*International Journal of Electronics and Electrical Engineering Vol. 8, No. 1, March 2020*

# Integrating Math, Physics and Microcontroller Technology into a Temperature Control System

Jie Sheng
School of Engineering and Technology, University of Washington, Tacoma, USA
Email: shengj2@uw.edu

***Abstract*—Control engineering is an interdisciplinary field requiring knowledges on math, physics, circuits, sensors, actuator, and microcontrollers (for digital control implementation). It also provides users experiences in testing, simulation and real-time implementation. This paper presents a course demonstration project illustrating the essence of control engineering. The objective is to design and apply a PID controller to maintain the temperature of a power resistor by using Arduino Mega 2560. After device modelling and testing by applying math and physics knowledges, the open-loop system dynamics can be determined and controllers can be designed. By comparing the simulation and experimental closed-loop system performances, the difference caused by plant-model mismatch is discussed. The importance of integrating math, physics, and circuits with microcontroller technology in embedded control systems is also highlighted.**

***Index Terms*—control engineering, microcontroller, temperature control**

## I. INTRODUCTION

Control engineering is an interdisciplinary field requiring knowledges on math, physics, circuits, sensors, actuators, programming and microcontrollers (for digital control implementation). It also provides users experiences in device testing, design simulation, and real-time implementation. This paper presents a course demonstration project illustrating the essence of control engineering. The objective is to design and apply a PID controller to maintain the temperature of a power resistor (KAL50 [1]) by using Arduino Mega 2560 [2]. A power resistor is a resistor designed and manufactured to dissipate large amounts of power in a compact physical package [3]. In order to control its temperature, a thermistor is attached as the temperature sensor, and its power dissipation is controlled by the Pulse Width Modulation (PWM) signal from Arduino together with an MOSFET transistor and certain external circuits.

To fulfil the temperature control purpose, students are required to apply their knowledges on mathematics and physics to first understand the readings of thermistor, i.e., determine the temperature based on the measured resistance. They are then expected to study dynamics of the open-loop system, i.e., the power resistor. After testing and simulation, students apply PID design methods to close the loop so that the power resistor can be controlled to stay at certain temperature.

We remark here that microcontrollers like Arduino, have become a mainstay of control systems laboratories. An open-source electronics platform based on low-cost hardware and easy-to-use software, Arduino has been quite popular in making interactive projects to meet both individual interests and academic educational requirements. Enhancing the teaching and learning of control systems by using Arduino-based projects is not new in engineering discipline, for example, [4] summarizes how this effort was taken in the college of engineering at Qassim University Saudi Arabia. [5] uses Arduino UNO [6] to collect and process the temperature sensor signal, and generate the fan speed control accordingly.

In addition to Arduino, a variety of other microcontrollers have also been effectively used in control engineering education. Examples include [7] and [8] where PID methods were implemented for temperature control using Atmega32 [9] and Atmega16 [10], respectively; and [11] where a visualization education tool was designed for a PID temperature control system using PIC18F4550 [12], just to name a few.

In the remaining of this paper, we will first describe in Section II the procedure of how to read the thermistor and how to derive the model of a power resistor, followed by PID controller design and simulation results in Section III. Section IV presents detailed implementation and experimental results, and Section V concludes the paper.

## II. DEVICE MODELING

KAL50 - the power resistor to be controlled is shown in Fig. 1, where a thermistor is attached to measure the temperature. In the following, we will first describe how to convert the direct measurement (resistance in the unit of ohm) to be the temperature associated with the power resistor. We will then try several tests to find the dynamics of the power resistor KAL50. A picture illustrating how the circuits are built to connect devices will also be provided, to ensure successful signals reading and testing.

---

Manuscript received September 24, 2019; revised February 6, 2020.

doi: 10.18178/ijeee.8.1.6-10

Figure 1. A power resistor with thermistor attached

## A. *Reading the Thermistor*

The Steinhart–Hart equation [13] is a model of the resistance of a semiconductor at different temperatures. It is often used to derive a precise temperature of a thermistor, and has the format as shown in Equation (1)

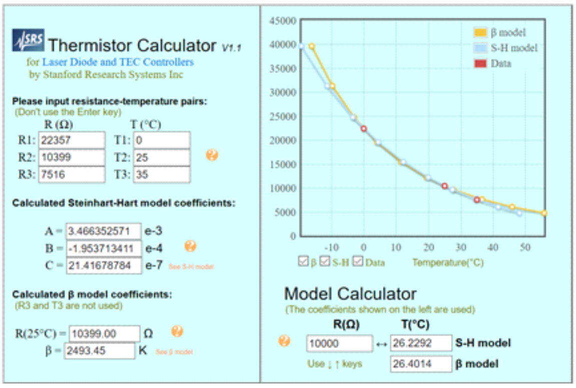
$$\frac{1}{T} = A + B \times \ln R + C \times (\ln R)^3 \qquad (1)$$

where $T$ is the temperature (in kelvins), $R$ is the resistance (in ohms) at temperature $T$; $A$, $B$, and $C$ are the Steinhart–Hart coefficients, which vary depending on the type and model of thermistor and the temperature range of interests. The most general form of the applied equation contains a term $(\ln R)^2$, but this is frequently neglected because it is typically much smaller than the other coefficients, and is therefore not shown above.

To get the coefficients $A$, $B$, and $C$, tests need to be taken for collecting data; three sets of data are used and summarized below in Table I. We then apply the online thermistor calculator [14] to decide $A$, $B$ and $C$.

TABLE I. DATA COLLECTED TO DETERMINE SH MODEL COEFFICIENTS

| Resistance (Ω) | Temperature (℃) |
|---|---|
| 22357 | 0 |
| 10399 | 25 |
| 7516 | 35 |



Thermistor Calculator v1.1
for Laser Diode and TEC Controllers
by Stanford Research Systems Inc

Please input resistance-temperature pairs:
(Don't use the Enter key)

| | R (Ω) | | T (°C) |
|---|---|---|---|
| R1: | 22357 | T1: | 0 |
| R2: | 10399 | T2: | 25 |
| R3: | 7516 | T3: | 35 |

Calculated Steinhart-Hart model coefficients:

A = 3.466352571 e-3
B = -1.953713411 e-4
C = 21.41678784 e-7

Calculated β model coefficients:
(R3 and T3 are not used)

R(25°C) = 10399.00 Ω
β = 2493.45 K

Model Calculator
(The coefficients shown on the left are used)

| R(Ω) | | T(°C) | |
|---|---|---|---|
| 10000 | ↔ | 26.2292 | S-H model |
| | | 26.4014 | β model |

Figure 2. Online thermistor calculator to find coefficients A, B, C

Fig. 2 shows the calculated values, i.e., $A$ = 3.47e-3, $B$ = -1.95e-4, and $C$ = 21.4e-7. With these coefficients, the temperature can be calculated by the following equation

$$T = \frac{1}{A + B \times \ln R + C \times (\ln R)^3} \qquad (2)$$

## B. *Dynamics of the Power Resistor*

A power resistor is designed and manufactured to dissipate large amounts of power. To study its dynamics, we have tried several tests and plotted the data for analysis purpose.

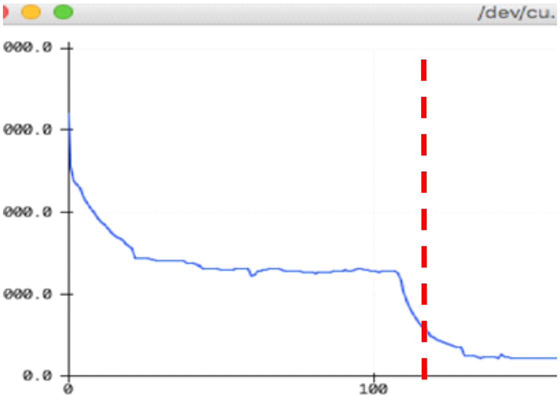


Figure 3. Temperature changes from 0 to 25 then 25 to 85 (in Celsius) when applying a 12 Volts input to KAL50

Fig. 3 shows the thermistor's resistance change along with time. Accordingly it reflects the temperature change from 0 degrees Celsius to 85 degrees Celsius. More specifically, the first part of the graph (to the left of the vertical red dashed line) represents the thermistor's resistance change matching its temperature change from 0 degrees Celsius to 25 degrees Celsius. The second part (to the right of the vertical red dashed line) represents the thermistor's reaction when a 12V input was applied to the KAL 50 Chassis Mount Resistor.

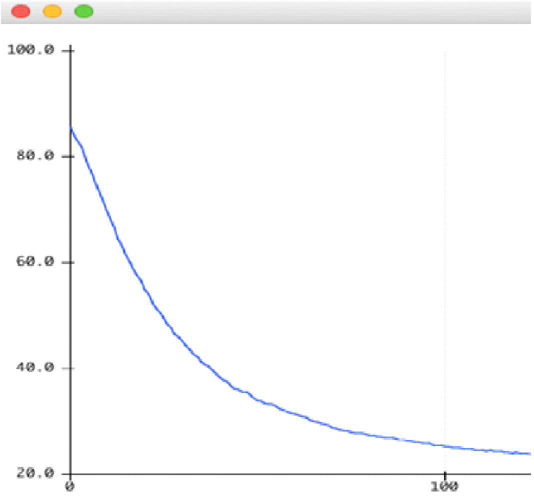
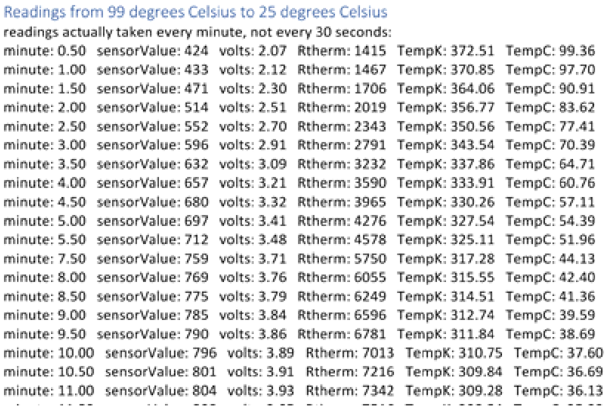


Figure 4. Power resistor cool down from 85 Celsius to room temperature 24 Celsius

We then let the power resistor cool down to the room temperature 24 degrees Celsius; the change of temperature versus time is shown in Fig. 4 where the time horizon shows the index of time sequence at a sampling interval of 4.5 seconds.

Both Fig. 3 and Fig. 4 have shown the dynamics of a typical first-order system. To determine the time constant, which is the reciprocal of the decaying rate, of the power resistor, we can either do an approximation (≈ 5 minutes) by visual inspection of the graph in Fig. 4, or by an analysis of the monitored output (≈ 11 minutes) as shown in Fig. 5. By averaging these results, we finally decide the time constant of the power resistor is $\tau = 8\,minutes$.



```
Readings from 99 degrees Celsius to 25 degrees Celsius
readings actually taken every minute, not every 30 seconds:
minute: 0.50   sensorValue: 424   volts: 2.07   Rtherm: 1415   TempK: 372.51   TempC: 99.36
minute: 1.00   sensorValue: 433   volts: 2.12   Rtherm: 1467   TempK: 370.85   TempC: 97.70
minute: 1.50   sensorValue: 471   volts: 2.30   Rtherm: 1706   TempK: 364.06   TempC: 90.91
minute: 2.00   sensorValue: 514   volts: 2.51   Rtherm: 2019   TempK: 356.77   TempC: 83.62
minute: 2.50   sensorValue: 552   volts: 2.70   Rtherm: 2343   TempK: 350.56   TempC: 77.41
minute: 3.00   sensorValue: 596   volts: 2.91   Rtherm: 2791   TempK: 343.54   TempC: 70.39
minute: 3.50   sensorValue: 632   volts: 3.09   Rtherm: 3232   TempK: 337.86   TempC: 64.71
minute: 4.00   sensorValue: 657   volts: 3.21   Rtherm: 3590   TempK: 333.91   TempC: 60.76
minute: 4.50   sensorValue: 680   volts: 3.32   Rtherm: 3965   TempK: 330.26   TempC: 57.11
minute: 5.00   sensorValue: 697   volts: 3.41   Rtherm: 4276   TempK: 327.54   TempC: 54.39
minute: 5.50   sensorValue: 712   volts: 3.48   Rtherm: 4578   TempK: 325.11   TempC: 51.96
minute: 7.50   sensorValue: 759   volts: 3.71   Rtherm: 5750   TempK: 317.28   TempC: 44.13
minute: 8.00   sensorValue: 769   volts: 3.76   Rtherm: 6055   TempK: 315.55   TempC: 42.40
minute: 8.50   sensorValue: 775   volts: 3.79   Rtherm: 6249   TempK: 314.51   TempC: 41.36
minute: 9.00   sensorValue: 785   volts: 3.84   Rtherm: 6596   TempK: 312.74   TempC: 39.59
minute: 9.50   sensorValue: 790   volts: 3.86   Rtherm: 6781   TempK: 311.84   TempC: 38.69
minute: 10.00   sensorValue: 796   volts: 3.89   Rtherm: 7013   TempK: 310.75   TempC: 37.60
minute: 10.50   sensorValue: 801   volts: 3.91   Rtherm: 7216   TempK: 309.84   TempC: 36.69
minute: 11.00   sensorValue: 804   volts: 3.93   Rtherm: 7342   TempK: 309.28   TempC: 36.13
```

Figure 5. Monitored output when the power resistor cool down from 99 Celsius to 25 Celsius

### C. *Circuits for Device Modeling and Testing*

To read the 10KOhm thermistor, we have used a voltage divider as shown in the right half of Fig. 6, where a reference resistor with the same resistance (10KOhm) is connected in series with the thermistor.

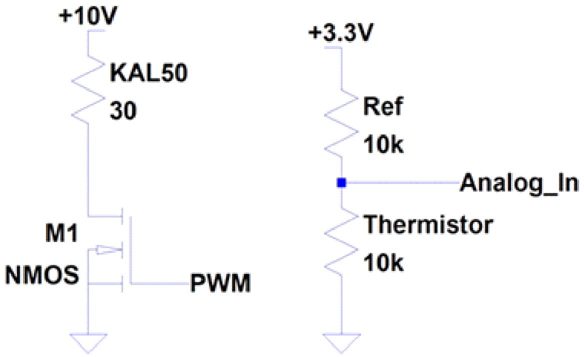


Figure 6. Thermistor reading and power resistor testing circuits

To determine the dynamics of KAL50, we have provided an external 10 volts power supply to heat the power resistor, as shown in the left half of Fig. 6.

Arduino reads in the thermistor data (as shown in the right half of Fig. 6), converts it into relevant temperature, and runs the P/PI/PID controller algorithm. Arduino also outputs the PWM cycle to control the power of KAL50. For a larger control signal, an MOSFET transistor [15] is used as shown in the left half of Fig. 6.

## III. CONTROLLER DESIGN AND SIMULATION RESULTS

With the modelling and testing results presented in Section II including the time constant $\tau = 8\,minutes$, the temperature control system can be illustrated by Fig. 7.

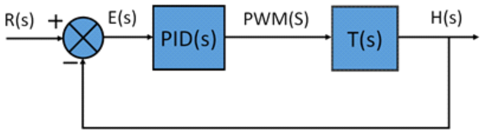


Figure 7. Temperature control system block diagram

PID controllers' design, mainly, the tuning of parameters $K_P$, $K_I$, and $K_D$ is handled by using MATLAB/Simulink [16]. Based on the results given by Auto-tuning of PID, after some trial and error, we adopted a *P* controller with $K_P = 80$; the step response of the simulated closed-loop system is shown in Fig. 8. When a *PI* controller was applied with $K_P = 80$, $K_I = 0.2$ the step response of the simulated system is shown in Fig. 9. Not surprisingly, the steady-state error in Fig. 8 was eliminated, and the response was speeded up.

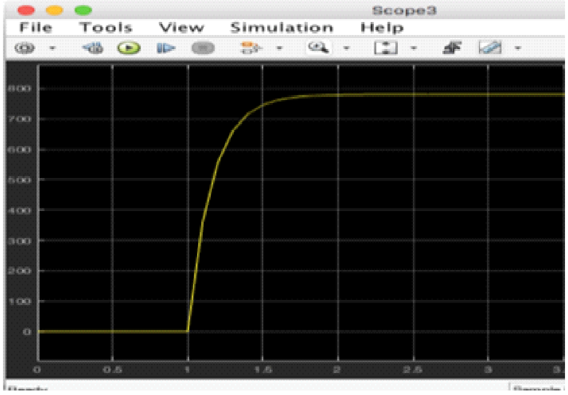


Figure 8. Step response of the simulated closed-loop system (P controller)

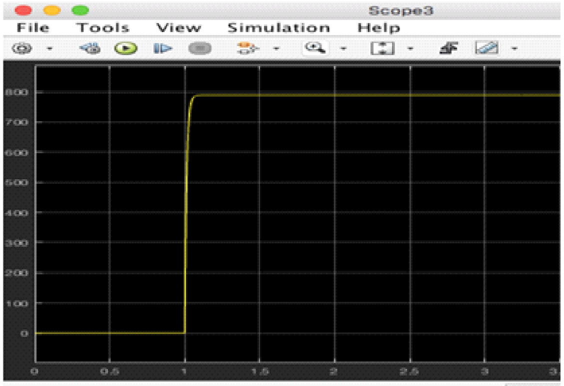


Figure 9. Step response of the simulated closed-loop system (PI controller)

Compared to PID design shown in [7] and [8], our PID parameters were chosen in a relatively intuitive way. After all, the goal of this project is to demonstrate how math, physics and microcontroller technology are combined together to solve a real control engineering problem.

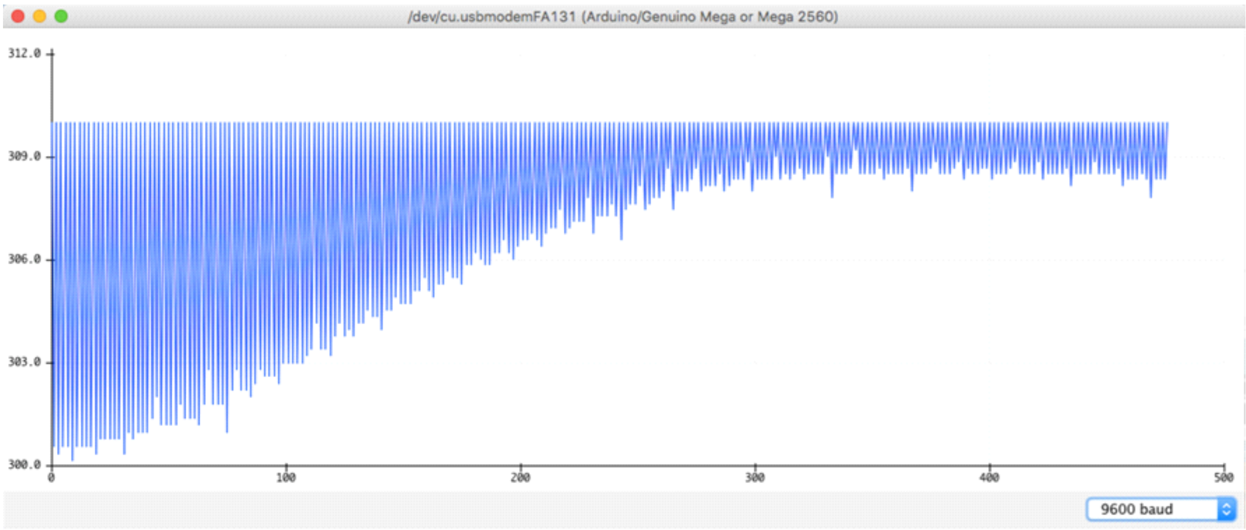


Figure 10. Step response of the real-time closed-loop system (P controller)

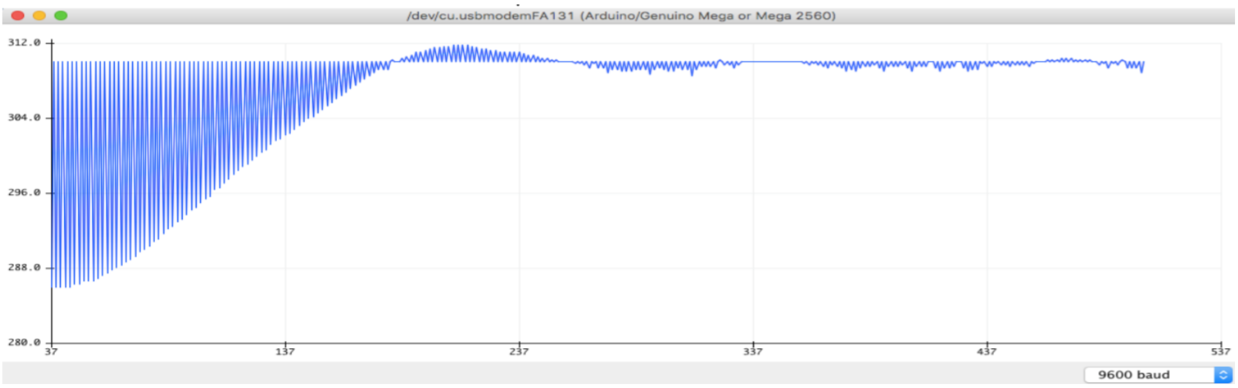


Figure 11. Step response of the real-time closed-loop system (PI controller)

## IV. IMPLEMENTATION AND EXPERIMENTAL RESULTS

As expected, when the controller designed in Section III was implemented and applied to the real system, the system performance would be different due to plant-model errors. With the same $P$ controller, i.e., $K_P = 80$, step response of the real-time closed-loop system is shown in Fig. 10 where the blue area represents the error between the desired temperature and actual temperature. $PI$ controller with $K_P = 80$, and $K_I = 0.2$ was then used to remove the steady-state error; the real-time closed-loop step response in Fig. 11 shows the effectiveness of PI controller.

## V. CONCLUSIONS

A power resistor is a small dynamic system designed to dissipate large amounts of power. When voltages being applied across it, its temperature will change and show typical first-order system characteristics. It is thus a good illustration example to teach students concepts about modelling (using their math and physics knowledges), testing, and simulation. Arduino is a good hardware platform to collect data, process data, compute the controller algorithm and output PWM signal to drive external devices. The real-time implementation of a control algorithm via microcontroller teaches students existence of the plant-model mismatch, and thus motivates them to consider more advanced methods to improve control performance.

This paper summarizes a temperature control project in teaching devices and control to undergraduate students. We remark that the design and development of control theories is not the focus of this paper; instead, we want to emphasize that control engineering is an interdisciplinary field.

How to deliver the key concepts in an effective way, bring hands-on experience to labs, and meanwhile expose students to the up-to-date technologies becomes very challenging nowadays for educators. This project is designed to highlight the importance of two fundamentals in teaching control systems: (1) math and physics – they are crucial in helping students understand how a device works and transforms one type of signal to another type of signal; (2) microcontroller technology as well as circuits – they make it feasible to measure and test signals by connecting different devices and components together. As a conclusion, this paper presents a temperature control project from an educator’s perspective. With the rapid progress of information and computer technologies, it is our goal to keep building interesting projects which not only integrate modern technologies (including hardware/software tools) into teaching of classical control theories, but also highlight the importance of math/physics fundamentals in solving real control engineering problems.

## REFERENCES

[1] KAL Series. [Online]. Available: https://www.seielect.com/Catalog/SEI-kal.pdf

[2] A. Kurniawan, *Arduino Mega 2560 - A Hands-on Guide for Beginner*, Kindle edition, PE Press, 2019.

[3] Power resistor. [Online]. Available: http://www.resistorguide.com/power-resistor

[4] H. M. Omar, "Enhancing automatic control learning through Arduino-based projects," *European Journal of Engineering Education*, pp. 1-12, 2017.

[5] P. Roy, J. Saha, N. Dutta, and S. Chandra, "Microcontroller based automated room light and fan controller," in *Proc. Emerging Trends in Electronic Devices and Computational Techniques*, 2018, pp. 1-4.

[6] W. A. Smith, *Arduino Uno Hardware Manual: A Reference and User Guide for the Arduino Uno Hardware and Firmware*, CreateSpace Independent Publishing Platform, 2019.

[7] M. Elnour and W. I. M. Taha, "PID and fuzzy logic in temperature control system," in *Proc. International Conference on Computing, Electrical and Electronic Engineering*, 2013, pp. 172-177.

[8] S. Ranjan, A. Sharma, and P. Chaudhary, "An effective temperature controller system using PID mechanism," in *Proc. Innovative Applications of Computational Intelligence on Power, Energy and Controls with Their Impact on Humanity*, 2014, pp. 182-185.

[9] H. Bernstein, *Mikrocontroller Grundlagen der Hard- und Software der Mikrocontroller ATtiny2313, ATtiny26 und ATmega32*, Springer Fachmedien Wiesbaden, 2015.

[10] W. Budiharto, *Panduan Praktikum Mikrokontroler AVR ATmega16*, Elex Media Komputindo, 2013.

[11] A. K. R. Segundo, J. A. N. Cocota, and D. V. M. Ferreira, "Development of an educational tool for control engineering," in *Proc. IEEE Global Engineering Education Conference*, 2015, pp. 594-601.

[12] M. Rafiquzzaman, *Microcontroller Theory and Applications with the PIC18F*, Wiley & Sons, 2011.

[13] Steinhart–Hart equation. [Online]. Available: https://en.wikipedia.org/wiki/Steinhart–Hart_equation

[14] Thermistor Calculator. [Online]. Available: https://www.thinksrs.com/downloads/programs/therm%20calc/ntccalibrator/ntccalculator.html

[15] N-channel enhancement mode vertical DMOS FET. [Online]. Available: https://www.diodes.com/assets/Datasheets/ZVN4206A.pdf

[16] A. K. Tyagi, *MATLAB and SIMULINK for Engineers*, Oxford Higher Education, 2012.

**Jie Sheng** holds a PhD degree in Electrical Engineering, and completed her dissertation entitled generalized predictive control for multi-rate systems, at the University of Alberta, Canada in 2002.

She is currently an Associate Professor at the School of Engineering and Technology, University of Washington, Tacoma (UWT). Before that, she had been an Assistant Professor, a Lecturer, and an NSERC Postdoctoral Fellow, at DigiPen Institute of Technology, the University of New South Wales, and the University of Illinois in Urbana-Champaign, respectively. At UWT, in addition to teaching core courses at the Computer Engineering and Systems program, she has also supervised a variety of exciting research projects involving Unmanned Vehicle Systems, Vision-based IOT Devices, and Drone-based Wireless Sensor Networks. She has published over 55 papers in refereed journals and proceedings. Her current research interests include control systems, system identification, parameter estimation, and engineering education.

Dr. Sheng has been the reviewer for journals including TIMC, ISA Transactions, Circuits, Systems, and Signal Processing, Mechanical Systems and Signal Processing, and the program committee member for a number of international conferences and workshops. She is a member of IEEE, IEEE WIE, IES and ASEE. She has been the faculty supervisor of IEEE student branch at UWT since 2018.