PT-100

Shristy Sharma

Contents

***Abstract*—This document contains a lab report on the modeling of the voltage-temperature characteristics of the PT-100 RTD (Resistance Temperature Detector) using least squares method.**

1 Training Data

Training data obtained by PT-100 to train the Arduino is shown in Table 1.

Temperature (°C)	Voltage (V)
35	2.63
55	2.66
49	2.65
85	2.71
75	2.70
65	2.68
27	2.61

TABLE 1: Training data.

The effective schematic circuit diagram is shown in Figure 1.

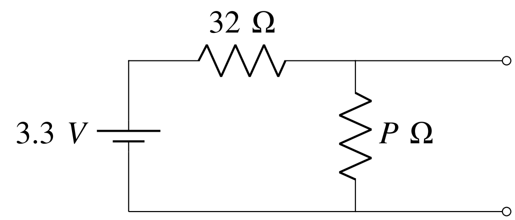

Fig. 1: Schematic Circuit Diagram to Measure the Output of PT-100 (P).

2 Model

For the PT-100, we use the Callendar-Van Dusen equation

$$V(T) = V(0)\left(1 + AT + BT^2\right) \tag{1}$$

$$\implies c = \mathbf{n}^\top \mathbf{x} \tag{2}$$

where

$$c = V(T),\ \mathbf{n} = V(0)\begin{pmatrix}1\\A\\B\end{pmatrix},\ \mathbf{x} = \begin{pmatrix}1\\T\\T^2\end{pmatrix} \tag{3}$$

For multiple points, (2) becomes

$$\mathbf{X}^\top \mathbf{n} = \mathbf{C} \tag{4}$$

where

$$\mathbf{X} = \begin{pmatrix}1 & 1 & \dots & 1\\ T_1 & T_2 & \dots & T_n\\ T_1^2 & T_2^2 & \dots & T_n^2\end{pmatrix} \tag{5}$$

$$\mathbf{C} = \begin{pmatrix}V(T_1)\\ V(T_2)\\ \vdots\\ V(T_n)\end{pmatrix} \tag{6}$$

and $\mathbf{n}$ is the unknown.

3 Solution

We approximate $\mathbf{n}$ by using the least sqaures method. The Python code `codes/pt100.py` solves for $\mathbf{n}$.

The calculated value of $\mathbf{n}$ is

$$\mathbf{n} = \begin{pmatrix}2.5577569\\ 2.0663864 \times 10^{-3}\\ -2.9546268 \times 10^{-6}\end{pmatrix} \tag{7}$$

The approximation is shown in Fig. 2.

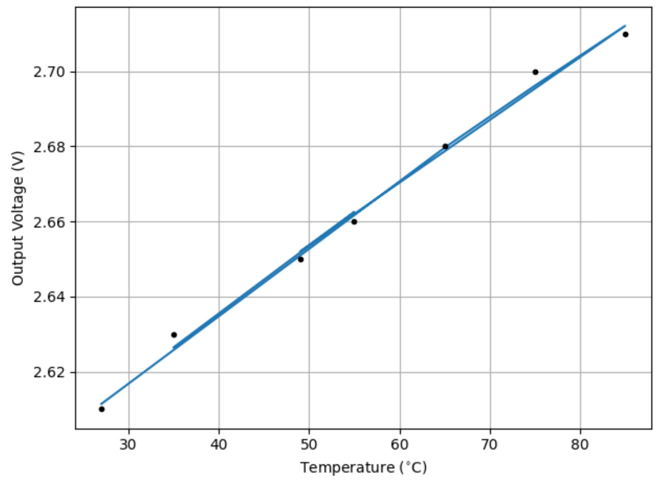

Fig. 2: Training the model.

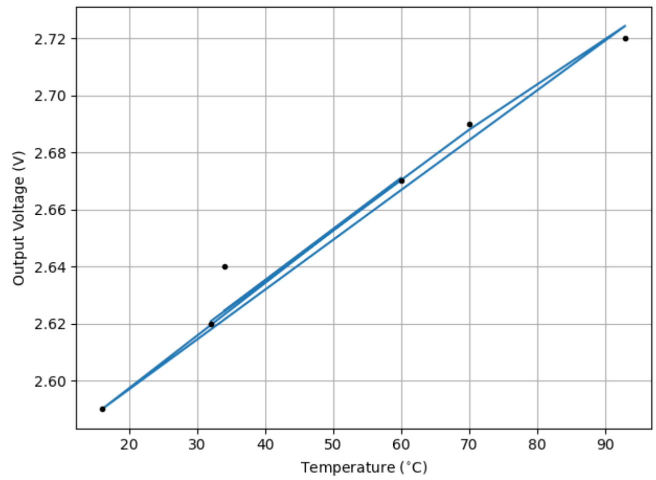

Fig. 3: Validating the model.

Thus, the approximate model is given by

$$V(T) = 2.5577569 + \left(2.0663864 \times 10^{-3}\right) T - \left(2.9546268 \times 10^{-6}\right) T^2 \quad (8)$$

Equation 8 can be written in the form of,

$$ax^2 + bx + c = 0 \quad (9)$$

$$\implies 2.9546268 \times 10^{-6} T^2 + 2.0663864 \times 10^{-3} T - (2.5577569 - V(T)) = 0 \quad (10)$$

Now, we can use the quadratic formula to find the value of the temperature.(which has been done in Arduino)

4 Validation

The validation dataset is shown in Table 2. The results of the validation are shown in Fig. 3.

Temperature (°C)	Voltage (V)
34	2.64
60	2.67
16	2.59
93	2.72
70	2.69
32	2.62

TABLE 2: Validation data.

5 Conclusion

The interfacing has been done between Arduino and LCD where, the temperature has been displayed. The modelling of the sensor has been done using Python and has been executed using a microcontroller. This project demonstrates how machine learning methods can be used to model the behaviour of an unknown component, and find the right parameters that fit the model.