



**KIMC**  
CONTROLS

# Digital Designer's Guide

## **Important notices**

©2010, KMC Controls, Inc.

BACstage, FullBAC and TotalControl are trademarks of KMC Controls, Inc.

WinControl XL Plus, NetSensor and the KMC logo are registered trademarks of KMC Controls, Inc.

All rights reserved. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language in any form by any means without the written permission of KMC Controls, Inc.

Printed in U.S.A.

## **Disclaimer**

The material in this manual is for information purposes only. The contents and the product it describes are subject to change without notice. KMC Controls, Inc. makes no representations or warranties with respect to this manual. In no event shall KMC Controls, Inc. be liable for any damages, direct or incidental, arising out of or related to the use of this manual.

## **KMC Controls**

P.O. Box 497

19476 Industrial Drive

New Paris, IN 46553

U.S.A.

TEL: 1.574.831.5250

FAX: 1.574.831.5252

E-mail: [info@kmccontrols.com](mailto:info@kmccontrols.com)

# Contents

## Part 1: DDC practices

|         |   |      |
|---------|---|------|
| AN0504L | Connecting inputs and outputs to KMC controllers..... | 1-3  |
| AN0604D | Tips for connecting 24-volt power.....                | 1-15 |

## Part 2: Networking connectivity

|         |   |      |
|---------|---|------|
| AN0903A | Building Controls & Your Ethernet Network .....                   | 2-3  |
| AN0404A | Planning BACnet networks.....                                     | 2-7  |
| AN0704A | Connecting EIA-485 network segments with fiber optic cables ..... | 2-31 |

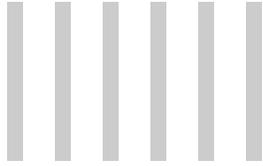
## Part 3: Job specific controllers

|         |  |       |
|---------|--|-------|
| AN0404B | KMD-7001 and KMD-7051 VAV Controller.....              | 3-3   |
| AN0404C | KMD-7002 and KMD-7052 VAV Controller.....              | 3-29  |
| AN0905A | KMD-7003 and KMD-7053 VAV Controller.....              | 3-43  |
| AN0404D | KMD-7011 VAV Controller .....                          | 3-67  |
| AN0404E | KMD-7013 VAV Controller .....                          | 3-89  |
| AN0404F | KMD-7101/02 Fan Coil Unit Controller.....              | 3-109 |
| AN0404G | KMD-7301 AHU Controller.....                           | 3-127 |
| AN0404H | KMD-7302 Roof Top Unit Controller.....                 | 3-143 |
| AN0404I | KMD-7401 Heat Pump Controller.....                     | 3-157 |
| AN0905B | Using the NetSensor on preprogrammed controllers ..... | 3-171 |
| AN0805A | The NetSensor VAV balancer program.....                | 3-175 |

## Part 4: Program modules

|         |  |      |
|---------|--|------|
| AN0504A | Dial-out program for KMD-5501 .....                              | 4-3  |
| AN0504B | Calculating enthalpy .....                                       | 4-9  |
| AN0504C | Controlling a tri-state valve actuator from two outputs .....    | 4-13 |
| AN0504E | Examples of lead/lag programs .....                              | 4-19 |
| AN0604A | Optimal start/stop routine.....                                  | 4-27 |
| AN0909A | A simplified optimum start program .....                         | 4-31 |
| AN0504F | Programming override timers .....                                | 4-39 |
| AN0504G | Working with pulse inputs .....                                  | 4-43 |
| AN0504H | Calculating boiler reset temperature.....                        | 4-45 |
| AN0604B | Calculating discharge reset based on space temperature .....     | 4-53 |
| AN0504I | Signal conditioning .....  | 4-55 |
| AN0504J | Example programs using tables .....                              | 4-59 |
| AN0604C | Monitoring refrigeration systems for moisture and flash gas..... | 4-65 |
| AN1009A | Determining occupancy with motion sensing.....                   | 4-75 |
| AN0410A | Measuring outside air temperature .....                          | 4-83 |





# Part 1

## DDC practices

The application notes in this part of the Digital Designers Guide covers topics that are common to connecting all KMC Controllers to HVAC systems and components.

---

|  |      |
|--|------|
| AN0504L Connecting inputs and outputs to KMC controllers ..... | 1-3  |
| AN0604D Tips for connecting 24-volt power .....                | 1-15 |



## Application Note—AN0504L

---

Revision A

# Connecting inputs and outputs to KMC controllers

This application note describes methods to connect input and output devices to KMC digital controllers.

---

### Related materials

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ Installation and operation guides for equipment connecting to the controller
- ◆ User's manuals for Acuity, BACstage or WinControl XL Plus
- ◆ System drawings
- ◆ Installation and operation guides for equipment to which the KMC controllers will be connected

### Topics covered in this application note

The information covered in this application note describes various methods for connecting a KMC controller—either a KMD series or BACnet model—to external equipment and devices. The examples in the illustrations show connections to a KMD-5801 controller; exact connecting points for other KMC models can be found in the installation and operation guides for those controllers.

- ◆ [Connecting inputs](#) on page 1-4. Active voltage inputs and passive inputs are both described in this section.
- ◆ [Connecting outputs](#) on page 1-8. This section includes directly connecting devices and using the HPO-6700 series output boards.
- ◆ [Current loop inputs and outputs](#) on page 1-11.

## Connecting inputs

KMC Controls designed the KMD an BAC series controllers to respond to either analog or digital signals (analog and binary signals in BACnet controllers). For the exact method to program each type of input see the on-line help and the reference manual for the software in use.

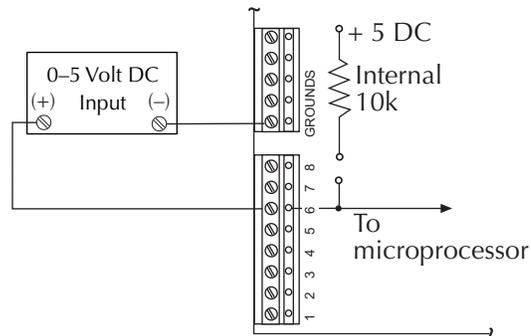
By using the optional pull-up resistors, either passive or active devices may be connected to the inputs. Active voltage inputs are described in the following section; see [Passive inputs](#) on page 1-5 for examples of thermistor, resistive and switch contact inputs.

### Active voltage source inputs

This section includes examples of active voltage inputs; current inputs are described in [Current loop inputs and outputs](#) on page 1-11.

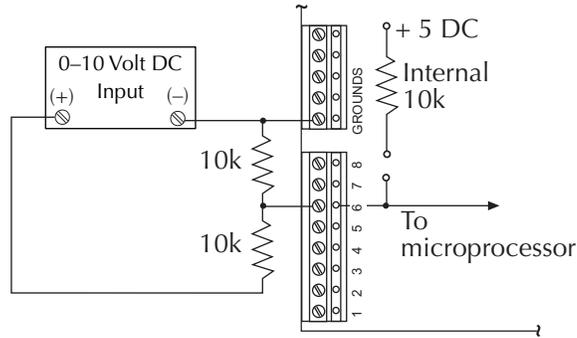
Controller inputs respond to a voltage signal which can be either an analog or digital format. To connect active devices to a universal input on a KMC controller, connect the output signal from the device—typically identified as (+), *Signal* or *Output*—to the appropriate input terminal on the controller. Connect the common or ground of the device to the nearest input ground terminal. Illustration 1 and Illustration 2 are examples of input configurations for active inputs.

The circuit in Illustration 1 is an input configuration suitable for signals up to 5 volts DC. Set pull-up resistor options for *None* or *Off*.



**Illustration 1** Voltage input from a 5-volt device

Inputs on KMC controllers are limited to input signals that are less than 5 volts DC. To use a higher voltage input, add a resistive voltage divider. Illustration 2 is an example of a voltage divider suitable for a 0–10 volt input. Set pull-up resistor options for *None* or *Off*.



**Illustration 2 Voltage divider for 0-10 volt DC input**

**Passive inputs**

Passive inputs, unlike active inputs, do not produce an active input signal. They require a voltage source that can be sensed by the controller. All KMC controllers include a pull-up resistor option which, when selected, supplies voltage to the passive device. This voltage can then be detected by the input circuit in the controller.

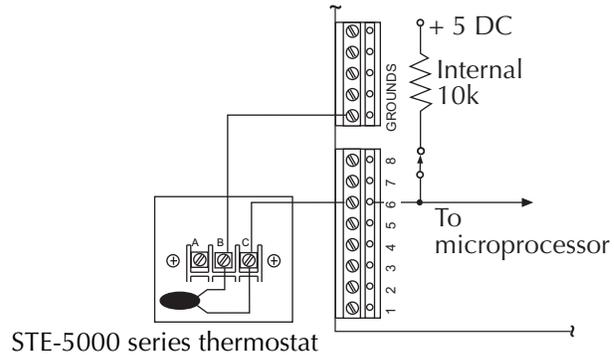
The various models of KMC controllers use different methods to select the pull-up resistor. Table 1 lists the various methods to select pull-up resistors in controllers. For the exact method to select pull-up resistors, see the installation guide supplied with the controller.

**Table 1 Pull-up resistors**

| Controller type       | Pull-up value    | Selection method  |
|-----------------------|------------------|-------------------|
| BAC-5801              | 10kΩ             | DIP switch        |
| BAC-5831              | 1k and 10kΩ      | DIP switch        |
| BAC-7000 series       | 10kΩ             | DIP switch        |
| BAC-7300 series       | 10kΩ             | DIP switch        |
| KMD-5220 input module | 250, 1k and 10kΩ | Jumper            |
| KMD-5205<br>KMD-5270  | 1k and 10kΩ      | Jumper            |
| KMD-5801<br>KMD-5802  | 10kΩ             | DIP switch        |
| KMD-5821              | 1k and 10kΩ      | DIP switch        |
| KMD-5831              | 1k and 10kΩ      | Jumper            |
| 5100 and 6000 series  | 10kΩ             | Clip out resistor |

**Thermistor input**

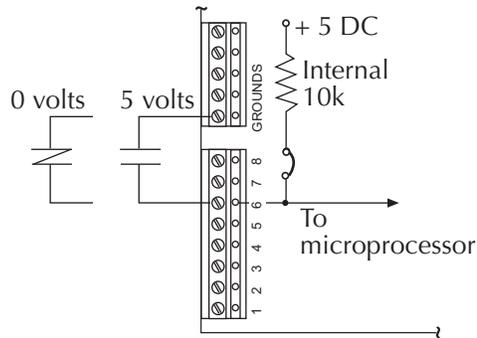
A thermistor is a passive input and must be used with a pull-up resistor. Connect thermistors between an input terminal and analog ground. For thermistors in KMC sensors, enable a 10kΩ pull-up resistor.



**Illustration 3 Thermistor input**

**Switch contact input**

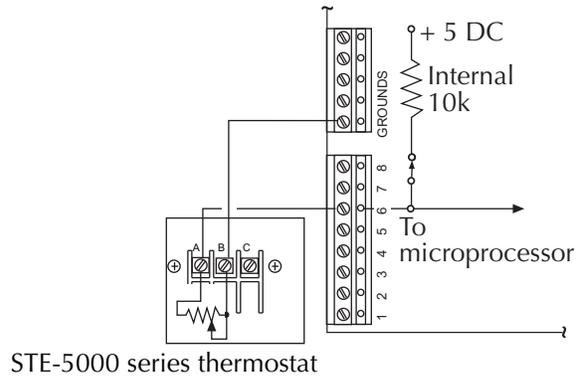
Switch and relay contacts are passive inputs and require the internal pull-up resistor. This configuration produces 5 volts for open contacts and 0 volts for closed contacts. Enable a 10kΩ pull-up resistor for switch and relay contacts.



**Illustration 4 Switch contact inputs**

**Variable resistance input**

A potentiometer connected across an input and ground will produce a voltage proportional to the potentiometer setting. Use a 10kΩ pull-up resistor for 10kΩ potentiometer.



**Illustration 5 Potentiometer input**

**Connecting outputs**

Outputs on KMC direct digital controllers are of two different types:

- ◆ Universal outputs which can connect directly to equipment that requires a DC signal
- ◆ Triac and mechanical relay outputs for AC powered equipment

To connect the output of a KMC controller to a 4–20 milliampere device, see [Current loop inputs and outputs](#) on page 1-11.

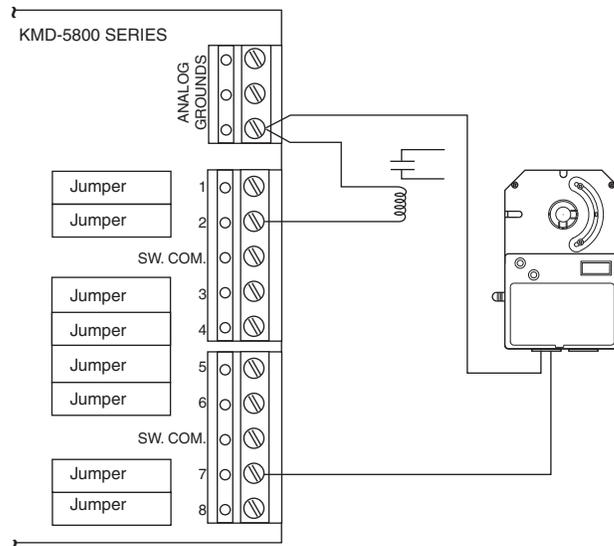
For equipment that cannot connect directly to a universal output, additional output types are available with optional HPO cards:

**Table 2 Output cards**

| Card model number | Output type                   |
|-------------------|-------------------------------|
| HPO-6701          | Triac                         |
| HPO-6702          | 0-10 volts DC analog          |
| HPO-6703          | Normally open relay contact   |
| HPO-6704          | 4-20 milliampere current loop |
| HPO-6705          | Normally closed relay contact |

**Direct connection**

For devices that can be powered within the output specification of the controller, directly connect the device to the output terminal and an *Analog Ground* terminal. Illustration 6 shows examples of connections to a 12-volt DC relay and a TSP-6001 actuator. Output 2 is programmed as a digital output in KMD series controllers and a binary object in the BACnet controllers. Output 7 is programmed as an analog output in both KMD and BAC series controllers.



**Illustration 6 Connections to analog outputs**

For manual control of the output, install a HPO-6702 output override board. In addition to manual control, an HPO-6702 provides higher output current than a universal output.

### Controlling AC devices

To control a device with 24 volts AC, use one of the following output override cards:

**Table 3 Output cards for 24 volt AC loads**

| Card model number | Output type            |
|-------------------|------------------------|
| HPO-6701          | Triac                  |
| HPO-6703          | Normally open switch   |
| HPO-6705          | Normally closed switch |

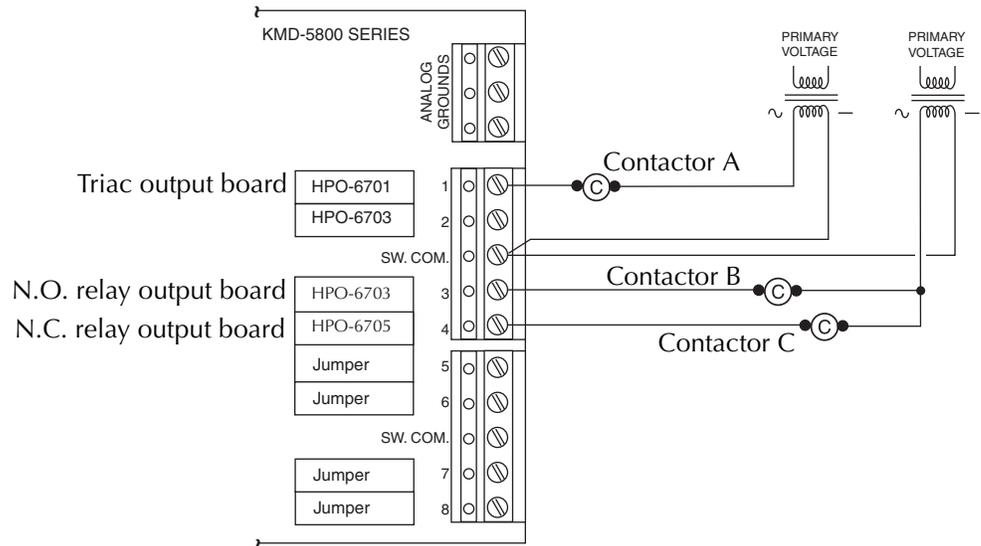
The examples in Illustration 7 show details for using an output override board to connect 24-volt contactors to a controller. *Contactors A* is powered by an external transformer and switched by a triac output board installed in output 1. The common side (–) of the transformer is connected to the *SW COM* or switch common terminal in the same bank as the output board. *Contactors B* and *C* are controlled by relay output boards. They are powered by an optional second transformer which is also connected to the *SW COM* terminal. All three of the outputs for this example are programmed as digital outputs (binary outputs for BACnet controllers).

When using output override boards, observe the following:

- ◆ Remove the jumpers from the output board location *before* connecting 24 volts or any other external signal.
- ◆ For HPO-6701, HPO-6703 or HPO-6705 use the switched common terminal in the bank in which the output board is installed.
- ◆ Verify the voltage and switching specifications in the installation guide of installed board.
- ◆ Devices powered from a common transformer must be switched by boards installed in the same bank.

**!** **Caution**

When using the HPO-6701, HPO-6703 or HPO-6705 output override boards, use only the *SW COM* (switched common) and not the *Analog Ground* terminals. The switched common terminals are isolated from the grounds used for the universal output analog circuitry in the controller. Connecting a 24-volt device to an analog ground will result in improper operation and may result in equipment damage.



**Illustration 7** Controlling 24-volt AC device



**Note**

KMD-6000 and some KMD-7000 series devices may include a triac output to switch AC loads. For these types of outputs, use the *Triac Return* terminal for the common side of the triac instead of the *Analog Ground* terminal. Exact connection details and specifications are included in the installation guide for each controller.

## Current loop inputs and outputs

Current-mode data transmission is a preferred technique for many applications, particularly in industrial settings or when a sensor must be located over 100 feet (30 meters) from the controller.

- ◆ Current mode transmitters don't require a precise or stable supply voltage.
- ◆ When installed in a remote location, their immunity to errors from voltage drop developed over long cables provides improved accuracy over voltage devices.
- ◆ A two-wire, twisted-pair cable offers good noise immunity.

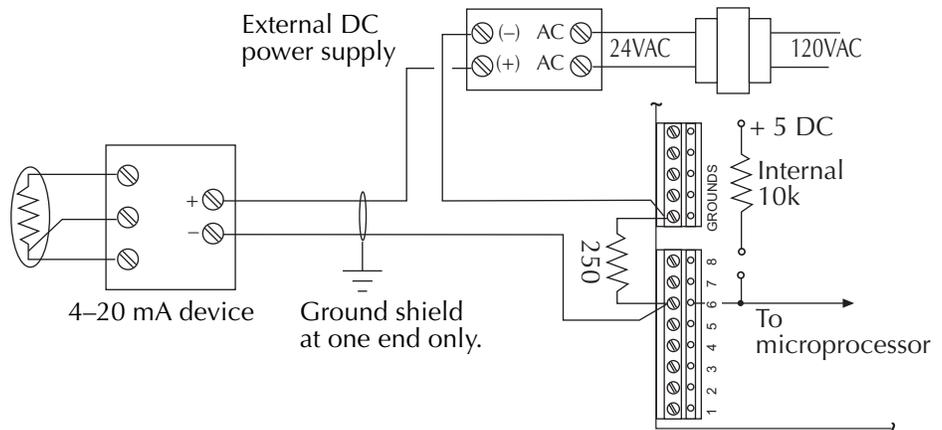
In addition to the advantages listed above, the 4 milliamperes of current not required to transfer data serves two purposes:

- ◆ It can power a remote module.
- ◆ It provides a distinction between zero (4mA) and no information (no current flow).

### 4–20 mA current source input

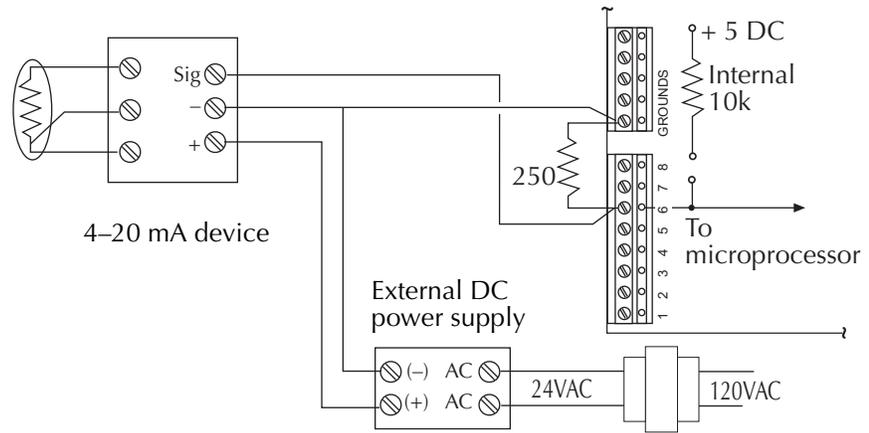
Illustration 8, Illustration 9, and Illustration 10 are examples of a 4-20 milliamper transducer connected to an analog input. A 250Ω resistor placed across the input and ground develops a voltage from the current signal. The signal is now a voltage in the range of 1–5 volts and can be sensed by the input of the controller. For both of these examples, program the input as an analog input and set the pull-up resistor option to *None* or *Off*.

**Two-wire transmitters** Two-wire transmitters, also referred to as loop powered, receive their power from an external regulated DC supply. The power supply is wired in series with the negative (-) terminal of the controller and transmitter.



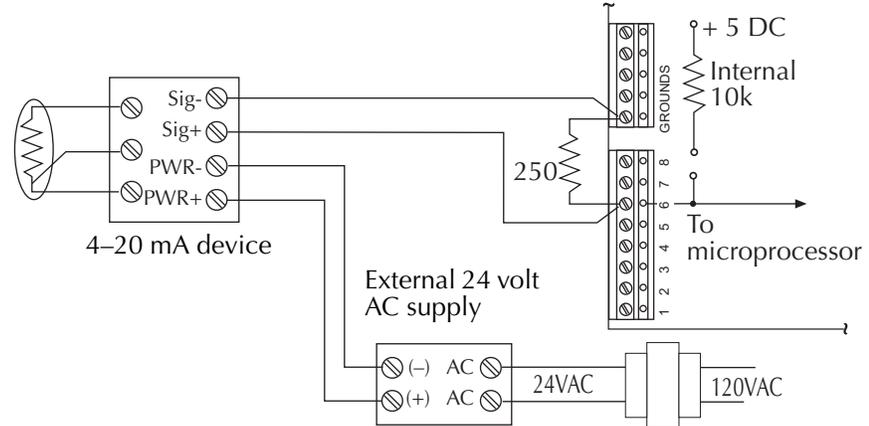
**Illustration 8** Current loop input from two-wire device

**Three-wire transmitters** The three-wire transmitter has a dedicated negative (-) terminal which is connected to both the negative terminals on both the transmitter and power supply.



**Illustration 9 Three-wire 4-20 mA current loop**

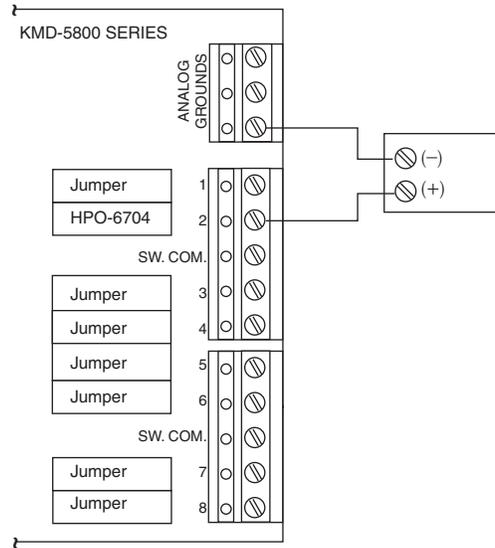
**Four-wire transmitters** The four-wire current loop is similar to a three-wire but with a negative (-) terminal for the power supply and the transmitter.



**Illustration 10 Current loop input from four-wire device**

**4-20 milliampere source output**

Install an HPO-6704 output override board to drive a 4-20 milliampere current loop load device. Connect the device between the output terminal and *Analog Ground* as shown in Illustration 11. Program the output as an analog output.



**Illustration 11 4-20 Current loop output**

For controllers that do not have space for an HPO-6704 override board, an REE-2002 from KMC Controls converts 5 volts DC to a 4-20 mA signal.



## Application Note–AN0604D

---

Revision B

### Tips for connecting 24-volt power

This application note covers choosing a transformer and connecting 24 volt AC power to a KMC Controls DDC controller.

---

Multiple factors must be considered before choosing and connecting a transformer to power a KMC controller. This application note provides information to help system designers and installers make decisions about powering controllers with transformers and then installing them correctly.



---

#### Danger

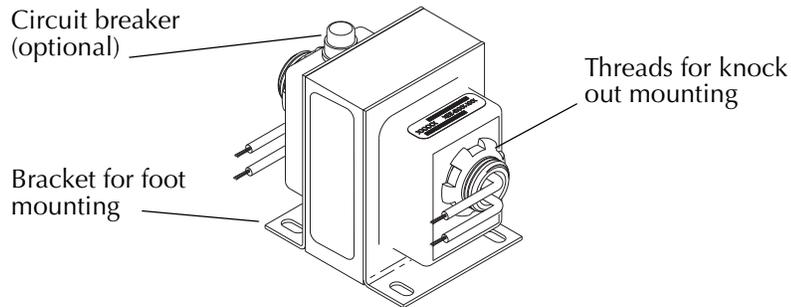
You can be killed or seriously injured if you contact live, high voltage circuits. Disconnect AC mains and follow all local lock-out and tag-out procedures when connecting or working on transformers.

#### Choosing a transformer

Most KMC controllers require a 24-volt, AC power source for operation. The recommend source for power is a UL listed, Class 2 transformer. When choosing a transformer, consider the following:

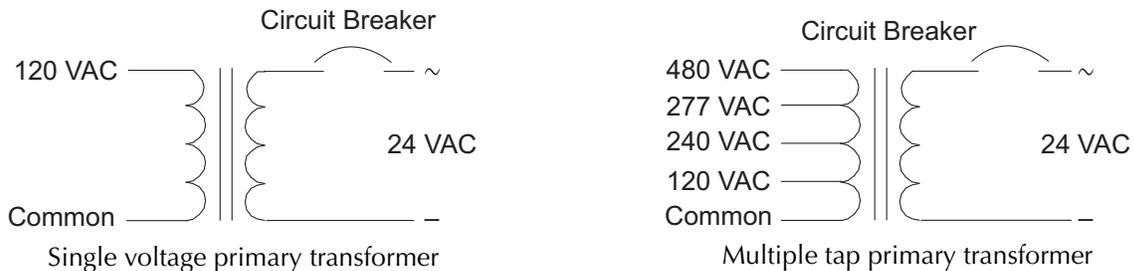
- ◆ Mounting
- ◆ Primary voltage
- ◆ Secondary voltage
- ◆ Load capacity

**Mounting** Transformers supplied by KMC Controls may be either foot or knock-out mounted. Foot mounting is usually used inside of an enclosure. Knock-out mount attaches the transformer directly to a conduit fitting such as a cover plate for a handy-box.



**Illustration 1** KMC XEE-6000 series transformer

**Primary voltage** The primary voltage rating of the transformer must match the voltage of the AC mains to which it will connect. Transformers are manufactured for either a single primary voltage such 120 or 240 volts or multiple voltages. The two XEE-6000 series transformers in Illustration 2 are examples of both types of primaries.



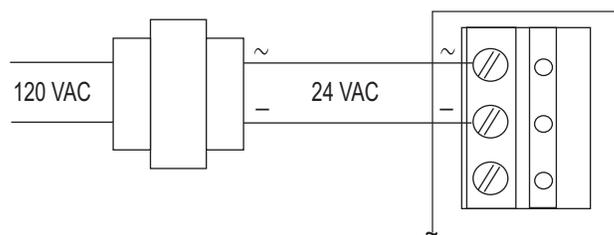
**Illustration 2** Transformer schematics

**Secondary voltage** The secondary voltage for KMC transformers is 24 volts AC. Some transformers include an optional circuit breaker which protects the transformer and wiring from current that exceeds the capacity of the circuit components.

**Load capacity** The load capacity is a measure of how much power the transformer can deliver to attached devices. This is usually expressed as VA, an abbreviation for Volt-Amperes. See [Calculating transformer size](#) on page 1-21 for additional information on transformer load capacity.

## Connecting to controllers

KMC Controls recommends supplying power to each controller from a transformer dedicated to the controller. Illustration 3 is a typical connection of a KMC XEE-6000 series transformer connected to a KMC controller. The usual practice is to connect the phase (usually blue) wire of the transformer secondary to the phase or ~ terminal and the common (usually brown) wire connected to the – or ground terminal. Other transformers may use a different color scheme. Unless required by an electrical code requirement, do not ground either lead of the secondary.

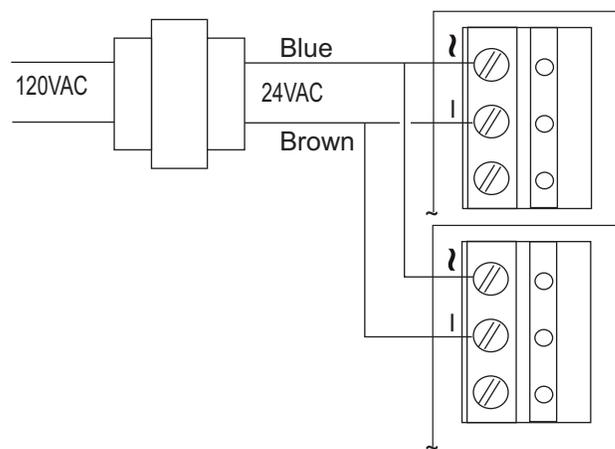


**Illustration 3** Connecting to a single controller

### Connecting several controllers to a single transformer

When more than one controller must be powered from a common transformer, connect as shown in Illustration 3. In addition, observe the following requirements:

- ◆ Connect the phase (~) terminals in parallel with other phase terminals and the commons (-) in parallel with the other common terminals. See Illustration 4.
- ◆ If the secondary must be grounded, ground it at only one location. See [Grounded secondary connections](#) on page 1-18.
- ◆ Connect only controllers with the same type of power supply. Controllers with half-wave power supplies and full-wave power supplies cannot be powered directly by the same transformer. See [Grounded secondary connections](#) on page 1-18.
- ◆ Choose a transformer with the correct VA rating. See [Calculating transformer size](#) on page 1-21.
- ◆ Choose a wire size that limits the voltage drop to no more than 5%. See [Calculating voltage drop](#) on page 1-22.



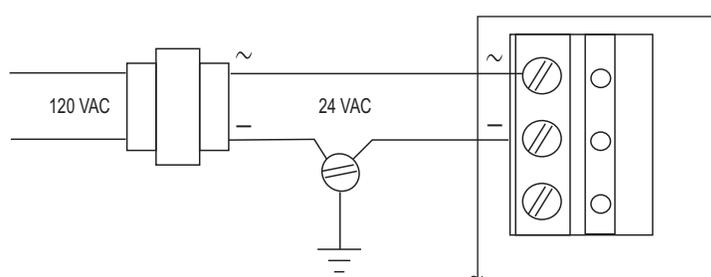
**Illustration 4** Powering two controllers from one transformer

#### Grounded secondary connections

KMC recommends that secondaries of transformers powering network devices be floating! Some conditions exist, however, that require that the secondary of a transformer be connected to earth ground. In the United States, for example:

- ◆ The National Electrical Code® requires (section 250.20) grounding transformers for AC systems of less than 50 volts if the primary voltage exceeds 150 volts to ground or if the main transformer supplying power to the building is ungrounded.
- ◆ Some municipal electrical codes more restrictive than the NEC may require that the secondaries of all transformers that supply 24 VAC be grounded.

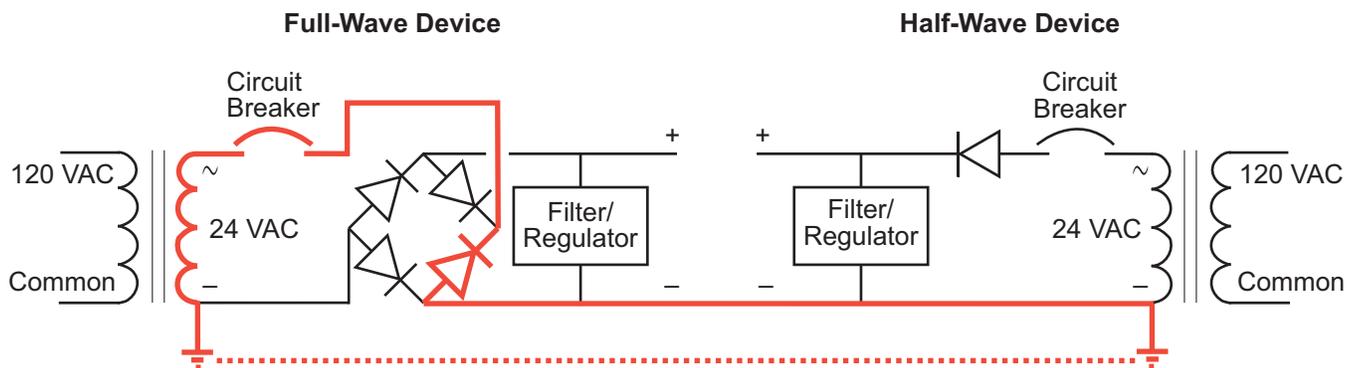
**IF a grounded secondary is mandated**, ground the common (–) wire from the transformer to an equipment grounding screw in the equipment enclosure. If more than one controller is powered from the transformer, ground both at the same ground screw.



**Illustration 5** Grounding the secondary of a transformer (when required)

If grounded transformers are used on a network containing both **full-wave** devices and **half-wave** devices, problems may occur because of the circuit differences in the power supplies. The common connection (direct or indirect) on the network and the grounds on the transformers may set up an inadvertent short circuit. In the example shown in Illustration 6 below, the transformer on the full-wave device is shorted across one of the rectifier diodes during half the AC cycle. Likely symptoms of such a situation may include:

- ◆ A circuit breaker or fuse opens.
- ◆ The power supply and/or transformer becomes overheated or damaged

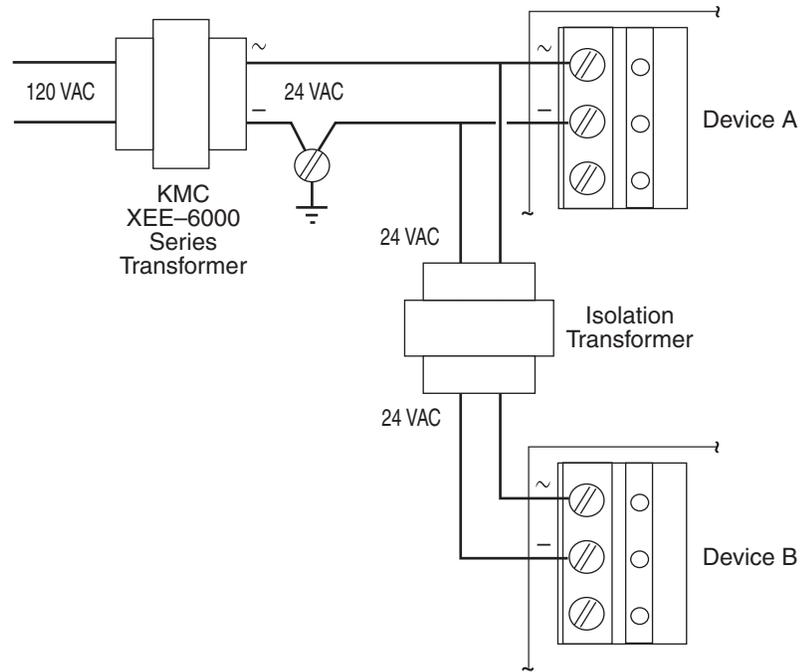


**Illustration 6** Shorting of circuit from connected (grounded) full-wave and half-wave devices

Even with grounded transformers, this should not be a problem:

- ◆ If all the devices are full-wave or all the devices are half-wave. (Nearly all currently available KMC devices are half-wave.)
- ◆ If the full-wave and half-wave device connections remain truly isolated.

**If circumstances require the use of a grounded transformer with a full-wave and a half-wave device on a network**, one possible solution may be the use of an isolation transformer. The example shown in Illustration 7 below, shows a single XEE-6000 series transformer and a 24 VAC isolation transformer. (Be sure the transformer is rated to handle the extra load—the total nominal load on a transformer should be no more than 80% of the transformer's rated value.) An isolation transformer could also be used between a transformer and a single device.



***Illustration 7 Powering a full-wave and a half-wave device from a grounded transformer***

**Grounded transformer secondaries may also become a source of a network ground loop.** A ground loop occurs in a network when a small voltage potential exists between two or more “ground” connections. This potential introduces an unwanted current in a signal path, adding noise to the network and possibly even damaging equipment. Symptoms of ground loops in KMC controller networks may include one or more of the following network communication problems:

- ◆ Misreadings of inputs.
- ◆ Panels drop off the network.
- ◆ Multiple attempts required to open data screens.
- ◆ Unable to pass network points.
- ◆ Network bulbs illuminate.

Good network planning and wiring practices will avoid most problems with ground loops and save considerable time and expense. To avoid ground loops, take care to provide only one electrical path to ground and follow good wiring practices.

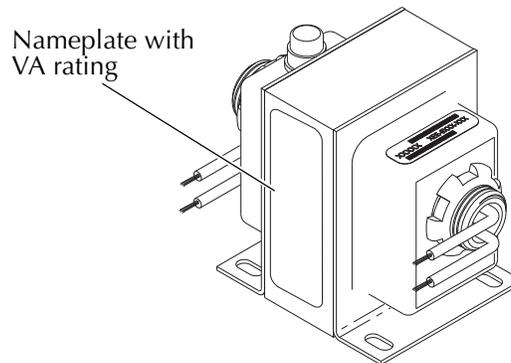
## Calculating transformer size

An inadequately sized transformer may result in erratic network problems or controller failure. The proper method to choose the correct size for a transformer is calculate the total VA load and then choose a transformer with a rating 25% higher than the total load. For example, three controllers, each rated at 10 VA can be powered from a transformer rated for 40 VA loads.

Example:

$$\begin{aligned} 10 \text{ VA} + 10 \text{ VA} + 10 \text{ VA} &= 30 \text{ VA} \\ 30 \text{ VA} \times 1.25 &= 40 \text{ VA} \end{aligned}$$

The VA rating is available from the transformer manufacturer and is usually located on the nameplate.



**Illustration 8** Typical nameplate with VA rating

## Calculating voltage drop

Inadequate voltage caused by voltage drop may result in erratic network problems or controller failure. Voltage drop occurs when power flows along a length of wire. To minimize the effects of voltage drop, connect transformers as close to their connected controllers as possible. The size of the connected load and the gauge of the connecting wire has a significant effect on an acceptable distance between the transformer and the controller.

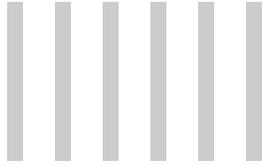
Use the following formula to calculate voltage drop based on the length and size of wire and the load presented by a controller. Limit the voltage drop at each controller to no more than 5% or 1.2 volts AC.

Example:

$$\begin{aligned} \text{Voltage drop} &= \text{Amps} \times (\text{length of circuit in ft}) \times 2 \times (\text{ohms per 1000 ft}) / 1000 \\ &= I \times L \times 2 \times R / 1000 \end{aligned}$$

**Table 1 Wire resistance**

| Wire Size—AWG, stranded | Ohms / 1000 ft (at 77°F) |
|-------------------------|--------------------------|
| 22                      | 16.46                    |
| 20                      | 10.35                    |
| 18                      | 6.512                    |
| 16                      | 4.095                    |
| 14                      | 2.576                    |
| 12                      | 1.620                    |



# Part 2

## Networking connectivity

AN0903A Building Controls & Your Ethernet Network.....2-3  
AN0404A Planning BACnet networks .....2-7  
AN0704A Connecting EIA-485 network segments with fiber optic cables...2-31



## Application Note—AN0903A

---

# Building Controls & Your Ethernet Network

**Audience:** Information technology professionals.

**Scope:** This document discusses the use of digital building control systems on existing facility Ethernet infrastructures.

**Purpose:** Understand the minimal impact that our digital product line has on your network.

---

## A Brief on Building Automation

KMC Controls designs and manufactures products intended for HVAC (heating, ventilation, air conditioning) control and related facility automation functions. These building automation systems (BAS) utilize direct digital controllers (DDC), some of which can reside on your existing Ethernet network. It is, therefore, important for you to understand the nature of the traffic created by our DDC products.

Only our Tier 1 controllers reside on the Ethernet LAN. Tier 2 controllers and devices are connected via RS-485 networking. All controllers use “points” of data (such as input from a sensor) to perform their control function. The controllers are governed by our facility management software packages (WinControl or Acuity). Finally, rest assured that all KMC control systems are installed by trained and knowledgeable KMC partners who are authorized controls contractors.

## Bandwidth Utilization

It is not possible, due to system variability, to affix a specific number here. In general terms, however, our controllers take only a minutely small portion of your Ethernet bandwidth. Some bandwidth usage is Fixed, other usage is Dynamic, and still other usage is User-Defined. With regard to Fixed usage, each controller is set to send and receive data only every 20 to 30 seconds. This traffic probably accounts for no more than 6 Kb at peak while at most times it is idle. Dynamic usage refers to alarm conditions. These exceptions occur rarely; even in large and complex automation systems. In addition, each alarm will account for only up to several hundred bytes or so of bandwidth. User defined traffic is more unpredictable. Users can set up WinControl or Acuity to have the controllers

accessed and perform any number of tasks. Again, even at peak, however, this traffic accounts for only a small percentage of your available Ethernet traffic bandwidth.

(See [Bandwidth Calculation](#) on page 2-4)

### Network Usage—A Summary

- ◆ A Network Status UDP packet (40 bytes, UDP datagram) is sent every 23 seconds to each Ethernet connected controller in the system.
- ◆ A Network Points packet (0 to 528 bytes, UDP datagram) is sent every 24 seconds to each Ethernet connected controller in the system.
- ◆ Alarms (up to 1400 bytes, UDP datagram) are sent to each Ethernet connected controller as they occur. Multiple alarms will be concatenated into one message. These same packets are directed to each operator workstation that is both logged in and set to receive such alarms.
- ◆ Operator requests (24 bytes, UDP datagram) are directed to the desired controller. Replies can be up to 1400 bytes.
- ◆ Operator sends can be up to 1400 bytes (UDP datagram); controllers acknowledge with a 15-byte UDP datagram.
- ◆ The building operator determines and controls the frequency of activity. Operator workstation refresh rates can be set as often as once per second. In addition, multiple windows may be open simultaneously. Only those windows which are in Update mode (as opposed to View mode), generate network traffic. However, workstations can also be programmed to maintain historical files known as trend logs. Workstation settings determine the number of trend logs and the frequency of updates. Trend log packets (UDP datagrams) are 1400 bytes.
- ◆ Packets transmitted on the same Ethernet sub-network are fragmented at the Maximum Transmission Unit (MTU) size set at the controller level (not related to your network MTU size). Default size is 1500 bytes.
- ◆ Routers and gateways have a tendency to “forget” our controllers on the system, deleting the controllers from routing tables. Therefore, our Tier 1 controllers will “ping” these devices every two minutes.

### Bandwidth Calculation

While not meant to be definitive or comprehensive, the following formula can provide you with a general bandwidth calculation for controller-to-controller bandwidth.

$$\text{max. bandwidth per controller} = 3000 \text{ bytes} \times (\text{number of Tier 1 controllers in system} - 1) \text{ every 25 seconds}$$

This formula does not account for Dynamic or User-Defined traffic.

## Protocols, Ports & Broadcasting

### Protocols Supported

- ◆ Internet Protocol (IP) over Ethernet
- ◆ Internet Control Message Protocol (ICMP)
- ◆ Universal Datagram Protocol (UDP)
- ◆ Address Resolution Protocol (ARP)
- ◆ BACnet (IP, MSTP, Ethernet 802.3)
- ◆ Proprietary protocol (encapsulated in UDP)

**Port Numbers Used**

Our system uses ports 21068, 21069, 21080, and 21081 to send and receive UDP packets from either our software front end or our OPC server.

Responses are directed to the requesting port number.

**Broadcasting**

- ◆ In the traditional sense of the word, KMC controllers do not perform any broadcasting functions.
- ◆ ARP messages are used whenever a given Tier 1 controller needs to determine the MAC address of other controllers. This “broadcast” ARP is of no consequence to the remainder of your Ethernet network.
- ◆ We generally recommend that our Tier 1 controllers be designated as “broadcast servers” in our facility management software. Once again, however, this traffic is only from controller to controller.
- ◆ All UDP transmissions are directed (no broadcasts).

**Security**

It is impossible to foresee all potential security risks. That said, our controllers pose little threat to your network in terms of security. Our Tier 1 controllers have no user-accessible operating system, no file structure, and no user-accessible storage media. In addition, the vast majority of our systems are built on our proprietary protocol.

Even in the worst case scenario (knowledgeable DDC specialist with access to the software front end) it is not possible to “hack into” our controllers and propagate anything to the remainder of the Ethernet network. At most, such a motivated individual could only influence the performance of a particular controller and its connected equipment.

KMC offers a Tier 1 controller, known as WebLite (KMD-5270), which incorporates a built-in HTTP server. This controller requires an IP address and can send out emails and serve up web pages (using TCP/IP and HTTP protocols). However, like our other Tier 1 controllers, it has no provision to receive and/or propagate infected files.



## Application Note—AN0404A

---

Revision B

# Planning BACnet networks

### About this application note

This application note describes the principles behind planning a BACnet installation. It is intended for control system designers and programmers who are experienced with the principles of building automation or HVAC controls but have yet to design a BACnet installation. Subjects covered in this document are:

- ◆ Creating a simple network with the MS/TP protocol.
- ◆ Assigning MAC addresses, device instance numbers and network numbers.
- ◆ Using routers to create internetworks on LANs, WANs and the internet.
- ◆ Avoiding network loops and broadcast storms.
- ◆ Integrating BACnet with the KMC Controls KMD series of products.
- ◆ The principles of MS/TP network wiring.

### Related reference materials

The examples in this application note cover only system level details and do not cover detailed information about the individual pieces of KMC equipment in the examples. Refer to the individual installation guides supplied with the equipment for exact details about each piece of equipment.

- ◆ Installation and operation guide for each advanced application controller
- ◆ BAC-5050 FullBAC Router
- ◆ KMD-5567 Suppressor Module
- ◆ KMD-5575 Network Repeater Isolator

### Required software

KMC BACnet controllers and the BAC-5050 router require the following software for configuration and programming.

- ◆ *Router Configuration Tool* to configure the BAC-5050.
- ◆ *BACstage* to configure and program KMC BACnet controllers.
- ◆ *Hardware Configuration Manager* to configure KMC Tier 1 controllers with a BACnet option.
- ◆ *WinControl XL Plus* to program a KMC Tier 1 controller with a BACnet option.
- ◆ *TotalControl* (optional) as a high level, multi-protocol user interface.

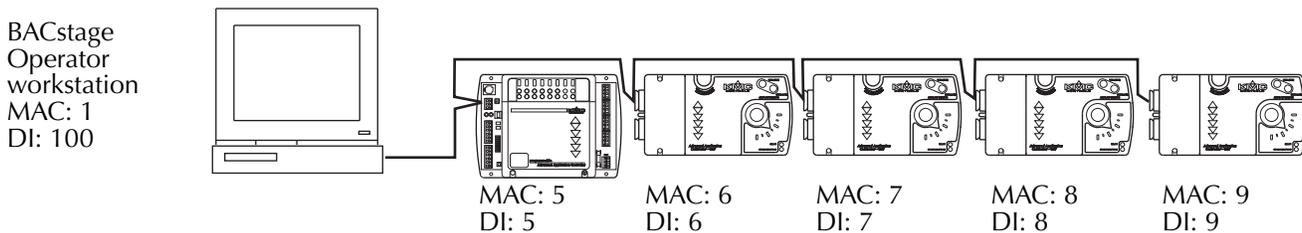
## Equipment from other manufacturers

In addition to the KMC documentation and software, if equipment from another manufacturer is planned for installation, you will need the installation manuals and recommend software for that equipment.

## Simple network

A simple network requires only two or more controllers, an operator workstation and network wiring. The example MS/TP (master slave/token passing) network in Illustration 1 connects an operator workstation, four variable air volume controllers, and a controller with eight inputs and eight outputs. Each of the devices, including the workstation, was configured with the following parameters before they were connected to the network.

- ◆ Each device is assigned a MAC address.
- ◆ Each device is assigned a device instance number.
- ◆ All devices are configured to communicate at the same baud rate.



**Illustration 1 Example MS/TP network**



Key terms!

**Network** A collection of devices that directly communicate over a shared medium or protocol without crossing a BACnet router.

**MAC address** The MAC (Media Address Control) address uniquely identifies a device on its MS/TP network.

- ◆ The MAC address must be unique across the network.
- ◆ Valid MS/TP MAC addresses are 0-127 for master devices and 0-254 for slave devices.
- ◆ The practical limit for the number of devices on a single MS/TP network is about 60. The actual number of controllers will depend upon the volume of network traffic between controllers.

**Device instance** A number that uniquely identifies a device on an internetwork.

- ◆ The device instance number must be unique across the internetwork. (Internetworks are discussed in [Creating an internetwork](#) on page 2-10.)
- ◆ Valid instance number's range from 0 to 4,194,303.
- ◆ The device instance number is assigned by the BACnet system designer. For KMC BACnet controllers it is set in the controller during configuration with BACstage.
- ◆ It is by reference to the device instance number that data is exchanged between BACnet devices.

In this simple network, *BACstage* is connected through either a serial or USB port to the MS/TP network. Its default device instance is 100 and the MAC address is 1.



#### Detail

MS/TP network principles and wiring are discussed in the section [MS/TP networking practices](#) on page 2-24.

## Adopting a numbering system

Good network planning requires a well thought out numbering scheme for device instance and MAC numbers.

### MS/TP MAC numbers

MAC address numbers are assigned to all active devices on an MS/TP network.

- ◆ If a router is present, use MAC 0 for the router's MS/TP address. See [Creating an internetwork](#) on page 2-10 for details on using routers.
- ◆ Reserve MAC addresses 1-4 for service tools such as BACstage on a technicians laptop. This will result in a small, but acceptable, performance loss when controllers on the network look for MAC addresses that may not be present.
- ◆ Begin numbering controllers at MAC address 5 and number consecutively until each controller is assigned a MAC address.

### Device instance numbers

**Small and medium systems** For simple installations, such as the network in Illustration 1, the device instance number is the same as the MAC address. For more complex systems, the device instance number is created by combining the network number (See [Creating an internetwork](#) on page 2-10 for details on network numbers) with the MS/TP MAC address.

**Larger systems** More sophisticated numbering schemes may also be used that describes the location of each piece of equipment on a campus. For this type of number system, the device instance is composed of a building number, floor number and the MAC address.

- ◆ The format of BBFFMMM assumes one MS/TP network per floor.
- ◆ BB—the first two digits indicate building number (01-41).
- ◆ FF—the second two digits indicate floor number (01-94).
- ◆ MMM—the last three digits indicate the MAC address.

### Network numbers

Number BACnet networks sequentially. Network numbers, routers and internetworks are covered in detail in a following section. The default BAC-5050 network numbers are:

- ◆ Networks 1-4 for MS/TP ports 1-4
- ◆ Network 5 for the 8802-3 port
- ◆ Networks 6-9 for IP ports 1-4.

## Creating an internetwork

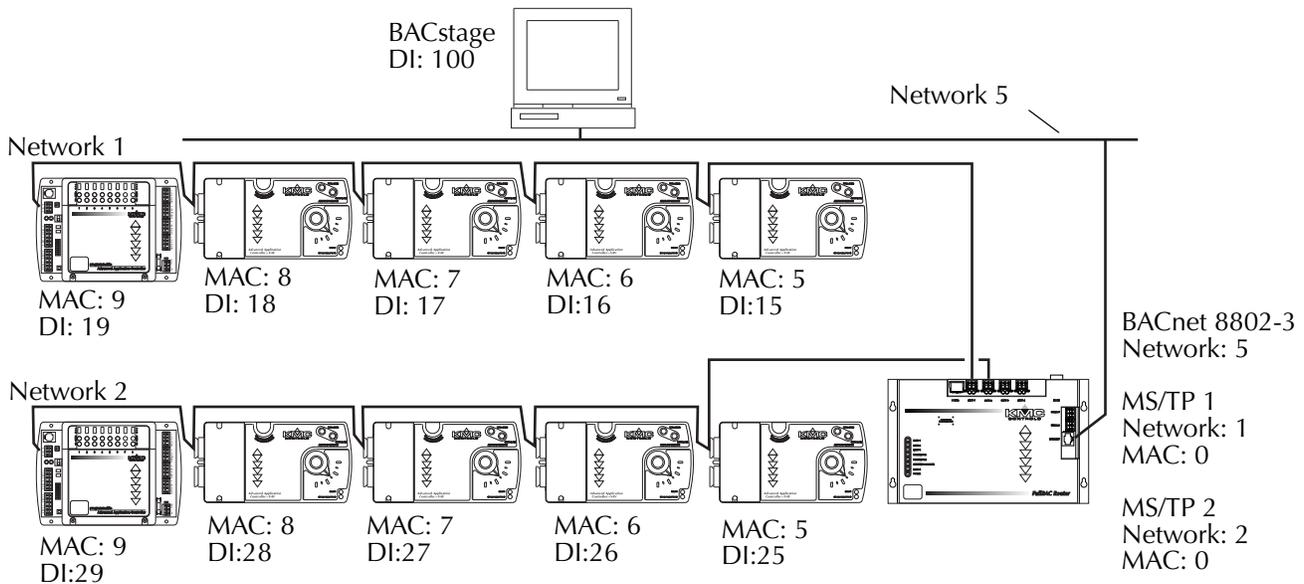
For systems that require more than 127 master devices or systems that are distributed through large buildings or connect several buildings together, a single MS/TP network is not practical and may not be possible. To overcome MS/TP network limitations, expand the system by joining two or more networks with a router. Routers may also be required when connecting to other manufacturers equipment or to TotalControl or other high-end operator workstations.



**Internetwork** Two or more networks connected by one or more routers. Each of the networks is identified with a network number.

The internetwork in Illustration 2 is composed of three separate networks joined by a BAC-5050 router.

- ◆ Network 1 connects all of the devices on the top MS/TP network to the router.
- ◆ Network 2 connects all of the devices on the bottom MS/TP network to the router.
- ◆ Network 5 connects the operator workstations with BACnet 8802-3 over the LAN.
- ◆ The router was enabled to use BACnet 8802-3 with *Router Configuration Tools*.
- ◆ The MAC addresses for the router on BACnet 8802-3 Network 5 and BACstage use the Ethernet MAC address assigned by the manufacturer.



**Illustration 2 Internetwork composed of two MS/TP networks**

In Illustration 2, the MS/TP MAC address numbers are duplicated between Network 1 and Network 2. The two devices on Network 5, the router and BACstage running on the computer, use the MAC addresses assigned to them by the manufacturer. This is the same MAC address used also by the Ethernet LAN. The BAC-5050 router does not contain objects and therefore does not require a device instance number.



**Router** A BACnet router is a device that connects two or more BACnet networks. The BACnet networks may be any of the following protocols:

- ◆ MS/TP
- ◆ BACnet IP
- ◆ BACnet 8802-3
- ◆ Point-to-point
- ◆ ARCNET
- ◆ Lonworks

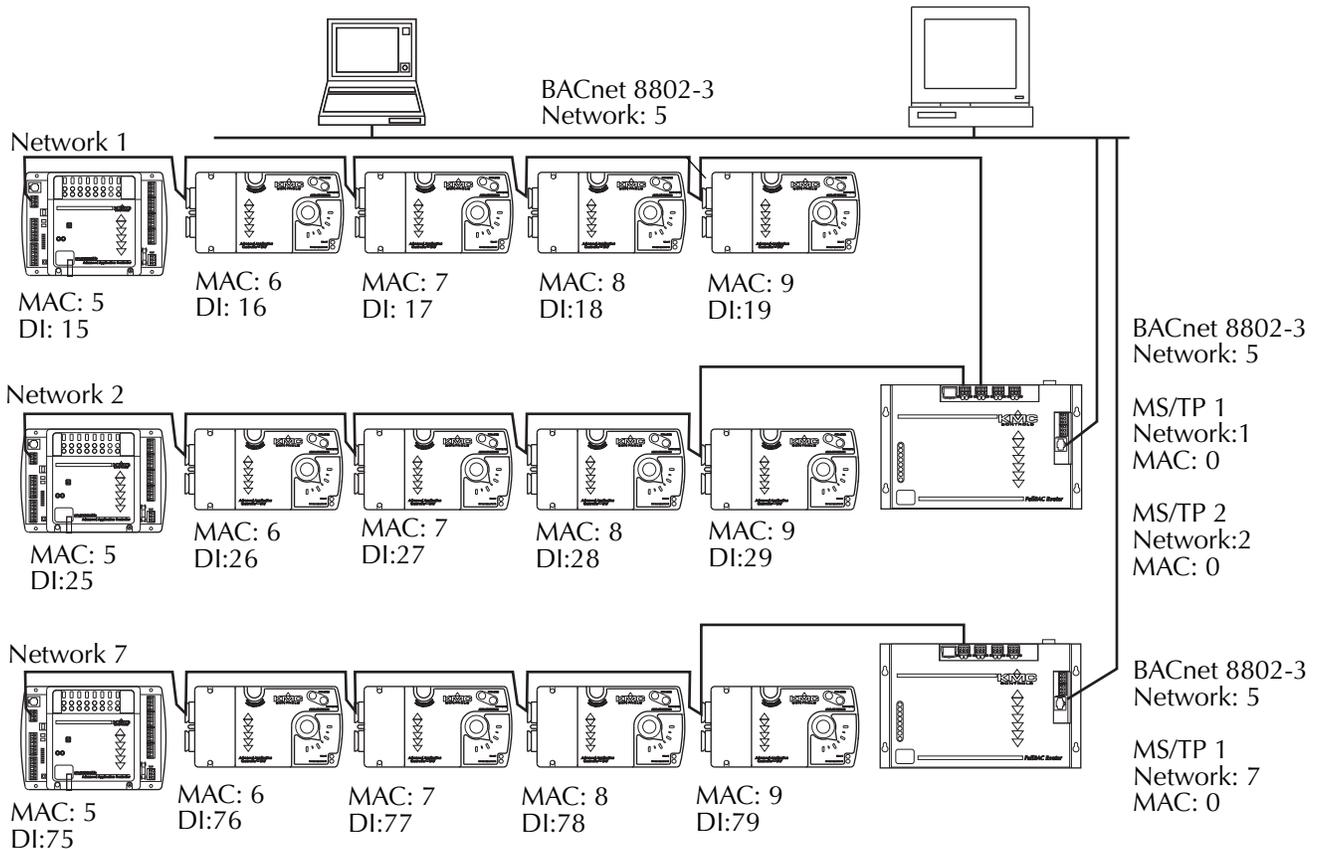
The KMC Controls BAC-5050 routes MS/TP, BACnet IP, BACnet 8802-3 and point-to-point traffic.

**Network numbers** uniquely identify a network across the entire internetwork.

- ◆ They are assigned by the system designer and are configured into the router before the router is placed on the network. For the BAC-5050 router, use *Router Configuration Tools* to assign network numbers.
- ◆ All devices on the same network use the same network number.
- ◆ Valid network numbers range from 1 to 65,534.
- ◆ Network numbers are assigned only in routers.
- ◆ BACnet routers identify devices and route traffic to them by using the combination of the network number and MAC address.

**Larger internetworks**

For systems that require multiple routers, for example one for each floor or wing of a building, the usual connection method is to use the buildings existing Ethernet LAN. The Ethernet LAN then becomes the backbone of the BACnet internetwork. The 8802-3 port in all of the routers are assigned to the same BACnet network number.



**Illustration 3 Two routers on an internetwork**

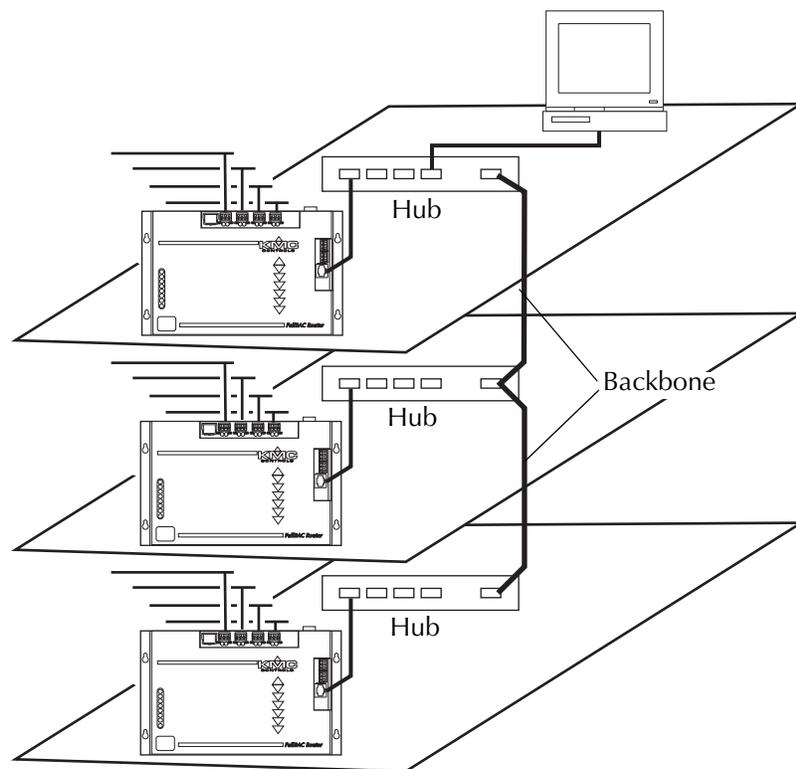
As with the one-router internetwork in Illustration 2, the MAC addresses on each of the MS/TP networks are exact duplicates. The device instance numbers are unique and continue using the numbering scheme with the network number as the first digit. Both routers in Illustration 3 were configured with the BACnet 8802-3 protocol enabled and assigned to network 5. For simple networks such as Illustration 3, all routers and the workstations must be on the same Ethernet broadcast domain.

## Using the existing LAN

The internetwork shown in Illustration 4 is a simple diagram of an internetwork for a large building. Each of the BAC-5050 routers are connected to a network hub for the floor on which the equipment is located. The hubs are connected by a backbone which may be any type of Ethernet media.

Some important things to consider about this example:

- ◆ Each of the twelve MS/TP networks require unique network numbers.
- ◆ The BACnet network connected over the building's Ethernet requires an additional network number.
- ◆ Each BAC-5050 router requires a unique IP address that is supplied by the network administrator.
- ◆ Unless a tunnel network is created, the BAC-5050 routers and hubs must be on the same Ethernet subnet and use the same subnet mask. Tunnel networks are discussed in [Crossing the Internet and IP subnetworks](#) on page 2-17.

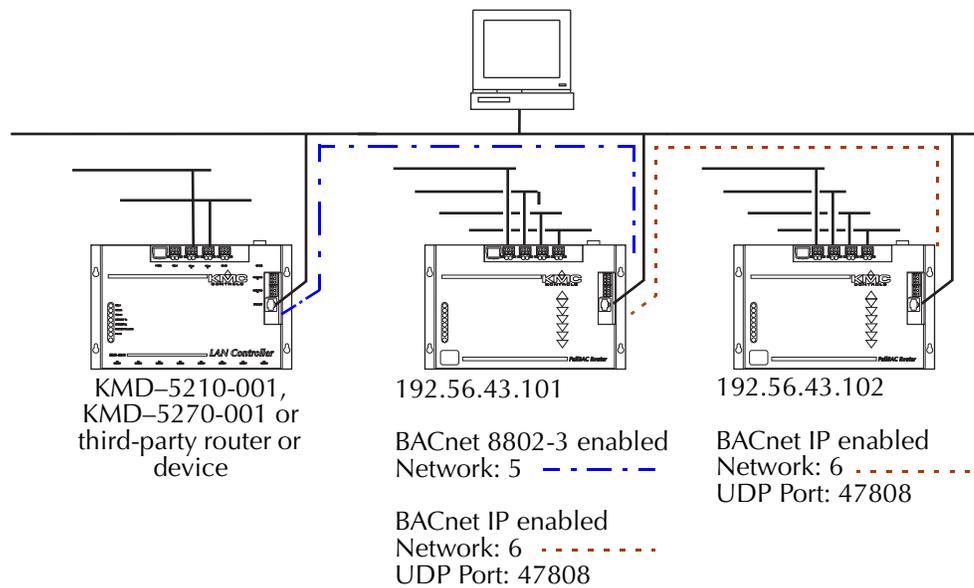


**Illustration 4** A multi-floor internetwork

The LAN backbone and hubs are usually supplied, configured and maintained by the IT department. The system administrator will supply IP addresses and subnet masks for the BAC-5050 routers.

## Mixing BACnet IP and 8802-3

Some installations, such as the expansion of an existing network, will require using both BACnet IP and BACnet 8802-3 protocols. BACnet 8802-3 is also the only protocol available in the KMD-5210-001 and KMD-5270-001 (See [Integrating BACnet with KMC digital networks](#) on page 2-23.) To use both protocols on the same internetwork, enable one router for both BACnet 8802-3 and BACnet IP. In Illustration 5, Network 5, the BACnet 8802-3 network, and Network 6, the BACnet IP network, are considered separate BACnet networks even though they connect to the same physical media, in this example the building's LAN.



**Illustration 5** BACnet IP and 8802-3 on the same LAN

New term! →

**Port** An interface on a computer, either physical or logical, to which you can connect a device. Examples of physical ports are connections for disk drives, display screens, keyboards, networks, etc. Ports may also be a logical connection on networks. For example, UDP port 80 is used for HTTP traffic.

In addition to the IP address for each router, the routers with BACnet IP enabled also require a UDP port number for each enabled BACnet IP network. Typical UDP port numbers are 47,808 to 47,817 (0xBAC0 to 0xBAC9 in hexadecimal). It is with a UDP port number that the router directs BACnet IP traffic to each of the four available BACnet IP networks on each BAC-5050.

## Avoiding loops

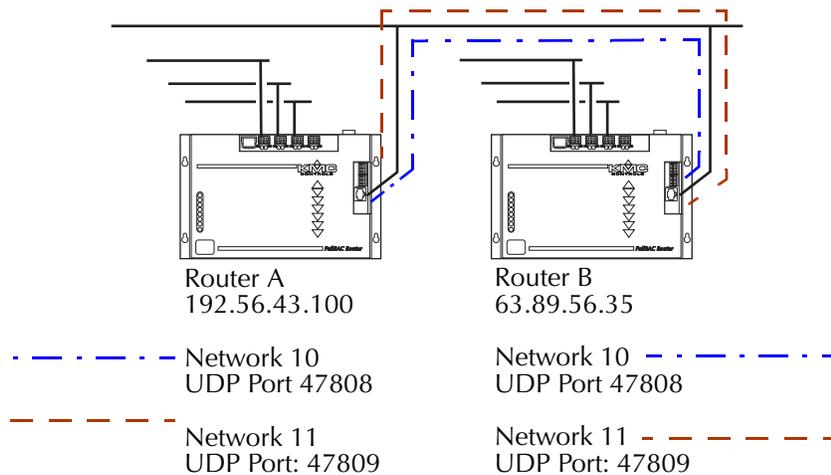
Loops create the most serious problem on an internetwork. They occur when more than one network path exists between any two devices. For example, in the internetwork shown in Illustration 6, both routers are configured with both BACnet 8802-3 and BACnet IP enabled for the same network. Network messages (shown with dashed lines) travel between the two routers over two different network paths. The first time one of the routers broadcasts a message to the network, it will start a broadcast storm of repeated messages.

In this example, the broadcast storm begins when Router A sends a broadcast message to the internetwork on the BACnet 8802-3 network. Router B receives the message and repeats it on all networks *except* the one on which it received the message. The message goes out not only on the MS/TP ports, it is also rebroadcast on the BACnet IP port for Router B. Router A then sees the message on its BACnet IP port and repeats it on all ports including the BACnet 8802-3 port where the message originated. This continues until the router becomes overloaded. Beginning with firmware release 1.4, the BAC-5050 router will detect and disable networks that create loops. Routers from other manufacturers may not detect a loop and could still create a problem.



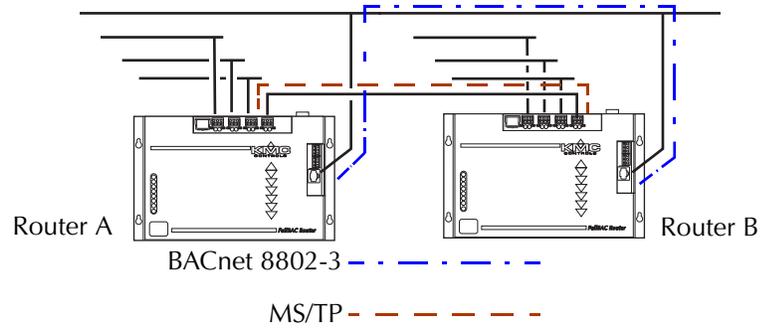
### Caution

Do not configure a network with a loop. A loop will result in a broadcast storm of repeated messages that will bring down a network. A loop forms when a path is created from a router out to one or more routers and then back to the original router without using a network number twice. Any type of network can form part of the loop.



**Illustration 6** A loop on the internetwork

A loop will start also when two routers are connected with the Ethernet LAN and MS/TP. If both routers are enabled for MS/TP and either BACnet 8802-3 or BACnet IP, a storm will start.



***Illustration 7 An MS/TP loop***

## Crossing the Internet and IP subnetworks

BACnet systems often consist of several remote networks that are joined together to form a single BACnet internetwork. Because internet access is widely available, it is a preferred method to join these remote networks. There are some limitations when an internetwork crosses IP subnetworks.

Special handling is required because the BACnet protocol uses broadcasting as a means to quickly send a message to several devices at one time. However, to minimize the impact on network traffic, most system administrators will block broadcast messages from crossing IP routers but will allow broadcasting within the subnetwork. This prevents key BACnet services such as "Who Is" and time synchronization broadcasts from reaching all parts of a BACnet internetwork.

To work around this limitation two methods are available using KMC products.

- ◆ Crossing IP subnets with tunnel routing.
- ◆ Crossing IP subnets with a BBMD described on page 19.

### Crossing IP subnets with tunnel routing

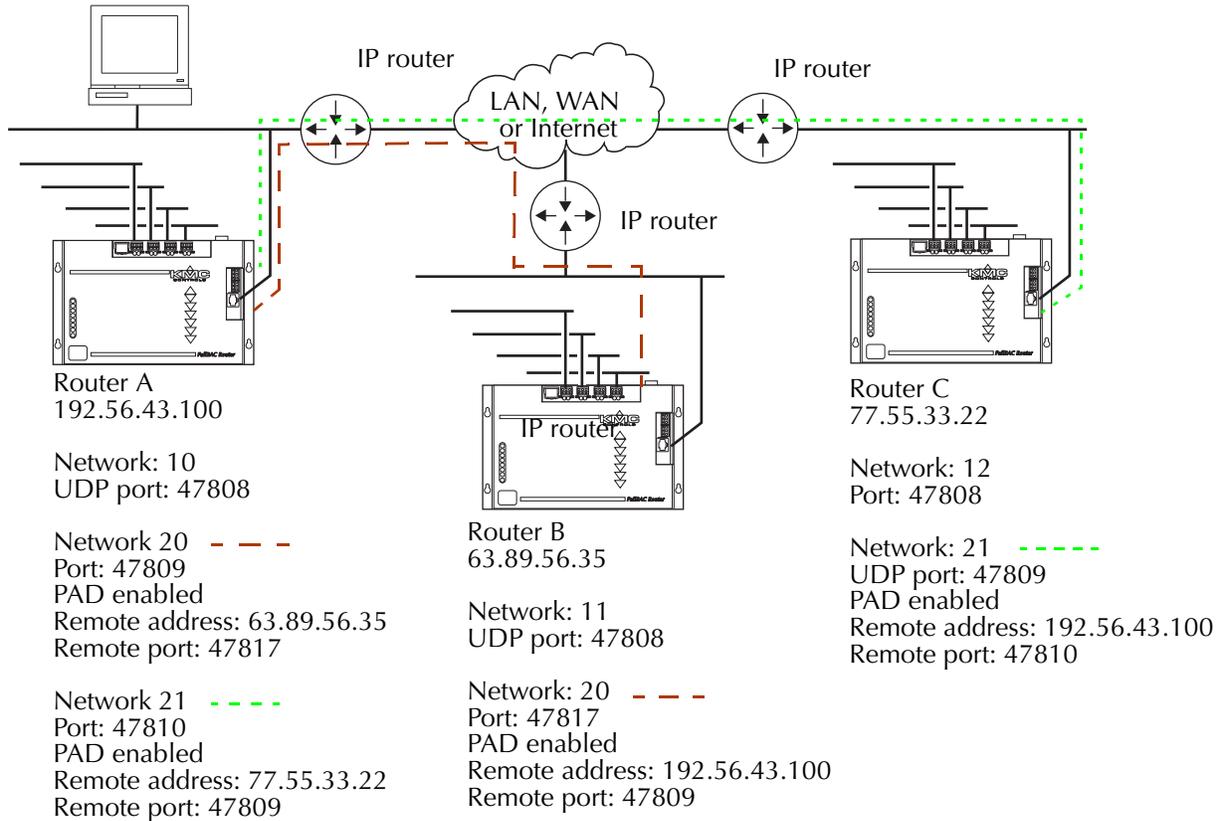
One method of extending BACnet internetworks across IP subnetworks is to use a pair of BACnet PAD routers to form a tunnel network. A pair of PAD routers pass BACnet messages between each other that are blocked by a network router.



**PAD router** A BACnet IP PAD is a special type of router that connects two or more BACnet networks that are separated by at least one IP-only router. The PAD router monitors network traffic for BACnet messages addressed to the other



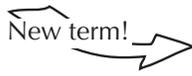
In Illustration 9, Network 20, the original tunnel network from Illustration 8, is still in place. A third tunnel router, Router C, is added to the network to form a second tunnel network. In this example, Router A routes traffic between its BACnet IP network, network 10 (not shown for clarity), and tunnel networks 20 and 21.



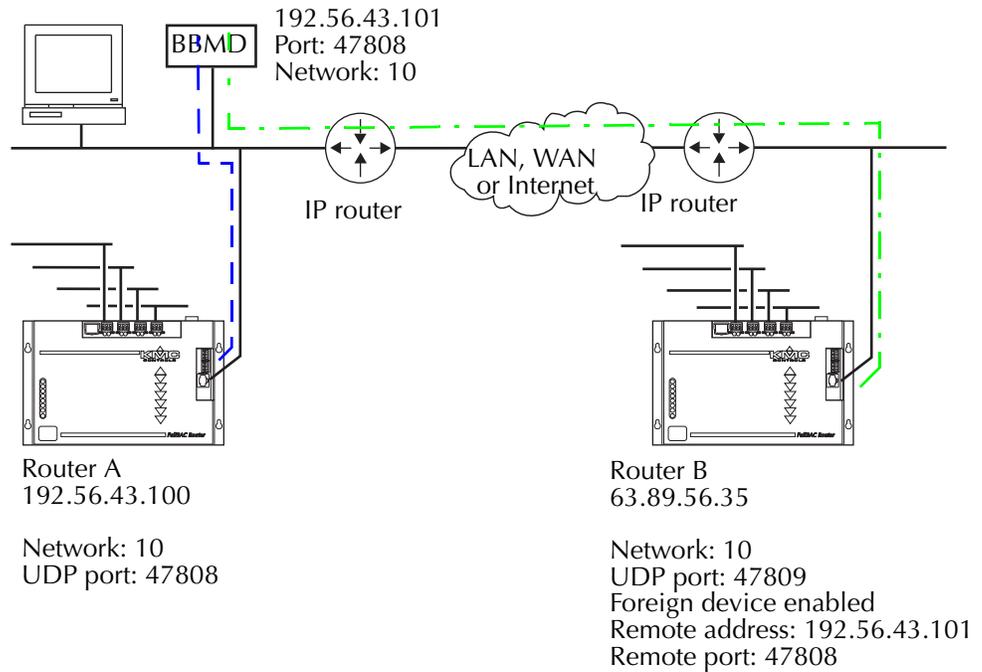
**Illustration 9 Two tunnel networks**

**Crossing IP subnets with a BBMD**

An alternative to tunnel routing for crossing IP subnetworks is to install one or more BACnet Broadcast Management Devices (BBMD) on the internetwork.



**BBMD** A BACnet broadcast-management device (BBMD) is a special type of routing device that is used in BACnet/IP networks to distribute broadcast messages across multiple IP subnets. A BBMD may be a dedicated device or software that runs on computer.



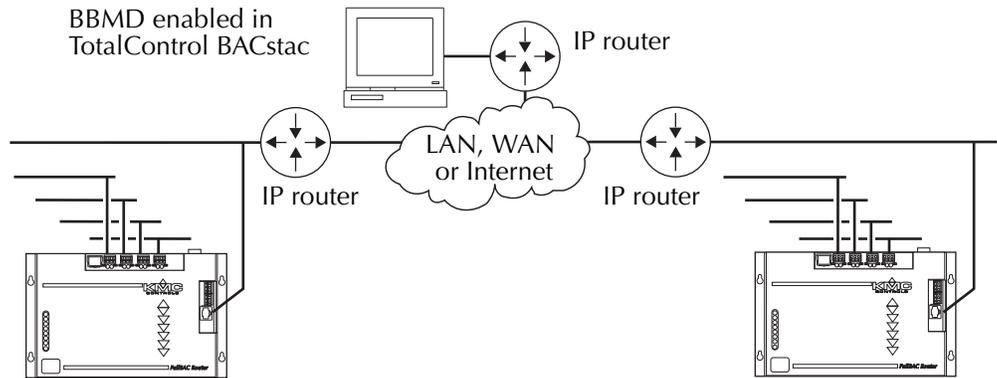
**Illustration 10 BBMD and foreign devices**

In Illustration 10, Router A is configured as a normal BACnet IP router. It is required only for routing traffic to the MS/TP networks or any other BACnet IP network on its IP subnet. The configuration for Router B is similar to PAD routing in the previous section except that the router is configured as a BACnet Foreign Device registered with the BBMD.

The BBMD method of crossing an IP boundary has several advantages over tunnel routing:

- ◆ When the BBMD is configured, you do not need to know any router IP addresses because the routers will register themselves with the BBMD.
- ◆ Expanding the internetwork is simpler than with PAD routing. Adding a new network with an additional router requires only that the router registers with the BBMD which is configured only at the new router. PAD routers require configuration at both routers.
- ◆ Internetwork topology with a BBMD is a simple, single star topology. With PAD routing and only four PAD networks available on a BAC-5050, large internetworks become complicated multiple stars with multiple points of failure.

The TotalControl workstation from KMC Controls includes a BBMD as part of the Cimetrics driver. The BAC-5050 routers in the internetwork are then configured with an IP network configured as a foreign device and the IP address and UDP port number of the BBMD in BACstac driver.



**Illustration 11** TotalControl BBMD with foreign device



#### Detail

Because of network conflicts, BACstage and TotalControl can operate on the same computer under the following configuration.

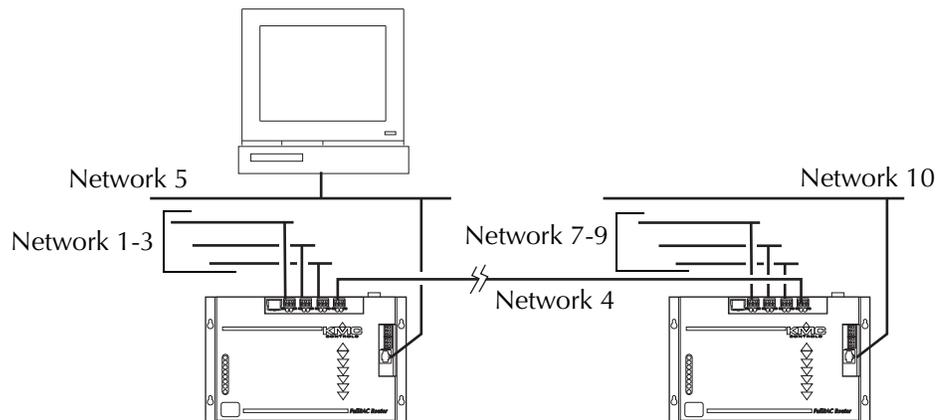
- ◆ Install a second network interface card.
- ◆ Configure a BAC-5050 with two different networks each with a unique UPD port number. Configure BACstage to use one of the ports; the TotalControl driver will then use the other.

More robust internetworks use a BBMD on each IP subnetwork. All BBMDs are aware of other BBMDs but foreign devices need register only with the BBMD on its own subnetwork. This presents a redundant path for broadcast messages to cross IP boundaries. If one BBMD stops operating, the broadcast message will still be routed by other BBMDs.

## Connecting routers with MS/TP

Two routers may be connected by a common MS/TP network. Because connecting the internetwork with MS/TP presents a throughput limitation, use this method only in situations where a limited amount of traffic crosses the MS/TP network.

- ◆ The ports on both routers must be set to use the same baud rate.
- ◆ The MS/TP MAC addresses for the routers on the connecting network must be unique.
- ◆ Controllers may also connect to the MS/TP network. They also must be addressed with unique MAC addresses.
- ◆ Avoid configurations that require high network traffic on the linking MS/TP network.
- ◆ See [MS/TP networking practices](#) on page 2-24 when connecting two routers with an MS/TP network.



**Illustration 12** Two routers connected with MS/TP

## Integrating BACnet with KMC digital networks

Three models of KMC Tier 1 controllers include an interface between a KMC digital network and a BACnet internetwork.

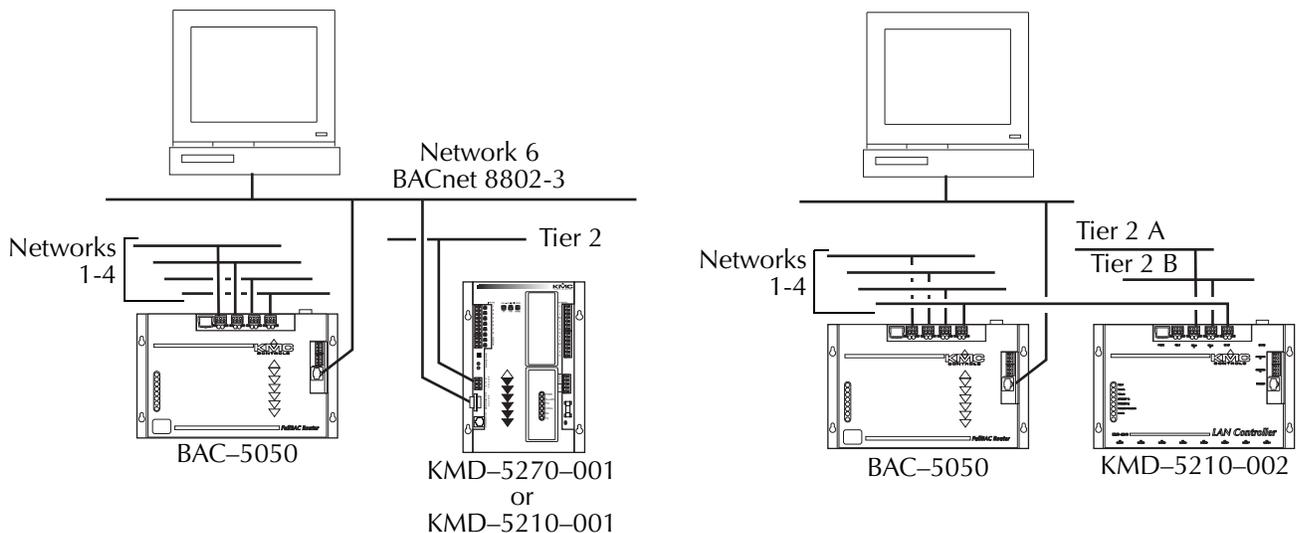
- ◆ The KMD-5270-001 Web Lite connects with BACnet 8802-3.
- ◆ The KMD-5210-001 LAN Controller connects with BACnet 8802-3.
- ◆ The KMD-5210-002 LAN Controller connects through MS/TP.

Programming and configuration details are included in the installation and operation guide supplied with each controller. The following details apply to using a Tier 1 controller as an interface to a BACnet internetwork:

- ◆ Inputs, outputs and variables within the Tier 1 controller appear as BACnet device input, output and value objects.
- ◆ A point configured as a KMD digital point will appear as a BACnet binary object. Analog points appear as analog objects.
- ◆ To be visible as an object to BACnet devices, a descriptor must be entered for the point.
- ◆ Use *BAC-SET*, *BAC-GET* and *BAC-RLQ* in Control Basic to read from and write to objects in other BACnet devices.

Use the following programs to configure and program the Tier 1 controller.

- ◆ Use *Hardware Configuration Manager* to assign a KMD panel number, ethernet information and a BACnet device instance, name and description to the Tier 1 controller.
- ◆ The Tier 1 panel number assigned with HCM becomes the MS/TP MAC address in the KMD-5210-002.
- ◆ Use *WinControlXL Plus* to program the Tier 1 controllers.



**Illustration 13** Connecting to KMD networks

## MS/TP networking practices

The MS/TP (master slave/token passing) protocol is unique to BACnet and is implemented using the EIA-485 signaling standard. Up to 128 BACnet master devices or 256 slave devices can connect to a single MS/TP network.



**Slave** devices cannot initiate requests for data; they can reply only to messages from other devices. They are best suited for simple, low-cost functions.

**Master** devices can initiate but require more processing and memory capacity than slave devices. The KMC advanced application controllers are MS/TP master devices.



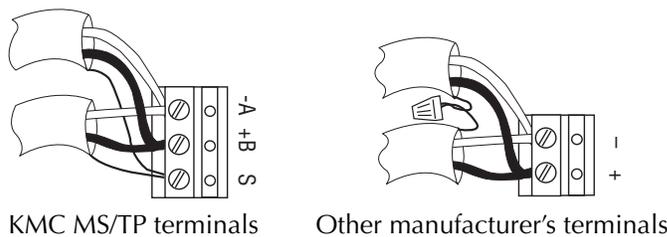
**MS/TP network segment** An electrically separate section of a network. An MS/TP network segment contains no more than 32 devices of any type. Repeaters connect the segments of an MS/TP network.

**Repeater** A network device used to regenerate analog or digital signals distorted by transmission loss. An EIA-485 repeater cannot perform intelligent routing. The network may be divided into two or more segments with repeaters.

### Connecting controllers

Use proper cable and the following principles when connecting a controller to an MS/TP network:

- ◆ You may connect up to 128 BACnet master devices or 256 BACnet slave devices to one MS/TP network.
- ◆ Use 18 gauge, twisted pair, shielded cable with capacitance of no more than 50 picofarads per foot for all network wiring. Belden cable model 82760 meets the cable requirements.
- ◆ Connect the -A terminal in parallel with all other -A or - terminals.
- ◆ Connect the +B terminal in parallel with all other +B or + terminals.
- ◆ For shielding techniques, see [Single point shielding](#) on page 2-25.

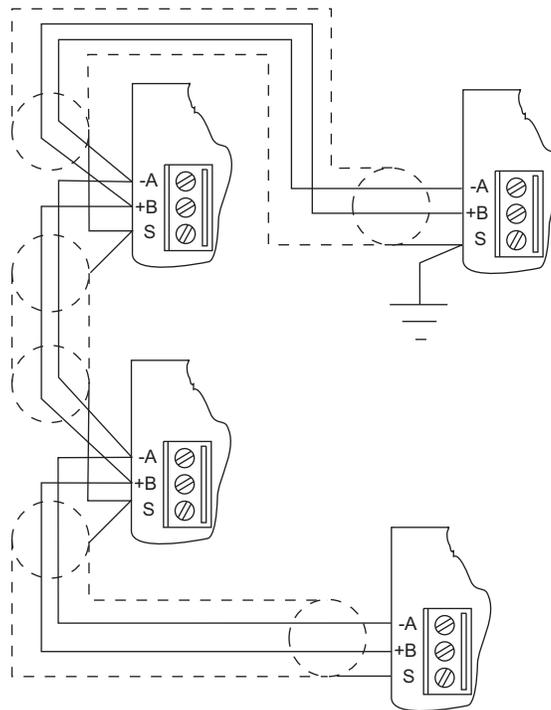


**Illustration 14** Connecting to a controller

**Single point shielding**

Connect the shield at only one end of the cable. The router is usually in a location that provides a good equipment ground and is the recommended ground point. However, either end of the cable can be grounded if a good equipment ground is available.

- ◆ Connect the shields of the cables together at each controller. For KMC BACnet controllers use the S terminal.
- ◆ Connect the shield for each segment to an earth ground at one end only. See [Splitting MS/TP networks with repeaters](#) on page 2-28.

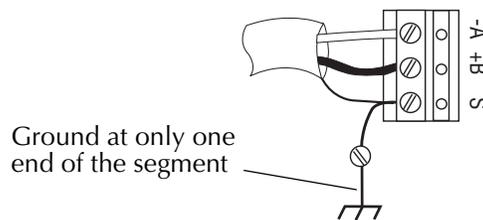


**Illustration 15 Shielding detail**



**Note**

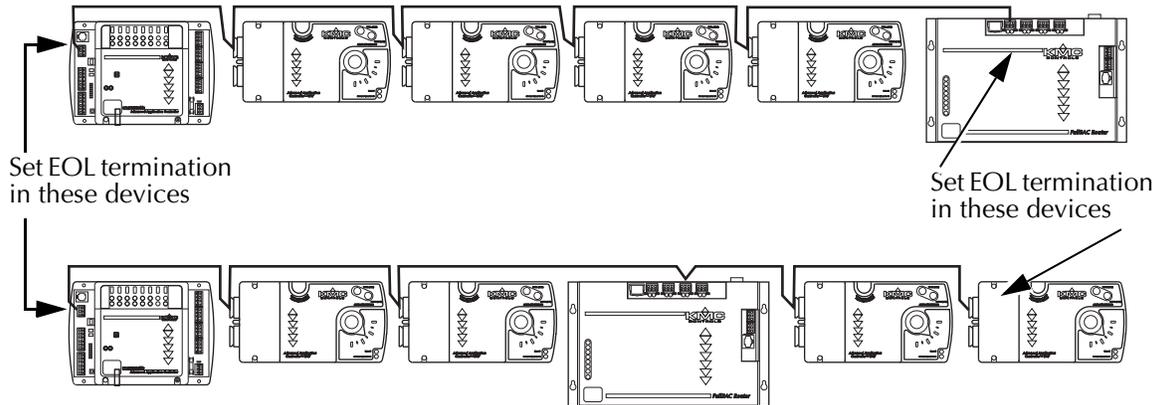
If the ground point is at a KMC controller or router with an S terminal, the S terminal can be used a connection point between the cable shield and equipment ground.



**Illustration 16 Shield grounding with KMC controllers**

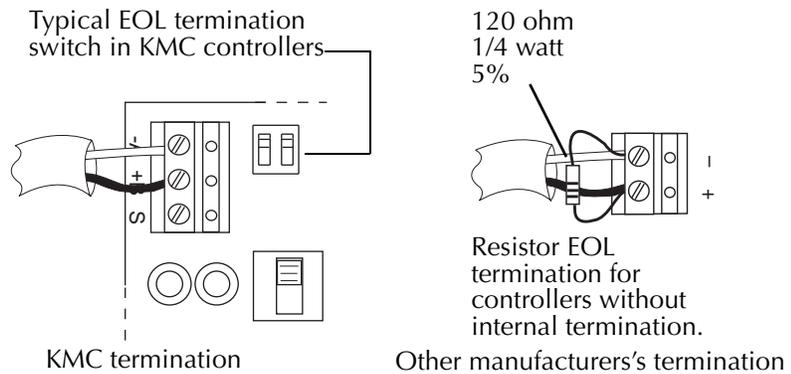
**End-of-line termination**

Each MS/TP network segment requires end-of-line termination for proper operation of the network. Proper termination prevents signal degradation and EMI interference with other system wiring.



**Illustration 17 EOL termination**

To terminate an MS/TP network segment at a KMC BACnet controller, use the end-of-line termination switches (See Illustration 18). Other manufacturer's use different methods for end-of-line terminations which may be as simple as an external resistor shown also in Illustration 18.

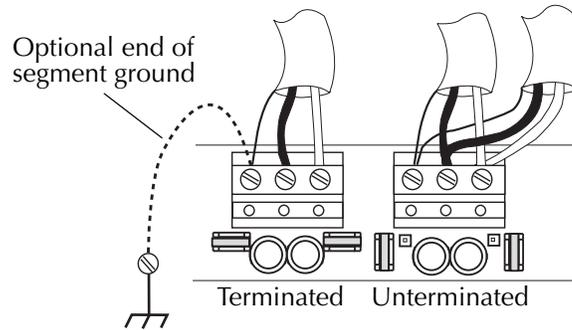


**Illustration 18 End-of-line termination methods**

**BAC-5050 end-of-line termination**

Termination jumpers are located on both sides of each MS/TP connector (one each for the -A and +B terminals; see Illustration 19.)

To activate the end-of-line termination, place the jumpers over the pair of termination pins. If termination is not required, position the jumpers so they cover only one termination pin.

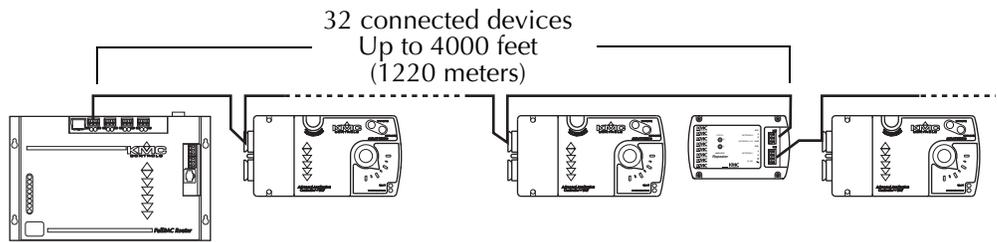


**Illustration 19 BAC-5050 End-of-line termination**

### Splitting MS/TP networks with repeaters

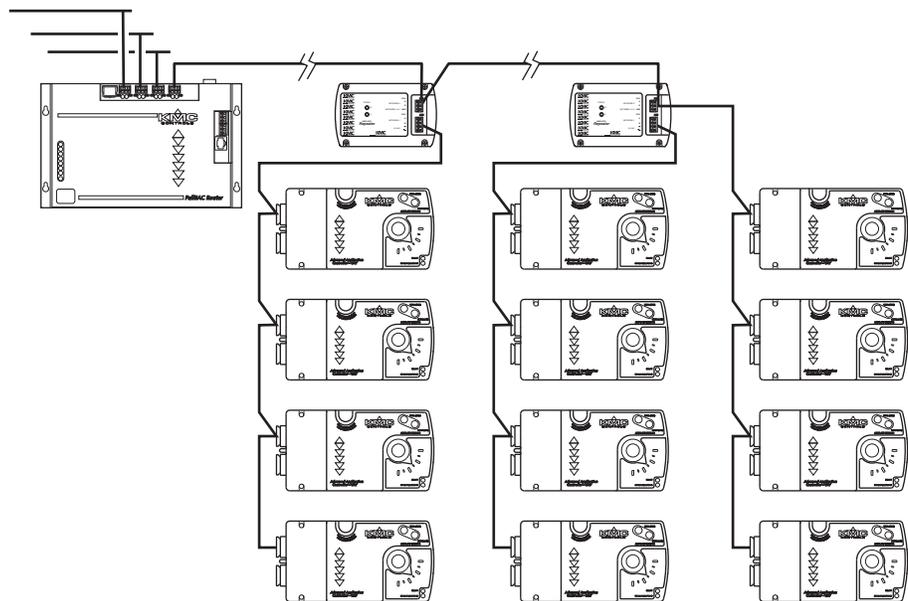
Use repeaters to split the MS/TP network into physical network segments.

- ◆ Connect no more than 32 devices to any one segment.
- ◆ Add repeaters as required for additional controllers up to the limit of the MS/TP network.
- ◆ Total cable length on any segment must be less than 4000 feet (1220 meters)
- ◆ Ground the shield at only one end of each segment.
- ◆ Add end-of-line termination to both ends of each segment.



*Illustration 20 Typical MS/TP network segment*

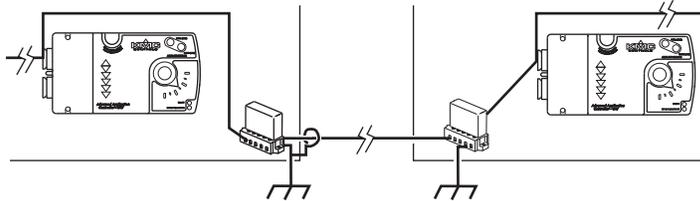
When the physical layout of building requires a star topology, use repeaters to branch from a main MS/TP trunk. The network in Illustration 21 is an example of a star topology.



*Illustration 21 MS/TP star topology*

**Building to building issues**

If the MS/TP network cable runs between two buildings, place a KMD-5567 surge suppressor in the cable at both buildings. Locate the suppressors as close to the cable exits as possible and use a good earth ground. See the installation and operation guide for the KMD-5567 for wiring details.



**Illustration 22 MS/TP between buildings**

Use a KMD-5567 also when an MS/TP network exits a building and terminates at a roof top or air handling unit.



## Application Note-AN0704A

---

Revision A

### Connecting EIA-485 network segments with fiber optic cables

This application note describes methods for using fiber optic cable to replace the shielded twisted pair wiring between Tier 2 network segments.

---

The shielded twisted pair cable used to connect Tier 2 controllers and other EIA-485 (RS-485) network applications is both cost effective and easy to connect. However, shielded cable does have limitations.

- ◆ For KMC digital networks, shielded cable is limited to a total network length no longer than 4000 feet.
- ◆ When routed between buildings, devices on the network are susceptible to damage from lighting.
- ◆ Strong electromagnetic fields produced by machinery may cause disruption to network communications.

By replacing the wiring between network segments with fiber optic cables and modems, these limitations can be overcome.

KMC Controls has tested fiber optic modems from two different manufacturers in several different configurations.

- ◆ The B & B Electronics are simple, cost effective devices. See [B&B Electronics fiber optic modem](#) on page 2-37.
- ◆ The Black Box devices are more expensive but offer added flexibility and include more operational indicators. See [Black Box FOM Line Driver](#) on page 2-33.

Devices from other manufactures may be suitable for this application. However, KMC Controls Technical Support has experience only with the devices described in this application note.

## Networking basics with fiber optic modems

When connecting Tier 2 network segments with fiber optic modems, observe the following principles:

- ◆ Configuring and numbering controllers in systems with Hardware Configuration Manager is identical to systems without fiber optic modems. Number the controllers sequentially and designate the highest numbered controller as the *Last Panel*.
- ◆ Connect no more than 124 Tier 2 controllers to a network.
- ◆ Connect no more than 31 controllers between two fiber optic modems or between a fiber optic modem and a KMD-5575 repeater.
- ◆ The fiber optic modems in the example network diagrams are setup with end of line termination in place.
- ◆ Set the end of line termination in the controllers to *On* only in controllers that are physically at the end of a network segment.
- ◆ Connect the shields together at each controller. Ground one end of the shield for each segment to a good earth ground, typically at the location of the modem.

## Precautions when installing fiber optic cable

A few properties particular to fiber optic cable can cause problems with improper installation.

**Power loss** As the optic signal travels through the fiber core, the signal loses some speed through absorption, reflection, and scattering. This problem is easy to manage by minimizing the number of splices, making splices correctly and keeping connections clean.

**Microbending** Microbends are small deviations in fiber created by excessive bends, pinches, and kinks. Careful installation and selecting cable with reinforcing fibers minimizes this problem.

**Connector loss** Connector loss occurs when two fiber segments are misaligned. This problem is commonly caused by poor splicing. Scratches and dirt introduced during the splicing process can also cause connector loss.

**Coupling loss** Similar to connector loss, coupling loss results in reduced signal power and is from poorly terminated connector couplings.

Remember to be careful and use common sense when installing fiber cable. Use clean components. Keep dirt and dust to a minimum. Don't pull the cable excessively or bend it too sharply around any corners.

## Installing the fiber optic modems

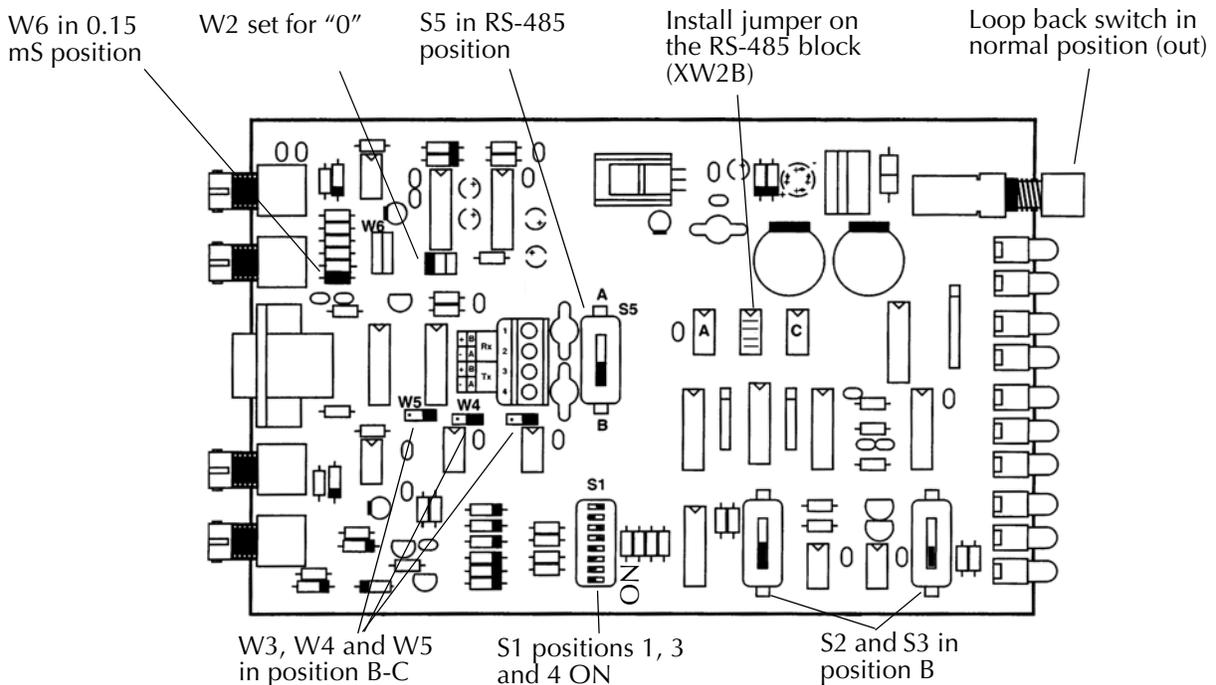
Before installing a fiber optic modem, review and follow the installation instructions supplied with the modem. Both types of fiber optic modems in this application note require 120 volts AC to power a supplied wall mount power supply.

## Black Box FOM Line Driver

Use the Black Box Fiber Optic Modem model ME540A-ST to connect network segments with up to 1 mile (1600 meters) of fiber optic cable. Three network configurations are shown in the following sections, Tier 2 building-to-building, Tier 2 star and LAN controller main network connections.

### Black Box building-to-building settings

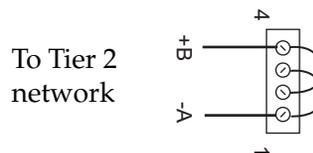
Open the Black Box ME540A by turning the thumb-screw on the bottom of the enclosure and removing the cover. Configure the Black Box line driver for interconnecting network segments from building-to-building, set the switches and jumpers as shown in Illustration 1. Tier 2 wiring connections are shown in Illustration 2.



**Illustration 1** Black Box building-to-building settings

### Connecting to the Tier 2 network

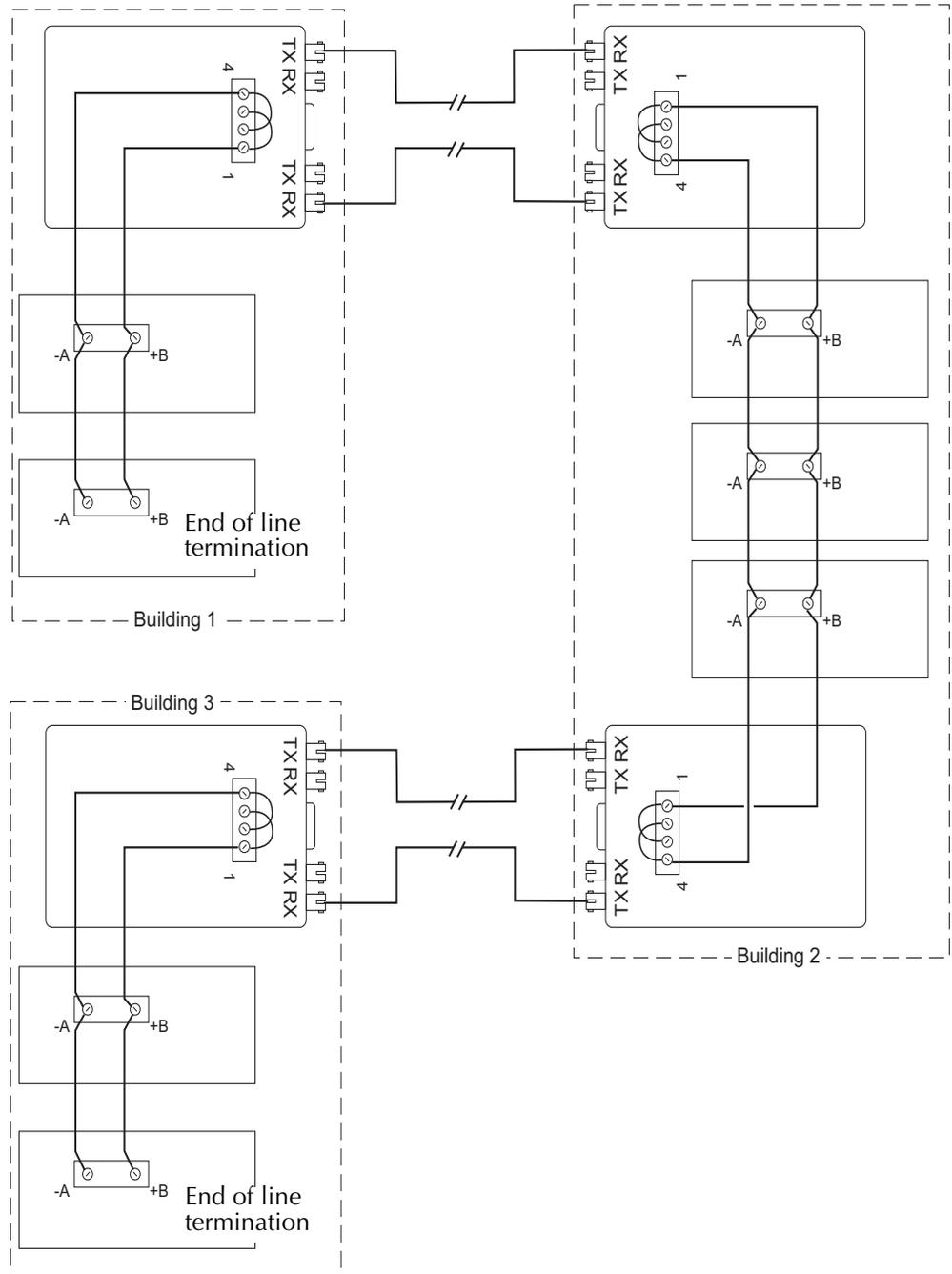
Connect the Tier 2 network to the terminal strip inside of the FOM as shown in Illustration 2. Install a jumper from terminals 1 and 3 and terminals 2 and 4. Connect the Tier 2 -A wire to terminal 4 and the +B wire to terminal 1.



**Illustration 2** Tier 2 connections at terminal strip

**Connecting from building to building**

Illustration 3 shows FOM line drivers connecting together Tier 2 network segments in three buildings. The ME504A-ST was configured to include end of line termination for each segment. The end of line termination jumpers (or termination switches are set to *Off*) on all controllers except those at the ends of the network.



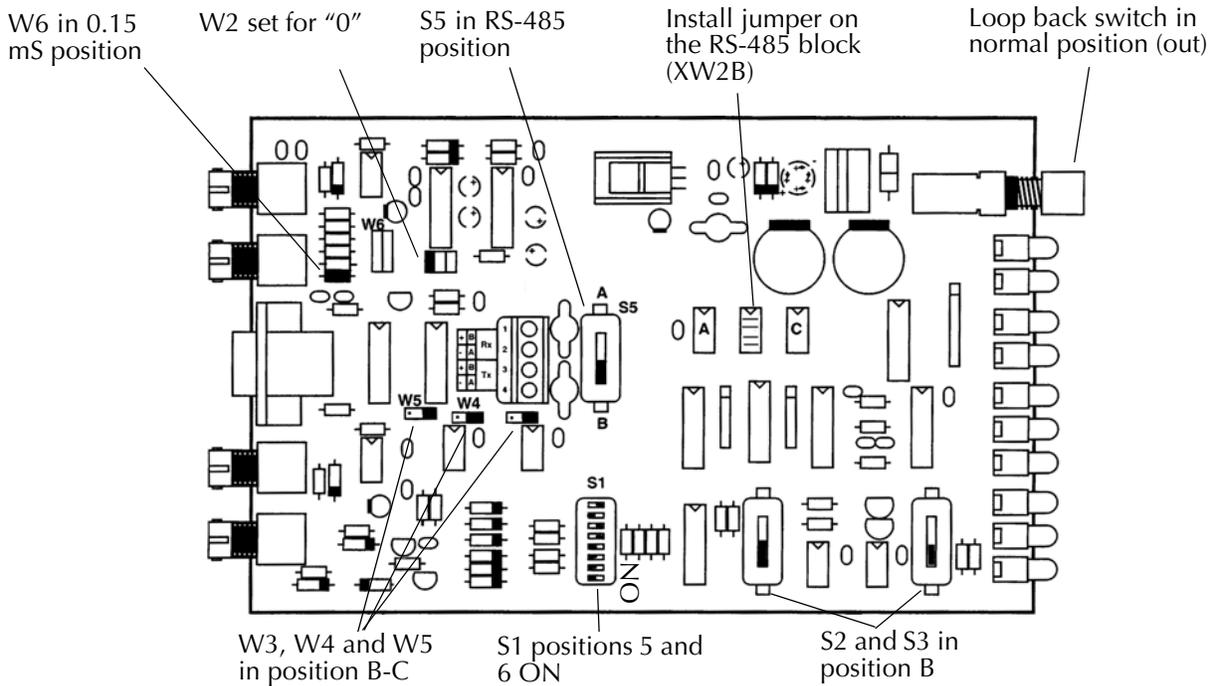
**Illustration 3 Tier 2 Black Box building-to-building network**

**Tier 2 star configuration**

Use the star configuration with the Black Box ME540A-ST to branch multiple network segments off of a common trunk.

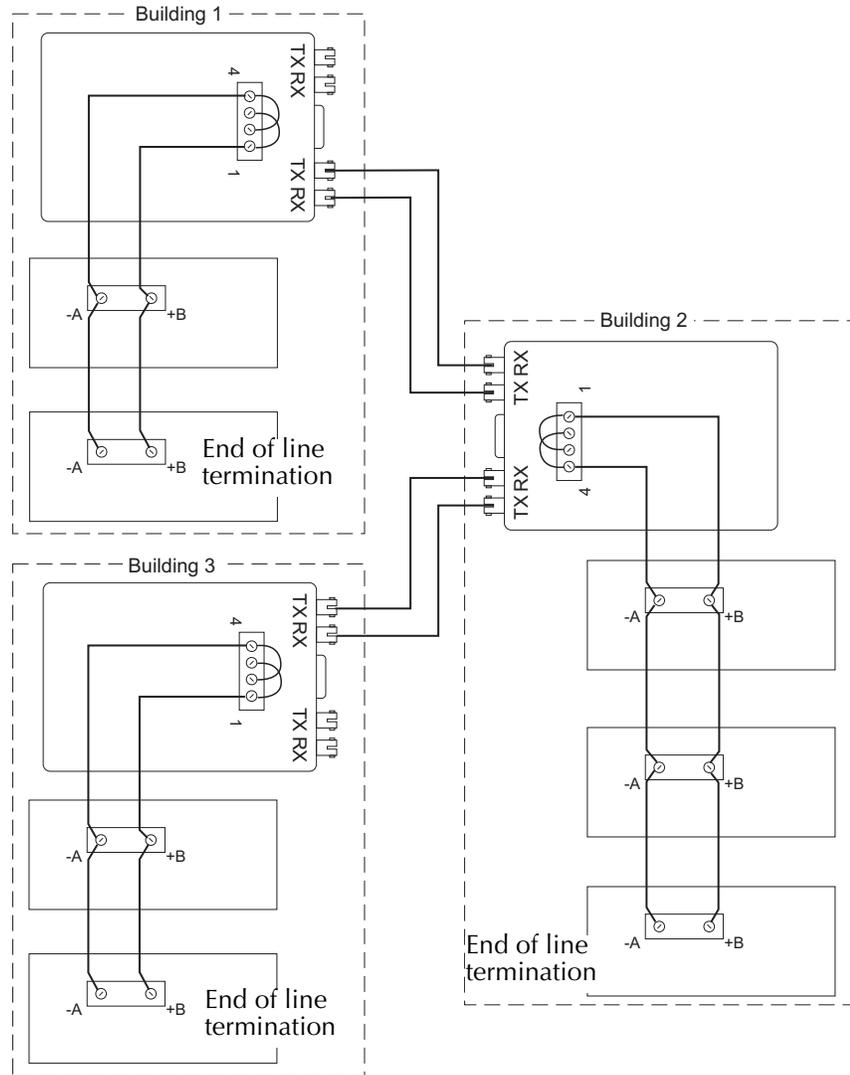
To configure the Black Box ME540A, open the modem by turning the thumb-screw on the bottom of the enclosure and removing the cover. To configure the Black Box line driver for interconnecting network segments for a star configuration, set the switches and jumpers as shown in Illustration 4. The difference between Illustration 1 and Illustration 4 is that the FOM is configured for network mode.

Connect the ME540A to the Tier 2 cable as shown in Illustration 2.



**Illustration 4 Black Box star configuration**

The following configuration can be used to join two or more SubNetwork controllers onto a fiber optic trunk. This example includes three end of line controllers.



**Illustration 5 Black Box star network**

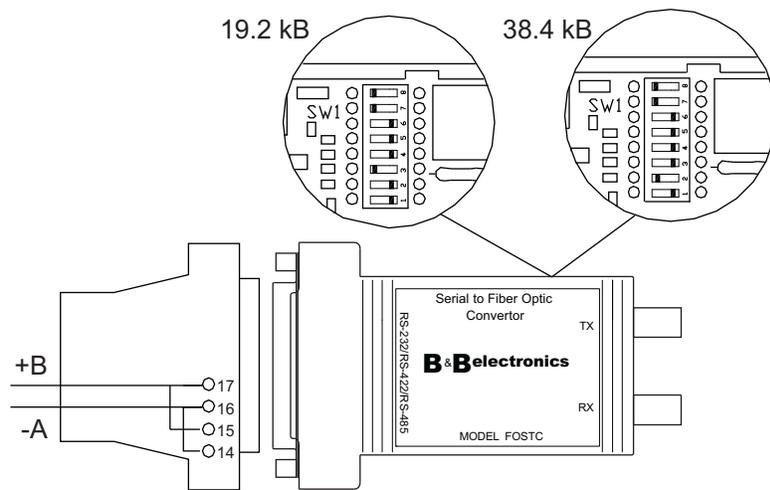
**B&B Electronics  
 fiber optic modem**

The B & B Electronics model FOSTC modem can be used to connect a KMC digital network using multimode fiber optic cable up to 2.5 miles. The modem uses ST type connectors for the fiber connections.

**Configuring the B & B Model FOSTC**

To configure B & B Model FOSTC, remove the cover. Once the internal board is exposed, set the switches on SW1 as shown in Illustration 6.

- ◆ Switches 7 & 8 *On* - this sets the FOSTC for to RS-485 (EIA-485) mode.
- ◆ Switch 6 *Off* - this sets the unit to terminating (point to point)
- ◆ Switch 3 *On* - this sets timing for 19200 baud OR Switch 2 *On* - this sets timing for 38400 baud
- ◆ Remainder of switches *Off*

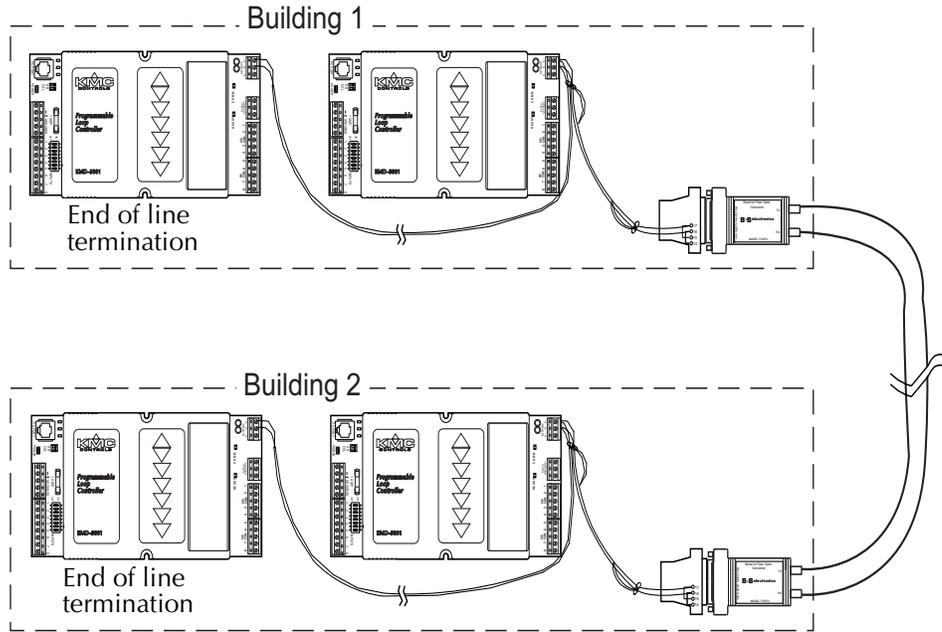


**Illustration 6 B&B Electronics confide**

Use a male DB-25 to make connection to the KMC digital network. Illustration 6 shows how to connect the B&B fiber optic modem to a Tier 2 network.

**B&B two buildings**

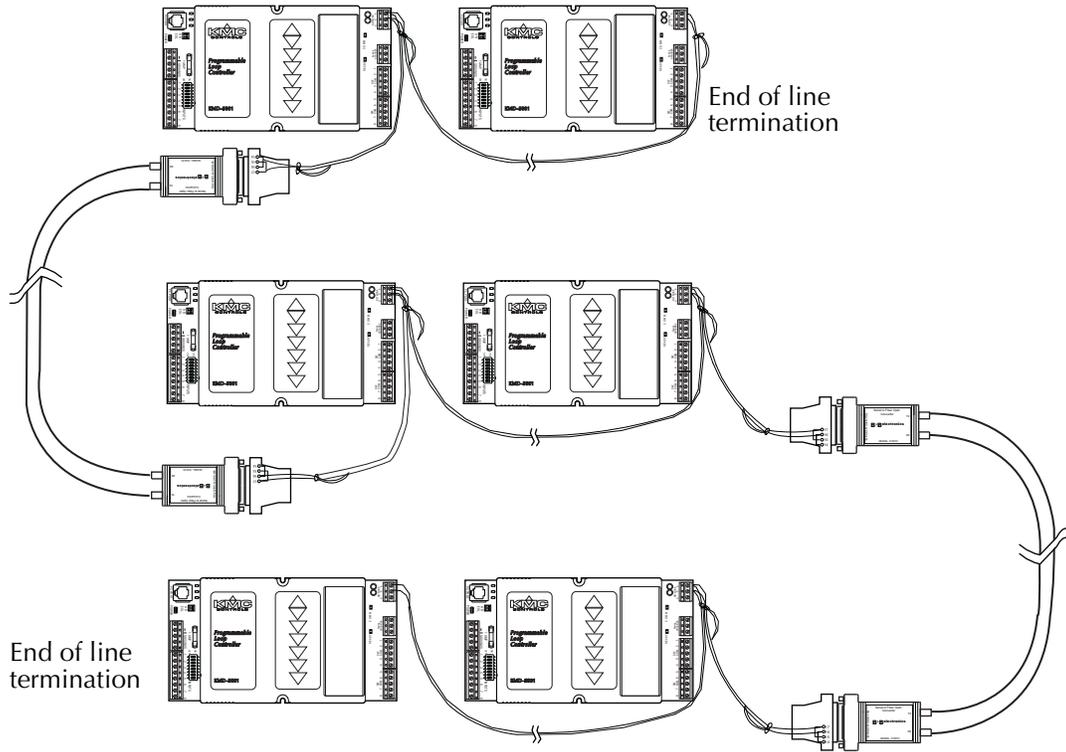
Illustration 7 shows a network that connects network segments in two different buildings. Building 1 and building 2 each have one Tier 2 controller that has the end of line jumper *On*.



**Illustration 7 B&B two-buildings connected**

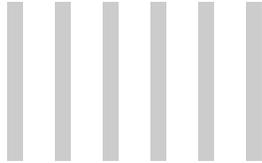
**Three buildings**

Illustration 8 shows the network layout for three buildings. Building 2 does not have any controllers with the end of line jumpers *On*. Both fiber modems are considered end of line.



**Illustration 8 B&B three-buildings connected**





# Part 3

## Job specific controllers

This part describes the programming supplied in the job specific DDC controllers manufactured by KMC Controls. Each application note includes a description of the program, configuring and programming with HCM and WinControl, a program list and schematics

---

|  |       |
|--|-------|
| AN0404B KMD-7001 and KMD-7051 VAV Controller.....              | 3-3   |
| AN0404C KMD-7002 and KMD-7052 VAV Controller .....             | 3-29  |
| AN0905A KMD-7003 and KMD-7053 VAV Controller.....              | 3-43  |
| AN0404D KMD-7011 VAV Controller .....                          | 3-67  |
| AN0404E KMD-7013 VAV Controller .....                          | 3-89  |
| AN0404F KMD-7101/02 Fan Coil Unit Controller.....              | 3-109 |
| AN0404G KMD-7301 AHU Controller .....                          | 3-127 |
| AN0404H KMD-7302 Roof Top Unit Controller .....                | 3-143 |
| AN0404I KMD-7401 Heat Pump Controller.....                     | 3-157 |
| AN0905B Using the NetSensor on preprogrammed controllers ..... | 3-171 |
| AN0805A The NetSensor VAV balancer program.....                | 3-175 |



## Application Note–AN0404B

---

Revision A

### **KMD-7001 and KMD-7051 VAV Controller**

This application note describes using a KMD–7001 and KMD–7051 for single-duct VAV installations. It covers using the controllers for heat/cool change over and proportional reheat.

---

#### **Related materials**

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ Installation and operation guides for KMD–7001, STE–6019, and REE–2102
- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide
- ◆ System plans with controller addresses
- ◆ AN0805A, The NetSensor Balancer Program

#### **Introduction**

The KMD-7001 is a KMC digital controller and an actuator built into one unit. A pressure-independent factory programmed VAV sequence automatically modulates flow into the conditioned space between Minimum and Maximum setpoints as defined by the user. This sequence allows for Heating, Cooling, or changeover applications with minimal user initialization. Additionally, user programming can customize each application as required.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory configurations allowing you to program any desired sequence.

The KMD-7001 VAV Terminal Controller contains the basic programming sequence required for three specific applications:

- ◆ Single Duct VAV (heating-cooling-changeover)
- ◆ Single Duct VAV with time proportional reheat
- ◆ Single Duct VAV with three-stage reheat (including fan-box applications).

In addition, there is a balance program for setting up VAV box parameters. Initially these devices will default to a standard Single Duct Cooling VAV configuration. You can use either of the Reheat functions by turning "ON" the respective programs and enabling a variable on the Variables screen. The following paragraphs provide specific details and explanations to help you use the KMD-7001.

## Velocity sensor

The KMD-7001 incorporates an on-board flow-through type sensor to determine duct airflow. Airflow through the sensor is approximately 1.12 SCFH (32 liters/hour) at .25 INWC (62.5 Pa) pressure drop. The sensor uses two RTD's (resistance temperature detectors) to measure flow. The RTD's are platinum films on a ceramic base chosen for long term stability and accuracy. The smaller 100 ohm RTD is self-heated to a constant temperature differential above ambient. The amount of power required varies with the specific airflow. A larger 1000-ohm RTD constantly senses ambient temperature and compensates. The sensor housing channels the airflow over the self-heated RTD in a manner to isolate its heat from the compensator. This minimizes shifts at low flows due to various mounting positions. The sensor housing itself is molded from plastic that carries the UL94-5V rating. The sensor ports are .188 inch diameter and readily accepts .25 inch O.D. "FR" tubing (.040 wall) or equivalent.

## Velocity limits

Minimum and Maximum Flow velocity limits can be defined in either of two ways: Through KMC digital operating software or with the NetSensor. Setting velocity limits through the KMC digital software is as simple as defining the value of a variable. Using this method the limits may be entered in any units desired (CFM, L/s, Pa, etc.) as long as a "conversion factor" is also provided. The "default" settings are always in Feet Per Minute (FPM). For details on using this method refer to the following section on standard VAV operation.

## Damper actuator

The actuator is capable of supplying a minimum of 50 inch pounds of torque to open or close the damper according to instructions from the KMD control sequence. The motor is an AC synchronous type for smooth, reliable operation and is coupled with a magnetic slip-clutch design that allows continuously stalling damper rotation without damage. A time-out function will stop the motor anytime the unit has run continuously in one direction for longer than 6 minutes. This ensures that the stops have been reached without needlessly running the motor.

Physical adjustable end-stops allow limiting rotation to less than the 90° rotation and a gear disengagement mechanism on the side of the case can be used for fast and simple installation and set-up. The red disengagement button is located on the left side of the actuator.

## Standard Variable Air Volume (VAV) Operation

The standard VAV sequence of operation has been pre-programmed into each KMD-7001 controller. Default operation is defined as:

- ◆ Single Duct Cooling
- ◆ Clockwise to close damper rotation
- ◆ Minimum and Maximum Limits at 500 and 2000 FPM respectively
- ◆ Cooling setpoint at 74°F
- ◆ Space Temperature Sensor connected to IN1

These units are capable of independent, stand-alone operation or they may be networked together with other KMC digital products. If stand-alone operation is desired, supplying 24 VAC power and connecting a 10K thermistor (STE-series) to

IN1 will enable full operation to the default parameters listed above. If these units will be networked together it may be necessary to configure each controller with a unique address and other set-up options using HCM.

The following is a simplified explanation of the control logic used in the standard VAV application:

- ◆ A full-function PID controller (CON1) is set to perform the thermostat function by comparing the space temperature (IN1) to the set point temperature (VAR25). In the COOL (default) changeover mode, thermostat action is Direct Acting and "Active" Setpoint is the Cooling setpoint temperature VAR30). Conversely, if HEAT changeover mode was selected, thermostat action would change to Reverse Acting and the "Active" set point would be the Heating Set Point (VAR29).
- ◆ A "requested" flow set point is calculated based on the thermostat information and Minimum / Maximum flow limits. Using this "Requested" flow a second full function PID Controller (CON2) looks at actual flow (IN4) as measured by the velocity sensor and instructs the damper to open or close accordingly.
- ◆ Parameters of either the Thermostat (CON1) or the Flow Controller (CON2) such as Proportional Band, Reset (Integral), or Rate (Derivative) functions can easily be changed to suit application requirements. The "BIAS" of 50% preset on CON2 should not be changed as this defines a satisfied "Neutral" position for the actuator drive circuit.

## Hardware Configuration Manager (HCM)

For networked operation, use the HCM software to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 1 Default Configuration for KMD-7001**

|                   |       |
|-------------------|-------|
| Panel Number      | 124   |
| Network Baud Rate | 38400 |
| Last Panel        | No    |
| Panel Type        | 116   |

## Setting up with WinControl XL

This section covers the set-up requirements to activate any of the pre-programmed options available in this device. At this point, communications should have been established with the unit.

A "Master" password is available to "sign-on" to each controller. Other security codes can be entered to control access to the system.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence.

The first step after signing on to the controller is “Load Descriptors One” or “Load Descriptors All”. Many of the descriptions have been pre-defined for Inputs, Outputs, and Variables to aid in configuring the options.

To use either the Time-Proportional Reheat (OUT1) program or the 3-stage Reheat (OUTS 1, 2, 3) program, go to the “Control Basic” programs screen and turn “ON” the appropriate program.

After the appropriate programs are enabled you can use the setup screen in Control Group #2 (GRP2) which lists the items most likely to be monitored. The following section outlines each of the parameters that have been pre-programmed into the device.

### Configuring inputs

Inputs 1, 2, and 3 can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:

| # | Description      | Manual                              | Value  | Units            | Decom.                              | Label    |
|---|------------------|-------------------------------------|--------|------------------|-------------------------------------|----------|
| 1 | ROOM TEMPERATURE | <input type="checkbox"/>            | 72.00  | Deg. F           | <input checked="" type="checkbox"/> | ROOMTEMP |
| 2 |                  | <input type="checkbox"/>            | 0.00   | Unused           | <input type="checkbox"/>            |          |
| 3 |                  | <input checked="" type="checkbox"/> | 0.00   | Unused           | <input type="checkbox"/>            |          |
| 4 | AIR FLOW SENSOR  | <input type="checkbox"/>            | 250.00 | Unused (Table 2) | <input checked="" type="checkbox"/> | FLOWSNSR |

**Illustration 1 Input configuration in WinControl XL**

**Input # 1–ROOM TEMPERATURE** Default selects a KMC 10K ohm thermistor as used in all of KMC’s STE series sensors.

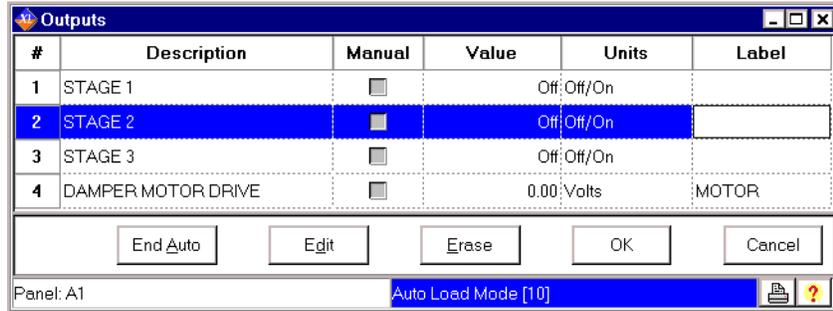
**Input #2** Unused

**Input #3** Unused

**Input #4–Air Flow Sensor** The on-board flow-sensor. Table 2 has been configured to convert the voltage input into feet per minute. This input is not accessible for any other use.

**Configuring outputs**

The KMD-7001 VAV Controller features three universal outputs (analog or digital, set in software) and an analog output to drive the damper motor. The KMC supplied control routines have been written to use the following output configuration:



**Illustration 2 Output configuration in WinControl XL**

**Output #1–Stage 1** This output is configured to activate the first stage of heating on VAV boxes with electric reheat (PRG3). Optionally, it can be used to control any 0-10 VDC device by adding your own custom programming in program area 5 (PRG5).

**Output #2–Stage 2** This output is configured to activate the second stage of heating on VAV boxes with electric reheat (PRG3). Optionally, it can be used to control any 0-10 VDC device by adding your own custom programming in program area 5 (PRG5).

**Output #3–Stage 3** This output is configured to activate the third stage of heating on VAV boxes with electric reheat (PRG3). Optionally, it can be used to control any 0-10 VDC device by adding your own custom programming in program area 5 (PRG5).

**Output #4–DAMPER MOTOR DRIVE** This output is used to control the damper position. It is not accessible for any other use on the KMD-7001. When the output is commanded to 0 volts the motor drives the damper counterclockwise, 10 volts drives the damper clockwise, and 5 volts stops the damper.

**Configuring variables**

A total of thirty-two system variables are available to perform various functions such as set points, limits, and modes. All variables can be changed or modified to suit the specific application. The following have been pre-defined for use in the KMD-7001 factory supplied control sequences:

**Variable #17–CHANGEOVER MODE (COOL/HEAT)** Chooses whether the unit will operate as a heating or cooling box by automatically setting the thermostat action as direct (COOL) or reverse (HEAT).

**Variable #18–CLOCKWISE TO CLOSE? (NO/YES)** Defines which way the damper will rotate to increase or decrease air-flow as required. “NO” will close the box in a counter-clockwise direction.

**Variable #19–MINIMUM FLOW** Sets the lowest design flow for the VAV box. The default setting is for a Feet-Per-Minute (FPM) value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable #20–MAXIMUM FLOW** Sets the highest design flow for the VAV box. The default setting is for a FPM value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable #21–VOLUME CONVERSION** Default value is 1.00 (no conversion). This factor allows all flow settings and live readings to be in any desired units. Since default calculations are in FPM, the duct area is required (in square feet) to get readings in CFM. Similarly, values can be converted to L/s, Pa, etc. with the proper factors applied.

**Variable #22–SENSOR CORRECTION** Default value is 1.00 (no correction required). This factor is provided to allow simple correction for erroneous sensor readings. This factor is multiplied against all flow readings and the product is stored as the ACTUAL FLOW (VAR23).

**Variable #23–ACTUAL FLOW** This value is calculated from the flow sensor input (IN4) and the sensor correction variable (VAR22) using the following formula:

$$\text{ACT-FLOW} = \text{FLOWSNR} * \text{VAR22}$$

Or:

$$\text{VAR23} = \text{IN4} * \text{VAR22}$$

Units for this Variable must always be in Feet per Minute for proper operation of the built-in control sequences. For flow displayed in desired units see Variable 26.

**Variable #24–REQUESTED FLOW** This is the flow set point based on the heating or cooling needs of the space. The value is set automatically in Program 1 (PRG1). Units for this Variable must always be in Feet per Minute for proper operation of the built-in control sequences. For flow displayed in desired units see Variable 26.

**Variable #25–ACTIVE TEMP SETPOINT** This is the temperature the controller will attempt to maintain in the space. The active set point is the heating set point (VAR29) in HEAT MODE (VAR17 = ON) and the cooling set point (VAR30) in COOL MODE (VAR17 = OFF)

**Variable #26–DUCT VOLUME READOUT** This variable has been set up to allow easy access to a live velocity read-out. Units can be converted using the Volume Conversion factor (VAR21) to relate to CFM, L/s, Pa, etc. as needed. The following formula is used in Program 1 (PRG1) to obtain this value:

$$\text{VOLUME1} = \text{ACT-FLOW} * \text{VOLFACTR}$$

Or:

$$\text{VAR26} = \text{VAR23} * \text{VAR21}$$

**Variable #27–REHEAT STATUS** This flag will enable or disable the REHEAT function. If reheat is disabled, all outputs being used in that reheat function are overridden to “OFF”. To choose one of the pre-programmed Reheat options, the appropriate program must be turned on. See the Program Menu for options. If neither reheat program is turned on, this variable has no purpose.

**Note**

A common application is to use this flag to disable the Reheat function during the summer mode.

**Variable #28–AUXILIARY FLOW** Assigning a flow set point other than 0 in this variable allows the VAV set point to be overridden when STAGE 1 (OUT1) heat is turned “ON”. The auxiliary flow set point is only enabled if STAGE HEATING (PRG3) is used, the REHEAT STATUS (VAR27) is enabled, and the value is non-zero. To disable the auxiliary flow, leave the value at 0 (zero).

**Variable #29–HEAT TEMP SETPOINT** This is the active set point used by CON1 when in the heating mode (VAR17 = ON). Units must match that of the temperature sensor on IN1 (deg. F or deg. C)

**Variable #30–COOL TEMP SETPOINT** This is the active set point used by CON1 when in the cooling mode (VAR17 = OFF). Units must match that of the temperature sensor on IN1 (deg. F or deg. C)

**Variable #31–DAMPER DRIVE TIME** This is the approximate time in seconds required for the damper to move from the fully closed to the fully open position or from the fully open to the fully closed position. It is used in program 1 to roughly determine the damper position in variable 32 (VAR32).

Approximate travel time is 300 seconds for 90 degrees, 200 seconds for 60 degrees and 150 seconds for 45 degrees. The end stops on the back side of the KMD-7001 need to be placed in their appropriate places to get the desired rotation.

**Variable #32–DAMPER POSITION** This is a calculated position based on the full stroke drive time (VAR31). Variable 31 must be set correctly for the damper position information to work properly.

Damper position is determined in Program 1 by measuring the time the damper has been commanded open or closed versus the time needed for complete travel. If the damper is stuck this variable will not necessarily indicate the correct damper position. It will however indicate the controllers attempt to move the damper. This variable is used only to give users an indication of what the controller has done and is doing to control the position of the damper. It is not used as feedback in the control sequence.

**Configuring PID control loops**

KMC digital products feature built-in, full function PID loop controllers. All parameters may be modified as necessary.

**Controller #1 (CON1)** is a direct-acting thermostat. It compares ROOM TEMPERATURE (IN1) to the ACTIVE TEMP SETPT (VAR25) to determine the VAV flow requirement.

**Controller #2 (CON2)** is set up as a direct-acting thermostat. It compares ACTUAL FLOW (VAR23) to the REQUESTED FLOW (VAR24) to operate the DAMPER (OUT4). All values used by the controller have been converted to FPM to allow a common proportional band value regardless of the units selected. Default is a 200 FPM proportional band which relates to approximately 60 FPM “dead band” to stabilize actuator operation. Default parameters do not include any integral or derivative function.



---

**Note**

The desired air flow is VAR24 and is automatically calculated by the VAV program. For example, to override the airflow to some other value under specific conditions, include a statement such as:

```
10 IF (condition) THEN VAR24 = (value)
```

All values at this point must be in FPM. Condition may be a variable that represents an abnormal event such as a smoke zone activation. To ensure close-off on a box, set the desired flow (value) to a number below zero such as -50. To ensure a wide open damper position use a flow rate higher than 3000 FPM (The highest value the on-board sensor can read).

**Controller #3 (CON3)** compares the ROOM TEMPERATURE (IN1) to the HEATING SET POINT (VAR29) for operation of the reheat stages when programs 2 or 3 are enabled.

**Control Basic programming**

The first program handles set points and flow control. There are two programs provided to control reheat in the VAV box. If neither sequence fits the required application, edit the programs or send blank programs to the controller and replace any of the factory programs. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.



---

**Caution**

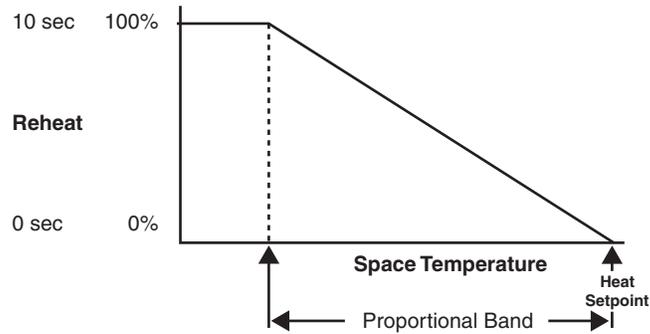
This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type).

**Program 1–Single Duct VAV** This program sets the ACTIVE SP (VAR25) based on the CHANGE OVER MODE (VAR17). When the CHANGE OVER MODE is OFF the ACTIVE SET POINT equals the Heating Set Point (VAR29). When the CHANGE OVER MODE is ON the ACTIVE SET POINT equals the Cooling Set Point (VAR30). The necessary Air flow is calculated and the damper motor is modulated to maintain this volume.

See [Program 1– SINGLE DUCT VAV](#) on page 3-20 for a complete listing of the program.

**Program Two–TIME PROPORTIONAL REHEAT** This uses only one digital output (OUT1) by controlling the percentage of “ON” time over some specific time interval. For the pre-programmed sequence a fixed 10-second time base is used; A call for 73% heating would turn “ON” the output for 7.3 seconds and turn “OFF”

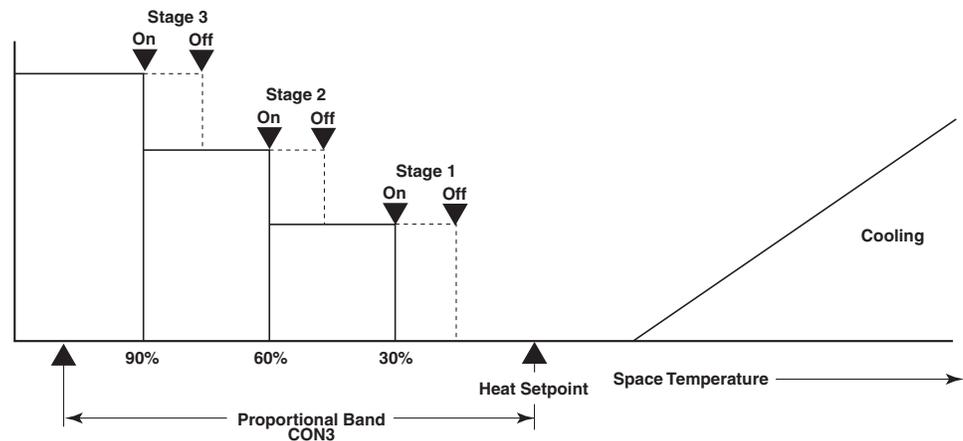
for the remaining 2.7 seconds. See Illustration 4. Using this method of reheat requires only one output. Outputs 2 and 3 are left for other output devices. Illustration 3 shows a graphical representation of Program 1.



**Illustration 3 Proportional Reheat**

See [Program 2–TIME PROPORTIONAL HT](#) on page 3-21 for a complete listing.

**Program 3–Three-stage Reheat** This program sequentially steps each stage of reheat on as the requirement for heating increases (using Outputs 1, 2, and 3). Each stage turns “ON” at a fixed point (30%, 60%, and 90%) of the heating thermostat (CON3) value. The controller changes 0-100% over the proportional band range. Illustration 4 shows the sequence graphically.



**Illustration 4 Three-stage reheat**

This program may also be used for FAN-BOX applications. Connect the FAN relay to STAGE 1 (OUT1) and 2 stages of reheat to outputs 2 and 3. A differential is built into the sequence to prevent cycling the outputs over small temperature variations.

An optional AUXILIARY FLOW limit has been pre-programmed for use with the 3-stage Reheat sequence. This feature allows increasing the VAV flow set point to some value other than Minimum Flow anytime Stage 1 of heating is enabled. The value of Auxiliary flow entered in VAR28 must be in the same units as Minimum

and Maximum flow since the same “conversion factor” (VAR21) is used by the programming. If this feature is NOT being used, leave the Auxiliary Flow (VAR28) value at zero (0) and the program will ignore it.

The programming has been written to instruct the unit to control flow at the Auxiliary Flow limit whenever conditions call for stage 1 heating. If it is desired to initiate the Auxiliary Flow before actually turning “ON” the first stage of heat, add “Delay” time in the appropriate column of OUT1 on the Outputs screen (up to 99 min. 50 sec.). The delay will take precedence over all other software actions.

See [Program 3 3–STAGE REHEAT](#) on page 3-21 for a complete listing

**Program 4–Balancer override** This program is used with the NetSensor to force the controller to maintain either a minimum or a maximum air flow setpoint. Details on using the NetSensor can be found in applications note *AN0805A, The NetSensor Balancer Program*.

### **Program 5–Available for user programming**

#### **Configuring tables**

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a

potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7001:

**Table #1** Ready for user programming.

**Table #2** Correlates velocity pressure from the on-board flow sensor to feet per minute.

**Table #3** Correlates velocity pressure of CSP-5001 VAV controller to feet per minute.



---

#### **Note**

These tables use nominal values for correlation. Due to component tolerances and the resistance of field wiring, these values may need correction on an individual basis. If the value read is too low, enter a positive calibration in the input column labeled “cal”. If the value read is high, a negative calibration can be entered. If necessary, reconfigure Table #2 and/or Table #3. The complete Table #2 and Table #3 from the factory are shown in figure 3. If the tables have been reconfigured and the user wants the factory configuration for Table #2 and Table #3 it will be necessary to use Clear Panel or re-enter the values.

**!** **Caution**

Using Clear Panel will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type.

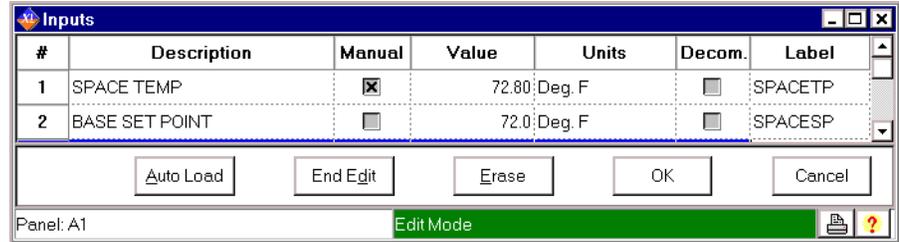
| #  | Table 1 | Unused | Table 2 | Unused | Table 3 |
|----|---------|--------|---------|--------|---------|
| 1  | 0       | 0      | 0       | 0      | 0       |
| 2  | 0       | 0      | 0.29    | 0      | 0.35    |
| 3  | 0       | 0      | 0.4     | 92     | 0.46    |
| 4  | 0       | 0      | 0.8     | 401    | 0.86    |
| 5  | 0       | 0      | 1       | 530    | 1.1     |
| 6  | 0       | 0      | 1.5     | 803    | 1.7     |
| 7  | 0       | 0      | 2       | 1032   | 2.3     |
| 8  | 0       | 0      | 2.5     | 1253   | 2.5     |
| 9  | 0       | 0      | 3       | 1505   | 2.9     |
| 10 | 0       | 0      | 3.5     | 1824   | 3.25    |

**Illustration 5 Table configuration in WinControl XL**

The primary control logic in this model has a heating and cooling setpoint defined in the software as variables. For applications requiring room set point control at the thermostat, follow these steps:

1. Define an Input as the base set point—such as IN2 using the STE-5000 Series Room thermostat. Enter Table 1 in the Range column of the Input window.

- Set up the table as shown in Table 2 or Table 3. The WinControl Table 1 will convert the voltage from the dial into degrees Fahrenheit.



**Illustration 6** Alternate input configuration in WinControl XL

**Table 2** STE-5200 Conversion Table

|   | Value |      |
|---|-------|------|
| 1 | 0.00  | 86.0 |
| 2 | 0.710 | 85.0 |
| 3 | 1.07  | 80.0 |
| 4 | 1.38  | 75.0 |
| 5 | 1.67  | 70.0 |
| 6 | 1.91  | 65.0 |
| 7 | 2.10  | 60.0 |
| 8 | 2.23  | 56.0 |

**Table 3** STE-5000 Conversion Table

|   | Value |      |
|---|-------|------|
| 1 | 0.00  | 54.0 |
| 2 | 0.540 | 58.0 |
| 3 | 1.27  | 66.0 |
| 4 | 1.56  | 70.0 |
| 5 | 1.80  | 74.0 |
| 6 | 2.01  | 78.0 |
| 7 | 2.50  | 90.0 |
| 8 | 3.00  | 90.0 |

- Add programming in program area #5 to set the Heat and Cool set points. The following lines will set the Heating set point to 2° below the set point shown on the thermostat dial and the Cooling set point to 2° above the thermostat dial setting.

*Example:*  
 10 VAR29 = IN2 - 2  
 20 VAR30 = IN2 + 2

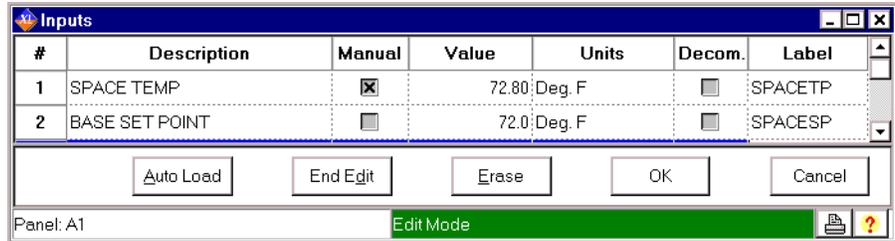
## Using the NetSensor

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7001, change the PID controllers #1 and #3 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor.

## Setback (unoccupied) setpoints

Many applications require “setback” or “setup” temperatures as a means of energy savings in unoccupied periods. To accomplish this, it is necessary to establish a condition of occupancy and define when and how these modes occur.

A variable can be set up as a “flag” to indicate the current mode of operation. The units “Un/Occupy” work well for this purpose.

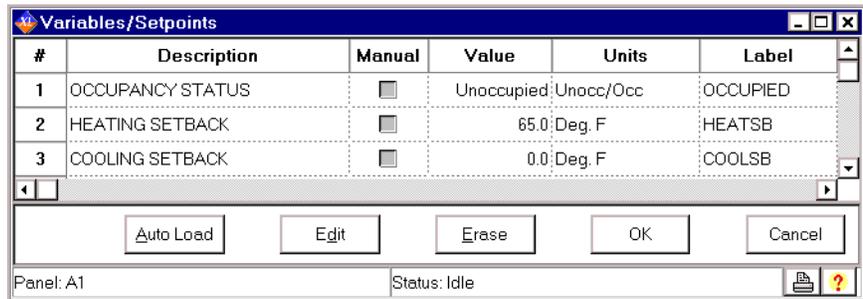


**Illustration 7 Input option for occupancy setpoints**

Program 5 can contain the code needed to change the state of this Variable based on a weekly schedule as shown in this example:

```
10 IF WS1 THEN START OCCUPIED ELSE STOP OCCUPIED
```

The flag “OCCUPIED” (VAR1 in this example) is “ON” in the occupied mode and off in the unoccupied mode. Two more variables can be set up to define the heating and / or cooling set-back temperatures.



**Illustration 8 Alternate variables**

Now, some simple programming can be written to determine the set point based on the occupancy mode:

```
10 IF OCCUPIED THEN HEAT-SP = SPACESP - 2 , COOL-SP = SPACESP + 2
20 IF NOT OCCUPIED THEN HEAT-SP = HEATSB , COOL-SP = COOLSB
```

When in the occupied mode, the heating set point is 2 degrees less than the set point on the thermostat dial and the cooling set point is 2 degrees above. In the unoccupied mode (line 20), the heating set point is set to whatever value is in VAR2 and the cooling set point is set equal to VAR3. It is not necessary to use variables for heating set-back or cooling set-up, but doing so makes modification of these values much easier than getting into the program to change them.

## **Proportional reheat**

A short line of code can be entered into Program area 5 to control hot water reheat. Since Controller #3 (CON3) is set up as a reverse acting thermostat, its value will change from 0 to 100% as temperature falls below the heating setpoint. Configure the output to a range of 0-100% then use:

```
40 OUT1 = CON3
```

## **Other options**

One of the biggest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the KMC supplied sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated. The WinControl XL User's Manual provides help and examples for programming Control Basic.

**WinControl settings** The following tables display the WinControl XL Plus settings for this application note.

**Table 4 Input configuration**

|   | Description      | Manual | Value      | Units        | Average | Label    |
|---|------------------|--------|------------|--------------|---------|----------|
| 1 | ROOM TEMPERATURE | auto   | 74.0 deg.F | K10K-40->250 | 64      | ROOMTEMP |
| 2 |                  | auto   | 0.00       | Unused       |         |          |
| 3 |                  | auto   | 0.00       | Unused       |         |          |
| 4 | AIR FLOW SENSOR  | auto   | 1115       | Table 2      | 64      | FLAWSNSR |

**Table 5 Output configuration**

|   | Description        | Manual | Output     | Units     | 0 % | 100 | Delay | Security | Label |
|---|--------------------|--------|------------|-----------|-----|-----|-------|----------|-------|
| 1 | STAGE 1            | auto   | Off        | Off / On  |     |     | 0:00  | 0        |       |
| 2 | STAGE 2            | auto   | Off        | Off / On  |     |     | 0:00  | 0        |       |
| 3 | STAGE 3            | auto   | Off        | Off / On  |     |     | 0:00  | 0        |       |
| 4 | DAMPER MOTOR DRIVE | auto   | 5.00 Volts | 0.0 -> 10 | 0.0 | 10  |       | 0        | MOTOR |

**Table 6 Variables**

|      | Description                   | Manual | Value    | Units      | Label     |
|------|-------------------------------|--------|----------|------------|-----------|
| 1-16 | (Available for user programs) |        |          |            |           |
| 17   | CHANGEOVER MODE               | auto   | auto     | Cool       | Cool/Heat |
| 18   | CLOCKWISE TO CLOSE ?          | auto   | auto     | Yes        | No / Yes  |
| 19   | MINIMUM FLOW                  | auto   | auto     | 500.00     | MIN-FLOW  |
| 20   | MAXIMUM FLOW                  | auto   | auto     | 2000.00    | MAX-FLOW  |
| 21   | VOLUME CONVERSION             | auto   | auto     | 1.00       | VOLFACTR  |
| 22   | SENSOR CORRECTION             | auto   | auto     | 1.00       |           |
| 23   | ACTUAL FLOW                   | auto   | auto     | 0.00       | ACT-FLOW  |
| 24   | REQUESTED FLOW                | auto   | auto     | 0.00       | RQSTFLOW  |
| 25   | ACTIVE TEMP SETPOINT          | auto   | auto     | 74.00      | ACTIVESP  |
| 26   | DUCT VOLUME READOUT           | auto   | 0.00     |            | VOLUME1   |
| 27   | REHEAT STATUS                 | auto   | Disabled | Dis/Enable | REHEAT    |
| 28   | AUXILLARY FLOW                | auto   | auto     |            | AUX-FLOW  |
| 29   | HEAT TEMP SETPOINT            | auto   | 68.00    | deg.F      | HEAT-SP   |
| 30   | COOL TEMP SETPOINT            | auto   | 74.00    | deg.F      | COOL-SP   |
| 31   | DAMPER DRIVE TIME             | auto   | 300.00   | Sec.       | DRV-TIME  |
| 32   | DAMPER POSITION               | auto   | 50.00    | %          | DMPR-POS  |

**Table 7 Controllers**

|   | In Label | In Value   | A/M  | Output | Setpoint | Value | Action | Prop | Reset | Rate | Bias |    |
|---|----------|------------|------|--------|----------|-------|--------|------|-------|------|------|----|
| 1 | ROOMTEMP | 0.00 deg.F | auto | 50.0 % | ACTIVESP | 74.0  | +      | 2    | 0     | H    | 0.00 | 50 |
| 2 | ACT-FLOW | 0.00       | auto | 0.0 %  | RQSTFLOW | 1250  | +      | 200  | 0     | H    | 0.00 | 50 |
| 3 | ROOMTEMP | 0.00 deg.F | auto | 0.0 %  | HEAT-SP  | 68.0  | -      | 2    | 0     | H    | 0.00 | 0  |
| 4 |          | 0.00       | auto | 0.0 %  |          | 0.00  | +      | 0    | 0     | H    | 0.00 | 0  |

**Table 8 Tables**

|    | TABLE 1<br>Value | TABLE 2<br>Value | TABLE 3<br>Value |       |       |       |
|----|------------------|------------------|------------------|-------|-------|-------|
| 1  | 0.00             | 0.00             | 0.00             | 0.00  | 0.00  | 0.00  |
| 2  | 0.00             | 0.00             | 0.140            | 0.00  | 0.350 | 0.00  |
| 3  | 0.00             | 0.00             | 0.200            | 75.0  | 0.460 | 276.0 |
| 4  | 0.00             | 0.00             | 0.400            | 327.0 | 0.860 | 566.0 |
| 5  | 0.00             | 0.00             | 0.800            | 563.0 | 1.10  | 700.0 |
| 6  | 0.00             | 0.00             | 1.00             | 664.0 | 1.70  | 1112  |
| 7  | 0.00             | 0.00             | 1.50             | 904.0 | 2.30  | 1500  |
| 8  | 0.00             | 0.00             | 2.00             | 1143  | 2.50  | 1648  |
| 9  | 0.00             | 0.00             | 2.50             | 1425  | 2.90  | 1875  |
| 10 | 0.00             | 0.00             | 3.00             | 1707  | 3.25  | 2087  |
| 11 | 0.00             | 0.00             | 3.50             | 2051  | 3.90  | 2531  |
| 12 | 0.00             | 0.00             | 4.00             | 2394  | 4.30  | 2800  |
| 13 | 0.00             | 0.00             | 4.50             | 2816  | 4.60  | 3000  |
| 14 | 0.00             | 0.00             | 5.00             | 3237  | 5.00  | 3000  |
| 15 | 0.00             | 0.00             | 5.50             | 3237  | 6.00  | 3000  |

**Table 9 NET SENSOR #1**

| Button | Point                | Value | Type   | Range     | Display Only |
|--------|----------------------|-------|--------|-----------|--------------|
| 1      |                      | 0.1   | Analog | Deg.F     |              |
| 2      | ACTIVE TEMP SETPOINT | 0     | Analog | 0 Decimal | Can Change   |
| 3      | DUCT VOLUME READOUT  | 0     | Analog | 0 Decimal | Display Only |
| 4      | VOLUME CONVERSION    | 0.00  | Analog | 2 Decimal | Can Change   |
| 5      | MINIMUM FLOW         | 0     | Analog | 0 Decimal | Can Change   |
| 6      | SENSOR CORRECTION    | 0.00  | Analog | 2 Decimal | Can Change   |
| 7      | MAXIMUM FLOW         | 0     | Analog | 0 Decimal | Can Change   |
| Aux    | BALANCER override    | Off   | State  | Off/On    |              |

**Table 10 Control Basic program list**

|   | <b>Description</b>   | <b>Run</b> | <b>A/M</b> | <b>Exit</b> | <b>Label</b> |
|---|----------------------|------------|------------|-------------|--------------|
| 1 | SINGLE DUCT VAV      | Yes        | auto       | No          |              |
| 2 | TIME PROPORTIONAL HT | No         | auto       | No          |              |
| 3 | 3-STAGE REHEAT       | No         | auto       | No          |              |
| 4 | BALANCER override    | No         | auto       | No          |              |
| 5 | USER PROGRAMS        | No         | auto       | No          |              |

### Program 1– SINGLE DUCT VAV

```
1 REM BALANCER OVRD
5 IF TIME-ON( BALANCER~override ) > 0:30:00 THEN STOP
BALANCER~override
7 IF BALANCER~override THEN END

10 IF CHANGEOVER~MODE THEN ACTIVE~TEMP~SETPOINT =
HEAT~TEMP~SETPOINT ELSE ACTIVE~TEMP~SETPOINT =
COOL~TEMP~SETPOINT
20 IF CHANGEOVER~MODE THEN A = 1 - CON1 / 100 ELSE A = CON1 /
100
30 REQUESTED~FLOW = ( ( MAXIMUM~FLOW - MINIMUM~FLOW~ ) * A +
MINIMUM~FLOW~ ) / VOLUME~CONVERSION
40 ACTUAL~FLOW~ = AIR~FLOW~SENSOR * SENSOR~CORRECTION : REM
ACTUAL FLOW CORRECTED
50 DUCT~VOLUME~READOUT = ACTUAL~FLOW~ * VOLUME~CONVERSION : REM
LIVE VOLUME

60 IF CON2 > 70 AND CLOCKWISE~TO~CLOSE~? THEN START B
70 IF CON2 > 70 AND NOT CLOCKWISE~TO~CLOSE~? THEN START C
80 IF CON2 < 30 AND CLOCKWISE~TO~CLOSE~? THEN START C
90 IF CON2 < 30 AND NOT CLOCKWISE~TO~CLOSE~? THEN START B
100 IF CON2 > 35 AND CON2 < 65 THEN STOP B , STOP C
110 IF+ B THEN STOP C
120 IF+ C THEN STOP B

125 X = 1 / SCANS
130 IF CON2 < 99 AND CON2 > 1 THEN 140 ELSE 210
140 IF CON2 < 36 OR CON2 > 64 THEN START D ELSE STOP D
150 IF TIME-ON( D ) > 0:04:00 THEN 210
170 Y = X + Y
180 IF Y > 5 THEN Y = 0
190 Z = ABS( CON2 - 50 ) / 50
200 IF Y > Z THEN STOP B , STOP C

210 IF B THEN DAMPER~MOTOR~DRIVE = 10
220 IF C THEN DAMPER~MOTOR~DRIVE = 0
230 IF NOT B AND NOT C THEN DAMPER~MOTOR~DRIVE = 5
234 IF DAMPER~MOTOR~DRIVE > 7 THEN START G ELSE STOP G
236 IF DAMPER~MOTOR~DRIVE < 3 THEN START H ELSE STOP H
240 IF TIME-ON( G ) > 0:06:00 OR TIME-ON( H ) > 0:06:00 THEN
DAMPER~MOTOR~DRIVE = 5 , START I ELSE STOP I
246 IF TIME-ON( I ) > 6:00:00 THEN STOP G , STOP H , STOP B ,
STOP C

250 IF CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE < 3 THEN
DAMPER~POSITION = DAMPER~POSITION + X * 100 / DAMPER~DRIVE~TIME
260 IF CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE > 7 THEN
DAMPER~POSITION = DAMPER~POSITION - X * 100 / DAMPER~DRIVE~TIME
270 IF NOT CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE < 3 THEN
DAMPER~POSITION = DAMPER~POSITION - X * 100 / DAMPER~DRIVE~TIME
280 IF NOT CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE > 7 THEN
DAMPER~POSITION = DAMPER~POSITION + X * 100 / DAMPER~DRIVE~TIME

290 IF DAMPER~POSITION > 100 THEN DAMPER~POSITION = 100
300 IF DAMPER~POSITION < 0 THEN DAMPER~POSITION = 0
310 END
```

**Program 2-TIME PROPORTIONAL HT**

```

20 REM ----- TIME PROPORTIONAL REHEAT SEQUENCE
-----
25 REM -- Designed to drive a solid state relay connected to
electric reheat.
26 REM Time base is fixed at 10 seconds - ie, a call for 50%
heat turns on
27 REM relay (OUT1) for 5 seconds, off for 5 seconds.
28 REM Uses CON#3 for control of reheat operation! ( Full PID
function )
30 IF NOT REHEAT~STATUS THEN STOP STAGE~1 , END
40 K = 1 / SCANS
50 L = L + K : IF L > 10 THEN L = 0
60 IF L < CON3 / 10 THEN START STAGE~1 ELSE STOP STAGE~1
70 END
    
```

**Program 3 3-STAGE REHEAT**

```

10 REM ----- 3-STAGE REHEAT SEQUENCE
-----
20 REM -- STAGES ON AT 30, 60, 90 % OF PROPORTIONAL BAND BELOW
SETPOINT.
30 REM Auxillary flow setpoint is used only when AUX-FLOW
(VAR28) is not
40 REM zero and anytime stage 1 heat (OUT1) is requested.
50 REM Entire reheat sequence is easily disabled by toggling
VAR27 on-off.
60 IF NOT REHEAT~STATUS THEN STOP STAGE~1 , STOP STAGE~2 , STOP
STAGE~3 , STOP A , END
70 IF CON3 > 30 THEN START STAGE~1 , START A
80 IF CON3 > 60 THEN START STAGE~2
90 IF CON3 > 90 THEN START STAGE~3
100 IF CON3 < 80 THEN STOP STAGE~3
110 IF CON3 < 50 THEN STOP STAGE~2
120 IF CON3 < 20 THEN STOP STAGE~1 , STOP A
130 IF AUXILLARY~FLOW = 0 THEN 150 : REM Since Auxillary flow is
zero its not used
140 IF A THEN REQUESTED~FLOW = AUXILLARY~FLOW /
VOLUME~CONVERSION
150 END
    
```

### Program 4–BALANCER override

```
10 REM **** Balancer control of unit to either MIN or MAX flow
***
20 REM ** Uses var25 activesp in digital fashion to select
desired flow **

25 IF+ ACTIVE~TEMP~SETPOINT > 1 THEN ACTIVE~TEMP~SETPOINT = 1
27 IF+ ACTIVE~TEMP~SETPOINT < 0 THEN ACTIVE~TEMP~SETPOINT = 0
30 IF ACTIVE~TEMP~SETPOINT THEN REQUESTED~FLOW = MAXIMUM~FLOW /
VOLUME~CONVERSION ELSE REQUESTED~FLOW = MINIMUM~FLOW~ /
VOLUME~CONVERSION
40 ACTUAL~FLOW~ = AIR~FLOW~SENSOR * SENSOR~CORRECTION : REM
ACTUAL FLOW CORRECTED (FPM)
50 DUCT~VOLUME~READOUT = ACTUAL~FLOW~ * VOLUME~CONVERSION : REM
LIVE VOLUME READOUT - UNITS MATCH VOLUME

60 IF CON2 > 70 AND CLOCKWISE~TO~CLOSE~? THEN START B
70 IF CON2 > 70 AND NOT CLOCKWISE~TO~CLOSE~? THEN START C
80 IF CON2 < 30 AND CLOCKWISE~TO~CLOSE~? THEN START C
90 IF CON2 < 30 AND NOT CLOCKWISE~TO~CLOSE~? THEN START B
100 IF CON2 > 35 AND CON2 < 65 THEN STOP B , STOP C
110 IF+ B THEN STOP C
120 IF+ C THEN STOP B

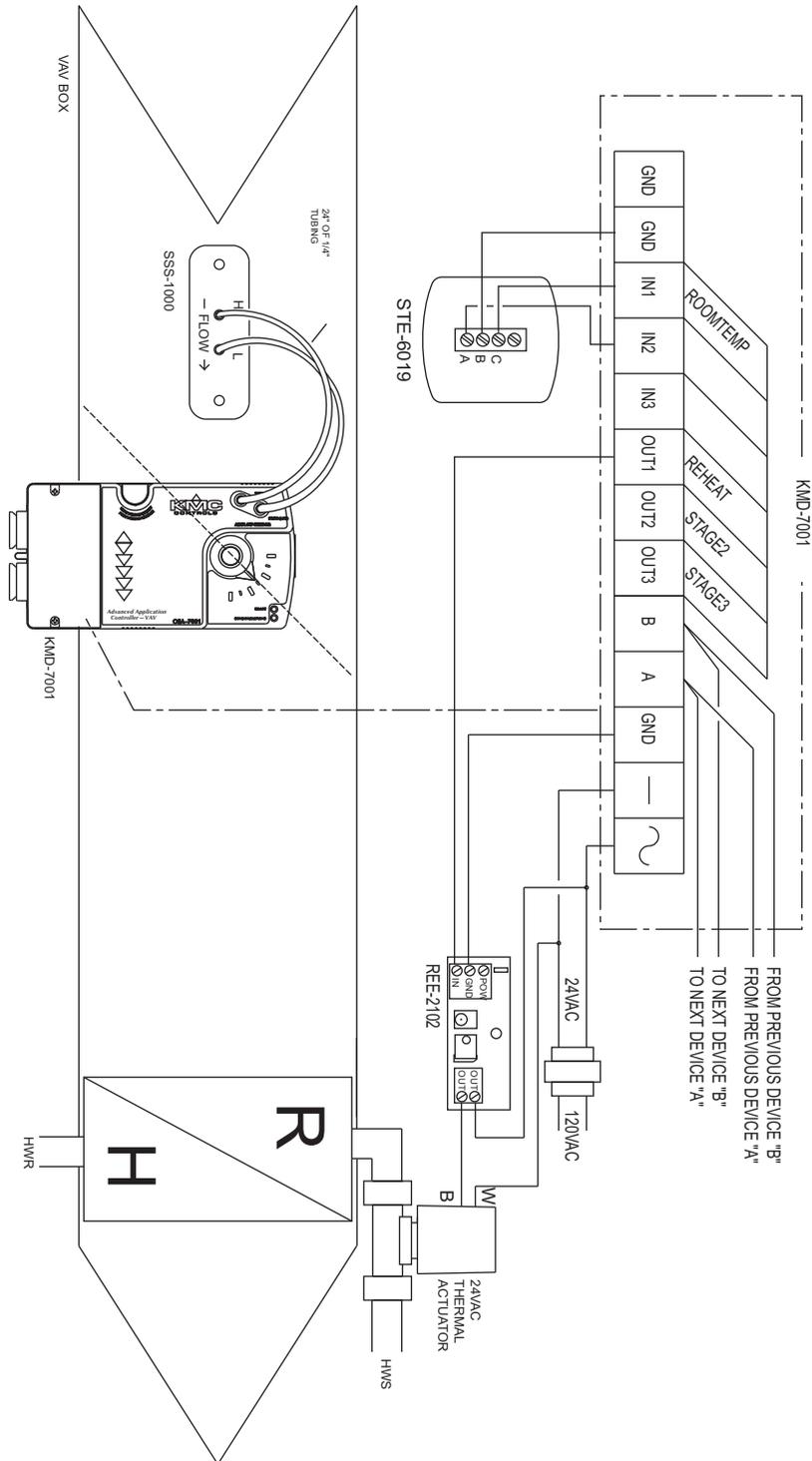
125 X = 1 / SCANS
130 IF CON2 < 99 AND CON2 > 1 THEN 140 ELSE 210
140 IF CON2 < 36 OR CON2 > 64 THEN START D ELSE STOP D
150 IF TIME-ON( D ) > 0:04:00 THEN 210
170 Y = X + Y
180 IF Y > 5 THEN Y = 0
190 Z = ABS( CON2 - 50 ) / 50
200 IF Y > Z THEN STOP B , STOP C

210 IF B THEN DAMPER~MOTOR~DRIVE = 10
220 IF C THEN DAMPER~MOTOR~DRIVE = 0
230 IF NOT B AND NOT C THEN DAMPER~MOTOR~DRIVE = 5

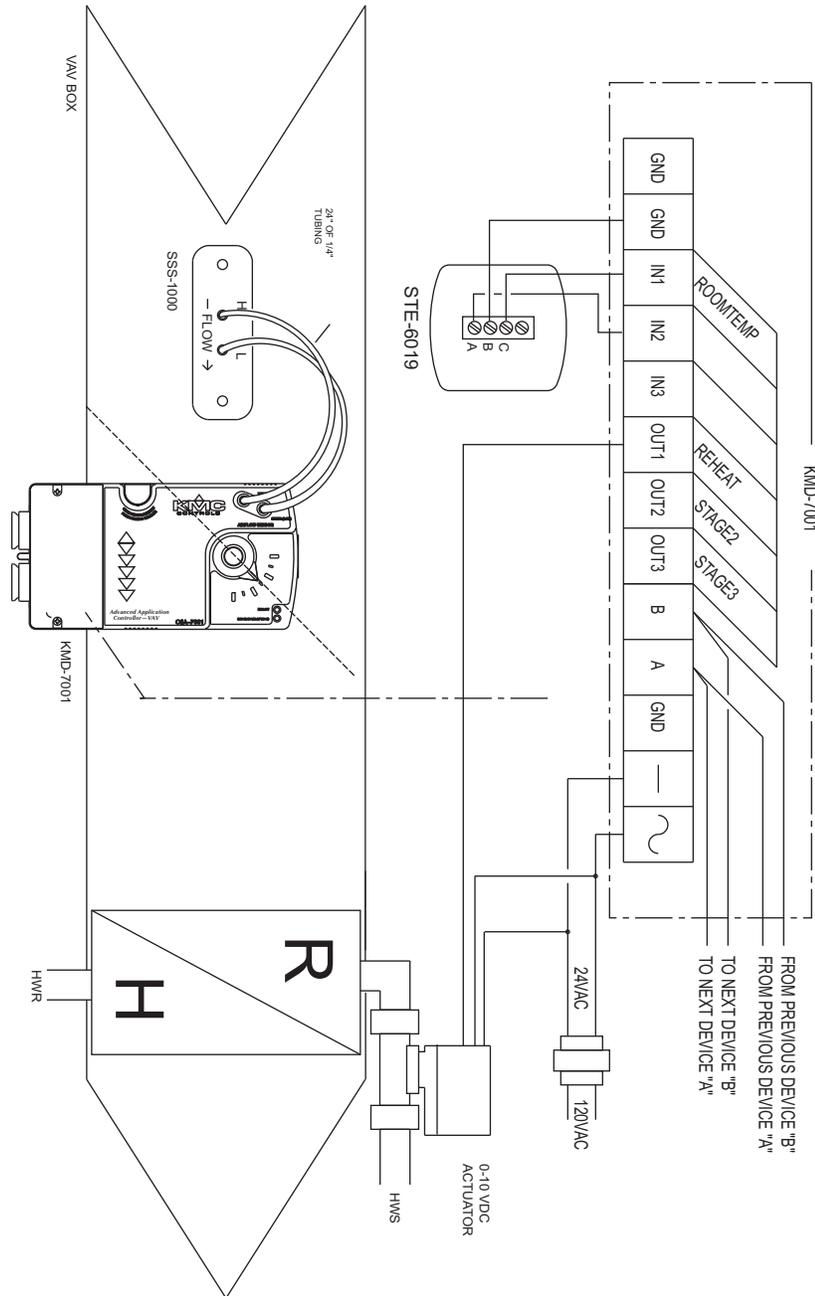
250 IF CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE < 3 THEN
DAMPER~POSITION = DAMPER~POSITION + X * 100 / DAMPER~DRIVE~TIME
260 IF CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE > 7 THEN
DAMPER~POSITION = DAMPER~POSITION - X * 100 / DAMPER~DRIVE~TIME
270 IF NOT CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE < 3 THEN
DAMPER~POSITION = DAMPER~POSITION - X * 100 / DAMPER~DRIVE~TIME
280 IF NOT CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE > 7 THEN
DAMPER~POSITION = DAMPER~POSITION + X * 100 / DAMPER~DRIVE~TIME

290 IF DAMPER~POSITION > 100 THEN DAMPER~POSITION = 100
300 IF DAMPER~POSITION < 0 THEN DAMPER~POSITION = 0
310 END
```

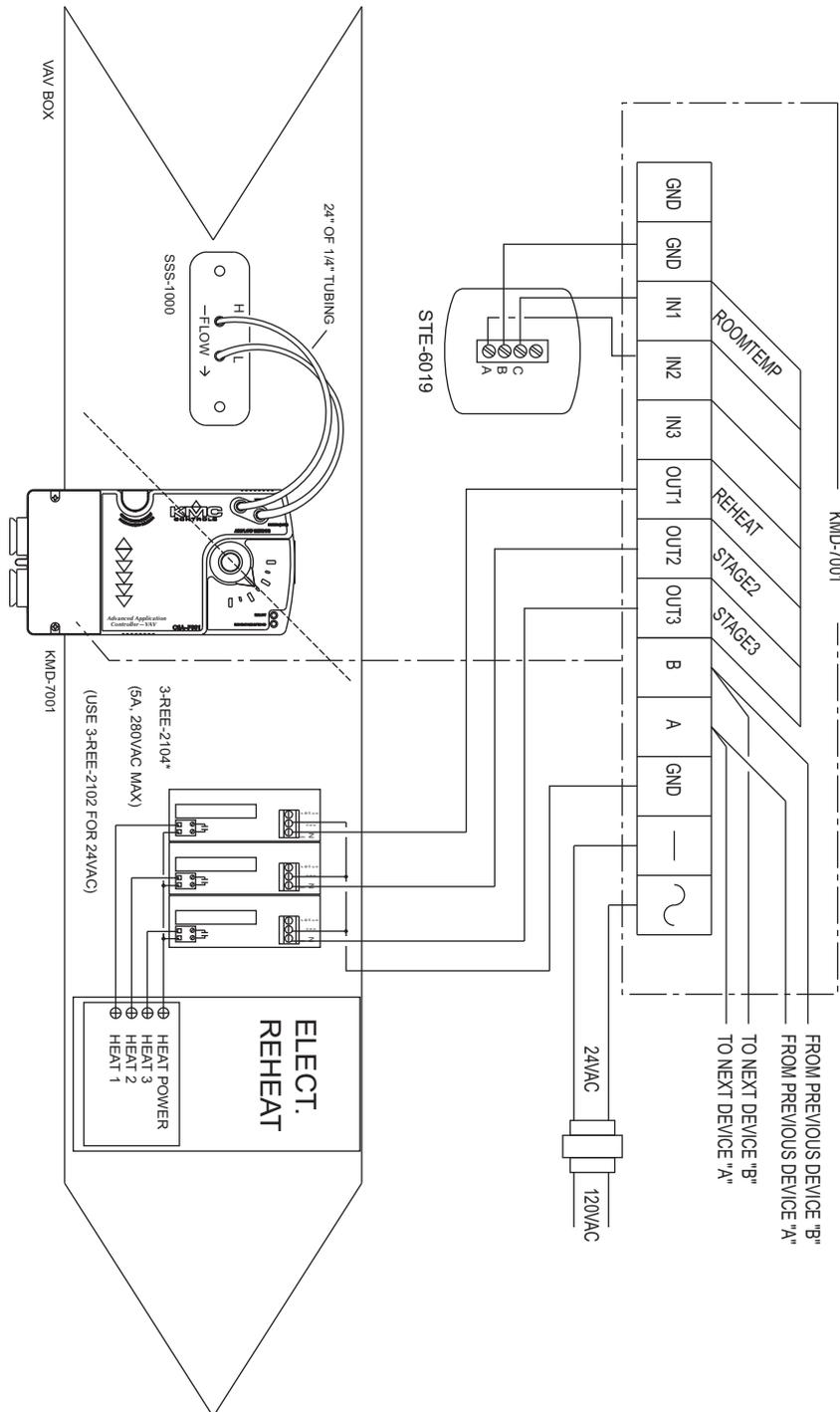
**Application drawings**



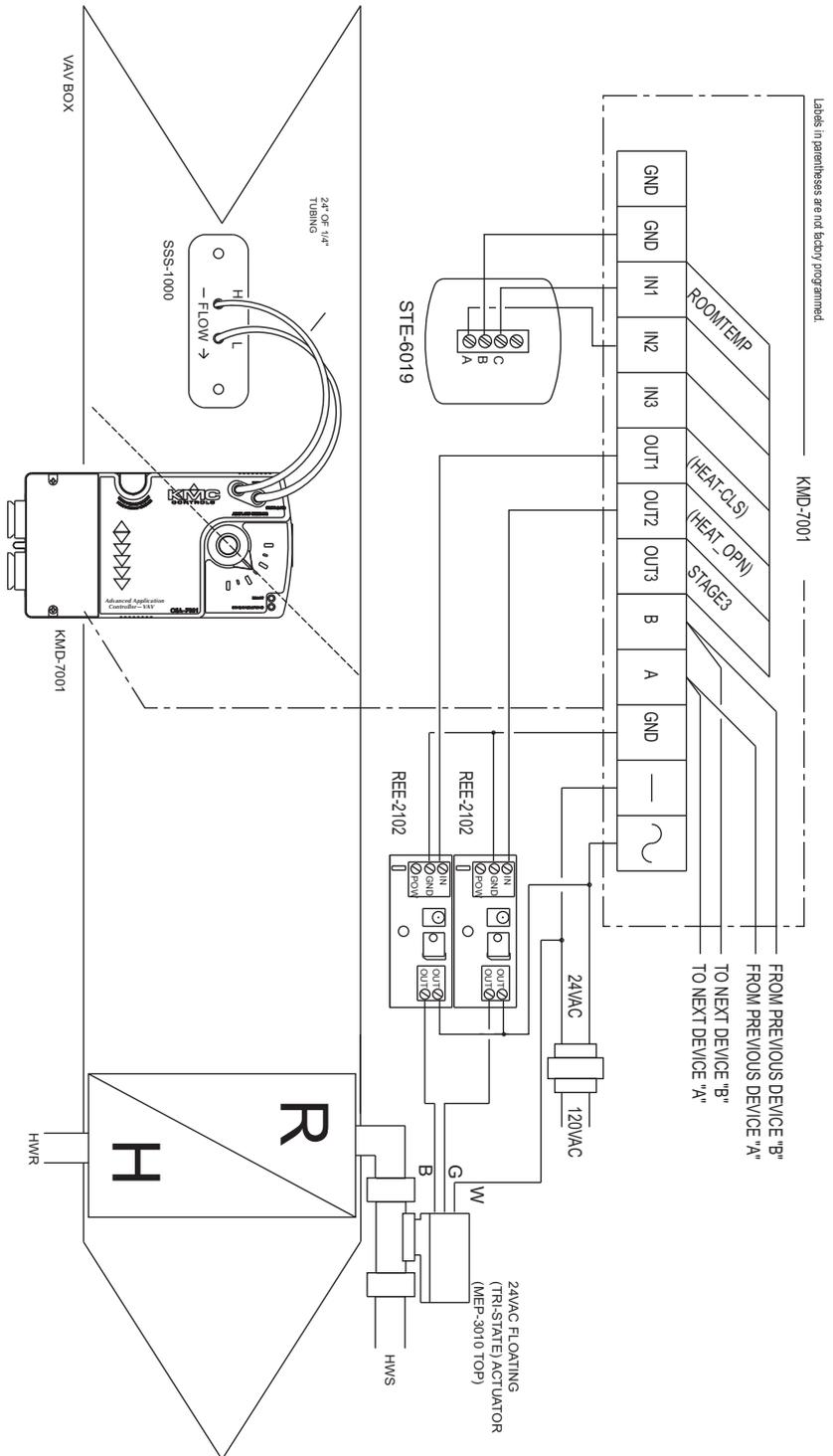
**Illustration 9 Variable Air Volume with 24 volt AC reheat valve**



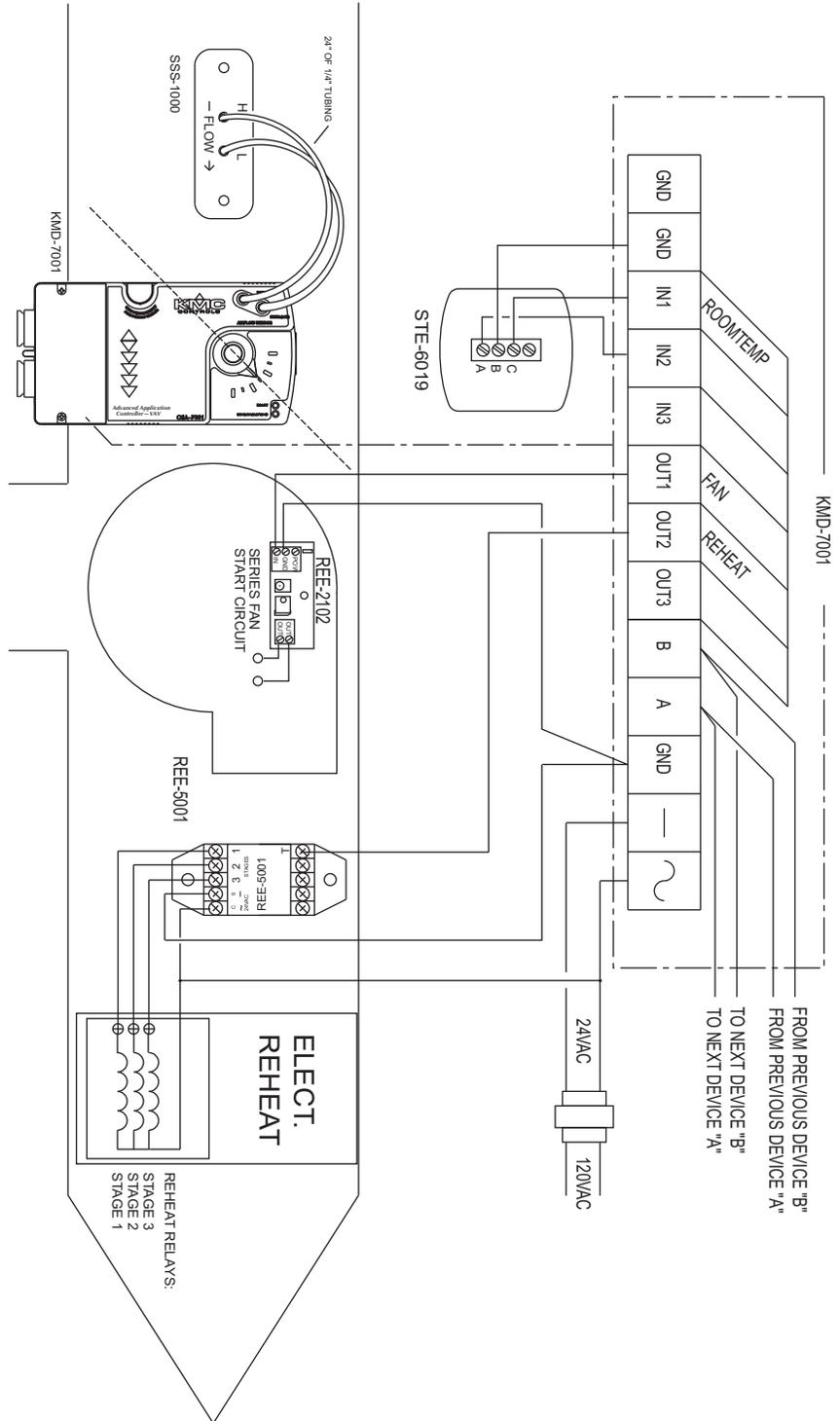
**Illustration 10 Variable Air Volume with 0-10 volt DC reheat valve**



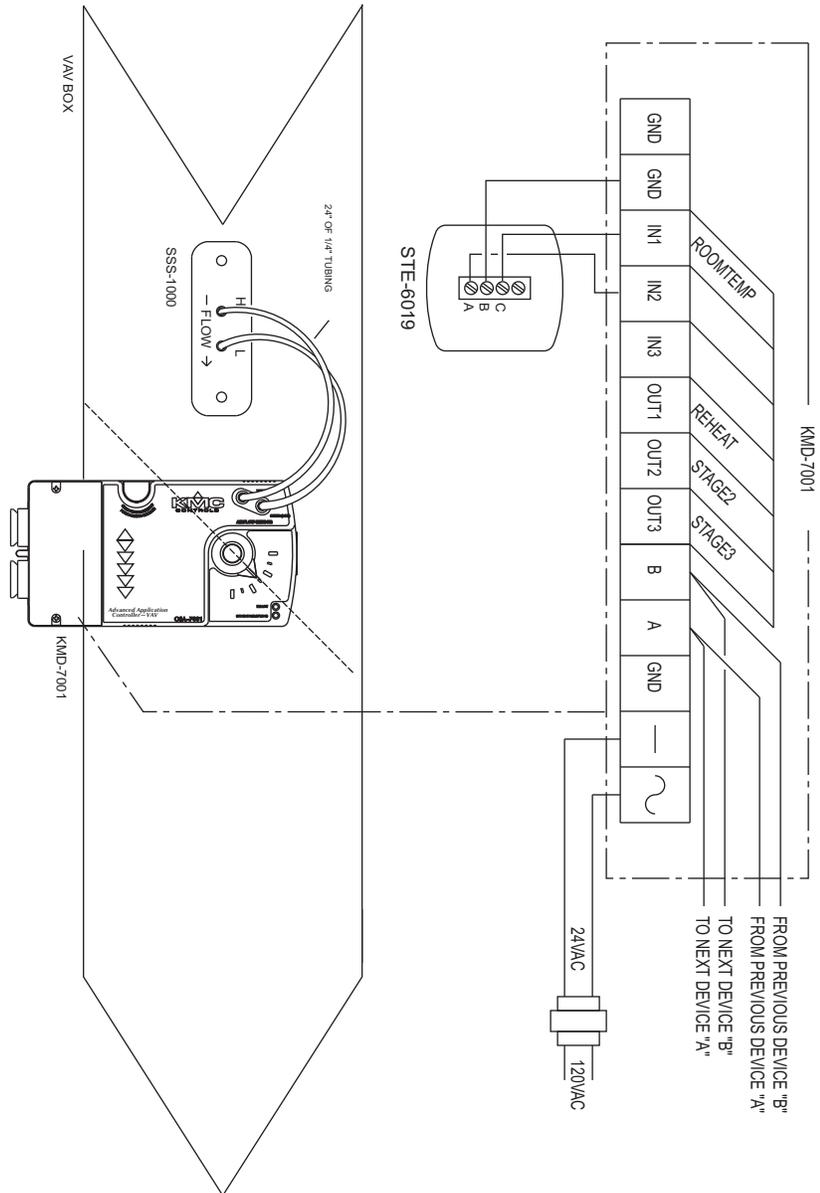
**Illustration 11 Variable Air Volume with three-stage electric reheat**



**Illustration 12 VAV with 24 volt AC tri-state reheat valve**



**Illustration 13** VAV with three-stage electric reheat and series fan



**Illustration 14 VAV with cooling only**

## Application Note–AN0404C

---

Revision A

### **KMD–7002 and KMD–7052 VAV Controller**

This application note describes using a KMD–7002 and KMD–7052 for dual-duct, variable air HVAC control.

---

#### **Related materials**

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ Installation and operation guides for KMD–7002, TSP–6001 and STE–5012
- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide
- ◆ System plans with controller addresses
- ◆ AN0805A, The NetSensor Balancer Program

#### **Introduction**

The KMD-7002 is a KMC DDC Controller and an actuator built into one unit. A pressure-independent, pre-programmed VAV sequence automatically modulates flow into the conditioned space between Minimum and Maximum setpoints as defined by the user. This model is programmed to control the TSP–6001 slave duct controller. User programming can customize each application as required.

If the application does not require the factory setup and programming options, the user can send a panel file with his own setup parameters and programming to the controller. This will remove all of the factory configurations allowing custom programming the desired sequence.

This feature allows the user to custom tailor the control sequence to match specific applications. The following paragraphs explain the pre-programmed control sequence in detail and offer suggestions for user-programming to customize operation of the controllers.

The KMD-7002 VAV Terminal Controller contains the basic programming sequence required for two specific applications:

- ◆ Single Duct VAV
- ◆ Slave Duct VAV

#### **Velocity sensor**

The KMD-7002 incorporates an on-board flow-through type sensor to determine duct airflow. Airflow through the sensor is approximately 1.12 SCFH (32 liters/hour) at .25 INWC (62.5 Pa) pressure drop. The sensor uses two RTD's (resistance temperature detectors) to measure flow. The RTD's are platinum films

on a ceramic base chosen for long-term stability and accuracy. The smaller 100 ohm RTD is self-heated to a constant temperature differential above ambient. The amount of power required varies with the specific airflow. A larger 1000-ohm RTD constantly senses ambient temperature and compensates. The sensor housing channels the airflow over the self-heated RTD in a manner to isolate its heat from the compensator. This minimizes shifts at low flows due to various mounting positions. The sensor housing itself is molded from plastic that carries the UL94-5V rating. The sensor ports are .188-inch diameter and readily accept .25 inch O.D. "FR" tubing (.040 wall) or equivalent.

## Velocity limits

Minimum and Maximum Flow velocity limits can be defined through KMC digital operating software. Setting velocity limits through the KMC digital software is as simple as defining the value of a variable. Using this method the limits may be entered in any units desired (CFM, L/s, Pa, etc.) as long as a "conversion factor" is also provided. The "default" settings are always in Feet Per Minute (FPM). For details on using this method refer to the following section on standard VAV operation.

## Damper actuator

The actuator is capable of supplying a minimum of 50 inch pounds of torque to open or close the damper according to instructions from the KMD control sequence. The motor is an AC synchronous type for smooth, reliable operation and is coupled with a magnetic slip-clutch design that allows continuously stalling damper rotation without damage. A time-out function will stop the motor anytime the unit has run continuously in one direction for longer than 6 minutes. This ensures that the stops have been reached without needlessly running the motor.

Physical adjustable end-stops allow limiting rotation to less than the 90° rotation and a gear

disengagement mechanism on the side of the case can be used for fast and simple installation and set-up. The red disengagement button is located on the left side of the actuator.

## Standard Variable Air Volume (VAV) Operation

The standard VAV sequence of operation has been pre-programmed into each KMD-7002 controller. Default operation is defined as:

- ◆ Dual Duct Cooling
- ◆ Clockwise to close damper rotation
- ◆ Minimum and Maximum Limits at 500 and 2000 FPM respectively
- ◆ Cooling setpoint at 74°F
- ◆ Space Temperature Sensor connected to IN1

These units are capable of independent, stand-alone operation or they may be networked together with other KMC digital products. If stand-alone operation is desired, supplying 24 VAC power and connecting a 10K thermistor (STE-series) to IN1 will enable full operation to the default parameters listed above. If these units will be networked together it may be necessary to configure each panel with a unique address and other set-up options using HCM.

## Hardware Configuration Manager (HCM)

For networked operation, use the HCM software to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 1 Default Configuration for KMD-7002**

|                   |       |
|-------------------|-------|
| Panel number      | 124   |
| Network Baud Rate | 38400 |
| Last Panel No     | No    |
| Panel Type        | 123   |

## Setting up with WinControl XL

This section covers the set-up requirements to activate any of the pre-programmed options available in this device. At this point, communications should have been established with the unit.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence.

The first step after signing on to the panel is to "Load Descriptors One" or "Load Descriptors All". Many of the descriptions have been pre-defined for variables, outputs, etc. to aid in configuring the desired options. To use either program for box 1 or 2, go to the Control Basic programs screen and turn "ON" the appropriate program.

You can use the setup screen in Control Group 2 (GRP2) which lists the items most likely to be monitored. The following section outlines each of the parameters that have been pre-programmed into the device. Details on customizing the unit can be found in the section titled "User Programming".

### Configuring inputs

Inputs 1 and 2 can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:

| # | Description         | Manual                   | Value   | Units            | Decom.                              | Label    |
|---|---------------------|--------------------------|---------|------------------|-------------------------------------|----------|
| 1 | ROOM TEMPERATURE    | <input type="checkbox"/> | 54.14   | Deg. F           | <input checked="" type="checkbox"/> | ROOMTEMP |
| 2 |                     | <input type="checkbox"/> | 0.00    | Unused           | <input type="checkbox"/>            |          |
| 3 | FLOW SENSOR DUCT #2 | <input type="checkbox"/> | 3474.72 | Unused (Table 2) | <input checked="" type="checkbox"/> | FLOSENS2 |
| 4 | FLOW SENSOR DUCT #1 | <input type="checkbox"/> | 638.30  | Unused (Table 2) | <input type="checkbox"/>            | FLOSENS1 |

**Illustration 1** Input configuration in WinControl XL

**Input 1–ROOM TEMPERATURE** Default selects a KMC 10K ohm thermistor as used in all of KMC Controls STE series sensors.

**Input 2** Unused

**Input 3–FLOW SENSOR DUCT 2** This is the flow voltage signal from the “VEL” terminal of the TSP-6001 slave duct controller. Table 2 has been configured to convert the voltage input into feet per minute.

**Input 4–FLOW SENSOR DUCT 1** The on-board flow-sensor. Table 2 has been configured to convert the voltage input into feet per minute. This input is not accessible for any other use.

### Configuring outputs

The KMD-7002 VAV Controller features three universal outputs ( analog or digital, set in software) and an analog output to drive the damper motor. The KMC supplied control routines have been written to use the following output configuration:

| # | Description     | Manual                              | Value | Units  | Label   |
|---|-----------------|-------------------------------------|-------|--------|---------|
| 1 |                 | <input checked="" type="checkbox"/> | Off   | Off/On |         |
| 2 |                 | <input type="checkbox"/>            | Off   | Off/On |         |
| 3 | DAMPER MOTOR #2 | <input type="checkbox"/>            | 5.00  | Volts  | MOTOR#2 |
| 4 | DAMPER MOTOR #1 | <input type="checkbox"/>            | 10.00 | Volts  | MOTOR#1 |

**Illustration 2** Output configuration in WinControl XL

**Output 1–Not Used** This output is available for additional control devices

**Output 2–Not Used** This output is available for additional control devices

**Output 3–DAMPER MOTOR 2** This output goes to the “MOTOR” terminal on the TSP-6001 slave duct controller. It is used to control the slave damper position.

**Output 4–DAMPER MOTOR 1** This output is used to control the main damper position. It is not accessible for any other use on the KMD-7002.

### Configuring variables

A total of thirty-two system variables are available to perform various functions such as set points, limits, and modes. All variables can be changed or modified to suit the specific application. The following have been pre-defined for use in the KMD-7002 factory supplied control sequences:

**Variable 15–COOLING STAT SETPT** The space cooling set point.

**Variable 16–HEATING STAT SETPT** The space heating set point

**Variable 17–MODE BOX 1 (COOL/HEAT)** Chooses whether the unit will operate as a heating or cooling box by automatically setting the thermostat action as direct (COOL) or reverse (HEAT).

**Variable 18–CLOCKWISE TO CLOSE1 (NO/YES)** Defines which way the damper will rotate to increase or decrease air-flow as required. “NO” will close the box in a counter-clockwise direction.

**Variable 19–MINIMUM FLOW BOX 1** Sets the lowest design flow for the VAV box. The default setting is for a Feet-Per-Minute (FPM) value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable 20–MAXIMUM FLOW BOX 1** Sets the highest design flow for the VAV box. The default setting is for a FPM value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable 21–VOLUME CONVERSION BOX 1** Default value is 1.00 (no conversion). This factor allows all flow settings and live readings to be in any desired units. Since default calculations are in FPM, the duct area is required (in square feet) to get readings in CFM. Similarly, values can be converted to L/s, Pa, etc. with the proper factors applied.

**Variable 22–SENSOR CORRECTION BOX 1** Default value is 1.00 (no correction required). This factor is provided to allow simple correction for erroneous sensor readings. This factor is multiplied against all flow readings and the product is stored as the ACTUAL FLOW (VAR23).

**Variable 23–ACTUAL FLOW BOX 1** This value is calculated from the flow sensor input (IN4) and the sensor correction variable (VAR22) using the following formula:

$$\text{ACT-FLOW} = \text{FLOWSNSR} * \text{VAR22}$$

Or:

$$\text{VAR23} = \text{IN4} * \text{VAR22}$$

**Variable 24–REQUESTED FLOW BOX 1** This is the flow set point based on the heating or cooling needs of the space. The value is set automatically in Program 1 (PRG1).

**Variable 25–MODE BOX 2 (COOL/HEAT)** Chooses whether the unit will operate as a heating or cooling box by automatically setting the thermostat action as direct (COOL) or reverse (HEAT).

**Variable 26-CLOCKWISE TO CLOSE2 (NO/YES)** Defines which way the damper will rotate to increase or decrease air-flow as required. “NO” will close the box in a counter-clockwise direction.

**Variable 27-MINIMUM FLOW BOX 2** Sets the lowest design flow for the VAV box. The default setting is for a Feet-Per-Minute (FPM) value. Other units can be used by providing a conversion factor in Variable 29 (VAR29).

**Variable 28-MAXIMUM FLOW BOX 2** Sets the highest design flow for the VAV box. The default setting is for a FPM value. Other units can be used by providing a conversion factor in Variable 29 (VAR29).

**Variable 29-VOLUME CONVERSION BOX 2** Default value is 1.00 (no conversion). This factor allows all flow settings and live readings to be in any desired units. Since default calculations are in FPM, the duct area is required (in square feet) to get readings in CFM. Similarly, values can be converted to L/s, Pa, etc. with the proper factors applied.

**Variable 30-SENSOR CORRECTION BOX 2** Default value is 1.00 (no correction required). This factor is provided to allow simple correction for erroneous sensor readings. This factor is multiplied against all flow readings and the product is stored as the ACTUAL FLOW (VAR31).

**Variable 31-ACTUAL FLOW BOX 2** This value is calculated from the flow sensor input (IN3) and the sensor correction variable (VAR30) using the following formula:

$$\text{ACT-FLOW} = \text{FLOWSNR} * \text{VAR30}$$

Or:

$$\text{VAR23} = \text{IN3} * \text{VAR30}$$

**Variable 32-REQUESTED FLOW BOX 2** This is the flow set point based on the heating or cooling needs of the space. The value is set automatically in Program 1 (PRG1).

### Configuring controllers (PID control loops)

KMC digital products feature built-in, full function PID (Proportional, Integral, Derivative) controllers. All parameters may be modified as necessary.

**Controller 1 (CON1)** is a direct-acting thermostat. It compares ROOM TEMPERATURE (IN1) to the COOLING SETPOINT (VAR15) to determine the VAV flow requirement.

**Controller 2 (CON2)** compares the ROOM TEMPERATURE (IN1) to the HEATING SET POINT (VAR16) for operation of the damper in the heating mode.

**Controller 3 and 4 (CON3 and CON4)** are set up as direct-acting thermostats. They compare ACTUAL FLOW (VAR23 and VAR31) to the REQUESTED FLOW (VAR24 and VAR32) to operate the DAMPER motors (OUT3 and OUT4). All values used by the controller have been converted to FPM to allow a common proportional band value regardless of the units selected. Default is a 200 FPM proportional band which relates to approximately 60 FPM “dead band” to stabilize actuator operation. Default parameters do not include any integral or derivative function.



**Note**

The desired air flow is VAR24 (VAR32 for slave box) and is automatically calculated by the VAV pro-gram. For example, to override the airflow to some other value under specific conditions, write a “user” statement such as:

```
10 IF (condition) THEN VAR24 = (value)
```

All values at this point must be in FPM. Condition may be a variable that represents an abnormal event such as a smoke zone activation. To ensure close-off on a box, set the desired flow (value) to a number below zero such as -50. To ensure a wide open damper position use a flow rate higher than 3000 FPM (The highest value the on-board sensor can read).

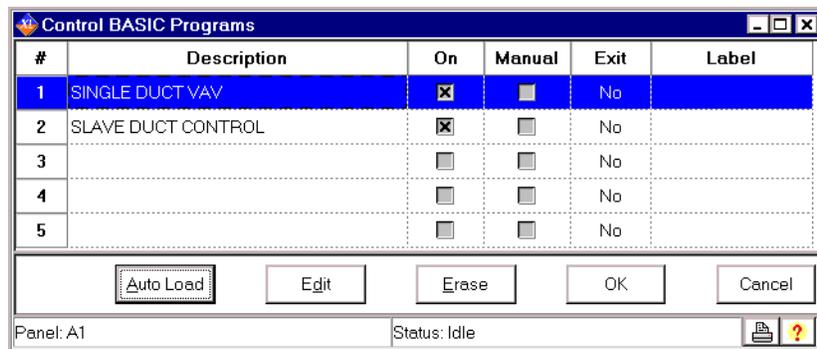
**Control Basic programming**

Control basic programs one and two may be edited if the sequence does not fit the application. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.



**Caution**

This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type)



**Illustration 3** Configuring Control Basic in WinControl XL

**Table 2**

**Program 1–SINGLE DUCT VAV** The necessary air flow (VAR24) is calculated and the damper motor is modulated to maintain this volume.

See [Program 1– SINGLE DUCT VAV](#) on page 3-41 for a complete listing.

**Program 2–SLAVE DUCT VAV** The necessary air flow (VAR32) is calculated and the damper motor is modulated to maintain this volume.

See [Program 2– SLAVE DUCT CONTROL](#) on page 3-41 for a complete listing.

**Program 3** Available for user programming.

**Program 4** Available for user programming.

**Program 5** Available for user programming.

### Configuring tables

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7002:

**Table1** Ready for user programming.

**Table 2** Correlates velocity pressure from the on-board flow sensor to feet per minute.

**Table 3** Correlates velocity pressure of CSP-5001 VAV controller to feet per minute.



#### Note

These tables use nominal values for correlation. Because of component tolerances and the resistance of field wiring, these values may need correction on an individual basis. If the value reading is too low, enter a positive calibration in the input column labeled “cal”. If the value read is high, a negative calibration can be entered. If necessary, reconfigure Table 2 and/or Table 3. The complete Table 2 and Table 3 from the factory are shown in figure 3. If the tables have been reconfigure and the user wants the factory configuration for Table 2 and Table 3 it will be necessary to use Clear Panel or re-enter the values.

**!** **Caution**

Using Clear Panel will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type.)

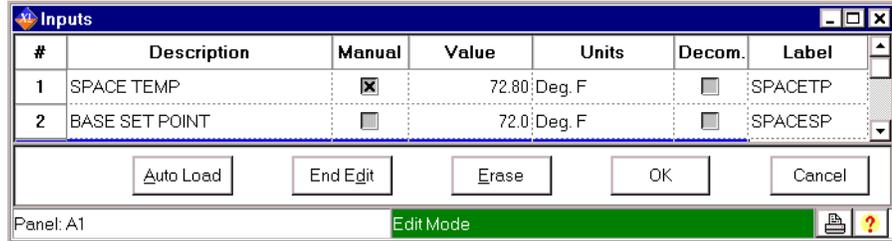
| #  | Table 1 | Unused | Table 2 | Unused | Table 3 | Unused |
|----|---------|--------|---------|--------|---------|--------|
| 1  | 0       | 0      | 0       | 0      | 0       | 0      |
| 2  | 0       | 0      | 0.29    | 0      | 0.35    | 0      |
| 3  | 0       | 0      | 0.4     | 92     | 0.46    | 276    |
| 4  | 0       | 0      | 0.8     | 401    | 0.86    | 566    |
| 5  | 0       | 0      | 1       | 530    | 1.1     | 700    |
| 6  | 0       | 0      | 1.5     | 803    | 1.7     | 1112   |
| 7  | 0       | 0      | 2       | 1032   | 2.3     | 1500   |
| 8  | 0       | 0      | 2.5     | 1253   | 2.5     | 1648   |
| 9  | 0       | 0      | 3       | 1505   | 2.9     | 1875   |
| 10 | 0       | 0      | 3.5     | 1824   | 3.25    | 2007   |
| 11 | 0       | 0      | 4       | 2247   | 3.9     | 2531   |
| 12 | 0       | 0      | 4.5     | 2813   | 4.3     | 2800   |
| 13 | 0       | 0      | 5       | 3557   | 4.6     | 3000   |
| 14 | 0       | 0      | 5.5     | 3557   | 5       | 3000   |
| 15 | 0       | 0      | 5.5     | 3557   | 6       | 3000   |

**Illustration 4** *Configuring tables in WinControl XL Plus*

**Room setpoint at thermostat**

The primary control logic in this model has a heating and cooling set point defined in the software as variables. For applications requiring room set point control at the thermostat, follow these steps:

Define an Input as the base set point - such as IN2 using the STE-5000 Series Room thermostat. Enter Table 1 in the Range column of the Input window. Set up the Table as shown in Table 3 and Table 4. Table 1 will convert the voltage from the dial into degrees Fahrenheit.



**Illustration 5 Input option**

TABLE 3 STE 5000 Conversion Table

|   | Value |      |
|---|-------|------|
| 1 | 0.00  | 86.0 |
| 2 | 0.710 | 85.0 |
| 3 | 1.07  | 80.0 |
| 4 | 1.38  | 75.0 |
| 5 | 1.67  | 70.0 |
| 6 | 1.91  | 65.0 |
| 7 | 2.10  | 60.0 |
| 8 | 2.23  | 56.0 |

TABLE 4 STE 5000 Conversion Table

|   | Value |      |
|---|-------|------|
| 1 | 0.00  | 54.0 |
| 2 | 0.540 | 58.0 |
| 3 | 1.27  | 66.0 |
| 4 | 1.56  | 70.0 |
| 5 | 1.80  | 74.0 |
| 6 | 2.01  | 78.0 |
| 7 | 2.50  | 90.0 |
| 8 | 3.00  | 90.0 |

Add programming in one of the unused programs to set the Heat and Cool set points. The following lines will set the Heating set point to 2° below the set point shown on the thermostat dial and the Cooling set point to 2° above the thermostat dial setting.

```
10 VAR15 = IN2 + 2
20 VAR16 = IN2 - 2
```

**Using the NetSensor**

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7002, change the PID controllers 1 and 2 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor.

**Other options**

One of the strongest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the KMC supplied sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated.

**WinControl settings** The following tables display the WinControl XL Plus settings for this application note.

**Table 5 Input configuration**

|   | Description        | Manual | Input      | Units        | Average | Label    |
|---|--------------------|--------|------------|--------------|---------|----------|
| 1 | ROOM TEMPERATURE   | auto   | 74.0 deg.F | K10K-40->250 | 64      | ROOMTEMP |
| 2 |                    | auto   | 0.00       | Unused       |         |          |
| 3 | FLOW SENSOR DUCT 2 | auto   | 1569       | Table 2      | 64      | FLOSENS2 |
| 4 | FLOW SENSOR DUCT 1 | auto   | 1260       | Table 2      | 64      | FLOSENS1 |

**Table 6 Output configuration**

| Description | Manual         | Output | Units      | 0 %       | 100 | Delay | Security | Label  |
|-------------|----------------|--------|------------|-----------|-----|-------|----------|--------|
| 1           | auto           | Off    | Off / On   |           |     | 0:00  | 0        |        |
| 2           | auto           | Off    | Off / On   |           |     | 0:00  | 0        |        |
| 3           | DAMPER MOTOR 2 | auto   | 0.00 Volts | 0.0 -> 10 | 0.0 | 10    | 0        | MOTOR2 |
| 4           | DAMPER MOTOR 1 | auto   | 10.0 Volts | 0.0 -> 10 | 0.0 | 10    | 0        | MOTOR1 |

**Table 7 Variables**

| Description                        | Manual | Value   | Units     | Label    |
|------------------------------------|--------|---------|-----------|----------|
| 1-14 (Available for user programs) |        |         |           |          |
| 15 COOLING STAT SETPT              | auto   | 74.00   | deg.F     | COOL/SP  |
| 16 HEATING STAT SETPT              | auto   | 68.00   | deg.F     | HEAT/SP  |
| 17 MODE BOX 1                      | auto   | Cool    | Cool/Heat | HEAT-1   |
| 18 CLOCKWISE TO CLOSE1             | auto   | Yes     | No / Yes  | CWCLOSE1 |
| 19 MINIMUM FLOW BOX 1              | auto   | 500.00  |           | MIN1     |
| 20 MAXIMUM FLOW BOX 1              | auto   | 2000.00 |           | MAX1     |
| 21 VOLUME CONVERSION 1             | auto   | 1.00    |           | VOLCONV1 |
| 22 SENSOR CORRECTION 1             | auto   | 1.00    |           |          |
| 23 ACTUAL FLOW BOX 1               | auto   | 1260.41 |           | ACTFLOW1 |
| 24 REQUESTED FLOW BOX 1            | auto   | 500.00  |           | RQSTFLO1 |
| 25 MODE BOX 2                      | auto   | Heat    | Cool/Heat | HEAT-2   |
| 26 CLOCKWISE TO CLOSE2             | auto   | Yes     | No / Yes  | CWCLOSE2 |
| 27 MINIMUM FLOW BOX 2              | auto   | 500.00  |           | MIN2     |
| 28 MAXIMUM FLOW BOX 2              | auto   | 2000.00 |           | MAX2     |
| 29 VOLUME CONVERSION 2             | auto   | 1.00    |           | VOLCONV2 |
| 30 SENSOR CORRECTION 2             | auto   | 1.00    |           |          |
| 31 ACTUAL FLOW BOX 2               | auto   | 1569.26 |           | ACTFLOW2 |
| 32 REQUESTED FLOW BOX 2            | auto   | 2000.00 |           | RQSTFLO2 |

**Table 8 Controllers (PID control loops)**

| Controlled point | In Value   | Manual | Output  | Setpoint | Value | Action | Prop | Reset | H/M | Rate | Bias |
|------------------|------------|--------|---------|----------|-------|--------|------|-------|-----|------|------|
| 1 ROOMTEMP       | 0.00 deg.F | auto   | 0.0 %   | COOL/SP  | 74.0  | +      | 2    | 0     | H   | 0.00 | 0    |
| 2 ROOMTEMP       | 0.00 deg.F | auto   | 100.0 % | HEAT/SP  | 68.0  | -      | 2    | 0     | H   | 0.00 | 0    |
| 3 ACTFLOW1       | 0.00       | auto   | 100.0 % | RQSTFLO1 | 500.0 | +      | 200  | 0     | H   | 0.00 | 50   |
| 4 ACTFLOW2       | 0.00       | auto   | 0.0 %   | RQSTFLO2 | 2000  | +      | 200  | 0     | H   | 0.00 | 50   |

**Table 9 Tables**

|    | TABLE 1 | TABLE 2 | TABLE 3 |       |       |       |
|----|---------|---------|---------|-------|-------|-------|
|    | Value   | Value   | Value   |       |       |       |
| 1  | 0.00    | 0.00    | 0.00    | 0.00  | 0.00  | 0.00  |
| 2  | 0.00    | 0.00    | 0.140   | 0.00  | 0.350 | 0.00  |
| 3  | 0.00    | 0.00    | 0.200   | 75.0  | 0.460 | 276.0 |
| 4  | 0.00    | 0.00    | 0.400   | 327.0 | 0.860 | 566.0 |
| 5  | 0.00    | 0.00    | 0.800   | 563.0 | 1.10  | 700.0 |
| 6  | 0.00    | 0.00    | 1.00    | 664.0 | 1.70  | 1112  |
| 7  | 0.00    | 0.00    | 1.50    | 904.0 | 2.30  | 1500  |
| 8  | 0.00    | 0.00    | 2.00    | 1143  | 2.50  | 1648  |
| 9  | 0.00    | 0.00    | 2.50    | 1425  | 2.90  | 1875  |
| 10 | 0.00    | 0.00    | 3.00    | 1707  | 3.25  | 2087  |
| 11 | 0.00    | 0.00    | 3.50    | 2051  | 3.90  | 2531  |
| 12 | 0.00    | 0.00    | 4.00    | 2394  | 4.30  | 2800  |
| 13 | 0.00    | 0.00    | 4.50    | 2816  | 4.60  | 3000  |
| 14 | 0.00    | 0.00    | 5.00    | 3237  | 5.00  | 3000  |
| 15 | 0.00    | 0.00    | 5.50    | 3237  | 6.00  | 3000  |

**Table 10 Control Basic program list**

| Description          | On  | Manual | Exit | Label |
|----------------------|-----|--------|------|-------|
| 1 SINGLE DUCT VAV    | Yes | auto   | No   |       |
| 2 SLAVE DUCT CONTROL | Yes | auto   | No   |       |
| 3                    | No  | auto   | No   |       |
| 4                    | No  | auto   | No   |       |
| 5 USER PROGRAMS      | No  | auto   | No   |       |

**Program 1- SINGLE DUCT VAV**

```

10 IF HEAT-1 THEN A = CON2 / 100 ELSE A = CON1 / 100
20 RQSTFLO1 = ( ( MAX1 - MIN1 ) * A + MIN1 ) / VOLCONV1
30 ACTFLOW1 = FLOSENS1 * VAR22 : REM ACTUAL FLOW CORRECTED (FPM)
40 IF CON3 > 70 AND CWCLOSE1 THEN START B
50 IF CON3 > 70 AND NOT CWCLOSE1 THEN START C
60 IF CON3 < 30 AND CWCLOSE1 THEN START C
70 IF CON3 < 30 AND NOT CWCLOSE1 THEN START B
80 IF CON3 > 35 AND CON3 < 65 THEN STOP B , STOP C
90 IF+ B THEN STOP C
100 IF+ C THEN STOP B
110 IF CON3 < 99 AND CON3 > 1 THEN 120 ELSE 190
120 IF CON3 < 36 OR CON3 > 64 THEN START D ELSE STOP D
130 IF TIME-ON( D ) > 0:04:00 THEN 190
140 X = 1 / SCANS
150 Y = X + Y
160 IF Y > 5 THEN Y = 0
170 Z = ABS( CON3 - 50 ) / 50
180 IF Y > Z THEN STOP B , STOP C
190 IF B THEN MOTOR1 = 10
200 IF C THEN MOTOR1 = 0
210 IF NOT B AND NOT C THEN MOTOR1 = 5
220 IF TIME-ON( B ) > 0:06:00 OR TIME-ON( C ) > 0:06:00 THEN
MOTOR1 = 5
230 END

```

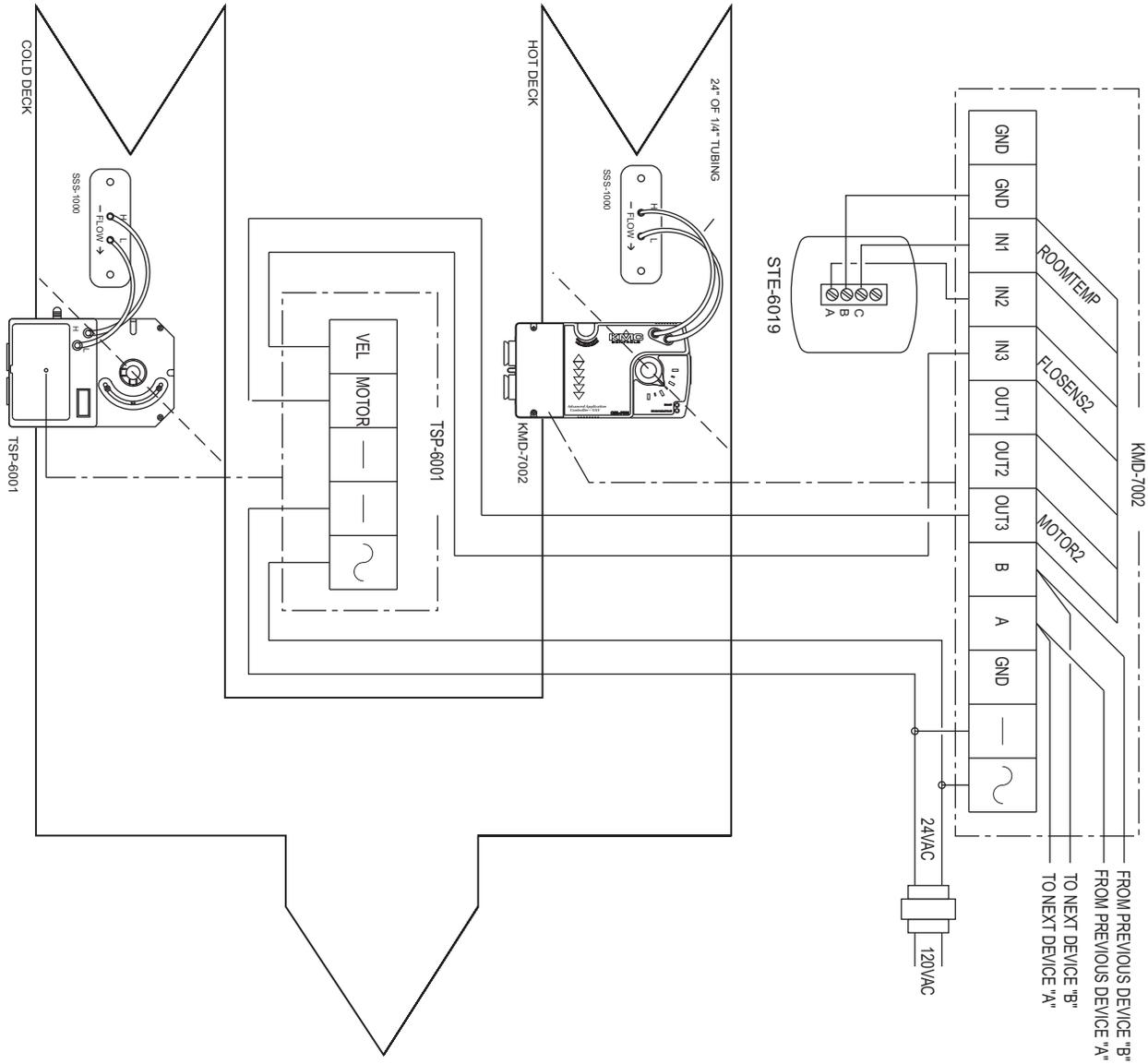
**Program 2- SLAVE DUCT CONTROL**

```

10 IF HEAT-2 THEN A = CON2 / 100 ELSE A = CON1 / 100
20 RQSTFLO2 = ( ( MAX2 - MIN2 ) * A + MIN2 ) / VOLCONV2
30 ACTFLOW2 = FLOSENS2 * VAR30 : REM ACTUAL FLOW CORRECTED (FPM)
40 IF CON4 > 70 AND CWCLOSE2 THEN START B
50 IF CON4 > 70 AND NOT CWCLOSE2 THEN START C
60 IF CON4 < 30 AND CWCLOSE2 THEN START C
70 IF CON4 < 30 AND NOT CWCLOSE2 THEN START B
80 IF CON4 > 35 AND CON4 < 65 THEN STOP B , STOP C
90 IF+ B THEN STOP C
100 IF+ C THEN STOP B
110 IF CON4 < 99 AND CON4 > 1 THEN 120 ELSE 190
120 IF CON4 < 36 OR CON4 > 64 THEN START D ELSE STOP D
130 IF TIME-ON( D ) > 0:04:00 THEN 190
140 X = 1 / SCANS
150 Y = X + Y
160 IF Y > 5 THEN Y = 0
170 Z = ABS( CON4 - 50 ) / 50
180 IF Y > Z THEN STOP B , STOP C
190 IF B THEN MOTOR2 = 5
200 IF C THEN MOTOR2 = 0
210 IF NOT B AND NOT C THEN MOTOR2 = 2.25
220 IF TIME-ON( B ) > 0:06:00 OR TIME-ON( C ) > 0:06:00 THEN
MOTOR2 = 2.25
230 END

```

**Application drawings**



**Illustration 6 KMD-7002 dual-duct application**

## Application Note—AN0905A

---

Revision A

### **KMD-7003 and KMD-7053 VAV Controller**

This application note describes using a KMD-7003 and KMD-7053 for single-duct VAV installations. It covers using the controllers for heat/cool change over and proportional reheat.

---

#### **Related materials**

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ Installation and operation guides for KMD-7003 and STE-5013
- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide
- ◆ System plans with controller addresses
- ◆ AN0805A, The NetSensor Balancer Program

#### **Introduction**

The KMD-7003 is a KMC digital controller and an actuator built into one unit. A pressure-independent factory programmed VAV sequence automatically modulates flow into the conditioned space between Minimum and Maximum setpoints as defined by the user. This sequence allows for Heating, Cooling, or changeover applications with minimal user initialization. Additionally, user programming can customize each application as required.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory configurations allowing you to program any desired sequence.

The KMD-7003 VAV Terminal Controller contains the basic programming sequence required for two specific applications:

- ◆ Single Duct VAV (heating-cooling-changeover)
- ◆ Single Duct VAV with time proportional reheat

In addition, there is a balance program for setting up VAV box parameters. Initially these devices will default to a standard Single Duct Cooling VAV configuration. You can use either of the Reheat functions by turning "ON" the respective programs and enabling a variable on the Variables screen. The following paragraphs provide specific details and explanations to help you use the KMD-7003.

## Velocity sensor

The KMD-7003 incorporates an on-board flow-through type sensor to determine duct airflow. Airflow through the sensor is approximately 1.12 SCFH (32 liters/hour) at .25 INWC (62.5 Pa) pressure drop. The sensor uses two RTD's (resistance temperature detectors) to measure flow. The RTD's are platinum films on a ceramic base chosen for long term stability and accuracy. The smaller 100 ohm RTD is self-heated to a constant temperature differential above ambient. The amount of power required varies with the specific airflow. A larger 1000-ohm RTD constantly senses ambient temperature and compensates. The sensor housing channels the airflow over the self-heated RTD in a manner to isolate its heat from the compensator. This minimizes shifts at low flows due to various mounting positions. The sensor housing itself is molded from plastic that carries the UL94-5V rating. The sensor ports are 0.188 inch diameter and readily accepts 0.25 inch O.D. "FR" tubing (.040 wall) or equivalent.

## Velocity limits

Minimum and Maximum Flow velocity limits can be defined in either of two ways: Through KMC digital operating software or with the NetSensor. Setting velocity limits through the KMC digital software is as simple as defining the value of a variable. Using this method the limits may be entered in any units desired (CFM, L/s, Pa, etc.) as long as a "conversion factor" is also provided. The "default" settings are always in Feet Per Minute (FPM). For details on using this method refer to the following section on standard VAV operation.

## Damper actuator

The actuator is capable of supplying a minimum of 50 inch pounds of torque to open or close the damper according to instructions from the KMD control sequence. The motor is an AC synchronous type for smooth, reliable operation and is coupled with a magnetic slip-clutch design that allows continuously stalling damper rotation without damage. A time-out function will stop the motor anytime the unit has run continuously in one direction for longer than 6 minutes. This ensures that the stops have been reached without needlessly running the motor.

Physical adjustable end-stops allow limiting rotation to less than the 90° rotation and a gear disengagement mechanism on the side of the case can be used for fast and simple installation and set-up. The red disengagement button is located on the left side of the actuator.

## Standard Variable Air Volume (VAV) Operation

The standard VAV sequence of operation has been pre-programmed into each KMD-7003 controller. Default operation is defined as:

- ◆ Single Duct Cooling
- ◆ Clockwise to close damper rotation
- ◆ Minimum and Maximum Limits at 500 and 2000 FPM respectively
- ◆ Cooling setpoint at 74°F
- ◆ Space Temperature Sensor connected to IN1

These units are capable of independent, stand-alone operation or they may be networked together with other KMC digital products. If stand-alone operation is desired, supplying 24 VAC power and connecting a 10K thermistor (STE-series) to

IN1 will enable full operation to the default parameters listed above. If these units will be networked together it may be necessary to configure each controller with a unique address and other set-up options using HCM.

The following is a simplified explanation of the control logic used in the standard VAV application:

- ◆ A full-function PID controller (CON1) is set to perform the thermostat function by comparing the space temperature (IN1) to the set point temperature (VAR25). In the COOL (default) changeover mode, thermostat action is Direct Acting and "Active" Setpoint is the Cooling setpoint temperature VAR30). Conversely, if HEAT changeover mode was selected, thermostat action would change to Reverse Acting and the "Active" set point would be the Heating Set Point (VAR29).
- ◆ A "requested" flow set point is calculated based on the thermostat information and Minimum / Maximum flow limits. Using this "Requested" flow a second full function PID Controller (CON2) looks at actual flow (IN4) as measured by the velocity sensor and instructs the damper to open or close accordingly.
- ◆ Parameters of either the Thermostat (CON1) or the Flow Controller (CON2) such as Proportional Band, Reset (Integral), or Rate (Derivative) functions can easily be changed to suit application requirements. The "BIAS" of 50% preset on CON2 should not be changed as this defines a satisfied "Neutral" position for the actuator drive circuit.

## Hardware Configuration Manager (HCM)

For networked operation, use the HCM software to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 11 Default Configuration for KMD-7003**

|                   |       |
|-------------------|-------|
| Panel Number      | 124   |
| Network Baud Rate | 38400 |
| Last Panel        | No    |
| Panel Type        | 116   |

## Setting up with WinControl XL

This section covers the set-up requirements to activate any of the pre-programmed options available in this device. At this point, communications should have been established with the unit.

A "Master" password is available to "sign-on" to each controller. Other security codes can be entered to control access to the system.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence.

The first step after signing on to the controller is “Load Descriptors One” or “Load Descriptors All”. Many of the descriptions have been pre-defined for Inputs, Outputs, and Variables to aid in configuring the options.

To use either the Time-Proportional Reheat (OUT1) program or the 3-stage Reheat (OUTS 1, 2, 3) program, go to the “Control Basic” programs screen and turn “ON” the appropriate program.

After the appropriate programs are enabled you can use the setup screen in Control Group #2 (GRP2) which lists the items most likely to be monitored. The following section outlines each of the parameters that have been pre-programmed into the device.

**Configuring inputs**

Inputs 1, 2, and 3 can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:

| # | Description      | Manual                              | Value  | Units            | Decom.                              | Label    |
|---|------------------|-------------------------------------|--------|------------------|-------------------------------------|----------|
| 1 | ROOM TEMPERATURE | <input type="checkbox"/>            | 72.00  | Deg. F           | <input checked="" type="checkbox"/> | ROOMTEMP |
| 2 |                  | <input type="checkbox"/>            | 0.00   | Unused           | <input type="checkbox"/>            |          |
| 3 |                  | <input checked="" type="checkbox"/> | 0.00   | Unused           | <input type="checkbox"/>            |          |
| 4 | AIR FLOW SENSOR  | <input type="checkbox"/>            | 250.00 | Unused (Table 2) | <input checked="" type="checkbox"/> | FLOWSNSR |

**Illustration 7 Input configuration in WinControl XL**

**Input # 1–ROOM TEMPERATURE** Default selects a KMC 10K ohm thermistor as used in all of KMC’s STE series sensors.

**Input #2** Unused

**Input #3** Unused

**Input #4–Air Flow Sensor** The on-board flow-sensor. Table 2 has been configured to convert the voltage input into feet per minute. This input is not accessible for any other use.

**Configuring outputs**

The KMD-7003 VAV Controller features three user available outputs and an analog output to drive the damper motor. The KMC factory supplied control routines have been written to use the following output configuration:

| # | Description        | Manual                              | Value | Units      | Label |
|---|--------------------|-------------------------------------|-------|------------|-------|
| 1 | STAGE 1            | <input type="checkbox"/>            |       | Off/Off/On |       |
| 2 | STAGE 2            | <input checked="" type="checkbox"/> |       | Off/Off/On |       |
| 3 | STAGE 3            | <input type="checkbox"/>            |       | Off/Off/On |       |
| 4 | DAMPER MOTOR DRIVE | <input type="checkbox"/>            | 0.00  | Volts      | MOTOR |

**Illustration 8 Output configuration in WinControl XL**

**Output #1–Stage 1** The factory program defines Output 1 as the first stage of heating on VAV boxes with electric reheat (PRG3). Optionally, it can be used to control any 0-10 VDC device by adding your own custom programming in program area 5 (PRG5).

**Output #2–Stage 2** The factory program defines Output 2 as the second stage of heating on VAV boxes with electric reheat (PRG3). This is a triac that can be programmed to switch valve actuators or 24-volt reheat circuits. Optionally, it can be used to control any 24-volt circuit by adding your own custom programming in program area 5 (PRG5).

**Output #3–Stage 3** The factory program defines Output 3 as the third stage of heating on VAV boxes with electric reheat (PRG3). Optionally, it can be used to control any 24-volt circuit by adding your own custom programming in program area 5 (PRG5).

**Output #4–DAMPER MOTOR DRIVE** This output is used to control the damper position. It is not accessible for any other use on the KMD-7003. When the output is commanded to 0 volts the motor drives the damper counterclockwise, 10 volts drives the damper clockwise, and 5 volts stops the damper.

**! Caution**

When connecting loads to the triac (Output 2) or relay (Output 3), use only the terminal marked RET associated with the triac or relay for 24-volt circuits.

**Configuring variables**

A total of thirty-two system variables are available to perform various functions such as set points, limits, and modes. All variables can be changed or modified to suit the specific application. The following have been pre-defined for use in the KMD-7003 factory supplied control sequences:

**Variable #17–CHANGEOVER MODE (COOL/HEAT)** Chooses whether the unit will operate as a heating or cooling box by automatically setting the thermostat action as direct (COOL) or reverse (HEAT).

**Variable #18–CLOCKWISE TO CLOSE? (NO/YES)** Defines which way the damper will rotate to increase or decrease air-flow as required. “NO” will close the box in a counter-clockwise direction.

**Variable #19–MINIMUM FLOW** Sets the lowest design flow for the VAV box. The default setting is for a Feet-Per-Minute (FPM) value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable #20–MAXIMUM FLOW** Sets the highest design flow for the VAV box. The default setting is for a FPM value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable #21–VOLUME CONVERSION** Default value is 1.00 (no conversion). This factor allows all flow settings and live readings to be in any desired units. Since default calculations are in FPM, the duct area is required (in square feet) to get readings in CFM. Similarly, values can be converted to L/s, Pa, etc. with the proper factors applied.

**Variable #22–SENSOR CORRECTION** Default value is 1.00 (no correction required). This factor is provided to allow simple correction for erroneous sensor readings. This factor is multiplied against all flow readings and the product is stored as the ACTUAL FLOW (VAR23).

**Variable #23–ACTUAL FLOW** This value is calculated from the flow sensor input (IN4) and the sensor correction variable (VAR22) using the following formula:

$$\begin{aligned} \text{ACT-FLOW} &= \text{FLOWSNSR} * \text{VAR22} \\ \text{Or:} \\ \text{VAR23} &= \text{IN4} * \text{VAR22} \end{aligned}$$

Units for this Variable must always be in Feet per Minute for proper operation of the built-in control sequences. For flow displayed in desired units see Variable 26.

**Variable #24–REQUESTED FLOW** This is the flow set point based on the heating or cooling needs of the space. The value is set automatically in Program 1 (PRG1). Units for this Variable must always be in Feet per Minute for proper operation of the built-in control sequences. For flow displayed in desired units see Variable 26.

**Variable #25–ACTIVE TEMP SETPOINT** This is the temperature the controller will attempt to maintain in the space. The active set point is the heating set point (VAR29) in HEAT MODE (VAR17 = ON) and the cooling set point (VAR30) in COOL MODE (VAR17 = OFF)

**Variable #26–DUCT VOLUME READOUT** This variable has been set up to allow easy access to a live velocity read-out. Units can be converted using the Volume Conversion factor (VAR21) to relate to CFM, L/s, Pa, etc. as needed. The following formula is used in Program 1 (PRG1) to obtain this value:

$$\begin{aligned} \text{VOLUME1} &= \text{ACT-FLOW} * \text{VOLFACTR} \\ \text{Or:} \\ \text{VAR26} &= \text{VAR23} * \text{VAR21} \end{aligned}$$

**Variable #27–REHEAT STATUS** This flag will enable or disable the REHEAT function. If reheat is disabled, all outputs being used in that reheat function are overridden to “OFF”. To choose one of the pre-programmed Reheat options, the appropriate program must be turned on. See the Program Menu for options. If neither reheat program is turned on, this variable has no purpose.



#### Note

A common application is to use this flag to disable the Reheat function during the summer mode.

**Variable #28–AUXILIARY FLOW** Assigning a flow set point other than 0 in this variable allows the VAV set point to be overridden when STAGE 1 (OUT1) heat is turned “ON”. The auxiliary flow set point is only enabled if STAGE HEATING (PRG3) is used, the REHEAT STATUS (VAR27) is enabled, and the value is non-zero. To disable the auxiliary flow, leave the value at 0 (zero).

**Variable #29–HEAT TEMP SETPOINT** This is the active set point used by CON1 when in the heating mode (VAR17 = ON). Units must match that of the temperature sensor on IN1 (deg. F or deg. C)

**Variable #30–COOL TEMP SETPOINT** This is the active set point used by CON1 when in the cooling mode (VAR17 = OFF). Units must match that of the temperature sensor on IN1 (deg. F or deg. C)

**Variable #31–DAMPER DRIVE TIME** This is the approximate time in seconds required for the damper to move from the fully closed to the fully open position or from the fully open to the fully closed position. It is used in program 1 to roughly determine the damper position in variable 32 (VAR32).

Approximate travel time is 300 seconds for 90 degrees, 200 seconds for 60 degrees and 150 seconds for 45 degrees. The end stops on the back side of the KMD-7003 need to be placed in their appropriate places to get the desired rotation.

**Variable #32–DAMPER POSITION** This is a calculated position based on the full stroke drive time (VAR31). Variable 31 must be set correctly for the damper position information to work properly.

Damper position is determined in Program 1 by measuring the time the damper has been commanded open or closed versus the time needed for complete travel. If the damper is stuck this variable will not necessarily indicate the correct damper position. It will however indicate the controllers attempt to move the damper. This variable is used only to give users an indication of what the controller has done and is doing to control the position of the damper. It is not used as feedback in the control sequence.

#### Configuring PID control loops

KMC digital products feature built-in, full function PID loop controllers. All parameters may be modified as necessary.

**Controller #1 (CON1)** is a direct-acting thermostat. It compares ROOM TEMPERATURE (IN1) to the ACTIVE TEMP SETPT (VAR25) to determine the VAV flow requirement.

**Controller #2 (CON2)** is set up as a direct-acting thermostat. It compares ACTUAL FLOW (VAR23) to the REQUESTED FLOW (VAR24) to operate the DAMPER (OUT4). All values used by the controller have been converted to FPM to allow a common proportional band value regardless of the units selected. Default is a 200 FPM proportional band which relates to approximately 60 FPM “dead band” to stabilize actuator operation. Default parameters do not include any integral or derivative function.



---

**Note**

The desired air flow is VAR24 and is automatically calculated by the VAV program. For example, to override the airflow to some other value under specific conditions, include a statement such as:

```
10 IF (condition) THEN VAR24 = (value)
```

All values at this point must be in FPM. Condition may be a variable that represents an abnormal event such as a smoke zone activation. To ensure close-off on a box, set the desired flow (value) to a number below zero such as -50. To ensure a wide open damper position use a flow rate higher than 3000 FPM (The highest value the on-board sensor can read).

**Controller #3 (CON3)** compares the ROOM TEMPERATURE (IN1) to the HEATING SET POINT (VAR29) for operation of the reheat stages when programs 2 or 3 are enabled.

**Control Basic programming**

The first program handles set points and flow control. There are two programs provided to control reheat in the VAV box. If neither sequence fits the required application, edit the programs or send blank programs to the controller and replace any of the factory programs. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.



---

**Caution**

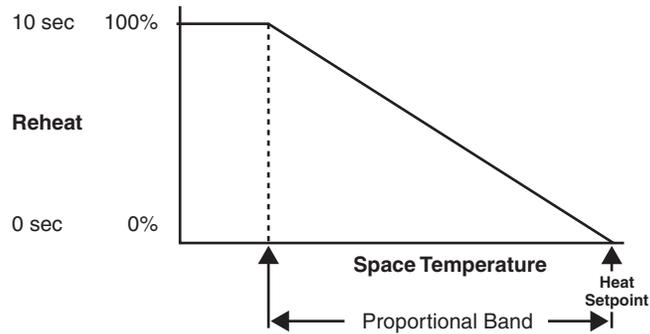
This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type).

**Program 1–Single Duct VAV** This program sets the ACTIVESP (VAR25) based on the CHANGEOVER MODE (VAR17). When the CHANGEOVER MODE is OFF the ACTIVE SET POINT equals the Heating Set Point (VAR29). When the CHANGEOVER MODE is ON the ACTIVE SET POINT equals the Cooling Set Point (VAR30). The necessary Air flow is calculated and the damper motor is modulated to maintain this volume.

See [Program 1– SINGLE DUCT VAV](#) on page 3-61 for a complete listing of the program.

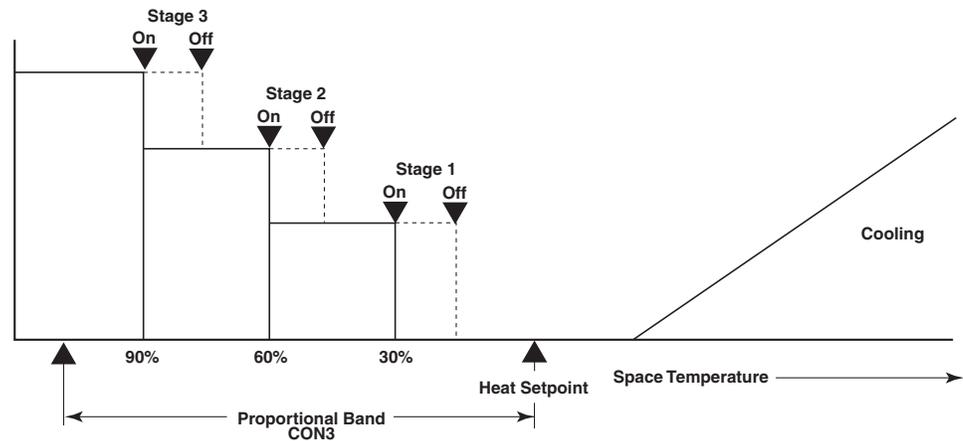
**Program Two–TIME PROPORTIONAL REHEAT** See [Program 2–TIME PROPORTIONAL HT](#) on page 3-62 for a complete listing. As supplied from the factory, this uses only one digital output (OUT1) by controlling the percentage of

“ON” time over some specific time interval. For the pre-programmed sequence a fixed 10-second time base is used; A call for 73% heating would turn “ON” the output for 7.3 seconds and turn “OFF” for the remaining 2.7 seconds. See Illustration 10. Using this method of reheat requires only one output. Outputs 2 and 3 are left for other output devices. Illustration 9 shows a graphical representation of Program 1.



**Illustration 9 Proportional Reheat**

**Program 3—Three-stage Reheat** As supplied from the factory, this program sequentially steps each stage of reheat as the requirement for heating increases (using Outputs 1, 2, and 3). Each stage turns “ON” at a fixed point (30%, 60%, and 90%) of the heating thermostat (CON3) value. The controller changes 0-100% over the proportional band range. Illustration 10 shows the sequence graphically.



**Illustration 10 Three-stage reheat**

This program may also be used for FAN-BOX applications. Connect the FAN relay to STAGE 1 (OUT1) and 2 stages of reheat to outputs 2 and 3. A differential is built into the sequence to prevent cycling the outputs over small temperature variations.

An optional AUXILIARY FLOW limit has been pre-programmed for use with the 3-stage Reheat sequence. This feature allows increasing the VAV flow set point to some value other than Minimum Flow anytime Stage 1 of heating is enabled. The value of Auxiliary flow entered in VAR28 must be in the same units as Minimum

and Maximum flow since the same “conversion factor” (VAR21) is used by the programming. If this feature is NOT being used, leave the Auxiliary Flow (VAR28) value at zero (0) and the program will ignore it.

The programming has been written to instruct the unit to control flow at the Auxiliary Flow limit whenever conditions call for stage 1 heating. If it is desired to initiate the Auxiliary Flow before actually turning “ON” the first stage of heat, add “Delay” time in the appropriate column of OUT1 on the Outputs screen (up to 99 min. 50 sec.). The delay will take precedence over all other software actions.

See [Program 3 3–STAGE REHEAT](#) on page 3-62 for a complete listing

**Program 4–Balancer override** This program is used with the NetSensor to force the controller to maintain either a minimum or a maximum air flow setpoint. Details on using the NetSensor can be found in applications note *AN0805A, The NetSensor Balancer Program*.

### **Program 5–Available for user programming**

#### **Configuring tables**

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a

potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7003:

**Table #1** Ready for user programming.

**Table #2** Correlates velocity pressure from the on-board flow sensor to feet per minute.

**Table #3** Correlates velocity pressure of CSP-5001 VAV controller to feet per minute.



---

#### **Note**

These tables use nominal values for correlation. Due to component tolerances and the resistance of field wiring, these values may need correction on an individual basis. If the value read is too low, enter a positive calibration in the input column labeled “cal”. If the value read is high, a negative calibration can be entered. If necessary, reconfigure Table #2 and/or Table #3. The complete Table #2 and Table #3 from the factory are shown in figure 3. If the tables have been reconfigured and the user wants the factory configuration for Table #2 and Table #3 it will be necessary to use Clear Panel or re-enter the values.

**!** **Caution**

Using Clear Panel will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type.

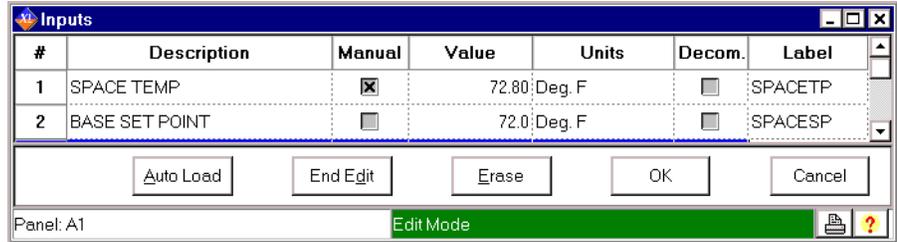
| #  | Table 1 | Unused | Table 2 | Unused | Table 3 |
|----|---------|--------|---------|--------|---------|
| 1  | 0       | 0      | 0       | 0      | 0       |
| 2  | 0       | 0      | 0.29    | 0      | 0.35    |
| 3  | 0       | 0      | 0.4     | 92     | 0.46    |
| 4  | 0       | 0      | 0.8     | 401    | 0.86    |
| 5  | 0       | 0      | 1       | 530    | 1.1     |
| 6  | 0       | 0      | 1.5     | 803    | 1.7     |
| 7  | 0       | 0      | 2       | 1032   | 2.3     |
| 8  | 0       | 0      | 2.5     | 1253   | 2.5     |
| 9  | 0       | 0      | 3       | 1505   | 2.9     |
| 10 | 0       | 0      | 3.5     | 1824   | 3.25    |

**Illustration 11 Table configuration in WinControl XL**

The primary control logic in this model has a heating and cooling setpoint defined in the software as variables. For applications requiring room set point control at the thermostat, follow these steps:

1. Define an Input as the base set point—such as IN2 using the STE-5000 Series Room thermostat. Enter Table 1 in the Range column of the Input window.

- Set up the table as shown in Table 12 or Table 13. The WinControl Table 1 will convert the voltage from the dial into degrees Fahrenheit.



**Illustration 12** Alternate input configuration in WinControl XL

**Table 12** STE-5200 Conversion Table

|   | Value |      |
|---|-------|------|
| 1 | 0.00  | 86.0 |
| 2 | 0.710 | 85.0 |
| 3 | 1.07  | 80.0 |
| 4 | 1.38  | 75.0 |
| 5 | 1.67  | 70.0 |
| 6 | 1.91  | 65.0 |
| 7 | 2.10  | 60.0 |
| 8 | 2.23  | 56.0 |

**Table 13** STE-5000 Conversion Table

|   | Value |      |
|---|-------|------|
| 1 | 0.00  | 54.0 |
| 2 | 0.540 | 58.0 |
| 3 | 1.27  | 66.0 |
| 4 | 1.56  | 70.0 |
| 5 | 1.80  | 74.0 |
| 6 | 2.01  | 78.0 |
| 7 | 2.50  | 90.0 |
| 8 | 3.00  | 90.0 |

- Add programming in program area #5 to set the Heat and Cool set points. The following lines will set the Heating set point to 2° below the set point shown on the thermostat dial and the Cooling set point to 2° above the thermostat dial setting.

*Example:*

```

10 VAR29 = IN2 - 2
20 VAR30 = IN2 + 2
    
```

## Adding fan control

The following topics modify Program 2 and Program 3 to include fan control.

### Modifications to Program 2

The following modification adds fan control to the supplied time-proportional reheat program.

- ◆ Change all references to *Stage 1* to *Stage 2*.
- ◆ Label Output 3 as FAN
- ◆ Replace Program 2 with the following Control Basic program:

*Example:*

```

10 REM ----- TIME PROPORTIONAL REHEAT SEQUENCE
-----
20 REM -- Designed to drive a solid state relay connected to
electric reheat.
30 REM Time base is fixed at 10 seconds - ie, a call for 50%
heat turns on
40 REM relay (OUT1) for 5 seconds, off for 5 seconds.
50 REM Uses CON#3 for control of reheat operation! ( Full PID
function )
60 IF NOT VAR27 THEN STOP OUT1 , END
70 K = 1 / SCANS
80 L = L + K : IF L > 10 THEN L = 0
90 IF L < CON3 / 10 THEN START OUT1 ELSE STOP OUT1
100 IF CON3 > 30 THEN START OUT3
110 IF CON3 < 10 THEN STOP OUT3
120 END

```

### Modifications to Program 3

The following modification changes the supplied three-stage reheat program to a two-stage reheat with fan control.

- ◆ Delete all references to *Stage 3*
- ◆ Label Output 3 as FAN
- ◆ In Program 3, starting at Line 60, replace all programs lines with the following lines of Control Basic:

*Example:*

```

60 IF NOT VAR27 THEN STOP OUT1 , STOP OUT2 , STOP OUT3 , STOP A
, END
70 IF CON3 > 15 THEN START OUT1 , START A
80 IF CON3 > 60 THEN START OUT2
90 IF CON3 < 40 THEN STOP OUT2
100 IF CON3 < 11 THEN STOP OUT1 , STOP A
110 IF A THEN VAR24 = VAR28 / VAR21
120 IF OUT1 OR OUT2 THEN START OUT3
130 IF NOT OUT1 AND NOT OUT2 THEN STOP OUT3
140 IF A THEN VAR24 = VAR28 / VAR21
150 END

```

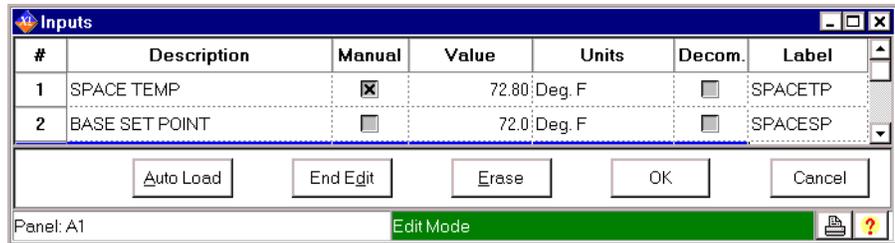
## Using the NetSensor

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7003, change the PID controllers #1 and #3 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor.

## Setback (unoccupied) setpoints

Many applications require “setback” or “setup” temperatures as a means of energy savings in unoccupied periods. To accomplish this, it is necessary to establish a condition of occupancy and define when and how these modes occur.

A variable can be set up as a “flag” to indicate the current mode of operation. The units “Un/Occupy” work well for this purpose.

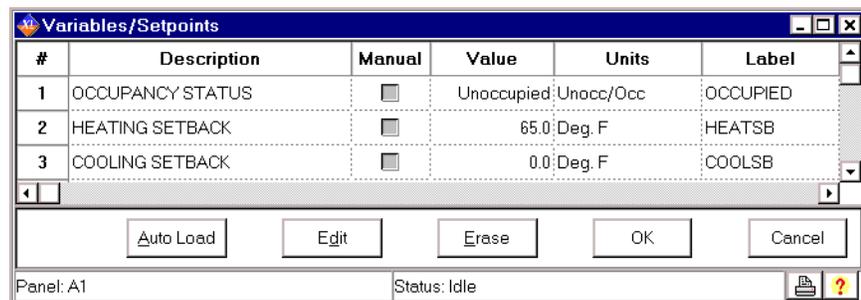


**Illustration 13** Input option for occupancy setpoints

Program 5 can contain the code needed to change the state of this Variable based on a weekly schedule as shown in this example:

```
10 IF WS1 THEN START OCCUPIED ELSE STOP OCCUPIED
```

The flag “OCCUPIED” (VAR1 in this example) is “ON” in the occupied mode and off in the unoccupied mode. Two more variables can be set up to define the heating and / or cooling set-back temperatures.



**Illustration 14** Alternate variables

Now, some simple programming can be written to determine the setpoint based on the occupancy mode:

```
10 IF OCCUPIED THEN HEAT-SP = SPACESP - 2 , COOL-SP = SPACESP +  
2  
20 IF NOT OCCUPIED THEN HEAT-SP = HEATSB , COOL-SP = COOLSB
```

When in the occupied mode, the heating set point is 2 degrees less than the set point on the thermostat dial and the cooling set point is 2 degrees above. In the unoccupied mode (line 20), the heating set point is set to whatever value is in VAR2 and the cooling set point is set equal to VAR3. It is not necessary to use variables for heating set-back or cooling set-up, but doing so makes modification of these values much easier than getting into the program to change them.

## Proportional reheat

A short line of code can be entered into Program area 5 to control hot water reheat. Since Controller #3 (CON3) is set up as a reverse acting thermostat, its value will change from 0 to 100% as temperature falls below the heating setpoint. Configure the output to a range of 0-100% then use:

```
40 OUT1 = CON3
```

## Other options

One of the biggest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the KMC supplied sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated. The WinControl XL User's Manual provides help and examples for programming Control Basic.

**WinControl settings** The following tables display the WinControl XL Plus settings for this application note.

**Table 14 Input configuration**

|   | Description      | Manual | Value      | Units        | Average | Label    |
|---|------------------|--------|------------|--------------|---------|----------|
| 1 | ROOM TEMPERATURE | auto   | 74.0 deg.F | K10K-40->250 | 64      | ROOMTEMP |
| 2 |                  | auto   | 0.00       | Unused       |         |          |
| 3 |                  | auto   | 0.00       | Unused       |         |          |
| 4 | AIR FLOW SENSOR  | auto   | 1115       | Table 2      | 64      | FLAWSNSR |

**Table 15 Output configuration**

|   | Description        | Manual | Output     | Units     | 0 % | 100 | Delay | Security | Label |
|---|--------------------|--------|------------|-----------|-----|-----|-------|----------|-------|
| 1 | STAGE 1            | auto   | Off        | Off / On  |     |     | 0:00  | 0        |       |
| 2 | STAGE 2            | auto   | Off        | Off / On  |     |     | 0:00  | 0        |       |
| 3 | STAGE 3            | auto   | Off        | Off / On  |     |     | 0:00  | 0        |       |
| 4 | DAMPER MOTOR DRIVE | auto   | 5.00 Volts | 0.0 -> 10 | 0.0 | 10  |       | 0        | MOTOR |

**Table 16 Variables**

|      | Description                   | Manual | Value    | Units      | Label     |
|------|-------------------------------|--------|----------|------------|-----------|
| 1-16 | (Available for user programs) |        |          |            |           |
| 17   | CHANGEOVER MODE               | auto   | auto     | Cool       | Cool/Heat |
| 18   | CLOCKWISE TO CLOSE ?          | auto   | auto     | Yes        | No / Yes  |
| 19   | MINIMUM FLOW                  | auto   | auto     | 500.00     | MIN-FLOW  |
| 20   | MAXIMUM FLOW                  | auto   | auto     | 2000.00    | MAX-FLOW  |
| 21   | VOLUME CONVERSION             | auto   | auto     | 1.00       | VOLFACTR  |
| 22   | SENSOR CORRECTION             | auto   | auto     | 1.00       |           |
| 23   | ACTUAL FLOW                   | auto   | auto     | 0.00       | ACT-FLOW  |
| 24   | REQUESTED FLOW                | auto   | auto     | 0.00       | RQSTFLOW  |
| 25   | ACTIVE TEMP SETPOINT          | auto   | auto     | 74.00      | ACTIVESP  |
| 26   | DUCT VOLUME READOUT           | auto   | 0.00     |            | VOLUME1   |
| 27   | REHEAT STATUS                 | auto   | Disabled | Dis/Enable | REHEAT    |
| 28   | AUXILLARY FLOW                | auto   | auto     |            | AUX-FLOW  |
| 29   | HEAT TEMP SETPOINT            | auto   | 68.00    | deg.F      | HEAT-SP   |
| 30   | COOL TEMP SETPOINT            | auto   | 74.00    | deg.F      | COOL-SP   |
| 31   | DAMPER DRIVE TIME             | auto   | 300.00   | Sec.       | DRV-TIME  |
| 32   | DAMPER POSITION               | auto   | 50.00    | %          | DMPR-POS  |

**Table 17 Controllers**

|   | In Label | In Value   | A/M  | Output | Setpoint | Value | Action | Prop | Reset | Rate | Bias |    |
|---|----------|------------|------|--------|----------|-------|--------|------|-------|------|------|----|
| 1 | ROOMTEMP | 0.00 deg.F | auto | 50.0 % | ACTIVESP | 74.0  | +      | 2    | 0     | H    | 0.00 | 50 |
| 2 | ACT-FLOW | 0.00       | auto | 0.0 %  | RQSTFLOW | 1250  | +      | 200  | 0     | H    | 0.00 | 50 |
| 3 | ROOMTEMP | 0.00 deg.F | auto | 0.0 %  | HEAT-SP  | 68.0  | -      | 2    | 0     | H    | 0.00 | 0  |
| 4 |          | 0.00       | auto | 0.0 %  |          | 0.00  | +      | 0    | 0     | H    | 0.00 | 0  |

**Table 18 Tables**

|    | TABLE 1 Value | TABLE 2 Value | TABLE 3 Value |
|----|---------------|---------------|---------------|
| 1  | 0.00          | 0.00          | 0.00          |
| 2  | 0.00          | 0.00          | 0.140         |
| 3  | 0.00          | 0.00          | 0.200         |
| 4  | 0.00          | 0.00          | 0.400         |
| 5  | 0.00          | 0.00          | 0.800         |
| 6  | 0.00          | 0.00          | 1.00          |
| 7  | 0.00          | 0.00          | 1.50          |
| 8  | 0.00          | 0.00          | 2.00          |
| 9  | 0.00          | 0.00          | 2.50          |
| 10 | 0.00          | 0.00          | 3.00          |
| 11 | 0.00          | 0.00          | 3.50          |
| 12 | 0.00          | 0.00          | 4.00          |
| 13 | 0.00          | 0.00          | 4.50          |
| 14 | 0.00          | 0.00          | 5.00          |
| 15 | 0.00          | 0.00          | 5.50          |

**Table 19 NET SENSOR #1**

| Button | Point                | Value | Type   | Range     | Display Only |
|--------|----------------------|-------|--------|-----------|--------------|
| 1      |                      | 0.1   | Analog | Deg.F     |              |
| 2      | ACTIVE TEMP SETPOINT | 0     | Analog | 0 Decimal | Can Change   |
| 3      | DUCT VOLUME READOUT  | 0     | Analog | 0 Decimal | Display Only |
| 4      | VOLUME CONVERSION    | 0.00  | Analog | 2 Decimal | Can Change   |
| 5      | MINIMUM FLOW         | 0     | Analog | 0 Decimal | Can Change   |
| 6      | SENSOR CORRECTION    | 0.00  | Analog | 2 Decimal | Can Change   |
| 7      | MAXIMUM FLOW         | 0     | Analog | 0 Decimal | Can Change   |
| Aux    | BALANCER override    | Off   | State  | Off/On    |              |

*Illustration 15 Control Basic program list*

|   | <b>Description</b>   | <b>Run</b> | <b>A/M</b> | <b>Exit</b> | <b>Label</b> |
|---|----------------------|------------|------------|-------------|--------------|
| 1 | SINGLE DUCT VAV      | Yes        | auto       | No          |              |
| 2 | TIME PROPORTIONAL HT | No         | auto       | No          |              |
| 3 | 3-STAGE REHEAT       | No         | auto       | No          |              |
| 4 | BALANCER override    | No         | auto       | No          |              |
| 5 | USER PROGRAMS        | No         | auto       | No          |              |

**Program 1– SINGLE DUCT VAV**

```

1 REM BALANCER OVRD
5 IF TIME-ON( BALANCER~override ) > 0:30:00 THEN STOP
BALANCER~override
7 IF BALANCER~override THEN END

10 IF CHANGEOVER~MODE THEN ACTIVE~TEMP~SETPOINT =
HEAT~TEMP~SETPOINT ELSE ACTIVE~TEMP~SETPOINT =
COOL~TEMP~SETPOINT
20 IF CHANGEOVER~MODE THEN A = 1 - CON1 / 100 ELSE A = CON1 /
100
30 REQUESTED~FLOW = ( ( MAXIMUM~FLOW - MINIMUM~FLOW~ ) * A +
MINIMUM~FLOW~ ) / VOLUME~CONVERSION
40 ACTUAL~FLOW~ = AIR~FLOW~SENSOR * SENSOR~CORRECTION : REM
ACTUAL FLOW CORRECTED
50 DUCT~VOLUME~READOUT = ACTUAL~FLOW~ * VOLUME~CONVERSION : REM
LIVE VOLUME

60 IF CON2 > 70 AND CLOCKWISE~TO~CLOSE~? THEN START B
70 IF CON2 > 70 AND NOT CLOCKWISE~TO~CLOSE~? THEN START C
80 IF CON2 < 30 AND CLOCKWISE~TO~CLOSE~? THEN START C
90 IF CON2 < 30 AND NOT CLOCKWISE~TO~CLOSE~? THEN START B
100 IF CON2 > 35 AND CON2 < 65 THEN STOP B , STOP C
110 IF+ B THEN STOP C
120 IF+ C THEN STOP B

125 X = 1 / SCANS
130 IF CON2 < 99 AND CON2 > 1 THEN 140 ELSE 210
140 IF CON2 < 36 OR CON2 > 64 THEN START D ELSE STOP D
150 IF TIME-ON( D ) > 0:04:00 THEN 210
170 Y = X + Y
180 IF Y > 5 THEN Y = 0
190 Z = ABS( CON2 - 50 ) / 50
200 IF Y > Z THEN STOP B , STOP C

210 IF B THEN DAMPER~MOTOR~DRIVE = 10
220 IF C THEN DAMPER~MOTOR~DRIVE = 0
230 IF NOT B AND NOT C THEN DAMPER~MOTOR~DRIVE = 5
234 IF DAMPER~MOTOR~DRIVE > 7 THEN START G ELSE STOP G
236 IF DAMPER~MOTOR~DRIVE < 3 THEN START H ELSE STOP H
240 IF TIME-ON( G ) > 0:06:00 OR TIME-ON( H ) > 0:06:00 THEN
DAMPER~MOTOR~DRIVE = 5 , START I ELSE STOP I
246 IF TIME-ON( I ) > 6:00:00 THEN STOP G , STOP H , STOP B ,
STOP C

250 IF CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE < 3 THEN
DAMPER~POSITION = DAMPER~POSITION + X * 100 / DAMPER~DRIVE~TIME
260 IF CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE > 7 THEN
DAMPER~POSITION = DAMPER~POSITION - X * 100 / DAMPER~DRIVE~TIME
270 IF NOT CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE < 3 THEN
DAMPER~POSITION = DAMPER~POSITION - X * 100 / DAMPER~DRIVE~TIME
280 IF NOT CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE > 7 THEN
DAMPER~POSITION = DAMPER~POSITION + X * 100 / DAMPER~DRIVE~TIME

290 IF DAMPER~POSITION > 100 THEN DAMPER~POSITION = 100
300 IF DAMPER~POSITION < 0 THEN DAMPER~POSITION = 0
310 END

```

### Program 2–TIME PROPORTIONAL HT

The following program is supplied with models KMD-7003 and KMD-7053. To modify the program for time-proportional reheat with a fan, see [Adding fan control](#) on page 3-55 for modification instructions.

```
20 REM ----- TIME PROPORTIONAL REHEAT SEQUENCE
-----
25 REM -- Designed to drive a solid state relay connected to
electric reheat.
26 REM Time base is fixed at 10 seconds - ie, a call for 50%
heat turns on
27 REM relay (OUT1) for 5 seconds, off for 5 seconds.
28 REM Uses CON#3 for control of reheat operation! ( Full PID
function )
30 IF NOT REHEAT~STATUS THEN STOP STAGE~1 , END
40 K = 1 / SCANS
50 L = L + K : IF L > 10 THEN L = 0
60 IF L < CON3 / 10 THEN START STAGE~1 ELSE STOP STAGE~1
70 END
```

### Program 3 3–STAGE REHEAT

The following program is supplied with models KMD-7003 and KMD-7053. To modify the program for 2-stage reheat with a fan, see [Adding fan control](#) on page 3-55 for modification instructions.

```
10 REM ----- 3-STAGE REHEAT SEQUENCE
-----
20 REM -- STAGES ON AT 30, 60, 90 % OF PROPORTIONAL BAND BELOW
SETPOINT.
30 REM Auxillary flow setpoint is used only when AUX-FLOW
(VAR28) is not
40 REM zero and anytime stage 1 heat (OUT1) is requested.
50 REM Entire reheat sequence is easily disabled by toggling
VAR27 on-off.
60 IF NOT REHEAT~STATUS THEN STOP STAGE~1 , STOP STAGE~2 , STOP
STAGE~3 , STOP A , END
70 IF CON3 > 30 THEN START STAGE~1 , START A
80 IF CON3 > 60 THEN START STAGE~2
90 IF CON3 > 90 THEN START STAGE~3
100 IF CON3 < 80 THEN STOP STAGE~3
110 IF CON3 < 50 THEN STOP STAGE~2
120 IF CON3 < 20 THEN STOP STAGE~1 , STOP A
130 IF AUXILLARY~FLOW = 0 THEN 150 : REM Since Auxillary flow is
zero its not used
140 IF A THEN REQUESTED~FLOW = AUXILLARY~FLOW /
VOLUME~CONVERSION
150 END
```

**Program 4–BALANCER override**

```

10 REM **** Balancer control of unit to either MIN or MAX flow
   ***
20 REM ** Uses var25 activesp in digital fashion to select
   desired flow **

25 IF+ ACTIVE~TEMP~SETPOINT > 1 THEN ACTIVE~TEMP~SETPOINT = 1
27 IF+ ACTIVE~TEMP~SETPOINT < 0 THEN ACTIVE~TEMP~SETPOINT = 0
30 IF ACTIVE~TEMP~SETPOINT THEN REQUESTED~FLOW = MAXIMUM~FLOW /
   VOLUME~CONVERSION ELSE REQUESTED~FLOW = MINIMUM~FLOW~ /
   VOLUME~CONVERSION
40 ACTUAL~FLOW~ = AIR~FLOW~SENSOR * SENSOR~CORRECTION : REM
   ACTUAL FLOW CORRECTED (FPM)
50 DUCT~VOLUME~READOUT = ACTUAL~FLOW~ * VOLUME~CONVERSION : REM
   LIVE VOLUME READOUT - UNITS MATCH VOLUME

60 IF CON2 > 70 AND CLOCKWISE~TO~CLOSE~? THEN START B
70 IF CON2 > 70 AND NOT CLOCKWISE~TO~CLOSE~? THEN START C
80 IF CON2 < 30 AND CLOCKWISE~TO~CLOSE~? THEN START C
90 IF CON2 < 30 AND NOT CLOCKWISE~TO~CLOSE~? THEN START B
100 IF CON2 > 35 AND CON2 < 65 THEN STOP B , STOP C
110 IF+ B THEN STOP C
120 IF+ C THEN STOP B

125 X = 1 / SCANS
130 IF CON2 < 99 AND CON2 > 1 THEN 140 ELSE 210
140 IF CON2 < 36 OR CON2 > 64 THEN START D ELSE STOP D
150 IF TIME-ON( D ) > 0:04:00 THEN 210
170 Y = X + Y
180 IF Y > 5 THEN Y = 0
190 Z = ABS( CON2 - 50 ) / 50
200 IF Y > Z THEN STOP B , STOP C

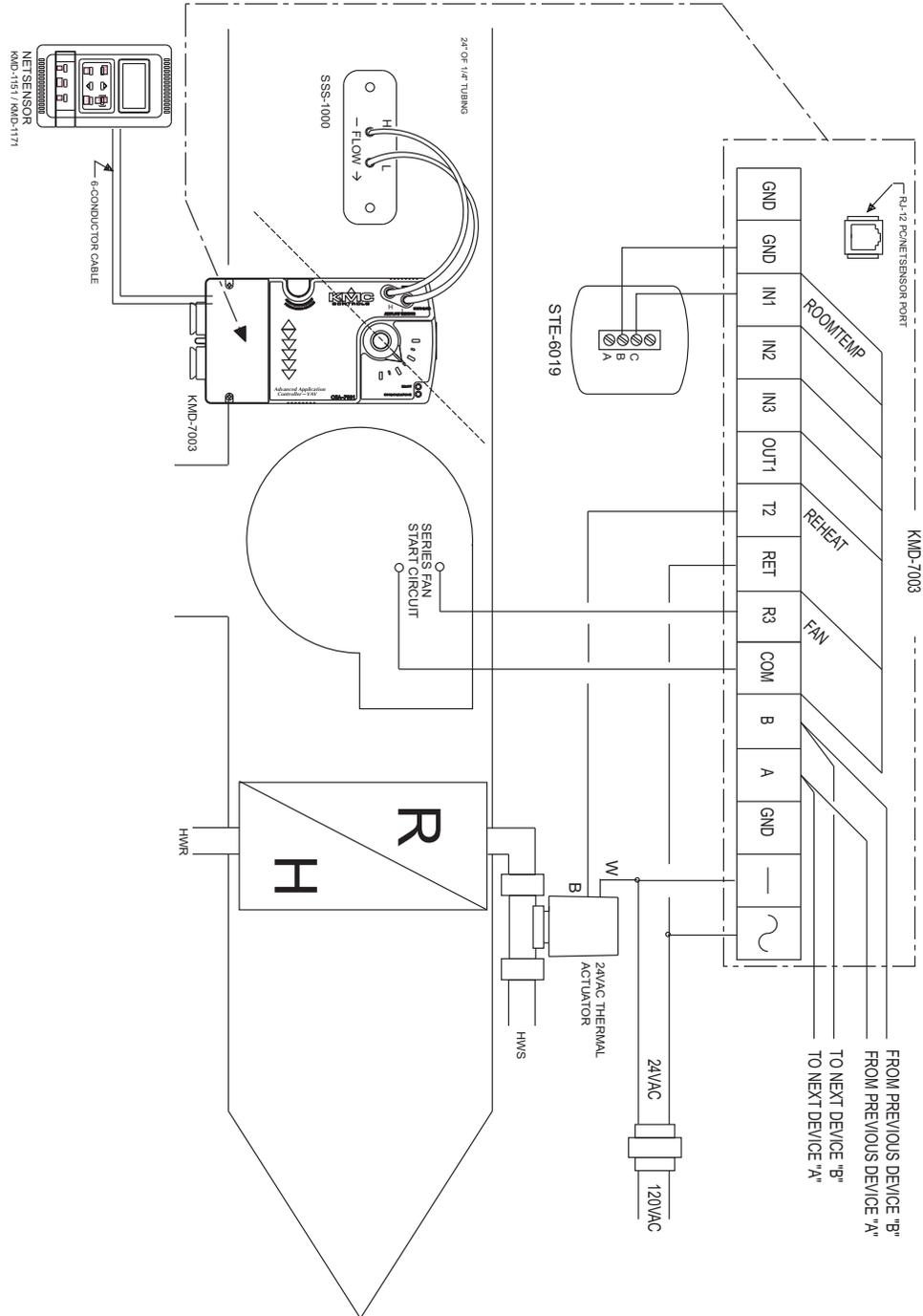
210 IF B THEN DAMPER~MOTOR~DRIVE = 10
220 IF C THEN DAMPER~MOTOR~DRIVE = 0
230 IF NOT B AND NOT C THEN DAMPER~MOTOR~DRIVE = 5

250 IF CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE < 3 THEN
   DAMPER~POSITION = DAMPER~POSITION + X * 100 / DAMPER~DRIVE~TIME
260 IF CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE > 7 THEN
   DAMPER~POSITION = DAMPER~POSITION - X * 100 / DAMPER~DRIVE~TIME
270 IF NOT CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE < 3 THEN
   DAMPER~POSITION = DAMPER~POSITION - X * 100 / DAMPER~DRIVE~TIME
280 IF NOT CLOCKWISE~TO~CLOSE~? AND DAMPER~MOTOR~DRIVE > 7 THEN
   DAMPER~POSITION = DAMPER~POSITION + X * 100 / DAMPER~DRIVE~TIME

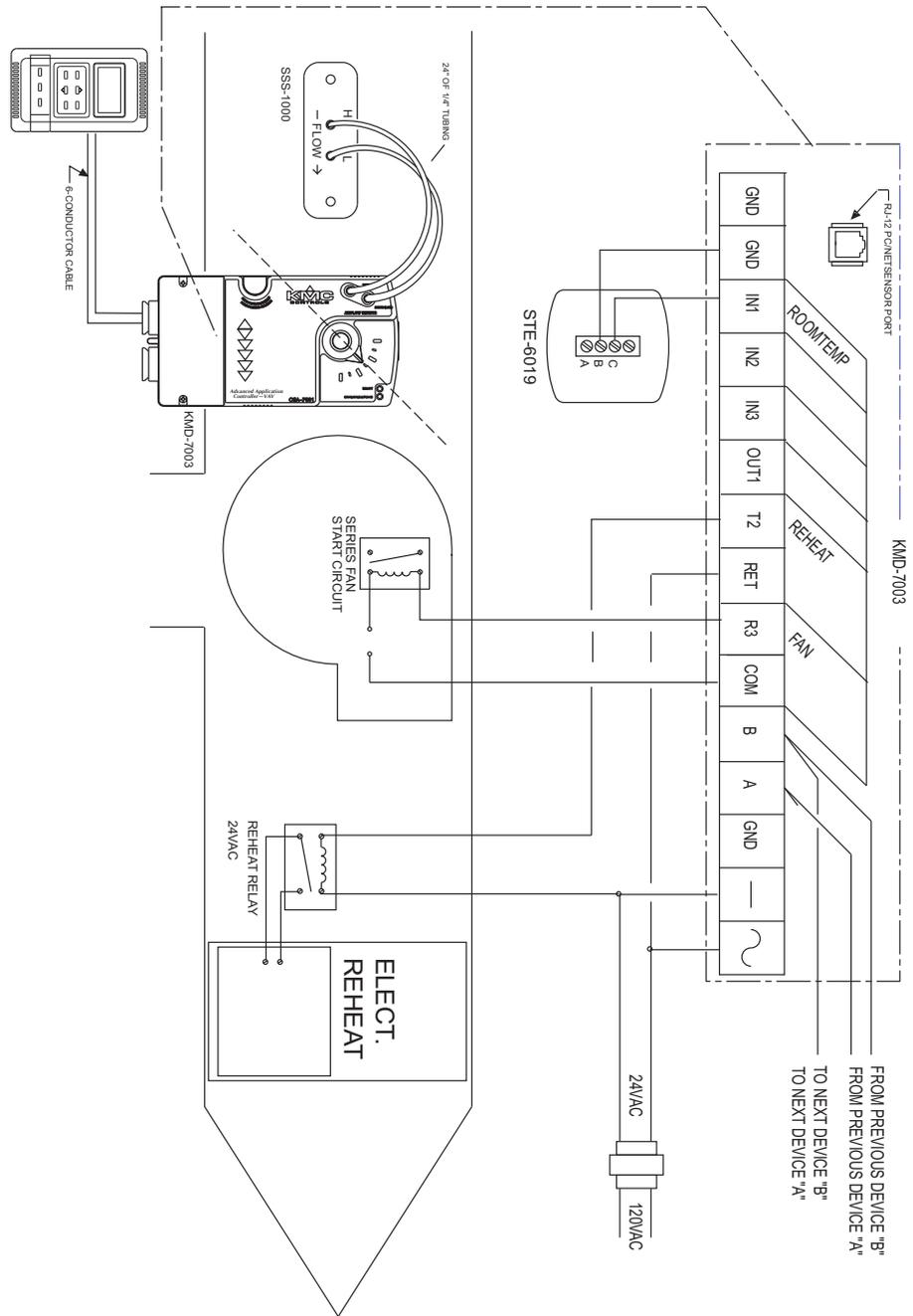
290 IF DAMPER~POSITION > 100 THEN DAMPER~POSITION = 100
300 IF DAMPER~POSITION < 0 THEN DAMPER~POSITION = 0
310 END

```

**Application drawings**



**Illustration 16 Variable Air Volume with hot-water reheat**



**Illustration 17** Variable Air Volume with electric reheat



## Application Note–AN0404D

---

Revision A

### **KMD–7011 VAV Controller**

This application note describes methods and programming to use the KMD–7011 controller in single-duct VAV installations with both electric and wet reheat. It includes methods to positioning a damper with electric or pneumatic actuators.

---

#### **Related materials**

In addition to the material presented in this installation and operation guide, review and have available the following reference materials.

- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide
- ◆ System plans with controller addresses
- ◆ AN0805A, The NetSensor Balancer Program

Installation guides for the following:

- ◆ KMD–7011
- ◆ MSP–5072
- ◆ STE–5012
- ◆ MCP–3611
- ◆ XEC–3001
- ◆ VCP–11xx
- ◆ REE–2104

#### **Introduction**

The KMD-7011 is a KMC digital DDC Controller with a built in flow sensor. It is similar to the KMD-7001 but is not built into a damper actuator. A pressure-independent “pre-programmed” VAV sequence automatically modulates flow into the conditioned space between Minimum and Maximum setpoints as defined by the user. This sequence allows for Heating, Cooling, or changeover applications with minimal user initialization. Additionally, user programming can customize each application as required.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory configurations allowing you to program any desired sequence.

This feature allows you to custom tailor the control sequence to match specific applications. The following paragraphs explain the pre-programmed control sequence in detail and offer suggestions for user-programming to customize operation of the controllers.

The KMD-7011 VAV Terminal Controller contains the basic programming sequence required for three specific applications:

- ◆ Single Duct VAV (heating-cooling-changeover)
- ◆ Single Duct VAV with time proportional reheat
- ◆ Single Duct VAV with three-stage reheat (including fan-box applications)

There is also a balancing program included to allow the use of the NetSensor as a balancing tool to assist VAV box setup. For complete information on the balance program see [AN0404B](#) on page 3-3.

Initially these devices will default to a standard Single Duct Cooling VAV configuration. You can use Either of the Reheat functions by simply turning “ON” the respective programs and “enabling” a variable on the Variables screen. The following paragraphs provide specific details and explanations to help you use the KMD-7011.

## Velocity sensor

The KMD-7011 incorporates an on-board flow-through type sensor to determine duct air flow. Air flow through the sensor is approximately 1.12 SCFH (32 liters/hour) at .25 INWC (62.5 Pa) pressure drop. The sensor uses two RTD's (resistance temperature detectors) to measure flow. The RTD's are platinum films on a ceramic base chosen for long term stability and accuracy. The smaller 100 ohm RTD is self-heated to a constant temperature differential above ambient. The amount of power required varies with the specific airflow. A larger 1000-ohm RTD constantly senses ambient temperature and compensates. The sensor housing channels the airflow over the self-heated RTD in a manner to isolate its heat from the compensator. This minimizes shifts at low flows due to various mounting positions. The sensor housing itself is molded from plastic which carries the UL94-5V rating. The sensor ports are .188 inch diameter and readily accepts .25 inch O.D. “FR” tubing (.040 wall) or equivalent.

## Velocity limits

Minimum and Maximum Flow velocity limits can be defined in either of two ways: Through KMC digital operating software or with the NetSensor. Setting velocity limits through the KMC digital software is as simple as defining the value of a variable. Using this method the limits may be entered in any units desired (CFM, L/s, Pa, etc.) as long as a “conversion factor” is also provided. The “default” settings are always in Feet Per Minute (FPM). For details on using this method refer to the following section on standard VAV operation.

## Standard Variable Air Volume (VAV) Operation

The standard VAV sequence of operation has been pre-programmed into each KMD-7011 controller. Default operation is defined as:

- ◆ Single Duct Cooling
- ◆ Minimum and Maximum Limits at 500 and 2000 FPM respectively
- ◆ Cooling setpoint at 74°F
- ◆ Space Temperature Sensor connected to IN1

These units are capable of independent, stand-alone operation or they may be networked together with other KMC digital products. If stand-alone operation is desired, simply supplying 24 VAC power and connecting a 10K thermistor (STE-series) to IN1 will enable full operation to the default parameters listed above. If these units will be networked together it may be necessary to configure each panel with a unique address and other set-up options using HCM. Refer to the Operator's Manual for details on using HCM.

The following is a simplified explanation of the control logic used in the standard VAV application:

- ◆ A full-function PID controller (CON1) is set to perform the thermostat function by comparing the space temperature (IN1) to the set point temperature (VAR25). In the COOL (default) changeover mode, thermostat action is Direct Acting and "Active" Setpoint is the Cooling setpoint temperature (VAR30). Conversely, if HEAT changeover mode was selected, thermostat action would change to Reverse Acting and the "Active" set point would be the Heating Set Point (VAR29).
- ◆ A "requested" flow set point is calculated based on the thermostat information and Minimum / Maximum flow limits. Using this "Requested" flow a second full function PID Controller (CON2) look at actual flow (IN4) as measured by the velocity sensor and instructs the damper to open or close accordingly.
- ◆ Parameters of either the Thermostat (CON1) or the Flow Controller (CON2) such as Proportional Band, Reset (Integral), or Rate (Derivative) functions can easily be changed to suit application requirements. The "BIAS" of 50% preset on CON2 should not be changed as this defines a satisfied "Neutral" position for the actuator drive circuit.

## Hardware Configuration Manager (HCM)

For networked operation, use the HCM software to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must

be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 1 Default Configuration for KMD-7011**

|                   |           |
|-------------------|-----------|
| Panel Number      | 124       |
| Network Baud Rate | 38400     |
| Last Panel        | Unchecked |
| Panel Type        | 119       |

## Setting up with WinControl XL

This section covers the set-up requirements to activate any of the pre-programmed options available in this device. At this point, communications should have been established with the unit.

A "Master" password is available to "sign-on" to each panel. Other security codes can be entered to control access to the system.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence

The first step after signing on to the panel is to "Load Descriptors One" or "Load Descriptors All". Many of the descriptions have been pre-defined for variables, outputs, etc. to aid in configuring the desirable options.

To use either the time-proportional (OUT1) Reheat program or the 3-stage (OUTS 1, 2, 3) Reheat program, go to the "Control Basic" programs screen and turn "ON" the appropriate program.

After the appropriate programs are enabled you can use the setup screen in Control Group 2 (GRP2) which lists the items most likely to be monitored. The following section outlines each of the parameters that have been pre-programmed into the device.

**Configuring inputs**

Inputs 1, 2, and 3 can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:

| # | Description      | Manual                              | Value  | Units            | Decom.                   | Label    |
|---|------------------|-------------------------------------|--------|------------------|--------------------------|----------|
| 1 | ROOM TEMPERATURE | <input checked="" type="checkbox"/> | 72.00  | Deg. F           | <input type="checkbox"/> | ROOMTEMP |
| 2 |                  | <input type="checkbox"/>            | 0.00   | Unused           | <input type="checkbox"/> |          |
| 3 |                  | <input type="checkbox"/>            | 0.00   | Unused           | <input type="checkbox"/> |          |
| 4 | AIR FLOW SENSOR  | <input type="checkbox"/>            | 250.00 | Unused (Table 2) | <input type="checkbox"/> | FLAWSNSR |

**Illustration 1** Input configuration with WinControl XL

**Input 1 -Room Temperature** Default selects a KMC 10K ohm thermistor as used in all of KMC's STE series sensors.

**Input 2** Unused

**Input 3** Unused

**Input 4 Air Flow Sensor** The on-board flow-sensor. Table 2 has been configured to convert the voltage input into feet per minute. This input is not accessible for any other use.

**Configuring outputs**

The KMD-7011 VAV Controller features a triac output for energizing a fan start circuit, two universal outputs ( analog or digital, set in software) and two triacs to drive the damper motor open or closed. The KMC supplied control routines have been written to use the following output configuration:

| # | Description         | Manual                   | Value | Units      | Label |
|---|---------------------|--------------------------|-------|------------|-------|
| 1 | STAGE 1             | <input type="checkbox"/> |       | Off/Off/On |       |
| 2 | STAGE 2             | <input type="checkbox"/> |       | Off/Off/On |       |
| 3 | STAGE 3             | <input type="checkbox"/> |       | Off/Off/On |       |
| 4 | DAMPER MOTOR DRIVER | <input type="checkbox"/> | 5.00  | Volts      | MOTOR |

**Illustration 2** Output configuration with WinControl XL

**Output 1-Stage 1** This output is configured to activate the first stage of heating on VAV boxes with electric reheat (PRG3). Optionally, it can be used to control any 0-10 VDC device by adding your own custom programming in program area 5 (PRG5).

**Output 2–Stage 2** This output is configured to activate the first stage of heating on VAV boxes with electric reheat (PRG3). Optionally, it can be used to control any 0-10 VDC device by adding your own custom programming in program area 5 (PRG5).

**Output 3–Stage 3** This output is configured to activate the first stage of heating on VAV boxes with electric reheat (PRG3). Optionally, it can be used to control any 0-10 VDC device by adding your own custom programming in program area 5 (PRG5).

**Output 4–DAMPER MOTOR DRIVE** This output is used to control the damper position. The damper will be positioned based on this percentage of the output signal. With a value of 50% the damper will be at 50% of the output voltage range.

### Configuring variables

A total of thirty-two system variables are available to perform various functions such as set points, limits, and modes. All variables can be changed or modified to suit the specific application. The following have been pre-defined for use in the KMD-7011 factory supplied control sequences:

**Variable 17–CHANGEVER MODE (COOL/HEAT)** Chooses whether the unit will operate as a heating or cooling box by automatically setting the thermostat action as direct (COOL) or reverse (HEAT).

**Variable 18–CLOCKWISE TO CLOSE? (NO/YES)** Defines which way the damper will rotate to increase or decrease air-flow as required. “NO” will close the box in a counter-clockwise direction.

**Variable 19–MINIMUM FLOW** Sets the lowest design flow for the VAV box. The default setting is for a Feet-Per-Minute (FPM) value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable 20–MAXIMUM FLOW** Sets the highest design flow for the VAV box. The default setting is for a FPM value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable 21–VOLUME CONVERSION** Default value is 1.00 (no conversion). This factor allows all flow settings and live readings to be in any desired units. Since default calculations are in FPM, the duct area is required (in square feet) to get readings in CFM. Similarly, values can be converted to L/s, Pa, etc. with the proper factors applied.

**Variable 22–SENSOR CORRECTION** Default value is 1.00 (no correction required). This factor is provided to allow simple correction for erroneous sensor readings. This factor is multiplied against all flow readings and the product is stored as the ACTUAL FLOW (VAR23).

**Variable 23–ACTUAL FLOW** This value is calculated from the flow sensor input (IN4) and the sensor correction variable (VAR22) using the following formula:

$$\text{ACT-FLOW} = \text{FLOWSNR} * \text{VAR22}$$

Or:

$$\text{VAR23} = \text{IN4} * \text{VAR22}$$

**Variable 24–REQUESTED FLOW** This is the flow set point based on the heating or cooling needs of the space. The value is set automatically in Program 1 (PRG1).

**Variable 25–ACTIVE TEMP SETPOINT** This is the temperature the controller will attempt to maintain in the space. The active set point is the heating set point (VAR29) in HEAT MODE (VAR17=ON) and the cooling set point (VAR30) in COOL MODE (VAR17 = OFF)

**Variable 26–DUCT VOLUME READOUT** This variable has been set up to allow easy access to a live velocity read-out. Units can be converted using the Volume Conversion factor (VAR21) to relate to CFM, L/s, Pa, etc. as needed. The following formula is used in Program 1 (PRG1) to obtain this value:

$$\text{VOLUME1} = \text{ACT-FLOW} * \text{VOLFACTR}$$

Or:

$$\text{VAR26} = \text{VAR23} * \text{VAR21}$$

**Variable 27–REHEAT STATUS** This flag will enable or disable the REHEAT function. If reheat is disabled, all outputs being used in that reheat function are overridden to “OFF”. To choose one of the pre-programmed Reheat options, the appropriate program must be turned on. See the Program Menu for options. If neither reheat program is turned on, this variable has no purpose.



#### Note

A common application is to use this flag to disable the Reheat function during the summer mode.

**Variable 28–AUXILIARY FLOW** Assigning a flow set point other than 0 in this variable allows the VAV set point to be overridden when STAGE 1 (OUT1) heat is turned “ON”. The auxiliary flow set point is only enabled if STAGE HEATING (PRG3) is used, the REHEAT STATUS (VAR27) is enabled, and the value is non-zero. To disable the auxiliary flow, simply leave the value at 0 (zero).

**Variable 29–HEAT TEMP SETPOINT** This is the active set point used by CON1 when in the heating mode (VAR17=ON). Units must match that of the temperature sensor on IN1 (deg. F or deg. C)

**Variable 30–COOL TEMP SETPOINT** This is the active set point used by CON1 when in the cooling mode (VAR17=OFF). Units must match that of the temperature sensor on IN1 (deg. F or deg. C)

#### Configuring controllers (PID control loops)

KMC digital products feature built-in, full function PID (Proportional, Integral, Derivative) controllers. All parameters may be modified as necessary.

**Controller 1 (CON1)** is a direct-acting thermostat. It compares ROOM TEMPERATURE (IN1) to the ACTIVE TEMP SETPT (VAR25) to determine the VAV flow requirement.

**Controller 2 (CON2)** is set up as a direct-acting thermostat. It compares ACTUAL FLOW (VAR23) to the REQUESTED FLOW (VAR24) to operate the DAMPER (OUT4). All values used by the controller have been converted to FPM to allow a common proportional band value regardless of the units selected. Default is a 200

FPM proportional band which relates to approximately 60 FPM “dead band” to stabilize actuator operation. Default parameters do not include any integral or derivative function.



**Note**

The desired air flow is VAR24 and is automatically calculated by the VAV program. For example, to override the airflow to some other value under specific conditions, simply write a “user” statement such as:

```
10 IF (condition) THEN VAR24 = (value)
```

All values at this point must be in FPM. Condition may be a variable that represents an abnormal event such as a smoke zone activation. To ensure close-off on a box, set the desired flow (value) to a number below zero such as -50. To ensure a wide open damper position use a flow rate higher than 3000 FPM (The highest value the on-board sensor can read).

**Controller 3 (CON3)** compares the ROOM TEMPERATURE (IN1) to the HEATING SET POINT (VAR29) for operation of the reheat stages when programs 2 or 3 are enabled.

**Control Basic programming**

The first program (PRG1) handles set points and flow control. There are two programs provided to control reheat in the VAV box. If neither sequence fits the required application, simply edit the programs or send blank programs to the controller and replace any of the factory programs. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.



**Caution**

This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type)

| # | Description          | On                                  | Manual                   | Exit | Label |
|---|----------------------|-------------------------------------|--------------------------|------|-------|
| 1 | SINGLE DUCT VAV      | <input checked="" type="checkbox"/> | <input type="checkbox"/> | No   |       |
| 2 | TIME PROPORTIONAL HT | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 3 | 3-STAGE REHEAT       | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 4 | BALANCER CONTROL     | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 5 | USER PROGRAMS        | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |

Buttons: Auto Load, Edit, Erase, OK, Cancel

Panel: A1 | Status: Idle

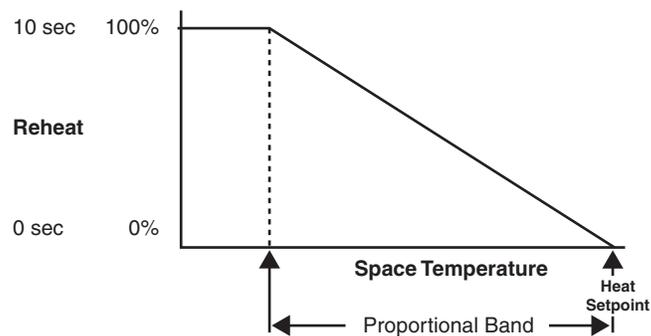
*Illustration 3 program list*

**PROGRAM 1–SINGLE DUCT VAV** This program sets the ACTIVESP (VAR25) based on the CHANGEOVER MODE (VAR17). When the CHANGEOVER MODE is off, the ACTIVE SET POINT equals the Heating Set Point (VAR29). When the CHANGEOVER MODE is on, the ACTIVE SET POINT equals the Cooling Set

Point (VAR30). The necessary Air flow is calculated and the damper motor is modulated to maintain this volume. A damper position is calculated and this value is stored in VAR32.

See [Program 1–Single duct VAV](#) on page 3-83 for a complete listing.

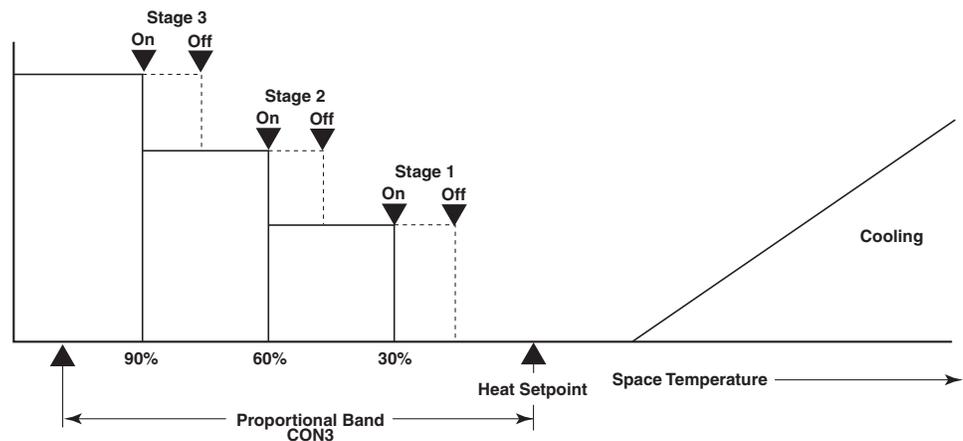
**PROGRAM 2–Time Proportional Reheat** This uses only one “digital” output (OUT1) by controlling the percentage of “ON” time over some specific time interval. For the pre-programmed sequence a fixed 10-second time base is used; A call for 73% heating would turn “ON” the output for 7.3 seconds and turn “OFF” for the remaining 2.7 seconds. See Illustration 4. Using this method of reheat requires only one output.



**Illustration 4 Time Proportional Reheat**

See [Program 2–Time proportional heat](#) on page 3-83 for a complete listing.

**PROGRAM 3–Three-Stage Reheat** This program sequentially steps each stage of reheat on as the requirement for heating increases (using Outputs 1, 2, and 3). Each stage turns “ON” at a fixed point (30%, 60%, and 90%) of the heating thermostat (CON3) value. The controller changes 0-100% over the proportional band range. Illustration 5 illustrates the sequence graphically.



**Illustration 5 Three Stage Reheat**

This program may also be used for FAN-BOX applications. Connect the FAN relay to STAGE 1 (OUT1) and 2 stages of reheat to outputs 2 and 3. A differential is built into the sequence to prevent cycling the outputs over small temperature variations.

An optional AUXILIARY FLOW limit has been pre-programmed for use with the 3-stage Reheat sequence. This feature allows increasing the VAV flow set point to some value other than Minimum Flow anytime Stage 1 of heating is enabled. The value of Auxiliary flow entered in VAR28 must be in the same units as Minimum and Maximum flow since the same “conversion factor” (VAR21) is used by the programming. If this feature is NOT being used, leave the Auxiliary Flow (VAR28) value at zero (0) and it will be ignored by the program.

The programming has been written to instruct the unit to control flow at the Auxiliary Flow limit whenever conditions call for stage 1 heating. If it is desired to initiate the Auxiliary Flow before actually turning “ON” the first stage of heat, simply put a “Delay” time in the appropriate column of OUT1 on the Outputs screen (up to 99 min. 50 sec.). The delay will take precedence over all other software actions.

See [Program 3-Three-stage reheat](#) on page 3-84 for a complete listing.

**PROGRAM 4-BALANCER OVERRIDE** This program is used with the NetSensor to force the controller to maintain either a minimum or a maximum air flow setpoint. Details on using the NetSensor can be found in applications note AN0805A, *The NetSensor Balancer Program*.

See [Program 4-Balancer control](#) on page 3-85 for a complete listing.

### Configuring tables

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7011:

**Table 1** Available for user programming.

**Table 2** Correlates velocity pressure from the on-board flow sensor to feet per minute.

**Table 3** Correlates the dial setting of STE-5000 Series thermostats to degrees F.



---

#### Note

These tables use nominal values for correlation. Due to component tolerances and the resistance of field wiring, these values may need correction on an individual basis. If the value read is too low, enter a positive calibration in the input column labeled “Cal”. If the value read is high, a negative calibration can be entered. If necessary, reconfigure Table 2 and/or Table 3. The

complete Table 2 and Table 3 from the factory are shown in figure 3. If the tables have been reconfigured and the user wants the factory configuration for Table 2 and Table 3 it will be necessary to use Clear Panel or re-enter the values.

**! Caution**

Using Clear Panel will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type.)

| #  | Table 1 | Unused | Table 2 | Unused | Table 3 | Unused |
|----|---------|--------|---------|--------|---------|--------|
| 1  | 0       | 0      | 0       | 0      | 0       | 54     |
| 2  | 0       | 0      | 0.05    | 0      | 0.05    | 54     |
| 3  | 0       | 0      | 0.15    | 190    | 0.54    | 58     |
| 4  | 0       | 0      | 0.4     | 390    | 0.94    | 62     |
| 5  | 0       | 0      | 0.7     | 530    | 1.27    | 66     |
| 6  | 0       | 0      | 1       | 640    | 1.42    | 68     |
| 7  | 0       | 0      | 1.5     | 860    | 1.56    | 70     |
| 8  | 0       | 0      | 2       | 1080   | 1.68    | 72     |
| 9  | 0       | 0      | 2.5     | 1360   | 1.8     | 74     |
| 10 | 0       | 0      | 3       | 1640   | 1.91    | 76     |
| 11 | 0       | 0      | 3.5     | 2000   | 2.01    | 78     |
| 12 | 0       | 0      | 4       | 2380   | 2.2     | 82     |
| 13 | 0       | 0      | 4.5     | 2820   | 2.36    | 86     |
| 14 | 0       | 0      | 5       | 3250   | 2.5     | 90     |
| 15 | 0       | 0      | 5.5     | 3250   | 3       | 90     |

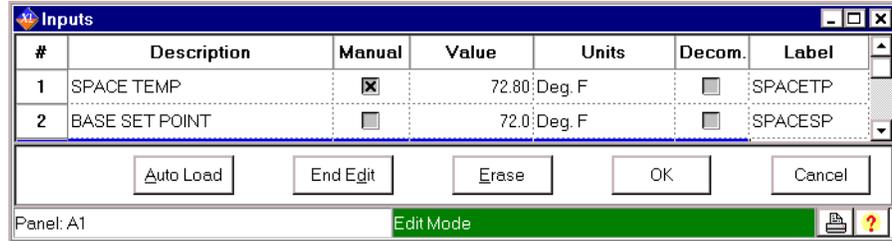
*Illustration 6 Configuring tables in WinControl XL*

**Room setpoint at thermostat**

The primary control logic in this model simply has a heating and cooling set point defined in the software as variables. For applications requiring room set point control at the thermostat, follow these steps:

1. Define an Input as the base setpoint - such as IN2 using the STE-5000 Series Room thermostat.

2. Enter Table 1 in the Range column of the Input window. Set up the Table as shown in Table 2 or Table 3. Table 1 will convert the voltage from the dial into degrees Fahrenheit.



**Illustration 7 Input option for occupancy setpoints**

TABLE 2 0-12 VDC Conversion Table

|   | Value |      |
|---|-------|------|
| 1 | 0.00  | 86.0 |
| 2 | 0.710 | 85.0 |
| 3 | 1.07  | 80.0 |
| 4 | 1.38  | 75.0 |
| 5 | 1.67  | 70.0 |
| 6 | 1.91  | 65.0 |
| 7 | 2.10  | 60.0 |
| 8 | 2.23  | 56.0 |

TABLE 3 0-12 VDC Conversion Table

|   | Value |      |
|---|-------|------|
| 1 | 0.00  | 54.0 |
| 2 | 0.540 | 58.0 |
| 3 | 1.27  | 66.0 |
| 4 | 1.56  | 70.0 |
| 5 | 1.80  | 74.0 |
| 6 | 2.01  | 78.0 |
| 7 | 2.50  | 90.0 |
| 8 | 3.00  | 90.0 |

3. Add programming in PRG5 to set the Heat and Cool set points. The following lines will set the Heating set point to 2° below the set point shown on the thermostat dial and the Cooling set point to 2° above the thermostat dial setting.

```
10 VAR29 = IN2 - 2
20 VAR30 = IN2 + 2
```

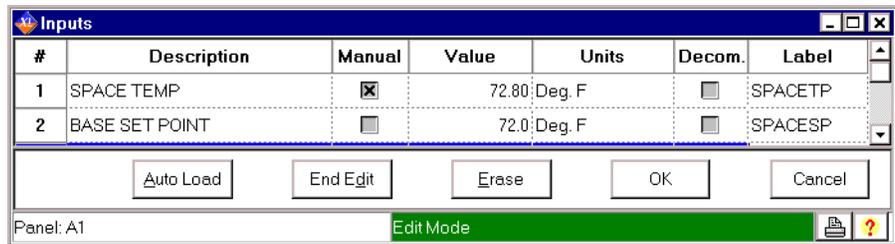
## Using the NetSensor

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7011, simply change the PID controllers 1 and 3 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor.

## Setback (unoccupied) setpoints

Many applications require “setback” or “setup” temperatures as a means of energy savings in unoccupied periods. To accomplish this, it is necessary to establish a condition of occupancy and define when and how these modes occur.

A variable can be set up as a “flag” to indicate the current mode of operation. The units “Un/Occupy” work well for this purpose.

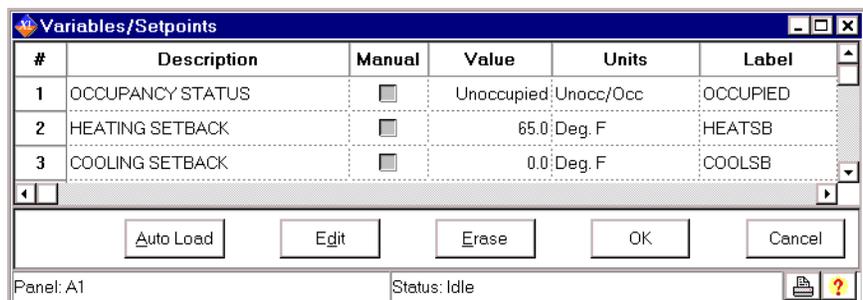


**Illustration 8 Variable option for occupancy setpoints**

Program 5 (PRG5) can contain the code needed to change the state of this Variable based on a weekly schedule as shown in this example:

```
10 IF WS1 THEN START OCCUPIED ELSE STOP OCCUPIED
```

The flag “OCCUPIED” (VAR1 in this example) is “ON” in the occupied mode and off in the unoccupied mode. Two more variables can be set up to define the heating and / or cooling set-back temperatures.



**Illustration 9 Variable option for occupancy setpoints**

Now, some simple programming can be written to determine the set point based on the occupancy mode:

```
10 IF OCCUPIED THEN HEAT-SP = SPACESP - 2 , COOL-SP = SPACESP + 2
```

```
20 IF NOT OCCUPIED THEN HEAT-SP = HEATSB , COOL-SP = COOLSB
```

When in the occupied mode, the heating set point is 2 degrees less than the set point on the thermostat dial and the cooling set point is 2 degrees above. In the unoccupied mode (line 20), the heating set point is set to whatever value is in VAR2 and the cooling set point is set equal to VAR3. It is not necessary to use variables for heating set-back or cooling set-up, but doing so makes modification of these values much easier than getting into the program to change them.

## **Proportional reheat**

A short line of code can be entered into Program area 5 to control hot water reheat. Since Controller 3 (CON3) is set up as a reverse acting thermostat, its value will change from 0 to 100% as temperature falls below the heating setpoint. Configure the output to a range of 0-100% then use:

```
40 OUT1 = CON3
```

## **Other options**

One of the biggest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the factory supplied sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated. The KMC digital Technical Operator's Manual provides help and examples for programming Control Basic

**WinControl settings** The following tables display the WinControl XL PI7s settings for this application note.

**Table 4 Input**

| IN | Description      | Manual | Value   | Units                | Average | Label    |
|----|------------------|--------|---------|----------------------|---------|----------|
| 1  | ROOM TEMPERATURE | Auto   | 146.153 | Deg.F KMC10K Type II | 64      | ROOMTEMP |
| 2  |                  | Auto   | 0       | Unused               | 64      |          |
| 3  |                  | Auto   | 0       | Unused               | 64      |          |
| 4  | AIR FLOW SENSOR  | Auto   | 755.255 | Table 2              | 64      | FLAWSNSR |

**Table 5 Output**

| OUT | Description         | Manual | Value | Units  | Security | Delay | Label |
|-----|---------------------|--------|-------|--------|----------|-------|-------|
| 1   | STAGE 1             | Auto   | Off   | Off/On | 0        | 0:00  |       |
| 2   | STAGE 2             | Auto   | Off   | Off/On | 0        | 0:00  |       |
| 3   | STAGE 3             | Auto   | Off   | Off/On | 0        | 0:00  |       |
| 4   | DAMPER MOTOR DRIVER | Auto   | 0     | 0-100% | 0        | 0:00  | MOTOR |

**Table 6 Variables**

| VAR  | Description                   | Manual | Value    | Units              | Label    |
|------|-------------------------------|--------|----------|--------------------|----------|
| 1-15 | (Available for user programs) |        |          |                    |          |
| 16   | FLOW DEADBAND                 | Auto   | 50       | Feet Per Minute    | DB       |
| 17   | CHANGEOVER MODE               | Auto   | Cool     | Cool/Heat          | HEATING  |
| 18   | NORMALLY OPEN DMPR ?          | Auto   | Yes      | No/Yes             | NOR-OPEN |
| 19   | MINIMUM FLOW                  | Auto   | 500      | Unused             | MIN-FLOW |
| 20   | MAXIMUM FLOW                  | Auto   | 2000     | Unused             | MAX-FLOW |
| 21   | VOLUME CONVERSION             | Auto   | 1        | Unused             | VOLFACTR |
| 22   | SENSOR CORRECTION             | Auto   | 1        | Unused             |          |
| 23   | ACTUAL FLOW                   | Auto   | 756.415  | Feet Per Minute    | ACT-FLOW |
| 24   | REQUESTED FLOW                | Auto   | 2000     | Feet Per Minute    | RQSTFLOW |
| 25   | ACTIVE TEMP SETPOINT          | Auto   | 74       | Unused             | ACTIVESP |
| 26   | DUCT VOLUME READOUT           | Auto   | 756.701  |                    | VOLUME1  |
| 27   | REHEAT STATUS                 | Auto   | Disabled | Dis/Enabled        | REHEAT   |
| 28   | AUXILLARY FLOW                | Auto   | 0        | Unused             | AUX-FLOW |
| 29   | HEAT TEMP SETPOINT            | Auto   | 68       | Degrees Fahrenheit | HEAT-SP  |
| 30   | COOL TEMP SETPOINT            | Auto   | 74       | Degrees Fahrenheit | COOL-SP  |
| 31   |                               | Auto   | 0        |                    |          |
| 32   |                               | Auto   | 0        |                    |          |

**Table 7 Controllers (PID control loops)**

|   | Controlled point | In Value   | Man. | Output  | Setpoint | Value | Action | Prop. | Reset | H/M | Rate | Bias |
|---|------------------|------------|------|---------|----------|-------|--------|-------|-------|-----|------|------|
| 1 | ROOMTEMP         | 0.00 deg.F | auto | 0.0 %   | ACTIVESP | 74.0  | +      | 2     | 0     | H   | 0.00 | 50   |
| 2 | ACT-FLOW         | 0.00       | auto | 100.0 % | RQSTFLOW | 500.0 | +      | 200   | 0     | H   | 0.00 | 50   |
| 3 | ROOMTEMP         | 0.00 deg.F | auto | 100.0 % | HEAT-SP  | 68.0  | -      | 2     | 0     | H   | 0.00 | 0    |
| 4 |                  | 0.00       | auto | 0.0 %   |          | 0.00  | +      | 0     | 0     | H   | 0.00 | 0    |

**Table 8 Tables**

|    | Table 1 | Unused | Table 2 | Unused | Table 3 | Unused |
|----|---------|--------|---------|--------|---------|--------|
| 1  | 0       | 0      | 0       | 0      | 0       | 54     |
| 2  | 0       | 0      | 0.05    | 0      | 0.05    | 54     |
| 3  | 0       | 0      | 0.15    | 190    | 0.54    | 58     |
| 4  | 0       | 0      | 0.4     | 390    | 0.94    | 62     |
| 5  | 0       | 0      | 0.7     | 530    | 1.27    | 66     |
| 6  | 0       | 0      | 1       | 640    | 1.42    | 68     |
| 7  | 0       | 0      | 1.5     | 860    | 1.56    | 70     |
| 8  | 0       | 0      | 2       | 1080   | 1.68    | 72     |
| 9  | 0       | 0      | 2.5     | 1360   | 1.8     | 74     |
| 10 | 0       | 0      | 3       | 1640   | 1.91    | 76     |
| 11 | 0       | 0      | 3.5     | 2000   | 2.01    | 78     |
| 12 | 0       | 0      | 4       | 2380   | 2.2     | 82     |
| 13 | 0       | 0      | 4.5     | 2820   | 2.36    | 86     |
| 14 | 0       | 0      | 5       | 3250   | 2.5     | 90     |
| 15 | 0       | 0      | 5.5     | 3250   | 3       | 90     |

**Table 9 Control Basic list**

| PRG | Description          | On  | A/M  | Label |
|-----|----------------------|-----|------|-------|
| 1   | SINGLE DUCT VAV      | YES | Auto |       |
| 2   | TIME PROPORTIONAL HT | NO  | Auto |       |
| 3   | 3-STAGE REHEAT       | NO  | Auto |       |
| 4   | BALANCER CONTROL     | NO  | Auto |       |
| 5   | USER PROGRAMS        | NO  | Auto |       |

**Program 1—Single duct VAV**

```

1 REM -- Balancer control of unit via prg#4
5 IF TIME-ON( BALANCER~CONTROL ) > 0:30:00 THEN STOP
BALANCER~CONTROL
7 IF BALANCER~CONTROL THEN END

10 IF CHANGEOVER~MODE THEN ACTIVE~TEMP~SETPOINT =
HEAT~TEMP~SETPOINT ELSE ACTIVE~TEMP~SETPOINT =
COOL~TEMP~SETPOINT
20 IF CHANGEOVER~MODE THEN A = 1 - CON1 / 100 ELSE A = CON1 /
100
30 REQUESTED~FLOW = ( ( MAXIMUM~FLOW - MINIMUM~FLOW ) * A +
MINIMUM~FLOW ) / VOLUME~CONVERSION
40 ACTUAL~FLOW~ = AIR~FLOW~SENSOR * SENSOR~CORRECTION : REM
ACTUAL FLOW CORRECTED (FPM)
50 DUCT~VOLUME~READOUT = ACTUAL~FLOW~ * VOLUME~CONVERSION : REM
LIVE VOLUME READOUT - UNITS MATCH
60 IF ABS( ACTUAL~FLOW~ - REQUESTED~FLOW ) < FLOW~DEADBAND THEN
END

70 REM adjust OUTPUT slowly to reach setpoints

80 IF INTERVAL( 0:00:01 ) THEN 90 ELSE END
90 IF ABS( ACTUAL~FLOW~ - REQUESTED~FLOW ) < MAX( FLOW~DEADBAND
* 4 , REQUESTED~FLOW * 0.2 ) THEN C = 0.05 ELSE C = 0.5
100 IF NORMALLY~OPEN~DMPR~? AND ACTUAL~FLOW~ > REQUESTED~FLOW
THEN DAMPER~MOTOR~DRIVER = DAMPER~MOTOR~DRIVER + C
110 IF NORMALLY~OPEN~DMPR~? AND ACTUAL~FLOW~ < REQUESTED~FLOW
THEN DAMPER~MOTOR~DRIVER = DAMPER~MOTOR~DRIVER - C
120 IF NOT NORMALLY~OPEN~DMPR~? AND ACTUAL~FLOW~ >
REQUESTED~FLOW THEN DAMPER~MOTOR~DRIVER = DAMPER~MOTOR~DRIVER -
C
130 IF NOT NORMALLY~OPEN~DMPR~? AND ACTUAL~FLOW~ <
REQUESTED~FLOW THEN DAMPER~MOTOR~DRIVER = DAMPER~MOTOR~DRIVER +
C
140 IF DAMPER~MOTOR~DRIVER > 100 THEN DAMPER~MOTOR~DRIVER = 100
150 IF DAMPER~MOTOR~DRIVER < 0 THEN DAMPER~MOTOR~DRIVER = 0
160 END

```

**Program 2—Time proportional heat**

```

20 REM ----- TIME PROPORTIONAL REHEAT SEQUENCE
-----
25 REM -- Designed to drive a solid state relay connected to
electric reheat.
26 REM Time base is fixed at 10 seconds - ie, a call for 50%
heat turns on
27 REM relay (OUT1) for 5 seconds, off for 5 seconds.
28 REM Uses CON#3 for control of reheat operation! ( Full PID
function )
30 IF NOT REHEAT~STATUS THEN STOP STAGE~1 , END
40 K = 1 / SCANS
50 L = L + K : IF L > 10 THEN L = 0
60 IF L < CON3 / 10 THEN START STAGE~1 ELSE STOP STAGE~1
70 END

```

### Program 3-Three-stage reheat

```
10 REM ----- 3-STAGE REHEAT SEQUENCE
-----
20 REM -- STAGES ON AT 30, 60, 90 % OF PROPORTIONAL BAND BELOW
SETPOINT.
30 REM Auxillary flow setpoint is used only when AUX-FLOW
(VAR28) is not
40 REM zero and anytime stage 1 heat (OUT1) is requested.
50 REM Entire reheat sequence is easily disabled by toggling
VAR27 on-off.
60 IF NOT REHEAT~STATUS THEN STOP STAGE~1 , STOP STAGE~2 , STOP
STAGE~3 , STOP A , END
70 IF CON3 > 30 THEN START STAGE~1 , START A
80 IF CON3 > 60 THEN START STAGE~2
90 IF CON3 > 90 THEN START STAGE~3
100 IF CON3 < 80 THEN STOP STAGE~3
110 IF CON3 < 50 THEN STOP STAGE~2
120 IF CON3 < 20 THEN STOP STAGE~1 , STOP A
130 IF AUXILLARY~FLOW = 0 THEN 150 : REM Since Auxillary flow is
zero its not used
140 IF A THEN REQUESTED~FLOW = AUXILLARY~FLOW /
VOLUME~CONVERSION
150 END
```

**Program 4-Balancer control**

```

10 REM **** Balancer control of unit to either MIN or MAX flow
***
20 REM ** Uses var25 activesp in digital fashion to select
desired flow **

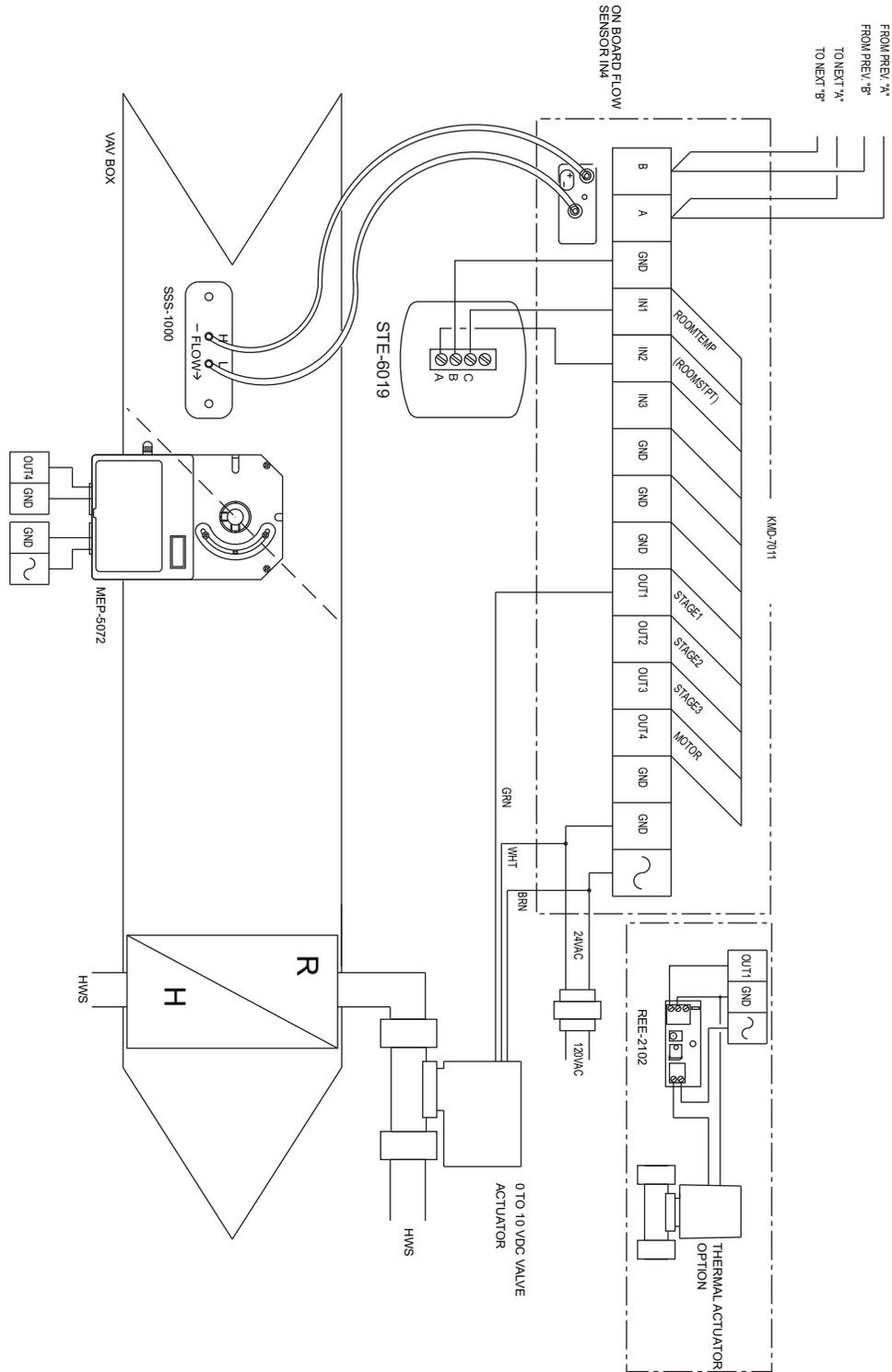
25 IF+ ACTIVE~TEMP~SETPOINT > 1 THEN ACTIVE~TEMP~SETPOINT = 1
27 IF+ ACTIVE~TEMP~SETPOINT < 0 THEN ACTIVE~TEMP~SETPOINT = 0
30 IF ACTIVE~TEMP~SETPOINT THEN REQUESTED~FLOW = MAXIMUM~FLOW /
VOLUME~CONVERSION ELSE REQUESTED~FLOW = MINIMUM~FLOW~ /
VOLUME~CONVERSION
40 ACTUAL~FLOW~ = AIR~FLOW~SENSOR * SENSOR~CORRECTION : REM
ACTUAL FLOW CORRECTED (FPM)
50 DUCT~VOLUME~READOUT = ACTUAL~FLOW~ * VOLUME~CONVERSION : REM
LIVE VOLUME READOUT - UNITS MATCH
60 IF ABS( ACTUAL~FLOW~ - REQUESTED~FLOW ) < FLOW~DEADBAND THEN
END

70 REM adjust OUTPUT slowly to reach setpoints

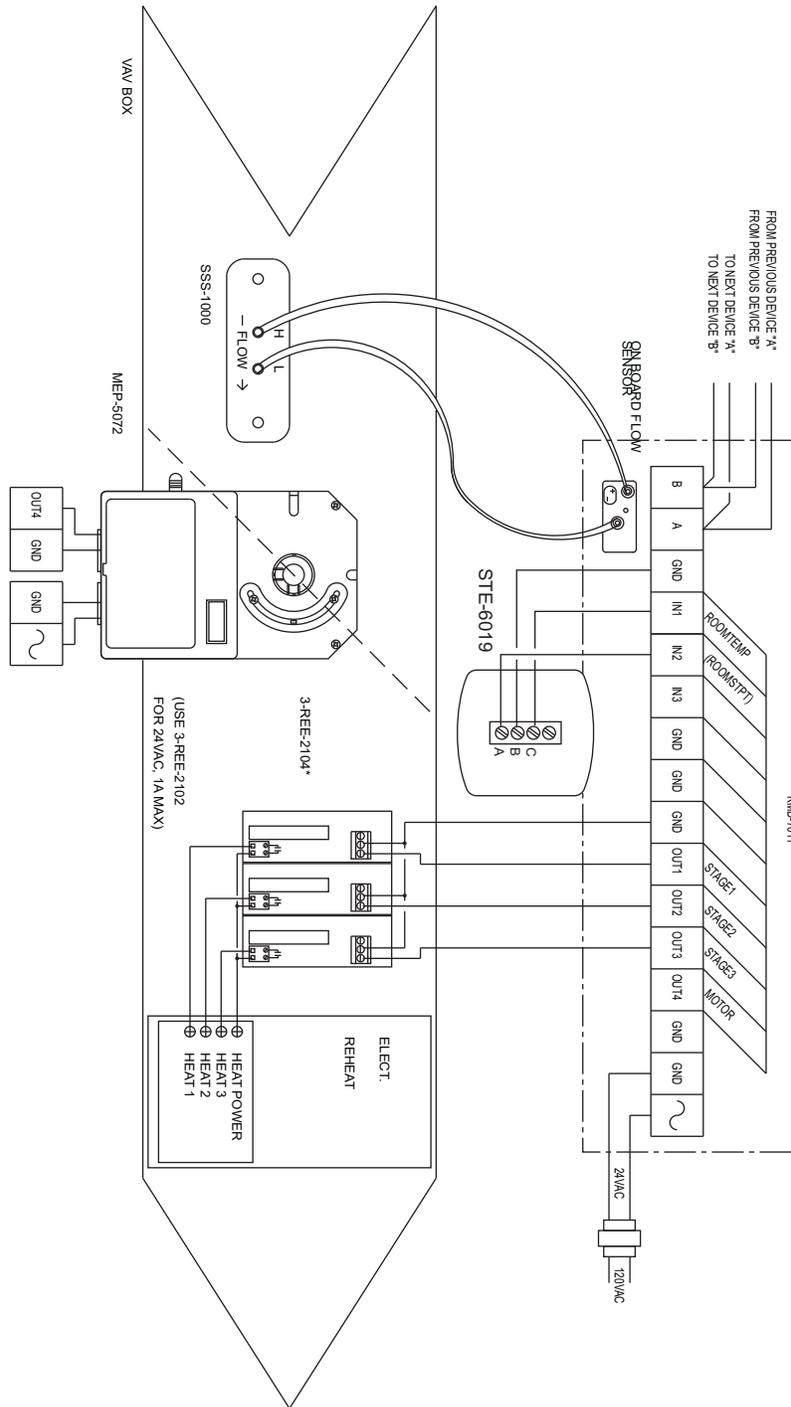
80 IF INTERVAL( 0:00:01 ) THEN 90 ELSE END
90 IF ABS( ACTUAL~FLOW~ - REQUESTED~FLOW ) < MAX( FLOW~DEADBAND
* 4 , REQUESTED~FLOW * 0.2 ) THEN C = 0.05 ELSE C = 0.5
100 IF NORMALLY~OPEN~DMPR~? AND ACTUAL~FLOW~ > REQUESTED~FLOW
THEN DAMPER~MOTOR~DRIVER = DAMPER~MOTOR~DRIVER + C
110 IF NORMALLY~OPEN~DMPR~? AND ACTUAL~FLOW~ < REQUESTED~FLOW
THEN DAMPER~MOTOR~DRIVER = DAMPER~MOTOR~DRIVER - C
120 IF NOT NORMALLY~OPEN~DMPR~? AND ACTUAL~FLOW~ >
REQUESTED~FLOW THEN DAMPER~MOTOR~DRIVER = DAMPER~MOTOR~DRIVER -
C
130 IF NOT NORMALLY~OPEN~DMPR~? AND ACTUAL~FLOW~ <
REQUESTED~FLOW THEN DAMPER~MOTOR~DRIVER = DAMPER~MOTOR~DRIVER +
C
140 IF DAMPER~MOTOR~DRIVER > 100 THEN DAMPER~MOTOR~DRIVER = 100
150 IF DAMPER~MOTOR~DRIVER < 0 THEN DAMPER~MOTOR~DRIVER = 0
160 END

```

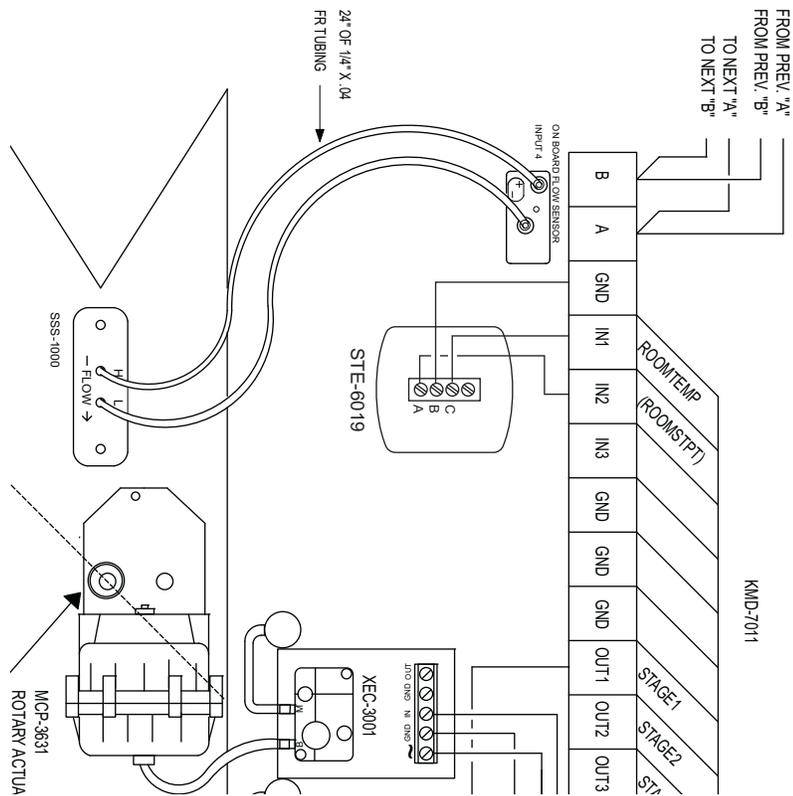
Application drawings



**Illustration 10 KMD-7011 with electronic actuator and wet reheat**



**Illustration 11 KMD-7011 with electronic actuator and three-stage reheat**



**Illustration 12 KMD-7011 with pneumatic actuator and wet reheat**

## Application Note–AN0404E

---

Revision A

### **KMD–7013 VAV Controller**

This application note describes methods and programming to use the KMD–7013 controller in single-duct VAV installations with motor controlled floating damper actuators and external fans. It includes using the controller for heat/cool change over, three-stage and proportional reheat.

---

#### **Related materials**

In addition to the material presented in this installation and operation guide, review and have available the following reference materials.

- ◆ Installation and operation guides for KMD–7011, MEP–3011 and STE–5012
- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide
- ◆ System plans with controller addresses
- ◆ AN0805A, The NetSensor Balancer Program

#### **Introduction**

The KMD-7013 is a KMC digital DDC Controller with a built-in flow sensor. It is similar to the KMD-7001 but is not built into a damper actuator. This controller was designed specifically to control floating damper actuators. It contains an output terminal for driving a damper open and one for driving it closed. A pressure-independent “pre-programmed” VAV sequence automatically modulates flow into the conditioned space between Minimum and Maximum setpoints as defined by the user. This sequence allows for Heating, Cooling, or changeover applications with minimal user initialization. Additionally, user programming can customize each application as required.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory configurations allowing you to program any desired sequence.

The KMD-7013 VAV Terminal Controller contains the basic programming sequence required for three specific applications:

- ◆ Single Duct VAV (heating-cooling-changeover)
- ◆ Single Duct VAV with time proportional reheat
- ◆ Single Duct VAV with three-stage reheat (including fan-box applications)

Initially these devices will default to a standard Single Duct Cooling VAV configuration. You can use either one of the Reheat functions by simply turning “ON” the respective programs and “enabling” a variable on the Variables screen. The following paragraphs provide specific details and explanations to help you use the KMD-7013.

## Velocity sensor

The KMD-7013 incorporates an on-board flow-through type sensor to determine duct air flow. Air flow through the sensor is approximately 1.12 SCFH (32 liters/hour) at .25 INWC (62.5 Pa) pressure drop. The sensor uses two RTD's (resistance temperature detectors) to measure flow. The RTD's are platinum films on a ceramic base chosen for long term stability and accuracy. The smaller 100 ohm RTD is self-heated to a constant temperature differential above ambient. The amount of power required varies with the specific air flow. A larger 1000 ohm RTD constantly senses ambient temperature and compensates. The sensor housing channels the air flow over the self-heated RTD in a manner to isolate its heat from the compensator. This minimizes shifts at low flows due to various mounting positions. The sensor housing itself is molded from plastic which carries the UL94-5V rating. The sensor ports are .188 inch diameter and readily accepts .25 inch O.D. “FR” tubing (.040 wall) or equivalent.

## Velocity limits

Minimum and Maximum Flow velocity limits can be defined in either of two ways: Through KMC digital operating software or with the NetSensor. Setting velocity limits through the KMC digital software is as simple as defining the value of a variable. Using this method the limits may be entered in any units desired (CFM, L/s, Pa, etc.) as long as a “conversion factor” is also provided. The “default” settings are always in Feet Per Minute (FPM). For details on using this method refer to the following section on standard VAV operation.

## Standard Variable Air Volume (VAV) operation

The standard VAV sequence of operation has been pre-programmed into each KMD-7013 controller. Default operation is defined as:

- ◆ Single Duct Cooling
- ◆ Minimum and Maximum Limits at 500 and 2000 FPM respectively
- ◆ Cooling setpoint at  $76 \times F$
- ◆ Space Temperature Sensor connected to IN1

These units are capable of independent, stand-alone operation or they may be networked together with other KMC digital products. If stand-alone operation is desired, simply supplying 24 VAC power and connecting a 10K thermistor (STE-series) to IN1 will enable full operation to the default parameters listed above. If these units will be networked together it will be necessary to configure each panel with a unique address and other set-up options using HCM. Refer to the Operator's Manual for details on using HCM.

The following is a simplified explanation of the control logic used in the standard VAV application:

- ◆ A full-function PID controller (CON1) is set to perform the thermostat function by comparing the space temperature (IN1) to the set point temperature (VAR25). In the COOL (default) changeover mode, thermostat action is Direct Acting and “Active” Setpoint is the Cooling setpoint temperature (VAR30). Conversely, if HEAT changeover mode was selected, thermostat action would change to Reverse Acting and the “Active” set point would be the Heating Set Point (VAR29).
- ◆ A “requested” flow set point is calculated based on the thermostat information and Minimum / Maximum flow limits. Using this “Requested” flow a second full function PID Controller (CON2) looks at actual flow (IN4) as measured by the velocity sensor and instructs the damper to open or close accordingly.
- ◆ Parameters of either the Thermostat (CON1) or the Flow Controller (CON2) such as Proportional Band, Reset (Integral), or Rate (Derivative) functions can easily be changed to suit application requirements. The “BIAS” of 50% preset on CON2 should not be changed as this defines a satisfied “Neutral” position for the actuator drive circuit.

### **Configuring the controller with Hardware Configuration Manager (HCM)**

Use the HCM program to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 1 Default Configuration for KMD-7013**

|                   |           |
|-------------------|-----------|
| Panel Number      | 124       |
| Network Baud Rate | 38400     |
| Last Panel        | Unchecked |
| Panel Type        | 117       |

### **Setting up the controller with WinControl XL**

This section covers the set-up requirements to activate any of the pre-programmed options available in this device. At this point, communications should have been established with the unit.

A “Master” password is available to “sign-on” to each panel. Other security codes can be entered to control access to the system.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence.

The first step after signing on to the panel is to “Load Descriptors One” or “Load Descriptors All”. Many of the descriptions have been pre-defined for variables, outputs, etc. to aid in configuring the desirable options.

To use either the time-proportional (OUT1) Reheat program or the 3-stage (OUTS 1, 2, 3) Reheat program, go to the “Control Basic” programs screen and turn “ON” the appropriate program. After the appropriate programs are enabled you can use the setup screen in Control Group 2 (GRP2) which lists the items most likely to be monitored. The following section outlines each of the parameters that have been pre-programmed into the device.

**Configuring inputs**

Inputs 1, 2, and 3 can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:

| # | Description      | Manual                              | Value  | Units            | Decom.                              | Label    |
|---|------------------|-------------------------------------|--------|------------------|-------------------------------------|----------|
| 1 | ROOM TEMPERATURE | <input type="checkbox"/>            | 72.00  | Deg. F           | <input checked="" type="checkbox"/> | ROOMTEMP |
| 2 |                  | <input type="checkbox"/>            | 0.00   | Unused           | <input type="checkbox"/>            |          |
| 3 |                  | <input checked="" type="checkbox"/> | 0.00   | Unused           | <input type="checkbox"/>            |          |
| 4 | AIR FLOW SENSOR  | <input type="checkbox"/>            | 250.00 | Unused (Table 2) | <input checked="" type="checkbox"/> | FLOWSNSR |

**Illustration 1** Input configuration with WinControl XL

**Input 1–Room Temperature** Default selects a KMC 10K ohm thermistor as used in all of KMC's STE series sensors.

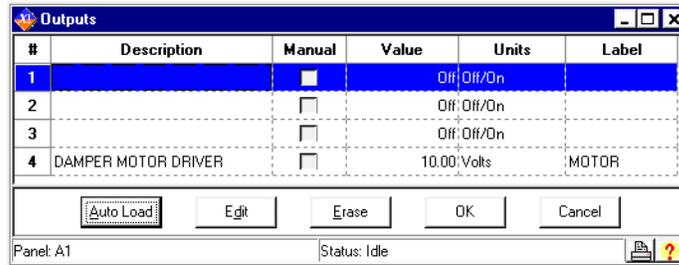
**Input 2** Unused

**Input 3** Unused

**Input 4–Air Flow Sensor** The on-board flow-sensor. Table 2 has been configured to convert the voltage input into feet per minute. This input is not accessible for any other use.

**Configuring outputs**

The KMD-7013 VAV Controller features a triac output for energizing a fan start circuit, two universal outputs ( analog or digital, set in software) and two triacs to drive the damper motor open or closed. The KMC supplied control routines have been written to use the following output configuration:



*Illustration 2 Output configuration with WinControl XL*

**Output 1–Unused (TRIAC)** This is an optically isolated 24vAC triac for starting a fan or controlling a reheat element.

**Output 2–Unused** This is a universal output that can be used to control a proportional device or a stage of reheat.

**Output 3–Unused** This is a universal output that can be used to control a proportional device or a stage of reheat.

**Output 4–DAMPER MOTOR DRIVE** This output is used to control the damper position. It is set up to control a damper motor by turning on one of two triacs to drive the damper open or closed. A value of 0 volts turns on the “CLOSE” damper triac, a value of 10 volts turns on the “OPEN” triac, and a value of 5 volts keeps both triacs off. The damper drive triacs will supply 24VAC at the “OPEN” and “CLOSE” terminals with reference to the “COM” common terminal.

**Configuring variables**

A total of thirty-two system variables are available to perform various functions such as set points, limits, and modes. All variables can be changed or modified to suit the specific application. The following have been pre-defined for use in the KMD-7013 KMC supplied control sequence.

**Variable 17–CHANGEVER MODE (COOL/HEAT)** Chooses whether the unit will operate as a heating or cooling box by automatically setting the thermostat action as direct (COOL) or reverse (HEAT).

**Variable 19–MINIMUM FLOW** Sets the lowest design flow for the VAV box. The default setting is for a Feet-Per-Minute (FPM) value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable 20–MAXIMUM FLOW** Sets the highest design flow for the VAV box. The default setting is for a FPM value. Other units can be used by providing a conversion factor in Variable 21 (VAR21).

**Variable 21–VOLUME CONVERSION** Default value is 1.00 (no conversion). This factor allows all flow settings and live readings to be in any desired units. Since default calculations are in FPM, the duct area is required (in square feet) to get readings in CFM. Similarly, values can be converted to L/s, Pa, etc. with the proper factors applied.

**Variable 22–SENSOR CORRECTION** Default value is 1.00 (no correction required). This factor is provided to allow simple correction for erroneous sensor readings. This factor is multiplied against all flow readings and the product is stored as the ACTUAL FLOW (VAR23).

**Variable 23–ACTUAL FLOW** This value is calculated from the flow sensor input (IN4) and the sensor correction variable (VAR22) using the following formula:

$$\text{ACT-FLOW} = \text{FLOWSNSR} * \text{VAR22}$$

Or:

$$\text{VAR23} = \text{IN4} * \text{VAR22}$$

**Variable 24–REQUESTED FLOW** This is the flow set point based on the heating or cooling needs of the space. The value is set automatically in Program 1 (PRG1).

**Variable 25–ACTIVE TEMP SETPOINT** This is the temperature the controller will attempt to maintain in the space. The active set point is the heating set point (VAR29) in HEAT MODE (VAR17 = ON) and the cooling set point (VAR30) in COOL MODE (VAR17 = OFF)

**Variable 26–DUCT VOLUME READOUT** This variable has been set up to allow easy access to a live velocity read-out. Units can be converted using the Volume Conversion factor (VAR21) to relate to CFM, L/s, Pa, etc. as needed. The following formula is used in Program 1 (PRG1) to obtain this value:

$$\text{VOLUME1} = \text{ACT-FLOW} * \text{VOLFACTR}$$

Or:

$$\text{VAR26} = \text{VAR23} * \text{VAR21}$$

**Variable 27–REHEAT STATUS** This flag will enable or disable the REHEAT function. If reheat is disabled, all outputs being used in that reheat function are overridden to “OFF”. To choose one of the pre-programmed Reheat options, the appropriate program must be turned on. See the Program Menu for options. If neither reheat program is turned on, this variable has no purpose.



---

**Note**

A common application is to use this flag to disable the Reheat function during the summer mode.

**Variable 28–AUXILIARY FLOW** Assigning a flow set point other than 0 in this variable allows the VAV set point to be overridden when STAGE 1 (OUT1) heat is turned “ON”. The auxiliary flow set point is only enabled if STAGE HEATING (PRG3) is used, the REHEAT STATUS (VAR27) is enabled, and the value is non-zero. To disable the auxiliary flow, simply leave the value at 0 (zero).

**Variable 29–HEAT TEMP SETPOINT** This is the active set point used by CON1 when in the heating mode (VAR17 = ON). Units must match that of the temperature sensor on IN1 (deg. F or deg. C)

**Variable 30–COOL TEMP SETPOINT** This is the active set point used by CON1 when in the cooling mode (VAR17 = OFF). Units must match that of the temperature sensor on IN1 (deg. F or deg. C)

**Variable 31–DAMPER DRIVE TIME** This is the approximate time in seconds required for the

damper to move from the fully closed to the fully open position or from the fully open to the fully closed position. It is used in program 1 to roughly determine the damper position in variable 22 (VAR22).

**Variable 32–DAMPER POSITION** This is a calculated position based on the full stroke drive time (VAR31). Damper position is determined in Program 1 by measuring the time the damper has been commanded open or closed versus the time needed for complete travel. If the damper is stuck or the motor fails this variable will not necessarily indicate the correct damper position. It will however indicate the controllers attempt to move the damper. This variable is used only to give an operator an indication of what the controller has done and is doing to control the position of the damper. It is not used as feed back in the control sequence.

### Configuring controllers (PID control loops)

KMC digital products feature built-in, full function PID (Proportional, Integral, Derivative) controllers. All parameters may be modified as necessary.

**Controller 1 (CON1)** is a direct-acting thermostat. It compares ROOM TEMPERATURE (IN1) to the ACTIVE TEMP SETPT (VAR25) to determine the VAV flow requirement.

**Controller 2 (CON2)** is set up as a direct-acting thermostat. It compares ACTUAL FLOW (VAR23) to the REQUESTED FLOW (VAR24) to operate the DAMPER (OUT4). All values used by the controller have been converted to FPM to allow a common proportional band value regardless of the units selected. Default is a 200 FPM proportional band which relates to approximately 60 FPM “dead band” to stabilize actuator operation. Default parameters do not include any integral or derivative function.



### Note

The desired air flow is VAR24 and is automatically calculated by the VAV program. For example, to override the airflow to some other value under specific conditions, simply write a “user” statement such as:

```
10 IF (condition) THEN VAR24 = (value)
```

All values at this point must be in FPM. Condition may be a variable that represents an abnormal event such as a smoke zone activation. To ensure close-off on a box, set the desired flow (value) to a number below zero such as -50. To ensure a wide open damper position use a flow rate higher than 3000 FPM (The highest value the on-board sensor can read).

**Controller 3 (CON3)** compares the ROOM TEMPERATURE (IN1) to the HEATING SET POINT (VAR29) for operation of the reheat stages when programs 2 or 3 are enabled.

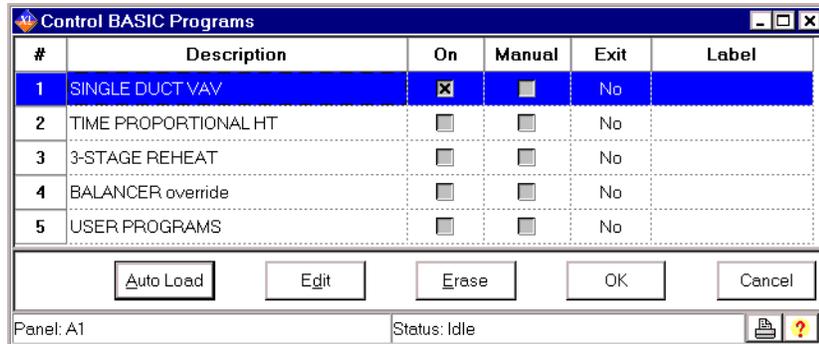
**Control Basic programming**

The first program (PRG1) handles set points and flow control. There are two programs provided to control reheat in the VAV box. If neither sequence fits the required application, simply edit the programs or send blank programs to the controller and replace any of the factory programs. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.



**Caution**

This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type).



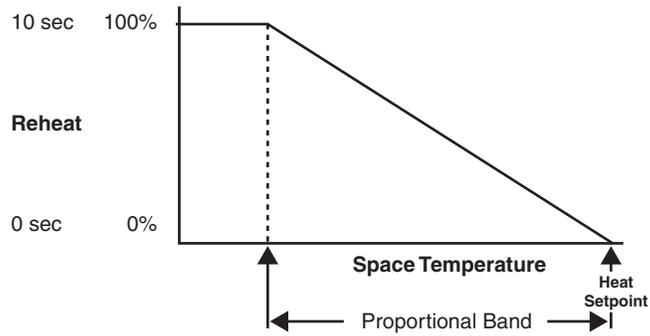
*Illustration 3 Control Basic list*

**PROGRAM 1–Single duct VAV**

This program sets the ACTIVESP (VAR25) based on the CHANGEOVER MODE (VAR17). When the CHANGEOVER MODE is off, the ACTIVE SET POINT equals the Heating Set Point (VAR29). When the CHANGEOVER MODE is on, the ACTIVE SET POINT equals the Cooling Set Point (VAR30). The necessary Air flow is calculated and the damper motor is modulated to maintain this volume. A damper position is calculated and this value is stored in VAR32.

**Program 2–Time proportional reheat**

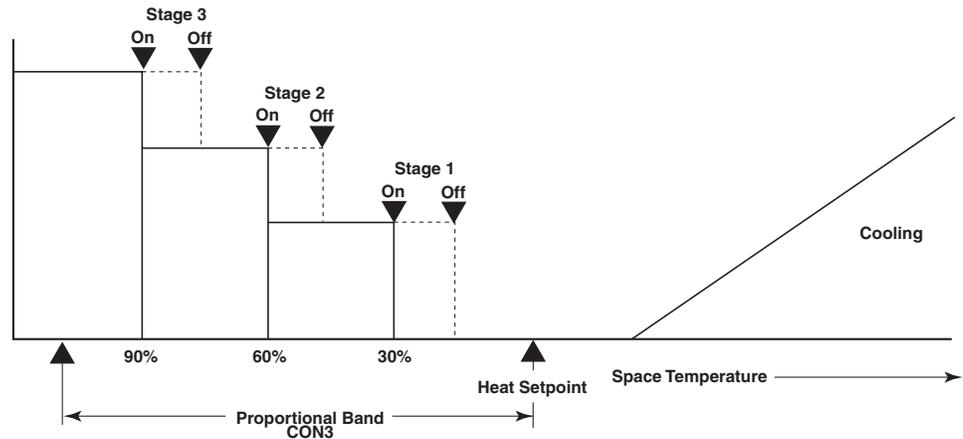
This uses only one “digital” output (OUT1) by controlling the percentage of “ON” time over some specific time interval. For the pre-programmed sequence a fixed 10-second time base is used; A call for 73% heating would turn “ON” the output for 7.3 seconds and turn “OFF” for the remaining 2.7 seconds. See Illustration 4. Using this method of reheat requires only one output.



**Illustration 4 Time Proportional Reheat**

**Program 3–Three-stage reheat**

This program sequentially steps each stage of reheat on as the requirement for heating increases (using Outputs 1, 2, and 3). Each stage turns “ON” at a fixed point (30%, 60%, and 90%) of the heating thermostat (CON3) value. The controller changes 0-100% over the proportional band range. Illustration 5 shows the sequence graphically.



**Illustration 5 Three Stage Reheat**

This program may also be used for FAN-BOX applications. Connect the FAN relay to STAGE 1 (OUT1) and 2 stages of reheat to outputs 2 and 3. A differential is built into the sequence to prevent cycling the outputs over small temperature variations.

An optional AUXILIARY FLOW limit has been pre-programmed for use with the 3-stage Reheat sequence. This feature allows increasing the VAV flow set point to some value other than Minimum Flow anytime Stage 1 of heating is enabled. The value of Auxiliary flow entered in VAR28 must be in the same units as Minimum and Maximum flow since the same “conversion factor” (VAR21) is used by the programming. If this feature is NOT being used, leave the Auxiliary Flow (VAR28) value at zero (0) and it will be ignored by the program.

The programming has been written to instruct the unit to control flow at the Auxiliary Flow limit whenever conditions call for stage 1 heating. If it is desired to initiate the Auxiliary Flow before actually turning “ON” the first stage of heat, simply put a “Delay” time in the appropriate column of OUT1 on the Outputs screen (up to 99 min. 50 sec.). The delay will take precedence over all other software actions.

#### **Program 4–Balancer override**

This program is used with the NetSensor to force the controller to maintain either a minimum or a maximum air flow setpoint. Details on using the NetSensor can be found in in applications note AN0805A, *The NetSensor Balancer Program..*

#### **Program 5–Not used**

Available for user programming.

#### **Configuring tables**

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7013:

**Table 1** Ready for user programming.

**Table 2** Correlates velocity pressure from the on-board flow sensor to feet per minute.

**Table 3** Correlates the dial setting of STE-5000 Series thermostats to degrees F.



---

#### **Note**

These tables use nominal values for correlation. Due to component tolerances and the resistance of field wiring, these values may need correction on an individual basis. If the value read is too low, enter a positive calibration in the input column labeled “cal” on the Inputs window. If the value read is high, a negative calibration can be entered. If necessary, reconfigure Table 2 and/or Table 3. The complete Table 2 and Table 3 from the factory are shown in figure 3. If the tables have been reconfigured and the user wants the factory configuration for Table 2 and Table 3 it will be necessary to use Clear Panel or re-enter the values.

**!** **Caution**

Using Clear Panel will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type.)

| #  | Description          | Manual                   | Value    | Units        | Label     |
|----|----------------------|--------------------------|----------|--------------|-----------|
| 21 |                      | <input type="checkbox"/> | 0.00     | Unused       |           |
| 22 | OCCUPANCY MODE       | <input type="checkbox"/> | Occupied | Unocc/Occ    | OCCUPIED  |
| 23 | ECONOMIZER OPERATION | <input type="checkbox"/> | Disabled | Dis/Enabled  | ECONOMIZ  |
| 24 | FREEZE PROTECT RESET | <input type="checkbox"/> | Normal   | Normal/Alarm | RESET-FZ  |
| 25 | ACTIVE COOLING SETPT | <input type="checkbox"/> | 57.00    | Deg. F       | COOL\$SPT |
| 26 | SETBACK COOL SETPT   | <input type="checkbox"/> | 0.00     | Deg. F       | C\$SETBK  |
| 27 | T'STAT DEADBAND      | <input type="checkbox"/> | 4.00     | Deg. F       | DEADBAND  |
| 28 | ACTIVE HEATING SETPT | <input type="checkbox"/> | 53.00    | Deg. F       | HEAT\$SPT |
| 29 | SETBACK HEAT SETPT   | <input type="checkbox"/> | 62.00    | Deg. F       | H\$SETBK  |
| 30 | MIXED AIR LOW LIMIT  | <input type="checkbox"/> | 50.00    | Deg. F       | LOWLIMIT  |
| 31 | FREEZE PROTECT LIMIT | <input type="checkbox"/> | 40.00    | Deg. F       | FREEZ-UP  |
| 32 | MIN. DAMPER POSITION | <input type="checkbox"/> | 0.00     | %            | MIN\$DMPR |

Panel: A1 Status: Idle

**Illustration 6 KMD-7013 Tables**

**Room setpoint at thermostat**

The primary control logic in this model simply has a heating and cooling set point defined in the software as variables. For applications requiring room set point control at the thermostat, follow the following steps:

1. Define an Input as the base set point - such as IN2 using the STE-5000 Series Room thermostat. Enter Table 3 in the Range column of the Input window. Table 3 will convert the voltage from the dial into degrees Fahrenheit.

| # | Description    | Manual                              | Value | Units  | Decom.                   | Label   |
|---|----------------|-------------------------------------|-------|--------|--------------------------|---------|
| 1 | SPACE TEMP     | <input checked="" type="checkbox"/> | 72.80 | Deg. F | <input type="checkbox"/> | SPACETP |
| 2 | BASE SET POINT | <input type="checkbox"/>            | 72.0  | Deg. F | <input type="checkbox"/> | SPACESP |

Panel: A1 Edit Mode

**Illustration 7 Input option**

2. Add programming in PRG5 to set the Heat and Cool set points. The following lines will set the Heating set point to 2° below the set point shown on the thermostat dial and the Cooling set point to 2× above the thermostat dial setting.

## Using the NetSensor

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7013, simply change the PID controllers 1 and 3 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor.

## Setback (unoccupied) setpoints

Many applications require “setback” or “setup” temperatures as a means of energy savings in unoccupied periods. To accomplish this, it is necessary to establish a condition of occupancy and define when and how these modes occur.

A variable can be set up as a “flag” to indicate the current mode of operation. The units “Un/Occupy” work well for this purpose.

| # | Description      | Manual                   | Value    | Units     | Label    |
|---|------------------|--------------------------|----------|-----------|----------|
| 1 | OCCUPANCY STATUS | <input type="checkbox"/> | Occupied | Unocc/Occ | OCCUPIED |
| 2 |                  | <input type="checkbox"/> | 0.00     | Unused    |          |
| 3 |                  | <input type="checkbox"/> | 0.00     | Unused    |          |

**Illustration 8 Variable option for setback**

Program 5 (PRG5) can contain the code needed to change the state of this Variable based on a weekly schedule as shown in this example:

```
30 IF WS1 THEN START OCCUPIED ELSE STOP OCCUPIED
```

The flag “OCCUPIED” (VAR1 in this example) is “ON” in the occupied mode and off in the unoccupied mode. Two more variables can be set up to define the heating and / or cooling set-back temperatures.

Now, some simple programming can be written to determine the set point based on the occupancy mode:

```
10 IF OCCUPIED THEN HEAT-SP = SPACESP - 2 , COOL-SP = SPACESP + 2
20 IF NOT OCCUPIED THEN HEAT-SP = HEATSB , COOL-SP = COOLSB
```

When in the occupied mode, the heating set point is 2 degrees less than the set point on the thermostat dial and the cooling set point is 2 degrees above. In the unoccupied mode (line 20), the heating set point is set to whatever value is in VAR2 and the cooling set point is set equal to VAR3. It is not necessary to use variables for heating set-back or cooling set-up, but doing so makes modification of these values much easier than getting into the program to change them.

A short line of code can be entered into Program area 5 to control a VAV with hot water reheat. Set the output for a range of 0-10 volts and write the statement:

```
40 OUT1 = CON3 / 10
```

Since Controller 3 (CON3) is set up as a reverse acting thermostat, its value will change from 0 to 100% as temperature falls below the heating set point. The division by 10 makes the output change from 0 to 10 volts as the controller changes from 0 to 100%. If the output range is selected as 0-100% then use:

$$40 \text{ OUT1} = \text{CON3}$$

**Other options**

One of the biggest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the KMC supplied sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated. The KMC digital Technical Operator's Manual provides help and examples for programming Control Basic.

**WinControl settings**

The following tables display the WinControl Plus settings for this application note.

**Table 2 Input configuration**

|   | Description      | Manual | Value | Units                | Average | Label    |
|---|------------------|--------|-------|----------------------|---------|----------|
| 1 | ROOM TEMPERATURE | Auto   | 74.0  | Deg.F KMC10K Type II | 128     | ROOMTEMP |
| 2 |                  | Auto   | 0     | Unused               | 64      |          |
| 3 |                  | Auto   | 0     | Unused               | 64      |          |
| 4 | AIR FLOW SENSOR  | Auto   | 0     | Table 2              | 64      | FLOWNSNR |

**Table 3 Output configuration**

|   | Description         | Manual | Output | Units      | 0 % | 100 | Delay | Security | Label |
|---|---------------------|--------|--------|------------|-----|-----|-------|----------|-------|
| 1 |                     | Auto   | Off    | Off/On     |     |     | 0:00  | 0        |       |
| 2 |                     | Auto   | Off    | Off/On     |     |     | 0:00  | 0        |       |
| 3 |                     | Auto   | Off    | Off/On     |     |     | 0:00  | 0        |       |
| 4 | DAMPER MOTOR DRIVER | Auto   | 10     | 0-10 Volts | 0.0 | 10  | N/A   | 0        | MOTOR |

**Table 4 Variable configuration**

|      | Description                   | Manual | Value   | Units           | Label    |
|------|-------------------------------|--------|---------|-----------------|----------|
| 1-16 | (Available for user programs) |        | 0       |                 |          |
| 17   | CHANGEVER MODE                | Auto   | Cool    | Cool/Heat       | HEATING  |
| 18   |                               | Auto   | 0       |                 |          |
| 19   | MINIMUM FLOW                  | Auto   | 500     | Unused          | MIN-FLOW |
| 20   | MAXIMUM FLOW                  | Auto   | 2000    | Unused          | MAX-FLOW |
| 21   | VOLUME CONVERSION             | Auto   | 1       | Unused          | VOLFACTR |
| 22   | SENSOR CORRECTION             | Auto   | 1       | Unused          |          |
| 23   | ACTUAL FLOW                   | Auto   | 798.343 | Feet Per Minute | ACT-FLOW |

**Table 4 Variable configuration**

|    | <b>Description</b>   | <b>Manual</b> | <b>Value</b> | <b>Units</b>       | <b>Label</b> |
|----|----------------------|---------------|--------------|--------------------|--------------|
| 24 | REQUESTED FLOW       | Auto          | 2000         | Feet Per Minute    | RQSTFLOW     |
| 25 | ACTIVE TEMP SETPOINT | Auto          | 76           | Degrees Fahrenheit | ACTIVESP     |
| 26 | DUCT VOLUME READOUT  | Auto          | 797.35       |                    | VOLUME1      |
| 27 | REHEAT STATUS        | Auto          | Disabled     | Dis/Enabled        | REHEAT       |
| 28 | AUXILLARY FLOW       | Auto          | 0            | Unused             | AUX-FLOW     |
| 29 | HEAT TEMP SETPOINT   | Auto          | 68           | Degrees Fahrenheit | HEAT-SP      |
| 30 | COOL TEMP SETPOINT   | Auto          | 76           | Degrees Fahrenheit | COOL-SP      |
| 31 | DAMPER DRIVE TIME    | Auto          | 300          | Seconds            | DRV-TIME     |
| 32 | DAMPER POSITION      | Auto          | 5.57551      | Percent %          | DMPR-POS     |

**Program 1-SINGLE DUCT VAV**

```

5 REM Balancer has control of box by program 4
6 IF TIME-ON( PRG4 ) > 0:30:00 THEN STOP PRG4
8 IF PRG4 THEN END

10 IF HEATING THEN ACTIVESP = HEAT-SP ELSE ACTIVESP = COOL-SP
20 IF HEATING THEN A = 1 - CON1 / 100 ELSE A = CON1 / 100
30 RQSTFLOW = ( ( MAX-FLOW - MIN-FLOW ) * A + MIN-FLOW ) /
VOLFACTR
40 ACT-FLOW = FLOWSNSR * VAR22 : REM ACTUAL FLOW CORRECTED (FPM)
50 VOLUME1 = ACT-FLOW * VOLFACTR : REM LIVE VOLUME READOUT -
UNITS MATCH VOLUME
60 IF CON2 > 70 THEN START C , STOP B
70 IF CON2 < 30 THEN START B , STOP C
80 IF CON2 > 35 AND CON2 < 65 THEN STOP B , STOP C
81 X = 1 / SCANS
82 IF CON2 < 99 AND CON2 > 1 THEN 83 ELSE 100
83 IF CON2 < 36 OR CON2 > 64 THEN START D ELSE STOP D
90 IF TIME-ON( D ) > 0:04:00 THEN GOTO 100

91 Y = X + Y : IF Y > 5 THEN Y = 0
92 Z = ABS( CON2 - 50 ) / 50
95 IF Y > Z THEN STOP B , STOP C

100 IF B THEN MOTOR = 10
110 IF C THEN MOTOR = 0
120 IF NOT B AND NOT C THEN MOTOR = 5

140 IF MOTOR < 2 THEN DMPR-POS = DMPR-POS - X * 100 / DRV-TIME
150 IF MOTOR > 8 THEN DMPR-POS = DMPR-POS + X * 100 / DRV-TIME
160 IF DMPR-POS > 100 THEN DMPR-POS = 100
170 IF DMPR-POS < 0 THEN DMPR-POS = 0
180 END

```

**Program 2-TIME PROPORTIONAL HT**

```

20 REM ----- TIME PROPORTIONAL REHEAT SEQUENCE
-----
25 REM -- Designed to drive a solid state relay connected to
electric reheat.
26 REM Time base is fixed at 10 seconds - ie, a call for 50%
heat turns on
27 REM relay (OUT1) for 5 seconds, off for 5 seconds.
28 REM Uses CON3 for control of reheat operation! ( Full PID
function )
30 IF NOT REHEAT THEN STOP OUT1 , END
40 K = 1 / SCANS
50 L = L + K : IF L > 10 THEN L = 0
60 IF L < CON3 / 10 THEN START OUT1 ELSE STOP OUT1
70 END

```

**Program 3-Three-STAGE REHEAT**

```

10 REM ----- 3-STAGE REHEAT SEQUENCE
-----
20 REM -- STAGES ON AT 30, 60, 90 % OF PROPORTIONAL BAND BELOW
SETPOINT.
30 REM Auxillary flow setpoint is used only when AUX-FLOW
(VAR28) is not
40 REM zero and anytime stage 1 heat (OUT1) is requested.
50 REM Entire reheat sequence is easily disabled by toggling
VAR27 on-off.
60 IF NOT REHEAT THEN STOP OUT1 , STOP OUT2 , STOP OUT3 , STOP A
, END
70 IF CON3 > 30 THEN START OUT1 , START A
80 IF CON3 > 60 THEN START OUT2
90 IF CON3 > 90 THEN START OUT3
100 IF CON3 < 80 THEN STOP OUT3
110 IF CON3 < 50 THEN STOP OUT2
120 IF CON3 < 20 THEN STOP OUT1 , STOP A
130 IF AUX-FLOW = 0 THEN 150 : REM Since Auxillary flow is zero
its not used
140 IF A THEN RQSTFLOW = AUX-FLOW / VOLFACTR
150 END

```

**Program 4-BALANCER OVERRIDE**

```

10 REM **** Balancer control of unit to either MIN or MAX flow
   ***
20 REM ** Uses var25 activesp in digital fashion to select
   desired flow **

25 IF+ ACTIVESP > 1 THEN ACTIVESP = 1
27 IF+ ACTIVESP < 0 THEN ACTIVESP = 0
30 IF ACTIVESP THEN RQSTFLOW = MAX-FLOW / VOLFACTR ELSE RQSTFLOW
   = MIN-FLOW / VOLFACTR
40 ACT-FLOW = FLOWSNSR * VAR22 : REM ACTUAL FLOW CORRECTED (FPM)
50 VOLUME1 = ACT-FLOW * VOLFACTR : REM LIVE VOLUME READOUT -
   UNITS MATCH VOLUME

70 IF CON2 > 70 THEN START C
90 IF CON2 < 30 THEN START B
100 IF CON2 > 35 AND CON2 < 65 THEN STOP B , STOP C
110 IF+ B THEN STOP C
120 IF+ C THEN STOP B

125 X = 1 / SCANS
130 IF CON2 < 99 AND CON2 > 1 THEN 140 ELSE 210
140 IF CON2 < 36 OR CON2 > 64 THEN START D ELSE STOP D
150 IF TIME-ON( D ) > 0:04:00 THEN 210
170 Y = X + Y
180 IF Y > 5 THEN Y = 0
190 Z = ABS( CON2 - 50 ) / 50
200 IF Y > Z THEN STOP B , STOP C

210 IF B THEN MOTOR = 10
220 IF C THEN MOTOR = 0
230 IF NOT B AND NOT C THEN MOTOR = 5

270 IF MOTOR < 3 THEN DMPR-POS = DMPR-POS - X * 100 / DRV-TIME
280 IF MOTOR > 7 THEN DMPR-POS = DMPR-POS + X * 100 / DRV-TIME

```

Schematics

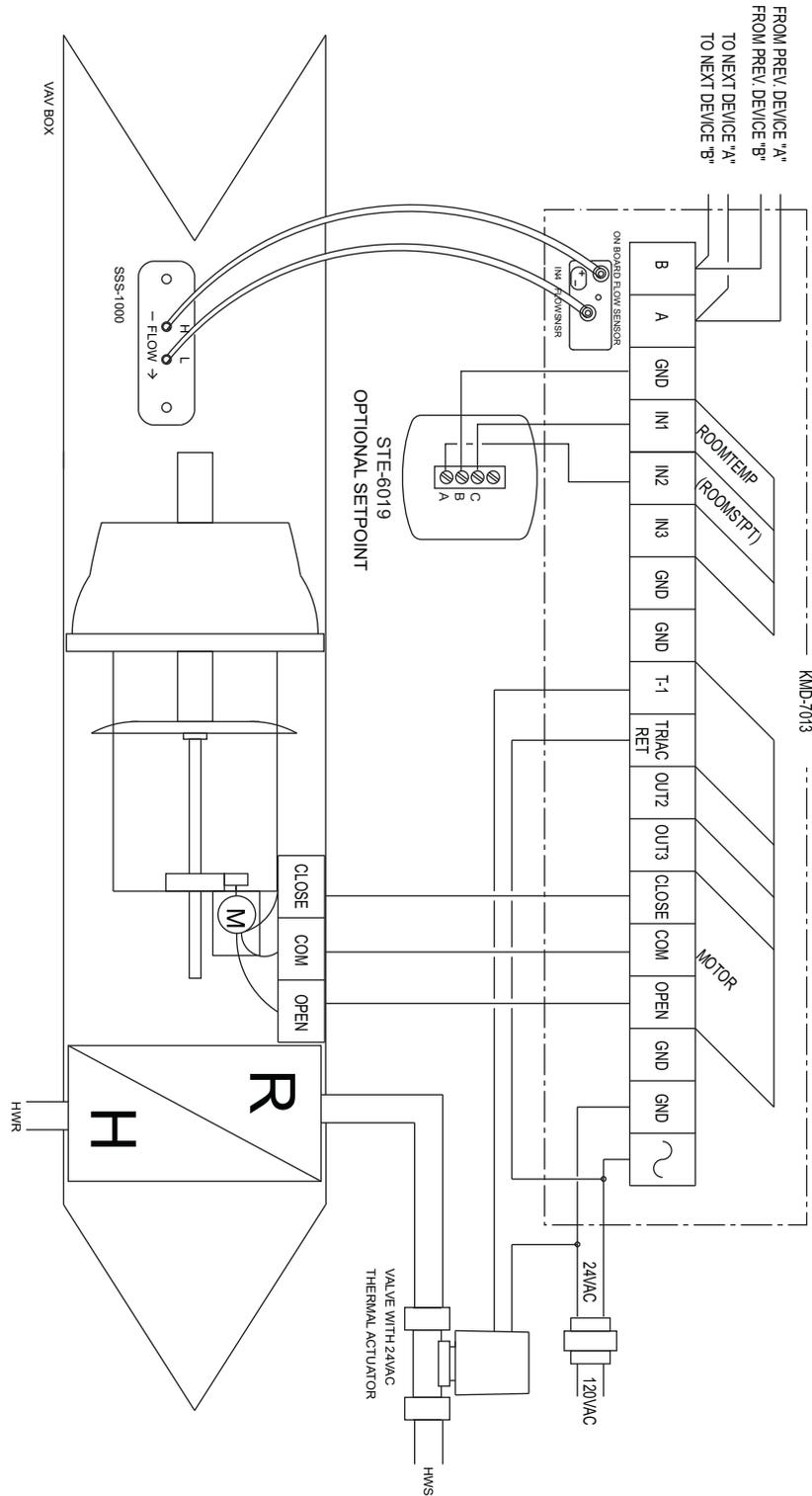
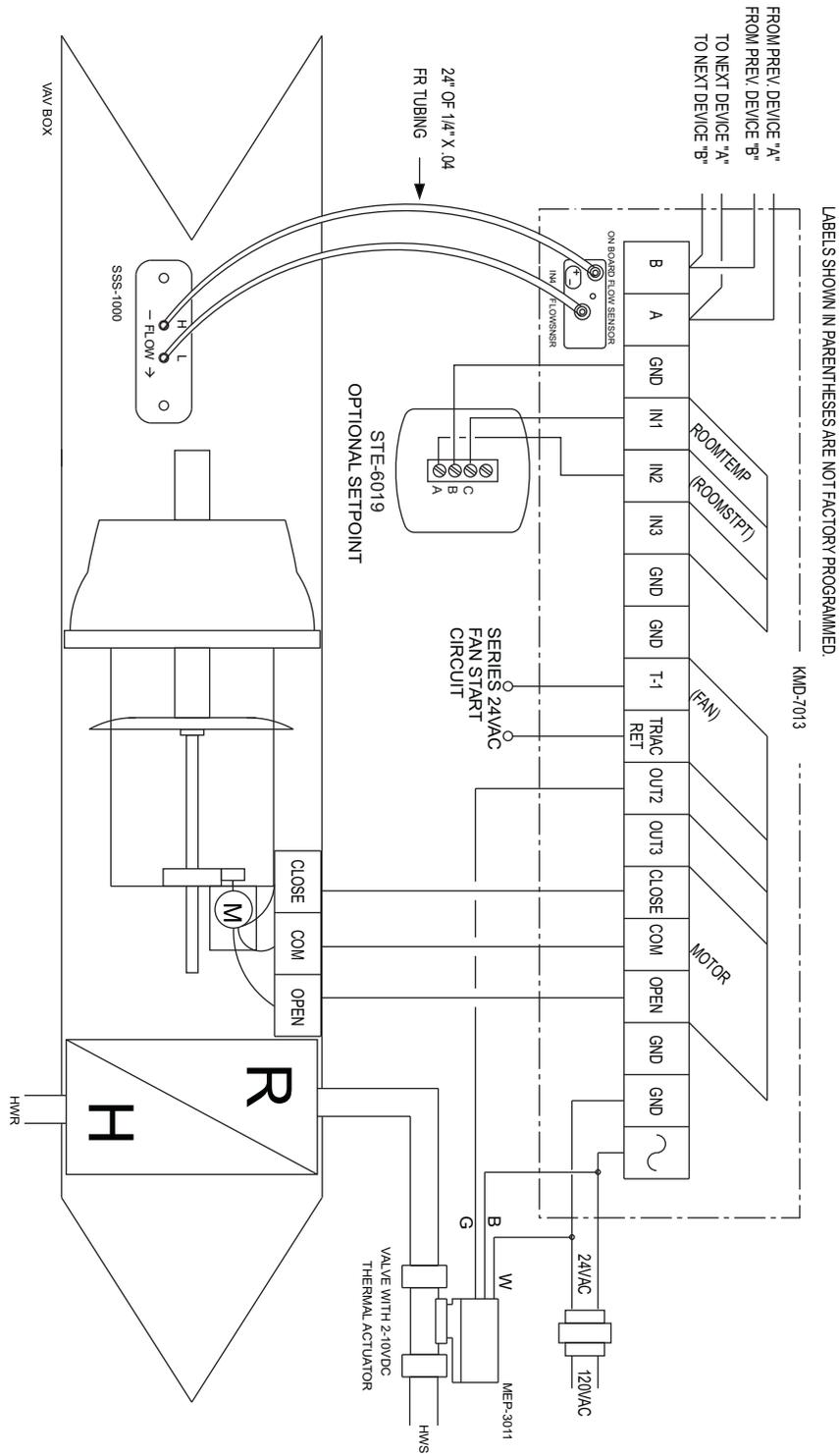
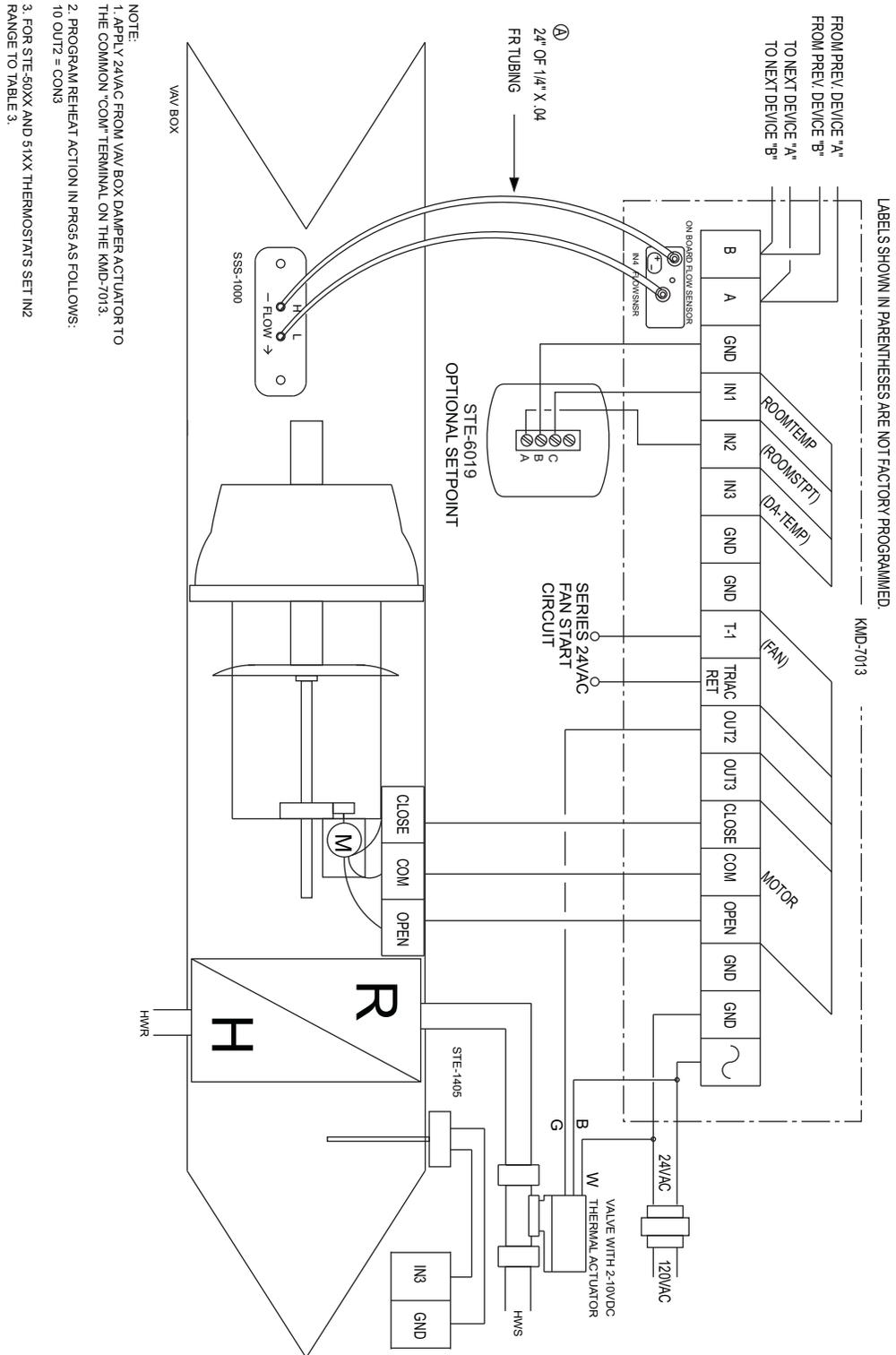


Illustration 9 KMD-7013 with screw-drive actuator and wet reheat



**Illustration 10 KMD-7013 with screw-drive actuator , wet reheat and fan control**



- NOTE:
1. APPLY 24VAC FROM VAV BOX DAMPER ACTUATOR TO THE COMMON "COM" TERMINAL ON THE KMD-7013.
  2. PROGRAM REHEAT ACTION IN PRG5 AS FOLLOWS:  
10 OUT2 = COM3
  3. FOR STE-60XX AND 51XX THERMOSTATS SET IN2 RANGE TO TABLE 3.

Illustration 11 KMD-7013 with screw-drive actuator, wet reheat, fan control and DAT sensor



## Application Note–AN0404F

---

### **KMD–7101/02 Fan Coil Unit Controller**

This application note describes methods and programming to use the KMD–7101 and KMD–7102 with hot and chilled water coils and multi speed fans.

---

#### **Related materials**

In addition to the material presented in this installation and operation guide, review and have available the following reference materials.

- ◆ Installation and operation guides for KMD–71010/7102 and STE–5015-10
- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide
- ◆ System plans with controller addresses

#### **Introduction**

The KMD-7101 and KMD-7102 Fan Coil Controllers are four input/ four output controllers pre-programmed to operate typical Fan Coil Units. Model KMD-7101 includes three, 30 amp mechanical relays for staging fan speeds and two, 1-amp triac relays for operating heating and cooling valves. Model KMD-7102 includes the same output configuration as the KMD-7101 with an additional 30 amp mechanical relay to operate an Auxiliary Heating Element.

Several different control routines have been programmed into these controllers for options such as Time-Proportional (modulating) Valve Control or On-Off Valve Operation, Automatic or Manual Fan Speed Control, and on Model KMD-7102 an optional Auxiliary Heating Sequence.

The primary sequence of operation compares the Room Temperature (IN1) to the Space Setpoint (IN2) to operate the appropriate valve (modulating or ON/OFF) and Fan speed to maintain the space temperature. If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory configurations allowing you to program any desired sequence.

This feature allows you to custom tailor the control sequence to match specific applications. The following paragraphs explain the pre-programmed control sequence in detail and offer suggestions for user-programming to customize operation of the controllers.

## Hardware Configuration Manager (HCM)

For networked operation, use the HCM software to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 1 Default Configuration for KMD-7101**

|                     |           |
|---------------------|-----------|
| • Panel Number      | 124       |
| • Network Baud Rate | 38400     |
| • Last Panel        | Unchecked |
| • Panel Type        | 120       |

## Setting up with WinControl XL

This section covers the set-up requirements to activate any of the pre-programmed options available in this device. At this point, communications should have been established with the unit.

A “Master” password is available to “sign-on” to each controller. Other security codes can be entered to control access to the system.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence.

The first step after signing on to the controller is to “Load Descriptors One” or “Load Descriptors All”. Many of the descriptions have been pre-defined for Inputs, Outputs, and Variables to aid in configuring the options.

To use a program or programs for a specific application, go to the “Control Basic” programs screen and turn “ON” the appropriate program.

**Configuring inputs**

All Inputs can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:

| # | Description       | Manual                   | Value | Units            | Decom                               | Label   |
|---|-------------------|--------------------------|-------|------------------|-------------------------------------|---------|
| 1 | SPACE TEMPERATURE | <input type="checkbox"/> | 75.00 | Deg. F           | <input checked="" type="checkbox"/> | SPACETP |
| 2 | SPACE SETPOINT    | <input type="checkbox"/> | 74.00 | Unused (Table 3) | <input type="checkbox"/>            | SPACESP |
| 3 |                   | <input type="checkbox"/> | 0.00  | Unused           | <input type="checkbox"/>            |         |
| 4 |                   | <input type="checkbox"/> | 0.00  | Unused           | <input type="checkbox"/>            |         |

**Illustration 1 Input configuration in WinControl XL**

**Input 1-SPACE TEMPERATURE** Default selects a KMC 10K ohm thermistor as used in all of KMC's STE series sensors.

**Input 2-SPACE SETPOINT** Default selects Table 3 which has been setup for the potentiometer used on KMC's STE-5000 series thermostats.

**Input 3** Unused

**Input 4** Unused

**Configuring outputs**

The KMD-7101/02 Controllers features 30 amp, 240 VAC, mechanical relays and 1 amp, 24 VAC triacs directly onto the board assembly. The KMC supplied control routines have been written to use the following output configuration:

| # | Description        | Manual                   | Value  | Units  | Label    |
|---|--------------------|--------------------------|--------|--------|----------|
| 1 | 3-STAGE FAN SPEED  | <input type="checkbox"/> | 100.00 | %      | FANSPEED |
| 2 | AUXILIARY HEATING  | <input type="checkbox"/> | Off    | Off/On | AUX-HEAT |
| 3 | HEAT VALVE CONTROL | <input type="checkbox"/> | Off    | Off/On | HEATVALV |
| 4 | COOL VALVE CONTROL | <input type="checkbox"/> | On     | Off/On | COOLVALV |

**Illustration 2 Output configuration in WinControl XL**

**Output 1-3-STAGE FAN SPEED** (0-100%) This output is set up as an analog range with limits of 0 and 7.5 volts (0-100% is 0 to 7.5 volts). Although the range is analog this signal is used to “stage” each of three relays ON at 2.5, 5.0, and 7.5 volts for each of the fan speeds LOW, MED, and HIGH. The settings on this output should not be changed to allow full operation of the fan speed relays as described.

**Output 2-AUXILIARY HEATING (Off/On)** This output is dedicated to the activation of the fourth 30 amp mechanical relay used on the KMD-7102. This relay is not included on the KMD-7101 in order to minimize cost and therefore, this output is not active on that model.

**Output 3-HEAT VALVE CONTROL** This output activates the onboard triac relay typically connected to a heating valve such as KMC's VEP series. The KMC supplied programming can be selected to Time-Proportion (Modulate) this output or simply turn it ON/OFF as required.

**Output 4-COOL VALVE CONTROL** This output activates the onboard triac relay typically connected to a cooling valve such as KMC's VEP series. The KMC supplied programming can be selected to Time-Proportion (Modulate) this output or simply turn it ON/OFF as required.

### Configuring variables

A total of thirty-two system variables are available to perform various functions such as set points, limits, and modes. All variables can be changed or modified to suit the specific application. The following have been pre-defined for use in the KMD-7101/02 KMC supplied control sequences:

**Variable 23-OCCUPANCY MODE** This variable can be toggled to indicate occupied / unoccupied status by user programming as desired. Default mode is occupied. The state must be determined by the user.

**Variable 24-CHANGEVER MODE (COOL/HEAT)** The KMC supplied control sequence chooses whether the unit will operate in heating or cooling by automatically setting the thermostat action as direct (COOL) or reverse (HEAT) based on the space requirements.

**Variable 25-HEAT SETBACK SETTING** In an unoccupied mode (VAR23 = Unocc), the Active Heat Setpoint (VAR26) becomes this temperature; The thermostat set point is ignored.

**Variable 26-ACTIVE HEAT SETPOINT** This is the temperature to be maintained in the space in the HEAT mode (VAR24 = HEAT).

**Variable 27-ACTIVE COOL SETPOINT** This is the temperature to be maintained in the space in the COOL mode (VAR24 = COOL).

**Variable 28-COOL SETBACK SETTING** In an unoccupied mode (VAR23 = Unocc), the Active Cool Setpoint (VAR27) becomes this temperature; The thermostat set point is ignored.

**Variable 29-THERMOSTAT DEADBAND** In an occupied mode (VAR23 = Occupy), the Active Heat Setpoint is the Space Setpoint (IN2) minus half this value and the Active Cool Setpoint is the Space Setpoint plus half this value. In the unoccupied mode the thermostat set point is not used and this value has no effect on the operation of the unit.

**Variable 30 - AUXILIARY HEAT** In order for the KMC supplied sequence to actually turn on the Auxiliary Heat relay this variable must first be enabled. This option makes it easy to disable electric heat operation under certain circumstances.

### Configuring controllers (PID control loops)

KMC digital products feature built-in, full function PID (Proportional, Integral, Derivative) controllers. On the KMD-7101/02 Controller 1 is a reverse-acting HEATING thermostat. Controller 2 is a direct-acting COOLING thermostat. All parameters may be modified as necessary.

**Control Basic programming**

Program areas 1 through 4 have specific KMC supplied programming burned into the controller at the factory. If none of sequence fits the required application, simply edit the programs or send blank programs to the controller and replace any of the factory programs. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.

**! Caution**

This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type)

The following explanations are provided for the KMC programs in the KMD-7101/02:

| # | Description          | On                                  | Manual                   | Exit | Label |
|---|----------------------|-------------------------------------|--------------------------|------|-------|
| 1 | TIME PROP. VALVES    | <input checked="" type="checkbox"/> | <input type="checkbox"/> | No   |       |
| 2 | ON OFF VALVES        | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 3 | MANUAL FAN SPEED CRL | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 4 | AUXILIARY HEAT CTRL  | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 5 | USER PROGRAMMING     | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |

Buttons: Auto Load, Edit, Erase, OK, Cancel  
 Panel: A1      Status: Idle

**Illustration 3 Control Basic program list**

**Program 1–TIME PROPORTIONAL VALVES** This is the default program. The following operations are included in this sequence:

- Determine Active Set Points Based on Occupancy Status

In the Occupied mode (VAR23 = ON) the heating set point (VAR26) is calculated by subtracting 1/2 of the deadband value (VAR29) from the set point indicated on the thermostat dial (IN2). Similarly, the cooling set point (VAR27) is found by adding 1/2 the deadband to the set point.

In the Unoccupied mode (VAR23 = OFF) the heating set point (VAR26) becomes the heating setback temperature (VAR25) and the cooling set point (VAR27) becomes the cooling setback (VAR28).

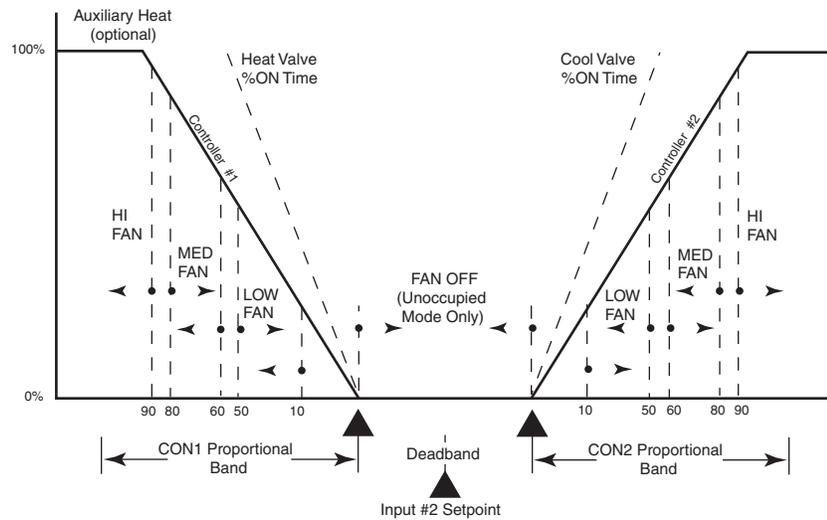
**Note**

User programming will be required to change occupancy modes. The KMC supplied sequence simply defaults to an Occupied mode on initial start-up.

**Change-over from Heating Mode to Cooling Mode** When the space temperature (IN1) falls below the active heating set point (VAR26) the heating mode is enabled. When the temperature rises above the Active Cooling Set Point (VAR27) the cooling mode is initiated (VAR24 = OFF)

**Valve Operation** A ten second time-based routine is used to modulate the appropriate valve from 0 to 100% as heating or cooling demand rises. The valve is modulated over the first half of the proportional band as defined in Controllers. This method ensures the valve is at 100% capacity before staging on additional fan speeds.

**Automatic Fan Speed Control** This sequence will automatically step the three different fan speeds as heating or cooling load requirements for the space change. Note that in the occupied mode (VAR24 = ON) the fan is not allowed to turn off (LOW, MEDIUM, and HIGH only). While in the Unoccupied Mode, the fan will turn off when no demand for heating or cooling exists. A differential between ON and OFF prevents cycling between fan speeds during small temperature fluctuations.



**Illustration 4 Time Proportional (Modulating) Valves-Auto Fan Speed (PRG1)**



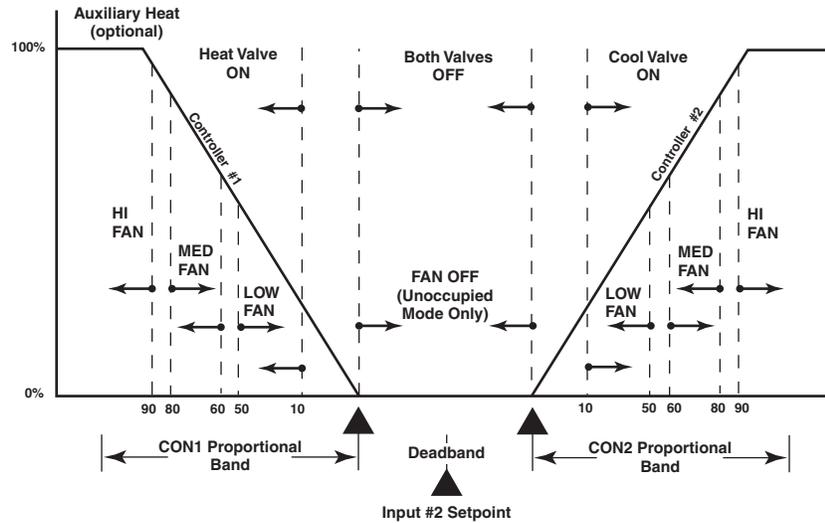
**Note**

Default operation is always for automatic control of the fan speeds. For manual fan speed operation, see the description for Program 3 below.

**Program 2 - ON-OFF VALVES**

This optional control sequence is identical to Program 1 except valve operation is assumed to be ON-OFF control rather than time proportional modulating operation. The same set points determination, heat-cool changeover and fan speed control. Valve operation is to turn "ON" the appropriate valve when the

load requirement exceeds 15% as determined by the PID controllers. Valves are turned off when load requirements fall below 1%. The built-in deferential prevents rapid cycling of the valves.



**Illustration 5 ON-OFF Valve Operation - Auto Fan Speed**



**Note**

Program 1 should be turned OFF (RUN = NO) if Program 2 will be used. Likewise, Program 2 should be left off when Program 1 operation is desired.

**Program 3—MANUAL FAN SPEED CONTROL** Utilization of this program will override the standard automatic fan speed control. This routine assumes a momentary Push-button is across the Room Temperature Sensor (IN1). When the button on the thermostat is pressed, it momentarily shorts the thermistor. Each time the button is pressed the fan speed will increment as follows: Occupied Mode (LO-MED-HI-MED-LO) or Unoccupied mode (OFF-LO-MED-HI-MED-LO-OFF). A feature has also been added that resets the fan speed to its lowest possible speed whenever occupancy (VAR23) changes state. For example, if occupancy is changed from occupied to unoccupied status, the fan will be automatically turned OFF. Pressing the button on the thermostat will again sequence the fan speed according to the above mentioned sequences. Remember, this panel will default to an occupied status on initial start-up. To control this status, a line of “user” programming will be required.

**Program 4 - AUXILIARY HEAT CONTROL** This routine is only valid for use on Model KMD-7102 Controllers (since the KMD-7101 does not include the output 2 relay). The purpose of this sequence is to turn on a secondary heat source if and when the heating demand reaches 100% load requirement. This will occur if the space is more than the proportional band temperature away from set point. An Auxiliary Heat enable/disable Variable (VAR30) has been included to allow easily locking-out the auxiliary heating. This Variable will default to a disabled status upon initial start-up and must be turned ON in the Variables screen or through user programming before the Auxiliary heat will operate. As a safety, if the fan speed is commanded OFF (set to zero) the Auxiliary Heat will also be turned OFF.



**Note**

The fan speed is not proof that the fan is actually running - only that the KMD system has commanded it to run. For installations requiring positive verification of airflow before Auxiliary Heat is used an airflow switch or other device must be used (two inputs are available for this and other uses).

**Program 5 - Available for user programming**

**Configuring tables**

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). ). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7101:

**Table1** Available for user programming.

**Table 2** Correlates the dial setting of STE-5200 Series thermostats to degrees F.

**Table 3** Correlates the slider position of STE-5000 Series thermostats to degrees F.



**Note**

These tables use nominal values for correlation. Due to component tolerances and the resistance of field wiring, these values may need correction on an individual basis. If the value read is too low, enter a positive calibration in the input column labeled “cal”. If the value read is high, a negative calibration can be entered. If necessary, reconfigure Table 2 and/or Table 3. The complete Table 2 and Table 3 from the factory are shown. If the tables have been reconfigured and the user wants the factory configuration for Table 2 and Table 3 it will be necessary to use Clear Panel or re-enter the values.



**Caution**

Using Clear Panel will erase any programming changes that have been made to the controller.

(Clear Panel does not affect changes made with HCM, or panel type.)

| #  | Table 1 | Unused | Table 2 | Unused | Table 3 | Unused |
|----|---------|--------|---------|--------|---------|--------|
| 1  | 0       | 0      | 0       | 86     | 0       | 54     |
| 2  | 0       | 0      | 0.71    | 85     | 0.05    | 54     |
| 3  | 0       | 0      | 1.07    | 80     | 0.538   | 58     |
| 4  | 0       | 0      | 1.38    | 75     | 0.939   | 62     |
| 5  | 0       | 0      | 1.67    | 70     | 1.274   | 66     |
| 6  | 0       | 0      | 1.91    | 65     | 1.421   | 68     |
| 7  | 0       | 0      | 2.1     | 60     | 1.558   | 70     |
| 8  | 0       | 0      | 2.23    | 56     | 1.684   | 72     |
| 9  | 0       | 0      | 3       | 55     | 1.801   | 74     |
| 10 | 0       | 0      | 0       | 0      | 1.911   | 76     |
| 11 | 0       | 0      | 0       | 0      | 2.013   | 78     |
| 12 | 0       | 0      | 0       | 0      | 2.198   | 82     |
| 13 | 0       | 0      | 0       | 0      | 2.362   | 86     |
| 14 | 0       | 0      | 0       | 0      | 2.5     | 90     |
| 15 | 0       | 0      | 0       | 0      | 3       | 90     |

Illustration 6

### Occupancy status

For simplicity, a system variable (VAR23) has already been defined as an “occupied status” indication. The KMC supplied routines do not define what constitutes an “occupied” status. However, the KMC routines do reference the status of this variable when determining room set points or for outside air damper operations. Typically, the occupancy status is determined by reference to a time schedule by writing a line of code in Program 5 such as:

```
10 IF WS1 THEN START VAR23 ELSE STOP VAR23
```

Where WS1 is weekly schedule 1 as defined in the KMD-7101/02 panel. Scheduled times are assigned with up to four ON and four OFF times per day, for each day of the week. See the KMC digital Operator's Manual for details on setting up schedules.

### Temperature setpoints

This model has been pre-programmed assuming a set point potentiometer is connected to Input 2. In some cases, it may be desirable to provide a “fixed” set point in the software to free-up the input for another purpose (or have a standardized set point for the whole building). The KMC supplied sequence determines the heating and cooling set points in an occupied mode as half the deadband below and above IN2 set point respectively. To change this to a fixed set point simply write a line of code in Program 5 such as:

```
70 IF OCCUPIED THEN HTG.SP = 72 , CLG.SP = 74
```

The numbers 72 and 74 can be substituted by different numbers or reference variables. Also note the KMC sequence defines the unoccupied heating and cooling set points by referencing "setback" variables. These too can be modified if necessary, or the cooling and heating set points can be set directly as shown below:

```
80 IF NOT OCCUPIED THEN HTG.SP = 66 , CLG.SP = 76
```

## Using the NetSensor

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7101, simply change the PID controllers 1 and 2 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor. Change both SPACESP in line 40 of program1 and program 2 to a variable mapped to button 2 on the netsensor.

## Other options

One of the biggest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the KMC supplied sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated. The WinControl XL User's Manual provides help and examples for programming Control Basic.

**WinControl settings** The following tables display the WinControl XL Plus settings for this application note.

**Table 2 Input configuration**

| Description         | Manual | Value   | Units                | Average | Label   |
|---------------------|--------|---------|----------------------|---------|---------|
| 1 SPACE TEMPERATURE | Auto   | 237.447 | Deg.F KMC10K Type II | 64      | SPACETP |
| 2 SPACE SETPOINT    | Auto   | 90      | Table 3              | 64      | SPACESP |
| 3                   | Auto   | 0       | Unused               | 64      |         |
| 4                   | Auto   | 0       | Unused               | 64      |         |

**Table 3 Output configuration**

| Description          | Manual | Output | Units  | 0% | 100% | Delay | Security | Label    |
|----------------------|--------|--------|--------|----|------|-------|----------|----------|
| 1 3-STAGE FAN SPEED  | Auto   | 100    | 0-100% | 0  | 7.5  |       | 0        | FANSPEED |
| 2 AUXILIARY HEATING  | Auto   | Off    | Off/On |    |      | 0:00  | 0        | AUX-HEAT |
| 3 HEAT VALVE CONTROL | Auto   | Off    | Off/On |    |      | 0:00  | 0        | HEATVALV |
| 4 COOL VALVE CONTROL | Auto   | On     | Off/On |    |      | 0:00  | 0        | COOLVALV |

**Table 4 Variables**

| Description                      | Manual | Value    | Units              | Label    |
|----------------------------------|--------|----------|--------------------|----------|
| 1-22 Available for user programs |        |          |                    |          |
| 23 OCCUPANCY MODE                | Auto   | Occupied | Unocc/Occ          | OCC.MODE |
| 24 CHANGEOVER MODE               | Auto   | Cool     | Cool/Heat          | HTG.MODE |
| 25 HEAT SETBACK SETTING          | Auto   | 60       | Degrees Fahrenheit | HTGSETBK |
| 26 ACTIVE HEAT SETPOINT          | Auto   | 88       | Degrees Fahrenheit | HTG.SP   |
| 27 ACTIVE COOL SETPOINT          | Auto   | 92       | Degrees Fahrenheit | CLG.SP   |
| 28 COOL SETBACK SETTING          | Auto   | 80       | Degrees Fahrenheit | CLGSETBK |
| 29 THERMOSTAT DEADBAND           | Auto   | 4        | Degrees Fahrenheit | DEADBAND |
| 30 AUXILIARY HEAT                | Auto   | Disabled | Dis/Enabled        | HT-ENABL |
| 31                               | Auto   | 0        |                    |          |
| 32                               | Auto   | 0        |                    |          |

**Table 5 Controllers (PID loops)**

| CON | Point | A/M  | Output | Setpoint | Action | Prop. | Reset | H/M  | Rate | Bias |
|-----|-------|------|--------|----------|--------|-------|-------|------|------|------|
| #1  | IN1   | Auto | 0%     | VAR26    | RA(-)  | 5     | 0     | Hour | 0    | 0    |
| #2  | IN1   | Auto | 100%   | VAR27    | DA(+)  | 5     | 0     | Hour | 0    | 0    |
| #3  |       | Auto | 0%     |          | DA(+)  | 0     | 0     | Hour | 0    | 0    |
| #4  |       | Auto | 0%     |          | DA(+)  | 0     | 0     | Hour | 0    | 0    |

**Table 6 Tables**

|    | <b>Table 1</b> | <b>Unused</b> | <b>Table 2</b> | <b>Unused</b> | <b>Table 3</b> | <b>Unused</b> |
|----|----------------|---------------|----------------|---------------|----------------|---------------|
| 1  | 0              | 0             | 0              | 86            | 0              | 54            |
| 2  | 0              | 0             | 0.71           | 85            | 0.05           | 54            |
| 3  | 0              | 0             | 1.07           | 80            | 0.538          | 58            |
| 4  | 0              | 0             | 1.38           | 75            | 0.939          | 62            |
| 5  | 0              | 0             | 1.67           | 70            | 1.274          | 66            |
| 6  | 0              | 0             | 1.91           | 65            | 1.421          | 68            |
| 7  | 0              | 0             | 2.1            | 60            | 1.558          | 70            |
| 8  | 0              | 0             | 2.23           | 56            | 1.684          | 72            |
| 9  | 0              | 0             | 3              | 55            | 1.801          | 74            |
| 10 | 0              | 0             | 0              | 0             | 1.911          | 76            |
| 11 | 0              | 0             | 0              | 0             | 2.013          | 78            |
| 12 | 0              | 0             | 0              | 0             | 2.198          | 82            |
| 13 | 0              | 0             | 0              | 0             | 2.362          | 86            |
| 14 | 0              | 0             | 0              | 0             | 2.5            | 90            |
| 15 | 0              | 0             | 0              | 0             | 3              | 90            |

**Table 7 Control Basic program list**

|   | <b>Description</b>   | <b>On</b> | <b>Manual</b> | <b>Label</b> |
|---|----------------------|-----------|---------------|--------------|
| 1 | TIME PROP. VALVES    | YES       | Auto          |              |
| 2 | ON OFF VALVES        | NO        | Auto          |              |
| 3 | MANUAL FAN SPEED CRL | NO        | Auto          |              |
| 4 | AUXILIARY HEAT CTRL  | NO        | Auto          |              |
| 5 | USER PROGRAMMING     | NO        | Auto          |              |

**Program 1-TIME PROP. VALVES**

```

30 REM DETERMINING ACTIVE SETPOINTS BASED ON OCCUPANCY MODE
40 IF OCCUPANCY~MODE THEN ACTIVE~HEAT~SETPOINT = SPACE~SETPOINT
  - THERMOSTAT~DEADBAND / 2 , ACTIVE~COOL~SETPOINT =
  SPACE~SETPOINT + THERMOSTAT~DEADBAND / 2
50 IF NOT OCCUPANCY~MODE THEN ACTIVE~HEAT~SETPOINT =
  HEAT~SETBACK~SETTING , ACTIVE~COOL~SETPOINT =
  COOL~SETBACK~SETTING
60 REM -- DETERMINE CHANGEOVER --
70 IF SPACE~TEMPERATURE < ACTIVE~HEAT~SETPOINT THEN START
  CHANGEOVER~MODE
80 IF SPACE~TEMPERATURE > ACTIVE~COOL~SETPOINT THEN STOP
  CHANGEOVER~MODE
85 REM Time-proportional control 10 second time base
86 REM valve opens 0-100% over 1/2 of proportional band in
  controllers
90 A = 1 / SCANS
100 B = B + A : IF B > 10 THEN B = 0
110 IF CHANGEOVER~MODE THEN 120 ELSE 170
120 REM ** HEATING MODE OPERATION **
130 STOP COOL~VALVE~CONTROL
150 IF B < CON1 / 5 THEN START HEAT~VALVE~CONTROL ELSE STOP
  HEAT~VALVE~CONTROL
160 GOTO 220
170 REM ** COOLING MODE OPERATION **
180 STOP HEAT~VALVE~CONTROL
200 IF B < CON2 / 5 THEN START COOL~VALVE~CONTROL ELSE STOP
  COOL~VALVE~CONTROL
210 REM Automatic FAN SPEED control
220 IF CHANGEOVER~MODE AND CON1 > 10 OR NOT CHANGEOVER~MODE AND
  CON2 > 10 THEN START C
230 IF CHANGEOVER~MODE AND CON1 > 60 OR NOT CHANGEOVER~MODE AND
  CON2 > 60 THEN START D
240 IF CHANGEOVER~MODE AND CON1 > 90 OR NOT CHANGEOVER~MODE AND
  CON2 > 90 THEN START E
250 IF CHANGEOVER~MODE AND CON1 < 80 OR NOT CHANGEOVER~MODE AND
  CON2 < 80 THEN STOP E
260 IF CHANGEOVER~MODE AND CON1 < 50 OR NOT CHANGEOVER~MODE AND
  CON2 < 50 THEN STOP D
270 IF CHANGEOVER~MODE AND CON1 < 1 OR NOT CHANGEOVER~MODE AND
  CON2 < 1 THEN STOP C
280 IF C THEN 3-STAGE~FAN~SPEED = 33
290 IF D THEN 3-STAGE~FAN~SPEED = 67
300 IF E THEN 3-STAGE~FAN~SPEED = 100
310 IF NOT C AND NOT D AND NOT E THEN 320 ELSE 330
320 IF NOT OCCUPANCY~MODE THEN 3-STAGE~FAN~SPEED = 0 ELSE
  3-STAGE~FAN~SPEED = 33
330 END

```

### Program 2-ON OFF VALVES

```
30 REM DETERMINING ACTIVE SETPOINTS BASED ON OCCUPANCY MODE
40 IF OCCUPANCY~MODE THEN ACTIVE~HEAT~SETPOINT = SPACE~SETPOINT
- THERMOSTAT~DEADBAND / 2 , ACTIVE~COOL~SETPOINT =
SPACE~SETPOINT + THERMOSTAT~DEADBAND / 2
50 IF NOT OCCUPANCY~MODE THEN ACTIVE~HEAT~SETPOINT =
HEAT~SETBACK~SETTING , ACTIVE~COOL~SETPOINT =
COOL~SETBACK~SETTING
60 REM -- DETERMINE CHANGEOVER MODE --
70 IF SPACE~TEMPERATURE < ACTIVE~HEAT~SETPOINT THEN START
CHANGEOVER~MODE
80 IF SPACE~TEMPERATURE > ACTIVE~COOL~SETPOINT THEN STOP
CHANGEOVER~MODE
90 REM CONTROL FOR ON-OFF VALVES
100 IF CHANGEOVER~MODE THEN 110 ELSE 160
110 REM ** HEATING MODE OPERATION **
120 STOP COOL~VALVE~CONTROL
130 IF CON1 > 15 THEN START HEAT~VALVE~CONTROL
140 IF CON1 < 1 THEN STOP HEAT~VALVE~CONTROL
150 GOTO 210
160 REM ** COOLING MODE OPERATION **
170 STOP HEAT~VALVE~CONTROL
180 IF CON2 > 15 THEN START COOL~VALVE~CONTROL
190 IF CON2 < 1 THEN STOP COOL~VALVE~CONTROL
200 REM Automatic FAN SPEED control
210 IF CHANGEOVER~MODE AND CON1 > 15 OR NOT CHANGEOVER~MODE AND
CON2 > 15 THEN START A
220 IF CHANGEOVER~MODE AND CON1 > 60 OR NOT CHANGEOVER~MODE AND
CON2 > 60 THEN START B
230 IF CHANGEOVER~MODE AND CON1 > 90 OR NOT CHANGEOVER~MODE AND
CON2 > 90 THEN START C
240 IF CHANGEOVER~MODE AND CON1 < 80 OR NOT CHANGEOVER~MODE AND
CON2 < 80 THEN STOP C
250 IF CHANGEOVER~MODE AND CON1 < 50 OR NOT CHANGEOVER~MODE AND
CON2 < 50 THEN STOP B
260 IF CHANGEOVER~MODE AND CON1 < 1 OR NOT CHANGEOVER~MODE AND
CON2 < 1 THEN STOP A
270 IF A THEN 3-STAGE~FAN~SPEED = 33
280 IF B THEN 3-STAGE~FAN~SPEED = 67
290 IF C THEN 3-STAGE~FAN~SPEED = 100
300 IF NOT A AND NOT B AND NOT C THEN 310 ELSE 320
310 IF NOT OCCUPANCY~MODE THEN 3-STAGE~FAN~SPEED = 0 ELSE
3-STAGE~FAN~SPEED = 33
320 END
```

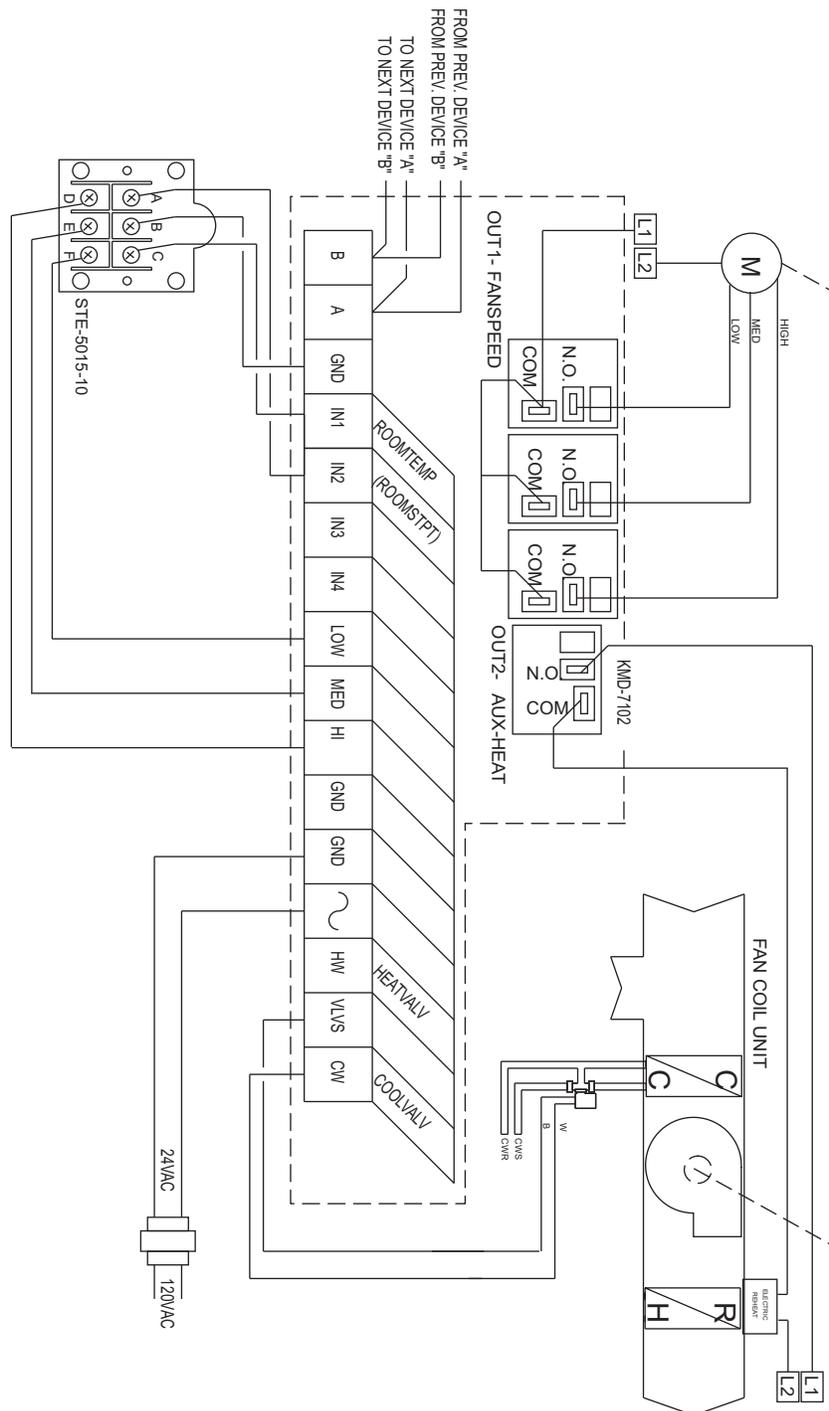
**Program 3-MANUAL FAN SPEED CRL**

```
30 REM ** Manual FAN SPEED routine overrides automatic
operation **
40 REM FAN resets to LOW (occupied) or OFF (unoccupied) whenever
50 REM Occupancy changes states or after power-loss!
60 IF+ OCCUPANCY~MODE THEN Z = 1
70 IF- OCCUPANCY~MODE THEN Z = 0
80 IF+ SENSOR-ON( SPACE~TEMPERATURE ) THEN 90 ELSE 110
90 IF Y THEN Z = Z - 1 ELSE Z = Z + 1
100 IF Z < 1 THEN STOP Y
105 IF Z > 2 THEN START Y
110 IF OCCUPANCY~MODE AND Z < 1 THEN Z = 2
120 IF Z = 1 THEN 3-STAGE~FAN~SPEED = 33
130 IF Z = 2 THEN 3-STAGE~FAN~SPEED = 67
140 IF Z = 3 THEN 3-STAGE~FAN~SPEED = 100
150 IF Z = 0 THEN 3-STAGE~FAN~SPEED = 0
160 END
```

**Program 4-AUXILIARY HEAT CTRL**

```
25 REM ***** OUTPUT IS NOT AVAILABLE ON KMD-7101 MODELS !!!
*****
30 REM -- AUXILIARY HEATING CONTROL ROUTINE --
35 REM Only allows AUX Heating if VAR30 is enabled
40 IF NOT AUXILIARY~HEAT THEN STOP AUXILIARY~HEATING , END
50 IF CON1 > 99 THEN START AUXILIARY~HEATING
60 IF CON1 < 80 THEN STOP AUXILIARY~HEATING
63 REM Stops AUX Heat if fan is not ON
65 IF 3-STAGE~FAN~SPEED = 0 THEN STOP AUXILIARY~HEATING
70 END
```





**Illustration 8 FCU with three-speed fan and electric reheat**



## Application Note–AN0404G

---

Revision B

### **KMD–7301 AHU Controller**

This application note describes methods and programming to use the KMD–7301 controller to operate an air handler.

---

#### **Related materials**

In addition to the material presented in this installation and operation guide, review and have available the following reference materials.

- ◆ Installation and operation guides for KMD–7301, MEP–1272 and STE–5000 series or STE–5200 series sensors.
- ◆ Installation and operation guides for CTE–3017 and STE–1400 series sensors.
- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide
- ◆ System plans with controller addresses

#### **Introduction**

The KMD-7301 Air Handling Unit (AHU) Controller is a 4-input/ 4-output DDC Controller primarily designed for operating small air handling units. This device has been pre-programmed with a KMC supplied control sequence to operate a fan, outside air damper, and heating/ cooling valves as necessary to maintain space setpoint temperature. An optional economizer sequence is included as well as coil freeze protection overrides to further complement system operations.

In addition to the basic control sequence, this device will determine “active” and “setback” temperature setpoints as well as primary outside damper positioning based on occupancy status.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory configurations allowing you to program any desired sequence.

This feature allows you to custom tailor the control sequence to match specific applications. The following paragraphs explain the pre-programmed control sequence in detail and offer suggestions for user-programming to customize operation of the controllers.

## Hardware Configuration Manager (HCM)

For networked operation, use the HCM software to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 8 Default configuration for KMD-7301**

|                    |           |
|--------------------|-----------|
| Panel Number       | 124       |
| Network Baud Rate  | 38400     |
| Last Panel         | No        |
| Phone compensation | Unchecked |
| Panel Type         | 122       |

## Setting up with WinControl

This section covers the set-up requirements to activate any of the preprogrammed options available in this device. At this point, communications should have been established with the unit.

A "Master" password is available to "sign-on" to each controller. Other security codes can be entered to control access to the system.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence.

The first step after signing on to the controller is to "Load Descriptors One" or "Load Descriptors All". Many of the descriptions have been pre-defined for Inputs, Outputs, and Variables to aid in configuring the options.

To use a program or programs for a specific application, go to the "Control Basic" programs screen and turn "ON" the appropriate program.

### Configuring inputs

All inputs on the KMD Controllers can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:

| # | Description          | Manual                   | Value  | Units            | Decom.                              | Label     |
|---|----------------------|--------------------------|--------|------------------|-------------------------------------|-----------|
| 1 | SPACE TEMPERATURE    | <input type="checkbox"/> | -16.03 | Deg. F           | <input checked="" type="checkbox"/> | SPACE\$TP |
| 2 | SETPOINT TEMPERATURE | <input type="checkbox"/> | 55.00  | Unused (Table 2) | <input checked="" type="checkbox"/> | SETPT\$TP |
| 3 | MIXED AIR TEMP       | <input type="checkbox"/> | 112.07 | Deg. F           | <input checked="" type="checkbox"/> | MIXED\$TP |
| 4 |                      | <input type="checkbox"/> | 0.00   | Unused           | <input type="checkbox"/>            |           |

**Illustration 9 Input configuration with WinControl XL**

**Input 1-SPACE TEMPERATURE** Default selects a KMC 10k ohm thermistor as used in all of KMC's STE Series sensors.

**Input 2-SETPOINT TEMPERATURE** Default range selects Table 2 which has been pre-defined to correlate slider position of the STE-5200 Series Thermostats into degrees F.

**Input 3-MIXED AIR TEMPERATURE** Default selects standard KMC 10K ohm thermistor in degrees F. This input is required for operation of the outside air damper with optional "economizer" and "Freeze protection" programming routines.

**Input 4** Unused

### Configuring outputs

The KMD-7301 Air Handling Unit Controller features one on-board triac relay (Digital only) and three universal outputs (analog or digital, set in software). The KMC supplied control routines have been written to use the following output configuration:

| # | Description    | Manual                   | Value  | Units  | Label     |
|---|----------------|--------------------------|--------|--------|-----------|
| 1 | FAN            | <input type="checkbox"/> | On     | Off/On | FAN       |
| 2 | COOLING VALVE  | <input type="checkbox"/> | 0.00   | %      | COOL\$VLV |
| 3 | HEATING VALVE  | <input type="checkbox"/> | 100.00 | %      | HEAT\$VLV |
| 4 | OUTSIDE DAMPER | <input type="checkbox"/> | 0.00   | %      | OUT\$DMPR |

**Illustration 10 Output configuration with WinControl XL**

**Output 1-FAN** This output uses a 1 amp, zero crossing triac with optical isolation. It is intended to switch a 24 VAC starter circuit to energize the fan motor.

**Output 2-COOLING VALVE** Default settings select 0 to 100% range equaling 0 to 10 VDC output from the KMD Controller.

**Output 3-HEATING VALVE** Default settings select 0 to 100% range equaling 0 to 10 VDC output from the KMD Controller.

**Output 4-OUTSIDE DAMPER** This output follows the same configuration as Outputs 2 and 3 to modulate an outdoor damper actuator from 0 to 10 volts ( 0 - 100% open ).

### Configuring variables

A total of thirty-two system variables are available to perform various functions such as setpoints, limits, and flags. The following have been pre-defined for use in the KMC supplied control sequences:

**Variable 22-OCCUPANCY MODE** This variable can be toggled to indicate occupied / unoccupied status by user programming. The Default is OCCUPIED. The state must be determined by the user.

**Variable 23-ECONOMIZER OPERATION** This variable can be easily toggled to enable or disable operation of the economizer control sequence. The Default is DISABLED The state must be determined by the user.

**Variable 24-FREEZE PROTECTION RESET** Optional Program 3 trips this variable into an alarm condition it mixed air temperature drops below the lower limit defined in Variable 31. Once tripped, this variable will need to be manually toggled back to a normal condition. See the explanation of Program 3 Freeze Protection for details.

**Variable 25 and 28-ACTIVE TEMPERATURE SETPOINTS** These are the temperature setpoints for cooling and heating that the unit will maintain. In an unoccupied condition these values become the appropriate "setback" temperature settings. In occupied modes these values are calculated based on Input 2 "SETPOINT TEMPERATURE" and Variable 27 "THERMOSTAT DEADBAND"

**Variable 26 and 29-SETBACK TEMPERATURE SETPOINTS** For cooling and heating, respectively. These are fixed values entered by the user.

**Variable 27-THERMOSTAT DEADBAND** Our method of determining setpoints for cooling and heating assures a neutral range or "deadband", exists between setpoints.

**Variable 30-MIXED AIRFLOW LIMIT** This variable establishes a desired mixed air temperature for operation of the outside air damper in an economizer sequence.

**Variable 31-FREEZE PROTECTION LOW LIMIT** When Program 3 for coil freeze protection is enabled, this Variable establishes a minimum temperature for the mixed air sensor. If the temperature falls below this point, the unit will initiate a self-protection sequence. Refer to the explanation of Program 3 for details.

**Variable 32–MINIMUM DAMPER POSITION** This variable allows easy access to changing the minimum damper position in an occupied mode as requested by the control sequence. The Default value is 25% outside air. In the unoccupied mode the damper output goes to 0%.

**Configuring controllers (PID loops)**

KMC digital products feature built-in, full function PID (Proportional, Integral, Derivative) controllers. In the KMD-7301, Controller 1 defines a direct-acting “cooling” thermostat while Controller 2 is set up as a reverse-acting “heating” thermostat. Controller 3 compares mixed air temperature to Variable 30 “MIXED AIR LOW LIMIT” for operation of the outside air damper in the economizer cycle. All parameters may be modified as necessary.

**Control Basic programs**

Program areas 1 through 4 have specific KMC supplied programming burned into the controller at the factory. If none of sequence fits the required application, simply edit the programs or send blank programs to the controller and replace any of the factory programs. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.

**! Caution**

This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type)

| # | Description         | On                                  | Manual                   | Exit | Label |
|---|---------------------|-------------------------------------|--------------------------|------|-------|
| 1 | AIR HANDLER ROUTINE | <input checked="" type="checkbox"/> | <input type="checkbox"/> | No   |       |
| 2 | ECONOMIZER OPTION   | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 3 | FREEZE PROTECTION   | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 4 |                     | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 5 | USER PROGRAMMING    | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |

Buttons: Auto Load, Edit, Erase, OK, Cancel  
 Panel: A1      Status: Idle

**Illustration 11 Control Basic listing**

The following explanations are provided for the KMC supplied programs in the KMD-7301:

**Program 1–AIR HANDLER ROUTINE** The first part of this program defines the active temperature set-points for use by the control routine. In the occupied (VAR22 is on) mode, the heating setpoint is calculated by subtracting half of the deadband value (VAR27) from the setpoint indicated on Input 2 (Thermostat dial). Similarly, the cooling setpoint is found by adding half of the deadband to the Input 2 value.

In the unoccupied mode (VAR22 is off) the cooling and heating setpoints simply assume the “setback” values defined in Variables 26 and 29 respectively.



**Note**

“User” programming will be required to change occupancy modes. The KMC supplied sequence simply defaults to an occupied mode on initial start-up.

Basic control of the outside air damper is to open the damper to minimum position (VAR32) in the occupied mode and to fully close the damper in an unoccupied mode. Additional damper control is available using the optional economizer or freeze protection programs detailed below.

Actual modulation of the heating and cooling valves (0 to 100%) is accomplished using the PID Controllers. These controllers compare room temperature (IN1) to the appropriate setpoint and generate a signal from 0 to 100%. The output point converts the 0 - 100% signal into a 0 to 10 volt signal to drive the valves.

The fan is called to operate at all times in an occupied mode. Additionally, if the mode is unoccupied and the controllers call for either heating or cooling to maintain setback temperatures the fan will also operate until the condition has been satisfied. A built-in differential prevents cycling the fan on and off due to minor temperature variations.

**Program 2–ECONOMIZER OPTION** If this program is set to “run” and the economizer cycle is enabled (VAR23), the outside air damper will be positioned according to Controller 3, but not less than the minimum damper position specified in Variable 32. The Controller 3 compares mixed air temperature to a pre-defined mixed air low limit as defined in Variable 30. If the mixed air temperature falls below the low limit, damper position will be modulated closed to attempt to maintain that temperature.

**Program 3–FREEZE PROTECTION** This optional sequence can be used to override operation of the air handler in an attempt to prevent coils from freezing. Basically, once the mixed air temperature, as sensed by Input 3, falls below the freeze protection limit temperature (defined in Variable 31 – typically 40 degrees) a variable (VAR24) is tripped into an alarm condition. If this variable is in alarm, the cooling valve is forced to 0%, the heating valve is opened to 100%, the fan is stopped, and the outside air damper will be commanded to 0% position (completely closed to the outside). An alarm will also be generated in the system at this time.



**Note**

Once VAR24 is tripped into an alarm condition, it must be manually reset or user programming can be written to automatically reset. For example:

```
10 IF MIXED$TP > FREEZE-UP + 5 THEN STOP RESET-FZ
```

The “+5” in the statement above provides a differential so mixed air temperature has to rise more than 5 degrees above the freeze-up temperature limit before resetting the alarm condition.

**Caution**

Use of an automatic reset should be considered carefully. If the mixed air temperature is getting below 40 there may be a physical problem that requires attention such as the outside dampers sticking open. Although the temperature in the mixed air chamber may rise to a safe level when the fan is off, the problem may still exist as does the risk of freezing the coil.

**Program 5—Available for user programming****Using custom tables**

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7301:

**Table 1** Ready for user programming.

**Table 2** Correlates slider position of STE-5200 Series Thermostat into degrees F.

**Table 3** Correlates the dial setting of STE-5000 Series Thermostat Sensor to degrees F.

**Note**

These tables use nominal values for correlation. Due to component tolerances and the resistance added due to wiring these values may need correction on an individual basis. If the value read is too low, enter a positive calibration in the input column labeled “cal”. If the value read is high, a negative calibration can be entered. If necessary, reconfigure Table 2 and /or Table 3. The complete Table 2 and Table 3 from the factory are shown. If the tables have been reconfigure and the user wants the factory configuration for Table 2 and Table 3 it will be necessary to use Clear Panel or re-enter the values.

**Caution**

Using Clear Panel will erase any programming changes that have been made to the controller.

(Clear Panel does not affect changes made with HCM, or panel type.)

| # | Table 1 | Unused | Table 2 | Unused | Table 3 | Unused |
|---|---------|--------|---------|--------|---------|--------|
| 1 | 0       | 0      | 0       | 0      | 0       | 0      |
| 2 | 0       | 0      | 0.14    | 0      | 0.35    | 0      |
| 3 | 0       | 0      | 0.2     | 75     | 0.46    | 276    |
| 4 | 0       | 0      | 0.4     | 327    | 0.86    | 566    |
| 5 | 0       | 0      | 0.8     | 563    | 1.1     | 7      |
| 6 | 0       | 0      | 1       | 664    | 1.7     | 112    |
| 7 | 0       | 0      | 1.5     | 904    | 2.3     | 15     |
| 8 | 0       | 0      | 2       | 1143   | 2.5     | 1648   |

**Illustration 12 Custom table**

## Occupancy status

For simplicity, a system variable (VAR22) has already been defined as an “occupied status” indication. The KMC supplied routines do not define what constitutes an “occupied” status. However, the KMC supplied routines do reference the status of this variable when determining room setpoints or for outside air damper operations. Typically, the occupancy status is determined by reference to a time schedule by writing a line of code such as:

```
10 IF WS1 THEN START VAR22 ELSE STOP VAR22
```

Where WS1 is weekly schedule 1 as defined in the panel.

## Temperature setpoints

This model has been pre-programmed assuming a setpoint potentiometer is connected to Input 2. In some cases, it may be desirable to provide a “fixed” setpoint in the software to free-up the input for another purpose (or have a standardized setpoint for the whole building). The KMC supplied sequence determines the heating and cooling setpoints in an occupied mode as half the deadband below and above IN2 setpoint respectively. To change this to a fixed setpoint simply write a line of code in Program 5 such as:

```
20 IF OCCUPIED THEN HEAT$SPT = 68 , COOL$SPT = 74
```

The numbers 68 and 74 can be substituted by different numbers or reference variables. Also note the KMC supplied sequence defines the unoccupied heating and cooling setpoints by referencing “setback” variables. These too can be modified if necessary.

## Economizer

If you choose to use the KMC supplied economizer routine it will be necessary to determine when an economizer cycle can be allowed. An efficient method to do this is to compare outside and inside enthalpy. To enable or disable operation of

the economizer is then based on where the air handler can get the least expensive cooling. If the enthalpy is lower outside the economizer is enabled. If the enthalpy is lower inside the economizer is disabled.

Here is an example of programming to make this comparison:

```
10 IF OUT-ENTH < IN-ENTH -1 THEN ENABLE VAR23
20 IF OUT-ENTH > IN-ENTH THEN DISABLE VAR23
```

A more simplistic approach to enabling the economizer is to look at outside air temperature. The following example looks at outside air and the cooling demand based on the controller output to the cooling valve:

```
10 IF CON1 > 5 AND OAT < 65 THEN ENABLE VAR23
20 IF CON1 < 1 AND OAT > 67 THEN DISABLE VAR23
```

This example requires a cooling demand (CON1) of greater than 5% and an outside air temperature (OAT) of less than 65 degrees before economizer operation is allowed. Once enabled, the economizer cycle will be disabled if either the demand for cooling falls off or the outside air temperature rises above 67 degrees.

## Using the NetSensor

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7301, simply change the PID controllers 1 and 2 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor. Change SETPT\$TP in line 10 and line 20 of program1 to a variable mapped to button 2 on the NetSensor.

## Other options

One of the biggest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the KMC supplied sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated. The KMC digital Technical Operator's Manual provides help and examples for programming Control Basic.

**WinControl settings** The following tables display the WinControl XL Plus settings for this application note.

**Table 9 Input**

|   | Description          | Manual | Value | Units                | Average | Label     |
|---|----------------------|--------|-------|----------------------|---------|-----------|
| 1 | SPACE TEMPERATURE    | Auto   | 72    | Deg.F KMC10K Type II | 64      | SPACE\$TP |
| 2 | SETPOINT TEMPERATURE | Auto   | 72    | Table 2              | 64      | SETPT\$TP |
| 3 | MIXED AIR TEMP       | Auto   | 55    | Deg.F KMC10K Type II | 64      | MIXED\$TP |
| 4 |                      | Auto   | 0     | Unused               | 64      |           |

**Table 10 Output**

|   | Description    | Man<br>ual | Value | Units  | 0%  | 100% | Delay | Security | Label     |
|---|----------------|------------|-------|--------|-----|------|-------|----------|-----------|
| 1 | FAN            | Auto       | On    | Off/On |     |      |       | 0        | FAN       |
| 2 | COOLING VALVE  | Auto       | 0     | 0-100  | 0.0 | 10   |       | 0        | COOL\$VLV |
| 3 | HEATING VALVE  | Auto       | 100   | 0-100  | 0.0 | 10   |       | 0        | HEAT\$VLV |
| 4 | OUTSIDE DAMPER | Auto       | 0     | 0-100  | 0.0 | 10   |       | 0        | OUT\$DMPR |

**Table 11 Controllers (PID control loops)**

|    | Point | Manual | Output | Setpoint | Action | Prop. | Reset | H/M  | Rate | Bias |
|----|-------|--------|--------|----------|--------|-------|-------|------|------|------|
| #1 | IN1   | Auto   | 0%     | VAR25    | DA(+)  | 3     | 0     | Hour | 0    | 0    |
| #2 | IN1   | Auto   | 100%   | VAR28    | RA(-)  | 3     | 0     | Hour | 0    | 0    |
| #3 | IN3   | Auto   | 100%   | VAR30    | DA(+)  | 5     | 0     | Hour | 0    | 0    |
| #4 |       | Auto   | 0%     |          | DA(+)  | 0     | 0     | Hour | 0    | 0    |

**Table 12 Variables**

| VAR  | Description                 | Manual | Value    | Units              | Label     |
|------|-----------------------------|--------|----------|--------------------|-----------|
| 1-21 | Available for user programs |        | 0        |                    |           |
| 22   | OCCUPANCY MODE              | Auto   | Occupied | Unocc/Occ          | OCCUPIED  |
| 23   | ECONOMIZER OPERATION        | Auto   | Disabled | Dis/Enabled        | ECONOMIZ  |
| 24   | FREEZE PROTECT RESET        | Auto   | Normal   | Normal/Alarm       | RESET-FZ  |
| 25   | ACTIVE COOLING SETPT        | Auto   | 57       | Degrees Fahrenheit | COOL\$SPT |
| 26   | SETBACK COOL SETPT          | Auto   | 0        | Degrees Fahrenheit | C\$SETBK  |
| 27   | T'STAT DEADBAND             | Auto   | 4        | Degrees Fahrenheit | DEADBAND  |
| 28   | ACTIVE HEATING SETPT        | Auto   | 53       | Degrees Fahrenheit | HEAT\$SPT |
| 29   | SETBACK HEAT SETPT          | Auto   | 62       | Degrees Fahrenheit | H\$SETBK  |
| 30   | MIXED AIR LOW LIMIT         | Auto   | 50       | Degrees Fahrenheit | LOWLIMIT  |
| 31   | FREEZE PROTECT LIMIT        | Auto   | 40       | Degrees Fahrenheit | FREEZ-UP  |
| 32   | MIN. DAMPER POSITION        | Auto   | 0        | Percent %          | MIN\$DMPR |

**Table 13 Tables**

|    | <b>Table 1</b> | <b>Unused</b> | <b>Table 2</b> | <b>Unused</b> | <b>Table 3</b> | <b>Unused</b> |
|----|----------------|---------------|----------------|---------------|----------------|---------------|
| 1  | 0              | 0             | 0              | 86            | 2.56E-09       | 54            |
| 2  | 0              | 0             | 0.71           | 85            | 0.05           | 54            |
| 3  | 0              | 0             | 1.07           | 80            | 0.54           | 58            |
| 4  | 0              | 0             | 1.38           | 75            | 0.94           | 62            |
| 5  | 0              | 0             | 1.67           | 70            | 1.27           | 66            |
| 6  | 0              | 0             | 1.91           | 65            | 1.42           | 68            |
| 7  | 0              | 0             | 2.1            | 60            | 1.56           | 70            |
| 8  | 0              | 0             | 2.23           | 56            | 1.68           | 72            |
| 9  | 0              | 0             | 3              | 55            | 1.8            | 74            |
| 10 | 0              | 0             | 0              | 0             | 1.91           | 76            |
| 11 | 0              | 0             | 0              | 2.56E-09      | 2.01           | 78            |
| 12 | 0              | 0             | 0              | 0             | 2.2            | 82            |
| 13 | 0              | 0             | 3.16E-06       | 4.52E-39      | 2.36           | 86            |
| 14 | 0              | 0             | 0              | 3.16E-06      | 2.5            | 90            |
| 15 | 0              | 0             | 0              | 0             | 3              | 90            |

**Table 14 Control Basic program list**

|    | <b>Description</b>  | <b>On</b> | <b>A/M</b> | <b>Label</b> |
|----|---------------------|-----------|------------|--------------|
| #1 | AIR HANDLER ROUTINE | YES       | Auto       |              |
| #2 | ECONOMIZER OPTION   | NO        | Auto       |              |
| #3 | FREEZE PROTECTION   | NO        | Auto       |              |
| #4 |                     | NO        | Auto       |              |
| #5 | USER PROGRAMMING    | NO        | Auto       |              |

### Program 1–Air Handler Routine

```
10 IF OCCUPANCY~MODE THEN ACTIVE~COOLING~SETPT =  
SETPOINT~TEMPERATURE + T'STAT~DEADBAND / 2  
20 IF OCCUPANCY~MODE THEN ACTIVE~HEATING~SETPT =  
SETPOINT~TEMPERATURE - T'STAT~DEADBAND / 2  
30 IF NOT OCCUPANCY~MODE THEN ACTIVE~COOLING~SETPT =  
SETBACK~COOL~SETPT  
40 IF NOT OCCUPANCY~MODE THEN ACTIVE~HEATING~SETPT =  
SETBACK~HEAT~SETPT  
50 REM ** outside damper operation **  
60 IF OCCUPANCY~MODE THEN OUTSIDE~DAMPER = MIN.~DAMPER~POSITION  
ELSE OUTSIDE~DAMPER = 0  
70 REM *** heating / cooling valve controls ***  
80 COOLING~VALVE = CON1  
90 HEATING~VALVE = CON2  
100 REM  
110 REM **** Typical FAN control ****  
120 IF OCCUPANCY~MODE OR COOLING~VALVE > 10 OR HEATING~VALVE >  
10 THEN START FAN  
130 IF NOT OCCUPANCY~MODE AND COOLING~VALVE < 1 AND  
HEATING~VALVE < 1 THEN STOP FAN  
140 END
```

### Program 2–Economizer Option

```
10 REM * ECONOMIZER CYCLE *  
20 IF NOT ECONOMIZER~OPERATION THEN END  
25 IF NOT OCCUPANCY~MODE THEN END  
30 OUTSIDE~DAMPER = CON3  
40 IF OUTSIDE~DAMPER < MIN.~DAMPER~POSITION THEN OUTSIDE~DAMPER  
= MIN.~DAMPER~POSITION  
50 END
```

### Program 3–Freeze Protection

```
10 IF MIXED~AIR~TEMP < FREEZE~PROTECT~LIMIT THEN START  
FREEZE~PROTECT~RESET  
17 IF FREEZE~PROTECT~RESET THEN 20 ELSE 40  
20 COOLING~VALVE = 0 : REM Closes cooling valve  
25 HEATING~VALVE = 100 : REM Opens heating valve  
30 OUTSIDE~DAMPER = -1 : REM ASSURES OUT4 IS ZERO VOLTS (  
OVERRIDES LIMITS )  
35 STOP FAN  
40 DALARM FREEZE~PROTECT~RESET , 1 , WARNING! FREEZE PROTECTION  
SHUTDOWN !! RESET VAR24 !  
45 REM to restore operation must reset VAR24 to Normal status (  
variables )  
50 END
```

Application drawings

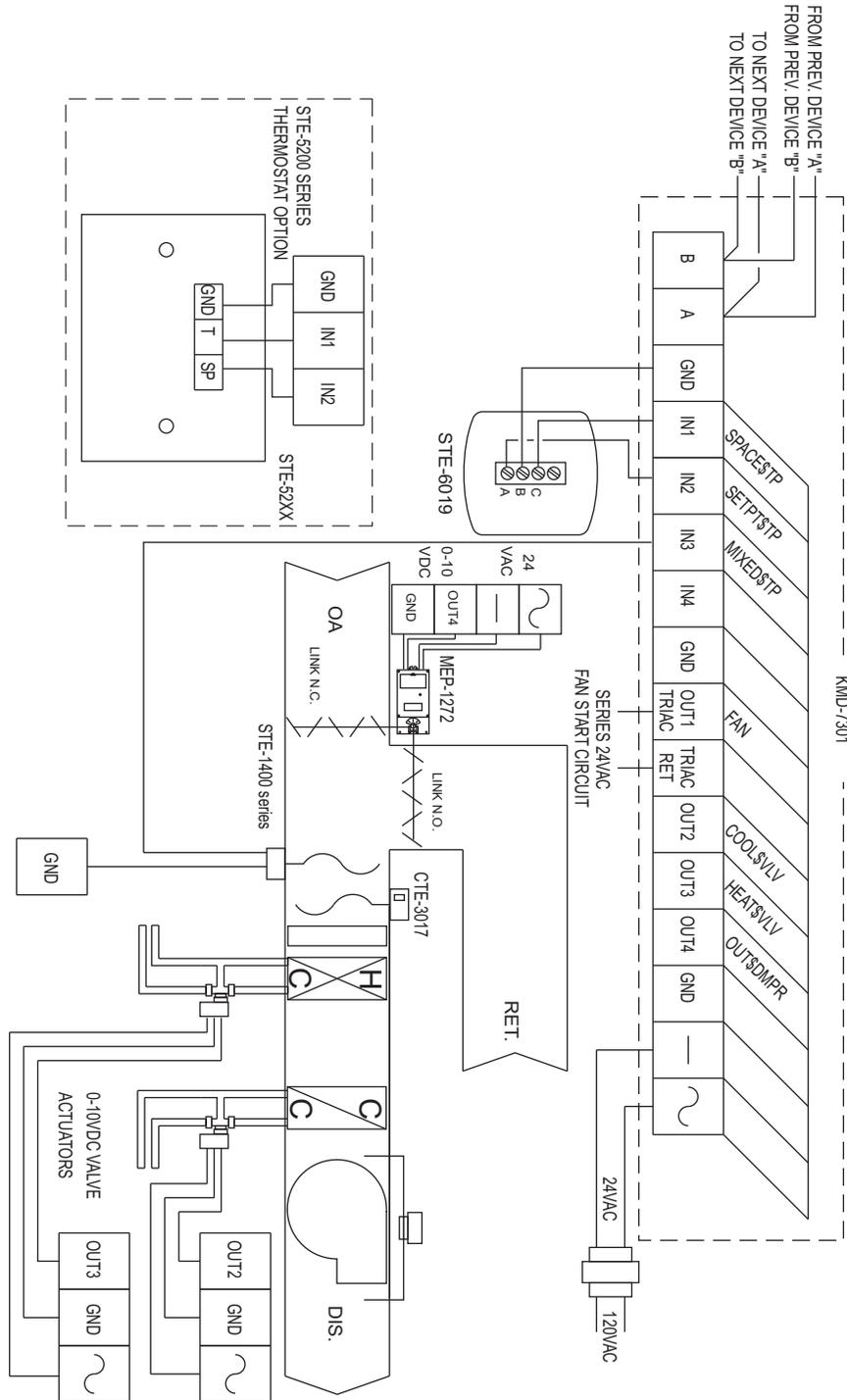


Illustration 13 AHU with hot and cold water valves and optional economizer

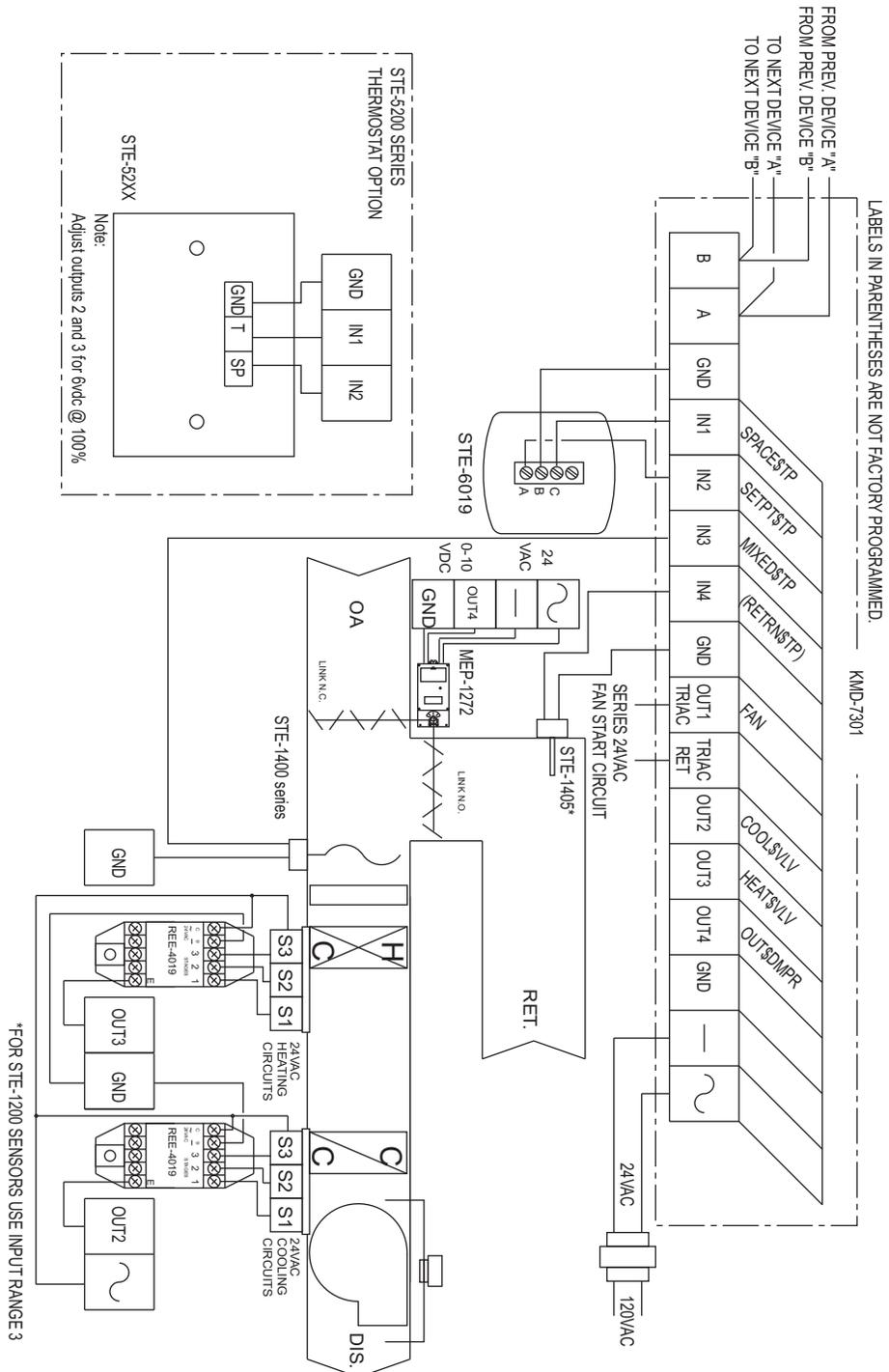
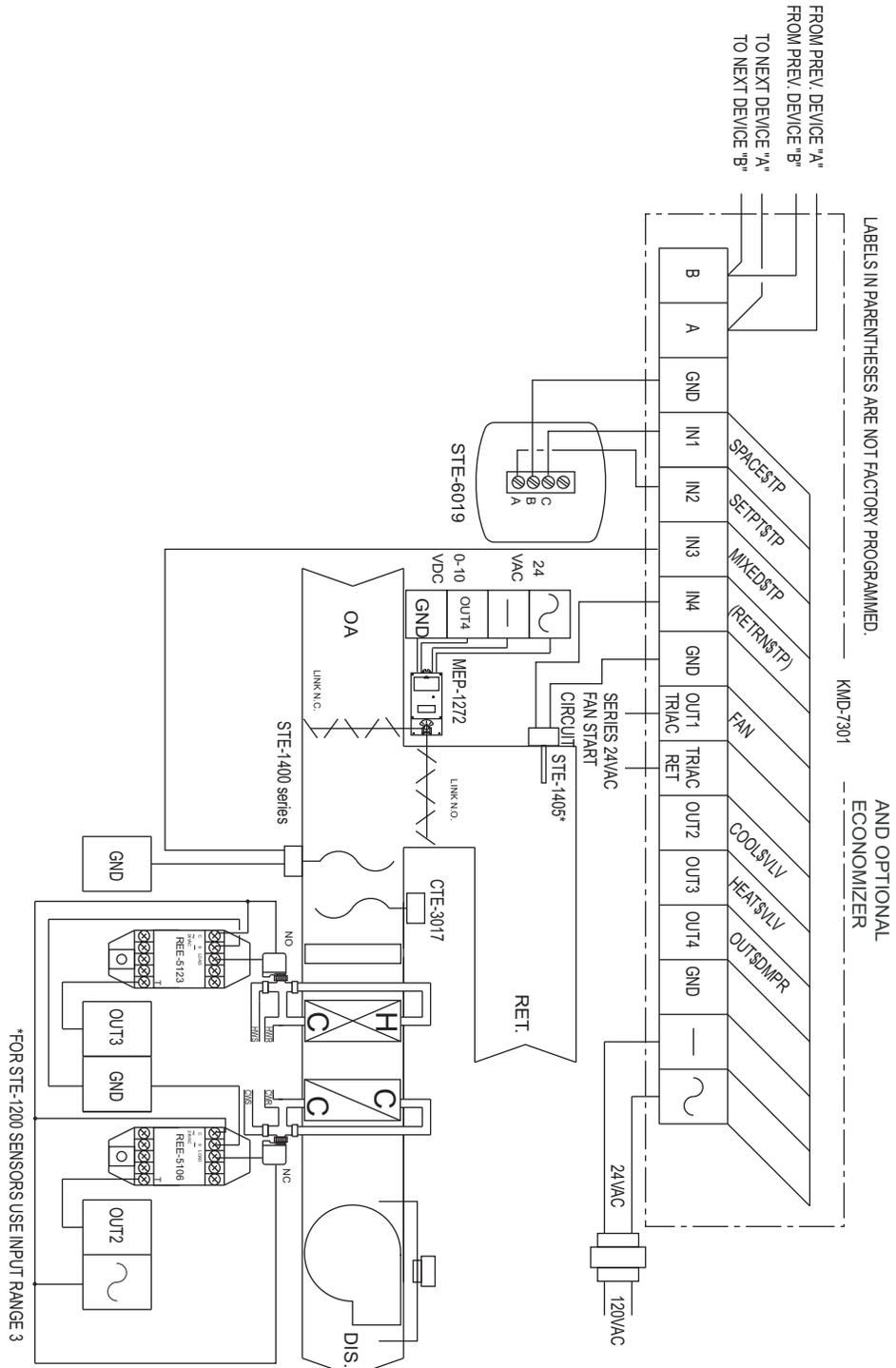


Illustration 14 AHU with three-stage heat and cool (10 volt valves) and optional economizer



**Illustration 15 AHU with three-stage heat and cool (24 volt valves) and optional economizer**



## Application Note–AN0404H

---

### **KMD–7302 Roof Top Unit Controller**

This application note describes methods and programming to use the KMD–7302 as a roof top unit controller with two-stage heating and cooling and includes optional economizer operation.

---

#### **Related materials**

In addition to the material presented in this installation and operation guide, review and have available the following reference materials.

- ◆ Installation and operation guides for KMD–7302, MEP–1222 and STE–5100 series or STE–5200 series sensors.
- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide
- ◆ System plans with controller addresses

#### **Introduction**

The KMD–7302 Roof Top Unit Controller is a four-input/four-output DDC Controller specifically designed for operating small air handling units with two-stage heating and cooling. This device has been pre-programmed with a KMC supplied control sequence to operate a fan, two stages of heating, and two stages of cooling to maintain the desired space temperature. An optional economizer sequence is included for units with modulating outside and return air dampers.

In addition to the basic control sequence, this device will determine “active” and “setback” temperature set points as well as primary outside damper positioning based on occupancy status.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory configurations allowing you to program any desired sequence.

This feature allows the user to custom tailor the control sequence to match specific applications. The following paragraphs explain the pre-programmed control sequence in detail and offer suggestions for user-programming to customize operation of the controllers.

## Hardware Configuration Manager (HCM)

For networked operation, use the HCM software to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 1 Default Configuration for KMD-7302**

|                    |           |
|--------------------|-----------|
| Panel Number       | 124       |
| Network Baud Rate  | 38400     |
| Last Panel         | Unchecked |
| Phone compensation | No        |
| Panel Type         | 18        |

## Setting up with WinControl

This section covers the set-up requirements to activate any of the pre-programmed options available in this device. At this point, communications should have been established with the unit.

A "Master" password is available to "sign-on" to each controller. Other security codes can be entered to control access to the system.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence.

The first step after signing on to the controller is to "Load Descriptors One" or "Load Descriptors All". Many of the descriptions have been pre-defined for Inputs, Outputs, and Variables to aid in configuring the options.

To use a program or programs for a specific application, go to the "Control Basic" programs screen and turn "ON" the appropriate program.

**Configuring inputs**

All inputs on the KMD Controllers can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:

| # | Description       | Manual                   | Value | Units            | Decom.                              | Label     |
|---|-------------------|--------------------------|-------|------------------|-------------------------------------|-----------|
| 1 | SPACE TEMPERATURE | <input type="checkbox"/> | 72    | Deg. F           | <input checked="" type="checkbox"/> | SPACE\$TP |
| 2 | BASE SET POINT    | <input type="checkbox"/> | 72    | Unused (Table 2) | <input type="checkbox"/>            | SETPT\$TP |
| 3 | MIXED AIR TEMP    | <input type="checkbox"/> | 55    | Deg. F           | <input checked="" type="checkbox"/> | MIXED\$TP |
| 4 |                   | <input type="checkbox"/> | 0.00  | Unused           | <input type="checkbox"/>            |           |

**Illustration 1 Input configuration in WinControl XL**

**Input 1–SPACE TEMPERATURE** Default selects a KMC 10k ohm thermistor as used in all of KMC's STE Series sensors.

**Input 2–BASE SET POINT** Default range selects Table 2 which has been pre-defined to correlate slider position of the STE-5200 Series Thermostats into degrees F. This input is used to determine the active set points for heating and cooling.

**Input 3–MIXED AIR TEMP** Default selects standard KMC 10K ohm thermistor in degrees F. This input is required for operation of the outside air damper with optional “economizer” programming routine. For STE-1200 series sensors, use range 2 or 3.

**Input 4 - Unused**

**Configuring outputs**

The KMD-7302 Roof Top Unit Controller features five on-board triac relays (Digital only) and one universal output ( analog or digital, set in software). The KMC supplied control routines have been written to use the following output configuration:

| # | Description          | Manual                   | Value | Units  | Label     |
|---|----------------------|--------------------------|-------|--------|-----------|
| 1 | FAN                  | <input type="checkbox"/> | On    | Off/On | FAN       |
| 2 | HEATING STAGE 1 & 2  | <input type="checkbox"/> | 0.00  | %      | HEAT\$STG |
| 3 | COOLING STAGE 1 & 2  | <input type="checkbox"/> | 0.00  | %      | COOL\$STG |
| 4 | OUTSIDE / RET DAMPER | <input type="checkbox"/> | 0.00  | %      | OUT\$DMPR |

**Illustration 2 Output configuration in WinControl XL**

**Output 1– FAN** This output uses a 1 amp, zero crossing triac with optical isolation. It is intended to switch a 24 VAC starter circuit to energize the fan motor.

**Output 2– HEATING STAGE 1 & 2** Default settings select 0 to 100% range . The first heating triac turns on above 40% and off below 30%. The second heating triac turns on when the output is above 80% and turns off below 70%.

**Output 3–COOLING STAGE 1 & 2** Default settings select 0 to 100% range . The first cooling triac turns on above 40% and off below 30%. The second cooling triac turns on when the output is above 80% and turns off below 70%.

**Output 4–OUTSIDE / RET DAMPER** his output configuration is set up to modulate an outdoor damper actuator from 2 to 10 volts ( 0 - 100% open ).

### Configuring variables

A total of thirty-two system variables are available to perform various functions such as set points, limits, and modes. All variables can be changed or modified to suit the specific application. The following have been pre-defined for use in the KMC supplied control sequences:

**Variable 17–MIXED AIR SET POINT** This is the temperature that Controller 3 (CON3) will maintain at the mixed air sensor (IN3) by adjusting the OUTSIDE / RET DAMPER (OUT4).

**Variable 18–COOLING STAGE 1** This variable is ON when the first stage of cooling is required (the first cooling triac is energized). Manually turning this variable ON will also force the first stage of cooling on.

**Variable 19–COOLING STAGE 2** This variable is ON when the second stage of cooling is required (the first and second cooling triacs are energized). Manually turning this variable ON will also force the first and second stages of cooling on.

**Variable 20–HEATING STAGE 1** This variable is ON when the first stage of heating is required (the first heating triac is energized). Manually turning this variable ON will also force the first stage of heating on.

**Variable 21–HEATING STAGE 2** This variable is ON when the second stage of heating is required (the first and second heating triacs are energized). Manually turning this variable ON will also force the first and second stages of heating on.

**Variable 22–OCCUPANCY STATUS** This variable can be toggled to indicate occupied / unoccupied status by user programming. It is used to start the fan in occupied times or when cooling or heating are needed in unoccupied mode. The Default is OCCUPIED. The user must determine the state.

**Variable 23–ECONOMIZER OPERATION** This variable can be easily toggled to enable or disable operation of the economizer control sequence. It should be set to ON when the outside air is suitable for cooling. The Default is DISABLED. The user must determine the state.

**Variable 24– HEAT / COOL MODE** Determines if the Roof Top Unit is in the heating or cooling mode. The default is COOL MODE. If this variable is off, only heating will take place when required by the space conditions. Cooling will be inoperable. When this variable is ON the cooling mode is on and only cooling will take place when required by the space conditions. Heating is disabled in COOL MODE. The user must determine the state.

**Variable 25 and 28–ACTIVE TEMPERATURE SET POINTS** These variables are the temperature set points for cooling and heating that the unit will maintain. In an unoccupied condition these values become the appropriate “setback” temperature settings (VAR26 and VAR29). In the occupied mode these values are calculated based on the “BASE SET POINT” (IN2) and “THERMOSTAT DEADBAND” (VAR 27). The set points are calculated in Program 1.

**Variable 26 and 29–SETBACK TEMPERATURE SET POINTS** For cooling and heating, respectively. These variables are fixed values entered by the user.

**Variable 27–THERMOSTAT DEADBAND** Our method of determining set points for cooling and heating insures a neutral range or “deadband”, exists between set points.

**Variable 30–STAGE DELAY** This variable establishes the minimum time the first stage of heating or cooling must run before the second is allowed to start. It also sets the minimum time between heating and cooling changeover .

**Variable 31–FAN DELAY** This is the minimum time the fan must run after the heating or cooling stages have been shut off.

**Variable 32–MINIMUM DAMPER POSITION** This variable allows easy access to changing the minimum damper position in an occupied mode as requested by the control sequence. The Default value is 25% outside air. In the unoccupied mode the damper output goes to 0%.

### **Configuring controllers (PID loops)**

KMC digital products feature built-in, full function PID (Proportional, Integral, and Derivative) controllers. In the KMD-7302, Controller 1 is set up as a direct-acting “cooling” thermostat. It compares SPACE TEMPERATURE (IN1) to the ACTIVE COOLING SETPT (VAR25) to operate the cooling stages (OUT3). Controller 2 defines a reverse-acting “heating” thermostat. It compares SPACE TEMPERATURE (IN1) to the ACTIVE HEATING SETPT (VAR28) to operate the heating stages (OUT2). Controller 3 compares mixed air temperature (IN3) to the MIXED AIR SET POINT (VAR17) for operation of the outside air damper in the economizer cycle. All parameters may be modified as necessary.

### Control Basic programs

Program areas 1 through 4 have specific KMC supplied programming burned into the controller at the factory. If none of sequence fits the required application, simply edit the programs or send blank programs to the controller and replace any of the factory programs. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.



### Caution

This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type)

The following explanations are provided for the KMC supplied programs in the KMD-7302:

| # | Description       | On                                  | Manual                   | Exit | Label |
|---|-------------------|-------------------------------------|--------------------------|------|-------|
| 1 | SET POINTS        | <input checked="" type="checkbox"/> | <input type="checkbox"/> | No   |       |
| 2 | FAN / HEAT / COOL | <input checked="" type="checkbox"/> | <input type="checkbox"/> | No   |       |
| 3 | ECONOMIZER        | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 4 |                   | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |
| 5 | USER PROGRAMMING  | <input type="checkbox"/>            | <input type="checkbox"/> | No   |       |

Buttons: Auto Load, Edit, Erase, OK, Cancel  
 Panel: A1 Status: Idle

**Illustration 3 Control Basic listing**

**Program 1–SET POINTS** The first part of this program serves to define the active temperature set points for use by the control routine.

In the occupied (VAR22 is ON) mode, the heating set point is calculated by subtracting half of the

deadband value (VAR27) from the set point indicated on the thermostat dial (IN2). Similarly, the cooling set point is found by adding half of the deadband to the thermostat setting.

In the unoccupied mode (VAR22 is OFF) the cooling and heating set points simply assume the “setback” values defined in Variables 26 and 29 respectively.



### Note

“User” programming will be required to change occupancy modes. The KMC supplied sequence simply defaults to an occupied mode on initial start-up.

This program also contains the basic sequence for the outside air damper. The damper is opened to minimum position (VAR32) in the occupied mode and is fully closed in the unoccupied mode. Additional damper control is available using the optional economizer program (PRG3).

**Program 2–FAN / HEAT / COOL** The fan is called to operate continuously in the occupied mode. In the unoccupied mode, if the controllers call for either heating or cooling to maintain the setback temperatures, the fan will run until the space temperature has been satisfied. A built-in differential prevents the fan from cycling on and off due to minor temperature variations.

**Note**

The triac outputs for heating and cooling are staged. The first stage must be on when the second stage is on.

In the heating mode (VAR24 is OFF), cooling stages are disabled. Controller 2 controls staging of the heat. If the controller is above 50% and the cooling has been off for the stage delay (VAR30) period, the first stage of heat will be turned ON. If the controller gets above 90% (The space continues to be too cold) and the stage delay (VAR30) for the first stage has timed out, the second stage of heating is started. The second stage will remain on until the controller falls below 50%. Both stages will be off once the controller output is below 10%.

In the cooling mode (VAR24 is ON), Heating stages are disabled. Controller 1 controls staging of the cooling. If the controller is above 50% and the heating has been off for the stage delay (VAR30) period, the first stage of cooling will be turned ON. If the controller gets above 90% (The space continues to be too warm) and the stage delay (VAR30) for the first stage has timed out, the second stage of cooling is started. The second stage will remain on until the controller falls below 50%. Both stages will be off once the controller output is below 10%.

**Note**

The user must determine Heating and Cooling Mode (VAR24). The default mode is COOL.

**Note**

Variables 18, 19, 20, and 21 are the cooling and heating flags. As long as Program 2 is running, these variables can be used to monitor the staging sequence. They can also be used to indirectly set the value of the outputs. You may find it easier to view and work with these variables ( ON and OFF units ) than the output points that have units in percentage.

**Program 3–ECONOMIZER** If this program is set to “run” and the economizer cycle is enabled (VAR23 = ON) the outside air damper will be positioned according to Controller 3, but not less than the minimum damper position specified in Variable 32. The Controller 3 compares mixed air temperature (IN3) to the mixed air set point as defined in Variable 17. The damper is controlled as a direct acting cooling device. If the mixed air temperature falls below the set point, the damper will be modulated closed to bring the temperature back to set point.

**Program 4 and 5 - Available for user programming** Program areas 4 and 5 are typically used to customize the control sequences for each particular application. The following sections discuss some of the most common options used with this model. The user will need to set values for OCCUPANCY MODE, ECONOMIZER MODE, and HEAT / COOL MODE in a program. It may also be necessary to

establish fan status here before running the cooling or heating. For complete details on writing your own custom logic in Control Basic Language, refer to the WinControl manual.

## Custom tables

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7302:

**Table1** Available for user programming.

**Table 2:** Correlates slider position of STE-5200 Series Thermostat into degrees F.

**Table 3** Correlates the dial setting of STE-5000 Series Thermostat Sensor to degrees F.

These tables use nominal values for correlation. Due to component tolerances and the resistance of field wiring, these values may need correction on an individual basis. If the value read is too low, enter a positive calibration in the input column labeled “cal”. If the value read is high, a negative calibration can be entered. . If necessary, reconfigure Table 2 and/or Table 3. The complete Table 2 and Table 3 from the factory are shown. If the tables have been reconfigured and the user wants the factory configuration for Table 2 and Table 3 it will be necessary to use Clear Panel or re-enter the values.



### Note

Using Clear Panel will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type.)

## Occupancy status

For simplicity, a system variable (VAR22) has already been defined as an "occupied status" indication. The KMC supplied routines do not define what constitutes an "occupied" status. However, the KMC supplied routines do reference the status of this variable when determining room set points or for outside air damper operations. Typically, the occupancy status is determined by reference to a time schedule by writing a line of code in Program 5 such as:

```
10 IF WS1 THEN START VAR22 ELSE STOP VAR22
```

Where WS1 is weekly schedule 1 as defined in the KMD-7302 panel. Scheduled times are assigned with up to four ON and four OFF times per day, for each day of the week. See the WinControl manual.

## Economizer

If you choose to use the KMC supplied economizer routine it will be necessary to determine when an economizer cycle can be allowed. An efficient method to do this is to compare outside and inside enthalpy. Operation of the economizer is then based on where the air handler can get the least expensive cooling. The lower the enthalpy is, the lower the heat content of the air. If the enthalpy is lower outside then the economizer is enabled. If the enthalpy is lower inside, the economizer is disabled. Here is an example of programming to make this comparison:

```
20 IF OUT-ENTH < IN-ENTH -1 THEN ENABLE VAR23
30 IF OUT-ENTH > IN-ENTH THEN DISABLE VAR23
```

In this example "OUT-ENTH" is the calculated outdoor enthalpy and "IN-ENTH" is the calculated indoor enthalpy. To make these calculations it is necessary to have outside temperature and humidity sensors as well as space or return air humidity and temperature sensors. A more simplistic approach to enabling the economizer is to look at outside air temperature. The following example looks at outside air and the cooling demand based on the controller output to the cooling valve:

```
20 IF CON1 > 5 AND OAT < 65 THEN ENABLE VAR23
30 IF CON1 < 1 OR OAT > 67 THEN DISABLE VAR23
```

This example requires a cooling demand (CON1) of greater than 5% and an outside air temperature (OAT) of less than 65 degrees before economizer operation is allowed. Once enabled, the economizer cycle will be disabled if either the demand for cooling falls off or the outside air temperature rises above 67 degrees.

## Heating/cooling mode

The KMC supplied program 2 uses Variable 24 to lockout cooling when heating is required and to lock-out heating when cooling is required. The mode can be selected based on factors such as outside air temperature and space temperature.

```
40 IF NOT ECONOMIZ AND OAT > 58 THEN START H/C-MODE
50 IF ECONOMIZ OR OAT < 55 THEN STOP H/C-MODE
```

This example uses the economizer variable to determine when the outside air is no longer suitable for cooling as a means of enabling mechanical cooling. ECONOMIZE (VAR23) is off when the damper is allowed to modulate past minimum position (VAR32).

## Fan proof

The fourth input (IN4) is available for some type of fan proof device. This may be an airflow switch, current transducer, or an auxiliary starter contact to show the fan has started. You can prohibit the heating and cooling from operating without the fan by adding a line of code in Program 5 such as:

```
60 IF NOT FAN$STAT THEN COOL$STG = 0 , HEAT$STG = 0
```

Where FAN\$STAT is proof the fan is running (IN4). Setting COOL\$STG and HEAT\$STG to 0 will keep all stages from operating until the fan has started.

## Temperature setpoints

This model has been pre-programmed assuming a set point potentiometer is connected to Input 2. In some cases, it may be desirable to provide a "fixed" set point in the software to free up the input for another purpose (or have a standardized set point for the whole building). The KMC supplied sequence determines the heating and cooling set points in an occupied mode as half the deadband below and above IN2 set point respectively. To change this to a fixed set point, simply write a line of code in Program 5 such as:

```
70 IF OCCUPIED THEN HEAT$SPT = 72 , COOL$SPT = 74
```

Different numbers or reference variables can substitute the numbers 72 and 74. Also note the KMC supplied sequence defines the unoccupied heating and cooling set points by referencing "setback" variables. These too can be modified if necessary, or the cooling and heating set points can be set directly as shown below:

```
80 IF NOT OCCUPIED THEN HEAT$SPT = 66 , COOL$SPT = 76
```

## Using the NetSensor

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7302, simply change the PID controllers 1 and 2 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor. Change SETPT\$TP in line 60 and line 70 of program1 to a variable mapped to button 2 on the NetSensor.

## Other options

One of the biggest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the KMC supplied sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated. The KMDC digital Technical Operator's Manual provides help and examples for programming Control Basic.

**WinControl settings** The following tables display the WinControl XL Plus settings for this applications note.

**Table 2 Inputs**

|   | Description       | Manual | Value   | Units                | Average | Label     |
|---|-------------------|--------|---------|----------------------|---------|-----------|
| 1 | SPACE TEMPERATURE | Auto   | 236.483 | Deg.F KMC10K Type II | 64      | SPACE\$TP |
| 2 | BASE SET POINT    | Auto   | 55      | Table 2              | 64      | SETPT\$TP |
| 3 | MIXED AIR TEMP    | Auto   | 184.042 | Deg.F KMC10K Type II | 64      | MIXED\$TP |
| 4 |                   | Auto   | 0       | Unused               | 64      |           |

**Table 3 Output**

|   | Description          | Manual | Value | Units  | 0%  | 100% | Security | Delay | Label     |
|---|----------------------|--------|-------|--------|-----|------|----------|-------|-----------|
| 1 | FAN                  | Auto   | On    | Off/On |     |      | 0        |       | FAN       |
| 2 | HEATING STAGE 1 & 2  | Auto   | 0     | 0-100% | 0.0 | 10   | 0        |       | HEAT\$STG |
| 3 | COOLING STAGE 1 & 2  | Auto   | 0     | 0-100% | 0.0 | 10   | 0        |       | COOL\$STG |
| 4 | OUTSIDE / RET DAMPER | Auto   | 25    | 0-100% | 2.0 | 10   | 0        |       | OUT\$DMPR |

**Table 4 Variables**

|      | Description                 | Manual | Value    | Units              | Label     |
|------|-----------------------------|--------|----------|--------------------|-----------|
| 1-16 | Available for user programs | Auto   | 0        |                    |           |
| 17   | MIXED AIR SET POINT         | Auto   | 55       | Degrees Fahrenheit | MIXD\$SPT |
| 18   | COOLING STAGE 1             | Auto   | Off      | Off/On             | COOLSTG1  |
| 19   | COOLING STAGE 2             | Auto   | Off      | Off/On             | COOLSTG2  |
| 20   | HEATING STAGE 1             | Auto   | Off      | Off/On             | HEATSTG1  |
| 21   | HEATING STAGE 2             | Auto   | Off      | Off/On             | HEATSTG2  |
| 22   | OCCUPANCY STATUS            | Auto   | Occupied | Unocc/Occ          | OCCUPIED  |
| 23   | ECONOMIZER OPERATION        | Auto   | Disabled | Dis/Enabled        | ECONOMIZ  |
| 24   | HEAT / COOL MODE            | Auto   | Cool     | Cool/Heat          | H/C-MODE  |
| 25   | ACTIVE COOLING SETPT        | Auto   | 57       | Degrees Fahrenheit | COOL\$SPT |
| 26   | SETBACK COOL SETPT          | Auto   | 80       | Degrees Fahrenheit | C\$SETBK  |
| 27   | T'STAT DEADBAND             | Auto   | 4        | Degrees Fahrenheit | DEADBAND  |
| 28   | ACTIVE HEATING SETPT        | Auto   | 53       | Degrees Fahrenheit | HEAT\$SPT |
| 29   | SETBACK HEAT SETPT          | Auto   | 62       | Degrees Fahrenheit | H\$SETBK  |
| 30   | STAGE DELAY                 | Auto   | 0:03:00  | Time               | STAGEDEL  |
| 31   | FAN DELAY                   | Auto   | 0:03:00  | Time               | FANDELAY  |
| 32   | MIN. DAMPER POSITION        | Auto   | 25       | Percent %          | MIN\$DMPR |

**Table 5 Tables**

| #  | Table 1 | Unused | Table 2 | Unused | Table 3 | Unused |
|----|---------|--------|---------|--------|---------|--------|
| 1  | 0       | 0      | 0       | 86     | 0       | 54     |
| 2  | 0       | 0      | 0.71    | 85     | 0.05    | 54     |
| 3  | 0       | 0      | 1.07    | 80     | 0.538   | 58     |
| 4  | 0       | 0      | 1.38    | 75     | 0.939   | 62     |
| 5  | 0       | 0      | 1.67    | 70     | 1.27    | 66     |
| 6  | 0       | 0      | 1.91    | 65     | 1.42    | 68     |
| 7  | 0       | 0      | 2.1     | 60     | 1.56    | 70     |
| 8  | 0       | 0      | 2.23    | 56     | 1.68    | 72     |
| 9  | 0       | 0      | 3       | 55     | 1.8     | 74     |
| 10 | 0       | 0      | 5       | 55     | 1.91    | 76     |
| 11 | 0       | 0      | 0       | 0      | 2.01    | 78     |
| 12 | 0       | 0      | 0       | 0      | 2.2     | 82     |
| 13 | 0       | 0      | 0       | 0      | 2.36    | 86     |
| 14 | 0       | 0      | 0       | 0      | 2.5     | 90     |
| 15 | 0       | 0      | 0       | 0      | 3       | 90     |

**Table 6 Control Basic listing**

| PRG | Description       | On  | Manual | Label |
|-----|-------------------|-----|--------|-------|
| #1  | SET POINTS        | YES | Auto   |       |
| #2  | FAN / HEAT / COOL | YES | Auto   |       |
| #3  | ECONOMIZER        | NO  | Auto   |       |
| #4  |                   | NO  | Auto   |       |
| #5  | USER PROGRAMMING  | NO  | Auto   |       |

**Program 1-SET POINTS**

```

30 REM DETERMINING ACTIVE SETPOINTS BASED ON OCCUPANCY MODE
50 IF NOT VAR22 THEN VAR25 = VAR26 , VAR28 = VAR29
60 IF VAR22 THEN VAR25 = IN2 + VAR27 / 2
70 IF VAR22 THEN VAR28 = IN2 - VAR27 / 2

100 REM *** Outside Damper Operation ***
110 IF VAR22 THEN OUT4 = VAR32 ELSE OUT4 = 0

300 END

```

**Program 2-FAN / HEAT / COOL**

```

30 REM *** FAN Control ***
40 IF OCCUPANCY~STATUS OR COOLING~STAGE~1~&~2 >= 50 OR
HEATING~STAGE~1~&~2 >= 50 THEN START FAN
50 IF NOT OCCUPANCY~STATUS AND TIME-OFF( HEATING~STAGE~1 ) >
FAN~DELAY AND TIME-OFF( COOLING~STAGE~1 ) > FAN~DELAY THEN STOP
FAN

60 REM *** Cooling / Heating ***
70 IF HEAT~/~COOL~MODE THEN 160 : REM Heat mode skips over
cooling sequence!

80 REM *** COOLING MODE ***
90 STOP HEATING~STAGE~1 : STOP HEATING~STAGE~2 : REM TURN OFF
HEATING
100 IF CON2 < 10 THEN STOP COOLING~STAGE~1 , STOP
COOLING~STAGE~2
110 IF CON2 > 50 AND TIME-OFF( HEATING~STAGE~1 ) > STAGE~DELAY
THEN START COOLING~STAGE~1
120 IF CON2 > 90 AND TIME-ON( COOLING~STAGE~1 ) > STAGE~DELAY
THEN START COOLING~STAGE~2
130 IF CON2 < 50 THEN STOP COOLING~STAGE~2
140 IF NOT COOLING~STAGE~1 THEN STOP COOLING~STAGE~2
150 GOTO 230

160 REM *** HEATING MODE ***
170 STOP COOLING~STAGE~1 : STOP COOLING~STAGE~2 : REM TURN OFF
COOLING
180 IF CON1 < 10 THEN STOP HEATING~STAGE~1 , STOP
HEATING~STAGE~2
190 IF CON1 > 50 AND TIME-OFF( COOLING~STAGE~1 ) > STAGE~DELAY
THEN START HEATING~STAGE~1
200 IF CON1 > 90 AND TIME-ON( HEATING~STAGE~1 ) > STAGE~DELAY
THEN START HEATING~STAGE~2
210 IF CON1 < 50 THEN STOP HEATING~STAGE~2
220 IF NOT HEATING~STAGE~1 THEN STOP HEATING~STAGE~2

230 IF NOT COOLING~STAGE~1 AND NOT COOLING~STAGE~2 THEN
COOLING~STAGE~1~&~2 = 0
240 IF COOLING~STAGE~1 THEN COOLING~STAGE~1~&~2 = 50
250 IF COOLING~STAGE~2 THEN COOLING~STAGE~1~&~2 = 100
260 IF NOT HEATING~STAGE~1 AND NOT HEATING~STAGE~2 THEN
HEATING~STAGE~1~&~2 = 0
270 IF HEATING~STAGE~1 THEN HEATING~STAGE~1~&~2 = 50
280 IF HEATING~STAGE~2 THEN HEATING~STAGE~1~&~2 = 100

290 END

```

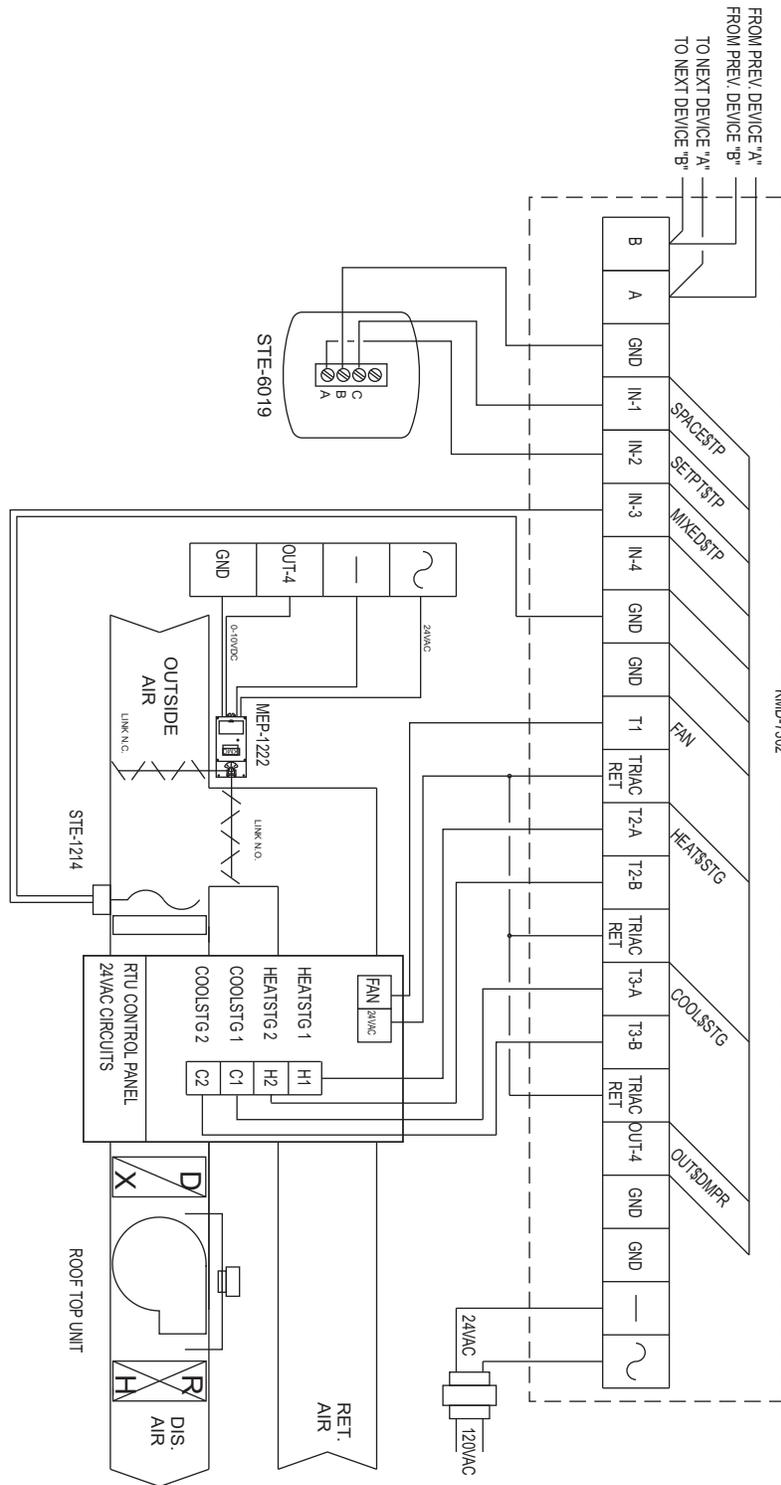
**Program 3-ECONOMIZER**

```

30 REM *** ECONOMIZER CYCLE ***
40 IF NOT ECONOMIZER~OPERATION THEN END
50 IF NOT OCCUPANCY~STATUS THEN END
60 OUTSIDE~/~RET~DAMPER = CON3
70 IF OUTSIDE~/~RET~DAMPER < MIN.~DAMPER~POSITION THEN
OUTSIDE~/~RET~DAMPER = MIN.~DAMPER~POSITION
80 END

```

**Application drawings**



**Illustration 4 Rooftop unit with two-stage heat and cool and optional economizer.**

## Application Note–AN0404I

---

### **KMD-7401 Heat Pump Controller**

This application note describes methods and programming to use the KMD–7401 as a heat pump controller with optional auxiliary heat.

---

#### **Related materials**

In addition to the material presented in this installation and operation guide, review and have available the following reference materials.

- ◆ Installation and operation guides for KMD–7401 and STE–5012
- ◆ WinControl XL Plus User's Manual
- ◆ Hardware Configuration Manager reference guide

#### **Introduction**

The KMD-7401 is a KMC digital DDC Controller pre-programmed as a Heat Pump Controller. The KMC supplied routine is designed to operate a Fan (OUT1), Compressor (OUT2), and a Reversing Valve (OUT3) as required to maintain a Space Temperature (IN1) according to a Space Setpoint (IN2). In addition to this basic control logic a variety of options have been incorporated. The controller determines active set points based on occupancy mode and Heat/Cool Changeover Mode as well as the set point selected on the local thermostat. Other options include a built-in Auxiliary Heating routine utilizing output 4 (OUT4) and Time Delays that can be adjusted to optimize system operation.

If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory configurations allowing you to program any desired sequence.

This feature allows you to custom tailor the control sequence to match specific applications. The following paragraphs explain the pre-programmed control sequence in detail and offer suggestions for user-programming to customize operation of the controllers.

#### **Hardware Configuration Manager (HCM)**

For networked operation, use the HCM software to configure the address, baud, and last panel setting. Each controller requires a distinct panel address and a common baud rate to reside on a network. The unit with the highest address must

be designated as LAST PANEL. Before running the HCM software to set these parameters, flip the network isolation switch located near the terminal strip. See the Operators Manual for details on HCM and networking controllers.

**Table 1 Default Configuration for KMD-7401**

|                    |           |
|--------------------|-----------|
| Panel Number       | 124       |
| Network Baud Rate  | 19200     |
| Last Panel         | Unchecked |
| Phone compensation | No        |
| Panel Type         | 121       |

## Setting up with WinControl

This section covers the set-up requirements to activate any of the pre-programmed options available in this device. At this point, communications should have been established with the unit.

A “Master” password is available to “sign-on” to each controller. Other security codes can be entered to control access to the system.

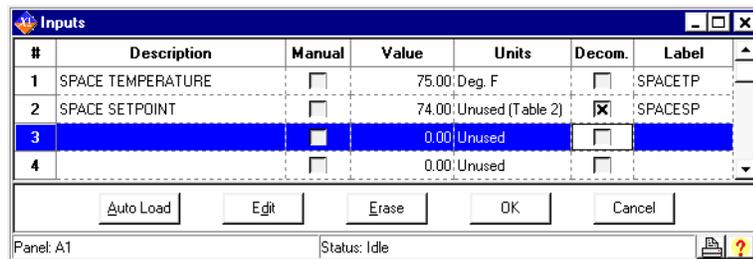
If you do not want to use any of the factory setup and programming options, send a panel file with your own setup parameters and programming to the controller. This will remove all of the factory programming allowing you to program any desired sequence.

The first step after signing on to the controller is to “Load Descriptors One” or “Load Descriptors All”. Many of the descriptions have been pre-defined for Inputs, Outputs, and Variables to aid in configuring the options.

To use a program or programs for a specific application, go to the “Control Basic” programs screen and turn “ON” the appropriate program.

### Configuring inputs

All Inputs can be selected through software for either analog or digital ranges. The pre-programmed sequences assume the following input configuration:



**Illustration 1 Input configuration with WinControl XL**

**Input 1–SPACE TEMPERATURE** Default selects a KMC 10K ohm thermistor as used in all of KMC’s STE series sensors.

**Input 2–SPACE SETPOINT** Default selects Table 2 which has been setup for the potentiometer used on KMC’s STE 5000 series thermostats.

**Input 3–** Unused

**Input 4–** Unused

**Configuring outputs**

The KMD-7401 Controller features 1 amp, 24 VAC, optically isolated - zero crossing Triacs built directly onto the board assembly. For this reason, all outputs should be configured for DIGITAL operation only (On/Off, Open/Close, etc.). The KMC supplied control routines have been written to use the following output configuration:

| # | Description          | Manual                   | Value | Units     | Label   |
|---|----------------------|--------------------------|-------|-----------|---------|
| 1 | FAN                  | <input type="checkbox"/> | On    | Off/On    | FAN     |
| 2 | HEAT PUMP COMPRESSOR | <input type="checkbox"/> | Off   | Off/On    | HTPUMP  |
| 3 | REVERSING VALVE      | <input type="checkbox"/> | Cool  | Heat\Cool | REV.VLV |
| 4 | AUXILIARY HEATING    | <input type="checkbox"/> | Off   | Off/On    | AUXHTG  |

**Illustration 2 Output configuration with WinControl XL**

**Output 1–FAN (Off/On)** Activates the fan by completing a 24 VAC circuit.

**Output 2–COMPRESSOR (Off/On)** Activates the compressor on a call for heating or cooling by completing a 24 VAC circuit .

**Output 3–REVERSING RELAY (Heat/Cool)** Changes the action of the heat pump. The heating mode is “OFF” and the cooling mode is “ON”.

**Output 4–AUXILIARY HEAT (Off/On)** An optional program is included to use an auxiliary heating unit.

**Configuring variables**

A total of thirty-two system variables are available to perform various functions such as set points, limits, and modes. All variables can be changed or modified to suit the specific application. The following have been pre-defined for use in the KMD-7401 KMC supplied control sequences:

**Variable 23–OCCUPANCY MODE** This variable can be toggled to indicate occupied / unoccupied status by user programming as desired. Default mode is occupied.

**Variable 24–CHANGEOVER MODE (COOL/HEAT)** Chooses whether the unit will operate in heating or cooling by automatically setting the thermostat action as direct (COOL) or reverse (HEAT).

**Variable 25–HEAT SETBACK SETTING** In an unoccupied mode (VAR23 = Unocc), the Active Heat Setpoint (VAR26) becomes this temperature; The thermostat set point is ignored.

**Variable 26–ACTIVE HEAT SETPOINT** This is the temperature to be maintained in the space in the HEAT mode (VAR24 = HEAT).

**Variable 27–ACTIVE COOL SETPOINT** This is the temperature to be maintained in the space in the COOL mode (VAR24 = COOL).

**Variable 28–COOL SETBACK SETTING** In an unoccupied mode (VAR23 = Unocc), the Active Cool Setpoint (VAR27) becomes this temperature; The thermostat set point is ignored.

**Variable 29–THERMOSTAT DEADBAND** In an occupied mode (VAR23 = Occupy), the Active Heat Setpoint is the Space Setpoint (IN2) minus half this value and the Active Cool Setpoint is the Space Setpoint plus half this value. In the unoccupied mode the thermostat set point is not used and this value has no effect on the operation of the unit.

**Variable 30–COMP. LOCKOUT TIMER** This is the minimum off time for the compressor. The compressor will not start until it has been off for this time interval. The compressor will remain off for one-half this time interval following a switch in the Changeover Mode (VAR24).

**Variable 31–AUX HEAT DELAY TIMER** This is the amount of time the Heat Pump must operate before the auxiliary heat (OUT4) is started.

**Variable 32–FAN SHUTOFF DELAY** In Unoccupied times, this is the length of time the fan will run after the compressor has turned off. During occupied times (VAR23 = Occupy) the fan will run continuously.

#### **Configuring controllers (PID control loops)**

KMC digital products feature built-in, full function PID (Proportional, Integral, Derivative) controllers. On the KMD-7401 Controller #1 is a reverse-acting HEATING thermostat. Controller #2 is a direct-acting COOLING thermostat. All parameters may be modified as necessary.

#### **About the Control Basic programs**

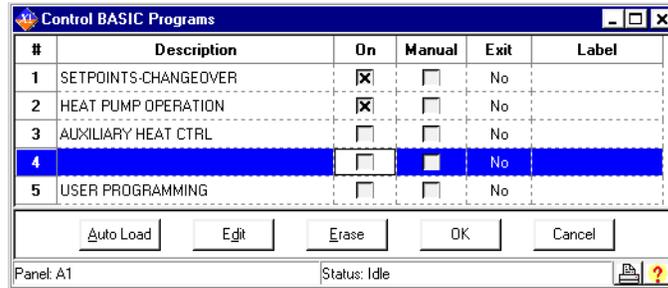
Program areas 1 through 4 have specific “canned” programming burned into the controller at the factory. If none of sequence fits the required application, simply edit the programs or send blank programs to the controller and replace any of the factory programs. The factory programs can be retrieved if necessary. The original factory programs can be re-installed again by using the Clear Panel function in the System Setup Menu.



#### **Caution**

This will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type)

The following explanations are provided for the “canned” programs in the KMD-7401:



**Illustration 3 Control Basic list**

**Program 1–SETPOINTS–CHANGEOVER** This program determines the active set point for heating and cooling modes in the occupied and unoccupied modes. In the unoccupied mode (VAR23 is off), the heating and cooling set points are set to the heating (VAR25) and cooling (VAR28) setback temperature set points. In the occupied mode the Active Heating Setpoint (VAR26) is the thermostat setting (IN2) minus one-half the deadband (VAR29). The Active Cooling Setpoint (VAR27) is the thermostat setting plus one-half the deadband.

This program also automatically sets the changeover mode based on the demands of the space.

**Program 2–HEAT PUMP OPERATION** This program handles the operation of the Fan (OUT1), the Compressor (OUT2), and the Reversing Valve. In the occupied mode the fan runs continuously. In the Unoccupied mode, the fan will turn ON whenever there is a call for either heating or cooling or upon a change from heating to cooling operation. In addition, the fan will start up after a power failure. Once the fan is on it will continue to run, until the compressor has been off for the time specified in the Fan Shut-Off Delay (VAR32).

The compressor will turn ON when the demand for either heating or cooling exceeds 55% as determined by the heating or cooling controllers (CON1 and CON2). Once on, the compressor will continue to operate until the demand for heating or cooling drops to less than 5%. The effective compressor deadband is 50% of the proportional band setup in the controllers. If a proportional band is 2 degrees, the compressor will continue operating until the room temperature (IN1) has changed by at least 1 degree. This built-in deadband serves to prevent the compressor from running for only short periods of time.

A Variable has been incorporated in the factory programming to prevent the compressor from turning ON unless it has been OFF for the time specified in the Compressor Lock-Out Timer (VAR30). This helps prevent short-cycling of the compressor.

The position of the Reversing Valve (OUT3) determines whether heating or cooling is required in the space. The factory program assumes a COOL mode is selected when the Reversing Valve is energized (OUT3 = COOL = ON). If no power is supplied to the Reversing Valve, a HEAT mode is initiated. A time delay prevents changing modes while the compressor is operating. Changing the reversing valve position is not allowed unless the compressor has been off for at least 1/2 the time period specified in the Compressor Lockout Timer (VAR30).

Once the position of the reversing valve is changed, the compressor is not allowed to turn ON until at least 1/2 the time period specified in the Compressor Lockout Timer (VAR30) has expired.

**Program 3–AUXILIARY HEAT CONTROL** An optional sequence has been included in the factory programming to enable operation of an auxiliary heater connected to output 4 (OUT4). A Delay Timer (VAR31) has been used to disable operation of the heater unless the heat pump compressor has been operating continuously for at least this time period. This delay allows the heat pump a chance to recover the space conditions by itself without the use of auxiliary heat. This is useful in cases where there has been a large change in the active set point, such as after a change from Unoccupied to Occupied modes. If the time delay has elapsed, the Auxiliary Heat will be commanded ON with a heating requirement greater than 95%. Once activated, the heater will run until the heating requirement of the space falls below 55% or if either the compressor or the fan are turned OFF. This program is off by default and must be turned ON to use the Auxiliary Heat option.

**Program 4 and 5** Available for user programming.

## Tables

Custom tables allow the user to create unique “ranges” to be used by inputs or other special applications. The most often used application of these custom tables is to establish a correlation between a potentiometer setting (a voltage) and a temperature value (such as the dial setting on a thermostat). ). The KMD-7000 Series Controllers allow user-access to all 3 tables. These tables have been defined for the following applications in the KMD-7401:

**Table 1:** Ready for user programming.

**Table 2:** Correlates the slider position of STE-5200 Series thermostats to degrees F.

**Table 3:** Correlates the dial setting of STE-5000 Series thermostats to degrees F.



### Note

These tables use nominal values for correlation. Due to component tolerances and the resistance of field wiring, these values may need correction on an individual basis. If the value read is too low, enter a positive calibration in the input column labeled “cal”. If the value read is high, a negative calibration can be entered. . If necessary, reconfigure Table 2 and/or Table 3. The complete

Table 2 and Table 3 from the factory are shown. If the tables have been reconfigured and the user wants the factory configuration for Table 2 and Table 3 it will be necessary to use Clear Panel or re-enter the values.

| # | Table 1 | Unused | Table 2 | Unused | Table 3 | Unused |
|---|---------|--------|---------|--------|---------|--------|
| 1 | 0       | 0      | 0       | 0      | 0       | 0      |
| 2 | 0       | 0      | 0.14    | 0      | 0.35    | 0      |
| 3 | 0       | 0      | 0.2     | 75     | 0.46    | 276    |
| 4 | 0       | 0      | 0.4     | 327    | 0.86    | 566    |
| 5 | 0       | 0      | 0.8     | 563    | 1.1     | 7      |
| 6 | 0       | 0      | 1       | 664    | 1.7     | 112    |
| 7 | 0       | 0      | 1.5     | 904    | 2.3     | 15     |
| 8 | 0       | 0      | 2       | 1143   | 2.5     | 1648   |

*Illustration 4 Custom tables*

**! Caution**

Using Clear Panel will erase any programming changes that have been made to the controller. (Clear Panel does not affect changes made with HCM, or panel type.)

## Occupancy status

For simplicity, a system variable (VAR23) has already been defined as an “occupied status” indication. The “canned” routines do not define what constitutes an “occupied” status. However, the “canned” routines do reference the status of this variable when determining room set points. Typically, the occupancy status is determined by reference to a time schedule by writing a line of code in Program 5 such as:

```
10 IF WS1 THEN START VAR23 ELSE STOP VAR23
```

Where WS1 is weekly schedule 1 as defined in the KMD-7401 panel. Scheduled times are assigned with up to four ON and four OFF times per day, for each day of the week. See the KMC digital Operator's Manual for details on setting up schedules.

## Temperature setpoints

This model has been pre-programmed assuming a set point potentiometer is connected to Input 2. In some cases, it may be desirable to provide a “fixed” set point in the software to free-up the input for another purpose (or have a standardized set point for the whole building). The “canned” sequence determines the heating and cooling set points in an occupied mode as half the dead band below and above IN2 set point respectively. To change this to a fixed set point simply write a line of code in Program 5 such as:

```
70 IF OCCUPIED THEN HTG.SP = 72 , CLG.SP = 74
```

The numbers 72 and 74 can be substituted by different numbers or reference variables. Also note the “canned” sequence defines the unoccupied heating and cooling set points by referencing “setback” variables. These too can be modified if necessary, or the cooling and heating set points can be set directly as shown below:

```
80 IF NOT OCCUPIED THEN HTG.SP = 66 , CLG.SP = 76
```

## Using the NetSensor

A NetSensor can be used to bring values into the controller without using any physical input. To add a NetSensor and use the factory programming in the KMD-7401, simply change the PID controllers 1 and 2 in-label from IN1 to the variable mapped to the NetSensor's onboard temperature sensor. Change both SPACESP in line 40 of program1 to a variable mapped to button 2 on the NetSensor.

**Other Options**

One of the biggest features of any KMC digital DDC Controller is the inherent programming ease and flexibility available in each and every unit. If the "canned" sequences do not satisfy the application at hand they may be turned off or edited and the entire control strategy can be manipulated. The KMC digital Technical Operator's Manual provides help and examples for programming Control Basic.

**WinControl settings** The following tables list the WinControl XL Plus settings for this application note.

**Table 2 Input**

|   | Description       | Manual | Value | Units                      | Average | Label   |
|---|-------------------|--------|-------|----------------------------|---------|---------|
| # | SPACE TEMPERATURE | Auto   | 75    | Deg.F<br>KMC10K<br>Type II | 64      | SPACETP |
| 2 | SPACE SETPOINT    | Auto   | 74    | Table 2                    | 64      | SPACESP |
| 3 |                   | Auto   | 0     | Unused                     | 64      |         |
| 4 |                   | Auto   | 0     | Unused                     | 64      |         |

**Table 3 Output**

|   | Description          | Manual | Value | Units      | 05 | 100% | Security | Delay | Label   |
|---|----------------------|--------|-------|------------|----|------|----------|-------|---------|
| 1 | FAN                  | Auto   | On    | Off/On     |    |      | 0        | 0:00  | FAN     |
| 2 | HEAT PUMP COMPRESSOR | Auto   | Off   | Off/On     |    |      | 0        | 0:00  | HTPUMP  |
| 3 | REVERSING VALVE      | Auto   | Cool  | Heat\\Cool |    |      | 0        | 0:00  | REV.VLV |
| 4 | AUXILIARY HEATING    | Auto   | Off   | Off/On     |    |      | 0        | 0:00  | AUXHTG  |

**Table 4 Variables**

|       | <b>Description</b>          | <b>Manual</b> | <b>Value</b> | <b>Units</b>       | <b>Label</b> |
|-------|-----------------------------|---------------|--------------|--------------------|--------------|
| #1-22 | Available for user programs |               |              |                    |              |
| #23   | OCCUPANCY MODE              | Auto          | Occupied     | Unocc/Occ          | OCC.MODE     |
| #24   | CHANGEOVER MODE             | Auto          | Cool         | Cool/Heat          | HTG.MODE     |
| #25   | HEAT SETBACK SETTING        | Auto          | 60           | Degrees Fahrenheit | HTGSETBK     |
| #26   | ACTIVE HEAT SETPOINT        | Auto          | 53           | Degrees Fahrenheit | HTG.SP       |
| #27   | ACTIVE COOL SETPOINT        | Auto          | 57           | Degrees Fahrenheit | CLG.SP       |
| #28   | COOL SETBACK SETTING        | Auto          | 80           | Degrees Fahrenheit | CLGSETBK     |
| #29   | THERMOSTAT DEADBAND         | Auto          | 4            | Degrees Fahrenheit | DEADBAND     |
| #30   | COMP. LOCKOUT TIMER         | Auto          | 0:05:00      | Time               | LOCKTIME     |
| #31   | AUX HEAT DELAY TIMER        | Auto          | 0:20:00      | Time               | AUXDELAY     |
| #32   | FAN SHUTOFF DELAY           | Auto          | 0:02:30      | Time               | FANDELAY     |

**Table 5 Controllers (PID control loops)**

|   | <b>Point</b> | <b>Manual</b> | <b>Output</b> | <b>Setpoint</b> | <b>Action</b> | <b>Prop.</b> | <b>Reset</b> | <b>H/M</b> | <b>Rate</b> | <b>Bias</b> |
|---|--------------|---------------|---------------|-----------------|---------------|--------------|--------------|------------|-------------|-------------|
| 1 | IN1          | Auto          | 0%            | VAR26           | RA(-)         | 2            | 0            | Hour       | 0           | 0           |
| 2 | IN1          | Auto          | 100%          | VAR27           | DA(+)         | 2            | 0            | Hour       | 0           | 0           |
| 3 |              | Auto          | 0%            |                 | DA(+)         | 0            | 0            | Hour       | 0           | 0           |
| 4 |              | Auto          | 0%            |                 | DA(+)         | 0            | 0            | Hour       | 0           | 0           |

**Table 6 Tables**

|    | <b>Table 1</b> | <b>Unused</b> | <b>Table 2</b> | <b>Unused</b> | <b>Table 3</b> | <b>Unused</b> |
|----|----------------|---------------|----------------|---------------|----------------|---------------|
| 1  | 0              | 0             | 0              | 86            | 0              | 54            |
| 2  | 0              | 0             | 0.71           | 85            | 0.05           | 54            |
| 3  | 0              | 0             | 1.07           | 80            | 0.538          | 58            |
| 4  | 0              | 0             | 1.38           | 75            | 0.939          | 62            |
| 5  | 0              | 0             | 1.67           | 70            | 1.274          | 66            |
| 6  | 0              | 0             | 1.91           | 65            | 1.421          | 68            |
| 7  | 0              | 0             | 2.1            | 60            | 1.558          | 70            |
| 8  | 0              | 0             | 2.23           | 56            | 1.684          | 72            |
| 9  | 0              | 0             | 3              | 55            | 1.801          | 74            |
| 10 | 0              | 0             | 0              | 0             | 1.911          | 76            |
| 11 | 0              | 0             | 0              | 0             | 2.013          | 78            |
| 12 | 0              | 0             | 0              | 0             | 2.198          | 82            |
| 13 | 0              | 0             | 0              | 0             | 2.362          | 86            |
| 14 | 0              | 0             | 0              | 0             | 2.5            | 90            |
| 15 | 0              | 0             | 0              | 0             | 3              | 90            |

**Table 7 Congrol Basic programs**

|   | <b>Description</b>  | <b>On</b> | <b>Manual</b> | <b>Label</b> |
|---|---------------------|-----------|---------------|--------------|
| 1 | SETPOINTS-CHANGEVER | YES       | Auto          |              |
| 2 | HEAT PUMP OPERATION | YES       | Auto          |              |
| 3 | AUXILIARY HEAT CTRL | NO        | Auto          |              |
| 4 |                     | NO        | Auto          |              |
| 5 | USER PROGRAMMING    | NO        | Auto          |              |

**Program 1-SETPOINTS-CHANGEVER**

```

30 REM DETERMINING ACTIVE SETPOINTS BASED ON OCCUPANCY MODE
40 IF OCCUPANCY~MODE THEN ACTIVE~HEAT~SETPOINT = SPACE~SETPOINT
  - THERMOSTAT~DEADBAND / 2 , ACTIVE~COOL~SETPOINT =
  SPACE~SETPOINT + THERMOSTAT~DEADBAND / 2
50 IF NOT OCCUPANCY~MODE THEN ACTIVE~HEAT~SETPOINT =
  HEAT~SETBACK~SETTING , ACTIVE~COOL~SETPOINT =
  COOL~SETBACK~SETTING
60 REM ----- DETERMINE CHANGEVER MODE
  -----
70 IF CON1 > 50 THEN START CHANGEVER~MODE
80 IF CON2 > 50 THEN STOP CHANGEVER~MODE
90 END
    
```

### Program 2-HEAT PUMP OPERATION

```
30 IF+ CHANGEVER~MODE THEN START A : REM enables start-up delay
sequence
40 IF- CHANGEVER~MODE THEN START A : REM enables start-up delay
sequence
50 IF A AND TIME-ON( A ) < COMP.~LOCKOUT~TIMER / 2 THEN STOP
HEAT~PUMP~COMPRESSOR , GOTO 90
55 IF TIME-ON( A ) > COMP.~LOCKOUT~TIMER THEN STOP A
70 IF CHANGEVER~MODE THEN START REVERSING~VALVE ELSE STOP
REVERSING~VALVE
75 IF+ REVERSING~VALVE THEN START B
76 IF- REVERSING~VALVE THEN START B
77 IF B AND TIME-ON( B ) < COMP.~LOCKOUT~TIMER / 2 THEN STOP
HEAT~PUMP~COMPRESSOR , GOTO 90
78 IF TIME-ON( B ) > COMP.~LOCKOUT~TIMER THEN STOP B
80 IF CHANGEVER~MODE THEN GOSUB 100 ELSE GOSUB 200
85 REM ----- Fan control routine -----
87 IF NOT OCCUPANCY~MODE AND TIME-OFF( HEAT~PUMP~COMPRESSOR ) >
FAN~SHUTOFF~DELAY THEN STOP FAN
90 IF OCCUPANCY~MODE OR A OR B OR HEAT~PUMP~COMPRESSOR THEN
START FAN
95 END
100 REM --- heating operation subroutine ----
110 IF CON1 < 5 THEN STOP HEAT~PUMP~COMPRESSOR
120 IF TIME-OFF( HEAT~PUMP~COMPRESSOR ) > COMP.~LOCKOUT~TIMER
THEN 130 ELSE 140
130 IF CON1 > 55 THEN START HEAT~PUMP~COMPRESSOR
140 RETURN
200 REM --- cooling operation subroutine ----
210 IF CON2 < 5 THEN STOP HEAT~PUMP~COMPRESSOR
220 IF TIME-OFF( HEAT~PUMP~COMPRESSOR ) > COMP.~LOCKOUT~TIMER
THEN 230 ELSE 240
230 IF CON2 > 55 THEN START HEAT~PUMP~COMPRESSOR
240 RETURN
```

### Program 3-AUXILIARY HEAT CTRL

```
30 REM ----- AUXILIARY HEAT CONTROL ( OUTPUT #4 )
-----
40 IF NOT CHANGEVER~MODE THEN STOP AUXILIARY~HEATING , END
50 IF TIME-ON( HEAT~PUMP~COMPRESSOR ) < AUX~HEAT~DELAY~TIMER
THEN GOTO 70
60 IF CON1 > 95 THEN START AUXILIARY~HEATING
70 IF CON1 < 55 THEN STOP AUXILIARY~HEATING
75 REM ***** only allows auxiliary heat if FAN and HEATPUMP
are ON *****
80 IF NOT HEAT~PUMP~COMPRESSOR OR NOT FAN THEN STOP
AUXILIARY~HEATING
90 END
```

Application drawings

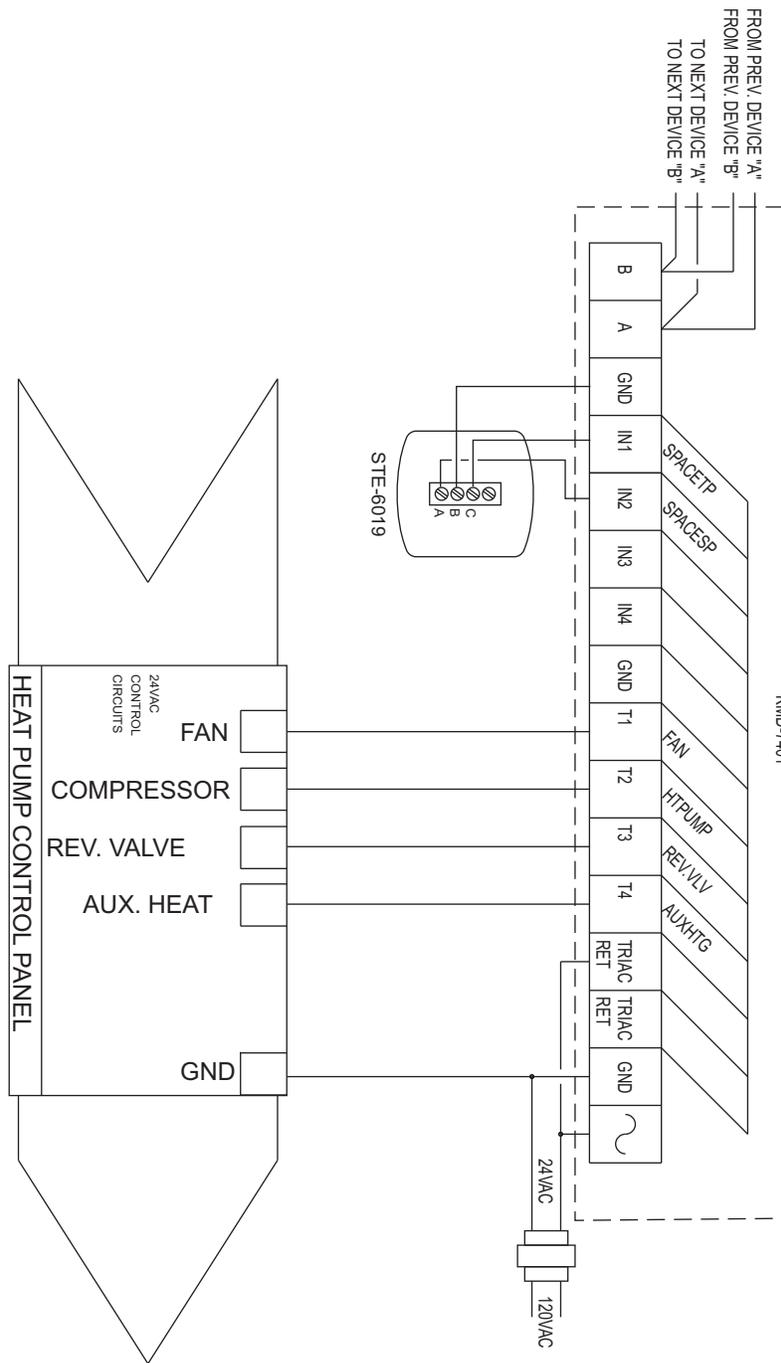


Illustration 5 Heat pump with optional auxiliary heat



Application Note–AN0905B

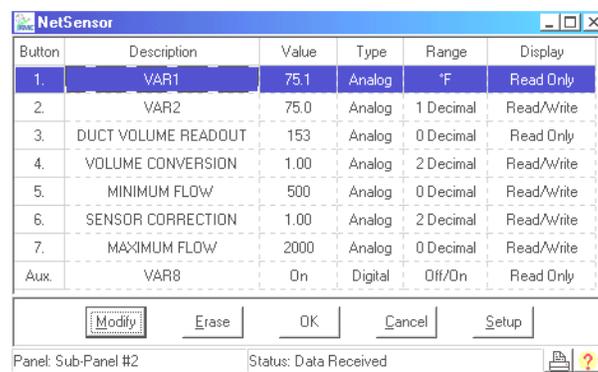
## Using the NetSensor on preprogrammed controllers

The factory-programmed sequences in the KMD–7000 series controllers use an STE series temperature sensor to measure space temperature. This application note lists the procedures to modify the programs to use a NetSensor.

The programming in the KMD–7000 series controllers assigns the sensor for space temperature to input 1 (IN1) and the space temperature set point to input 2 (IN2). If you are using the NetSensor to supply these values, the first two inputs on the controller can be used for other sensors. However, the controller will still attempt to look at these points for temperature and set point values. It will be necessary to redefine these points for the program to work properly.

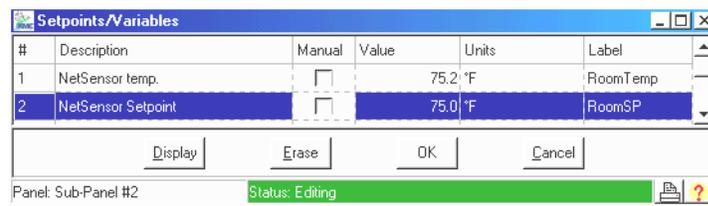
The KMD-7000 stores a connected NetSensor's information as variables. You assign these variables when programming the KMD–7000 controller. The following changes should be made to allow the factory program to use the NetSensor information instead of the first two inputs of the controller:

1. Under Control Menu, choose “NetSensor” and configure the NetSensor as follows:



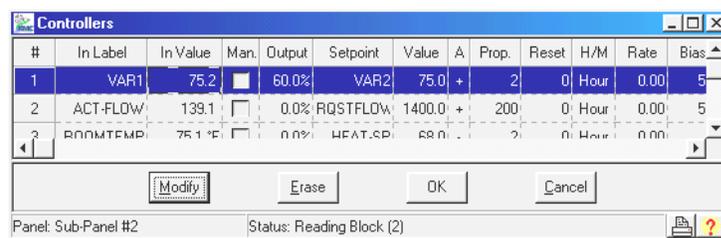
**Illustration 6 NetSensor dialog**

2. Configure the Variables in the KMD-7000 assigning the proper units:



**Illustration 7 Variables dialog**

- For any PID Controllers that use space **temperature** as an input, the “In Label” should be replaced with the Label or mnemonic of the variable assigned to the space **temperature** from the NetSensor.



**Illustration 8 PID controllers dialog**

- To use the NetSensor to supply space **set point** information it is necessary to add code to override the factory programs. The following lines are examples of code that can be used to determine the heating and cooling set points from the NetSensor **set point** value. In this example, “BASE-SP” is the label given to the variable that was assigned to the NetSensor **set point** value.

*Example:*

```

10 IF OCCUPIED THEN HEAT$SPT = BASE-SP - DEADBAND / 2
20 IF OCCUPIED THEN COOL$SPT = BASE-SP + DEADBAND / 2

```

The occupied mode must also be defined in either Program 5 or in a Weekly Schedule. In the unoccupied mode, the controller is set up to use the setback and set-up temperatures stored as variables (COOL\$SPT and HEAT\$SPT). If the NetSensor space set point is to be active at all times, the “IF OCCUPIED THEN” can be removed from each program line.

On VAV box controllers there is no occupied variable defined. Typically you will use an occupancy variable in the air handling unit controller where the schedule is held to determine when each box should use setback or setup temperature setpoints. Variables 13 and 14 can be used for this purpose in all KMD-7000 series VAV box controllers. Add the following lines to program 5 to allow unoccupied temperature setpoints.

*Example:*

```

30 IF NOT OCCUPIED THEN HEAT$SPT = VAR13
40 IF NOT OCCUPIED THEN COOL$SPT = VAR14

```

### Limiting Changeable Values

To create a restricted range on buttons the user is allowed to change, add the limits to Control Basic Program 5. The following example limits a setpoint to within 68 and 76. The setpoint is variable 2 as in the above information.

*Example:*

```
100 IF VAR2 < 68 THEN VAR2 = 68
110 IF VAR2 > 76 THEN VAR2 = 76
```

These lines can be added anywhere in program 5 as long as they are executed at every scan (do not locate them where they may be skipped over such as after a GOTO statement)




---

#### Note

Changes made to KMD-7000 controllers using the NetSensor are held only in RAM memory. A power outage will cause any changes to be lost and default to the values stored in flash (non-volatile) memory. It is advisable to store default values in the controller so that reasonable control can be maintained when power is restored.

The NetSensor data will be saved automatically to flash every 24-hours. In addition, you can write the data more frequently if required by the application. A control basic function can be called to perform this task by using a line such as:

*Example:*

```
70 IF Interval( 0:10:00 ) Then GOTO 80 else GOTO 100
80 IF ABS( ROOMSP - S ) > .5 THEN GOTO 90 ELSE GOTO 100
WRITE-TO-FLASH : S = ROOMSP
```

These lines will look for a change in the setpoint of greater than .5 degrees at which point the change will be written to flash. To avoid wearing out the flash and taking up processor time the write to flash should be used in moderation. Line 70 assures the write to flash will take place no more than once every 10 minutes. This is to accommodate the time period while the user is adjusting the setpoint so multiple writes do not occur.



## Application Note—AN0805A

---

### **The NetSensor VAV balancer program**

This application note describes a method to balance air flow using a flow hood, NetSensor as a service tool and the balancer program included in KMC VAV controllers.

---

#### **Related materials**

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ Installation and operation guides for the applicable KMC VAV controllers
- ◆ The application note from The Digital Designer's Guide that is appropriate for each controller
- ◆ WinControl XL Plus User's Manual
- ◆ System plans with controller addresses

#### **Additional equipment**

The air flow balancing method described in this application note requires the following equipment:

- ◆ A flow hood or other accurate method to measure air flow.
- ◆ A NetSensor and cable for installations that use a temperature sensor instead of a NetSensor.

#### **Overview**

KMC VAV controllers are pre-programmed with NetSensor function keys defined to use to turn the NetSensor into a simple service tool. With this feature, a balance technician can:

- ◆ Adjust setpoints for minimum and maximum air flow
- ◆ Enter volume conversion factors
- ◆ View air flow readings
- ◆ When necessary, establish sensor correction factors.

In addition, the NetSensor can be used to override commands to the controller forcing control at either minimum or maximum setpoints as necessary to verify actual unit flow.

A balancer can use one of two methods to setup and calibrate KMC VAV controllers:

## Adjusting airflow with WinControl software

If a computer is available, the most flexible method of defining minimum and maximum flow setpoints and sensor adjustment factors is with WinControl software to communicate directly with each controller. All user-programmable data is available using this method while security features allow limiting their access.

## Using a NetSensor as a balancer service tool

The following items are factory programmed for access by the NetSensor service tool:

There are two different ways to use the NetSensor service tool. Primarily, the difference is whether or not the box needs to be commanded to control at a specified flow setting for checkout and verification of actual versus measured flow.



### Note

After balance parameters are set, save the program for each controller that has been configured. Save each panel individually or use the Save Network option in the Panel Menu.

1. Control box flow Using Normal Wall Mounted Thermostat. If the VAV controller is equipped with a standard room thermostat with a setpoint dial the box can be forced to minimum or maximum flow by adjusting the room setpoint. For example, a cooling box is commanded to minimum flow by adjusting the setpoint higher than the space temperature. Maximum flow can be achieved by setting the thermostat below the current space temperature.
2. Force Box Control Using NetSensor

The Program 4 in a KMD preprogrammed VAV controller is a balancer override routine. By enabling this program, the VAV controller is forced to control at either minimum flow or maximum flow.

### Starting program 4

Starting program 4 depends upon the type of NetSensor in use.

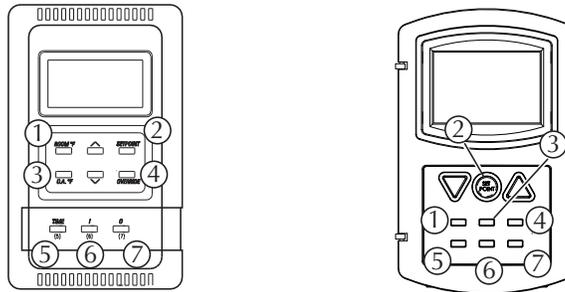
**NetSensors with LED displays** Each NetSensor contains a pair of auxiliary contact wires located on the back of the unit. By connecting these two wires together the NetSensor will activate Program 4 in the controller and attempt to operate at either minimum flow or maximum flow settings as defined by button 2 on the NetSensor.

**NetSensors with LCD displays** NetSensors KMD-1161 and KMD-1181 do not have auxiliary wires. Press buttons 5 and 7 together and then press an up arrow or down arrow button to change auxiliary input from *Off* to *On* to activate Program 4.



**Note**

When the Balancer Override routine takes control over the standard temperature control routine, accurate space temperature may not be maintained. Also note disable Program 4 *before* disconnecting the NetSensor to return control to the normal temperature sequence. The balancer routine will be canceled automatically after 30 minutes if the NetSensor is disconnected from the controller before the auxiliary contacts are opened.



**Illustration 9 NetSensor button numbers**

**Table 8 Button assignments**

| Button     | Descriptor | Variable | Function   |
|------------|------------|----------|--|
| 1          | ROOMTEMP   | 1        | Displays the temperature detected by the NetSensor. This value is not used in the balancing routine.   |
| 2-Normal   | ACTIVESP   | 25       | With Program 4 <i>Off</i> this button shows the active temperature setpoint.   |
| 2-Balancer | ACTIVESP   | 25       | When program 4 is running, set this value to 0 for minimum air flow or 1 for maximum air flow.<br>Negative numbers and numbers greater than 1 can be displayed for this button but these values will be corrected once the NetSensor updates its information with the connected controller. The controller will determine if the value is valid, and will display a legal value within 15 seconds. |
| 3          | VOLUME1    | 26       | This is the controller measured flow reading with units corrected and volume converted into appropriate units.   |
| 4          | VOLFACTR   | 21       | Volume conversion factor<br>This value multiplies directly against the Actual flow reading (which has been corrected for flow irregularities) to convert feet per minute into other units such as CFM. Typical volume conversion factors and an example of how to calculate these factors is included below.   |

**Table 8 Button assignments (Continued)**

| Button | Descriptor | Variable | Function   |
|--------|------------|----------|--|
| 5      | MIN-FLOW   | 19       | Minimum flow setpoint<br>Displays the minimum flow setpoint for the VAV box. The default unit of measure is in Feet Per Minute. If volume factor is provided the units can be changed as required. |
| 6      |            | 22       | Sensor correction factor<br>This value multiplies against the flow reading in feet per minute to correct for duct irregularities and tube placement.   |
| 7      | MAX-FLOW   | 20       | Maximum flow setpoint<br>Displays the maximum flow setpoint for the VAV box. The default unit of measure is in Feet Per Minute. If volume factor is provided the units can be changed as required. |

**Example balance method**

The following procedure is an example of how a VAV box might be balanced. Other methods may be used just as effectively. Always remember that there is up to 15 seconds between updates of the NetSensor. Modifications at the NetSensor will not take effect at the controller until an update has occurred. Follow these steps after plugging the NetSensor into the controller. The display should show a valid space temperature.

1. Determine the flow units desired and the box size. Enter the factor to convert Feet per minute into the desired units at button 4. The default value is 1.0.

**Example 1:**

The following example is for a 9-inch diameter duct (.375 foot radius) and for desired units in CFM.

$$A = \pi r^2 \text{ where } r \text{ is radius in Feet} \quad \text{Or:} \quad A = \pi^2/256 \text{ where } E \text{ is diameter in Inches.}$$

$$= 3.14 * .375^2 = .44 \text{ Square Feet} \quad \quad \quad = 3.14 * 9^2/576 = .44 \text{ Square Feet}$$

Enter .44 at button 4. This value is multiplied by the feet per minute flow in program 1 to arrive at CFM. The duct volume readout at button 3 will show flow in CFM units. Also, minimum and maximum flow must be defined in CFM values.

**Example 2:**

For a rectangular duct with a 10 inch height (H) and 12 inch width (W):

$$A = H * W / 144$$

$$= 10 * 12 / 144 = .83 \text{ Square Feet}$$

Enter .83 at button 4. This value is multiplied by the feet per minute flow in program 1 to arrive at CFM. The duct volume readout at button 3 will show flow in CFM units. Also, minimum and maximum flow must be defined in CFM values.

1. Set the minimum and maximum flow limits in button 5 and button 6 respectively.
2. Start the balance program by closing the switch on the auxiliary contact of the NetSensor. Or, adjust the room thermostat setpoint to command the unit to minimum or maximum flow as required.
3. Set button 2 (MIN 0 / MAX 1 button) to 1. This will place the flow setpoint at the maximum value as set in button 7. The box damper will drive to maintain the maximum flow setpoint. Wait for the value at button 3 to reach the setting for maximum CFM. Measure the flow from each supply louver to get the total supplied air at maximum. Differences between the displayed flow reading and field measurements may be due to sensor mounting location or turbulence. If the readings differ greatly, the flow reading can be adjusted by placing a multiplier in button 6. Use the following method to arrive at a the factor:

$$\text{Correction Factor (Button 6)} = \frac{\text{Actual Flow (Field Measurement)}}{\text{Controller Readout (Button 3)}}$$

If the measured volume is 200 CFM and the reading shown at button 3 is 180 CFM, then enter 1.11 at button 6 to increase the reading.

4. Set button 2 (MIN 0 / MAX 1 button) to 0. This will place the flow setpoint at the minimum value as set in button 5. The box damper will drive to maintain the minimum flow setpoint. Wait for the value displayed at button 3 to reach the setting for maximum CFM. Measure the flow from each supply louver to get the total supplied air at minimum. If the readings differ greatly, the flow reading can be adjusted by altering the minimum setpoint at button 5 to reach the desired reading.
5. Stop the balance program by opening the switch on the auxiliary contact on the NetSensor. Wait approximately 10 seconds for the unit to sense and recognize the open contact before disconnecting the NetSensor and moving to the next VAV box controller.





# Part 4

## Program modules

This sections in this part of the Programmer's Companion are examples of various modules of programming. These modules can be added to other programs as required to customize functions in KMC controllers.

---

|  |      |
|--|------|
| AN0504A Dial-out program for KMD-5501 .....                              | 4-3  |
| AN0504B Calculating enthalpy .....                                       | 4-9  |
| AN0504C Controlling a tri-state valve actuator from two outputs .....    | 4-13 |
| AN0504E Examples of lead/lag programs.....                               | 4-19 |
| AN0604A Optimal start/stop routine .....                                 | 4-27 |
| AN0909A A simplified optimum start program .....                         | 4-31 |
| AN0504F Programming override timers.....                                 | 4-39 |
| AN0504G Working with pulse inputs.....                                   | 4-43 |
| AN0504H Calculating boiler reset temperature .....                       | 4-45 |
| AN0604B Calculating discharge reset based on space temperature .....     | 4-53 |
| AN0504I Signal conditioning.....   | 4-55 |
| AN0504J Example programs using tables .....                              | 4-59 |
| AN0604C Monitoring refrigeration systems for moisture and flash gas..... | 4-65 |
| AN1009A Determining occupancy with motion sensing .....                  | 4-75 |
| AN0410A Measuring outside air temperature.....                           | 4-83 |



## Application Note—AN0504A

---

### Dial-out program for KMD-5501

This application note describes a method to connect and program a KMD-5501 for communications over a modem.

---

#### Applicable models

This note applies to KMD-5501/02 controls with firmware version 1.99/2.00 or later. The controllers must have the following Control Basic Statements available.

- ◆ PHONE ATDT string
- ◆ HANGUP
- ◆ ON-ERROR line #

These statements, when used with a KMD-5556 modified modem interface box, will allow a KMD-5501/02 panel to automatically dial a PC running the InControl software.

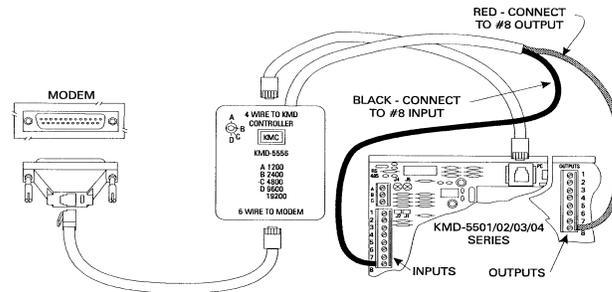
- ◆ ATDT in PHONE statement is the standard attention command used by most modems.
- ◆ Use ATDP for pulse dialing.
- ◆ The string in the *PHONE* statement is phone number to be dialed and should not contain any dashes or extra spaces.
- ◆ ON-ERROR statement detects whether or not the PHONE statement was successful. If not successful it directs the program to the specified line #.

#### Setup requirements

The remote site you wish the KMC digital system to call needs to have a Personal Computer running the InControl software and be positioned at the "SIGN-ON" menu. It strongly suggested a printer also be connected to the PC for recording the ALARM messages, as the message will only appear on the screen during the call. You must be sure the modems at both ends operate at the same BAUD RATE as the KMD network for communication to take place.

In addition, one specific KMD-5501/02 panel must be connected to a MODIFIED MODEM INTERFACE box (model # KMD-5556) on the job site. This interface box has two (2) leads which connect to the panel as follows:

- ◆ The output lead MUST be connected to OUTPUT #8 of the panel containing the PHONE program.
- ◆ The input lead is typically connected to INPUT #8 of the same panel but could be changed if the program is also changed to match. (ref. lines 50 & 260)



**Illustration 1 Modem connection**



**Note**

The OUTPUT wire is used to grab the modem's attention so it may receive call-out instructions from the panel. The PHONE command is instructed to utilize OUTPUT #8 and cannot be changed. OUT8 should be configured for OFF/ON (range 50) operation.

The input wire is used to let the panel know if the modem is available or whether it is already in use. If the modem is being used, such as someone calling into the system, the call-out routine will be disabled so it does not cut-off the current user. IN8 should be configured for OFF/ON (range 50) operation.

The modem itself needs to be properly configured by NOT echoing characters (ATE0), answering on the first ring (ATS0=1), and setting the carrier detect signal high (AT&C1). Other configuration parameters may also need to be defined but the KMD panel specifically requires these.

**Alarms explanation**

One example of using the PHONE statement is to have a job site call some remote location if a certain ALARM condition is detected. The first step to understanding how our example program works requires a solid knowledge of how the ALARM statements operate.

There are essentially three (3) separate status levels possible with every ALARM condition

**OFF** or not in alarm is the normal status when the ALARM condition is not true.

**ON** The alarm becomes ON whenever the condition FIRST becomes true.

Note this only happens at the first detection of the ALARM condition.

**RESTORED** an alarm is in restored status after the condition that determined it was ON alarm has been corrected. Note that this is NOT the same as the alarm being OFF even though the condition is no longer considered in an alarm state.

The ALARM statement is used to generate an alarm message when some condition is determined to be out of control. This message is then sent out to a Personal Computer (PC) to be viewed and/or printed so the operator may take appropriate action. Once an alarm has been triggered ON it will NOT return back to OFF until it has been restored AND an operator has received the restored status. By requiring the restored status to be acknowledged by an operator the system is being assured that someone has been informed of the problem - even if it is no longer in alarm conditions. If a PC is connected on-line with the system and is running IC.exe software all alarm status levels will be acknowledged automatically by that PC. As long as that PC is connected all ALARM messages will show-up on the screen and/or be printed. These messages will remain on the PC's screen until either it is disconnected from the network or an operator "clears" the messages from the Alarms menu. For a permanent hard copy it is suggested a printer be included to receive all alarm messages. If there are no PC's on-line with the network these ALARM messages will simply remain on the network until someone signs on either locally or remote via modem. Once connected these ALARMS will then be displayed appropriately.

## Typical application

The sample ALARM CALLOUT program we have included is designed to call a remote station upon detection of certain ALARM conditions. This program may be used either with or without a PC on-site. If a PC is connected locally the only message that will be sent to the remote site is the message included with the DALARM (delay alarm) statement. This statement waits to generate the alarm until 35 seconds after connection is made and since the remote station will be "on-line" with the network at the time the DALARM is triggered both PC's will receive the message. Probably the most logical message this statement should contain is which system (name or location) is calling so whoever reads the message knows where to return a call or go to visit.

If a PC will not be used on-site the DALARM message could be discarded. Note however that if a PC is not used any "restored" alarms will need to be acknowledged by the remote user before that particular ALARM condition will be capable of initiating another call-out routine. Normally this would not be a problem since it is assumed whoever receives the message will see to it the condition is restored and acknowledged accordingly. If you wish to have the system call-out both on detection of an ALARM as well as when the condition is restored an "IF-" statement could be added after each ALARM.

*Example:*

```
190 ALARM IN1 < 33 , 2 , temperature is close to freezing
200 ON-ALARM 260
201 IF- IN1 > 35 THEN GOTO 260
```

Note how the IF- statement matches the point the ALARM condition will "restore" by taking into account the differential value. In this case the alarm triggers if the temperature drops below 33 degrees but doesn't reset or "restore" until it is greater than 35 (Differentials such as this are often used to prevent a condition from repeatedly bouncing true/false).

Since line 260 initiates our call-out routine this program would now call the remote station each time the ALARM condition changed from one state to another and as the condition is restored the remote site serves to acknowledge it.

**Control Basic listing**

```
10 IF VAR32 = 0:00:00 THEN E = 0 , GOTO 140 : REM no callout
requested
20 IF C THEN GOTO 140 : REM already connected so skip callout
routine
30 IF E THEN GOTO 110 : REM connection error - go to error
sequence
40 REM ----- CALLOUT ROUTINE
-----
50 IF NOT IN8 THEN GOTO 170 : REM modem is already in use
60 PHONE ATDT 1234567890 : REM PUT PHONE NUMBER HERE
70 ON-ERROR 90
80 C = 1 : E = 0 : GOTO 170 : REM connection successful-read
alarms
90 E = 1 : END
100 REM ----- ERROR DETECTED ROUTINE
-----
110 IF R > 20 THEN R = 0 , E = 0 , VAR32 = 0 , C = 0 , HANGUP ,
END
120 IF TIME-ON( E ) MOD 0:05:00 < 0:00:05 THEN R = R + 1 , GOTO
60
130 REM ----- IS IT TIME TO END CALL?
-----
140 IF TIME-ON( C ) > VAR32 THEN 150 ELSE 170
150 C = 0 : R = 0 : VAR32 = 0 : E = 0 : HANGUP
160 REM ----- ENTER YOUR ALARM STATEMENTS HERE
-----
170 DALARM C , 35 , CALLED 1234567890 WITH ALARMS AT KREUTER
ENGINEERING
180 ON-ALARM 260
190 ALARM IN1 < 1 , 0.1 , IN1 IS LOW
200 ON-ALARM 260
210 ALARM IN2 < 1 , 0.1 , IN2 IS LOW
220 ON-ALARM 260
230 ALARM IN3 < 1 , 0.1 , IN3 IS LOW
240 ON-ALARM 260
250 END
260 IF NOT IN8 THEN END : REM disregard alarm - modem is in use
270 IF VAR32 = 0:00:00 THEN VAR32 = 0:02:00
280 END
```

**Program explanation**    **Main program lines 10 - 30**

**10** If VAR32 is zero it means we have no request to call out - so program skips down to reading alarm statements and assures all variables are reset.

**20** If "C" means that we are already connected to a remote location and the program skips down to see if its time to end the call.

**30** If "E" shows there was an error in trying to complete the call so the program branches off to an error routine to try calling again later.

**CALLOUT ROUTINE lines 50 - 90**

**50** This line looks to see if the modem is already busy. If it is, the callout routine must be skipped.

**60** PHONE ATDT 1234567890 This line is likely the most critical. Be sure there are no extra spaces or punctuation after the phone number. NOTE: While the program is executing this statement all programs in this panel are suspended until either a "connect" string is received or 45 sec have elapsed.

**70** If a "connect" string is not received within 45 seconds the ON-ERROR statement becomes true and the 'E' error flag is set in line #90.

**80** Since an error was not detected "C" is set high to show connection was successful.

**Error routine lines 100-120**

**110** This is a retry counter to specify how many times the panel should attempt calling when errors are detected. This number may be increased or decreased as desired.

**120** This line establishes the time interval between trying to call again - in this case every 5 minutes. This time may be changed as required.

**Time to end call?**

**140** Recall "C" is high when a successful connection is made. VAR32 is set for some time period the call will connect for (ref. line 260). When time connected exceeds the required time to connect the system hangs up and resets all variables back to zero.

### Alarms section

**170** A delay alarm statement (DALARM) must be used as shown with a 35 second time delay whenever a PC might be used on-site. This statement will ensure that at least one alarm message is sent every time a call is made and should contain a message pertaining to the system location or other identification. If a PC is on-site, each alarm message will get answered by that PC and no messages would be displayed during the call - unless of course an alarm is triggered while the remote connection is made and explains why the 35 second delay is used.

**180** each alarm condition you wish to trigger the phone call must be followed by the ON-ALARM statement to set VAR32 to some time period. This is what starts the whole call-out routine in motion.

**190** You may enter as many alarm statements/conditions as required up until the 1024 bytes available in this program are filled. Additional alarms may be placed in other programs just be sure to set VAR32 to a time period ON-ALARM if you want it to generate a phone call. (ref. 260 & 270)

**260** This function determines whether the modem is already in use (i.e., someone is already on-line with the system). If the modem is busy it is assumed whoever is on-line will receive these alarm messages and that it will not be necessary to generate a phone call at that point. Therefore VAR32 is left at zero and a call is not initiated.

**270** This line sets the time interval you wish to remain connected with the remote site. The minimum interval should be at least 2 minutes to assure alarm messages can be processed. Larger systems with more possible alarm messages may want to increase this connection time.



#### Note

An easy way to cancel out a phone routine is to set VAR32 to 0:00:00 as this will either end the call or cancel an error routine, whichever is necessary. Be sure all ALARM messages have been received and/or cleared to assure proper continued operation.

## Application Note—AN0504B

---

### Calculating enthalpy

This application note describes a method to calculate enthalpy in the early series KMD direct digital controllers.

---

#### Applicable models

This application note applies to the following models:

- ◆ KMD–5110 and KMD–5111 MultiNet controllers
- ◆ KMD–5501 and KMD–5502 local controllers
- ◆ KMD–6000 series terminal unit controllers.



#### Note

The firmware in KMD–5200 series, KMD–5800 series and KMD–7000 series includes enthalpy calculations. The programming in this application note does not apply.

#### Related materials

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ Installation and operation guide for the controller
- ◆ InControl or WinControl XL Plus User's Manual

#### Introduction

Enthalpy is a measure of the heat content within a given sample of air and is expressed in BTU's per pound of dry air. The example programs in this application note calculates a value for enthalpy for outside air and/or return air to evaluate which sample has a lower or higher heat content. The result of the calculation can then be used to open or close outside air dampers.

## Programming KMD-5100 MultiNet controllers

KMC Controls KMD-5100 series MultiNet Controllers have the programming ability to perform exponential and logarithm calculations. The following sample program uses these functions and can only be used only with the above mentioned units.

*Example:*

```
10 REM FULL ENTHALPY CALCULATION FOR USE IN MAIN PANELS
(KMD-5100)
20 REM Assumes standard atmospheric pressure (14.7 psi) ref
line 100
30 REM Enter temperature in degrees F (TEMP) in lines 50 and 110
40 REM Enter relative humidity percentage (OSH) in line 90
50 T = TEMP + 459.67 : REM Enter temperature
*****
60 N = LN( T )
70 L = -10440.4 / T - 11.29465 - 0.02702235 * T + 1.289036E-005
* T ^ 2 - 2.478068E-009 * T ^ 3 + 6.545967 * N
80 S = LN-1( L )
90 P = OSH / 100 * S : REM Enter humidity
*****
100 W = 0.62198 * P / ( 14.7 - P )
110 H = 0.24 * TEMP + W * ( 1061 + 0.444 * TEMP ) : REM Enter
temp *****
120 ENTHALPY = H : REM SET VARIABLE EQUAL TO H AS DESIRED
130 REM VALUE OF CALCULATION IS IN BTU/LB OF DRY AIR
```

## Programming KMD-5000 and KMD-6000 series controllers

For use KMC Controls local controllers—KMD-5501 and KMD-5502—or the terminal unit controllers—KMD-6000 series—the following programs calculate enthalpy values. Note the values listed below must be entered into one of the custom look-up tables; the example uses table #3. These values were derived from the ASHRAE HANDBOOK and represent the vapor pressure of air at a given temperature. As a compromise between precise accuracy and full range we have used values ranging from zero to 110 degrees Fahrenheit. Therefore lines 10 and 20 were added to simply say the enthalpy value if temperature is below zero will be 0 and if temperature is over 110 degrees enthalpy is 100 (ie, very high). You could customize your table to match requirements where necessary.

Both programs require two pieces of information:

- ◆ Relative Humidity
- ◆ Temperature (Dry Bulb)

## Program listing

*Example:*

```
5 REM Enthalpy calculation is valid for use between 0 and 110
degrees F
6 REM Requires using TABLE #3 as shown below
7 REM Enter TEMP in degrees F in lines 10, 20, 30, and 60
8 REM Enter HUMIDITY in line 40
9 REM line 70 ( H ) is enthalpy in BTU per LB
10 IF TEMP < 0 THEN H = 0 , END
20 IF TEMP > 110 THEN H = 100 , END
30 P = TBL( TEMP , 3 )
40 Q = HUMIDITY / 100 * P
50 W = 0.62198 * Q / ( 14.69 - Q )
60 H = 0.24 * TEMP + W * ( 1061 + 0.444 * TEMP )
70 ENTHALPY = H
80 END
```

(Line 60 or 70 would define enthalpy, such as 70 VAR1 = H)

**Table 1 Table 3 data**

| 0   | .185   |
|-----|--------|
| 10  | .03089 |
| 20  | .05049 |
| 30  | .08086 |
| 40  | .12172 |
| 50  | .17811 |
| 55  | .21410 |
| 60  | .25635 |
| 65  | .30574 |
| 70  | .36328 |
| 75  | .43008 |
| 80  | .50736 |
| 90  | .69889 |
| 100 | .95031 |
| 110 | 1.2764 |

## Application Note–AN0504C

---

Revision A–September 2005

# Controlling a tri-state valve actuator from two outputs

This application note describes a method to control a tri-state valve actuator with two universal outputs in a direct digital controller.

---

### Applicable models

The programming described in this application note controls a tri-state valve actuator. The KMD-5801 controller is shown in the Illustration 1 controls the an MEP-7703 tri-state actuator. However, any KMD series actuator with two universal outputs can be substituted for the KMC-5801.

### Related materials

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ Installation and operation guide for the KMD-5801 or the controller that will be installed.
- ◆ Installation and operation guides for REE-2102, NetSensor and the actuator under control.
- ◆ WinControl XL Plus User's Manual.
- ◆ HCM reference guide.
- ◆ System plans with controller addresses.

### Setting up with WinControl XL

This section covers the requirements to program the controller with WinControl XL or WinControl XL Plus.

#### Configuring inputs

This application note assumes a NetSensor is installed to control the setpoint and measure the space temperature. All universal inputs are available.

#### Configuring outputs

**Output #1–DRIVE TO CLOSE** Commands the actuator to drive the valve to the closed position.

**Output #2–DRIVE TO OPEN** Commands the actuator to drive the valve to the open position.

### Configuring variables

Four variables are required to execute this program. An optional fifth variable, VAR7, may be associated with NetSensor button #7 for humidity.

**Variable # 1–ROOM TEMPERATURE** Corresponds to the space temperature displayed by NetSensor button #1.

**Variable #2–HEAT TEMP SETPOINT** This corresponds to NetSensor button #2 which is the active set point used by CON3. Units must match that of the temperature sensor on VAR1 either degrees Fahrenheit or degrees Celsius.

**Variable #3–Drive Time** This is the approximate time in seconds required for the actuator to move from the fully closed to the fully open position or from the fully open to the fully closed position. VAR3 is used to roughly calculate the actuator position in variable 4 (VAR4).

**Variable #4–Valve Position** This is a calculated position based on the full stroke drive time (VAR3). Variable 3 must be set correctly for the damper position information to work properly.

Valve position is calculated by measuring the time the valve has been commanded open or closed versus the time needed for complete travel. If the valve is stuck, this variable will not indicate the correct valve position. It will however indicate the controllers attempt to move the valve. This variable is used only to give users an indication of what the controller has done and is doing to control the position of the damper. It is not used as feedback in the control sequence.

**Variable #7–Humidity (optional)** Corresponds to NetSensor button #7.

### Configuring PID control loops

**Controller #3 (CON3)** compares the ROOM TEMPERATURE (IN1) to the HEAT TEMP SETPOINT (VAR2) for operation of the reheat stages when programs 2 or 3 are enabled.

Controller 3 (CON3) is programmed as a reverse-acting heating thermostat with 50% bias as shown in Table 2. For cooling valve control, change CON3 to direct acting (+) in the *Action (A)* column.

### Local variables

B = ON when less heat is required (moves valve toward closed)

C = ON when more heat is required (moves valve toward opened position)

X = Count in seconds

Y, Z used in valve position calculation

**WinControl settings** The following tables display the WinControl XL Plus settings for this application note.

**Table 1 Output configuration**

| Description      | Manual | Output | Units    | 0 % | 100 | Delay | Security | Label    |
|------------------|--------|--------|----------|-----|-----|-------|----------|----------|
| 1 DRIVE TO CLOSE | auto   | Off    | Off / On |     |     | 0:00  | 0        | DRV-CLOS |
| 2 DRIVE TO OPEN  | auto   | Off    | Off / On |     |     | 0:00  | 0        | DRV-OPEN |
| 3- (Available)   |        |        |          |     |     |       |          |          |

**Table 2 Variables**

| Description          | Manual | Value | Units              | Label    |
|----------------------|--------|-------|--------------------|----------|
| 1 ROOM TEMPERATURE   | auto   | 0.00  | Degrees Fahrenheit | ROOMTEMP |
| 2 HEAT TEMP SETPOINT | auto   | 0.00  | Degrees Fahrenheit | HEAT-SP  |
| 3 VALVE POSITION     |        |       |                    | VLV-POS  |
| 4 DRIVE TIME         |        |       |                    | DRV-TIME |
| 5 (Available)        |        |       |                    |          |
| 6 (Available)        |        |       |                    |          |
| 7 HUMIDITY           |        |       |                    |          |
| 8- (Available)       |        |       |                    |          |

**Table 3 PID control loop**

| In Label        | In Value   | A/M  | Output | Setpoint | Value | A | Prop | Reset | Rate | Bias   |
|-----------------|------------|------|--------|----------|-------|---|------|-------|------|--------|
| 1-2 (Available) |            |      |        |          |       |   |      |       |      |        |
| 3 ROOMTEMP      | 0.00 deg.F | auto | 0.0 %  | HEAT-SP  | 68.0  | - | 2    | 0     | H    | 50.0 0 |
| 4 (Available)   |            |      |        |          |       |   |      |       |      |        |

**Table 4 NetSensor**

| Btn. | Point                       | Value | Type    | Range     | Display    |
|------|-----------------------------|-------|---------|-----------|------------|
| #1   | VAR1                        | 75.3  | Analog  | °F        | Read Only  |
| #2   | VAR2                        | 0     | Analog  | 0 Decimal | Read/Write |
| #3-6 | Available for user programs | 0     |         |           |            |
| #7   | VAR7                        | 42    | Analog  | 0 Decimal | Read/Write |
| #8   |                             | Off   | Digital | Off/On    | Read/Write |

### Program listing

*Example:*

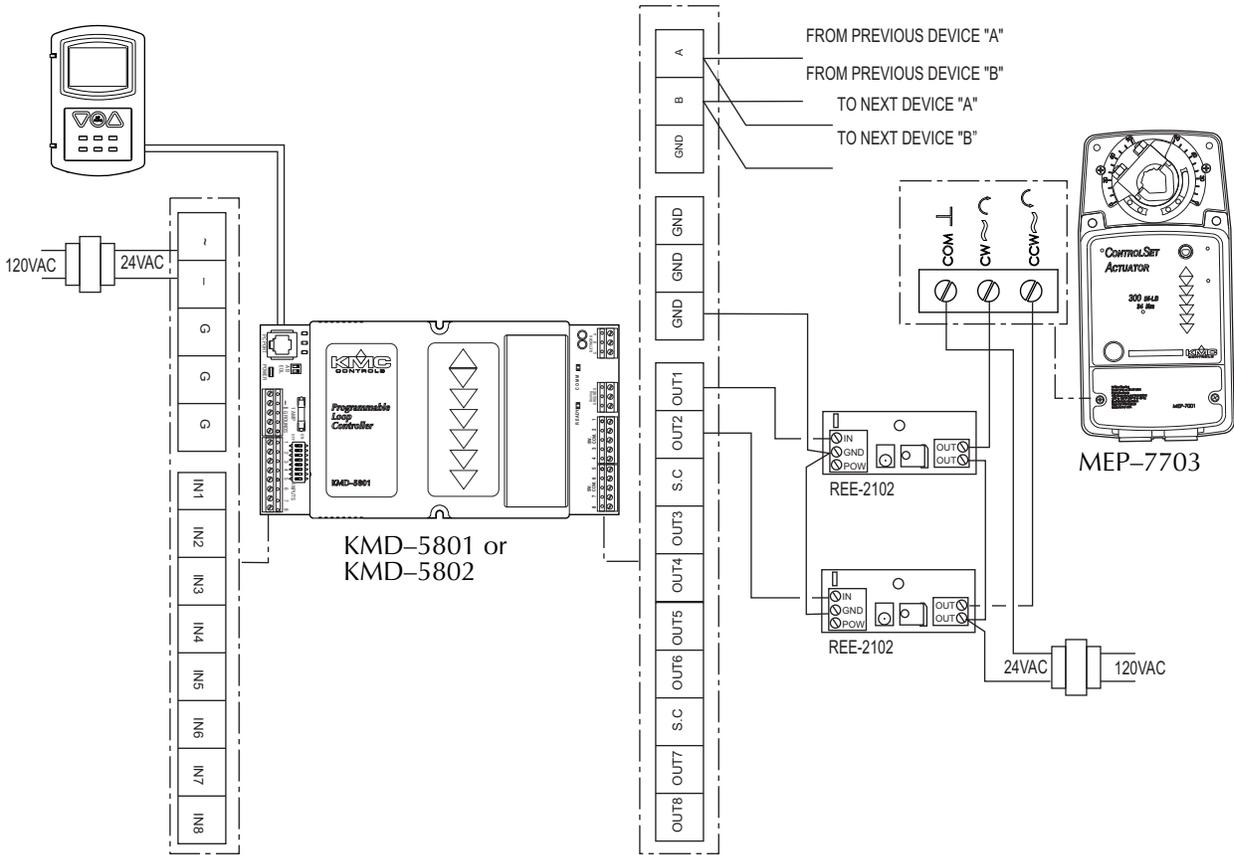
```
60 IF CON3 > 70 THEN START C , STOP B
70 IF CON3 < 30 THEN START B , STOP C
80 IF CON3 > 35 AND CON3 < 65 THEN STOP B , STOP C
81 X = 1 / SCANS
82 IF CON3 < 99 AND CON3 > 1 THEN 83 ELSE 100
83 IF CON3 < 36 OR CON3 > 64 THEN START D ELSE STOP D
90 IF TIME-ON( D ) > 0:04:00 THEN GOTO 100

92 REM The following will pulse the actuator as it nears
satisfied position to reduce overshoot
93 Y = X + Y : IF Y > 5 THEN Y = 0
94 Z = ABS( CON3 - 50 ) / 50
95 IF Y > Z THEN STOP B , STOP C

100 DRV-CLOS = B : DRV-OPEN = C

130 REM Lines 140 through 170 are optional to calculate valve
position based on stroke time
140 IF DRV-CLOS THEN VLV-POS = VLV-POS - X * 100 / DRV-TIME
150 IF DRV-OPEN THEN VLV-POS = VLV-POS + X * 100 / DRV-TIME
160 IF VLV-POS > 100 THEN VLV-POS = 100
170 IF VLV-POS < 0 THEN VLV-POS = 0
180 END
```

**Application drawing**



**Illustration 1 Tri-state valve actuator**



## Application Note—AN0504E

---

Revision A—September 2005

### Examples of lead/lag programs

This application note describes methods and programming to perform lead/lag sequential operation for two, three or four pieces of equipment.

---

#### About lead/lag

Lead/lag sequences control two or more pieces of equipment performing the same function, with one acting as a backup to the other. The *Lead* device is programmed to run when required by a controlled variable (demand). As the demand increases, the *Lag* device is called to run. Although the programs in this application note are written for pumps, they will work for other types of lead/lag equipment as well.

#### Lead/lag for two pieces of equipment

The following start/stop program is for two pieces of equipment that perform the same function. The lead will change weekly or on a failure of the lead unit.

*Example:*

```

5 REM ROTATE LEAD/LAG ON TUE OF EACH WEEK
10 IF+ DOW = TUE AND TIME > 09:00:00 THEN 20 ELSE 30
20 IF VAR1 THEN STOP VAR1 ELSE START VAR1
30 IF CON1 < 50 AND PMPS2RUN = 2 THEN PMPS2RUN = 1
32 IF CON1 > 50 AND PMPS2RUN = 0 THEN PMPS2RUN = 1
34 IF CON1 > 99 THEN PMPS2RUN = 2
36 IF CON1 < 1 THEN PMPS2RUN = 0
40 IF VAR1 AND PMPS2RUN = 1 THEN START PUMP1 ELSE STOP PUMP1
42 IF NOT VAR1 AND PMPS2RUN = 1 THEN START PUMP2 ELSE STOP PUMP2
44 IF PMPS2RUN = 2 THEN START PUMP1 , START PUMP2

50 REM ADD FOR ROTATION UPON PUMP FAILURE
60 DALARM PUMP1 AND NOT STATUS-1 , 300 , PUMP #1 FAILURE!
65 ON-ALARM 90

70 DALARM PUMP2 AND NOT STATUS-2 , 300 , PUMP #2 FAILURE!
75 ON-ALARM 90

80 END

90 REM ROTATE LEAD/LAG BASED ON FAILURE
95 IF VAR1 THEN STOP VAR1 ELSE START VAR1

```

**Program explanation**

**CON1** PID controller 1 is set up with the measured media and the setpoint of that media. For example, if the pumps are controlling pressure, the PID control loop would contain the input from the pressure sensor and the pressure setpoint. The value of CON1 represents the demand.

**PMPS2RUN** store the number of pumps to run based on the demand.

**VAR1** Variable 1 is used as the lead pump flag. When VAR1 is ON (1), pump 1 is lead. When VAR1 is OFF (0) pump 2 is lead.

**Other options**

To rotate pumps Lead/Lag cycle based on actual runtime:

Program the status of each pump in a runtime log and modify line 20 to read:

*Example:* 20 IF RT1 < RT2 THEN START VAR1 ELSE STOP VAR1

To specify a certain difference in runtime hours, use the following:

*Example:* 20 IF RT1 < RT2 - 200 THEN START VAR1 IF RT2 < RT1 - 200  
THEN STOP VAR1

These lines look for a difference of at least 200 hours between RT1 and RT2 before switching the lead/lag order.

**Three unit lead/lag**

The following program s three devices to run in order based on accumulated runtime. The the unit with the lowest runtime starts first, the unit with the most run time starts last, and the remaining unit is placed in the middle of the sequence. The arrangement is checked and modified if necessary once each week.

*Example:*

```

10 IF CON1 > 25 THEN START A
20 IF CON1 > 65 THEN START B
30 IF CON1 > 95 THEN START C
40 IF CON1 < 70 THEN STOP C
50 IF CON1 < 40 THEN STOP B
60 IF CON1 < 5 THEN STOP A

70 IF+ DOW = TUE AND TIME > 9:00:00 THEN 80 ELSE 150

80 IF RT1 = MIN( RT1 , RT2 , RT3 ) THEN SEQ = 100 , N = 1 , GOTO
110
90 IF RT2 = MIN( RT1 , RT2 , RT3 ) THEN SEQ = 200 , N = 2 ELSE
SEQ = 300 , N = 3
100 IF RT1 = MAX( RT1 , RT2 , RT3 ) THEN SEQ = SEQ + 1 , N = N +
1 , GOTO 120
110 IF RT2 = MAX( RT1 , RT2 , RT3 ) THEN SEQ = SEQ + 2 , N = N +
2 ELSE SEQ = SEQ + 3 , N = N + 3
120 IF N = 3 THEN SEQ = SEQ + 30 , GOTO 150
130 IF N = 4 THEN SEQ = SEQ + 20 , GOTO 150
140 IF N = 5 THEN SEQ = SEQ + 10 ELSE SEQ = 123 : REM DEFAULT
SEQUENCE

150 LEAD = SEQ \ 100 : X = SEQ - LEAD * 100
160 LAG1 = X \ 10 : LAG2 = X - LAG1 * 10

170 PUMP1 = A * ( 1 = LEAD ) + B * ( 1 = LAG1 ) + C * ( 1 = LAG2
)
180 PUMP2 = A * ( 2 = LEAD ) + B * ( 2 = LAG1 ) + C * ( 2 = LAG2
)
190 PUMP3 = A * ( 3 = LEAD ) + B * ( 3 = LAG1 ) + C * ( 3 = LAG2
)
200 END

```

To use this program, set up the run times to monitor each digital output in the sequence. In this example *PUMP1* is entered for the first runtime, *PUMP2* is the second runtime, and *PUMP3* is entered as the third runtime. Create variables *LEAD*, *LAG1*, and *LAG2* to store the number of the pump that will hold each position in the sequence. Create a variable labeled *SEQ* to store the sequence that will be used to run the equipment. The value of variable *SEQ* can be set to manual and you can enter your own order (for instance, if maintenance needs to be performed on a unit that does not have the maximum run time). Select range "0" in the units column for all four of these variables.

**Lines 10 through 60** take the analog information from a PID controller and choose a range in which each unit will be called to run. The controller may be set up to control pressure at a certain point in the building. As the pressure falls below set point, the PID controller output increases to bring on additional pumps. The local Variable "A" is ON when the lead pump is required to run. Variable "B" is on when the first lag pump is needed, and so on.

**Line 70** causes the sequence to change once per week, on Tuesday at 9:00 AM. Set this for a convenient time when operators will be on duty to catch unit failure alarms.

**Lines 80 through 140** sets the sequence with the smallest runtime unit up first and the longest runtime last. The unit with the greatest run time will be last in the sequence.

**Lines 150 through 160** break the SEQ Variable into the specific unit variables of Lead, Lag1, and Lag2. If the SEQ is "321", these lines of code make the Variable LEAD = 3, LAG1 = 2, LAG2 = 1. SEQ can be set manually and this section of the program will decipher the correct units to run.

**Lines 170 through 190** assigns the appropriate output based on sequence and demand. In line 170, if pump 1 is lead, then the statement "1 = LEAD" is a true and equals 1. When "A" is ON the pump is called to run. The rest of the equation is ignored unless pump 1 is other than lead.

### **Simplified runtime order**

Choosing a unit to run based on the exact runtime order is somewhat more difficult than necessary in most applications. With more than three units to sequence the programming becomes rather complicated in a Tier 2 controller. The simplified runtime order program uses a more basic method to set up the run sequence and this general method will be employed in programs with higher equipment counts.

The following program chooses one of four sequences to use in prioritizing equipment staging. The sequence is chosen based on the unit with the greatest run time. The unit with the most run time is sent to the end of the sequence once each week.

*Example:*

```
10 IF CON1 > 25 THEN START A
20 IF CON1 > 65 THEN START B
30 IF CON1 > 95 THEN START C
40 IF CON1 < 70 THEN STOP C
50 IF CON1 < 40 THEN STOP B
60 IF CON1 < 5 THEN STOP A

70 IF+ TIME > 9:00:00 THEN 80 ELSE 100

80 IF RT1 = MAX( RT1 , RT2 , RT3 ) THEN SEQ = 231 , 100
90 IF RT2 = MAX( RT1 , RT2 , RT3 ) THEN SEQ = 312 ELSE SEQ = 123

100 LEAD = SEQ \ 100 : X = SEQ - LEAD * 100
110 LAG1 = X \ 10
120 LAG2 = X - LAG1 X 10

130 PUMP1 = A * ( 1 = LEAD ) + B * ( 1 = LAG1 ) + C * ( 1 = LAG2
)
140 PUMP2 = A * ( 2 = LEAD ) + B * ( 2 = LAG1 ) + C * ( 2 = LAG2
)
150 PUMP3 = A * ( 3 = LEAD ) + B * ( 3 = LAG1 ) + C * ( 3 = LAG2
)
160 END
```

This program only verifies that the weekly sequence is chosen so that the highest run time unit is the last one called to run. It is possible that the pump with the second highest run time will be lead but the program will work to keep the runtimes equal (within a week). Creating a program to keep the runtimes in exact order is more difficult with the programming size and command limitations in the Tier 2 controllers. However, this can be done in the KMD-5210 Lan Controller if one is available.

To use this program, set up the run times to monitor each digital output in the sequence. In this example *PUMP1* is entered for the first runtime, *PUMP2* is the second runtime, and *PUMP3* is entered as the third runtime. Create variables *LEAD*, *LAG1*, and *LAG2* to store the number of the pump that will hold each position in the sequence. Create a variable labeled *SEQ* to store the sequence that will be used to run the equipment. The value of variable *SEQ* can be set to manual and you can enter your own order. For instance, if maintenance needs to be performed on pump 1 when it does not have the maximum run time, set *SEQ* to "231" to make pump 1 the second lag pump. Select *Unused* in the units column for all five of these variables.

**Lines 10 through 60** take the analog information from a PID controller and choose a range in which each unit will be called to run. The controller may be set up to control pressure at a certain point in the building. As the pressure falls below set point, the PID controller output increases to bring on additional pumps. The local variable "A" is ON when the lead pump is required to run. Variable "B" is on when the first lag pump is needed.

**Line 70** causes the sequence to change once per week, on Tuesday at 9:00 AM. Set this for a convenient time when operators will be on duty to catch unit failure alarms.

**Lines 80 through 90** sets the sequence with the smallest runtime unit up first and the longest runtime last. The unit with the greatest run time will be last in the sequence.

**Lines 100 through 120** break the *SEQ* Variable into the specific unit variables of Lead, Lag1, and Lag2. If the *SEQ* is "321", these lines of code make the Variable *LEAD* = 3, *LAG1* = 2, *LAG2* = 1. *SEQ* can be set manually and this section of the program will decipher the correct units to run.

**Lines 130 through 150** assigns the appropriate output based on sequence and demand. In line 130, if pump 1 is lead, then the statement "1 = LEAD" is a true and equals 1. When "A" is ON the pump is called to run. The rest of the equation is ignored unless pump 1 is other than lead.

## Four unit lead/lag

The following program chooses one of four sequences to prioritize equipment staging. The sequence is chosen based on the unit with the greatest run time. The unit with the most run time is sent to the end of the sequence once each week.

*Example:*

```

10 IF 1-CON1 > 20 THEN START A
20 IF 1-CON1 > 45 THEN START B
30 IF 1-CON1 > 70 THEN START C
40 IF 1-CON1 > 95 THEN START D
50 IF 1-CON1 < 80 THEN STOP D
60 IF 1-CON1 < 55 THEN STOP C
70 IF 1-CON1 < 30 THEN STOP B
80 IF 1-CON1 < 5 THEN STOP A
90 IF+ DOW = TUE AND TIME > 9:00:00 THEN 100 ELSE 130
100 IF 1-RT1 = MAX( 1-RT1 , 1-RT2 , 1-RT3 , 1-RT4 ) THEN SEQ =
2341 , 130
110 IF 1-RT2 = MAX( 1-RT1 , 1-RT2 , 1-RT3 , 1-RT4 ) THEN SEQ =
3412 , 130
120 IF 1-RT3 = MAX( 1-RT1 , 1-RT2 , 1-RT3 , 1-RT4 ) THEN SEQ =
4123 ELSE SEQ = 1234
130 LEAD = SEQ \ 1000 : X = SEQ - LEAD * 1000
140 LAG1 = X \ 100 : X = X - LAG1 * 100
150 LAG2 = X \ 10
160 LAG3 = X - LAG2 * 10
170 PUMP1 = A * ( 1 = LEAD ) + B * ( 1 = LAG1 ) + C * ( 1 = LAG2
) + D * ( 1 = LAG3 )
180 PUMP2 = A * ( 2 = LEAD ) + B * ( 2 = LAG1 ) + C * ( 2 = LAG2
) + D * ( 2 = LAG3 )
190 PUMP3 = A * ( 3 = LEAD ) + B * ( 3 = LAG1 ) + C * ( 3 = LAG2
) + D * ( 3 = LAG3 )
200 PUMP4 = A * ( 4 = LEAD ) + B * ( 4 = LAG1 ) + C * ( 4 = LAG2
) + D * ( 4 = LAG3 )

210 END

```

To use this program, set up the run times to monitor each digital output in the sequence. In this example *PUMP1* is entered for the first runtime, *PUMP2* is the second runtime, and *PUMP3* and *PUMP4* are entered as the third and fourth runtimes. Create variables *LEAD*, *LAG1*, *LAG2*, and *LAG3* to store the number of the pump that will hold each position in the sequence. Create a variable labeled *SEQ* to store the sequence that will be used to run the equipment. The value of variable *SEQ* can then be set to manual and you can enter your own order. For instance, if maintenance needs to be performed on a unit that does not have the maximum run time. Select range "0" in the units column for all five of these variables.

**Lines 10 through 80** take the analog information from a PID controller and choose a range in which each unit will be called to run. The controller may be set up to control pressure at a certain point in the building. As the pressure falls below set point, the PID controller output increases to bring on additional pumps. The local variable "A" is ON when the lead pump is required to run. Variable "B" is on when the first lag pump is needed.

**Line 90** causes the sequence to change once per week, on Tuesday at 9:00 AM. Set this for a convenient time when operators will be on duty to catch unit failure alarms.

**Lines 100 through 120** sets the sequence based on the longest runtime. The unit with the greatest run time will be last in the sequence.

**Lines 130 through 160** break the SEQ Variable into the specific unit variables of Lead, Lag1, Lag2, and Lag3. If the SEQ is "4321", these lines of code make the Variable LEAD = 4, LAG1 = 3, LAG2 = 2, and LAG3 = 1.

**Lines 170 through 200** assigns the appropriate output based on sequence and demand. In line 170, if pump 1 is lead, then "1 = LEAD" is a true statement and equals 1. When "A" is ON the pump is called to run. The rest of the equation is ignored unless pump 1 is other than lead.

This program only makes sure that the weekly sequence is chosen so that the highest run time pump is the last one called to run. It is possible that the pump with the second highest run time will be lead but the program will work to keep the runtimes equal (within a week). Creating a program to keep the runtimes in exact order is difficult with the programming size and command limitations in the Tier 2 controllers.

## **Expanding for more than four units**

If you have six or more pieces of equipment to use in a sequence the programming becomes cumbersome and will likely need to be separated into more than a one program area. This task can also be simplified by moving the programming into a KMD-5210 Lan Controller.



## Application Note—AN0604A

---

### Optimal start/stop routine

The program in this application note describes a method to adjust the start-up routine for building warm-up based on multiple factors.

---

We are assuming the routine will begin sometime after midnight (not before). A building performance is calculated in TIME per DEGREE temperature rise each day the optimum start routine is used. A required PRESTART time is calculated based on the last few day's performance and the result may be further influenced by large changes in outdoor air temperature or if the building has not been in the "occupied" mode for longer than 24 hours (such as weekends). Routines for optimum STOP could also be based on a similar format.

*Example:*

```

5 W = TIME-ON( C )
10 VAR14 = WS-ON( 1 , 1 ) : REM Today's first scheduled
occupancy time
20 IF TIME > VAR14 AND NOT C THEN GOTO 90
30 IF VAR14 = 0:00:00 THEN STOP C , GOTO 90
40 IF+ TIME > VAR14 - 3:01:00 THEN GOSUB 120
50 IF+ TIME > VAR14 - PRESTART THEN GOSUB 230
55 IF C THEN GOTO 60 ELSE 70
60 IF+ ABS( BLDG/SP - BLDG/TMP ) < 1 THEN GOSUB 260
70 IF- WS1 THEN STOP OCCUPANT , STOP C
80 END
90 REM STANDARD ROUTINE WITHOUT OPTIMUM START
100 IF WS1 THEN START OCCUPANT ELSE STOP OCCUPANT
110 END
120 REM CALCULATE NEW PRESTART TIME
130 Z = ( 3 * 1-RATE + 2 * 2-RATE + 3-RATE ) / 6
140 PRESTART = ABS( BLDG/SP - BLDG/TMP ) * Z
150 REM OUTDOOR AIR TEMP INFLUENCE
160 IF OUT-TP - LAST-OAT > 15 THEN PRESTART = PRESTART * 0.9
170 IF OUT-TP - LAST-OAT < -15 THEN PRESTART = PRESTART * 1.1
175 LAST-OAT = OUT-TP : REM RECORDS TEMP TO COMPARE WITH
TOMORROW
180 REM WEEKEND OR HOLIDAY INFLUENCE
190 IF TIME-OFF( OCCUPANT ) > 23:59:59 THEN PRESTART = PRESTART
* 1.15
200 IF PRESTART < 0:15:00 THEN PRESTART = 0:15:00
210 IF PRESTART > 3:00:00 THEN PRESTART = 3:00:00
220 RETURN
230 REM BEGINS OPTIMUM START ROUTINE
240 START OCCUPANT : START C : A = BLDG/TMP : B = OUT-TP
250 RETURN
260 REM OPTIMUM CYCLE COMPLETED SO CALCULATE TODAYS PERFORMANCE
& RESET
265 D = ABS( BLDG/TMP - A ) : IF D = 0 THEN D = 1
270 3-RATE = 2-RATE : 2-RATE = 1-RATE : 1-RATE = W / D
280 RETURN

```

## Program description

**Line 10** This statement serves to look into the weekly routine WR1 to see what time occupancy is expected on this day. ( first ON time listed ) It is possible to look at the second, third, or fourth time listed by changing the statement to read "WS-ON( 1 , 4 ) " See the KMD manual for further details / explanations.

**Line 20** If current time is past occupancy time and the optimum start routine is not running all this sequence is skipped. ( jumps to line 90 )

**Line 30** If time returned by the WS-ON statement is zeros then occupancy is not scheduled for today. Entire sequence will be skipped.

**Line 40** At 3hrs 1min before scheduled occupancy we are going to calculate an expected PRESTART time required to reach setpoint before the first occupant arrives. This value may be changed as required.

**Line 50** If the time between now and scheduled occupancy is less than the time required to warm the building we will start the warm-up sequence.

**Line 60** Once the OPTIMUM START routine is begun it monitors building temp to see when setpoint is reached (within 1 deg.). Once reached a new performance figure is calculated. ( See comments below )

**Line 70** This line assures that if the weekly routine occupancy time gets cancelled it will also cancel this warm-up sequence.

**Line 100** Program jumps to this line when the OPTIMUM START is not valid.

**Line 130** "Z" is a weighted average performance rate in TIME per DEGREE. We arbitrarily chose 3 times today's rate, 2 times yesterdays rate and 1 times rate the day before.

**Line 140** A new PRESTART TIME is determined by knowing how many degrees the building must be raised and how fast we expect temp to rise.

**Line 160** We then influence this time plus or minus 10% if the outside temperature today is more than 15 degrees different than it was last time we calculated a performance figure. We also add in a 15% factor if the building has not been warmed-up ( occupied ) for longer than 24 hours to compensate for weekends or holiday breaks.

**Line 240** When the OPTIMUM START sequence is begun outside and inside temperature stored for later reference.

**Line 270** After the warm-up sequence has been successfully completed each rate is stepped back a day while a new rate is calculated for today. These rates are determined by knowing how much time was required to raise the temperature how many degrees. The rate is then stored as a function of TIME per DEGREE.

## Comments

When writing programming for Control Basic keep in mind an error will occur if the sequence attempts to divide by zero ( i.e., logic error ). In this program we have prevented this scenario by including the statements in lines 265 and 270 to eliminate the possibility "D" could ever be zero!

Also, we have used many "local" variables ( a-z ) for convenience. It must be remembered these local variables will always default back to a value of "zero" or "off" each time the program is SENT or upon restore after a power failure. To eliminate this possibility the local variables A, B, C, and D could be replaced with standard system variables such as VAR1, VAR2, etc.

Once the OPTIMUM START ROUTINE is initiated and it begins to warm-up ( or Cool down ) the building the program is waiting for conditions to become "satisfied". Our example uses 1 degree within set point as a "satisfied" condition. Depending upon how these temperatures are measured, such as by an average, and what proportional bands are used in the control sequences it may be necessary to increase this value. Keep in mind that as set points are attained a normal control sequence will "throttle-down" so as to not overshoot the intended set point - this naturally increases the time it takes to actually reach set point and may or may not want to be included in the performance calculation of this OPTIMUM START ROUTINE.

Line 55 looks to see if the optimum start sequence has been activated. If has line 60 will look to see if the building temperature is within one degree of set point - at which time the program will GOSUB to a section calculating a performance figure.

This program can be modified in many ways to suit a particular application.



## Application Note—AN0909A

---

### A simplified optimum start program

This application note describes a method for calculating the optimum startup time based on the previous days warm-up period and the current outside air temperature. Example programs for both KMD and BACnet controllers are included.

---

The Control Basic programs in this application note calculate an optimum prestart period based on the following:

- ◆ Scheduled building occupancy
- ◆ The current outside air temperature
- ◆ The rate of warm-up from the previous day.

These programs are similar to the program in Application Note AN0604A but use only the previous days results instead of the results from the previous three days. Both KMD and BACnet Control Basic programs are listed beginning on page 4-37. In addition to the program listings, the configuration for the variables and value objects are listed beginning on page 4-33.

#### About the program

This program controls the state of the variable *Occupant*. This variable is True (1) or active for the following conditions:

- ◆ During a calculated prestart period
- ◆ When the value of the schedule that controls the routine is True (1).

Starting equipment or changing a setpoint temperature can then be based on the state of *Occupant*.

#### Applicable models

The programs described in this application note can be operated on both KMD and BACnet controllers from KMC Controls, Inc. However, for BACnet controllers that use Next Generation Basic the program listings in this application note will have to be converted to the Next Generation format.

## Related materials

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ The installation and operation guide for the controller on which the program will run.
- ◆ Installation and operation guides for other equipment such as temperature sensors or a NetSensor that will be used with the controller.
- ◆ Reference or user's manuals for BACstage or WinControl XL Plus.

## Setting up the program

Before running the program, set up the following points and objects.

### Enter a weekly schedule

Enter a schedule in Weekly Schedule #1 in a KMD controller or Schedule Object #1 in a BACnet controller for the time of day that people will occupy the building. The only times used by the program are the first *On* and *Off* periods in either type of schedule. Enter an *On* and *Off* period for each day of the entire week. In addition to entering times in the schedule, set the object reference type in the BACnet schedule to *binary*.

To use a schedule other than Schedule Object #1 or Weekly Schedule #1, modify lines 100 and 110.

### Assign outside air temperature

Outside air temperature—which is stored in OAT—influences the warm up period. Assign the value from a sensor connected to one of the controller's inputs or from an off-panel point.

### Monitoring the building temperature

Assign values for the building's temperature (BldgTemp) from either a sensor connected to one of the controller's inputs or from an off-panel point in another controller. The building temperature can also be assigned to the temperature sensor in a NetSensor connected to the controller.

### Assign a building setpoint

Values for the building's setpoint (BldgSp) can be supplied from a sensor connected to one of the controller's inputs, an off-panel point from another controller or a fixed value entered for AV3 or VAR3. The setpoint can also be assigned to the setpoint button in a NetSensor connected to the controller.

### Set values for PreStartMinimum and PreStartMaximum

PreStartMinimum and PreStartMaximum set the limits for the prestart period. Typically these values are set to 15 minutes and 3 hours. For BACnet controllers, enter the time in seconds (900 and 10,800 seconds). Set the BACnet objects to Out of Service and KMD points to Manual for these variables.

## Variables and value object configuration

The assignments of BACnet value objects and KMD variables are similar for programs in both types of controllers.

### Analog value objects and points

**AV1 or VAR1—ScheduleOnTime** Stores the time of day at which the object or point will change to On or True based on the schedule designated in line 100. In BACnet controllers time is displayed as seconds after midnight; KMD controllers shows time in an hours:minutes:seconds format. See [Enter a weekly schedule](#) on page 4-32.

**AV2 or VAR2—PreStartPeriod** A calculated value. This variable will never be longer than PreStartMimum or shorter than PreStartMinimum. The program will change the building to occupancy status at the time stored in ScheduledOnTime minus the value of PreStartPeriod.

**AV3 or VAR3—BuildingSetpoint** The setpoint temperature when the build is occupied. This can be directly entered as a fixed number or supplied from another point or object through Control Basic. See [Assign a building setpoint](#) on page 4-32.

**AV4 or VAR4—BuildingTemperature** The actual building temperature. This must be supplied from another point or object through Control Basic. See [Monitoring the building temperature](#) on page 4-32.

**AV5 or VAR5—RateWarmUp** A calculated value. This variable is calculated at the end of each prestart period and saved as seconds per degree of temperature change.

**AV6 or VAR6—PreStartMinimum** This is a fixed value. The variable PreStartPeriod will never be shorter than PreStartMinium. See [Set values for PreStartMinimum and PreStartMaximum](#) on page 4-32.

**AV7 or VAR7—PreStartMaximum** This is a fixed value. The variable PreStartPeriod will never be longer than PreStartMaximum. See [Set values for PreStartMinimum and PreStartMaximum](#) on page 4-32.

**AV8 or VAR8—LastOutsideAirTemp** At the end of the warm up period, this variable stores the actual outside air temperature to use on the following day to calculate the prestart period.

**AV9 or VAR9—OutsideAirTemperature** The actual outside air temperature. This must be supplied from another point or object through Control Basic. See [Assign outside air temperature](#) on page 4-32.

**AV10 or VAR10—ScheduleOffTime** Stores the time of day at which the object or point will change to Off or False based on the schedule designated in line 110. In BACnet controllers time is displayed as seconds after midnight; KMD controllers show time in an hours:minutes:seconds format. See [Enter a weekly schedule](#) on page 4-32.

**Table 1 BACnet analog value objects configuration**

| Value object | Description           | Name         | Present Value | Units   | Out of service |
|--------------|-----------------------|--------------|---------------|---------|----------------|
| AV1          | ScheduleOnTime        | SchedOnTime  |               | Seconds |                |
| AV2          | PreStartPeriod        | Prestart     |               | Seconds |                |
| AV3          | BuildingSetpoint      | BldgSp       |               | °F      |                |
| AV4          | BuildingTemperature   | BldgTemp     |               | °F      |                |
| AV5          | RateWarmUp            | Rate         |               | Seconds |                |
| AV6          | PreStartMinimum       | PreStartMin  | 900           | Seconds | X              |
| AV7          | PreStartMaximum       | PreStartMax  | 10800         | Seconds | X              |
| AV8          | LastOutsideAirTemp    | LastOAT      |               | Seconds |                |
| AV9          | OutsideAirTemperature | OAT          |               | Seconds |                |
| AV10         | ScheduleOffTime       | SchedOffTime |               | Seconds |                |

**Table 2 KMD analog variable points configuration**

| Point | Description           | Label        | Value    | Units | Manual |
|-------|-----------------------|--------------|----------|-------|--------|
| VAR1  | ScheduleOnTime        | SchedOnTime  |          | Time  |        |
| VAR2  | PreStartPeriod        | Prestart     |          | Time  |        |
| VAR3  | Building Setpoint     | BldgSp       |          | °F    |        |
| VAR4  | Building Temperature  | BldgTemp     |          | °F    |        |
| VAR5  | RateWarmUp            | Rate         |          | Time  |        |
| VAR6  | PreStartMinimum       | PreStartMin  | 00:15:00 | Time  | X      |
| VAR7  | PreStartMaximum       | PreStartMax  | 3:00:00  | Time  | X      |
| VAR8  | LastOutsideAirTemp    | LastOAT      |          | Time  |        |
| VAR9  | OutsideAirTemperature | OAT          |          | Time  |        |
| VAR10 | ScheduleOffTime       | SchedOffTime |          | Time  |        |

**Binary value objects and points**

**BV1 or VAR31—Occupant** This variable holds the occupancy status of the building.

**Table 3 BACnet binary value objects**

| Value object | Description     | Name     | Present Value | Active Inactive text | Units           |
|--------------|-----------------|----------|---------------|----------------------|-----------------|
| BV1          | OccupancyStatus | Occupant |               |                      | Unocc./Occupied |

**Table 4 KMD binary variable**

| Point | Description     | Label    | Present Value | Units           |
|-------|-----------------|----------|---------------|-----------------|
| VAR31 | OccupancyStatus | Occupant |               | Unocc./Occupied |

## Local variables

In addition to the variables and value objects listed in the previous section, the following local variables are used at several places in the program:

**Table 5 Local variables**

| Variable | Description  |
|----------|--|
| A        | Stores the value of the build temperature at the beginning of the prestart period.                                   |
| C        | Flags that the program is in the prestart period. The time that C is On is used to calculate Rate.                   |
| D        | The difference of the building temperature between the beginning of the prestart period and the end.                 |
| S        | A flag that is True when either the schedule indicates the building is occupied or during the prestart period.       |
| T        | At the end of the prestart period this variable stores the difference between the building temperature and setpoint. |

## Program description

This is a description of the programs in the section [Program listings](#) on page 4-37.

**Lines 100 and 110** These statements read and store the times that the building will change occupancy status.

**Line 120** If the current time is latter than the first scheduled occupancy time, the optimum start routine is canceled and program jumps to line 210.

**Line 130** If the time stored in ScheduleOnTime equals zero then building occupancy is not scheduled for the current day. The local variables S and C are cleared and the program jumps to line 210.

**Line 140** If the actual time is later than the value in ScheduleOnTime, the occupancy flag is set to true, the program jumps to line 210.

**Line 160** Before scheduled occupancy, the program begins running the subroutine to calculate an expected prestart time.

**Line 180** If the actual time is later than the scheduled occupancy minus the calculated prestart interval than the program runs the subroutine to begin the optimum start sequence.

**Line 200** Once the optimum start routine begins, the program monitors building temperature to determine when the temperature is within 1 degree of setpoint. Once setpoint is reached a new value for RateWarmUp is calculated.

**Line 210** This is the only exit from the program. Occupancy status is changed based on the local variable S.

**Line 240** A new prestart is calculated based on the required change in building temperature and the rate of change from the previous day. The new prestart time is limited by the variables PreStartMinimum and PreStartMaximum.

**Line 250** The absolute value of the required temperature change is calculated and stored in local variable T.

**Line 260** The prestart interval is calculated from the required temperature change and the rate calculated at the end of the previous day's prestart interval.

**Line 280** An influence for outside air temperature is applied based on the difference between the previous day's outside air temperature and the current outside temperature.

**Lines 300 and 310** The prestart interval is limited to be no longer than PreStartMaximum or shorter than PreStartMinimum.

**Line 340** The flags are set to start the optimum start routine. Local variable C is true only when the program is in the prestart period and is used to time how long it takes for the building to reach setpoint. Local variable "S indicates the prestart routine has started and building occupancy should change.

**Line 380** A new prestart interval rate is calculated from the change in building temperature and the time that local variable C was true. The result, in seconds per degree, is stored in variable RateWarmUp.

**Line 390** The change in building temperature is calculated from the temperature at the beginning of the prestart interval (local variable A) and the current building temperature. The result is stored in local variable D.

**Line 400** The local variable C is set to false. This stops the timer on line 390 and resets it for the next days rate calculation.

**Line 410** Current outside air temperature is stored in LastOAT to use the following day to calculate a prestart interval.

**Program listings**

The following programs can be copied from a PDF file and pasted into the Control Basic editors of either WinControl XL Plus or BACstage.

**Control Basic for KMD controllers**

```

100 IF INTERVAL( 0:01:00 ) THEN VAR1 = WS-ON( 1 , 1 )
110 IF INTERVAL( 0:01:00 ) THEN VAR10 = WS-OFF( 1 , 1 )
120 IF TIME > VAR10 THEN STOP S , GOTO 210
130 IF VAR1 = 0 THEN STOP C , STOP S , GOTO 210 : REM C is on
during the OPSTART calc, used to time the event
140 IF TIME > VAR1 AND NOT C THEN START S , GOTO 210 : REM S is
on when the equipment should run
150 REM TIME TO CALC PRESTART?
160 IF NOT VAR31 AND TIME < VAR10 AND TIME >= VAR1 - VAR7 THEN
GOSUB 240
170 REM TIME TO RUN OSTART SEQ?
180 IF NOT C AND NOT VAR31 AND TIME > VAR1 - VAR2 AND TIME <
VAR10 THEN GOSUB 340
190 REM TIME TO STOP OSTART SEQ?
200 IF+ C AND ABS( VAR3 - VAR4 ) < 1 OR TIME > VAR1 + 7200 THEN
GOSUB 370
210 REM STANDARD SCHEDULE
220 IF S THEN START VAR31 ELSE STOP VAR31
230 END
240 REM *** 1 *** CALCULATE NEW Prestart TIME
250 T = ABS( VAR3 - VAR4 )
260 VAR2 = ( T - 1 ) * ( T > 1 ) * VAR5
270 REM OUTDOOR AIR TEMP INFLUENCE
280 IF VAR9 < VAR4 THEN VAR2 = VAR2 * ( 100 - ( VAR9 - VAR8 ) )
/ 100 ELSE VAR2 = VAR2 * ( 100 + VAR9 - VAR8 ) / 100
290 REM Prestart Limits
300 IF VAR2 < VAR6 THEN VAR2 = VAR6
310 IF VAR2 > VAR7 THEN VAR2 = VAR7
330 RETURN
340 REM *** 2 *** Start OSTART Seq
350 START S : START C : A = VAR4 : REM Start equipment and store
start values
360 RETURN
370 REM *** 3 *** OSTART CYCLE COMPLETE CALCULATE PERFORMANCE
380 D = ABS( VAR4 - A ) : IF D = 0 THEN D = 1
390 VAR5 = TIME-ON( C ) / D
400 STOP C
410 VAR8 = VAR9 : REM RECORD OAT TO COMPARE WITH TOMORROW
420 RETURN

```

### Control Basic for BACnet controllers

```
100 IF INTERVAL( 0:05:00 ) THEN SchedOnTime = SCHEDON( 1 , 0 )
110 IF INTERVAL( 0:05:00 ) THEN SchedOffTime = SCHEDOFF( 1 , 0 )
120 IF TIME > SchedOffTime THEN STOP S , GOTO 210
130 IF SchedOnTime = 0 THEN STOP C , STOP S , GOTO 210 : REM C
is on during the OPSTART calc, used to time the event
140 IF TIME > SchedOnTime AND NOT C THEN START S , GOTO 210 :
REM S is on when the equipment should run

150 REM TIME TO CALC PRESTART?
160 IF NOT Occupant AND TIME < SchedOffTime AND TIME >=
SchedOnTime - PreStartMax THEN GOSUB 240

170 REM TIME TO RUN OSTART SEQ?
180 IF NOT C AND NOT Occupant AND TIME > SchedOnTime - Prestart
AND TIME < SchedOffTime THEN GOSUB 340

190 REM TIME TO STOP OSTART SEQ?
200 IF+ C AND ABS( BldgSp - BldgTemp ) < 1 OR TIME > SchedOnTime
+ 7200 THEN GOSUB 370

210 REM STANDARD SCHEDULE
220 IF S THEN START Occupant ELSE STOP Occupant
230 END

240 REM *** 1 *** CALCULATE NEW Prestart TIME
250 T = ABS( BldgSp - BldgTemp )
260 Prestart = ( T - 1 ) * ( T > 1 ) * Rate
270 REM OUTDOOR AIR TEMP INFLUENCE
280 IF OAT < BldgTemp THEN Prestart = Prestart * ( 100 - ( OAT -
LastOAT ) ) / 100 ELSE Prestart = Prestart * ( 100 + OAT -
LastOAT ) / 100
290 REM Prestart Limits
300 IF Prestart < PreStartMin THEN Prestart = PreStartMin
310 IF Prestart > PreStartMax THEN Prestart = PreStartMax
330 RETURN

340 REM *** 2 *** Start OSTART Seq
350 START S : START C : A = BldgTemp : REM Start equipment and
store start values
360 RETURN

370 REM *** 3 *** OSTART CYCLE COMPLETE CALCULATE PERFORMANCE
380 D = ABS( BldgTemp - A ) : IF D = 0 THEN D = 1
390 Rate = TIMEON( C ) / D
400 STOP C

410 LastOAT = OAT : REM RECORD OAT TO COMPARE WITH TOMORROW
420 RETURN
```

Application Note–AN0504F

---

## Programming override timers

This application note describes methods for starting override timers in a Control Basic program when a button on a thermostat is pressed.

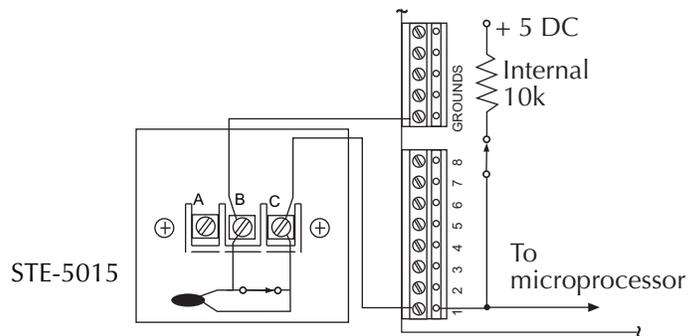
---

Programming equipment to automatically start or stop may also require a method for users to manually override the programming without using a computer. For example, users may require room lights operating from a weekly schedule or a fan starting in response to a heating and cooling cycle to operate independently for short periods. The programs in this application note describe two methods for starting an override timer in a KMC controller from either an STE series sensor or a NetSensor.

### Programming override timers with STE series thermostats

In Control Basic, the *Sensor-On* command recognizes a momentary shorting of a temperature sensor. This shorted condition can then be programmed to initiate a timed override as shown in the section [Program listing for STE-5015](#) on page 4-40.

Several models of the STE series room sensors include a push-button contact suitable for this type of operation. The circuit shown in Illustration 1 is a typical connection for a KMD-5015 temperature sensor with push button.



**Illustration 1** STE-5015 connected to a KMD-5801 controller

### Program listing for STE-5015

*Example:*

```
10 REM *** RUN THE FAN IN OCCUPIED MODE OR IN OVERRIDE ***
20 IF Occupied OR OVERRIDE THEN START FAN ELSE STOP FAN

30 REM *** IN UNOCCUPIED CHECK FOR PUSH BUTTON ***
40 IF+ SENSOR-ON( SPACE ) THEN START OVERRIDE , A = OVRDTIME

50 IF INTERVAL( 0:01:00 ) THEN A = A - 0:01:00
60 IF A <= 0 THEN STOP OVERRIDE , A = 0

70 REM TO SEE HOW MUCH TIME IS LEFT ON OVERRIDE, ADD THE
FOLLOWING LINE
80 TIMELEFT = A
90 END
```

### WinControl XL settings

**OCCUPIED** Assigned to weekly schedule #1 (WS1) or any other weekly schedule.

**OVERRIDE** is a digital variable with units configured as *OFF/ON*.

**FAN** Is a digital output with units configured for *Start/Stop*.

**OVRDTIME** is a variable configured with *Time* as units. The period of the override is set by this variable.

**SPACE** This is an analog input configured for either degrees Fahrenheit or degrees Celsius.

**TIMELEFT** is a variable configured with *Time* as units. This variable is optional and is used only to display the time of the override remaining.

### Programm operation

Output *FAN* (or other devices being controlled) starts when either the weekly schedule (WS1) or the override are activated. Each time the button is pressed the override time is reset to the period specified by *OVRDTIME*.

The time remaining on the override can be viewed as variable *TIMELEFT* and which then counts down in one-minute increments.

### Adding a user selected override period

To create a user selectable override based on the number of times the button is pushed, replace Line 40 with the lines below:

*Example:*

```
45 IF+ SENSOR-ON( SPACE ) THEN START OVERRIDE , A = A + OVRDTIME
46 IF OVRDTIME > MAXTIME THEN OVRDTIME = MAXTIME
```

Line 45 increases the override time each time the button is pressed. Line 46 limits the override time to the value of the variable *MAXTIME* and is optional.

## Programming override timers with a NetSensor

The following program activates an override timer when a button on the thermostat is selected and changed from *OFF* to *ON*.

```

10 REM *** RUN THE FAN IN OCCUPIED MODE OR IN OVERRIDE ***
20 IF OCCUPIED OR OVERRIDE THEN START FAN ELSE STOP FAN
30 REM *** IN UNOCCUPIED CHECK FOR PUSH BUTTON ***
40 IF+ OVERRIDE THEN A = OVRDTIME
50 IF INTERVAL( 0:01:00 ) THEN A = A - 0:01:00
60 IF A <= 0 OR NOT OVERRIDE THEN STOP OVERRIDE , A = 0
70 TIMELEFT = A : REM ALLOWS VIEWING THE REMAINING TIME IN
OVERRIDE MODE
80 TIMENOW = TIME : REM ALLOWS DISPLAYING THE CURRENT SYSTEM
TIME
90 END
    
```

### WinControl XL setup

**OCCUPIED** Assigned to weekly schedule #1 (WS1) or any available weekly schedule.

**OVERRIDE** is a digital variable with units configured as *OFF/ON*.

**FAN** Is a digital output with units configured for *Start/Stop*.

**OVRDTIME** is a variable configured with *Time* as units. The period of the override is set by this variable.

**TIMELEFT** is a variable configured with *Time* as units. This variable is optional and is used only to display the time remaining of the override period.

**TIMENOW** is a variable configured with *Time* as units. This variable is associated with button 5 to display time on LED models of NetSensor.

**Table 1 Variable assignments**

| VAR | Description        | A/M    | Value    | Units               | Label    |
|-----|--------------------|--------|----------|---------------------|----------|
| 1   | ROOM TEMPERATURE   | Auto   | 74.3     | Degrees Fahrenheit  | ROOMTEMP |
| 2   | HEAT TEMP SETPOINT | Auto   | 72       | Degrees Fahrenheit  | HEAT-SP  |
| 3   | (Available)        |        |          |                     |          |
| 4   | (Available)        |        |          |                     |          |
| 5   | REALTIME_NOW       | Auto   | 14:39:35 | Time                | TIMENOW  |
| 6   | (Available)        |        |          |                     |          |
| 7   | HUMIDITY           | Auto   | 51       | % Relative Humidity |          |
| 8   | OVERRIDE_TIME      | Manual | 0:10:00  | Time                | OVRDTIME |
| 9   | TIME_LEFT_ON_TIMER | Auto   | 0:00:00  | Time                | TIMELEFT |
| 10  | USER_OVERRIDE      | Auto   | Stop     | Stop/Start          | OVERRIDE |
| 11  | MAXIMUM_TIME       | Manual | 1:00:00  | Time                | MAXTIME  |
| 12– | (Available)        |        |          |                     |          |

The button assignments in the Net Sensor screen are shown Table 2:

**Table 2 NetSensor assignments**

|   | <b>Point</b>       | <b>Value</b> | <b>Type</b> | <b>Range</b> | <b>Display</b> |
|---|--------------------|--------------|-------------|--------------|----------------|
| 1 | ROOM TEMPERATURE   | 74.3         | Analog      | °F           | Read Only      |
| 2 | HEAT TEMP SETPOINT | 72           | Analog      | 0 Decimal    | Read/Write     |
| 3 | (Available)        |              |             |              |                |
| 4 | USER_OVERRIDE      | 0            | Analog      | 0 Decimal    | Read/Write     |
| 5 | REALTIME_NOW       | 14:39        | Analog      | Time         | Read/Write     |
| 6 | (Available)        |              |             |              |                |
| 7 | HUMIDITY           | 52           | Analog      | 0 Decimal    | Read/Write     |
| 8 | (Available)        |              |             |              |                |

**Program operation**

Output *FAN* (or other devices being controlled) starts when either the weekly schedule (*WS1*) or the override are activated. In the unoccupied mode, when a user presses the *OVRD* button on the NetSensor and then the up arrow button, the variable *OVERRIDE* changes to *START*. Local variable *A* is set to the value specified in variable *OVRDTIME*. The time defined in variable *OVRDTIME* is then subtracted from *A* until the value of *A* is 0. The override routine then ends and the fan, or other equipment is turned off.

The variable *TIMELEFT* displays the time remaining on the override and counts down in one-minute increments.

## Application Note—AN0504G

## Working with pulse inputs



### Note

Pulse counting is not available on KMD-6000 series controllers.

KMD controllers have two pulse input units input for measuring pulse signals from utility meters to calculate electrical and gas consumption or demand.

- ◆ Use the analog unit *0-255 Pulse per Min.* for pulse rates up to 4 Hz.
- ◆ Use the analog unit *0 To 2-32 Counts* for pulse rates up to 8 Hz.

The following program reads a pulse meter and then stores the total power as kilowatt/hours as a variable.

*Example:*

```

10 REM *** IN5 IS TIED TO ELECTIC METER AND EACH PULSE * .108 =
   KW
20 TOTAL-KW = IN5 * .108

```

**IN5** is a universal input point set up with units of *0 To 2-32 Counts*.

**TOTAL-KW** is a variable with units *kWh*.

Each pulse received at the input increments the value by 1 (up to 4.29 billion counts). It is necessary to obtain a multiplier from the meter to allow conversion into real units. In the example the multiplier is .108. Each pulse received by the KMD board is equal to .108 kWh. The variable *TOTAL-KW* stores the total kilowatt consumption recorded by the input point.

At some regular interval you should clear the counter input. In the following example the counter input total is stored in a variable and the counter input is cleared on the first day of the year.

```

30 IF+ DOY = 1 THEN 40 ELSE END
40 VAR25 = IN5 : IN5 = 0
50 END

```

**VAR25** holds the accumulated count for the previous year.

## Calculating a running average for kilowatt demand

Some meters provide a pulse output equivalent to the rate of consumption. The following program takes a reading each 2.5 minutes and averages the values to get an average for a 15-minute period.

*Example:*

```
10 IF INTERVAL( 0:02:30 ) THEN 20 ELSE 80
20 A = B : B = C : C = D : D = E : E = F : F = DEMAND
80 IF POWER-LOSS OR DEMAND = 0 THEN A = DEMAND , B = DEMAND , C
= DEMAND , D = DEMAND , E = DEMAND
90 DEMANDAV = AVG( A , B , C , D , E , F )
100 END
```

**DEMAND** Analog Input (KW) set for the unit *0-255 Pulse per Min.*

**DEMANDAV** A variable to store the average demand over the past 15 minutes.

The local variables (A, B, C....) are held only in RAM memory by the controller. After a power loss the values of these variables become 0. The *Power-Loss* statement is used to keep a realistic average in case of power failure to the board.

## Application Note—AN0504H

---

Revision A

### Calculating boiler reset temperature

This application note describes four methods to calculate boiler reset temperature with programming in a KMC controller.

---

Four methods of resetting a hot water supply setpoint based on outdoor air temperature are explained in this application note. Although a hot water system is used in the examples, many of the principles apply to other applications in which a setpoint must change based on external influences.

The example programs require a value for outside air temperature. The source of the temperature value is beyond the scope of this application note. The result of the example application programs is a value for the reset temperature for boiler operation. Choosing the range for the reset temperature and boiler operation is also beyond the scope of this application note.

#### Applicable models

The programs described in this application note can be operated on both KMD and BACnet controllers from KMC Controls, Inc. However, for BACnet controllers that use Next Generation Basic the program listings in this application note will have to be converted to the Next Generation format.

#### Related materials

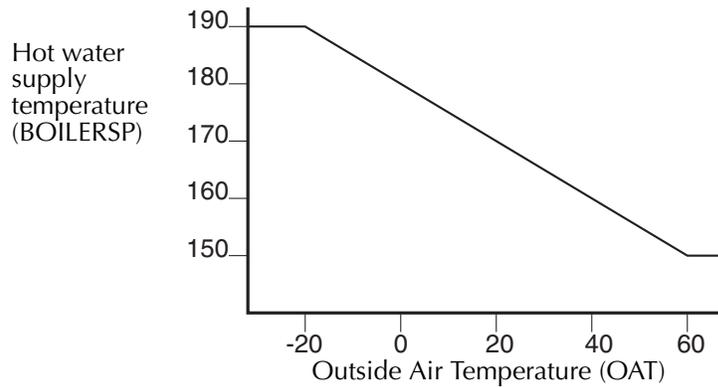
In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ The installation and operation guide for the controller on which the program will run.
- ◆ Installation and operation guides for other equipment such as temperature sensors that will be used with the controller.
- ◆ Reference manuals, user's manuals, or help for BACstage, WinControl XL Plus, or TotalControl.
- ◆ Application note AN0410A, *Measuring outside air temperature* covers methods for measuring outside air temperature.

## Boiler reset calculated from a lookup table

A table is a simple method to calculate boiler water reset temperature. Pairs of values that correspond to the temperature range are entered into a table. Control Basic is then used to look up and interpolate values for the reset temperature. If the temperature change is linear as shown in Illustration 1, only two rows of values are required for BACnet controllers. KMD controllers require a third row in the table to set the limit.

The line graph in Illustration 1 is a visual representation of [Table 1 on page 46](#)



**Illustration 1 The Table Method For resetting a setpoint**

Open a table and then in the first column, enter the operating range for the lowest and highest outside air temperatures. In the second column, enter the boiler reset temperatures that correspond to the outside air temperatures in the first column. In KMD controllers enter a third pair of values to set the lowest boiler reset temperature. The third pair is not required in BACnet controllers. The values that correspond to the graph in Illustration 1 are shown in the following tables.

**Table 1 Reset values for a KMD table**

| X Value | Y Value |
|---------|---------|
| -20     | 190     |
| 60      | 150     |
| 150     | 150     |

**Table 2 Reset values for a BACnet Control Basic table**

| X Value | Y Value |
|---------|---------|
| -20     | 190     |
| 60      | 150     |

Use the Control Basic *TBL* function to set the value of the variable *HWSsetpoint* to the output value of the table.

**KMD controllers**

```
10 HWSsetpoint = TBL( OutsideAirTemp , 1 )
```

**BACnet controllers**

```
10 AV12 = TBL( AV10 , 1 )
```

In the following example the setpoint changes only when the outside temperature changes more than 5° since the previous change to reset temperature. This prevents needless resetting of the hot water setpoint. The following example is Control Basic for a KMD Controller. The BACnet Control Basic is similar.

```
10 IF ABS( L - OAT ) < 5 THEN GOTO 20 ELSE HWSSP = TBL( OAT , 1 )
   , L = OAT
20 REM : Program continues here
```

**Calculating the setpoint with Control Basic**

If a table is not available, a reset temperature can be programmed in a Control Basic program. In the following example, the hot water setpoint *HWSSP* changes linearly based on the outside air temperature and four variables that set the limits. The limits are described in the section [Variables and value objects](#) on page 4-51.

The reset temperature changes only when the outside temperature changes more than 5° since the previous change to reset temperature. This prevents needless resetting of the hot water set point. Remove lines 10, 20, and 70 for continuous setpoint calculation.

The following Control Basic programs use one or more points or variables described in the section [Variables and value objects](#) on page 4-51.

**KMD program with mnemonics**

```
10 IF ABS( L - VAR10 ) > 5 THEN GOTO 30
20 GOTO 80
30 REM *** BOILER RESET ***
40 M = ( VAR14 - VAR13 ) / ( VAR15 - VAR16 )
50 B = VAR13 - M * VAR16
60 VAR12 = MIN( MAX( ( VAR10 - B ) / M , VAR16 ) , VAR15 )
70 L = VAR10
80 END
```

**KMD program with labels**

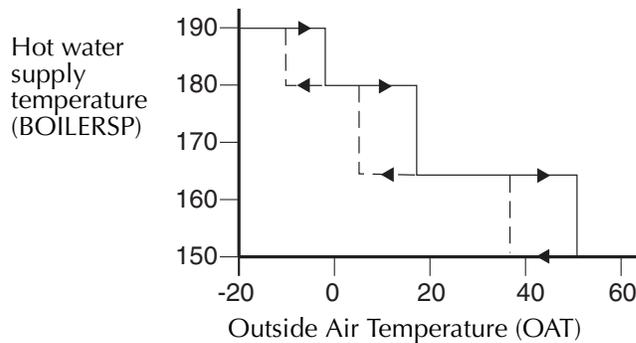
```
10 IF ABS( L - OAT ) > 5 THEN GOTO 30
20 GOTO 80
30 REM *** BOILER RESET ***
40 M = ( OATLO - OATHI ) / ( HWSHI - HWSLO )
50 B = OATHI - M * HWSLO
60 HWSSP = MIN( MAX( ( OAT - B ) / M , HWSLO ) , HWSHI )
70 L = OAT
80 END
```

**BACnet**

```
10 IF ABS( L - AV10 ) > 5 THEN GOTO 30
20 GOTO 80
30 REM *** BOILER RESET ***
40 M = ( AV14 - AV13 ) / ( AV15 - AV16 )
50 B = AV13 - M * AV16
60 AV12 = MIN( MAX( ( AV10 - B ) / M , AV16 ) , AV15 )
70 L = AV10
80 END
```

**Boiler reset by the step method**

The step method changes the reset temperature only when certain break points are reached. The resulting set point is shown in Illustration 2 over a varying outside air temperature. Break-points on falling outside temperature are shown as dotted lines.



**Illustration 2 The step reset method**

The following Control Basic programs use one or more points or variables described in the section [Variables and value objects](#) on page 4-51.

**KMD with mnemonics**

```
10 IF VAR11 < 0 THEN VAR12 = 190
20 IF VAR11 > 5 AND VAR11 < 20 THEN VAR12 = 180
30 IF VAR11 > 25 AND VAR11 < 40 THEN VAR12 = 160
40 IF VAR11 > 50 THEN VAR12 = 150
50 IF VAR12 > 200 OR VAR12 < 150 THEN VAR12 = 165
60 END
```

**KMD with labels**

```
10 IF OAT < 0 THEN HWSSP = 190
20 IF OAT > 5 AND OAT < 20 THEN HWSSP = 180
30 IF OAT > 25 AND OAT < 40 THEN HWSSP = 160
40 IF OAT > 50 THEN HWSSP = 150
50 IF HWSSP > 200 OR HWSSP < 150 THEN HWSSP = 165
60 END
```

**BACnet with mnemonics**

```
10 IF AV10 < 0 THEN AV12 = 190
20 IF AV10 > 5 AND AV10 < 20 THEN AV12 = 180
30 IF AV10 > 25 AND AV10 < 40 THEN AV12 = 160
40 IF AV10 > 50 THEN AV12 = 150
50 IF AV12 > 200 OR AV12 < 150 THEN AV12 = 165
60 END
```

**PID controller method**

Setup controller CON1 with an input of outside air temperature (OAT), setpoint of 60, (OATHI) proportional band of 100, and reverse acting. As the temperature drops below 60°, CON1 increments to 100%, resetting the setpoint from 150° to 190°.

Set up the controll loops as shown in either Table 3 or Table 4.

**Table 3 KMD PID control loop for boiler reset**

|   | Controlled point | In Value | Man. | Output | Setpoint | Value | Action | Prop. | Reset | H/M | Rate | Bias |
|---|------------------|----------|------|--------|----------|-------|--------|-------|-------|-----|------|------|
| 1 | OAT              |          | Auto | 3      | OATHI    | 60.0  | -      | 100   | 0     | H   | 0.00 | 0    |

**Table 4 BACnet PID control loop for boiler reset**

|   | Controlled Input | Setpoint | Value | Action  | Prop | Integral | Derivative | Max Output |
|---|------------------|----------|-------|---------|------|----------|------------|------------|
| 1 | AV11             | 60       | 60.0  | Reverse | 100  | 0        | 0          | 100        |

Set up the variables or value objects for the following Control Basic programs as described in the section [Variables and value objects](#) on page 4-51.

**KMD with labels**

10 HWSPP = MIN( 150 + CON1 / 2 , HWSHI )

**BACnet with mnemonics**

10 AV12 = MIN( 150 + LOOP1 / 2 , AV15 )

## Variables and value objects

All variable and value objects are configured for analog values. Not all variables are used for all example programs.

**Table 5 BACnet analog value objects configuration**

| Value object | Description    | Name  | Present Value | Units | Out of service |
|--------------|----------------|-------|---------------|-------|----------------|
| AV11         | OutsideAirTemp | OAT   |               | °F    |                |
| AV12         | HWSsetpoint    | HWSSP |               | °F    |                |
| AV13         | OATHiLimit     | OATHI | 60            | °F    |                |
| AV14         | OATLoLimit     | OATLO | -20           | °F    |                |
| AV15         | HWSHiLimit     | HWSHI | 190           | °F    |                |
| AV16         | HWSLoLimit     | HWSLO | 150           | °F    |                |

**Table 6 KMD analog variable points configuration**

| Point | Description    | Label | Value | Units | Manual |
|-------|----------------|-------|-------|-------|--------|
| VAR11 | OutsideAirTemp | OAT   |       | °F    |        |
| VAR12 | HWSsetpoint    | HWSSP |       | °F    |        |
| VAR13 | OATHiLimit     | OATHI | 60    | °F    | X      |
| VAR14 | OATLoLimit     | OATLO | -20   | °F    | X      |
| VAR15 | HWSHiLimit     | HWSHI | 190   | °F    | X      |
| VAR16 | HWSLoLimit     | HWSLO | 150   | °F    | X      |

**AV11 or VAR11—OutsideAirTemp** The outside air temperature. This may be derived from the real-time reading of an outside sensor such as model STE-1451 or from a value derived from the average of several temperature measurements made at regular intervals.

**AV12 or VAR12—HWSsetpoint** The calculated setpoint temperature for the boiler water in the hot water system.

**AV13 or VAR13—OATHiLimit** The outside air temperature that corresponds to the lowest hot water supply temperature. The outside temperature may rise above this value but the hot water setpoint will not go below the value set by the variable *HWSLoLimit*. This value must be greater than *OATLoLimit*.

**AV14 or VAR14—OATLoLimit** The outside air temperature that corresponds to the highest hot water supply temperature. The outside temperature may drop below this value but the hot water setpoint will not go above the value set by the variable *HWSHiLimit*. This value must be less than *OATHiLimit*.

**AV15 or VAR15—HWSHiLimit** The highest water temperature required by the system. This value is the setpoint when the outside temperature reaches the value set by *OATLoLimit*. This value must be greater than *HWSLoLimit*.

**AV16 or VAR16—HWSLoLimit** The lowest water temperature the hot water system will maintain. This value is the setpoint when the outside temperature reaches *OATHiLimit*. This value must be less than *HWSHiLimit*.

**VAR17—PIDloopSP** The loop setpoint when using the PID loop method in a KMD controller. In BACnet controllers this property can be entered directly into the loop object.

**Local variables** In addition to the variable points and value objects listed in the previous section, the following local variables are used at several places in the program:

**Table 7 Local variables**

| <b>Variable</b> | <b>Description</b> |
|-----------------|--------------------|
| B               |                    |
| L               | Last OAT           |
| M               |                    |

Application Note—AN0604B

---

## Calculating discharge reset based on space temperature

This application note describes two methods for calculating discharge reset temperature.

---

The examples in this application note shows how a formula can calculate an active discharge setpoint. For each method the system behavior should be observed to determine if modification is necessary when applying to an untested piece of equipment (one for which the formula has not yet been used).

### Fixed reset

The following program establishes a heating and cooling set point, determines occupancy mode set points, and sets the mode for an air handler or other temperature control unit. The discharge set point is calculated based on the space temperature. This set point is held within the low and high limits MIN\$DAT and MAX\$DAT defined as Variables.

*Example:*

```
10 REM DETERMINING ACTIVE SETPOINTS BASED ON OCCUPANCY MODE
20 IF NOT OCCUPIED THEN COOL$SPT = C$SETBK , HEAT$SPT = H$SETBK
30 IF OCCUPIED THEN COOL$SPT = SETPT$TP + DEADBAND / 2
40 IF OCCUPIED THEN HEAT$SPT = SETPT$TP - DEADBAND / 2
50 IF SPACE$TP < HEAT$SPT THEN START H/C-MODE
60 IF SPACE$TP > COOL$SPT THEN STOP H/C-MODE
70 E = SPACE$TP - SETPT$TP
80 DAT$SP = MIN( MAX( -6 * E + SETPT$TP , MIN$DAT ) , MAX$DAT )
90 END
```

The factor  $-6 * E$  in line 80 sets the discharge set point at six times the space temperature error (E).

For a space set point of 72° the following discharge setpoint values would be found for the space temperatures shown:

**Table 1**

|                   | SPACE | DAT SP |
|-------------------|-------|--------|
|                   | 68    | 96     |
|                   | 69    | 90     |
|                   | 70    | 84     |
| Heating Set Point | 71    | 78     |
|                   | 72    | 72     |
| Cooling Set Point | 73    | 66     |
|                   | 74    | 62     |
|                   | 75    | 54     |
|                   | 76    | 50(48) |

## Dynamic reset

Fixed reset offers a simple control solution by using a fixed formula to determine discharge reset. However, this may not work well for all systems. Using a controller with reset will allow the system to find the appropriate discharge based on the conditions of the space. The following program is similar to example one but controller *CON4* has been used to allow the formula to find the necessary discharge set point for the conditions. Removing the Reset (setting it to 0) gives results similar to the Fixed Reset example.

*Example:*

```

10 HEAT-SP = ROOM.SP - 1
20 COOL-SP = ROOM.SP + 1
30 IF HEATING THEN ACTIVESP = HEAT-SP ELSE ACTIVESP = COOL-SP
40 DAT.SP = MIN( MAX( CON4 / 2 - 25 + ROOM.SP , 50 ) , 120 )
50 END

```

The value of 50 is the lowest allowable discharge set point and 120 is the highest. These values could be set using variables to allow for convenient modification. A controller can be set up as shown below to set the integral reset function of a PID loop controller to adjust discharge air temperature as necessary to maintain space temperature at set point.

**Table 2 PID loop setup**

|   | In Label | In Value | A/M  | Output | Setpoint | Value | A | Prop | Reset | Rate | Bias |
|---|----------|----------|------|--------|----------|-------|---|------|-------|------|------|
| 4 | ROOMTEMP | 70.3     | auto | 56.2 % | ACTIVESP | 69.9  | + | 8    | M     | 0.00 | 50   |

The values shown will add about 1° per minute to the discharge set point when the space temperature is 2° below set point. Because every installation is different, the values for Proportional and Reset may need to be changed to optimize performance.

Application Note—AN0504I

---

## Signal conditioning

This application describes methods to minimize the effects of abrupt signal fluctuations by applying filter function algorithms.

---

Abnormal situations can cause sudden shifts in a sensed value. Mode and load transitions can cause drastic changes in a normally stable system. These changes can cause a rapid change in a PID controller that varies little in its steady state operation. This can cause outputs to ramp up or down. It is often not desirable to subject equipment to such abrupt changes. In many cases, it is preferable to ignore the condition entirely.

The program examples that follow can be used to buffer or dampen an input value (physical input or a calculated value) to allow more stable operation of equipment.

### Filtering a step function

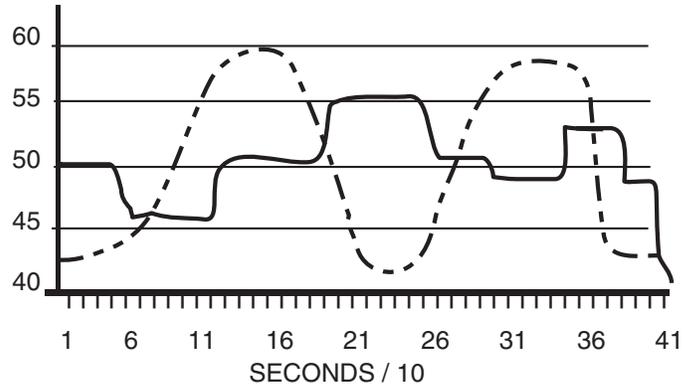
The step program protects equipment from transient conditions by limiting the rate of change of the output control signal. In this example the output can only change once every minute. No changes will occur until the time set in the interval statement has elapsed.

*Example:*

```
10 IF INTERVAL( 0:01:00 ) THEN 20 ELSE END
20 IF ABS( CONTROLR - BoilerSP ) > factor THEN 30 ELSE BoilerSP
   = CONTROLR , END
30 IF BoilerSP < CONTROLR THEN BoilerSP = BoilerSP + factor
40 IF BoilerSP > CONTROLR THEN BoilerSP = BoilerSP - factor
50 END
```

Here, the output of a PID controller is checked for rate of change to limit the Output value (BoilerSP). Line 10 sets the update time. The higher the update time, the more sluggish the output becomes. For a value such as a temperature set point it may be desirable to only change the output once every hour. Some equipment may need to be updated more frequently. Line 20 compares the amount the controller value has changed since the last update (one INTERVAL earlier) with the maximum allowed rate of change (Factor). For this example, *factor* should be in units of “% per minute”. If the Interval used is 1 hour (01:00:00), the variable

*factor* would be in units of “% per hour”. If the amount the controller changed is greater than the *factor*, then the output can only change by one factor at each interval.



**Illustration 1 Step Response to Rapidly Changing Value**

The solid line in Illustration 1 is the stepped output of the unstable input represented by the dashed line. As the value of the Variable “Factor” is decreased, the peak-to-peak amplitude of the stepped output becomes smaller.

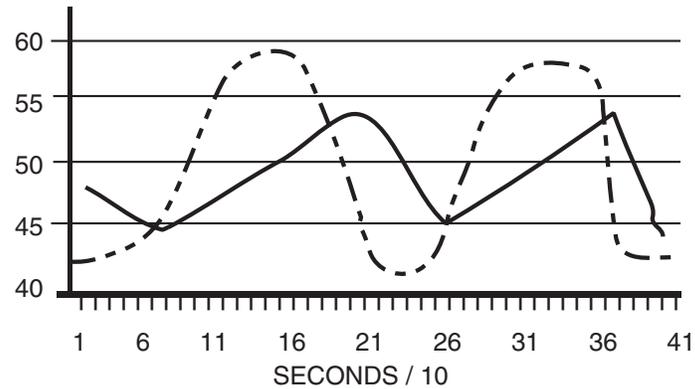
## Filtering a ramp function

The ramp routine will only allow a value to change at a specified rate or slower. The ramp will follow an applied signal exactly unless that signal changes too quickly. The ramp is adjusted at each scan, giving a smooth output suitable for sensitive equipment such as variable speed drives.

*Example:*

```
10 A = FACTOR / ( 60 * SCANS )
20 IF ABS( CONTROLR - OUT.VAL ) > A THEN 30 ELSE OUT.VAL =
CONTROLR , END
30 IF OUT.VAL < CONTROLR THEN OUT.VAL = OUT.VAL + A
40 IF OUT.VAL > CONTROLR THEN OUT.VAL = OUT.VAL - A
50 END
```

Factor is the maximum desired rate of change in units per minute. Line 10 calculates the local variable "A" as the maximum rate change per scan. If the input value (CONTROLR) has changed more than the maximum rate of change set by "factor" in a single scan, the OUT.VAL will be limited to the rate of "A".



**Illustration 2 Ramping Response to Rapidly Changing Input**

The solid line in Illustration 2 is the ramping output of the unstable input represented by the dashed line. As the value of the variable *Factor* is decreased, the peak-to-peak amplitude of the output becomes smaller.

## Filter function

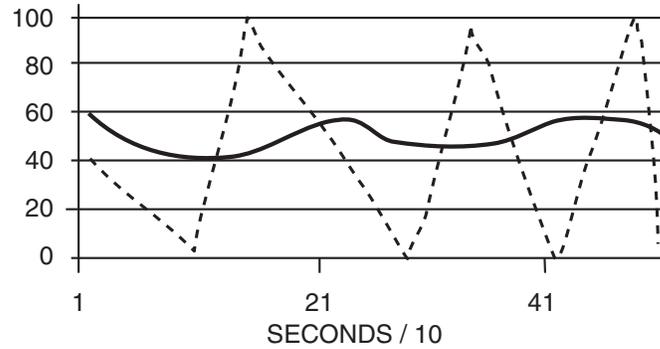
The filter program limits the rate of change of a value by averaging samples of the value. The Variable *FILTFACT* is the factor that determines the number of samples included in the average. If *FILTFACT* is set to 30 the result will be an average of the last 30 samples. The result is not a true average of the *FILTFACT* number of points collected. A sample taken more than *FILTFACT* samples before can effect the result if it is much greater or smaller than the other samples.

*Example:*

```
10 IF INTERVAL( 0:00:10 ) THEN 20 ELSE 40
20 A = A + 1 : IF A > FILTFACT THEN A = FILTFACT
30 SPACETMP = ( ( A - 1 ) * SPACETMP + BUTTON1 ) / A
40 END
```

In this example, the NetSensor temperature is being filtered. The thermostat was located near an outside door and it senses large drops in temperature when someone enters the room in the winter. The variable *BUTTON1* is assigned as the NetSensor temperature. The result is *SPACETMP*, which is the filtered value of the space temperature. This filtered value can be used instead of the direct sensor value to give a cushion from sudden impulses or abnormal situations. Line 10 sets the sample rate. If *FILTFACT* is set to 6 in this example, there will be 6 samples included in the average over one minute. Line 20 maintains the number of points

that have been collected. When the Program is initiated, "A" counts to FILTFACT as it collects samples. Thereafter, "A" equals FILTFACT. In Line 30 the result is determined by averaging the new sample into the previous result.



**Illustration 3 Filtered Response to a Rapidly Changing Value**

The solid line in Illustration 3 is the filtered output of the unstable input represented by the dashed line. As the value of the variable *Factor* is increased, the peak-to-peak amplitude of the output becomes smaller.

On standard inputs the "Aft" column can be set to the number of samples (at one sample per scan) to be included in an average. This averaging compensates for variations of the input to give a more stable reading. The NetSensor does not have this averaging feature because it is not a standard input. If the NetSensor is located in an area where it is exposed to fluctuations not representative of the space temperature, a filter program may be utilized to help stabilize the space temperature control. The update time on the NetSensor is about 6 to 8 seconds so the update interval should be 10 seconds or more.

For equipment that requires more frequent updating, line 10 of the filter program can be eliminated, as shown below. If this line is not used, the value of *FILTFACT* is the number of scans included in the average. Instead of waiting for an interval to elapse, every time the program is read a new sample is added to the filtered value.

*Example:*

```
20 A = A + 1 : IF A > FILTFACT THEN A = FILTFACT
30 FANSPEED = ( ( A - 1 ) * FANSPEED + CONTROLR ) / A
40 END
```

The *FILTFACT* Variable should be higher than when the interval statement was used since the samples are being taken more than once each second. If the *FILTFACT* Variable is set to 200, there will be 200 samples at one per scan included in the filtered value. In line 30 a *FANSPEED* output becomes the filtered value of a PID controller.

## Application Note–AN0504J

---

### Example programs using tables

This application note describes several methods of using tables in WinControl.

---

#### Calculating sunrise and sunset times for lighting control

In situations where photocells or other optical detectors are inadequate or not reliable, sunrise and sunset times can be calculated with acceptable accuracy. This method will work in an KMD–5801 controller. More accurate methods are available using the advanced features of the LAN Controller (KMD-5210).

If the sunrise and sunset times are not known for an area, search the Internet for tables (sunrise sunset times) or contact the U.S. Naval Observatory.

Choose a rise and set time from each month of the year. The KMC digital controller table will interpolate the times in between. Table 1 lists the sunrise and sunset times for each month for in Northern Indiana, the location of the KMC Controls factory. Locations in other longitudes require different entries.

**Table 1 Sunrise and sunset in Northern Indiana**

| Month     | Rise | Set   |
|-----------|------|-------|
| January   | 8:10 | 17:24 |
| February  | 7:55 | 17:59 |
| March     | 7:18 | 18:34 |
| April     | 6:27 | 19:09 |
| May       | 5:40 | 19:41 |
| June      | 5:12 | 20:11 |
| July      | 5:13 | 20:21 |
| August    | 5:38 | 20:01 |
| September | 6:09 | 19:17 |
| October   | 6:40 | 18:25 |
| November  | 7:15 | 17:38 |
| December  | 7:50 | 17:14 |

In KMC digital controllers time is kept in hours in a 24-hour format. To convert the times to a number that can be compared to the controllers clock, use the following formula:

$$\text{Time} = (\text{Hour} * 100) + (\text{Min} / .6) \quad (\text{Round the result to remove the decimal portion})$$

For example, in January sunrise is 8:10AM or:

$$\begin{aligned} \text{Time} &= ( 8 * 100 ) + ( 10 / .6 ) \\ &= 800 + 16.667 \\ &= 817 \end{aligned}$$

Enter the resulting data into two separate tables. You will create a table for sunrise times and a table for sunset times. For New Paris, Indiana the tables look like the ones shown below.

System: DEMO UNIT  
 Panel : 1

Wed Sep 10 15:42:12 2004  
 SubB #16 LIGHTING PANEL

**Table 2**

|    | <b>TABLE 1</b> |       | <b>TABLE 2</b> |      | <b>TABLE 3</b> |      |
|----|----------------|-------|----------------|------|----------------|------|
|    | <b>Value</b>   |       | <b>Value</b>   |      | <b>Value</b>   |      |
| 1  | 1.00           | 817.0 | 1.00           | 1740 | 0.00           | 0.00 |
| 2  | 32.0           | 792.0 | 32.0           | 1799 | 0.00           | 0.00 |
| 3  | 61.0           | 730.0 | 61.0           | 1857 | 0.00           | 0.00 |
| 4  | 92.0           | 645.0 | 92.0           | 1913 | 0.00           | 0.00 |
| 5  | 122.0          | 567.0 | 122.0          | 1968 | 0.00           | 0.00 |
| 6  | 153.0          | 520.0 | 153.0          | 2019 | 0.00           | 0.00 |
| 7  | 165.0          | 515.0 | 165.0          | 2030 | 0.00           | 0.00 |
| 8  | 183.0          | 522.0 | 183.0          | 2035 | 0.00           | 0.00 |
| 9  | 214.0          | 563.0 | 214.0          | 2002 | 0.00           | 0.00 |
| 10 | 245.0          | 615.0 | 245.0          | 1938 | 0.00           | 0.00 |
| 11 | 275.0          | 667.0 | 275.0          | 1842 | 0.00           | 0.00 |
| 12 | 306.0          | 725.0 | 306.0          | 1763 | 0.00           | 0.00 |
| 13 | 336.0          | 783.0 | 336.0          | 1723 | 0.00           | 0.00 |
| 14 | 366.0          | 817.0 | 366.0          | 1740 | 0.00           | 0.00 |
| 15 | 0.00           | 0.00  | 0.00           | 0.00 | 0.00           | 0.00 |

The following program can be used to find the rise or set time for the current day by looking it up on the table;

*Example:*

```
10 RISE = TBL( DOY , 1 )
20 IF TIME > RISE + DELRISE THEN START DAY
40 SET = TBL( DOY , 2 )
50 IF TIME > SET - DELSET THEN STOP DAY
```

The variable "DAY" can then be used to start and stop lighting contactors using a line similar to the following:

*Example:*

```
100 IF DAY THEN STOP PARKLITE ELSE START PARKLITE
```

Where "PARKLITE" is a digital output controlling outdoor lights.

## Splitting data over multiple tables

In some cases it may be necessary to use more than the 15 points available in a single Table. An example of such a case would be interpreting a complex equation such as one with multiple peaks. The example below shows how you can use 2 tables in an 8 X 8 to increase the accuracy of converting a non-standard input from a transmitter into a temperature reading.

System: The DEMO Panel  
 Panel : 1 SubA #1

Wed Sep 10 16:13: 2004

**Table 3**

|    | <b>TABLE 1</b> |       | <b>TABLE 2</b> |       | <b>TABLE 3</b> |      |
|----|----------------|-------|----------------|-------|----------------|------|
|    | <b>Value</b>   |       | <b>Value</b>   |       | <b>Value</b>   |      |
| 1  | 0.00           | 40.0  | 3.00           | 100.0 | 0.00           | 0.00 |
| 2  | 1.00           | 42.0  | 3.20           | 110.0 | 0.00           | 0.00 |
| 3  | 1.10           | 48.0  | 3.30           | 120.0 | 0.00           | 0.00 |
| 4  | 1.30           | 52.0  | 3.50           | 130.0 | 0.00           | 0.00 |
| 5  | 1.50           | 56.0  | 3.60           | 140.0 | 0.00           | 0.00 |
| 6  | 1.70           | 59.0  | 3.70           | 150.0 | 0.00           | 0.00 |
| 7  | 1.90           | 63.0  | 3.90           | 160.0 | 0.00           | 0.00 |
| 8  | 2.10           | 65.0  | 4.00           | 170.0 | 0.00           | 0.00 |
| 9  | 2.20           | 68.0  | 4.10           | 180.0 | 0.00           | 0.00 |
| 10 | 2.50           | 70.0  | 4.20           | 190.0 | 0.00           | 0.00 |
| 11 | 2.60           | 72.0  | 4.40           | 200.0 | 0.00           | 0.00 |
| 12 | 2.70           | 80.0  | 4.50           | 210.0 | 0.00           | 0.00 |
| 13 | 2.80           | 85.0  | 4.80           | 220.0 | 0.00           | 0.00 |
| 14 | 2.90           | 95.0  | 5.00           | 230.0 | 0.00           | 0.00 |
| 15 | 3.00           | 100.0 | 5.10           | 240.0 | 0.00           | 0.00 |

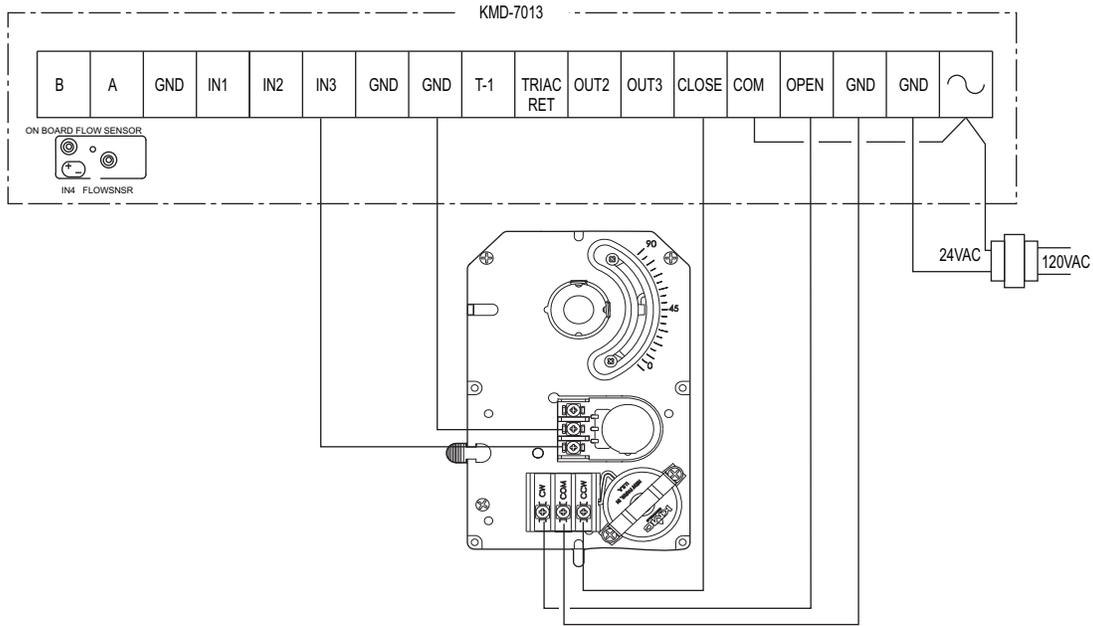
To link Table 1 and Table 2, use the following line of code in a Control Basic program area:

*Example:* `IF IN-VOLT <= 3 then TEMP = TBL( IN-VOLT , 1 ) ELSE TEMP = TBL( IN-VOLT , 2 )`

Where IN-VOLT is the nonlinear input voltage that varies between 1 and 5 volts. TEMP is a variable with units in °F (range 2) that stores the result of the interpolation from the table data points. When the input voltage is less than or equal to 3VDC the first table is used to determine the value of TEMP. When the voltage is above 3 the second table is used.

**Position feedback tables**

The following diagram shows MEP-5065 actuator with position feedback connected to a KMD-7013 controller. In this example, use WinControl XL to select a table for Input 3. Enter the values shown in Table 4 in the selected table.



**Illustration 1** MEP-5065 connected to measure position

**Table 4** Positioner values

| Voltage | %   |                      |
|---------|-----|----------------------|
| 0.00    | 0   | (Position Fully CCW) |
| 0.45    | 10  |                      |
| 0.83    | 20  |                      |
| 1.15    | 30  |                      |
| 1.43    | 40  |                      |
| 1.67    | 50  |                      |
| 1.88    | 60  |                      |
| 2.06    | 70  |                      |
| 2.22    | 80  |                      |
| 2.37    | 90  |                      |
| 2.50    | 100 | (Position fully CW)  |



**Note**

The voltage values are approximate. For precise position, measure the actual voltage for each position.



## Application Note–AN0604C

---

Revision A

### **Monitoring refrigeration systems for moisture and flash gas**

This application note describes a method to use a KMC Controls First Watch refrigerant monitor to detect flash gas and moisture in refrigerant. Programs for both KMD and BACnet controllers are included in this application note.

---

#### **Applicable models**

This application note describes a method to monitor refrigeration gas with a KMC SLE-1001 FirstWatch monitor connected to a controller from KMC Controls. Either a KMD-5801 or a BAC-5801 may be connected to the monitor. However, any KMD or BACnet series controller with three available inputs and two available outputs can be used with the examples in this application note.

#### **Related materials and software**

In addition to the material presented in this application note, review and have available the following reference materials.

- ◆ Installation and operation guide for the controller to which the SLE-1001 will be connected.
- ◆ Installation and operation guides for the SLE-1001 FirstWatch refrigeration monitor.
- ◆ System plans with controller addresses.

You will also need TotalControl, BACstage or WinControl XL Plus to program and configure the controller.

#### **Installation**

Install the SLE-1001 following the instructions in the manual *Installation and Operation Guide, FirstWatch Refrigerant Monitor*. Connect the SLE-1001 to the controller as shown in the illustration [FirstWatch connected to a BAC-5801 or KMD-5801 controller](#) on page 4-74.

## Setting up the monitor

Once the SLE-1001 is installed and connected to the controller, use TotalControl, BACstage or WinControl XL Plus to program and configure the controller. The refrigeration system to which the SLE-1001 is connected must be functional to set up the controller.

### Setting the moisture baseline

There may be different shades of green on the moisture indicator in the sight glass on which the SLE-1001 is installed. The variable *MOISTURE COLOR BASE* compensates for the differences in color from one sight glass to another. Use the following procedure to set the moisture baseline:

1. Operate the refrigeration system and SLE-1001 for 10 minutes.
2. Record the input value of Input 2, *MOISTURE*.
3. Enter the value recorded in *MOISTURE* into the variable *MOISTURE COLOR BASE*.

### Setting the flash gas baseline

The backside of the site glasses may have different shades of reflectivity because of aging or overheating during installation of the sight glass. The variable *FLASH BASE* compensates for the different shades of reflectivity from one sight glass to another. Use the following procedure to set the flash gas baseline:

1. Operate the refrigeration system and the SLE-1001 for 10 minutes.
2. Record the input value of Input 1, *Flash Gas*.
3. Enter the value recorded in *Flash Gas* into the variable *FLASH BASE*.

**Input configuration**

Connect the flash gas and moisture outputs of the SLE-1001 FirstWatch to the inputs of a controller. For example wiring see the drawing [FirstWatch connected to a BAC-5801 or KMD-5801 controller](#) on page 4-74. In addition to the SLE-1001 inputs, the program requires an input to signal that the compressor is running. Configure the inputs as shown in either [BACnet input object configuration](#) on page 4-67.

**AV1 or IN1—FLASH GAS** An active input connected to the output of the SLE-1001. Remove the input resistor or resistor jumper on the controller.

**AV2 or IN2—MOISTURE** An active input connected to the output of the SLE-1001. Remove the input resistor or resistor jumper on the controller.

**BV3 or IN3—UNIT STATUS** Typically this is a passive input such as a set of switch or relay contacts. An active input indicates the compressor is running.

**Table 1 BACnet input object configuration**

|   | Description | Name   | Present Value | Units  | Device Type      | Object Type | Filter weight |
|---|-------------|--------|---------------|--------|------------------|-------------|---------------|
| 1 | FLASH GAS   | FLASH  |               | %      | Volts 0-100% 0-5 | Analog      | 4             |
| 3 | MOISTURE    | H2O    |               | %      | Volts 0-100% 0-5 | Analog      | 7             |
| 3 | UNIT STATUS | STATUS |               | Off/On |                  | Binary      |               |

**Table 2 KMD input configuration**

|    | Description | Manual | Value | Units              | Average | Label  | Type    |
|----|-------------|--------|-------|--------------------|---------|--------|---------|
| #1 | FLASH GAS   | A      |       | 0-100% (0-5 Volts) | 16      | FLASH  | Analog  |
| #2 | MOISTURE    | A      |       | 0-100% (0-5 Volts) | 128     | H2O    | Analog  |
| #3 | UNIT STATUS | A      |       | Off/On             |         | STATUS | Digital |

## Output configuration

The Control Basic program uses two outputs to activate local alarm devices.

**BV1 or OUT1—FLASH OUPUT ALARM** This output becomes active when variable *FLASH GAS ALARM* is active. Connect the output to a local alarm as shown in Illustration 1 on page 74.

**BV2 or OUT2—H2O OUTPUT ALARM** This output becomes active when *MOISTURE ALARM* is active. Connect the output to a local alarm device as shown in Illustration 1 on page 74.

**Table 3 BAS-7301 output configuration**

| Description         | Name     | Present Value | Units        | Device Type | Object Type |
|---------------------|----------|---------------|--------------|-------------|-------------|
| 1 FLASH OUPUT ALARM | FLSH/OUT |               | Normal/Alarm |             | Binary      |
| 2 H2O OUTPUT ALARM  | H2O/OUT  |               | Normal/Alarm |             | Binary      |

**Table 4 KMD output settings**

| Description         | Manual | Value | Units  | 0% | 100% | Delay | Label    | Type    |
|---------------------|--------|-------|--------|----|------|-------|----------|---------|
| 1 FLASH OUPUT ALARM | A      |       | Off/On | 0  | 0    |       | FLSH/OUT | Digital |
| 2 H2O OUTPUT ALARM  | A      |       | Off/On | 0  | 0    |       | H2O/OUT  | Digital |

## Variables and value object configuration

The assignment of BACnet value objects or KMD variables is similar for programs in both types of controllers.

### Analog value objects and points

**AV4 or VAR4—FLASHBASE** The baseline for flash gas in a normally functioning system. The value is entered during set up. See [Setting the flash gas baseline](#) on page 4-66.

**AV5 or VAR5—FLASH GAS ALARM STPT** The setpoint for the flash gas alarm.

**AV6 or Var6—FLASH VALUE** The active value of flash gas on a scale of 0–100%. A 0% value indicates no flash gas and 100% is a strong indication of flash gas.

**AV7 or VAR7—FLASH DELAY** The Control Basic example program eliminates nuisance alarms with the delay value entered in variable *FLASH DELAY*.

**AV10 or VAR10—MOISTURE COLOR BASE** The variable *MOISTURE COLOR BASE* compensates for the differences in color from one sight glass to another. The value is entered during set up. See [Setting the moisture baseline](#) on page 4-66.

**AV11 or VAR11—MOISTURE ALARM STPT** The setpoint for the moisture alarm.

**AV12 or VAR12—MOISTURE VALUE** The variable *MOISURE VALUE* is the active value of moisture on a scale of 0–100%. No moisture is indicated by a 0% reading; a 100% reading is a strong indication of moisture.

### Binary value objects and points

**BV1 or VAR1—FLASH GAS ALARM** When the value of *FLASH VALUE* is equal to or greater than the variable *FLASH GAS STPT* then the variable *FLASH GAS ALARM* changes to an alarm state.

**BV2 or VAR2—MOISTURE ALARM** When *MOISTURE VALUE* becomes greater than or equal to *MOISTURE ALARM STPT* then *MOISTURE ALARM* changes to an alarm state.

**Table 5 KMD analog and digital variable settings**

|     | Description          | Manual | Value    | Units        | Label    | Type    |
|-----|----------------------|--------|----------|--------------|----------|---------|
| 1   | FLASH GAS ALARM      | A      | Normal   | Normal/Alarm | FL/ALRM  | Digital |
| 2   | MOISTURE ALARM       | A      | Normal   | Normal/Alarm | H2O/ALRM | Digital |
| 3   | Unused               |        |          |              |          |         |
| 4   | FLASH BASE           | M      | 0        | Percent %    | FL/BASE  | Analog  |
| 5   | FLASH GAS ALARM STPT | M      | 45       | Percent %    | FL/STPT  | Analog  |
| 6   | FLASH VALUE          | A      |          | Percent %    | FL/VALUE | Analog  |
| 7   | FLASH DELAY          | M      | 0:07:00  | Time         | FL/DELAY | Analog  |
| 8,9 | Unused               |        |          |              |          | Analog  |
| 10  | MOISTURE COLOR BASE  | M      |          | Percent %    | H2O/BASE | Analog  |
| 11  | MOISTURE ALARM STPT  | M      | 45       | Percent %    | H2O/STPT | Analog  |
| 12  | MOISTURE VALUE       | A      | 0.708673 | Percent %    | H2O/VALU | Analog  |

**Table 6 BACnet analog value objects**

| Value object | Description          | Name     | Present Value | Units   | Out of service |
|--------------|----------------------|----------|---------------|---------|----------------|
| AV1–AV3      | Not used             |          |               |         |                |
| AV4          | FLASH BASE           | FL/BASE  |               | %       |                |
| AV5          | FLASH GAS ALARM STPT | FL/STPT  | 45            | %       |                |
| AV6          | FLASH VALUE          | FL/VALUE |               | %       |                |
| AV7          | FLASH DELAY          | FL/DELAY | 420           | seconds |                |
| AV8, AV9     | Not used             |          |               |         |                |
| AV10         | MOISTURE COLOR BASE  | H2O/BASE |               | %       |                |
| AV11         | MOISTURE ALARM STPT  | H2O/STPT | 45            | %       |                |
| AV12         | MOISTURE VALUE       | H2O/VALU |               | %       |                |

**Table 7 BACnet binary value objects**

| Value object | Description     | Name     | Present Value | Active Inactive text | Units |
|--------------|-----------------|----------|---------------|----------------------|-------|
| BV1          | FLASH GAS ALARM | FL/ALRM  |               |                      |       |
| BV2          | MOISTURE ALARM  | H2O/ALRM |               |                      |       |

**Local variables**

In addition to the variables and value objects listed in the previous section, the following local variables are used at several places in the program.

**Table 8 Local variables**

| Variable | Description  |
|----------|--|
| A        | Used as a flag to indicate that flash gas is above the alarm setpoint. |
| B        | Used as a flag to indicate that moisture is above the alarm setpoint.  |

**Program description**

This is a description of the programs in the section [Programming listing](#) on page 4-73.

**Lines 10 and 20** Stores—as a percent—the calculated value of detected flash gas. The baseline for the calculation is *FLASH BASE*.

**Lines 30 and 40** Sets local variable “A” to *True* if the value of flash gas is above the setpoint value.

**Line 50** Sets local variable “A” to *False* if the value of flash gas falls 5% below the setpoint value.

**Lines 60 and 70** Starts a timer when local variable “A” is true. If “A” is *True* longer than the time set by the variable *FLASH DELAY*, the variable *FLASH GAS ALARM* becomes active. The output *FLASH OUPUT ALARM* also becomes active.

**Line 80 (KMD only)** The alarm is sent to the alarm handling program, either TotalControl or WinControl XL Plus.

**Lines 90 and 100** Stores—as a percent—the calculated value of detected moisture. The baseline for the calculation is *MOISTURE COLOR BASE*.

**Lines 110 and 120** Sets local variable “B” to *True* if the value of moisture is above the setpoint value.

**Line 130** Sets local variable “B” to *False* if the value of moisture falls 5% below the setpoint value.

**Line 140** Start or stops the variable *MOISTURE ALARM* based on the value of local variable “B”.

**Lines 150** The output *H2O OUPUT ALARM* becomes active or inactive based on the value of the variable *MOISTURE ALARM*.

**Line 160 (KMD only)** The alarm is sent to the alarm handling program, either TotalControl or WinControl XL plus.

## Additional and optional programming

In addition to configuring inputs, outputs, and variables, the following additional Control Basic programming may be required.

### BACnet notifications

To set up a BACnet controller to send notifications of when value objects *FLASH GAS ALARM* or *MOISTURE ALARM* change to an alarm state, set up a notification class or event enrollment object in the controller to send the notification to a BACnet net alarm handler. Typically, this is either BACstage or TotalControl. Configure notifications to be sent when binary variable objects *FLASH GAS ALARM* and *MOISTURE ALARM* change state.

### Alarm setpoints

Change the alarm setpoints by changing the variables *FLASH GAS ALARM STPT* or *MOISTURE ALARM STPT*.

### Unit status switch

Some refrigeration systems have solenoid valves downstream from the sight glass on which the SLE-1001 is mounted. In this type of installation the liquid refrigerant will not flash when the compressor shuts down. To adapt to this application, Input 3 is not required to monitor compressor status. Modify the program by modifying the following lines of Control Basic. This removes the programming sequence that supports Input 3.

#### *KMD example*

```
40 IF VAR6 > VAR5 THEN START A
50 IF VAR6 < VAR5 - 5 THEN STOP A

120 IF VAR12 > VAR11 THEN START B
130 IF VAR12 < VAR11 - 5 THEN STOP B
```

#### *BACnet example*

```
40 IF AV6 > AV5 THEN START A
50 IF AV6 < AV5 - 5 THEN STOP A

120 IF AV12 > AV11 THEN START B
130 IF AV12 < AV11 - 5 THEN STOP B
```

**Programming listing**

BACstage, TotalControl, and WinControl files are available on the KMC Controls web site.

**KMD listing**

```

10 REM **** FLASH GAS CALCULATION ****
20 VAR6 = ( IN1 - VAR4 ) / ( 100 - VAR4 ) * 100

30 REM **** FLASH GAS ALARM ****
40 IF VAR6 > VAR5 AND IN3 THEN START A
50 IF VAR6 < VAR5 - 5 OR NOT IN3 THEN STOP A
60 IF TIME-ON( A ) > VAR7 THEN START VAR1 ELSE STOP VAR1
70 IF VAR1 THEN START OUT1 ELSE STOP OUT1
80 DALARM VAR1 , 10 , THERE MAY BE FLASH GAS PRESENT AT THE SITE
GLASS

90 REM **** MOISTURE CALCULATION ****
100 VAR12 = ( IN2 - VAR10 ) / ( 100 - VAR10 ) * 100

110 REM **** MOISTURE ALARM ****
120 IF VAR12 > VAR11 AND IN3 THEN START B
130 IF VAR12 < VAR11 - 5 OR NOT IN3 THEN STOP B
140 IF B THEN START VAR2 ELSE STOP VAR2
150 IF VAR2 THEN START OUT2 ELSE STOP OUT2
160 DALARM VAR2 , 10 , THE SITE GLASS IS INDICATING MOISTURE IS
PRESENT

```

**BACnet listing**

```

10 REM **** FLASH GAS CALCULATION ****
20 AV6 = ( AI1 - AV4 ) / ( 100 - AV4 ) * 100

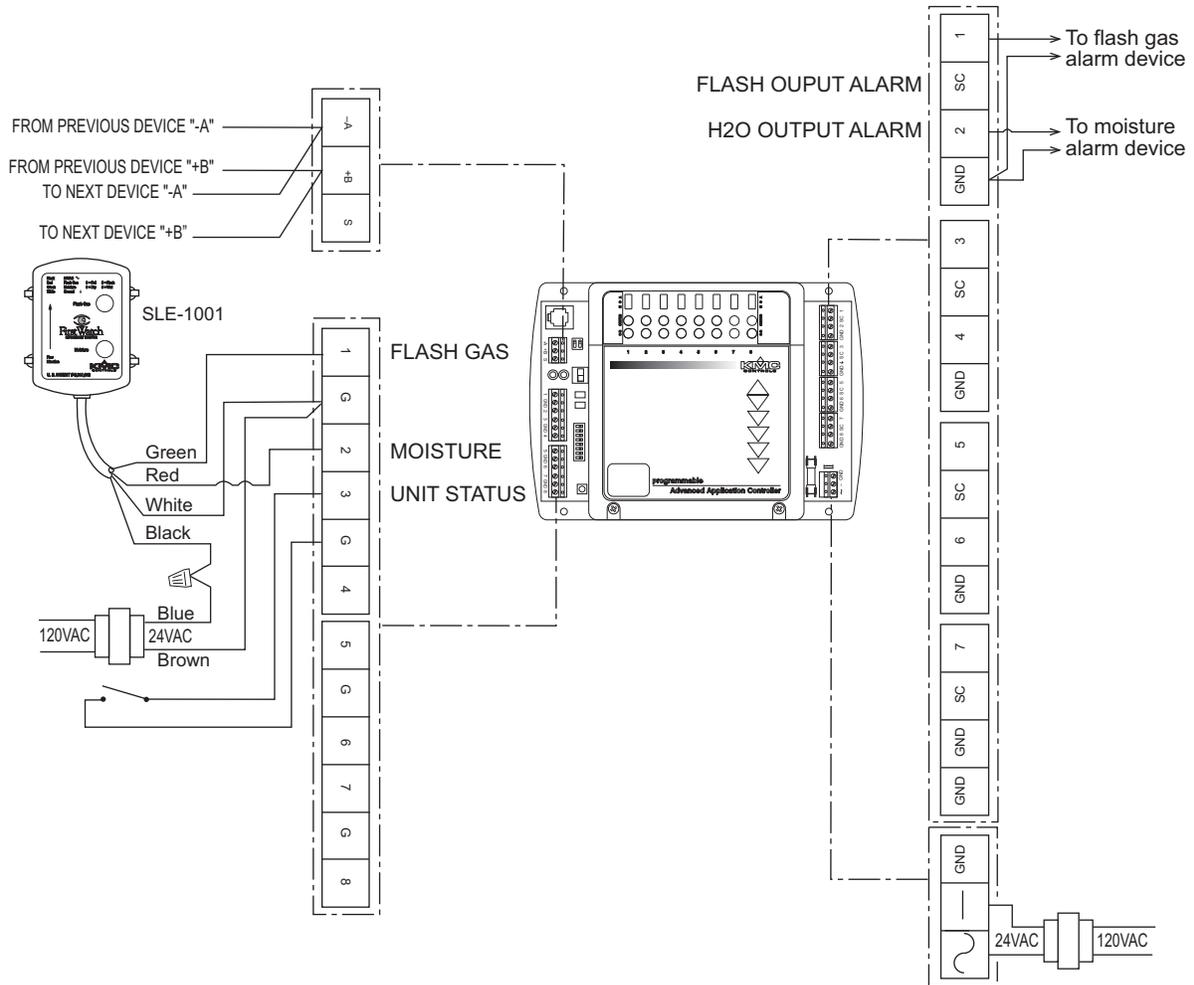
30 REM **** FLASH GAS ALARM ****
40 IF AV6 > AV5 AND BI3 THEN START A
50 IF AV6 < AV5 - 5 OR NOT BI3 THEN STOP A
60 IF TIMEON( A ) > AV7 THEN START BV1 ELSE STOP BV1
70 IF BV1 THEN START BO1 ELSE STOP BO1
80 REM Configure BV1 to send alarm to BACstage or TotalControl
when flash gas is present

90 REM **** MOISTURE CALCULATION ****
100 AV12 = ( AI2 - AV10 ) / ( 100 - AV10 ) * 100

110 REM **** MOISTURE ALARM ****
120 IF AV12 > AV11 AND BI3 THEN START B
130 IF AV12 < AV11 - 5 OR NOT BI3 THEN STOP B
140 IF B THEN START BV2 ELSE STOP BV2
150 IF BV2 THEN START BO2 ELSE STOP BO2
160 REM Configure BV2 to send alarm to BACstage or TotalControl
when moisture is present

```

### Application drawing



**Illustration 1** FirstWatch connected to a BAC-5801 or KMD-5801 controller

## Application Note–AN1009A

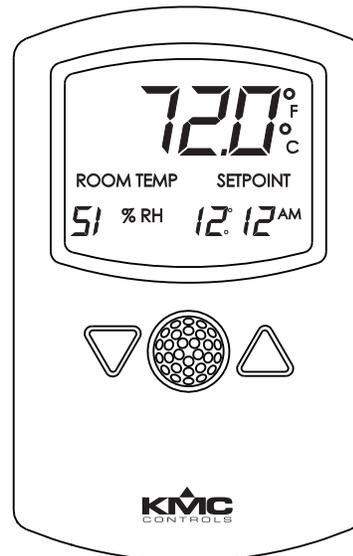
---

### Determining occupancy with motion sensing

This application note describes methods for determining room occupancy based on motion detected by NetSensor models KMD-1261 or KMD-1281.

---

NetSensor models KMD-1261 and KMD-1281 include an internal sensor to detect motion in a room. With a Control Basic program, the presence or absence of motion can be used to determine room occupancy. Occupancy can then be used to control lights or adjust setpoints.



*Illustration 1 KMD-1281*

#### Applicable models

The Control Basic programs described in this application note can be operated on KMD and BACnet controllers that are compatible with models KMD-1261 and KMD-1281.

## Related materials

In addition to the material presented in this application note, review and have available the following reference materials.

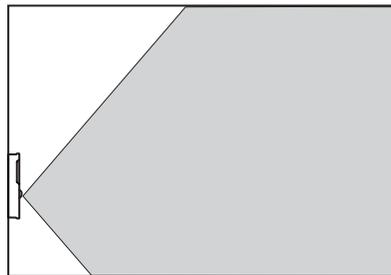
- ◆ The installation and operation guide for the KMD-1261 and KMD-1281 NetSensors.
- ◆ The installation and operation guide for the controller to which the NetSensor is connected.
- ◆ Installation and operation guides for other equipment such as external motion sensors that will be used with the controller.
- ◆ Reference manuals, user's manuals, or help for BACstage, WinControl XL Plus, or TotalControl.

## Installation

Mount the NetSensors on a wall that will have an unobstructed view of the typical traffic in the coverage area. When choosing a location, do not install the NetSensor in the following areas.

- ◆ Behind curtains or other obstructions
- ◆ In locations that will expose it to sunlight or heat sources
- ◆ Near a heating/cooling duct.

For details on the coverage pattern, see the installation and operation guide supplied with the NetSensor.



**Illustration 2 Typical motion sensing coverage area**

The effective detection range is approximately 10 meters or 33 feet. Factors that may reduce the range include:

- ◆ The difference between the surface temperature of the object and the background temperature of the room is too small.
- ◆ Object movement in a direct line toward the sensor.
- ◆ Very slow or very fast object movement.
- ◆ Obstructions as shown in the illustration [Large room with obstructions](#) on page 4-77.

False detections may be triggered by:

- ◆ The temperature inside the detection range suddenly changes because of the entry of cold or warm air from an air-conditioning or heating unit.
- ◆ The sensor being directly exposed to sunlight, an incandescent light, or other source of far-infrared rays.
- ◆ Small animal movement.

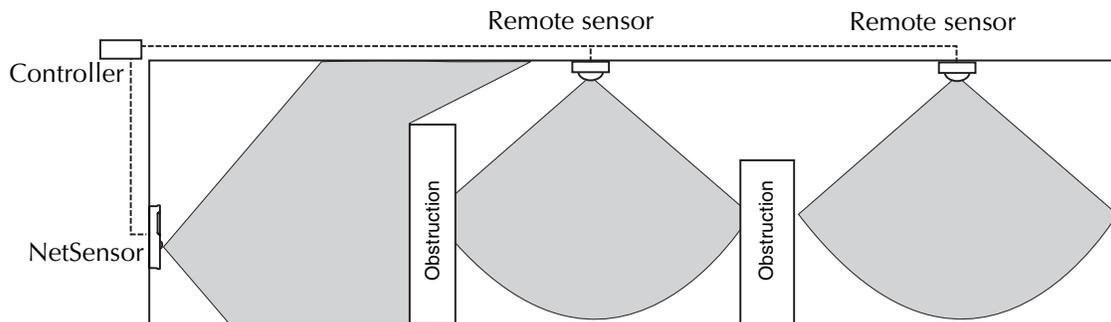
For some applications—such as hotel rooms, bedrooms, and other sleeping quarters—the lack of detected motion may not be a true indication that the room is unoccupied. For a more accurate determination of occupancy, combine motion sensors with door contacts. If no motion is detected during a predetermined time after a door opens or closes, then the mode changes to unoccupied. However, if motion is detected during the period after a door has been opened or closed, then the room can be assumed to be occupied. Custom programming for this type of application is not covered in this application note.

## Extending the coverage area

For areas that cannot be covered with a single NetSensor, add one or more remote motion sensors. Remote sensors may be required for the following applications.

- ◆ For areas that are too large for adequate coverage with a single the NetSensor.
- ◆ In areas with obstructions that block the sensor coverage.
- ◆ When the area is an irregular shape with blind spots such as an L-shaped room.

To extend the range of the sensor in the NetSensor, connect one or more remote motion sensors to controller input terminals. Exact wiring details for the remote sensors will depend on the type of external sensor.



**Illustration 3** Large room with obstructions

## **Programming KMD controllers to detect occupancy**

Use the following programs with KMD series Tier 2 controllers to set occupancy status based on motion detected in a room. Add the example Control Basic program to the KMD controller to which the NetSensor is connected. In addition to adding the Control Basic programs, configure the variables as described in section [Variable points](#) on page 4-79.

The controller to which the NetSensor is connected is typically programmed for setpoint, temperature and other functions. This programming is described in the installation and operation guide supplied with the NetSensor and in the help for WinControl XL Plus and TotalControl.

The next Control Basic examples are for applications using a single NetSensor to detect motion in a room.

```
10 IF+ VAR8 = -1 THEN START VAR9 , STOP A
20 IF VAR8 = -2 THEN START A
30 IF TIME-ON( A ) > 0:20:00 THEN STOP VAR9
```

The following program determines occupancy using both a NetSensor and a remote sensor connected to input IN1. For this example, the remote sensor value switches without delay between periods of motion or no motion.

```
10 IF+ VAR8 = -1 OR IN1 = 1 THEN START VAR9 , STOP A
20 IF VAR8 = -2 AND IN1 < 1 THEN START A
30 IF TIME-ON( A ) > 0:20:00 THEN STOP VAR9
```

If NetSensor is used with a remote motion sensor and the remote motion sensor includes an internal timer that resets when motion is detected, use the following Control Basic program. Set the time in line 30 (0 : 20 : 00) to match the time in the remote motion sensor.

```
10 IF+ VAR8 = -1 OR IN1 = 1 THEN START VAR9 , STOP A
20 IF VAR8 = -2 THEN START A
30 IF TIME-ON( A ) > 0:20:00 AND IN1 < 1 THEN STOP VAR9
```

**Local variables**

The Control Basic examples use only Local Variable A to control a timer.

**Variable points**

The variable points in these examples were chosen to be compatible with most factory programs supplied in KMD series controllers. Other variables can be used if the programs are modified.

**VAR8—NetSensorAux** This variable is associated with the Auxiliary function in the NetSensor. The Auxiliary function is explained in the installation and operation guide supplied with the NetSensor.

- ◆ A value of -1 indicates motion
- ◆ A value of -2 indicates no motion
- ◆ A value of 0 or 1 indicates the auxiliary function is active.

The value of the auxiliary function will remain at 0 for approximately 30 seconds after power is applied to the NetSensor.

**VAR9—RoomOccupied** When true, indicates the room is occupied based on motion.

**Table 1 KMD analog and digital variable settings**

|     | <b>Description</b>            | <b>Manual</b> | <b>Value</b> | <b>Units</b>    | <b>Label</b> | <b>Type</b> |
|-----|-------------------------------|---------------|--------------|-----------------|--------------|-------------|
| 1-7 | Not used for motion detection |               |              |                 |              |             |
| 8   | NetSensorAux                  |               |              | Unused          | NSaux        | Analog      |
| 9   | RoomOccupied                  |               |              | Unocc./Occupied | RmOcc        | Binary      |
| 10- | Not used for motion detection |               |              |                 |              |             |

## Programming BACnet controllers for occupancy

Use the following programs with BACnet BAC-5800, BAC-7000, BAC-7300 and BAC-7400 series controllers to set occupancy status based on motion detected in a room. Add the example Control Basic program to the controller to which the NetSensor is connected. In addition to adding the Control Basic programs, configure the value objects as described in section [Analog value objects](#) on page 4-81 and [Binary value objects](#) on page 4-81.

The controller to which the NetSensor is connected must also be programmed for setpoint, temperature and other functions. This programming is described in the installation and operation guide supplied with the NetSensor and in help for BACstage and TotalControl.

The following Control Basic example uses only the NetSensor to detect motion in a room.

```
10 IF+ AV8 = -1 THEN START BV9 , STOP A
20 IF AV8 = -2 THEN START A
30 IF TIMEON( A ) > 0:02:00 THEN STOP BV9
```

The following program determines occupancy using both a NetSensor and a remote sensor connected to input BI1. For this example, the remote sensor value switches without delay between periods of motion or no motion.

```
10 IF+ AV8 = -1 OR BV1 = 1 THEN START BV9 , STOP A
20 IF AV8 = -2 AND BV1 < 1 THEN START A
30 IF TIMEON( A ) > 0:20:00 THEN STOP BV9
```

If NetSensor is used with a remote motion sensor and the remote motion sensor includes an internal timer that resets when motion is detected, use the following Control Basic program. Set the time in line 30 (0:20:00) to match the time in the remote motion sensor.

```
10 IF+ AV8 = -1 OR BI1 = 1 THEN START BV9 , STOP A
20 IF AV8 = -2 THEN START A
30 IF TIMEON( A ) > 0:20:00 AND BI1 < 1 THEN STOP BV9
```

**Analog value objects**

**AV8—NetSensorAux** This variable is associated with the Auxiliary function in the NetSensor. The Auxiliary function is explained in the installation and operation guide supplied with the NetSensor.

- ◆ A value of -1 indicates motion
- ◆ A value of -2 indicates no motion
- ◆ A value of 0 or 1 indicates the auxiliary function is active.

The value of the auxiliary function will remain at 0 for approximately 30 seconds after power is applied to the NetSensor.

**Table 2 BACnet analog value objects**

| Value object | Description                   | Name         | Present Value | Units | Out of service |
|--------------|-------------------------------|--------------|---------------|-------|----------------|
| AV1–AV7      | Not used for motion detection |              |               |       |                |
| AV8          | NetSensorAuxillary            | NetSensorAux |               |       |                |
| AV9–AV40     | Not used for motion detection |              |               |       |                |

**Binary value objects**

**BV9—RoomOccupied** When true, indicates the room is occupied based on motion.

**Table 3 BACnet binary value objects**

| Value object | Description  | Name  | Present Value | Active<br>Inactive<br>text | Units           |
|--------------|--------------|-------|---------------|----------------------------|-----------------|
| BV1-BV8      | Not used     |       |               |                            |                 |
| BV9          | RoomOccupied | RmOcc |               |                            | Unocc./Occupied |
| BV10–AV40    |              |       |               |                            |                 |



## Application Note—AN0410A

---

### Measuring outside air temperature

This application note describes several methods for determining outside air temperature with values from multiple sensors or averaging several readings from one sensor.

---

Outside air temperature is used for many purposes such as setting airflow for an economizer, calculating boiler water temperature, or enabling chillers.

To improve accuracy or to provide redundancy the temperature readings from two or more sensors may be compared or averaged to determine an optimal value. In addition, averaging several readings will also provide some filtering of fluctuations due to local sensor variation.

### Installing outside sensors

The examples described in this application note do not require a particular sensor. When installing sensors, keep the following items in mind.

- ◆ Mount and connect the sensors as described in the installation guides provided with the sensors and the controllers.
- ◆ Sensors mounted on the north side of a building and out of direct sunlight will be the most accurate.
- ◆ The sensors need not be connected to the same controller. However, the example programs will have to be modified to reference the correct point or object for sensors not physically connected to the controller in which the Control Basic program is running.
- ◆ All sensors must be configured for the same unit of measure, either degrees Fahrenheit or degrees Celsius.

The KMC Controls model STE-1451 is suitable for the applications in the example Control Basic programs.



*Illustration 1 STE-1451 outside air temperature sensor*

## **Choosing between two sensors**

The simplest method to determine outside air temperature with multiple sensors is to use the Control Basic MIN function. This method assumes that the lower temperature value will be the correct temperature.

### **Example for KMD controllers**

The following Control Basic example is shown with WinControl XL Plus descriptions.

```
10 OutsideAirTemp = MIN( TempSensor1 , TempSensor2 )
```

**TempSensor1 and TempSensor2** These points may be either physical inputs or variables.

**OutsideAirTemp** This variable holds the results of the calculation. The value stored in this variable is typically used by other controllers on the network as the single global value for outside air temperature.

### **Control Basic examples for BACnet controllers**

The following Control Basic example is entered with mnemonics for the BACnet objects in KMD controllers.

```
10 AV10 = MIN( AI1 , AI2 )
```

**Inputs AI1 and AI2** The input objects may be either physical controller analog inputs or an analog value object that represents a temperature input.

**Value object AV10** This value object stores the results of the calculation. The present value of this object is typically used by other controllers on the network as the single global value for outside air temperature.

## Averaging values from multiple sensors

Averaging the values of multiple sensor will provide some filtering of fluctuations caused by local sensor variation. The following example averages the values of two sensors and then calculates a value that can be shared with the rest of the network devices. It also tests each sensor for a valid response. If the temperature reading is out of range for a sensor, the sensor is flagged and its reading is ignored.

### Example for KMD controllers

The following Control Basic example is shown with WinControl XL Plus labels.

*Example:*

```
10 OAT = AVG( TMPSNS1 , TMPSNS2 )
20 IF TMPSNS1 > 140 OR TMPSNS1 < -40 THEN START A ELSE STOP A
30 IF TMPSNS2 > 140 OR TMPSNS2 < -40 THEN START B ELSE STOP B
40 IF A THEN OAT = TMPSNS2
50 IF B THEN OAT = TMPSNS1
60 DALARM A , 20 , OUTSIDE SENSOR FAILURE ON Sensor 1
70 DALARM B , 20 , OUTSIDE SENSOR FAILURE ON Sensor 2
```

**TMPSNS1 and TMPSNS2** These points may be either physical inputs or variables.

**OAT** This variable holds the results of the calculation. The value stored in this variable is typically used by other controllers on the network as the single source for outside air temperature.

### Control Basic examples for BACnet controllers

The following Control Basic example is entered with mnemonics for the BACnet objects in KMD controllers.

**Analog inputs AI1 and AI2** The input objects may be either physical controller analog inputs or an analog value object that represents a temperature input.

**Value object AV21** This value object holds the results of the calculation. The value stored in this object is typically used by other controllers on the network as the single source for outside air temperature.

*Example:*

```
10 AV21 = AVG( AI1 , AI2 )
20 IF AI1 > 140 OR AI1 < -40 THEN START A
30 IF AI2 > 140 OR AI2 < -40 THEN START B
40 IF A THEN AV21 = AI2
50 IF B THEN AV21 = AI1
```

## Calculating a running average for an outside temperature

To minimize the effects of temperature fluctuations, use a running average value of outside air temperature. The following program measures the outside temperature at 10-minute intervals, stores the readings in local variables and then calculates the average temperature for the last hour.

### Example for KMD controllers

The following Control Basic example is entered with WinControl XL Plus descriptions.

```
10 IF INTERVAL( 0:10:00 ) THEN A = B , B = C , C = D , D = E , E = F , F = TempSensor1
20 IF POWER-LOSS OR A = 0 THEN A = TempSensor1 , B = TempSensor1 , C = TempSensor1 , D = TempSensor1 , E = TempSensor1 , F = TempSensor1
30 OATAverage = AVG( A , B , C , D , E , F )
100 END
```

**TempSensor1** Outside air temperature

**OATAverage** The average outside temperature over the past 60 minutes. This value is a stable temperature reading well suited for use in temperature resets.

**Local variables** The local variables (A, B, C....) are held only in memory by the controller. After a power loss or upon sending the program, the values of these variables become 0. Line 20 keeps these values to a realistic average for the initial hour after the program is sent and in the event of power failure in the controller.

### Control Basic examples for BACnet controllers

The following Control Basic example is entered with mnemonics for the BACnet objects in KMD controllers.

```
10 IF INTERVAL( 0:10:00 ) THEN A = B , B = C , C = D , D = E , E = F , F = AI1
20 IF POWERLOSS OR A = 0 THEN A = AI1 , B = AI1 , C = AI1 , D = AI1 , E = AI1 , F = AI1
30 AV23 = AVG( A , B , C , D , E , F )
100 END
```

**Input AI1** This object may be either a physical input or value object that represents the outside air temperature.

**Value object AV23** The average outside temperature over the past 60 minutes. This value is a stable temperature reading well suited for use in temperature resets.

**Local variables** The local variables (A, B, C....) are held only in memory of the controller. After a power loss or upon sending the program, the values of these variables become 0. Line 20 keeps these values to a realistic average for the initial hour after the program is sent and in the event of power failure in the controller.

# Index

## Numerics

- 4-20 mA current source input 1-11
- 4-20 mA current source output 1-13

## A

- AHU Controller 3-127
- Analog Outputs 1-8
- Application drawings
  - KMD7001 3-23
  - KMD-7002 3-42
  - KMD7003 3-64
  - KMD-7011 3-86
  - KMD-7013 3-105
  - KMD7051 3-23
  - KMD-7052 3-42
  - KMD7053 3-64
  - KMD-7101 3-124
  - KMD-7301 3-139
  - KMD-7302 3-156
  - KMD-7401 3-169
  - STE-1001 refrigerant monitor 4-74
  - Tri-state valve actuator 4-17
- Application notes
  - AN0404B 3-3, 3-43
  - AN0404C 3-29
  - AN0404D 3-67
  - AN0404E 3-89
  - AN0404F 3-109
  - AN0404G 3-127
  - AN0404H 3-143
  - AN0404I 3-157
  - AN0504A 4-3
  - AN0504B 4-9
  - AN0504C 4-13
  - AN0504E 4-19
  - AN0504F 4-39
  - AN0504G 4-43
  - AN0504H 4-45
  - AN0504I 4-55
  - AN0504J 4-59
  - AN0504L 1-3
  - AN0604A 4-27
  - AN0604B 4-53
  - AN0604C 4-65
  - AN0604D 1-15
  - AN0909A 4-31

## Applications

- Boiler reset 4-45
- Calculating enthalpy 4-9
- Calculating position with tables 4-63
- Calculating sunrise and sunset 4-59
- Discharge reset 4-53
- KMD-1261 4-75
- KMD-1281 4-75
- KMD-5110 4-9
- KMD-5501 4-9
- KMD-6000 4-9
- KMD-7001 3-3
- KMD-7002 3-29
- KMD-7003 3-43
- KMD-7011 3-67
- KMD-7013 3-89
- KMD-7051 3-3, 3-43
- KMD-7052 3-29
- KMD-7101 3-109
- KMD-7102 3-109
- KMD-7301 3-127
- KMD-7302 3-143
- KMD-7401 3-157
- Lead/lag programs 4-19
- Lighting control 4-59
- Motion sensing 4-75
- Optimal start/stop routine 4-27
- Refrigerant monitoring 4-65
- SLE-1001 4-65
- Tri-state valve actuator 4-13

## B

### BACnet

- Device instance number 2-8
  - Internetwork 2-10
  - Loops 2-15
  - MAC address 2-8
  - Network numbers 2-11
  - Networks 2-7
  - Router port 2-14
- BACnet Broadcast Management Device 2-20
- BBMD 2-20
- Boiler reset 4-45

## C

- Calculating enthalpy 4-9
- Chilled water 3-109
- Cimetrics BACstac 2-21
- Connecting
  - Inputs 1-4
  - Network segments with fiber optic cable 2-31
  - Outputs 1-8
  - To 24-volt transformers 1-17

**D**

Device instance number 2-8  
 Discharge reset 4-53  
 Dual-duct VAV 3-29

**E**

Economizer 3-134, 3-151

**F**

Fan Coil Unit 3-109  
 Fiber optic 2-31  
 FirstWatch 4-65

**H**

## HCM

KMD-7001 3-5  
 KMD-7002 3-31  
 KMD-7003 3-45  
 KMD-7011 3-69  
 KMD-7013 3-91  
 KMD-7051 3-5  
 KMD-7052 3-31  
 KMD-7053 3-45  
 KMD-7101 3-110  
 KMD-7102 3-110  
 KMD-7301 3-128  
 KMD-7302 3-144  
 KMD-7401 3-157

Heat Pump Controller 3-157

Heating-cooling-changeover 3-89

Hot water 3-109

**I**

## Inputs

4-20 mA Current source 1-11  
 Active voltage source 1-4  
 DC voltage 1-4  
 Four-wire 4-20 ma 1-12  
 Pasive 1-5  
 pulse 4-43  
 Switch contacts 1-6  
 Thermistor 1-6  
 Three-wire 4-20 mA 1-12  
 Two-wire 4-20 mA 1-11  
 Variable resistance 1-7

Internetwork 2-10

**K**

KMD-5501

Connecting to a modem 4-3

**L**

Lead/lag programs 4-19

four units 4-24  
 three units 4-21  
 two units 4-19

Lighting control 4-59

Loops in BACnet networks 2-15

**M**

MAC address 2-8

MS/TP networking 2-24

**N**

NetSensor

as a balancer service tool 3-176  
 detecting motion 4-75  
 using with preprogrammed controllers 3-171

Network numbers 2-11

Network segments

Connecting with fiber optic cable 2-31

**O**

Optimal start/stop routine 4-27

Outputs

4-20 mA 1-13  
 AC devices 1-9  
 Analog 1-8  
 Direct connection of DC devices 1-8  
 Triac 1-9  
 With override cards 1-8

Outside air temperature 4-83

**P**

PAD route 2-17

Passive inputs 1-5

Port

BACnet router 2-14  
 IP ports for KMC digital protocol 2-5

Position feedback using tables 4-63

Program listing

KMD-7001 3-20  
 KMD-7002 3-41  
 KMD-7003 3-61  
 KMD-7011 3-83  
 KMD-7013 3-102  
 KMD-7051 3-20  
 KMD-7052 3-41  
 KMD-7053 3-61  
 KMD-7101 3-121  
 KMD-7102 3-121  
 KMD-7301 3-138  
 KMD-7302 3-154  
 KMD-7401 3-167

Programming override timers 4-39

Programming with tables 4-59

Pulse inputs 4-43

## **R**

Refrigerant monitoring 4-65

Reheat 3-3, 3-43

Reset, boiler 4-45

Roof top unit controller 3-143

## **S**

Signal conditioning 4-55

Single Duct VAV 3-3, 3-43

Slave Duct VAV 3-29

SLE-1001 4-65

Sunrise and sunset 4-59

Switch contact inputs 1-6

## **T**

Tables, example programs 4-59

Temperature

- Boiler reset 4-45

- Discharge air 4-53

- Outside air 4-83

Three-stage reheat 3-3

Tier 1 protocols 2-4

Tier 2 fiber optic connection 2-31

Time proportional reheat 3-3, 3-43

Transformer

- Load capacity 1-21

- Primary voltage 1-16

- Selection 1-15

Triac outputs 1-9

Tri-state valve actuator 4-13

## **V**

Variable resistance input 1-7

VAV

- Dual-duct 3-29

- Heating-cooling-changeover 3-3, 3-43, 3-68, 3-89

- Three-stage reheat 3-3, 3-68

- Time proportional reheat 3-3, 3-43, 3-68

Velocity limits 3-4, 3-30, 3-44, 3-68, 3-90

Velocity sensor 3-4, 3-29, 3-44, 3-68, 3-90

Voltage drop 1-22

## **W**

Warm-up 4-31

