Data Collection, Temperature Sensors, and Calibration

Objective

- Design a circuit which outputs 0 .. 5V for a sensor (instrumentation amplifier)
- Determine a calibration function so that a PIC can determine temperature from the A/D reading

SPI Communications

Your PIC board communicates to the PC using SPI communications (serial peripherial interface). This for of communication uses a single wire to transmit data to the PC (TX) and a single wire to receive data (RX).



SPI communication with a PC (and a USB serial port)

This is how programs are download using a USB serial port. You can also use the serial port to send data back to the PC for data collection.

The way the data is encode is as follows:

- The data line idles high.
- When you want to send a byte, the data line goes low for one count.
- Eight data bits are then sent, least significant bit first.
- The data line then goes back to 5V until the next byte is to be sent.



Initializing the SPI Port: With SPI communication, timing is critical: the computer which sends the data and the computer which receives the data must know the time duration (T) of each bit. If these don't match, the data will look like gibberist.

The baud rate defines how fast data is sent out:

• 9600 baud means you send 9600 bits per second (or one bit is 1/9600 seconds long)

Other baud rates are supported as well:

Baud Rate	SPBRG	BRGH	BRG16	SYNC	Error (%)
2,400	255	0	1	0	-1.70%
4,800	129	0	1	0	-0.16%
9,600	255	1	1	0	-1.70%
19,200	129	1	1	0	-0.16%
38,400	64	1	1	0	-0.16%
57,600	42	1	1	0	-0.95%
115,200	21	1	1	0	+1.44%

Assuming you want to set up the serial port for 9600 baud, the following code will initialize the serial port:

```
// Turn on the serial port for 9600 baud
```

```
TRISC = TRISC | 0xC0;

TXIE = 0;

RCIE = 0;

BRGH = 1;

BRG16 = 1;

SYNC = 0;

SPBRG = 255;

TXSTA = 0x22;

RCSTA = 0x90;
```

Once you initialize the serial port, you can send data to the PC (in ascii) using the following commands:

```
void SCI_CRLF(void)
{
    while(!TRMT); // wait until the serial port is free
    TXREG = 13; // send a carriage return (ascii 13)
    while(!TRMT); // wait until the serial port is free
    TXREG = 10; // send a line feed (ascii 10)
    }
```

}

To send data to the serial port, the following subroutine can be used (similar to LCD_Write)

```
void SCI_Out(long int DATA, unsigned char D, unsigned char N)
{
   unsigned char A[10], i;
  while(!TRMT);
   if (DATA < 0) {
      TXREG = '-';
      DATA = -DATA;
      }
   else TXREG = ' ';
   for (i=0; i<10; i++) {
      A[i] = DATA % 10;
      DATA = DATA / 10;
      }
   for (i=D; i>0; i--) {
      if (i == N) { while(!TRMT); TXREG = '.'; }
      while(!TRMT); TXREG = A[i-1] + 48;
      }
```

SPI Routines (part of LCD_PortD.c)

SCI_Out(long int DATA, unsigned char D, unsigned char N)
 same as LCD_Out only data is sent to the serial port instead
SCI_CRLF(void)
 send a carriage return, line feed

Example: Read the A/D port every 100ms and send its reading to the serial port at 9600 baud:

```
// Subroutine Declarations
#include <pic18.h>
// Subroutines
#include
                 "lcd_portd.c"
unsigned int A2D_Read(unsigned char c)
   unsigned int result;
   unsigned char i;
   c = c \& 0x0F;
   ADCON0 = (c << 2) + 0x01;
   for (i=0; i<20; i++);
   GODONE = 1;
   while (GODONE);
   return(ADRES);
   }
// Main Routine
void main(void)
{
   int A2D;
                 // raw A/D readings
                 // Volts*100 (500 means 5.00V)
   int VOLT;
   unsigned int i, j;
// Turn on the serial port for 9600 baud
   TRISC = TRISC | 0xC0;
   TXIE = 0;
   RCIE = 0;
   BRGH = 1;
   BRG16 = 1;
   SYNC = 0;
   SPBRG = 255;
   TXSTA = 0x22;
   RCSTA = 0 \times 90;
// Turn on the A/D
   TRISA = 0 \times FF;
   TRISE = 0 \times 0F;
   ADCON2 = 0 \times 85;
   ADCON1 = 0 \times 07;
   ADCON0 = 0 \times 01;
   i = 0;
   while(1) {
      A2D = A2D_Read(0);
      VOLT = 0.488 * A2D;
      SCI_Out(VOLT, 4, 3);
      SCI_CRLF();
                               // send carriage return line feed
      Wait_ms(100);
      }
   }
```

This results in the following data appearing on the PC's terminal:

0.49 0.99 1.47 1.94 2.39 2.82 3.22 3.58 3.91 4.20 etc

This is the voltage applied to RA0 every 100ms. You can then copy this data in to Matlab for later analysis.

Example 1: The voltage of four AA batteries being discharged across a 5.6 Ohm resistor is recorded for 6 hours at a sampling rate of one sample every 6 seconds. The data looks like...

1.51	1.50	1.54	1.49
1.51	1.50	1.54	1.50
1.51	1.50	1.54	1.50
1.51	1.50	1.54	1.49
1.50	1.49	1.53	1.49
1.50	1.49	1.53	1.49
1.50	1.49	1.53	1.49
etc.			

Plotting the data



Example 2: A pot of water was placed on a stove. A thermistor was placed in the water to measure the temperature. The voltage across the thermistor was then recorded every 6.0 seconds.

The data looks like the following:

- 1.63 1.63 1.63 1.63 1.63
- 1.63
- 1.63
- etc

Plotting the data



By changing the sensor, you can collect data on

- Voltage
- Resistance
- Light
- Temperature
- Humidity
- Alcohol,
- etc.

Sensors

Sensors convert something you want to measure to something you can measure. For example, some sensors available in lab are:

- An ultrasonic range sensor which converts distance to a voltage where the pulse-width is proportional to distance.
- A light sensor which converts light level to resistance as

$$R \approx \frac{100,000}{Lux} \ \Omega$$

• A temperature sensor which converts temperature to resistance as

$$R \approx 1000 \cdot \exp\left(\frac{3533}{T} - 11.85\right) \,\Omega$$

Usually, but not always, the output of a sensor is a resistance. This is because

- It takes no energy to produce a resistance. (such sensors do not need to be powered)
- If you find a material which changes resistance with changing light / temperature / humidity / etc, you have a sensor.

What this means is that if you can measure a resistance, you can measure whatever the sensor detects.

One way to convert a resistance (which your microcentroller cannot measure) into a voltage (which your microcentroller can measure) is to use a voltage divider:



To convert a resistance to a voltage, a voltage divider can be used. Note that the sensor and microcontroller need a common ground

If the resulting variation in voltage (and A/D reading) is large enough, you can input this voltage directly to the A/D input of the PIC. If the range is small, however, it would help to amplify the voltage. An instrumentation amplifier is a circuit which does this.

Instrumentation Amplifiers:

The basic circuit used to convert a signal to 0 .. 5V is an instrumentation amplifier:



Instrumentation Amplifier:
$$Y = \left(\frac{R_1}{R_2}\right)(V_a - V_b)$$

Example: Assume R is a RTD (temperature sensitive resistor) with the following relationship:

 $R = 1000(1 + 0.004T) \Omega$

Design a circuit which outputs

- 0V at 0C and
- 5V at +50C

Solution: First, determine the resistance at the two endpoints:

0C: R = 1000

50C: R = 1200

Step 2: Convert resistance to voltage. Assume a voltage divider with a 1k resistor. Then

$$V_x = \left(\frac{R}{R+1000}\right) 5V$$

0C: Vx = 2.5V

50C:
$$Vx = 2.7273V$$

Step 3: Determine whether Vx connects to the + input (Va) or the - input (Vb)

The output, Y, is to increase as Vx increases. Connect the + input (Va)

Step 4: Determine the gain.

$$Gain = \frac{\text{change in output}}{\text{change in input}}$$
$$Gain = \left(\frac{5V-0V}{2.7273V-2.5V}\right) = 22$$

Pick R1/R2 = 22. Let

•
$$R1 = 10k$$

•
$$R2 = 220k$$

Step 5: Determine the offset.

The output should be 0V when the input is 2.5V (0C)

$$V_y = gain(V_a - V_b)$$
$$0V = 22(2.5V - V_b)$$
$$V_b = 2.5V$$

The resulting circuit is as follows:



Instrumentation Amplifier: Output is 0V at 0C and 5V at +50C

The output voltage vs. temperature is then:

```
-->T = [0:0.1:50]';
-->R = 1000 * (1 + 0.004*T);
-->Vx = R ./ (1000+R) * 5;
-->min(Vx)
```

```
2.5
-->max(Vx)
2.7272727
-->gain = (5 - 0) / (max(Vx) - min(Vx))
22.
-->Vb = min(Vx)
2.5
-->Vy = gain*(Vx - 2.5);
-->plot(T,Vy)
-->xlabel('Temperature (C)');
-->ylabel('Voltage (Vy)');
```



Output of the Instrumentation Amplifier

Calibration

Assume the output of the instrumentation amplifier feeds an A/D input, such as RA1. Calibration takes the raw A/D reading (0 .. 1023) and converts this back to what you're trying to measure (temperature in this case). Essentially, calibration is curve fitting.

Assume you want to curve fit the above data with a straight line:

 $T \approx a \cdot A/D + b$

where A/D is the 0..1023 raw A/D reading. Some types of calibration are:

- Endpoint Calibration: Pass a line through the endpoints
- Lease Squares: Come up with a line that minimizes the mean squared error

To find the least squares solution, rewrite the curve fit in matrix form:

$$Y_{501x1} = X_{501x2}A_{2x1}$$
$$T = \begin{bmatrix} A/D & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

In the previous Matlab code, there are 501 data points, making X a 501x2 matrix. You cannot invert a 501x2 matrix. To solve, multiply both sides by X transpose:

$$X^T Y = X^T X A$$

XTX is a 2x2 matrix, and can usually be inverted. A is then

$$A = \left(X^T X\right)^{-1} X^T Y$$

In Matlab:

```
-->A2D = Vy*1023/5;
-->A2D = round(A2D);
-->Y = T;
-->X = [A2D, A2D.^0];
-->A = inv(X'*X)*X'*Y
0.0488903
- 0.8000303
```

 $T \approx 0.0488903 \cdot A/D - 0.80003$

Plotting this in Matlab:

-->plot(A2D, T, A2D, X*A)
-->xlabel('Raw A/D reading');
-->ylabel('Temperature (C)');

11



Actual A/D vs Temperature Relationship (blue) and Linear Curve Fit (blue)

```
To determine the calibration error, plot the difference (the residual)
```



Calibration Error for a Linear Curve Fit for Temperature

The estimated temperature based upon the previous calibration function is off by 0.8 degrees.



Note that the residual looks like a parabola. You can reduce the error by using a parabolic curve fit. To do this, change the basis:

 $T \approx a \cdot A/D^2 + b \cdot A/D + c$ -->X = [A2D.^2, A2D, A2D.^0]; -->A = inv(X'*X)*X'*Y Warning : matrix is close to singular or badly scaled. rcond = 0.0000D+00 A = a = 0.0000046 b = 0.0441556 c = 0.0258837

The resulting residual error is then within 0.04C

```
-->max(T - X*A)
ans =
0.0424377
-->min(T - X*A)
ans =
- 0.0351971
```



Calibration Error for a Linear Curve Fit (blue) and Parabolic Curve Fit (red)

At this point, the quantization noise (the steps above) are dominating the error. There is no point in adding more terms.

More Sensors:

Several sites contain toys for electrical and computer engineers. My favorites are

- www.Digikey.com Electronics parts with a very good search engine and fast delivery ٠
- www.Jameco.com Also a good supplier ٠
- www.AdaFruit.com Good spot for microcontrollers ٠
- www.SparkFun.com Good spot for GPS, sensors, micrcontrollers, and custom interface boards ٠

If you go to Digikey and search "Sensors", you get over 19,000 hits: (results on 12/23/15)

Electronic Components and P ×									
← → C 🗋 www.digikey.com/product-search/en?keywords=sensors									
🔛 Apps 🛐 Yahoo 🦳 Home 🦳 Davies 🦳 NDSU 🦳 Sonja									
All Products •			Q						
_(_)Keu `)									
PRODUCTS	MANOFACTORERS	RESOURCES V							
Keywords: 😗 sensors									
In stock									
Lead free Date: Compliant									
Compliant									
Search Again									
Sensore Transducers - 1017 New Products									
Accelerometers (1135 items)									
Accessories (3898 items)									
Amplifiers (313 items) Capacitive Touch Sensors, Proximity Sensor ICs (606 items)									
Color Sensors (135 items)									
Current Transducers (1503 items)									
Encoders (4364 items)									
Flex Sensors (1 items)									
Float, Level Sensors (622 items) Flow Sensors (191 items)									
Force Sensors (104 items)									
Gas Sensors (78 items)									
Image Sensors, Camera (555 items)									
Inclinometers (55 items)									
LVDT Transcucers (Linear Variable Differential Transformer) (8 items)									
Magnetic Sensors - Compass, Magnetic Field (Modules) (24 items)									
Magnetic Sensors - Hall Effect, Digital Switch, Linear, Compass (ICs) (3747 items) Magnetic Sensors - Position Proximity Speed (Modules) (3288 items)									
Magnets (145 items)									
Moisture Sensors, Humidity (376 items)									
Multifunction (147 items)									
Optical Sensors - Ambient Light, IR, UV Sensors (736 items)									
Optical Sensors - Distance Measuring (41 items) Optical Sensors - Mouse (118 items)									
Optical Sensors - Photo Detectors - CdS Cells (59 items)									
Optical Sensors - Photo Detectors - Logic Output (134 items)									
Optical Sensors - Photodiodes (1062 items)									
Optical Sensors - Photoelectric, Industrial (11149 items)									
Optical Sensors - Photointerrupters - Slot Type - Logic Output (1063 items) Optical Sensors - Photointerrupters - Slot Type - Transistor Output (1171 items)									
Optical Sensors - Phototransistors (808 items)									
Optical Sensors - Reflective - Analog Output (332 items) Optical Sensors - Reflective - Logic Output (134 items)									
Position Sensors - Angle, Linear Position Measuring (1290 items)									
Pressure Sensors, Transducers (26790 items)									
Proximity Sensors (5/02 items) Proximity/Occupancy Sensors - Finished Units (240 items)									
RTD (Resistance Temperature Detector) (87 items)									
Shock Sensors (15 items) Solar Cells (103 items)									
Specialized Sensors (610 items)									

Strain Gauges (22 items) Temperature Regulators (Mechanical) (3947 items) Temperature Sensors, Transducers (3457 items) Temperature Switches (917 items) Thermistors - NTC (5293 items) Thermistors - PTC (1251 items) Thermocouple, Temperature Probe (431 items) Tilk Sensors (55 items) Ultrasonic Receivers, Transmitters (96 items) Vibration Sensors (58 items)

Most of these have a resistance output. What this means is that if you can measure resistance, you can measure acceleration, color, dust, flex, gas, incline, magnetic fields, motion, light, pressure, etc.

For example, suppose you want to find a temperature sensor:

Niai-Keu	All Products	v				Q		United S	States 1 Engli	-800-344-4 sh v U	539 SD
LECTRONICS	PRODUCTS	MANUFACTURER	S RESOURCI	ES 🔻 🖣	LIVE CHAT			🃜 0 item(s) ▼ L F	ogin or REGISTER	•
Keywords: 3 sensors In stock Lead free RoHS Compliant Search Again Product Index > Sensors, Transduce Results matching criteria: 5,293	rs > <u>Thermistors - N</u>	<u>10</u>						f	in G•		
To select multiple values within a box, h Manufacturer	old down 'Ctrl' while se Packa	electing values within th	he box. Series	Resistance in Ohms @ 25°C	Resistance Tolerance	B Value Tolerance	B0/50	B25/50	B25/75	B25/85	B2
Abracon LLC Ametherm Amphenol Advanced Sensors AVX Corporation Cantherm Crouzet Automation Curtis Instruments Inc. EPCOS (TDK) Honeywell Sensing and Control EMEA Honeywell Sensing and Productivity Sc	* Bulk Cut Tape (Digi-Reel® Tape & Re Tape & Re Tray	CT) 111 (CT) 112 × (TB) 115 tel (TR) 118 120 121 123		1 2.2 2.5 3 3.3 4.7 5 6 6 8	±0.01°C ±0.05°C ±0.1°C ±0.2% ±0.23°C ±0.25°C ±0.2°C ±0.2°C ±0.3% ±0.5%	+0.4% ±0.5% ±0.7% ±0.75% ±0.8% ±1.3% ±1.3% ±1.5% ±1.5%	2854K 2941K 3000K 3260K 3271K 3320K 3419K 3420K 3442K	* 1950K 2150K 2750K 2800K 2934K 29350K 3000K 3060K	- 3181K 3254K 3477K 3500K 3691K 3700K 3890K 3890K 3984K 4084K	- 2680K 2700K 2750K 2758K 2758K 2772K 2800K 2850K 2873K 2880K •	- 260 275 280 290 300 305 305 306 310

www.Digikey.com search for NTC Thermistor

What this tells you is you have 5,293 thermistors to choose from. To whittle this down a bit, you can narrow the search using

- Resistance @ 25C: Pretty much what is says. Low values are good for measuring wind speed (low R produces more self heating. The wind provides cooling. Temperature is thus a measurement of the cooling or wind speed). High values are good for low self-heating and low-power consumption.
- Resistance Tolerance: Smaller is better. The variation in the resistance at 25C (manufacturing tolerance.) Sort of a measure of the standard deviation only most people don't know what standard deviation is.
- B Value Tolerance: Smaller is better. How accurate you know the temperature / resistance relationship, The lower the number, the more precise the measurement (and the more it costs.)
- B0/50, B25/50, etc. The temperature-resistance relationship parameter (more on this later)
- Operating Temperature (off the page to the right): The range the sensor can operate
- Mounting Type (off the page to the right): Through hole (good for us) or surface mount (good for industry).

Let's narrow the search to

- 1000 Ohms at 25C
- Through Hole
- In Stock

This narrows the selection down to 13 thermistors. (note: through-hole is mostly for students. High-volume applications, like cell phones, use surface mount parts - which is where the bulk of electronics parts go.)

Compare Parts	R	Image	Digi-Key Part Number	Manufacturer Part Number	Manufacturer	Description	Quantity Available ?	Unit Price ? USD	Minimum Quantity ?	Packaging	Series	Resistance in Ohms @ 25°C	Resistance Tolerance
			▲ ▼	▲ ▼	▲ ▼	▲ ▼	▲ ▼	▲ ▼	▲ ▼		▲ ▼	▲ ▼	A V
	R	R	BC2519-ND	NTCLE100E3102JB0	Vishay BC Components	THERMISTOR NTC 1.0K 5% RADIAL	8,535 - Immediate	0.39000	1	Bulk 😮	<u>2381</u>	1k	±5%
	R	/	BC2394-ND	NTCLE100E3102HB0	Vishay BC Components	THERMISTOR NTC 1.0K 3% RADIAL	2,954 - Immediate	0.45000	1	Bulk 😨	-	1k	±3%
	2	/	BC2393-ND	NTCLE100E3102GB0	Vishay BC Components	THERMISTOR NTC 1.0K 2% RADIAL	1,428 - Immediate	0.67000	1	Bulk 😨	-	1k	±2%
	2	I all	KC016N-ND	<u>RL2004-582-97-D1</u>	Amphenol Advanced Sensors	THERMISTOR NTC 1K OHM @ 25C	1,233 - Immediate	2.07000	1	Bulk 😨	<u>RL2004</u>	1k	±10%
	2	X	480-3157-ND	192-102DEW-A01	Honeywell Sensing and Productivity Solutions	THERMISTOR NTC 1KOHM RADIAL	483 - Immediate	7.29000	1	Bulk 😨	<u>192</u>	1k	±1%
	2		495-2156-ND	<u>B57891M102J</u>	EPCOS (TDK)	THERMISTOR NTC 1.0K OHM 5% RAD	2,067 - Immediate	0.95000	1	Bulk 😨	-	1k	±5%

Selecting the bottom one on the list (fairly low price and its what we have in stock) tells you the price in quantities of 1 to 10,000

				All prices	are in US dollars.	
D	igi-Key Part Number	495-2156-ND	Price Break	Unit Price	Extended Price	
Quantity Available		Digi-Key Stock: 2,067 Can shin immediately	1	0.95000	0.95	
Manufacturer		EPCOS (TDK)	100	0.72200	51.68	
Manufacturer Part Number		B57891M102J	500	0.40014	200.07	
	Description	THERMISTOR NTC 1.0K OHM 5% RAD	1,000	0.34457	344.57	
Lead Free Status / RoHS Status		Lead free / RoHS Compliant	5,000	0.30566	1,528.31	
Moisture Sensitivity Level (MSL)		1 (Unlimited)	10,000	0.29455	2,945.48	
Quantity	ltem Number 👩	Customer Reference				
1	495-2156-ND ·	Add to Cart				
						Image shown is a represen Exact specifications should from the product data

Clicking on the data sheet brings up the specifications for the thermistor:

General technical data

Climatic category	(IEC 60068-1)		40/125/56	
Max. power	(at 25 °C)	P ₂₅	200	mW
Resistance tolerance		$\Delta R_{\rm B}/R_{\rm B}$	±5, ±10	%
Rated temperature		T _R	25	°C
Dissipation factor	(in air)	δ_{th}	approx. 3.5	mW/K
Thermal cooling time constant	(in air)	τ_{c}	approx. 12	s
Heat capacity		Cth	approx. 40	mJ/K

Electrical specification and ordering codes

R ₂₅	No. of R/T	B _{25/100}	Ordering code
Ω	characteristic	К	
1 k	1009	3930 ±3%	B57891M0102+000
1.5 k	1008	3560 ±3%	B57891M0152+000
2.2 k	1013	3900 ±3%	B57891M0222+000
3.3 k	2003	3980 ±3%	B57891M0332+000
4.7 k	2003	3980 ±3%	B57891M0472+000
6.8 k	2003	3980 ±3%	B57891M0682+000
10 k	4901	3950 ±3%	B57891M0103+000
15 k	2004	4100 ±3%	B57891M0153+000
22 k	2904	4300 ±3%	B57891M0223+000

This tells you

- Limit the self-heating to 200mW ($I^2R < 200mW$). For a 1k thermistor, limit the voltage to less than 14.14V across the thermistor at 25C
- Dissipation factor (in air): 3.5 mW/K. At equilibrium, power out = power in. Power in is self heating (I^2R). Power out is from cooling (3.5 mW/K).
- Thermal cooling time constant (in air): 12 seconds. It takes some time for the thermistor to warm up to air temperature. The thermistor will be within 5% of equilibrium in 3 time constants, or 36 seconds.
- B25/100: 3930K. This is the temperature resistance relationship where T is the temperature in degrees Kelvin (Celsius + 273)

$$R \approx 1000 \cdot \exp\left(\frac{3930}{T} - \frac{3930}{298K}\right)$$

Modeling in Matlab

Suppose this thermistor is used with a 1k resistor in a voltage divider. Determine the following relationships from 0C to 50C

- Resistance vs. temperature
- Voltage vs. temperature
- A/D reading vs. temperature

```
-->T = [0:0.1:50]';
-->K = T + 273;
-->R = 1000 * exp((3930 ./ K) - (3930 / 298) );
-->plot(T,R);
-->xlabel('Temperature (C)');
-->ylabel('Resistance (Ohms)');
```



Temperature vs. Resistance Relationship. Note the resistance is 1000 Ohms at 25C.

-->V = R ./ (1000 + R) * 5; -->plot(T,V); -->xlabel('Temperature (C)'); -->ylabel('Voltage (V)');



Temperature vs. Voltage Relationship.

-->A2D = V/5 * 4095; -->plot(A2D,T) -->xlabel('A/D Reading');



A/D Reading (what you know) vs. Temperature (what you want to measure)

Once you have the A/D reading vs. temperature (or light, pressure, gas, etc), you can use least squares to come up with a calibration function like we did before.

Self Heating:

As a sidelight, with thermistors you need to worry about self heating. By powering the thermistor, it warms up as

$$P = I^2 R = \frac{V^2}{R}$$

If exposed to air, the thermistor is able to dissipate heat as 3.5mW/K. This means, at equilibrium, the thermistor will be slightly warmer than the air as

$$dT \approx \left(\frac{V^2/R}{0.0035W/K}\right)$$

-->P = V.^2 ./ R; -->dT = P / 0.0035; -->plot(T,dT); -->xlabel('Temperature (C)'); -->ylabel('Self Heating Error (C)');



Self-Heating from 0..50C. This self-heating results in the measured temperature being off

Assuming you know you are measuring air temperature with no wind, you could compensate for this bias in your calibration function.

If you use a thermistor with a larger resistance, the self-heating will be smaller. If you use a thermistor with a smaller resistance, the self heating will be larger.

If there is wind, the heat dissipation will increase. This allows to measure wind speed by comparing the measured temperature of a low R (high self-heating) and high R thermistor (low self-heating) circuit.