CT435

PC Board Mount Temperature Controller

Features

- Two RTD temperature sensor inputs: Pt100 or Pt1000.
- Wide temperature sensing range: -70°C to 650°C.
- All controller features are configurable through the Modbus over UART interface.
- Two independent solid state open drain outputs – up to 5A each.
- Each output individually configurable for any variation of PID, On/Off, or Alarm control.
- Auto-tune feature estimates PID coefficients for several control types.
- 32-bit microprocessor executes both PID loops simultaneously at individually configurable rates up to 25 times/second.
- Addressable Modbus protocol allows for multiple units connected on one set of UART lines.
- No additional heat sinking required.
- Small package designed for PCB mounting.
- Operates from a 5V supply.
- Low cost.

Typical Applications

- Industrial or commercial heating
- Electronics cabinet warming
- Medical diagnostic equipment
- Semiconductor manufacturing

Design Aids

- AC218154 CT435 evaluation board
- Windows configuration and evaluation software

Related Models

- CT335 – Proportional PCB-mount temperature controller
- CT425 – PID universal controller

Description

The CT435 is an OEM micro-processor based PID temperature controller that offers two independent sensor inputs and two outputs. This low cost, PCB mount style PID controller is highly flexible and offers many configuration options. Using the Modbus over UART interface, system parameters, sensor temperatures, and output status may be read and/or written, allowing for complete system integration with existing micro-processors.

A 32-bit processor allows up to two PID loops to operate simultaneously at individually configurable loop rates up to 25Hz. An auto-tune feature generates PID coefficients, and configuration can be performed through a Windows application or through the Modbus over UART interface.

Operation

The CT435 controller can be configured to any variation of PID, On/Off control, or Alarm. On/Off control offers faster reaction time and better accuracy over thermostats. PID control minimizes temperature overshoot and offers steadier temperature control by utilizing proportional, integral, and derivative control factors. The inputs and outputs may be configured in any fashion, and all parameters are read/written through the addressable Modbus over UART interface. The controller and heaters may be powered from the same supply or separate supplies, as long as they share a common ground.

Electrical Characteristics and Ratings:

Sensor Inputs:
100 Ω at 0ºC Pt RTD, 2-leads (0.00385 TCR)
1000 Ω at 0ºC Pt RTD, 2-leads (0.00385 TCR)
2-wire connection
Open and shorted sensor detection

Electrical:
Input power: 5 to 24VDC, 20mA typical, 40mA max
Outputs: Two open drain solid state outputs
Max 60 VDC rated switching voltage

Number of Outputs in Use	Controller Supply Voltage	Ambient Temperature	Current Rating
1 Output	5-12 VDC	25°C	7 A
		70°C	4 A
	12-24 VDC	25°C	6 A
		70°C	3 A
2 Outputs	5-12 VDC	25°C	5 A
		70°C	3 A
	12-24 VDC	25°C	5 A
		70°C	2.5 A

Environmental:
Operating temperature: -40 to 70ºC (-40 to 158ºF)
Storage temperature: -55 to 85ºC (-67 to 185ºF)
Relative humidity: 90%, non-condensing

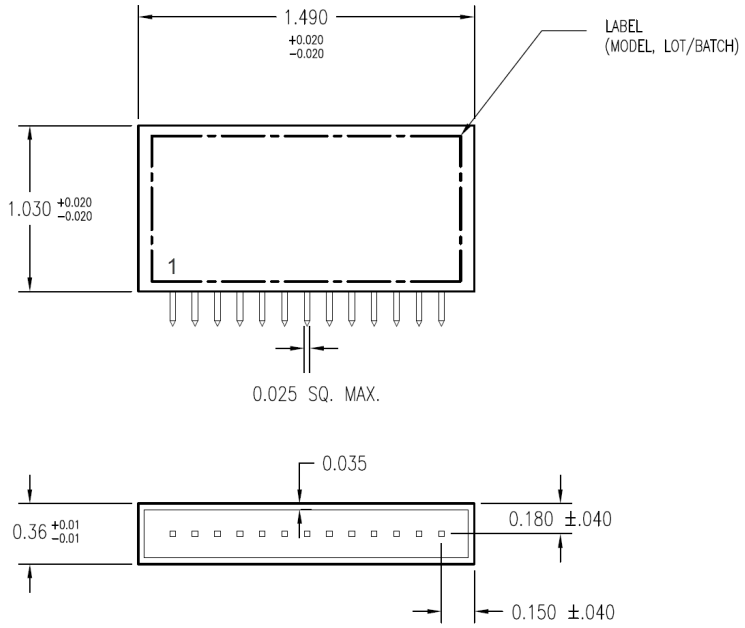
Accuracy:
25°C ambient: ±0.25º C of range
Full range ambient: ±1.5º C of range
System stability determined by overall system.

Communication:
Modbus over UART – 115.2kbps, no flow control, 8N1

Measurement Range:
-70°C-650°C (-94°F-1202°F)

Package:
Enclosure: ABS case, epoxy potted
Dimensions: 1.49x1.03x0.36”
Mounting: Pins on 0.1” center for mounting on PCB

Dimensions:



Pin Configuration:

Pin #	Function
1	POWER (V+)
2	GND
3	GND
4	Output 1
5	Output 2
6	Status output pin 1 (3.3V)
7	UART RX (3.3V, 5V tolerant)
8	UART TX (open drain)
9	Status output pin 2 (3.3V)
10	RTD input 2 (+)
11	RTD input 2 (-)
12	RTD input 1 (+)
13	RTD input 1 (-)

MINCO
A critical component of your success™

Common Application Diagram:

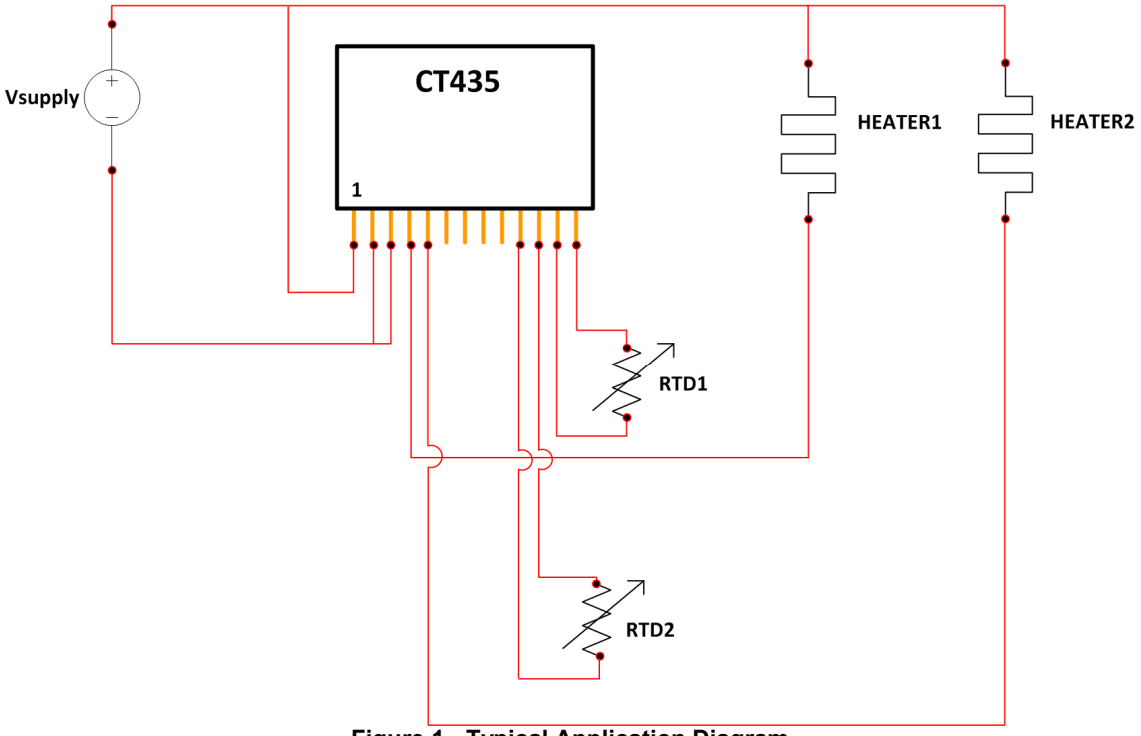


Figure 1 - Typical Application Diagram

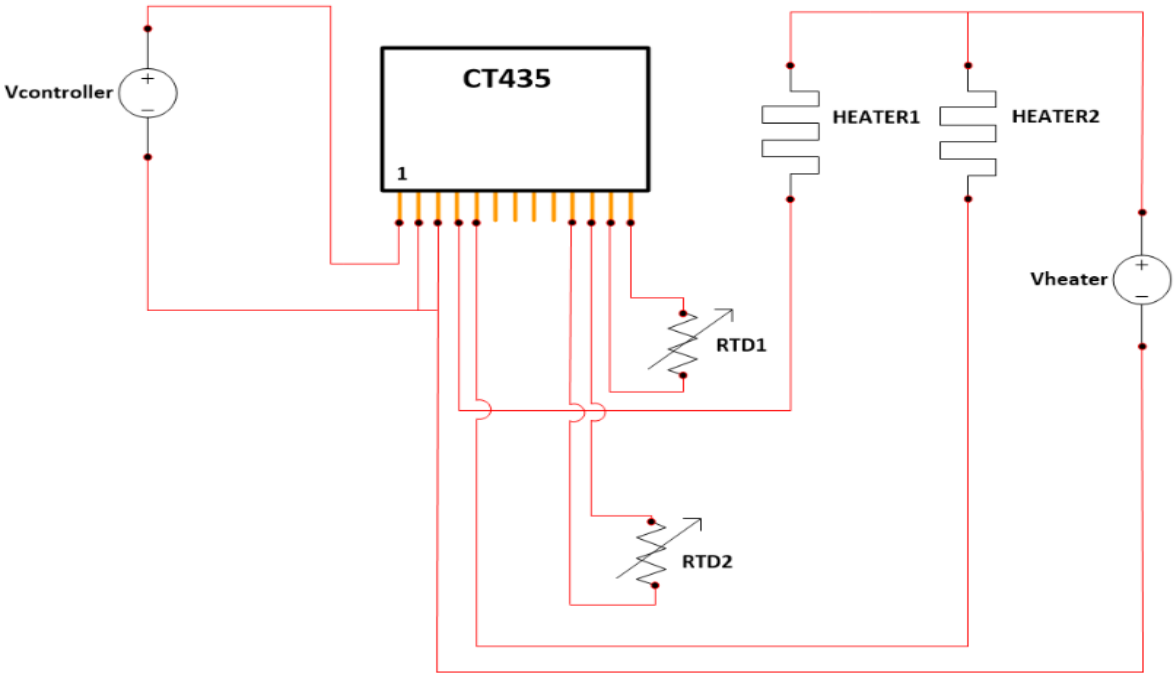


Figure 2 - Alternate Application Diagram

1. Application Information:

a. Inputs:

The CT435 can use two Platinum RTD temperature sensors (100 or 1000 Ω at 0°C, .00385 TCR) of the same type. The input type compatibility is determined in the hardware of the CT435 and must be ordered as such – it is not configurable in the application. The sensor inputs are not electrically isolated from the rest of the CT435 circuitry; therefore, care must be taken to not create a situation where an unintended short circuit is created.

Each sensor channel has two pin connections. The RTD is simply connected to these two pins, polarity does not matter. If the RTD has extra wires for lead compensation, these wires could be left disconnected or grouped together on the same pin. Care should be taken to ensure excess resistance is not added to the RTD lines through PCB traces or connections. Lead compensation can be performed by specifying an offset in the CT435's configuration.

b. Outputs:

Two open drain, solid state control outputs are provided on pins 4 and 5. Being open drain, the outputs act as a current "sink" to ground. **The heater supply must share a ground with the CT435 controller.** See the wiring diagram for an example circuit. The outputs are MOSFET-based for DC switching only, and are rated for a maximum of 60 VDC. See the specifications or de-rating curves for current ratings under varying conditions. The outputs are not isolated from the rest of the CT435 circuitry, and care must be taken.

Both outputs will support PID, On/Off, and Alarm functions. Each can operate independently from the other at a maximum PID loop rate of 25Hz (loop time of 40ms).

c. External components:

Very few external components are required for operation. They include:

- An external pull-up resistor on the UART TX line. This communication line is open drain on the CT435, therefore a pull-up resistor is required. This resistor should pull the UART TX line to the voltage level being used for UART communication.
- External pull-down resistors are recommended for output status pins 1 and 2. These pins operate at a 3.3V logic level and should not be allowed to float and should not be pulled high. If both pins are pulled high upon application of power, the CT435 will enter a manufacturing mode for firmware upgrades. This should only be performed if instructed by Minco.
- Any additional protection components. The RTD input lines are protected with transient voltage suppressors, and the power input has a clamping zener diode as well reverse polarity protection. All other pins contain no internal protection, but may be added by the user if deemed necessary.

d. Status pin functions:

Status pins 1 and 2 provide basic information regarding the status of the CT435 and may or may not be used. If both status pins are pulsing in unison, this could indicate a hardware failure although very low supply voltage may also cause this symptom. In this case, both outputs are de-energized and a power cycle is necessary to reset the unit.

e. Mounting:

The CT435 is a through-hole PCB-mount PID controller with 13 pins on a 0.1" spacing. It is compatible with hand soldering, wave soldering, and selective soldering. The pin-in-paste mounting method is not recommended and may damage the unit. The CT435 is epoxy potted and may be water washed accordingly. Active chemicals in cleaning systems such as a vapor degreaser may break down the adhesive of the label, epoxy, or plastic case and should be treated with caution.

f. PCB design considerations:

When designing a PCB for use with the CT435, several items should be considered:

- Heat and power dissipation – As with any electronic component, the CT435 will dissipate power and rise in temperature as it switches current. At the maximum ratings, the internal temperature of the controller may reach 90°C, and the external surfaces may reach 70°C. To aid in power dissipation, ensure that heater traces and ground traces are designed to handle desired current.
- Electromagnetic interference – The RTD inputs of the CT435 are relatively well filtered from EMI, but in high-noise environments, further RTD and power input filtering may be required.
- RTD traces and connections – Any additional resistance introduced into the RTD lines due to traces or connection will shift the RTD measurements. Care should be taken to minimize the impact or an offset should be used in the CT435 to ensure accurate readings.

2. Configuration:

Configuration is performed through the UART interface; this may be performed using the Windows application or by communicating with the CT435 using Modbus commands. The Evaluation Board simplifies this by converting the UART interface to USB for connection to any computer. Configuration parameters may be changed at any time during operation, even when actively controlling an application. All settings may be stored to non-volatile memory so removal of power does not cause loss of configuration.

All settings and control parameters may be written to or read from. Current temperature sensor readings are only available as read-only registers. A full listing of the registers that may be written or read can be found in section 5.

The following settings and registers are available.

a. General settings:

- **Modbus Address –** An unsigned 16-bit integer used to identify a given unit during Modbus communications. Valid addresses are 1-247.
- **Temperature Scale –** Determines if the CT435 will interact with the Windows application in Celsius or Fahrenheit scale. All internal calculations and direct UART interactions utilize the Celsius scale.
- **Type –** Set to either Pt100, Pt1000, or Disabled for platinum 100 ohm, platinum 1000 ohm, or no sensor connected, respectively.
- **Offset –** Applies an offset to the respective sensor input. This could be used for wire lead compensation.

b. Output setting (applicable to each output independently):

- **Source –** Determines which sensor input the respective output will use. Multiple outputs may share the same sensor input.
- **Function –** Disabled, PID, On/Off, or Alarm. On/Off and alarm function similarly with the only difference being how the Hysteresis is used. See the Hysteresis section below for detail. When set to disabled, the output is de-energized regardless of the Reverse Acting setting.
- **Reverse Acting –** Setting this to True causes the respective output to behave in the opposite manner, i.e. normally closed instead of normally open. This may be useful for cooling applications.
- **Over/Under –** Determines whether the output engages when the sensor temperature is over or under the Setpoint. This applies to the Alarm Function only.
- **Setpoint –** Determines the temperature that the selected sensor input is maintained to for PID and On/Off Functions. In the case of the Alarm Function, when the sensor is above or below this temperature the output will activate depending upon if Over or Under is selected, respectively.
- **Hysteresis –** Determines the band around the Setpoint where the output engages and disengages. When using the On/Off function, the Hysteresis value is evenly divided around the Setpoint. For example, if the Setpoint is 55.0°C and the Hysteresis is 0.1°C, the output will engage at 49.95°C and disengage at 50.05°C.

When using the Alarm function, the Hysteresis is placed on the side of the Setpoint that does not engage the output. For example, if the Setpoint is 55C, Over/Under is Over, and Hysteresis is 0.1C, the output will engage when the sensor exceeds 55.0C, but will disengage when the sensor is less than 54.9C.

The hysteresis value is meaningless to the PID Function and therefore is greyed out if that Function is used.

- **Minimum Duty Cycle –** Determines the minimum duty cycle, in percent, that the output will reach. This is the duty cycle at the output terminals with respect to the Reverse Acting setting. Normally this value is set to zero, however there may be some applications that require a minimum of more than zero. When the Function is set to Disabled, this setting is irrelevant.

- **Maximum Duty Cycle –** Determines the maximum duty cycle, in percent, that the output will reach. This is the duty cycle at the output terminals with respect to the Reverse Acting setting. Normally this value is set to 100; however, there may be some applications that require a maximum of less than 100. When the Function is set to Disabled, this setting is irrelevant.

c. PID settings (applicable to each output):

- **Method –** Determines if the PID coeffecients (Kp, Ki, Kd) will be entered manually by the user, or if they will be automatically generated from Autotune data by one of several algorithms.

  The information gained from an Autotune procedure is saved so the Method may be changed later by the user. As with all other settings, this is preserved through a power cycle only if written to nonvolatile memory.

  It should be noted that the various options for the Method are the common names given to the various algorithms. For example, Minimal Overshoot may or may not result in the least overshoot among the other available methods due to variability in application, and accuracy of the Autotune performed. It's recommended to try different Methods to find the best performer, and if desired, manually tweak from there.

  An easy, though generic option is to simply set Kp, Ki, and Kd to 100, 2, and 0, respectively. These are the default values the CT435 is shipped with, and work reasonably well in many applications. Kd could instead be set to 1000 for full PID. Again, these values are generic and are by no means intended to work in every application.

- **Kp –** The proportional coefficient.

- **Ki –** The integral coefficient.

- **Kd –** The derivative coefficient.

- **Loop Time –** The loop time for the PID Function, in milliseconds. This is the amount of time between each PID Function calculation, i.e. how often output duty cycle is recalculated based on current conditions. A lower number will theoretically result in faster response time. However, a Loop Time that is too low may result in an unstable system. In general, slow moving processes should have a higher Loop Time. Minimum loop time is 40ms. Maximum is 1000 seconds.

d. Autotune settings:

- **Autotune Band –** Determines the number of degrees above and below the Intended Setpoint the output is toggled during Autotuning, respectively. Generally, this should be set as high as tolerable to improve Autotune results, although 0.5°C is frequently sufficient.

- **Autotune Output Step –** Determines the output duty cycle while the output is engaged. The Autotune algorithm will cycle the output between fully de-energized ("off") and the Output Step. The purpose of the Autotune Output Step is to limit heater power in applications where 100% duty cycle would cause very rapid heating, making autotuning difficult. In general, keep this value as low as possible while allowing the heater to receive enough power to reach the intended temperatures.

- **Intended Setpoint –** Determines the temperature around which the Autotune algorithm will operate. The purpose of this value is to first bring the heater to the approximate temperature at which it would normally be operated before Autotuning, which may improve Autotune results.

- **Autotune Ku –** This is the gain factor determined during the Autotuning process. The value is stored in the CT435 to be used in calculating the PID parameters. See Table 1 below for these calculations.

- **Autotune Tu –** This is the oscillation period determined during the Autotuning process. The value is stored in the CT435 to be used in calculating the PID parameters. See Table 1 below for these calculations.

MINCO
A critical component of your success™

- **Autotune Type –** This register is used by the software interface when using the Evaulation Board to store what PID Method was last used. This should not be over-written, but will not cause performance issues if it is modified.

e. How Autotune works:

The Autotune feature automatically generates PID coefficients based on measurements performed using the actual sensor and load. Sometimes the values generated by Autotune are sufficient as the final operating values, but for better control, these may have to be manually tweaked.

The CT435 Autotune uses the Ziegler-Nichols method. Autotune operates by first engaging the output to the duty cycle defined by Autotune Output Step until the temperature reaches the Intended Setpoint. The output is left in the engaged state until the temperature exceeds the sum of Intended Setpoint and Autotune Band. The output is disengaged, and the temperature will begin to fall and the output is engaged again once the temperature is less than the difference between Intended Setpoint and Autotune Band. The output then continues to cycle in the same manner for a period of time dependent on the application.

Essentially, the output is toggled to cause the temperature to vary above and below the Intended Setpoint by approximately the Autotune Band. This process is repeated several times until the results are consistent.

The time between temperature peaks, the difference between temperature peaks and valleys, and the output drive duty cycle are used to determine the Ku and Tu values of the Ziegler-Nichols method. Below in Table 1 are the calculations for the Kp, Ki, and Kd parameters for various PID control Methods based on the Ku and Tu values determined during the Autotune process.

Once Autotuning is complete, control is immediately returned to the Control Type selected for that output.

Ziegler-Nichols Method			
Control Type	K_p	K_i	K_d
P	$0.5K_u$	-	-
PI	$0.45K_u$	$1.2K_p/T_u$	-
PD	$0.8K_u$	-	$K_pT_u/8$
Classic PID	$0.6K_u$	$2K_p/T_u$	$K_pT_u/8$
Pessen Integral Rule	$0.7K_u$	$0.4K_p/T_u$	$0.15K_pT_u$
Medium Overshoot	$0.33K_u$	$2K_p/T_u$	$K_pT_u/3$
Minimum Overshoot	$0.2K_u$	$2K_p/T_u$	$K_pT_u/3$

Table 1 - PID Parameter Calculations

There are two main methods that can be implemented in order to Autotune a given application:

- Autotuning can be performed by using the CT435 in the application by writing to the Autotune Band, Autotune Step, and Intended Setpoint registers followed by initiating the Autotuning process by setting the desired Autotune Start register(s) to 0x0001. The Autotuning Status register(s) can then be monitored to determine the progress, and once complete, the CT435 will set the Autotune Start register(s) to 0x0000.

- Autotuning can also be performed by using the CT435, as well as the heater(s) and sensor(s) used in the application, to the Evaluation Board. The software interface can then be used to easily initiate and monitor the Autotuning process as well as calculate and store the PID parameters. This method is recommended, if possible.

f. Submitting settings changes:

It is important to note that changes made to the settings of the CT435 are stored in RAM until written to the non-volatile memory. See the Communications section below for instructions to write the current settings to the non-volatile memory.

3. Switching Current De-Rating Curves:

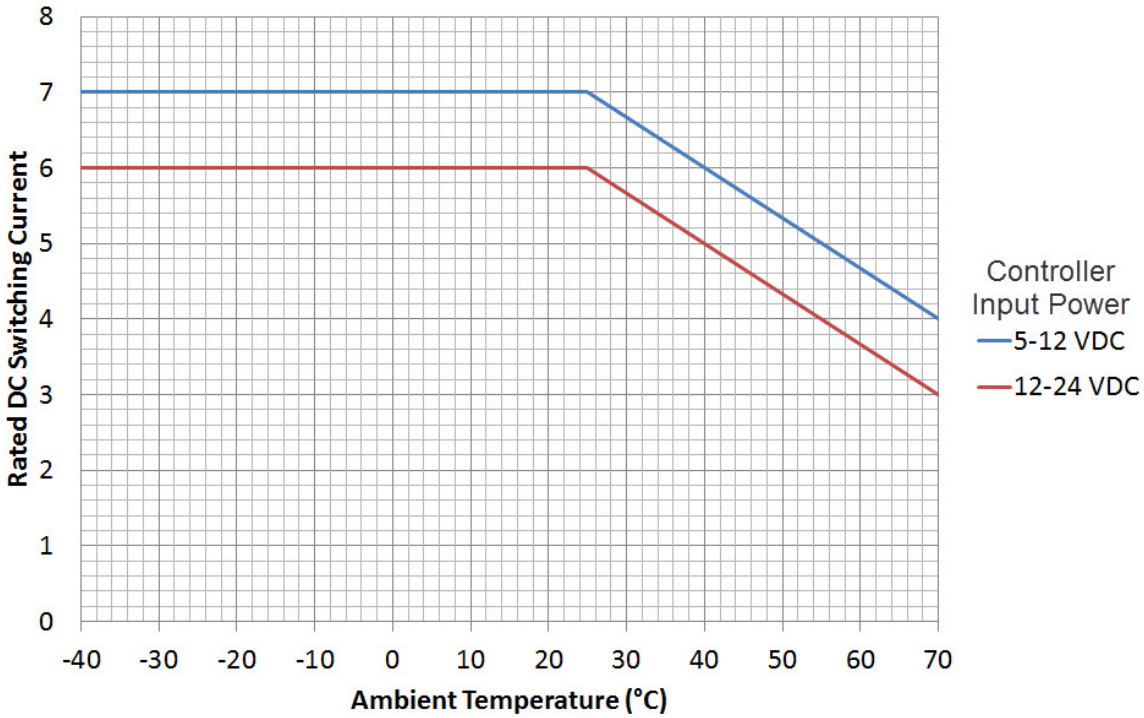


Figure 3 - Single Output De-rating Curve

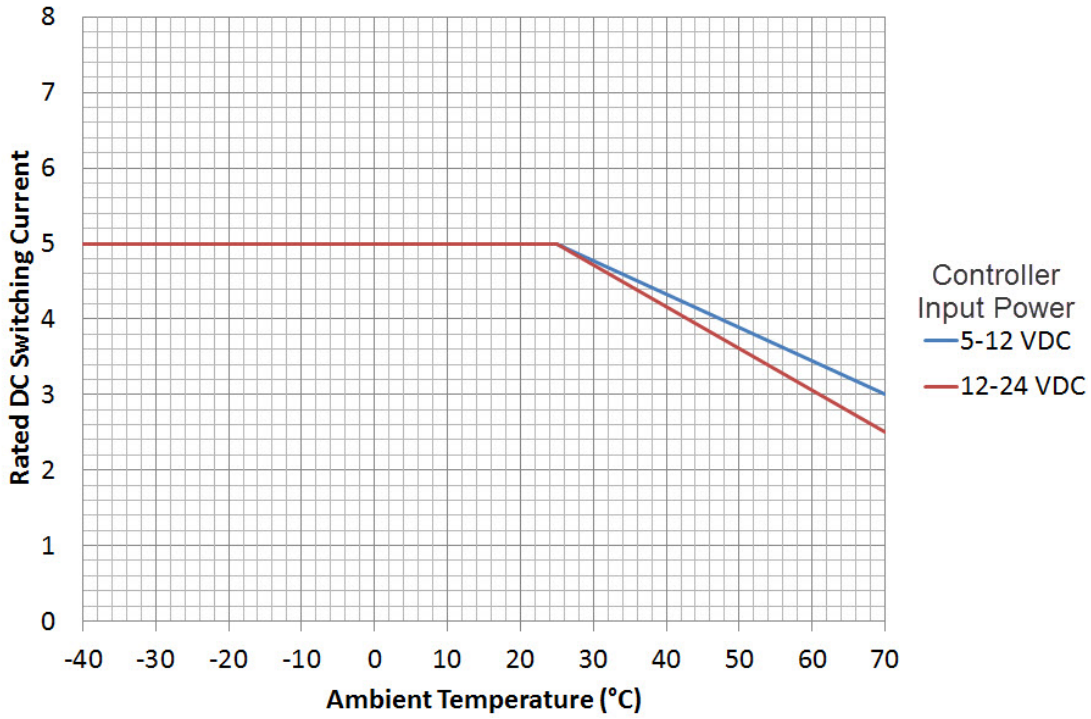


Figure 4 - Dual Output Derating Curve

MINCO
A critical component of your success™

4. Communications:

a. UART:

The CT435 uses a standard UART interface operated at 115.2kbps with no flow control using 8 data bits, no parity, and 1 stop bit. This requires only the RX line, TX line, and a ground. The UART RX pin on the CT435 is 3.3V and 5V logic tolerant, and the UART TX line is open drain, requiring an external pull-up to the logic level voltage.

b. Modbus:

The MODBUS interfaces on the CT435 allow access to all configuration values, sensor readings, and output status. The configuration values may be read or written, while the remaining values are read-only. Configuration values are considered to be in Holding Registers, while the read-only status values are considered to be in Input Registers.

If a written value is not within the valid range, the CT435 will disregard the change to that register. Register limitations and default values can be found in Table 2 - Modbus Holding Register Listing.

A Function Code is a designator in the Modbus command that specifies the type of command being sent. Three function codes are supported by the CT435: 0x03, 0x04, and 0x10. These are discussed in greater detail in the following sections.

Modbus addresses are allowed in the range from 1 to 247. If the CT435 receives a Modbus message intended for a different address, it will be disregarded.

c. Holding registers:

The holding registers contain the configuration values. These may be read or written by using function codes 0x03 or 0x10, respectively.

Changes made to any holding register via Modbus are not automatically saved to non-volatile memory, but rather exist only in RAM. In order to save the configuration to non-volatile memory after a change or multiple changes, the following command must be issued, which is essentially writing the value 0x1234 to the non-existent address of 0x0100:

(Modbus Address) 0x10 0xFF 0xE5 0x00 0x00 0x01 0x02 0xA0 0x0B (2 byte CRC)

If multiple changes are to be made, it's best to write those changes first, and then issue the above command to save all changes at once.

The format of the command to read values using Function Code 0x03 is as follows:

W	Modbus ID of the CT435
0x03	Function Code
X	16 bit starting register address, MSB first
Y	Number of 16 bit values to return, MSB first
CRC	2 byte cyclic redundancy check value

Example
Read two values from Modbus address 0x01, starting at address 0x0001:

0x01 0x03 0x00 0x01 0x00 0x02 0x95 0xCB

The format of the command to write values is as follows:

W	Modbus ID of the CT435
0x10	Function Code
X	16 bit starting register address
Y	16 bit value representing the number of 16 bit registers to write
Z	8 bit value representing the number of 8 bit data values to follow. This value equals (2 * y)
Value 1	The first 16 bit value to be written
Value n	Additional 16 bit values to be written
CRC	2 byte cyclic redundancy check

d. Input/output registers:

The input/output registers contain read-only values, such as sensor input values and current output states. These are read-only by using Function Code 0x04.

The format of the command to read values using Function Code 0x04 is as follows:

W	Modbus ID of the CT435
0x04	Function Code
X	16 bit starting register address, MSB first
Y	Number of 16 bit values to return, MSB first
CRC	2 byte cyclic redundancy check value

MINCO
A critical component of your success™

5. Modbus Register Listing:

a. Holding registers:

Register Address	Description	Value Format	Valid Range
0x0000	Input 1 RTD Type (Read-Only)	C	See Table 4 - Value Format Key
0x0002	Input 1 Sensor Offset	A	-10 to 10°C
0x0004	Input 2 RTD Type (Read-Only)	C	See Table 4 - Value Format Key
0x0006	Input 2 Sensor Offset	A	-10 to 10°C
0x0008	Output 1 Source	D	See Table 4 - Value Format Key
0x000A	Output 1 Control Type	E	See Table 4 - Value Format Key
0x000C	Output 1 Setpoint	A	-70 to 650°C
0x000E	Output 1 Hysteresis	A	0 to 100°C
0x0010	Output 1 Reverse Acting	F	See Table 4 - Value Format Key
0x0012	Output 1 PID Kp	A	-1,000,000 to 1,000,000
0x0014	Output 1 PID Ki	A	-1,000,000 to 1,000,000
0x0016	Output 1 PID Kd	A	-1,000,000 to 1,000,000
0x0018	Output 1 Alarm Over/Under	G	See Table 4 - Value Format Key
0x001A	Output 1 Autotune Start	H	See Table 4 - Value Format Key
0x001C	Output 1 Autotune Type	I	See Table 4 - Value Format Key
0x001E	Output 1 Autotune Ku	A	-10,000 to 10,000
0x0020	Output 1 Autotune Tu	A	-10,000 to 10,000
0x0022	Output 1 Autotune Band	A	0 to 720
0x0024	Output 1 Autotune Temperature	A	-70 to 650°C
0x0026	Output 1 Autotune Step	A	0 to 1000 (in tenths of a percent)
0x0028	Output 1 Minimum Duty Cycle	B	0 to 1000 (in tenths of a percent)
0x002A	Output 1 Maximum Duty Cycle	B	0 to 1000 (in tenths of a percent)
0x002C	Output 1 PID Loop Time	B	40 to 10,000 (in ms)
0x002E	Output 2 Source	D	See Table 4 - Value Format Key
0x0030	Output 2 Control Type	E	See Table 4 - Value Format Key
0x0032	Output 2 Setpoint	A	-70 to 650°C
0x0034	Output 2 Hysteresis	A	0 to 100°C
0x0036	Output 2 Reverse Acting	F	See Table 4 - Value Format Key
0x0038	Output 2 PID Kp	A	-1,000,000 to 1,000,000
0x003A	Output 2 PID Ki	A	-1,000,000 to 1,000,000
0x003C	Output 2 PID Kd	A	-1,000,000 to 1,000,000
0x003E	Output 2 Alarm Over/Under	G	See Table 4 - Value Format Key
0x0040	Output 2 Autotune Start	H	See Table 4 - Value Format Key
0x0042	Output 2 Autotune Type	I	See Table 4 - Value Format Key
0x0044	Output 2 Autotune Ku	A	-10,000 to 10,000
0x0046	Output 2 Autotune Tu	A	-10,000 to 10,000
0x0048	Output 2 Autotune Band	A	0 to 720
0x004A	Output 2 Autotune Temperature	A	-70 to 650°C
0x004C	Output 2 Autotune Step	A	0 to 1000 (in tenths of a percent)
0x004E	Output 2 Minimum Duty Cycle	B	0 to 1000 (in tenths of a percent)
0x0050	Output 2 Maximum Duty Cycle	B	0 to 1000 (in tenths of a percent)
0x0052	Output 2 PID Loop Time	B	40 to 10,000 (in ms)
0x0054	Temperature Scale (used only by Windows application)	K	See Table 4 - Value Format Key
0x0056	Modbus Address	M	1-247
0x0058	ID String (20 characters, 10 Modbus addresses)		ASCII

Table 2 - Modbus Holding Register Listing

b. Input registers:

Register Address	Description	Value Format
0x0000	Reserved	
0x0002	Firmware Version	N
0x0003	Hardware Version	N
0x0004	Number of NV Memory Writes	L
0x0006	Reserved	
0x0008	Reserved	
0x000A	Input 1 Temperature	A
0x000C	Input 1 Autotuning Status	J
0x000E	Input 2 Temperature	A
0x0010	Input 2 Autotuning Status	J
0x0012	Output 1 Duty Cycle	B
0x0014	Output 2 Duty Cycle	B

Table 3 - Modbus Input Register Listing

c. Value Format Key:

<table>
<tr><th>Value Format</th><th>Definition</th></tr>
<tr><td>A</td><td>32-bit IEEE 754 floating point value:

<table>
<tr><td>S</td><td>e = Exponent</td><td>m = Mantissa</td></tr>
<tr><td>1 bit</td><td>8 bits</td><td>23 bits</td></tr>
</table>

Value = $(-1)^S \times (1.m) \times 2^{e-127}$</td></tr>
<tr><td>B</td><td>Signed 32-bit integer, LSB first.

PID loop time is in units of milliseconds.
Minimum and Maximum Duty Cycle are in units of tenths of a percent.
ie. 500 = 50%</td></tr>
<tr><td>C</td><td>This register holds the Sensor Type setting for the respective input. The LSB is first.

<table>
<tr><th>Value</th><th>Type</th></tr>
<tr><td>0x0000</td><td>Off</td></tr>
<tr><td>0x0001</td><td>Platinum RTD 100Ω</td></tr>
<tr><td>0x0002</td><td>Platinum RTD 1000Ω</td></tr>
</table></td></tr>
<tr><td>D</td><td>This register holds the Output Source setting for the respective output. The LSB is first.

<table>
<tr><th>Value</th><th>Source</th></tr>
<tr><td>0x0000</td><td>Input 1</td></tr>
<tr><td>0x0001</td><td>Input 2</td></tr>
</table></td></tr>
<tr><td>E</td><td>This register holds the Output Control Type setting for the respective output. The LSB is first.

<table>
<tr><th>Value</th><th>Control Type</th></tr>
<tr><td>0x0000</td><td>Off (Disabled)</td></tr>
<tr><td>0x0001</td><td>PID</td></tr>
<tr><td>0x0002</td><td>On/Off</td></tr>
<tr><td>0x0003</td><td>Alarm</td></tr>
</table></td></tr>
</table>

MINCO
A critical component of your success™

<table>
<tr><td>F</td><td>This register holds the Reverse Acting setting for the respective output. The LSB is first.

<table><tr><th>Value</th><th>Reverse Acting</th></tr><tr><td>0x0000</td><td>Disabled</td></tr><tr><td>0x0001</td><td>Enabled</td></tr></table></td></tr>
<tr><td>G</td><td>This register holds the Alarm Over/Under setting when the respective output is set to the Alarm output type. The LSB is first.

<table><tr><th>Value</th><th>Alarm Over/Under</th></tr><tr><td>0x0000</td><td>Under</td></tr><tr><td>0x0001</td><td>Over</td></tr></table></td></tr>
<tr><td>H</td><td>This register interacts with the controller to begin an Autotune cycle. By setting this register to 0x0001, this triggers the controller to begin an Autotune cycle with the parameters current in the settings. When the Autotune cycle is complete, the CT435 will write 0x0000 to this register. The LSB is first.

<table><tr><th>Value</th><th>Start Autotune Cycle</th></tr><tr><td>0x0000</td><td>False</td></tr><tr><td>0x0001</td><td>True</td></tr></table></td></tr>
<tr><td>I</td><td>This register holds the Autotune Type setting. This register is ONLY used by the software interface when using the Evaulation Board to store what PID Method was last used. This should not be over-written, but will not cause performance issues if it is modified. The LSB is first.

<table><tr><th>Value</th><th>Autotune Type</th></tr><tr><td>0x0000</td><td>Manual Method</td></tr><tr><td>0x0001</td><td>P</td></tr><tr><td>0x0002</td><td>PI</td></tr><tr><td>0x0003</td><td>PD</td></tr><tr><td>0x0004</td><td>Classis PID</td></tr><tr><td>0x0005</td><td>Pessen Integral Rule</td></tr><tr><td>0x0006</td><td>Medium Overshoot</td></tr><tr><td>0x0007</td><td>Minimum Overshoot</td></tr></table></td></tr>
<tr><td>J</td><td>This register holds the status of the Autotuning cycle that is currently in process. This is applicable for each input channel. The LSB is first.

<table><tr><th>Value</th><th>Autotune Status</th></tr><tr><td>0x0000</td><td>Not Autotuning</td></tr><tr><td>0x0001</td><td>Ramp-up</td></tr><tr><td>0x0002</td><td>First cycle</td></tr><tr><td>0x0003</td><td>Second cycle</td></tr></table></td></tr>
<tr><td>K</td><td>This register holds the Temperature Scale setting for the software interface. The LSB is first.

<table><tr><th>Value</th><th>Reverse Acting</th></tr><tr><td>0x0000</td><td>Celsius</td></tr><tr><td>0x0001</td><td>Fahrenheit</td></tr></table></td></tr>
<tr><td>L</td><td>This register holds the current number of times the settings have been written to the on-board flash memory. Even though this block of memory is wear-leveled, it is advised to only write to the non-volatile memory when there is a change.
Unsigned 32-bit integer, LSB first.</td></tr>
<tr><td>M</td><td>This register holds the Modbus Address of the CT435. It is an unsigned 32 bit integer used to identify a given unit during Modbus communications. Valid addresses are 1-247.</td></tr>
<tr><td>N</td><td>This register holds the current firmware/hardware version.
Unsigned 16-bit integer, LSB first.</td></tr>
</table>

Table 4 - Value Format Key

6. Product Identification:

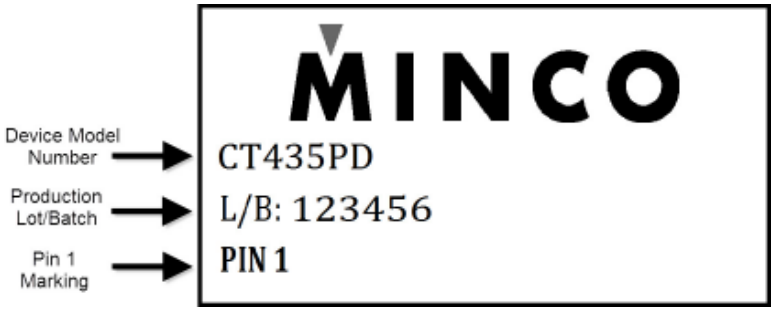


Figure 5 - Product Marking

7. Revision History:

Revision	Date	Change Description
Initial Release	6/14	N/A
A	3/17	Updated Modbus register list and communication settings

Table 5 - Revision History

8. How to Order:

CT435	Model Number: CT435
PD	Sensor Types: PD = 100 Ω Platinum RTD PF = 1000 Ω Platinum RTD
CT435PD ← Sample Part Number	

MINCO
A critical component of your success™

Notes: