

# Design of a Two-Point Temperature Monitoring System on a Heating Medium Based on Arduino Mega 2560 and IoT

Nataly Wisnu Anggara<sup>\*1)</sup>, Mario Gracio Anduinta Rhizma<sup>1)</sup>

<sup>1)</sup>Electrical Engineering Study Program, Universitas Pelita Harapan, Tangerang, Indonesia

## ABSTRACT

This research aims to design a two-point temperature monitoring system using 2 DS18B20 sensors on a heating medium using DS18B20 sensors controlled by an Arduino Mega 2560 and integrated with Internet of Things (IoT) technology. The measured data are processed by the microcontroller and transmitted via the ESP8266 module to the Firebase platform, then displayed on a smartphone application and recorded as historical data. The experimental results show that the DS18B20 sensors are able to read temperature changes linearly within the tested temperature range of approximately 27°C to 67°C and can follow the temperature increase trend of the heating medium. To improve measurement accuracy, calibration was performed using a linear regression method with a surface thermocouple as the reference instrument. After calibration, the measurement error was significantly reduced to 2.90% at the upper measurement point and 2.66% at the lower measurement point, indicating that the system is more representative for monitoring temperature changes at two measurement locations. The proposed demonstrates decent performance as an IoT-based temperature monitoring prototype, making it suitable for small-scale heating applications and educational purposes.

## ARTICLE INFO

**Keywords:** Arduino Mega 2560, Firebase, IoT, DS18B20 thermocouple sensor, prototype

**\* Corresponding author:**

[mario.rhizma@uph.edu](mailto:mario.rhizma@uph.edu)

**Article history:**

Submitted 11 Feb 2026

Revised 2 May 2026

Accepted 7 May 2026

Online Available 7 May 2026

Published 20 May 2026



## 1. Introduction

Rapid technological advancement has driven industrial systems to shift from conventional manual operations toward automated and more efficient processes. In manufacturing industries, process stability, particularly temperature control, plays a critical role in maintaining product quality and reducing defects.

Research on temperature monitoring using microcontrollers [1-3] has been extensively conducted, employing devices such as the AT89C51 [4], Arduino Uno [5-6], and Arduino Mega [7-9]. This study focuses on the use of an Arduino Mega in a two-point temperature monitoring system. The dual-sensor temperature control method can be applied in various contexts [10]; in this prototype project, it is specifically would be implemented to monitor the heating medium temperature in the production process of automotive components, specifically in the Am-Tube curing section.

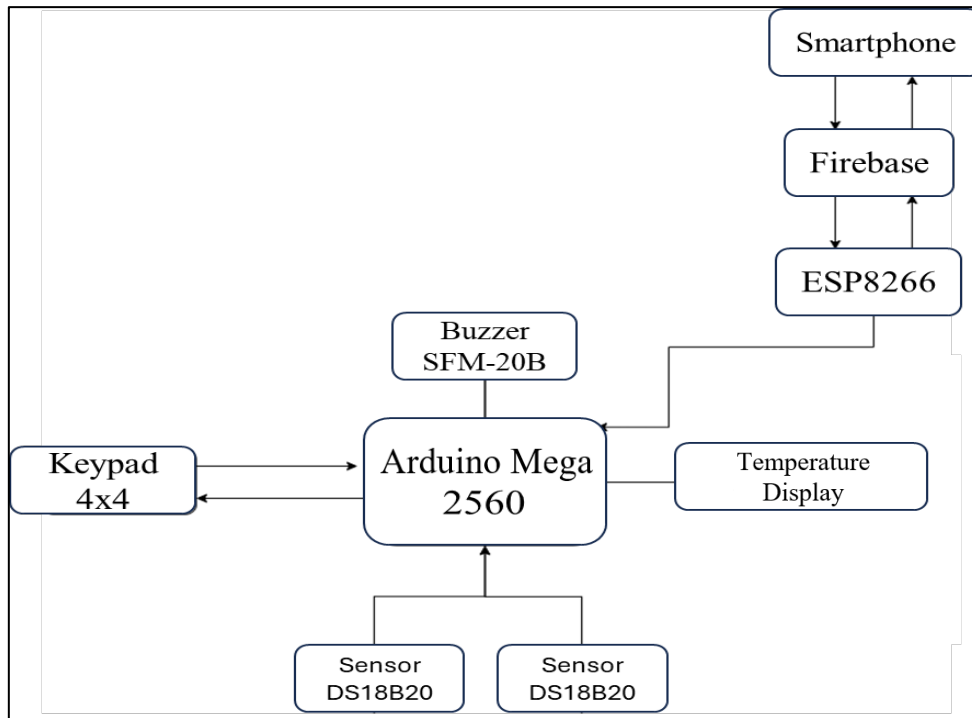
In the curing process of automotive components, temperature deviation can significantly affect mechanical properties and product reliability. Conventional monitoring methods often rely on indirect parameters or manual measuring, which are unable to represent real-time temperature conditions during machine operation. This limitation increases the risk of undetected process deviations. Therefore, this study proposes the development of a real time temperature monitoring and alarm system based on Internet of Things (IoT) technology.

The use of IoT technologies for monitoring temperatures in enclosed spaces and environmental conditions has been widely explored [11-12]. Likewise, the DS18B20 temperature sensor has been extensively utilized for temperature measurements in various applications [13-17]. The proposed system is expected to improve process visibility, reduce measurement errors, and support data-driven evaluation of temperature stability.

## 2. System Framework and Research Methodology

### 2.1 System Overview

The proposed system is designed as an IoT-based temperature monitoring platform consisting of sensing, processing, communication, and visualization layers as shown on **Figure 1**.



**Figure 1** Block diagram of IoT-based temperature monitoring system

Temperature data are acquired using digital sensors, processed by a microcontroller, transmitted through a wireless module, and displayed in real time via mobile application and cloud database. As shown on **Figure 2**, the keypad functions to provide commands to the LCD in accordance with predefined settings. In this study, the commands configured in the operating system are as follows:

1. Button A displays the actual temperature of the upper and lower sensors.
2. Button B displays the Top Mold parameters.
3. Button C displays the Bottom Mold parameters.
4. Button D displays the Alarm parameters (continuous flip-flop mode).
5. The “#” button functions to reset the alarm. After the keypad is pressed, a display appears prompting the user to enter the reset code, which is 0000.

The buzzer functions as an alarm when the temperature exceeds the predetermined threshold, while the actual measured temperature is displayed on 20x4 LCD Display, and also on user smartphone.



**Figure 2** 20x4 LCD Display with keypad for display commands

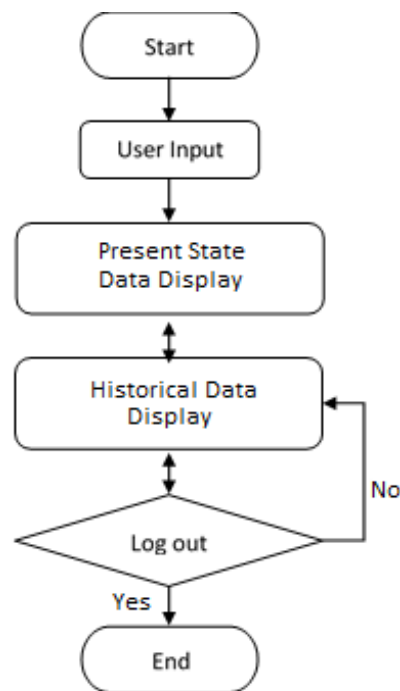
## 2.2 Hardware Configuration

**Table 1** Hardware components specification

No.	Components	Detail
1.	Computer	Windows 10, Intel Core i5, RAM 16GB, SSD 512 GB
2.	Arduino Module	ATmega 2560
3.	Thermocouple Module	MAX6675
4.	Wi-Fi Module	ESP8266
5.	Relay	
6.	Keypad	4x4
7.	Temperature Sensor	DS18B20
8.	Buzzer	SFM-20B
9.	LCD	20 x 4
10.	Smartphone	Android Operating System
11.	Android Software	Kodular

The hardware system consists of DS18B20 digital temperature sensors for temperature measurement, an Arduino Mega 2560 as the main processing unit, and an ESP8266 Wi-Fi module for data transmission. The sensors provide digital temperature readings, which are periodically processed by the microcontroller before being sent to the cloud database. The relays are used to control external loads such as system cooling fans or warning indicator lights.

## 2.3 Software Architecture



**Figure 3** Flowchart of temperature monitoring and alarm system

The flowchart of software architecture is shown on **Figure 3**. The software system includes firmware developed using Arduino IDE, a Firebase Realtime Database for cloud-based data storage, and a mobile application developed using Kodular as the user interface. Google Spreadsheets is utilized for historical data logging and further analysis. The application allows users to monitor temperature conditions in real time and review historical data remotely.

## 2.4 Temperature Measurement and Calibration Method

Temperature readings obtained from the DS18B20 sensors are initially treated as raw data. To improve measurement accuracy, a calibration process is applied using linear regression with a thermocouple

surface sensor as the reference instrument. The calibration equation is expressed as:

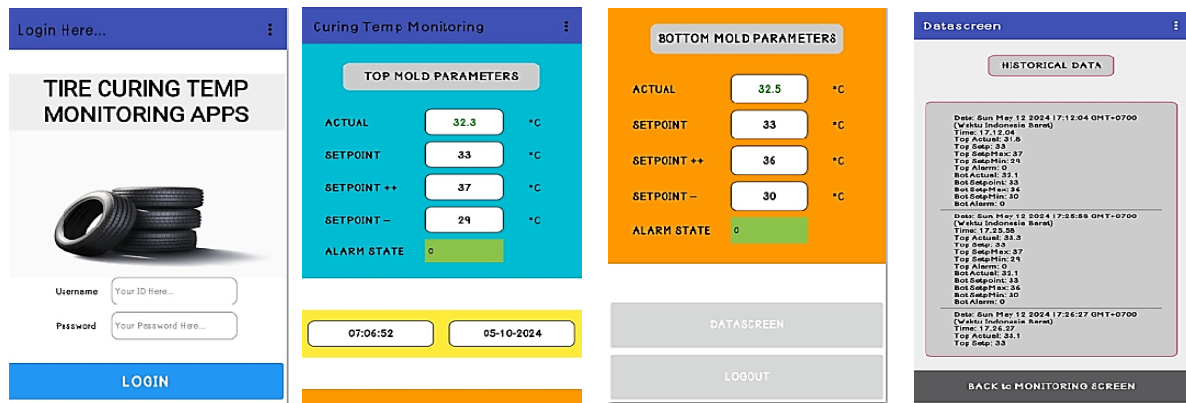
$$T = aX + b \quad (1)$$

where  $X$  represents the raw sensor reading, and  $a$  and  $b$  are regression coefficients obtained from experimental data. The calibrated temperature values are then used for monitoring, alarm triggering, and data logging.

### 3. Results and Discussion

#### 3.1 Real-Time Monitoring Performance

The developed system demonstrates the capability to display temperature data in real time through a mobile application interface, as shown on **Figure 4**. Data transmission from the microcontroller to the cloud database occurs continuously during system operation, enabling users to monitor temperature conditions remotely. This feature provides better visibility of temperature behavior compared to manual measurement methods, which only capture data at specific moments. The real-time monitoring capability allows early detection of abnormal temperature conditions, supporting faster decision-making and improving overall system reliability.



**Figure 4** Mobile application interface for real-time temperature monitoring

#### 3.2 Temperature Measurement Results Before Calibration

The initial measurement results indicate discrepancies between the temperature values measured by the DS18B20 sensors and the thermocouple surface reference. These differences are observed throughout the measurement duration and become more pronounced during periods of rapid temperature change.

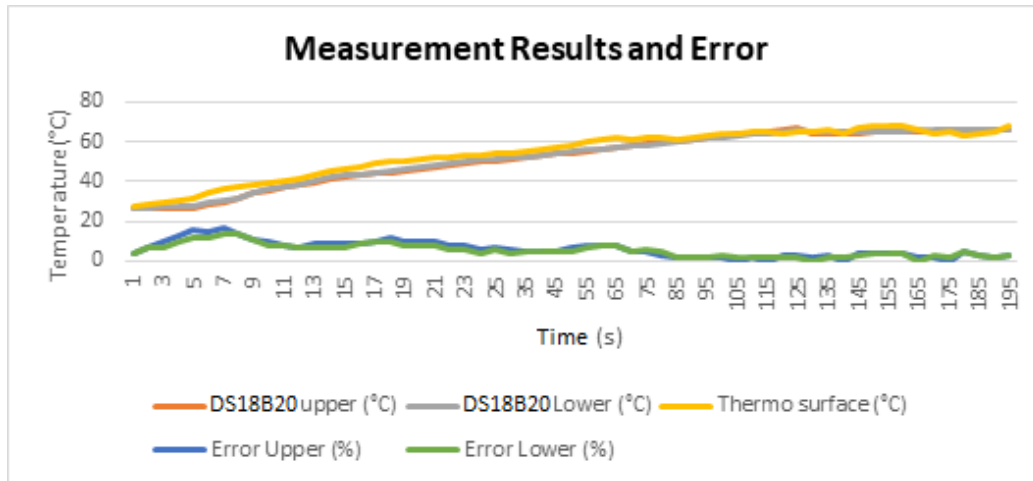
The deviation is mainly caused by differences in sensor response characteristics and measurement principles. The DS18B20 sensor exhibits a slower response compared to the thermocouple surface, resulting in lag during dynamic temperature variations. This initial measurement is shown on **Figure 4**.

#### 3.3 Error Analysis Before Calibration

In this experiment, the test medium for sensor accuracy testing consisted of an aluminium plate heated using a heat source. Two DS18B20 sensors were attached to the aluminium surface using aluminium foil to ensure adequate thermal contact, and were positioned adjacent to a surface thermocouple used as the reference measurement instrument. The two sensors were positioned approximately 5 cm apart, and the data were sampled at 1 second intervals for 200 datasets. The test was conducted to compare the readings of the DS18B20 sensors with the surface thermocouple, as well as to evaluate the system's accuracy before and after calibration.

Error analysis before calibration shows relatively high percentage errors for both upper and lower sensor readings. The highest error values occur at the beginning of the measurement process, where temperature changes rapidly. In this condition, small absolute differences between sensor readings and reference values can lead to large percentage errors. These results is displayed on **Figure 5**, confirm that

raw DS18B20 sensor data are not sufficiently accurate for applications requiring precise temperature measurement without calibration. The  $R^2$  coefficient for upper sensor is 0.964, and 0.968 for lower sensor. The two sensors show a strong linear relationship with time, but the average error of the upper sensor reached **6.05%**, while the lower sensor showed an average error of **5.35%**. This error must be improved for better accuracy.

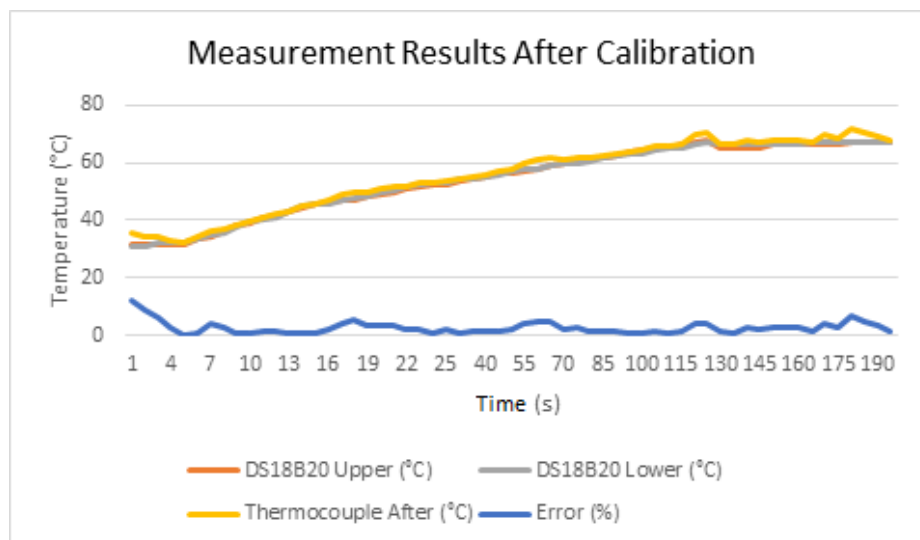


**Figure 5** Temperature measurement results before calibration

### 3.4 Results After Linear Regression Calibration

After applying linear regression calibration, the corrected temperature values show improved agreement with the thermocouple surface reference. The calibration process reduces systematic deviations by adjusting raw sensor readings using regression coefficients obtained from experimental data.

Based on the results, the average measurement error decreases significantly after calibration, as shown on **Figure 6**. This reduction indicates that the linear regression method effectively enhances the accuracy of the temperature monitoring system.



**Figure 6** Temperature and error results after linear regression calibration

The  $R^2$  coefficient improved for the two sensors, the  $R^2$  of the upper sensor is 0.982, and 0.984 for lower sensor. The average error of the upper sensor decreased to **2.90%**, while the lower sensor error was reduced to **2.66%**.

### 3.5 Discussion

The experimental results demonstrate that calibration is essential when using general-purpose digital

temperature sensors for monitoring applications that require higher accuracy. Although the DS18B20 sensor is not specifically designed for surface temperature measurement, linear regression calibration significantly improves its measurement performance.

However, minor discrepancies remain during rapid temperature transitions due to the inherent response characteristics of the sensor. For industrial-scale implementation requiring fast response and high accuracy, industrial-grade sensors such as thermocouples are recommended.

The subsequent temperature readings are transmitted via the ESP8266 module to a Firebase database and displayed in a monitoring application developed using Kodular. The data transmission process operates periodically at 1 second interval, enabling users to monitor temperature variations in real time through a smartphone.

#### 4. Conclusion

This study has successfully developed a prototype of an IoT-based temperature monitoring and alarm system capable of providing real-time temperature data through a mobile application. The integration of digital temperature sensors, a microcontroller platform, cloud database, and mobile interface enable continuous monitoring and remote access to temperature information. The experiment results indicate that the uncalibrated DS18B20 sensors exhibit relatively high measurement errors when compared to the thermocouple surface reference. Before calibration, the average error of the upper sensor reached **6.05%**, while the lower sensor showed an average error of **5.35%**. These results confirm that direct sensor readings are not sufficient for applications requiring higher accuracy.

After applying linear regression calibration, a significant improvement in measurement accuracy was achieved. The average error of the upper sensor decreased to **2.90%**, while the lower sensor error was reduced to **2.66%**. This reduction demonstrates that the proposed calibration method minimizes systematic measurement deviations and enhances the reliability of the monitoring system.

Although minor discrepancies remain during rapid temperature changes due to inherent sensor response characteristics, the developed system is suitable as a prototype IoT-based monitoring solution. Further improvements can be achieved by employing industrial-grade sensors and control devices for implementation in real industrial environments.

#### Acknowledgement

The author would like to express gratitude to Universitas Pelita Harapan for the academic support provided during this research. Appreciation is also extended to all parties who contributed to the completion of this study through technical assistance, guidance, and constructive feedback.

#### References

- [1] B. Li and J.H. Lei, "Design of Industrial Temperature Monitoring System Based on Single Chip Microcontroller," *2011 International Conference on Computer Science and Service System (CSSS)*, Nanjing, 2011, pp. 342-344, <https://doi.org/10.1109/CSSS.2011.5974595>.
- [2] S. Ponnusamy, R. Samikannu, B. A. Tlhabologo, W. Ullah and S. Murugesan, "Design and Development of Microcontroller-Based Temperature Monitoring and Control System for Power Plant Generators," *IOP Conference Series: Materials Science and Engineering*, Volume 1055, International Virtual Conference on Robotics, Automation, Intelligent Systems and Energy (IVC RAISE 2020) 15th December 2020, Erode, India <https://doi.org/10.1088/1757-899X/1055/1/012158>
- [3] D. K. Fisher, H. Kebede, "A Low-Cost Microcontroller-Based System to Monitor Crop Temperature and Water Status," *Computers and Electronics in Agriculture*, Volume 74, Issue 1, 2010, Pages 168-173, ISSN 0168-1699, <https://doi.org/10.1016/j.compag.2010.07.006>.
- [4] H. Zhu and L. Bai, "Temperature Monitoring System Based on AT89C51 Microcontroller," *2009 IEEE International Symposium on IT in Medicine & Education*, Jinan, China, 2009, pp. 316-320, <https://doi.org/10.1109/ITIME.2009.5236408>.
- [5] K.K. Khaing, K. S. Raju, G.R. Sinha, W.Y. Swe, "Automatic Temperature Control System Using Arduino," *Proceedings of the Third International Conference on Computational Intelligence and Informatics*, 2020, Advances in Intelligent Systems and Computing, vol 1090. Springer, Singapore. [https://doi.org/10.1007/978-981-15-1480-7\\_18](https://doi.org/10.1007/978-981-15-1480-7_18)

- [6] J. Hubbart, T. Link, C. Campbell, and D. Cobos, "Evaluation of a Low-Cost Temperature Measurement System for Environmental Applications," 2005, *Hydrological Processes* Vol. 19 Issue 7: 1517-1523. <https://doi.org/10.1002/hyp.5861>
- [7] A. Indriani, Hendra, and Y. Witanto, "Error of Assembly Microcontroller Arduino Mega and ATmega in the Control of Temperature for Heating and Cooling System," *AMM*, vol. 842, pp. 324-328, Jun. 2016, <https://doi.org/10.4028/www.scientific.net/amm.842.324>.
- [8] P. Megantoro, A. A. Bagus, H. F. A. Kusuma, S. A. Reina, M. Fadhilah, "Automation System of Temperature Calibrator for Room Thermostat Using Arduino Mega256," *Proceedings of the International Conference on Advanced Technology and Multidiscipline (ICATAM 2024)*, 2024. [https://doi.org/10.2991/978-94-6463-566-9\\_14](https://doi.org/10.2991/978-94-6463-566-9_14)
- [9] M. K. Fadzly, Yiling, M. F. Rosli, T. Amarul and M. S. M. Effendi, "Smart Air Quality Monitoring System Using Arduino Mega," *Materials Science and Engineering*, Volume 864, 2nd Joint Conference on Green Engineering Technology & Applied Computing 2020 4-5 February 2020, Bangkok, Thailand. <https://doi.org/10.1088/1757-899X/864/1/012215>
- [10] X. Gao, Q. Ma, X. Sun, J. Wang and S. Li, "Research on a Dual-Closed-Loop Temperature Control Method and System Based on Dual Sensors for Infrared Object Simulation," *IEEE Sensors Journal*, vol. 19, no. 23, pp. 11553-11561, 1 Dec.1, 2019, <https://doi.org/10.1109/JSEN.2019.2935555>.
- [11] M. A. Muslim, R. A. Setyawan, A. Basuki, A. A. Razak, F. P. Hario and E. Fernando, "IOT Based Climate Monitoring System," *Earth and Environmental Science, Volume 746, 3rd International Conference on Life and Applied Sciences for Sustainable Rural Development (ICLAS-SURE 2020)*, November 18-19, 2020 Central Java, Indonesia. <https://doi.org/10.1088/1755-1315/746/1/012044>
- [12] R. F. Maulana, M. A. Ramadhan, W. Maharani and M. I. Maulana, "Rancang Bangun Sistem Monitoring Suhu dan Kelembapan Berbasis IOT Studi Kasus Ruang Server IT Telkom Surabaya," *Indonesian Journal of Multidisciplinary on Social and Technology*, 2023, 1(3), 224-231. <https://doi.org/10.31004/ijmst.v1i3.169>
- [13] R. Saha, S. Biswas, S. Sarmah, S. Karmakar and P. Das, "A Working Prototype Using DS18B20 Temperature Sensor and Arduino for Health Monitoring," *SN Computer Science Vol.2*, article number 33, 2021. <https://doi.org/10.1007/s42979-020-00434-2>
- [14] Y. X. Wu, D. Liu, and X. H. Kuang, "A Temperature Detecting System Based on DS18B20," *Advanced Materials Research*, vol. 328-330, pp. 1806-1809, Sept. 2011, <https://doi.org/10.4028/www.scientific.net/amr.328-330.1806>.
- [15] R. A. Koestoer, Y. A. Saleh, I. Roihan, Harinaldi, "A Simple Method for Calibration of Temperature Sensor DS18B20 Waterproof in Oil Bath Based on Arduino Data Acquisition System," *AIP Conference Proceedings*, 25 January 2019; 2062 (1): 020006. <https://doi.org/10.1063/1.5086553>
- [16] S. Budijono and Felita, "Smart Temperature Monitoring System Using ESP32 and DS18B20," *Earth and Environmental Science, Volume 794, 4th International Conference on Eco Engineering Development*, 10-11 November 2020, Banten, Indonesia, <https://doi.org/10.1088/1755-1315/794/1/012125>
- [17] A. N. Fathoni, N. Hudallah, R. D. M. Putri, K. Khotimah, T. Rijanto and M. Ma'arif, "Design Automatic Dispenser for Blind People based on Arduino Mega using DS18B20 Temperature Sensor," *2020 Third International Conference on Vocational Education and Electrical Engineering (ICVEE)*, Surabaya, Indonesia, 2020, pp. 1-5, <https://doi.org/10.1109/ICVEE50212.2020.9243254>