of the object and it's functionality. |
| CV | 50006 | BitStr | RW | RAM | Current Digital Input Value indicates the current state of the associated digital input. CV is a bitmap with bit 0 corresponding to Digital Input 1, bit 1=Digital Input 2, etc. |
| IP | 51536 | BitStr | RW | NRAM 0 | Input Polarities specifies how the controller represents the input internally. 0=Normal polarity. Low Voltage is displayed as present_value =0 and High Voltage is displayed as present_value =1. 1=Reverse polarity. Low Voltage is displayed as present_value =1 and High Voltage is displayed as present_value =0. IP is a bitmap with bit 0 corresponding to Digital Input 1, bit 1=Digital Input 2, etc. |
| OI | 53065 | BitStr | RW | NRAM 0 | Overridden Digital Inputs Bitmap indicates those inputs whose out_of_service property is true. OI is a bitmap with bit 0 corresponding to Digital Input 1, bit 1=Digital Input 2, etc. |
| RE | 53829 | BitStr | RO | RAM | Inputs with Unreliable Values indicates whether the reading from the corresponding digital input is considered reliable. 0=Reliable 1=Unreliable RE is a bitmap with bit 0 corresponding to Digital Input 1, bit 1=Digital Input 2, etc. |

A.15 ANALOG OUTPUT SUMMARY

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|-------------------|--------|---------------------------------------|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (253), Instance 0 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Analog Output Summary | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Proprietary (253) | indicates membership in a particular object type class. |
| DN | 50254 | CharStr | RW | NRAM .. | Description user defined description of the object and it's functionality. |
| AM | 49485 | BitStr | RW | RAM | Outputs in Automatic Mode specifies how the analog output is controlled. 0=Manual 1=Automatic AM is a bitmap with bit 0 corresponding to AO 1, bit 1=AO 2, etc. |
| V1 | 54833 | Real | RW | RAM | Current Value for Output 1 indicates the present_value of output 1. |
| V2 | 54834 | Real | RW | RAM | Current Value for Output 2 indicates the present_value of output 2. |
| RE | 53829 | BitStr | RO | RAM | Outputs with Unreliable States indicates whether the reading from the corresponding analog output is considered reliable. 0=Reliable 1=Unreliable RE is a bitmap with bit 0 corresponding to AO 1, bit 1=AO 2, etc. |

A.16 HOOD PRESENCE DETECTOR

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|----------------|--------|------------------------------------|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (252), Instance 1 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Hood Presence Detector | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | Proprietary (252) | indicates membership in a particular object type class. |
| DN | 50254 | CharStr | RW | NRAM " " | Description user defined description of the object and it's functionality. |
| HP | 51280 | Unsigned | RO | NRAM - | Hood Presence Detector Input Status indicates the current value of the input selected for Hood Presence. |
| PT | 53332 | Unsigned | RW | NRAM 5 | Hood Presence Duration specifies the time, in minutes, that hood presence occupancy will occur |
| IA | 51521 | CharStr | RW | NRAM " " | Hood Presence Input Channel specifies the property associated with the object specified in IC to be used for occupancy detection. |
| IC | 51523 | CharStr | RW | NRAM " " | Hood Presence Input Attribute specifies the object of the input to be used for occupancy detection. |
| PD | 53316 | Unsigned | RW | NRAM 0 | Hood Presence In Use Delay specifies the time, in minutes, that the hood presence detector status, HP , must be positive before the controller will switch to occupied mode. |
| PR | 53330 | Unsigned | RO | NRAM | Hood Presence Time Remaining indicates the time, in minutes, before the controller reverts to unoccupied mode. |

A.17 DIGITAL OUTPUT SUMMARY

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|-------------------|--------|---------------------------------------|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (251), Instance 0 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Digital Output Summary | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Proprietary (251) | indicates membership in a particular object type class. |
| DN | 50254 | CharStr | RW | NRAM .. | Description user defined description of the object and it's functionality. |
| CV | 50006 | BitStr | RW | RAM 0 | Digital Outputs which are On indicates which outputs are currently on. CV is a bitmap with bit 0 corresponding to Digital Output 1. |
| OU | 53077 | BitStr | RO | RAM | Actual Output States indicates the actual output states of the digital outputs. OU is a bitmap with bit 0 corresponding to Digital Output 1. |
| IP | 51536 | BitStr | RW | NRAM | IO Polarities indicates the value of the polarity property for each of the digital outputs. IP is a bitmap with bit 0 corresponding to Digital Output 1. |
| RE | 53829 | BitStr | RO | RAM | Outputs with Unreliable Values indicates whether the reading from the corresponding digital output is considered reliable. 0=Reliable 1=Unreliable RE is a bitmap with bit 0 corresponding to Digital Output 1. |
| ES | 50515 | BitStr | RO | NRAM | Emergency Mode Output State Bitmap specifies the output state of Digital Output 1 when Emergency Mode is entered. |

A.18 CONTROL SUMMARY

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|----------------|--------|---------------------------------------|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (250), Instance 0 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM Control Summary | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Proprietary (250) | indicates membership in a particular object type class. |
| DN | 50254 | CharStr | RW | NRAM “ “ | Description user defined description of the object and it's functionality. |
| FV | 50774 | Real | RO | RAM | Face Velocity specifies the current measured face velocity. |
| SP | 54096 | Real | RW | NRAM | Face Velocity Setpoint specifies the control setpoint for Face Velocity. |
| SB | 54082 | Real | RW | NRAM | Face Velocity Setpoint Setback specifies the amount of setback in setpoint units to use when the hood is unoccupied |
| EF | 50502 | Real | RO | RAM | Exhaust Flow specifies the current measured exhaust flow. |
| SC | 54083 | Real | RW | RAM | Sensor Coefficient specifies the sensor coefficient calibration value. |
| EE | 50501 | Unsigned | RW | NRAM | Enable Emergency indicates whether or not emergency mode is enabled or disabled 0 = Disabled 1 = Enabled |
| DC | 50243 | Unsigned | RW | NRAM | Disable Control indicates whether active control is enabled or disabled 0 = Control Enabled 1 = Control Disabled |
| DP | 50256 | Real | RW | NRAM | Disable Position specifies a value that Analog Output 1;present-value should be commanded to when control is disabled |
| IC | 51523 | CharStr | RW | NRAM | Input Object for Disable Control specifies the object code used as a status to enable/disable control. |
| IA | 51521 | CharStr | RW | NRAM | Input Property for Disable Control specifies the property code used in conjunction with IC to enable/disable control. |
| CS | 50003 | Unsigned | RW | NRAM | Control Sequence specifies which control methodology is currently selected. 0 = Auto-Flow 1 = Auto-Flow with Fuzzy Logic 2 = Auto-Flow with Sash Position 3 = Sash Position 4 = Sash Position with Shaft Pot Feedback 5 = Sash Position with Exhaust Feedback |

A.19 PIECEWISE CURVES 1-4

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|----------------|--------|---|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (247), Instance 1-4 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | - NRAM | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Proprietary (247) | indicates membership in a particular object type class. |
| DN | 50254 | CharStr | RW | NRAM " | Description user defined description of the object and it's functionality. |
| X1 | 55345 | Real | RW | NRAM 0 | Point 1's value in raw counts specifies the x coordinate of point 1. |
| X2 | 55346 | Real | RW | NRAM 0 | Point 2's value in raw counts specifies the x coordinate of point 2. |
| X3 | 55347 | Real | RW | NRAM 0 | Point 3's value in raw counts specifies the x coordinate of point 3. |
| X4 | 55348 | Real | RW | NRAM 0 | Point 4's value in raw counts specifies the x coordinate of point 4. |
| X5 | 55349 | Real | RW | NRAM 0 | Point 5's value in raw counts specifies the x coordinate of point 5. |
| X6 | 55350 | Real | RW | NRAM 0 | Point 6's value in raw counts specifies the x coordinate of point 6. |
| X7 | 55351 | Real | RW | NRAM 0 | Point 7's value in raw counts specifies the x coordinate of point 7. |
| X8 | 55352 | Real | RW | NRAM 0 | Point 8's value in raw counts specifies the x coordinate of point 8. |
| X9 | 55353 | Real | RW | NRAM 0 | Point 9's value in raw counts specifies the x coordinate of point 9. |
| XA | 55361 | Real | RW | NRAM 0 | Point 10's value in raw counts specifies the x coordinate of point 10. |
| XB | 55362 | Real | RW | NRAM 0 | Point 11's value in raw counts specifies the x coordinate of point 11. |
| Y1 | 55601 | Real | RW | NRAM 0 | Point 1's value in engineering units specifies the y coordinate of point 1. |
| Y2 | 55602 | Real | RW | NRAM 0 | Point 2's value in engineering units specifies the y coordinate of point 2. |
| Y3 | 55603 | Real | RW | NRAM 0 | Point 3's value in engineering units specifies the y coordinate of point 3. |
| Y4 | 55604 | Real | RW | NRAM 0 | Point 4's value in engineering units specifies the y coordinate of point 4. |
| Y5 | 55605 | Real | RW | NRAM 0 | Point 5's value in engineering units specifies the y coordinate of point 5. |

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|-----------|--------------|-----------|--------|-------------------|---|
| Y6 | 55606 | Real | RW | NRAM 0 | Point 6's value in engineering units specifies the y coordinate of point 6. |
| Y7 | 556017 | Real | RW | NRAM 0 | Point 7's value in engineering units specifies the y coordinate of point 7. |
| Y8 | 55608 | Real | RW | NRAM 0 | Point 8's value in engineering units specifies the y coordinate of point 8. |
| Y9 | 55609 | Real | RW | NRAM 0 | Point 9's value in engineering units specifies the y coordinate of point 9. |
| YA | 55617 | Real | RW | NRAM 0 | Point 10's value in engineering units specifies the y coordinate of point 10. |
| YB | 55618 | Real | RW | NRAM 0 | Point 11's value in engineering units specifies the y coordinate of point 11. |

A.20 SASH AREA

| Property | Identifier # | Data Type | Access | Storage & Default | Description |
|--------------------------|--------------|-------------------|--------|--|---|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (247), Instance 1-24 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | - NRAM | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Proprietary (247) | indicates membership in a particular object type class. |
| DN | 50254 | CharStr | RW | NRAM " | Description user defined description of the object and it's functionality. |
| CV | 50006 | Real | RO | RAM | Current Value (Calculated Open Sash Area) indicates the current calculated open sash area. |
| CF | 49990 | Real | RO | RAM | Calculated Face Velocity Setpoint indicates the current calculated Face Velocity setpoint. |
| CE | 49989 | Real | RO | RAM | Calculated Exhaust Flow Setpoint indicates the current calculated Exhaust Flow setpoint. |
| BA | 49729 | Real | RO | RAM | Calculated Bypass Area indicates the current calculated bypass area. |
| ST | 54100 | Unsigned | RW | NRAM | Sash Type specifies the installed sash door type on the fume hood. |
| FA | 50753 | Real | RW | NRAM | Fixed Area specifies the total fixed dedicated for the sash area |
| HL | 51276 | Real | RW | NRAM | Hood Airflow Leakage specifies the amount of hood airflow leakage determined by the balancer. |
| BH | 49736 | Real | RW | NRAM | Bypass Height specifies the total bypass height. |
| TW | 54359 | Real | RW | NRAM | Total Sash Width specifies the total sash width. |

A.21 BROADCAST 0 - 7

| Property | Identifier # | Data Type | Access | Default | Description |
|--------------------------|--------------|----------------|--------|---|--|
| object_identifier | 75 | BACnet ObjID | RO | - Proprietary (240), Instance 0 - 7 | a numeric code that is used to identify the object. |
| object_name | 77 | CharStr | RW | NRAM | represents a name for the object that is unique internetwork-wide. |
| object_type | 79 | BACnet ObjType | RO | - Proprietary (240) | indicates membership in a particular object type class. |
| BE | 49733 | Boolean | RW | NRAM 0 | Broadcast Enable specifies whether the controller should broadcast the value of CV over the network. 0=No 1=Yes |
| BZ | 49498 | Boolean | RW | NRAM 0 | Broadcast Zone/Global specifies whether the controller's broadcasts are sent to the zone or broadcast globally. 0=Zone broadcast 1=Global broadcast |
| CE | 49989 | Boolean | RW | NRAM | Enthalpy Calculation Enable specifies whether Broadcast 2 should calculate the Enthalpy based on the outside air temperature and humidity readings in Broadcast 0 and Broadcast 1. NOTE: This property exists only in the Broadcast 2 object. |
| CV | 50006 | Real | RW | RAM | Current Value indicates the current value of the referenced object specified in IC and IA. |
| IA | 51521 | CharStr | RW | NRAM 0 | Input property specifies the property associated with the object specified in IC to be broadcast. |
| IC | 51523 | CharStr | RW | NRAM | Input object specifies the input object to be broadcast. |
| RB | 53826 | Boolean | RW | NRAM 0 | Receive Broadcast? specifies that the controller should accept broadcast values for CV. |