OSTİM TEKNİK ÜNİVERSİTESİ
A N K A R A

OSTIM TECHNICAL UNIVERSITY
Engineering Faculty

Department of Electrical-Electronics Engineering

Graduation Project

200202012 Enes SEZGİN
210202013 Ülkü Ece KUŞÇU

Dr. Şenol GÜLGÖNÜL

May 2025
ANKARA

OSTIM TECHNICAL UNIVERSITY
Engineering Faculty

Department of Electrical-Electronics Engineering

Graduation Project

200202012 Enes SEZGİN
210202013 Ülkü Ece KUŞÇU

Dr. Şenol GÜLGÖNÜL

May 2025 ANKARA

LİSANS BİTİRME PROJESİ ONAY FORMU

Enes Sezgin ve Ülkü Ece Kuşçu tarafından Ostim Teknik Üniversitesi Elektrik Elektronik Mühendisliği yönetiminde hazırlanan “TEMPERATURE CONTROLLER WITH PID ON-OFF AND SLOW PWM FUNCTIONALITY” başlıklı lisans bitirme projesi tarafımızdan incelenmiş, kapsamı ve niteliği açısından bir Lisans Bitirme Projesi olarak kabul edilmiştir.

Danışman	:	Dr. Şenol Gülgönül
Jüri Üyesi 1	:	Prof. Dr. Volodymyr Yurchenko
Jüri Üyesi 2	:	Prof. Dr. Yalçın Ata
Bölüm Başkanı	:	İsmail Hakkı Altaş

PREFACE

I would like to express my sincere gratitude to everyone who supported me throughout the preparation of this graduation project report. First and foremost, I am deeply thankful to my esteemed advisor, Dr. Şenol GÜLGÖNÜL, for his invaluable guidance, patience, and contributions that significantly enhanced the academic quality of this work. I would also like to thank Research Assistant Hatice Kübra ERSARI for her continuous support and assistance during the project. My sincere appreciation goes to our Department Chair, Prof. İsmail Hakkı ALTAŞ Ph.D. for his leadership and contributions to the development of our department.

Additionally, I would like to thank the Department of Electrical and Electronics Engineering at OSTİM Technical University for providing all the facilities necessary for this study, as well as the Faculty of Engineering and the Rectorate of OSTİM Technical University for their continued support.

Finally, I would like to extend my heartfelt thanks to my family for their unwavering encouragement and support throughout my education.

May 2025

Enes SEZGİN
Ülkü Ece KUŞÇU

TABLE OF CONTENTS

ABSTRACT

In this project, an educational temperature control system is designed to teach PID and ON-OFF control algorithms, which are widely used in industrial automation systems. The system was developed to enable high school and university level students to learn control systems practically.

In the project, a PT100 sensor was used as temperature sensor and Arduino Uno board was used as microcontroller. Temperature data was read via MAX31865 RTD module and a 12V 55W headlamp bulb was used as the heater. A 16x2 LCD screen and buttons were used as the user interface to access different control modes of the system. The system supports three different control modes: PID Control, ON-OFF Control and SlowPWM Control.

The most suitable coefficients for the PID algorithm were determined by system identification and PID tuning in MATLAB Simulink environment. Using the temperature-time data, the transfer function of the system was extracted and PID coefficients suitable for this model were obtained. With the developed system, the heater control was realized according to the target temperature and control parameters received from the user, and the temperature data were monitored instantaneously on the LCD.

With this project, the basic principles of control systems are demonstrated in a simple and applicable structure and a practical educational tool is provided for students to better understand the concepts.

SYMBOLS AND ABBREVIATIONS

SYMBOLS

K: System gain,
T: Time constant,
θ: Delay time.
***P(t)*:** Instantaneous measured temperature of the system (°C)

***SP*:** Setpoint temperature (°C)

***e(t)*:** The error between the setpoint temperature and the instantaneous measured temperature of the system

***u(t)*:** PID Control output (for our system, the unit can be V (Volt) or A (Ampere).

***Kp*:** Proportional gain of the PID

***Ki*:** Integral gain of PID

***Kd*:** Differential gain of PID

$\boldsymbol{R_t}$: Measured resistance,
$\boldsymbol{R_0}$: Reference resistance
$\boldsymbol{\alpha}$**:** Platinum temperature coefficient
$\Delta \boldsymbol{T}$**:** Temperature change.
$\boldsymbol{u(t-\Delta t)}$**:** To avoid uncertainty, the previous control output is taken as the basis in the unstable region.

ABBREVIATIONS

LCD: Liquid Crystal Display

RTD: Resistance Temperature Detector

PID: Proportional Integral Derivative

SSR: Solid State Relay

PWM: Pulse Width Modulation

IEEE: Institude of Electrical and Electronics Engineers

MATLAB: Matrix Laboratory

IDE: Integrated Development Environment

IEC: International Electrotechnical Comission

TSE: Türk Standartları Enstitüsü

ISO: International Standart of Organization

ADC: Analog Digital Converter

SPI: Serial Peripheral Interface

I2C: Inter-Integrated Circuit

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1. General Information

This study covers the design of an educational device developed to teach the basic principles of temperature control systems. Within the scope of the project, a low-cost and portable temperature control system has been created in which PID and ON-OFF control algorithms are applied, and control modes can be selected with the user interface. The system is designed in an integrated structure that includes the steps of sensor reading, receiving parameters from the user, generating the control signal and driving the heater. This control system is programmed with Arduino Uno microcontroller and offers an easy-to-use user interface with LCD screen.

This project topic has been chosen to provide an educational level understanding of temperature control problems frequently encountered in industry. PID and ON-OFF control algorithms are the most widely used methods in automation systems. Therefore, it is very important for students to learn the basic principles of these algorithms in practice. Professional controllers in the current market are usually expensive, complex and not designed for education. With this project, a model that facilitates the learning process and visually presents the basic behavior of the system is presented.

As a result of the study, the basic components of control systems, concepts such as error signal, feedback, gain are explained in a simple structure, and the effects of different control methods on the system response are made observable. Thus, students gained not only theoretical knowledge but also practical experience. This is an important contribution to control engineering education.

Nowadays, temperature control systems are widely used in many areas such as plastic injection molding machines, ovens, industrial heating systems, air conditioning devices. Especially PID control algorithm is preferred for fast and stable response in these systems. The model presented in this project is designed as a small-scale and understandable version of these systems in the industry and designed as a device that can be used for educational purposes.

1.2. Literature Research

Temperature control systems are among the most widely used structures in the automation world and teaching these systems with educational models has an important place in engineering education. For this reason, various studies on PID and ON-OFF control systems are evaluated both academically and practically.

Yılmaz et al. (2020), in their study titled “Temperature Regulation with PID Control

Application”, tested the success of the PID control algorithm in temperature regulation using Arduino and LM35 temperature sensor. In their study, the effects of different PID coefficients on the system response were evaluated and it was emphasized that the optimum values should be determined experimentally [1].

In their study published in IEEE Xplore, Zhao et al. (2019) developed a remote access system by combining a PID-based temperature controller with IoT infrastructure. The system includes a PID algorithm, temperature sensor, and microcontroller unit and offers cloud-based data monitoring. This study draws attention to the role of PID control in industry 4.0 applications [2].

Similarly, Aras and Demirtaş (2018), in their study published in Tübitak Journal of Electrical, Electronics and Computer Sciences, designed a low-cost PID training kit that can be used in university laboratories. The study provides an interactive interface that allows instructors to experiment with different PID parameters [3].

In the master's thesis written by Güzel (2021), the performance of industrial temperature control systems was modeled in MATLAB-Simulink environment and different control techniques were compared in YOK Thesis Library. In this thesis, the advantages of PID and fuzzy control methods were evaluated and it was observed that PID control gives more stable results in practice [4].

In another English-language study, Singh et al. (2020) presented Simulink modeling of PID temperature control used in embedded systems in their article titled "Simulation of Temperature Control Using PID in Embedded Systems." This study graphically presents the effects of PID parameters on system behavior [5].

In a study published in IEEE Xplore by Yan et al. (2019), the application of an auto-tunable PID controller to a temperature control system is discussed. In this study, the challenges faced by the conventional PID method are discussed and it is stated that significant improvements in the performance of the system are achieved thanks to the automatic tuning capability [6].

These studies reveal the importance of PID and ON-OFF temperature control systems both in education and industry and shed light on the place of this thesis project in literature.

1.3. Authenticity

This study is designed for teaching temperature control systems and differs from its counterparts in the literature by combining three different control algorithms, PID, ON- OFF and SlowPWM, in a single device. One of the most unique aspects of the project is that it was developed entirely for educational purposes. The system is designed in a simple, interactive and modular structure to enable high school and university level students to learn control algorithms practically.

The temperature-time data used for training the control algorithms were collected directly from the developed hardware and these data were processed in MATLAB Simulink environment to extract the transfer function. The data sets used in this context were not taken from ready-made sources; they were created by the project team to reflect the actual behavior of the system. Thus, the coefficients of the PID control algorithm were optimized by modeling over the real system. This process provides students with the opportunity to establish a strong link between theoretical knowledge and practice.

Both the hardware and software of the system were designed from scratch. Electronic circuit components, user interface, Arduino programming, sensor integration and control algorithms were all developed by the project team. In addition, an interactive structure that can receive real-time parameters from the user with the LCD screen menu system has been created, which strengthens the learning-oriented structure of the project.

In all these respects, the project stands out as a unique educational tool that differs from previous similar studies in terms of its educational approach, data collection, modeling and holistic design of the entire system.

1.4. Methods

In this project, a holistic development method was followed to demonstrate the applicability of temperature control algorithms for educational purposes. The project process consisted of five main phases: idea development, literature research, system design, simulation studies and testing on real hardware.

The idea of the project was to make temperature control, one of the basic subjects of control engineering, teachable to high school and university students. Based on the importance of hands-on learning in education, it is aimed to design a system where control algorithms can be presented in a simple and understandable way.

Following the clarification of the idea, a detailed literature review was carried out through academic publications, theses and technical documents on PID and ON-OFF control systems. By analyzing similar projects in the literature, existing solutions, methods and hardware used were evaluated and the most suitable structure for this project was determined.

In the design phase, all hardware components of the system (Arduino, PT100 sensor, MAX31865, SSR relay, LCD screen, buttons, etc.) were determined and circuit plans were created to work in harmony with each other. In the software part, menu-based control interface and algorithm flows were prepared with the codes developed on Arduino IDE. Push-button supported menus were created to allow the user to interact with the system.

In the simulation phase, the mathematical model of the system was extracted, and the transfer function was obtained with the System Identification Tool in MATLAB Simulink environment. The obtained model was optimized using PID Tuner and the most suitable PID coefficients were determined for the system.

In the last phase, the implementation process, the developed hardware and software were physically brought together, the system was run and the responses of the temperature values to the control algorithms were observed. Then, the system performance was analyzed graphically with the BetterSerialPlotter tool and the accuracy of the algorithms was tested.

Each step was carried out directly by the students and a systematic method focused on educational goals was followed throughout the project.

1.5. Common Impact

This project aims to make a significant contribution in the field of engineering education by introducing an education-oriented and low-cost temperature control system. The biggest widespread impact of the developed system is that it makes it possible to practically teach the basic algorithms of control engineering (PID, ON-OFF) to high school and university level students. In this way, students can directly experience concepts such as feedback, system gain, error signal, etc. through a real system without being limited to theoretical knowledge.

At the national level, especially in vocational and technical high schools or engineering faculties where laboratory facilities are limited, the use of cost-effective and portable systems such as this project can provide equal opportunities in education. Lecturers can offer their students hands-on control experiments both in-class and remotely. This system can also be used as a reference experimental setup for laboratory applications in engineering faculties.

On a local scale, this project has the potential to improve the quality of education in technical education institutions. It can also be directly used in practice-oriented programs such as vocational schools or faculties of applied sciences. Due to its contribution to technical education, the project can also have social impacts; it can increase students' interest in the engineering discipline and strengthen their professional skills.

It makes an indirect contribution in terms of employment. Students' mastery of control systems can facilitate their employment as more qualified personnel in sectors such as industrial automation, process control, HVAC systems after graduation. This provides benefits in terms of both individual development and meeting the needs of the sector in the long run.

The project also has the potential to generate publications. In particular, it can serve as an example for academic studies in areas such as engineering education, applied teaching techniques, microcontroller-based control systems. The study can be published in national and international platforms such as TÜBİTAK Science and Technical Journal, Engineer and Machinery Journal, OTU Thesis Library or Journal of Higher Education.

In these respects, the project is not only an individual graduation project, but also a disseminable and sustainable model that improves the quality of education.

1.6. Standarts

Temperature Sensors and Measurement Standarts

- **IEC 60751** – *Industrial platinum resistance thermometers and industrial platinum temperature sensors.* (It is the international reference standard for sensors such as the PT100).
- **IEEE 1451.4** – *Standard for a Smart Transducer Interface for Sensors and Actuators: Mixed-Mode Communication Protocols.*
- **TSE ISO 651** – *Sıcaklık ölçüm standartları, sensör doğruluğu ve uygulama yöntemlerini kapsar.*

Industrial Control and Automation Systems Standards

- **IEC 61131-3** – *Endüstriyel otomasyon sistemlerinde kullanılan programlanabilir kontrolörler için yazılım standartlarını belirler.* (Microcontrollers like Arduino can be compatible when configured with this logic).
- **IEC 61508** – *Functional safety of electrical/electronic/programmable electronic safety-related systems.* (Especially in the use of SSR relay, it is taken into consideration for circuit safety).
- **IEEE 1588** – *Precision Clock Synchronization Protocol for Networked Measurement and Control Systems.* (Recommended standard for time-based data collection applications.)

Electrical Safety and Connection Standards

- **IEC 60204-1** – *Safety of machinery – Electrical equipment of machines.*
- **TSE EN 61010-1** – *Elektriksel ölçüm, kontrol ve laboratuvar kullanımı için güvenlik gereksinimleri.*

Microcontroller Systems and Embedded Software Standards

- **ISO/IEC/IEEE 24765** – *Systems and software engineering—Vocabulary* (Contains embedded software development terms and definitions).
- **MISRA-C** – *Mikrodenetleyici projelerinde kullanılan C dili için güvenli kodlama standartları.* (It can be referenced in terms of configuration in Arduino codes).

1.7. Work Schedule

Schedule 1.1.1. Work-Time Schedule

İP No	**Name and Purpose of the Work Package**	**Worked by Whom**	**Time Interval**	**Contribution to the Project**
1	Creating the project idea, literature research and revealing the original idea	Enes SEZGİN (L) Ülkü Ece KUŞÇU	September 2024	The project idea was clarified and originality analysis was made. It was supported and grounded with literature.
2	Determination of the methods to be applied and theoretical model studies	Ülkü Ece KUŞÇU (L) Enes SEZGİN	September - November 2024	Control methods, sensor structure, hardware-microcontroller relationships are defined.
3	Hardware design and component specification	Enes SEZGİN (L) Ülkü Ece KUŞÇU	November - December 2024	Hardware such as PT100, SSR relay, LCD, Arduino Shield were determined and the system schematic was drawn.
4	Creating the simulation model and extracting the PID coefficients in MATLAB	Enes SEZGİN (L) Ülkü Ece KUŞÇU	December 2024 -January 2025	A heater suitable for the project was selected. Experiments were carried out with different types of heaters on the market.
5	Hardware integration and software preparation	Enes SEZGİN (L) Ülkü Ece KUŞÇU	January 2025	Arduino codes were developed, button menu structure and control algorithms were installed.

Schedule 1.1.2. Work-Time Schedule (Continued)

İP No	Name and Purpose of the Work Package	Worked by Whom	Time Interval	Contribution to the Project
6	Assembling the prototype system and testing the temperature control system	Enes SEZGİN (L) Ülkü Ece KUŞÇU	January - February 2025	The physical installation is done, sensors are connected, and the heater is switched on.
7	Conducting tests and verifying results	Enes SEZGİN (L) Ülkü Ece KUŞÇU	February - March 2025	System response was taken according to set temperature and data was tested on LCD and serial monitor. System optimization was improved by revising the control algorithms.
8	Writing the graduation book and preparing for presentation	Enes SEZGİN (L) Ülkü Ece KUŞÇU	March -May 2025	The transfer function was extracted, and the coefficients were optimized with PID Tuner via Simulink. All project steps are reported and supported with graphs and tables.
9	Preparation of project thesis and presentation	Ülkü Ece KUŞÇU (L) Enes SEZGİN	May -June 2025	The final thesis report and poster presentation of the project were prepared in accordance with the specified procedures and principles.
10	Final exams, Graduation Project Exhibition and presentations and completion of the project	Enes SEZGİN (L) Ülkü Ece KUŞÇU	June 2025	In the Graduation Project Exhibition, the prototype was exhibited in a working and suitable condition, and the poster presentations of the graduation thesis were submitted to the jury and evaluated by the jury.

Schedule 1.2.1. Risk analysis and Plan B

İP No	Contribution to the Project	Risk Analysis
1	Work Packages Organization and Work ManagementThe project idea was clarified, and originality analysis was made. It is supported and grounded with literature.	If insufficient content is found in the literature review, a new focus area can be determined in consultation with the advisor.
2	Control methods, sensor structure, hardware-microcontroller relationships are defined.	If undecided about the methodology, similar graduation projects and open source projects can be analyzed to determine a direction.
3	Hardware such as PT100, SSR relay, LCD, Arduino Shield were determined, and the system schematic was drawn.	For hardware components that cannot be supplied, alternative parts are listed. In particular, replacement options for the SSR relay and PT100 have been identified.
4	A heater suitable for the project was selected. Experiments were carried out with different types of heaters on the market.	If the system model cannot be extracted in MATLAB Simulink, sample modeling can be done through transfer functions of ready-made similar systems.
5	Arduino codes were developed; button menu structure and control algorithms were installed.	If unexpected errors occur during the code development process, basic functions can be temporarily provided by taking part from sample Arduino projects.
6	The physical installation is done, sensors are connected, and the heater is switched on.	Spare circuit elements and alternative connection diagrams have been prepared in case of failure during physical installation.

Schedule 1.2.2. Risk analysis and Plan B (Continued)

7	System response was taken according to set temperature and data was tested on LCD and serial monitor. System optimization was improved by revising the control algorithms.	If the test results do not give the expected performance, the PID parameters are updated and optimized again in MATLAB.
8	The transfer function was extracted, and the coefficients were optimized with PID Tuner via Simulink. All project steps are reported and supported with graphs and tables.	If there is a loss of time during the writing process, certain sections will be scheduled to be written in order every day according to the writing template.
9	The final thesis report and poster presentation of the project were prepared in accordance with the specified procedures and principles.	If there is a delay for the presentation file and poster, a template-based presentation can be prepared based on the presentation templates already made.
10	In the Graduation Project Exhibition, the prototype was exhibited in a working and suitable condition, and the poster presentations of the graduation thesis were submitted to the jury and evaluated by the jury.	If there is a power outage or system failure on the day of the prototype exhibition, support will be provided with test videos showing the operation of the system.

1.8. Work Packages Organization and Work Management

The first 5 work packages are related to Engineering Design and the last 5 work packages are related to the Capstone Project. The students who take part in the project should take turns leading these work packages and warn the other students in charge of that work package to complete the work package they are responsible for on time. Each student should be the leader of at least one work package and should take responsibility for ensuring that it is completed by telling the other students what they need to do.

Who will lead which work package and who among the other students in charge of that work package will do which work in how much time should be given in detail under this sub-heading.

2. THEORETICAL BACKGROUND

2.1. General Information

In this study, it is investigated how a system based on temperature control can be realized with different control algorithms (PID, ON-OFF, SlowPWM). These methods, which are widely used in control systems, are designed to ensure that the system remains constant at a certain target value. Within the scope of the study, a temperature controller equipped with an Arduino microcontroller was developed and three different control algorithms were successfully implemented on this system.

The system generally consists of a temperature measurement element (PT100 sensor), signal processing module (MAX31865), controller (Arduino Uno), output driver element (SSR Relay), user interface (LCD screen and buttons) and heater (headlight bulb). The desired temperature value and control parameters are entered via the user interface, and then the system generates output by deciding on the error value with temperature feedback.

The theoretical basis of the developed system is based on control theory and basic heat transfer principles. Since temperature can change continuously as a physical quantity, systematic and feedback control algorithms are needed to keep this change in balance. The theoretical foundations of the control techniques and hardware elements used in the project are detailed in the following sub-headings.

In PID ON-OFF Controller Temperature Control Device

2.2. System Modeling and Optimization of PID Coefficients with MATLAB Simulink

In the design of control systems, it is a critical step to mathematically model the actual system behavior and determine the appropriate control parameters. In this project, a system model was created in MATLAB Simulink environment in order to understand the dynamic characteristics of the temperature control system and to optimize the performance of the PID control algorithm.

Importance of Transfer Function

The transfer function defines the mathematical relationship between the input and output of the system. This function is usually expressed using the Laplace transform and is a fundamental tool for analyzing the dynamic response of the system. Thanks to the transfer function, the response time, stability and other important properties of the system can be calculated.

Model Extraction with System Identification Tool

The temperature-time data obtained from the real system was analyzed using MATLAB's System Identification Toolbox. This tool allows you to estimate the most appropriate transfer function model from experimental data. In the project, the temperature change when the heater is fully powered was recorded and this data set was used for modeling.

The transfer function obtained by this method usually consists of transport delay and first or second order dynamic components. For example, the transfer function is expressed as follows: (Reference is made in the introduction for necessary abbreviations and symbols).

$$G(s) = \frac{K}{\tau s + 1} e^{-\theta s}$$

Equation 2.1.

Parameter Optimization with PID Tuner

The obtained transfer function model is connected to the PID Controller block in Simulink and the PID coefficients (Kp,Ki,Kd) are optimized using MATLAB's PID Tuner application. The PID Tuner adjusts the parameters to improve the system response according to user-specified targets. These targets include a rise time, settling time and overshoot.

$$u(t) = Kp\, e(t) + Ki \int e(t)dt + Kd\, \frac{de(t)}{dt}$$

Equation 2.2.

$$e(t) = SP - P(t)$$

Equation 2.3.

As a result of the optimization, the system reaches the set temperature in a stable, fast and low oscillation manner. This process was verified by experimental tests and the obtained PID parameters were applied to the real system.

Academic and Practical Contributions

This modeling and optimization approach is a concrete example of how theoretical control methods can be integrated with real-world applications. It has been an important educational material for students in terms of directly relating the mathematical foundations of the PID algorithm learned in theory to its effects on the physical system. In addition, this method optimizes the performance of the control algorithm to increase energy efficiency and system robustness.

2.3. Temperature Sensors and Measurement

In temperature control projects, accurate and reliable measurement of temperature is vital for system performance. In this study, PT100 sensor belonging to the resistance temperature detector (RTD) class is used. PT100 is a high accuracy sensor that measures temperature from the temperature dependent resistance value of platinum wire. "PT" stands for platinum, while "100" stands for the resistance value of 100 ohms at 0 °C. The variation of the resistance of the sensor with temperature has an approximately linear relationship as follows:

$$R_t = R_0 (1 + \alpha \Delta T)$$

Equation 2.4.

These sensors can typically operate in the range from -200 °C to 850 °C and are widely used in industrial applications. The temperature measurement in the project requires precise conversion of the analog resistance change of the PT100 sensor into digital data. The MAX31865 module is used for this purpose.

2.4. MAX31865 RTD Thermocouple Amplifier Module

MAX31865 is a converter integration that converts the resistance signal from PT100 and similar resistance type temperature sensors into digital data format. It communicates with the microcontroller via SPI communication protocol. With the bridge circuit principle, it measures the small resistance changes of the sensor with high accuracy and transmits them to the microcontroller as digital data. In this way, temperature values can be read precisely by Arduino.

2.5. Arduino and Microcontroller Based Control

Arduino Uno is an open-source development board based on the 8-bit AVR microcontroller. It is suitable for controlling sensors and actuators with digital and analog pins, communication interfaces such as SPI, UART. In the project, Arduino implements control algorithms by processing temperature data from MAX31865 and operates the heater via SSR relay. In addition, a user interface is created through push-buttons, enabling parameter input and menu navigation.

2.6. ON-OFF Control Algorithm

ON-OFF control is a simple control method that allows the output of the system to be only on or off. The output is active when the error signal is positive, otherwise it remains off. Although this method is preferred especially in simple applications, it can cause temperature fluctuations and continuous on-off cycles in the system.

Mathematically:

$$u(t) = \begin{cases} 1, & e(t) - H < 0 \\ 0, & e(t) + H > 0 \\ u(t - \Delta t), & otherwise \end{cases}$$

Equation 2.5.

2.7. PID Control Algorithm

PID control evaluates the error signal using proportional, integral and derivative components. It uses PID coefficients (Kp, Ki, Kd) to optimize the system performance. PID control output:

$$u(t) = Kp\, e(t) + Ki \int e(t)dt + Kd\, \frac{de(t)}{dt}$$

Equation 2.6.

The advantages of this algorithm are fast response, error correction over time and increased system stability.

2.8. SlowPWM Control Method

SlowPWM is another method used to precisely control the heater. It is similar to PID logic but with differences. The output of the system is controlled by PWM waves. As the system approaches the setpoint temperature, its output decreases proportionally. When it reaches the setpoint, it provides precise control by controlling the system with a PWM signal in a very low range such as 3-4%. For adjustment, it is sufficient to enter only the desired setpoint temperature and does not require any other input.

2.9. Output Control with SSR Relay

SSR relays are faster, quieter and more durable switching elements than mechanical relays. In the project, it was used to turn the heater on and off according to the control signal of Arduino. It switches at high frequency, especially when generating a signal at low PWM. A mechanical relay cannot switch this fast, and even if it could, its mechanical parts would break down in a short time.

3. DESIGN

3.1. General Information

In this project, both hardware and software components of the temperature control system are carefully designed. The design process includes various engineering calculations, system integration, component selection and cost planning based on the theoretical background. The physical structure of the system was realized by considering the boxing details, component layouts and safety precautions. All design decisions were made to ensure the reliability, accuracy and user-friendliness of the system. In addition, the bill of materials, preliminary cost calculation and legal assessment are detailed before project implementation.

3.2. Dimensioning

In the power box (heating chamber), the heater and sensor components are installed in an insulated box. The insulation used an aluminum sheath in the form of a shiny material to insulate the heat and protect the signal and power cables behind it. The dimensions of the box were 18.5 cm × 12 cm × 8 cm and the internal volume was adjusted to be sufficient for the safe operation of the PT100 sensor and the 12V 55W headlight bulb (8cm × 4.5cm × 4.5cm × 4.5cm). The heater was fixedly mounted inside the box and the PT100 was positioned near the heater (~2cm). The SSR relay (6cm x 4.5cm x 4.5cm) was mounted just above the outside of the box to avoid being affected by the high temperature.

In the temperature controller, the LCD screen and buttons are placed on the front panel of the system. The dimensions of the box were set as 10cm x 10cm x 8cm, and the dimensions were chosen to make the wiring inside easy, to make the inside of the box look tidy and for easy placement of the elements. Arduino Uno (6.8cm x 5.4cm) and MAX31865 thermocouple module (2.5cm x 2.8cm) were fixed inside the box. The signal cable distance between the heating chamber and the temperature controller is approximately 20cm.

3.3. System Components and Selections

a. **PT100 Sensor:** This platinum resistance temperature sensor, widely used in industrial standards, shows a resistance of 100 ohms at 0 °C and has a temperature coefficient of 0.00385 $°C^{-1}$. The two-wire connection enables fast integration without taking up space in the system. It has a linear response and can operate from -200 °C to +850 °C. Selected for high accuracy temperature measurement. 2-wire version is used.

b. **MAX31865 RTD Thermocouple Amplifier Module:** Designed to measure small signals from RTD sensors, this module communicates over SPI protocol. It has 15-bit ADC resolution and offers 60 Hz and 50 Hz filtering options. It can be supplied with 3.3V or 5V and is compatible with PT100/ PT1000 sensors. It is used to transmit the resistance data from the PT100 sensor to the Arduino via SPI.

c. **Arduino Uno:** Based on the Atmega328P microcontroller, this board has a 16 MHz crystal oscillator, 14 digital I/O pins (6 of which support PWM output), 6 analog inputs, USB connection, reset button and power jack. It contains 32 KB flash memory (0.5 KB for the bootloader), 2 KB SRAM and 1 KB EEPROM. It was chosen to implement advanced control algorithms and manage the user interface. It is also very suitable for ease of use in software and for the educational purpose of the project.

d. **LCD 16x2 + I2C Module:** This display, which can show 16 characters × 2 lines of text, uses only 2 pins (SCL, SDA) with the I2C communication module. It usually has a PCF8574 chip. Contrast can be adjusted, it works with 5V. In order to provide interaction with the user, a pin-saving I2C communication display is preferred.

e. **SSR Relay (Solid State Relay):** This relay operates with 12V DC control voltage and can switch voltages up to 240V on the AC side. Thanks to its optocoupler structure, the microcontroller side is electrically isolated. It is silent and long- lasting, resistant to sudden on/off. A semiconductor relay, which works quieter and faster than mechanical relays, is used to switch the heater circuit.

f. **12V 55W Headlight Bulb:** This halogen bulb, which is used in vehicle headlights, acts as a heater within the scope of the project. It works with 12V DC and draws approximately 4.6A current. Its thermal dissipation is fast and can reach high temperature in a small volume. It functions as a heater of suitable power and size for the system.

g. **Push-Buttons:**These are mechanical buttons typically of the NO (normally open) type. They are read via digital input pins on the Arduino, where internal pull-up resistors are used to establish logical 0 and 1 levels. They serve functions such as menu navigation, confirmation, and increment/decrement operations. Specifically, they are utilized for menu control, input of the temperature setpoint, and adjustment of PID parameters.

3.4. Applied Methods

The following methods were implemented within the scope of the project:

- **Hardware Design:** The placement and wiring of all components were systematically arranged, with particular attention paid to temperature effects and electromagnetic interference.
- **Control Methods:** ON-OFF, PID, and SlowPWM algorithms were implemented via software. The PID coefficients were optimized using MATLAB Simulink.
- **Numerical Analysis:** The system behavior was analyzed using a transfer function model, and the PID control was configured accordingly.

3.5. Software

The software, developed in the Arduino IDE environment, was structured to support three primary control methods: ON-OFF, PID, and SlowPWM. Multi-level menu structures were created for navigation, with interaction managed through push-button controls. The software flow consists of the following key sections:

- System initialization and welcome screen
- Menu structure and transitions
- Data input from the user (temperature setpoint, PID parameters)
- Temperature measurement and execution of control algorithms
- Information display on the LCD

Details of the software code can be provided in the appendix upon the advisor's request. MATLAB Simulink was used for simulation throughout the project; the temperature- time curve of the system was modeled, and PID tuning was performed. Simulation details will be discussed in the following section.

3.6. Cost of Materials and Economic Analysis

Table 3.1. Material List

Name of material	Aim of use	Unit price (TL)	Piece	Price (TL)
PT100 Thermoresistance Temperature Sensor	Selected for high accuracy temperature measurement. 2-wire version used.	66,21₺	1	66,21₺
MAX31865 RTD Temperature Sensor Amplifier	It was used to transmit the resistance data from the PT100 sensor to the Arduino via SPI.	720,98₺	1	720,98₺
Arduino UNO Rev3	It was chosen to implement advanced control algorithms and manage the user interface.	187,29₺	1	187,29₺
16x2 LCD Display with I2C Module	In order to provide interaction with the user, a pin-saving I2C communication display was preferred.	121,07₺	1	121,07₺
SSR Relay 40A DC/DC	A semiconductor relay, which operates quieter and faster than mechanical relays, is used to switch the heater circuit.	316,00₺	1	316,00₺
Bosch H7 12V 55-60W headlight bulb	It functions as a heater of appropriate power and size for the system.	128,99₺	1	128,99₺
Push button	Menu control is used for temperature set value entry and PID parameters adjustment.	4,50₺	4	18,00₺
12V 8.5A DC Power Supply	It was used to power the system.	273,00₺	1	273,00₺
Others	3D device box, heater room box, jumper cables, insulation materials, etc.	850₺	1	850,00₺
			TOTAL	2681.54₺

An economic analysis of our temperature controller project reveals that the system was built with a total cost of 2681.54₺. Overall, the design balances functionality, educational value, and cost-efficiency, making it a feasible solution for teaching automation concepts.

3.7. Legal Aspects

This project is an educationally developed temperature control system, primarily designed for use in laboratory and instructional environments. Therefore, direct commercial production or large-scale industrial applications were not targeted within the scope of this project. However, certain legal considerations must be taken into account during the development and implementation phases.

Safety and Electrical Installation Regulations

The electronic components and power supplies used in the project operate at 12V DC voltage and low current levels and have been designed in compliance with low-voltage and user safety principles as defined by the Electrical Installation Regulations. The risk of electric shock has been minimized, ensuring conformity with relevant safety standards.

Waste Management and Environmental Legislation

At later stages of the project, appropriate recycling and disposal of components are essential to reduce environmental impact and manage electronic waste properly.
Compliance with Law No. 2872 on the Environment and related regulations regarding the disposal of electronic waste is mandatory.

Copyrights and Software Licenses

The software developed within the scope of the project, along with the open-source libraries utilized, have been used in accordance with the relevant license terms and within permitted limitations on source code usage. All software components were developed ethically with respect to copyright laws.

Educational Use and Liability

Due to the educational nature of the project, it is advisable to include a clear liability statement for users. Any technical malfunction or safety risk arising from applications outside laboratory environments shall be the responsibility of the users. More specific legal regulations may be examined and detailed in subsequent phases. In general, the project is legally compliant and safe for educational and research use.

4. SIMULATION STUDIES

4.1. General Information

Within the scope of the project, simulation studies were conducted in **MATLAB Simulink** and **Better Serial Plotter** environments to observe the behavior of the temperature control system in advance and to analyze the effects of control algorithms. During the simulation process, a mathematical model of the system was developed, with the equations used in the model explained in the theoretical background section. Using ready-made software packages, the system was modeled under conditions similar to the real setup. This approach allowed for a reliable analysis of system behavior prior to physical experimental testing.

4.2. Simulation Software

PID Operating Function:

The software is defined within an infinite while loop. The function time is determined by the millis function, which sets the scanning interval of the program. Integral anti- windup is implemented to prevent the integral error from diverging to infinity, which is crucial for system stability. The derivative term is calculated by subtracting the previous error value from the current error and dividing by the elapsed time. The PID output is then computed. Very small PID outputs are filtered out using a dead-band filter. The calculated values are displayed on the LCD as temperature and PID output data.

```
while (true) {
  if (millis() - previousMillis >= interval) {
    previousMillis = millis();

    float temperature = thermo.temperature(RNOMINAL, RREF);
    input = temperature;
    error = setpoint - input;

    integral += error * dt;

    // integral anti-windup
    integral = constrain(integral, -50, 50);

    double derivative = (error - lastError) / dt;

    output = Kp * error + Ki * integral + Kd * derivative;
    output = constrain(output, 0, 255);


    // Deadband filter
    if (abs(error) < 0.2 && output < 30) {
      output = 0;
    }

    analogWrite(controlPin, (int)output);
    lastError = error;
    // PID output scaling
    double sc_output = output * 100 / 255;

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Sicaklik: ");
    lcd.print(temperature, 1);

    lcd.setCursor(0, 1);
    lcd.print("PID: ");
    lcd.print((int)sc_output);

    Serial.print(temperature);
    Serial.print("\t");
    Serial.println((int)sc_output);
  }
```

Figure 4.2.1. PID Operating Function

On-Off Operating Function:

The program's execution (scanning) interval is also calculated using the millis function (1 second was used in the application). The temperature value is obtained via the thermo.temperature() function from the MAX31865 module library. If the measured temperature is less than the set temperature minus the hysteresis value, the heater is turned on. Conversely, if the temperature exceeds the set temperature plus the hysteresis value, the heater is turned off. The instantaneous temperature and the relay's on/off status are displayed on the screen.

```
void runOnOffControl() {
  static unsigned long previousMillis = 0;
  const long interval = 1000;

  while (true) {
    if (millis() - previousMillis >= interval) {
      previousMillis = millis();

      float temperature = thermo.temperature(RNOMINAL, RREF);

      if (temperature <= (setTemp - hist)) {
        digitalWrite(controlPin, HIGH);
      } else if (temperature >= (setTemp + hist)) {
        digitalWrite(controlPin, LOW);
      }

      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("TEMP: ");
      lcd.print(temperature, 1);

      lcd.setCursor(0, 1);
      lcd.print("RLY: ");
      lcd.print(digitalRead(controlPin) ? "OPEN" : "CLSD");

      Serial.print(temperature);
      Serial.print("\t");
      Serial.println(digitalRead(controlPin));
    }
```

Figure 4.2.2. On-Off Operating Function

Saving Data to MATLAB Simulink Workspace:

In the experiment conducted to measure the full-power heating time and temperature data, all collected data were saved in .csv or .xlsx formats. To load these data into the MATLAB Workspace environment, the following code was used. Temperature and time data, saved in column format, were extracted and stored as variables (objects).
Subsequently, these data were utilized in the System Identification Tool for modeling the system behavior.

```
% Excel dosyasını oku
data = readtable('PID_data_2.xlsx');

% Zaman, sıcaklık (output), ve PID çıkışı (input) sütunlarını ayır
time = data{:,1};                    % Program Time [s]
temp = data{:,2};                    % temperature (output)
pid_pwm = data{:,3} * (255/100);     % PID_output scaled to 0-255 (input)

Ts = mean(diff(time));   % örnekleme zamanı (varsayalım sabit)
data_id_2 = iddata(temp, pid_pwm, Ts);  % çıkış: sıcaklık, giriş: PID sinyali

ident
```

Figure 4.2.3. Importing Data on MATLAB Workspace

4.3. System Modelling

The modeling approach for the system to be simulated is explained, including the model equations or structure. Necessary explanations are provided to clarify how the model operates.

Proteus Model:

The two-wire PT100 sensor is connected to the RTD+ and RTD– terminals of the MAX31865 module. The VDD pins are connected to 5V, and the GND pins are connected to the Arduino GND pins. RRef is the reference resistor for the PT100; temperature variations are calculated based on this reference resistance. The module's SPI communication pins—SCK (SCLK), SDI (MOSI), SDO (MISO), and CS—are connected respectively to Arduino pins 13, 12, 11, and 10. The LCD display's VDD is connected to +5V and VSS to GND. The I2C communication pins SCL and SDA are connected to Arduino analog input pins A5 and A4. Arduino pin 6 serves as the SSR relay control pin and is connected to the relay's +VDC input; the relay's –VDC input is connected to GND. The line from the 12V power supply is connected to the relay's +VDC output, which then connects to the positive terminal of the heater. The negative terminal of the heater returns to the power supply's GND, thus completing the circuit.

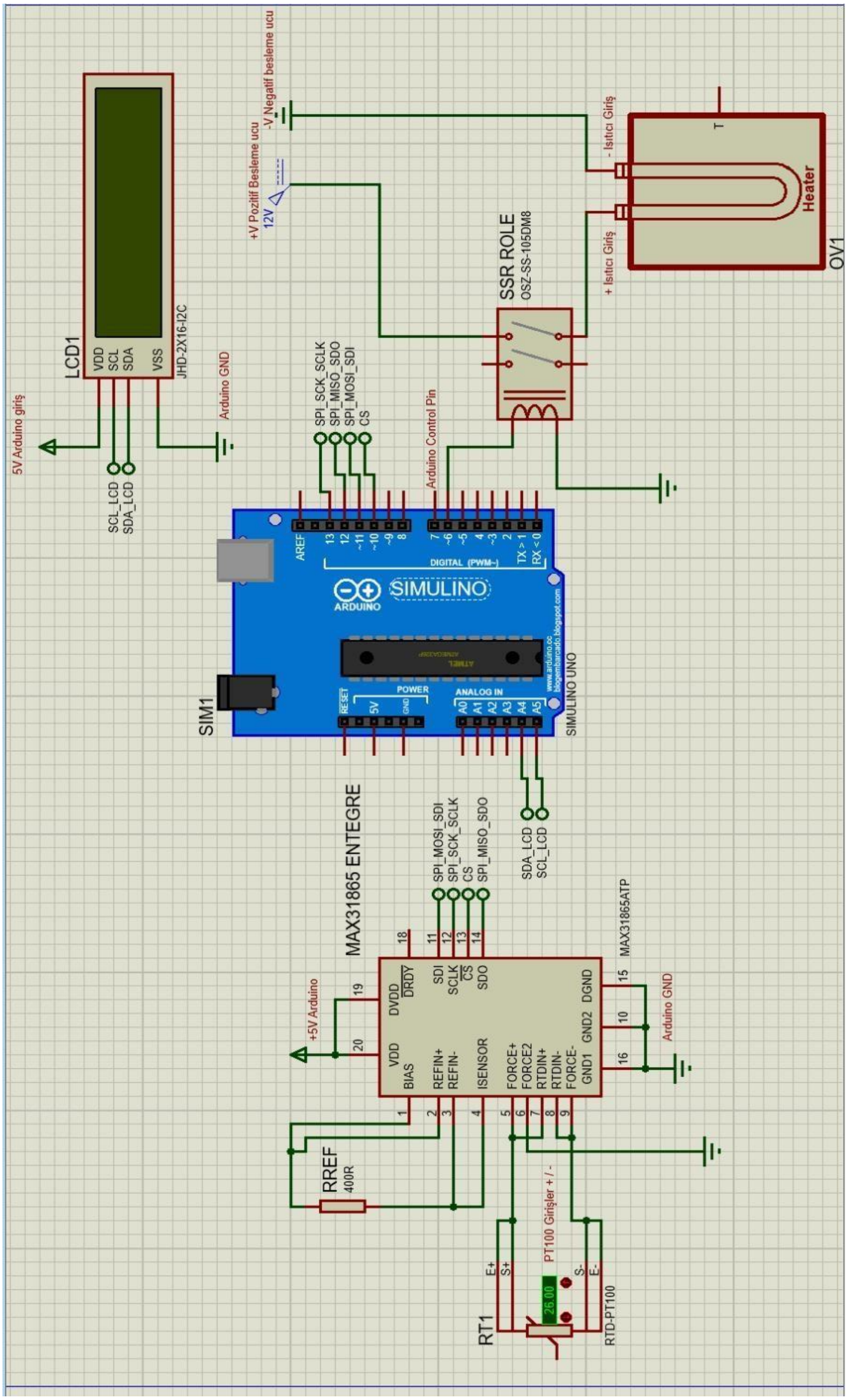

Figure 4.3.1. Proteus Model

MAX31865 RTD Amplifier Module:

A more detailed circuit schematic of the module is provided here. The purpose of this module is to amplify the very small resistance changes occurring with temperature variations, converting the signal to a level that can be detected by the Arduino.

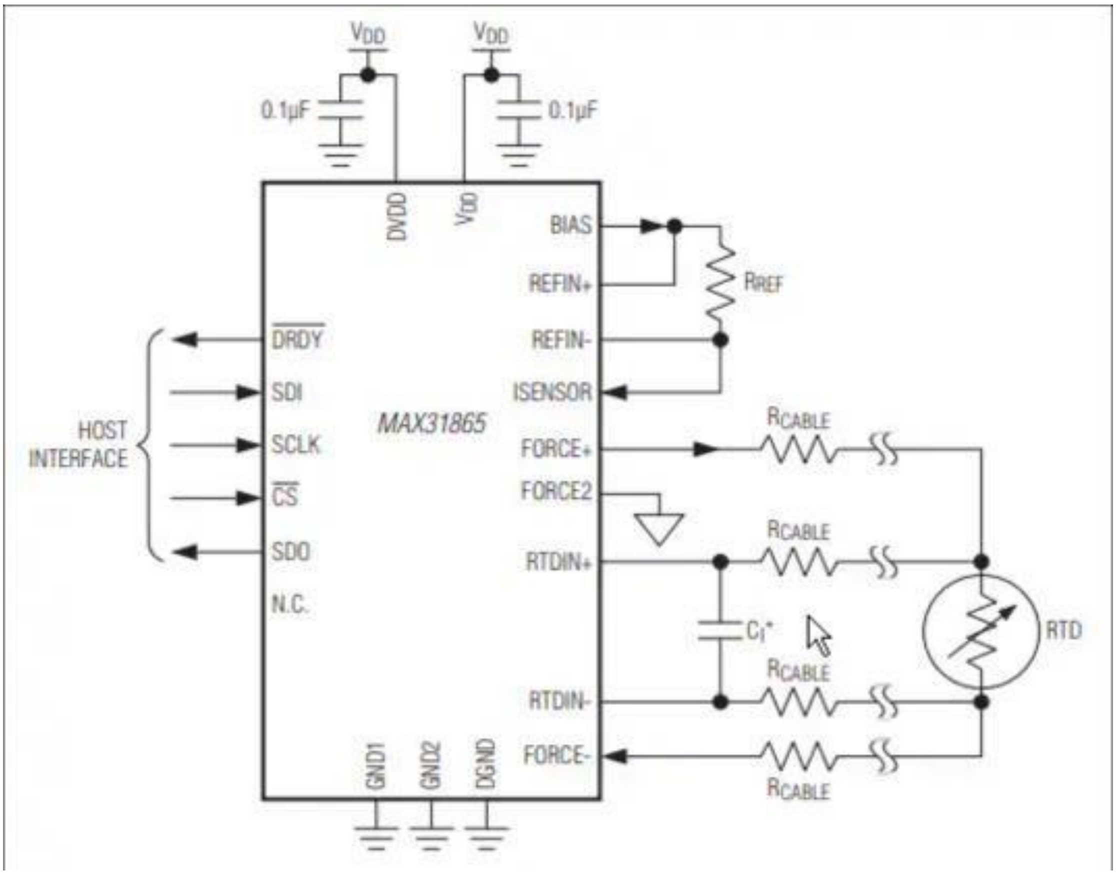

Figure 4.3.2. Amplifier Module MAX31865 RTD Module Circuit

Transfer Function Development Model:

The data saved to the workspace are imported into the **System Identification Toolbox**. These imported data are processed in the **Estimate with Transfer Function** mode, which extracts the transfer function parameters from the available data. In the graph, the value "–8" represents an 8-second delay, while the other coefficients correspond to the transfer function parameters.

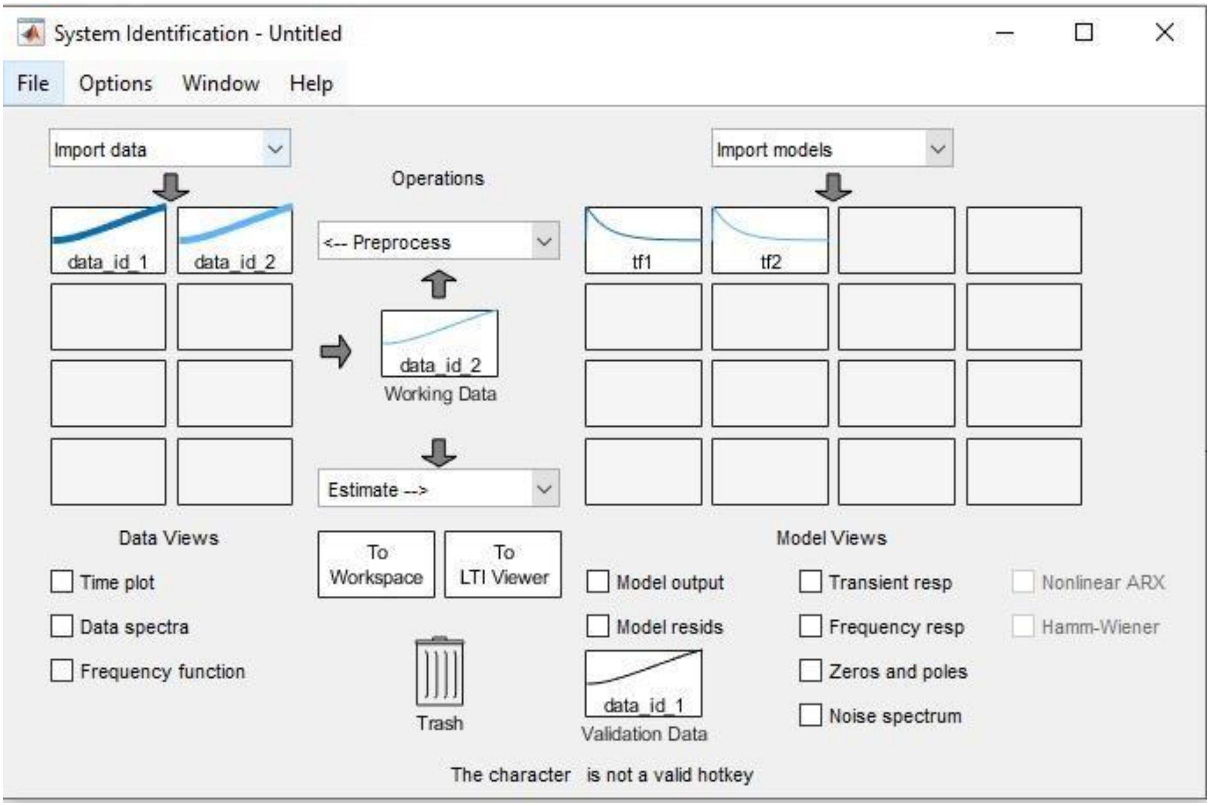

Figure 4.3.3. System Identification Toolbox

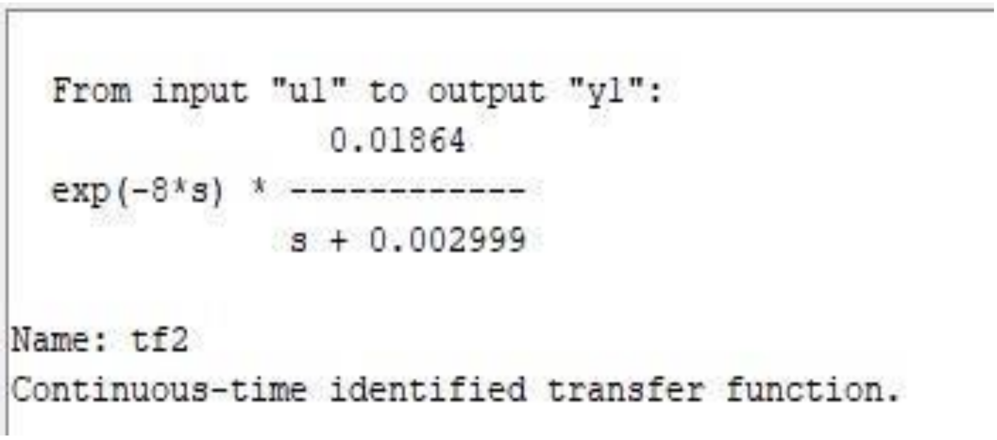

Figure 4.3.4. Transfer Function Coefficients

MATLAB Simulink Model:

The set temperature value was configured as 40°C. The initial temperature was measured as 20.39°C. The process temperature is continuously accumulated from the initial value in an attempt to reach the set temperature. The difference between these values (error) is continuously updated to create a feedback mechanism. This mechanism enables the PID controller to generate an output appropriate to the error. The transfer function here represents the mathematically modeled system, created using the measured temperature data. The transport delay corresponds to the time delay between when the temperature data is received from the heater and processed by the Arduino. This delay was measured and entered as a fixed value. The scope block allows visualization of the system output as a graph.

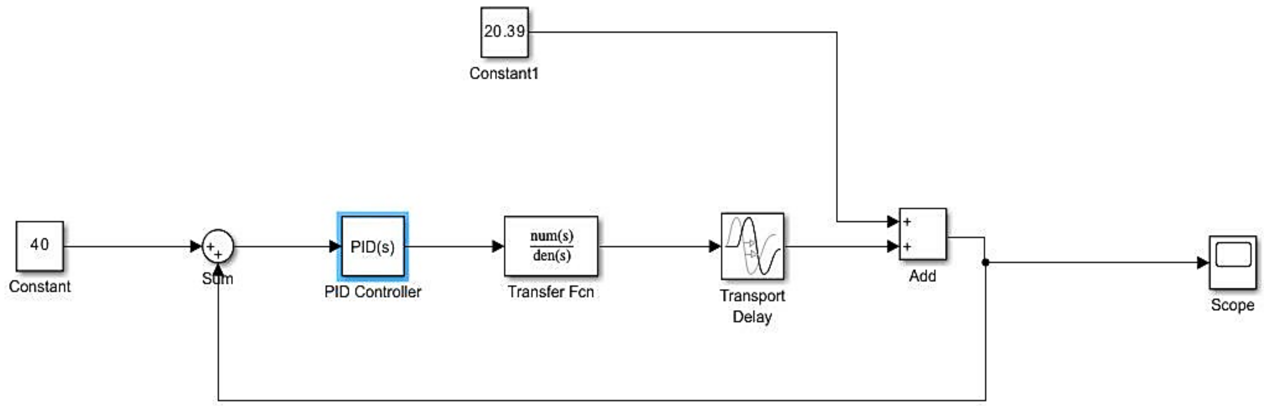

Figure 4.3.5. MATLAB PID Simulink Model

4.4. Simulation

Simulation diagrams and the procedure for conducting the simulation are explained in this section.

PID Tuning:

Based on the given reference model, the system can be calibrated as desired. The system output's transient behavior (heating response) can be adjusted to be more "aggressive" or "robust" depending on requirements. Additionally, settings for making the system response slower or faster are configured to determine the speed of the system's reaction. Once these adjustments are made suitably for our system, the tuning process provides appropriate PID coefficients. The optimally tuned PID parameters obtained are (Kp = 1.6, Ki = 0.2, Kd = 3.5).

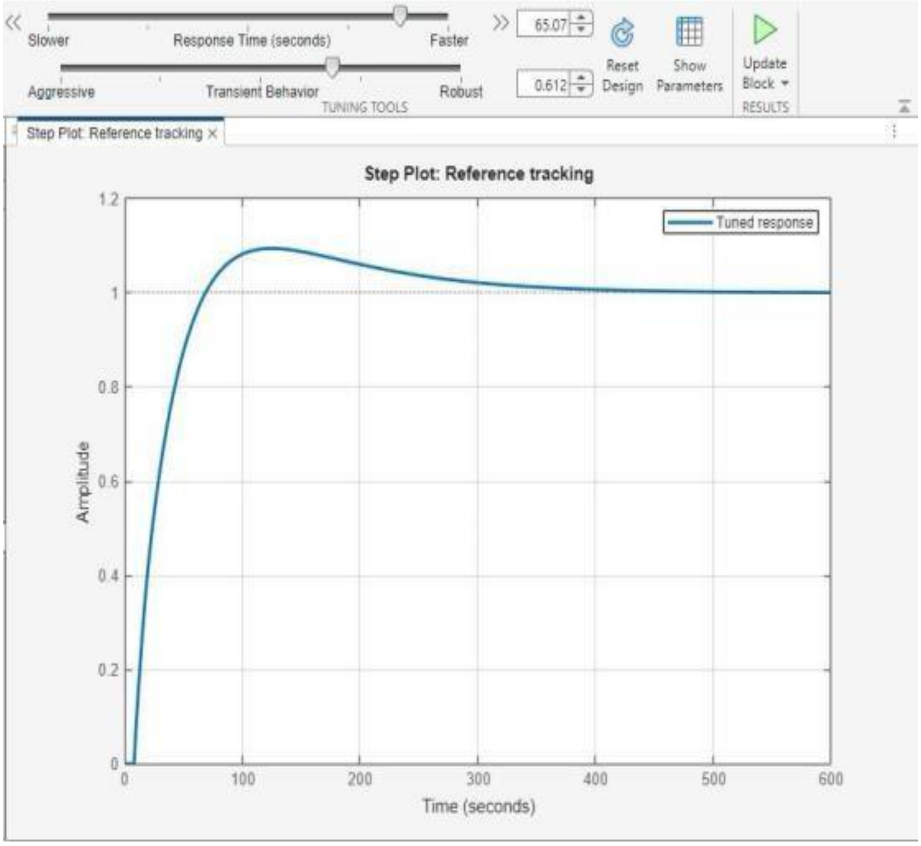

Figure 4.4.1. PID Tuning

Show Parameters

Controller Parameters

	Tuned
P	-1.5589
I	-0.011531
D	-3.9466
N	0.027709

Performance and Robustness

	Tuned
Rise time	42.2 seconds
Settling time	353 seconds
Overshoot	10.1 %
Peak	1.1
Gain margin	15 dB @ 0.157 rad/s
Phase margin	66.6 deg @ 0.0277 rad/s
Closed-loop stability	Stable

Close

Figure 4.4.2. PID Tuning Parameters

PID Control Transfer Function: Graph showing the similarity between our real system data and the transfer function we modeled. The graph shows 87.12% similarity of the model.

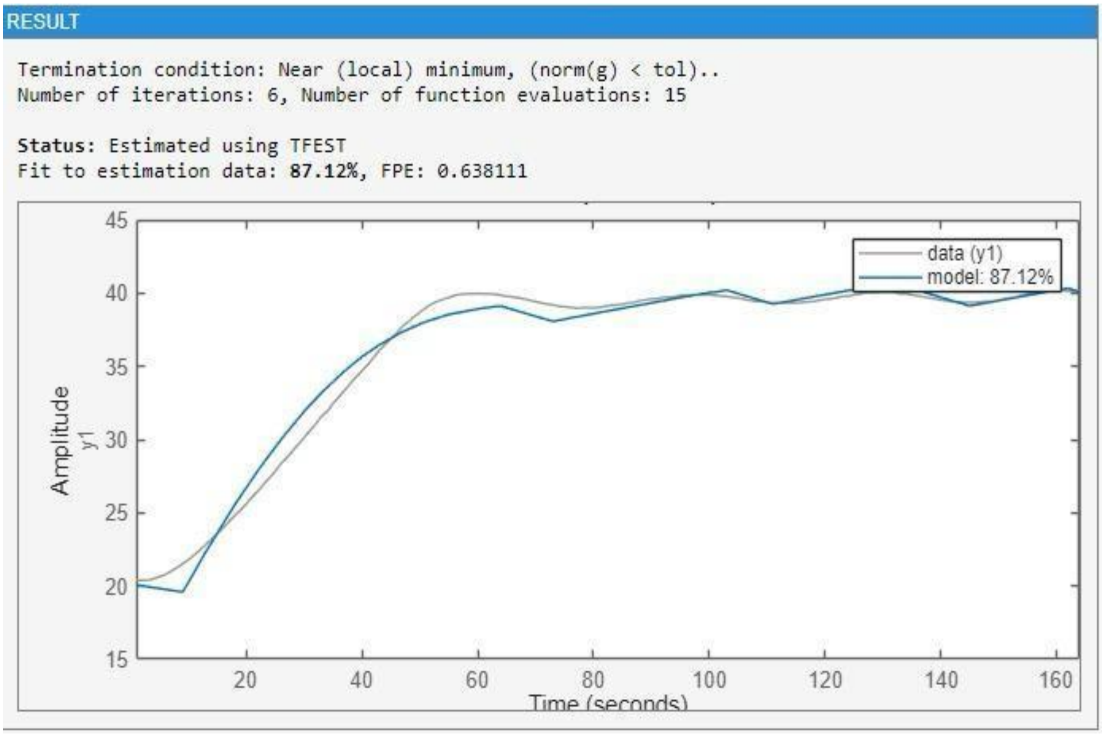

Figure 4.4.3. PID Transfer Function

On-Off Control Transfer Function: Graph showing the similarity between our real system data and the transfer function we modeled. The graph shows 91.94% similarity of the model.

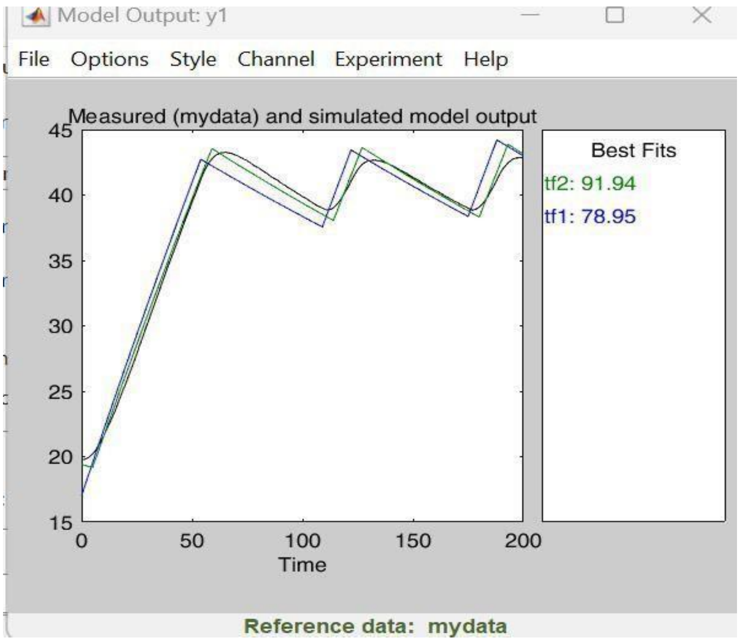

Figure 4.4.4. On-Off Transfer Function

On-Off Control Operation: For a set value of 40°C and a hysteresis value of 1°C, the temperature-time graph of the system is as shown in the figure. The system provides precise control with a loss of approximately 1.2°C (excluding the hysteresis value). (The blue graph shows the temperature-time graph, and the orange graph shows the relay on or off status)

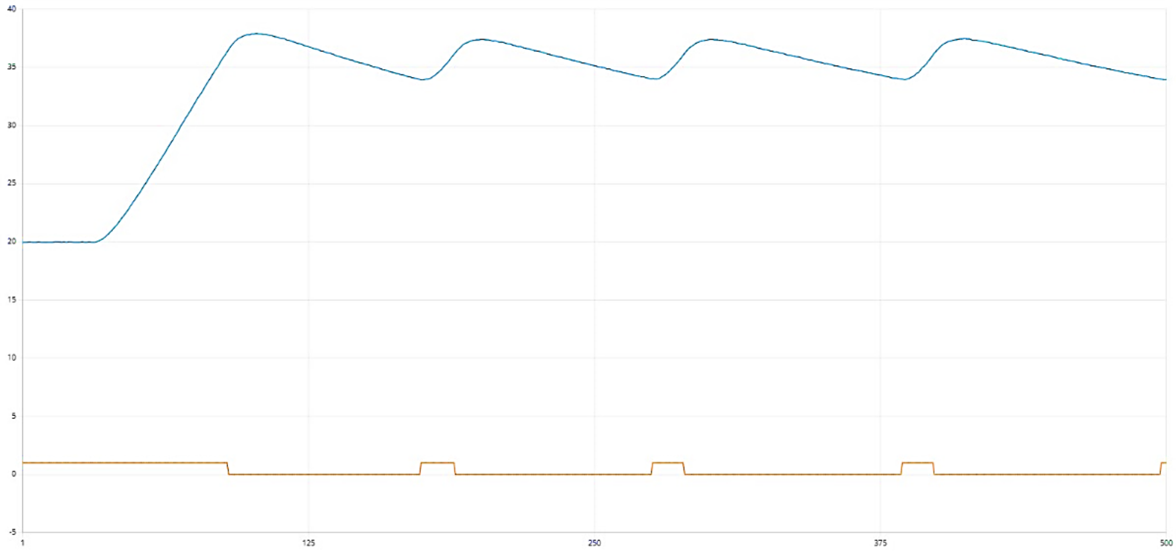

Figure 4.4.5. On-Off Control Operation

PID Control Operation: For a set value of 40°C, the temperature-time graph of the system is as shown in the figure. The system provides very precise control with very small losses of approximately -/+ 0.4°C. (The blue graph shows the temperature-time graph, and the orange graph shows the PID output value as a percentage.)

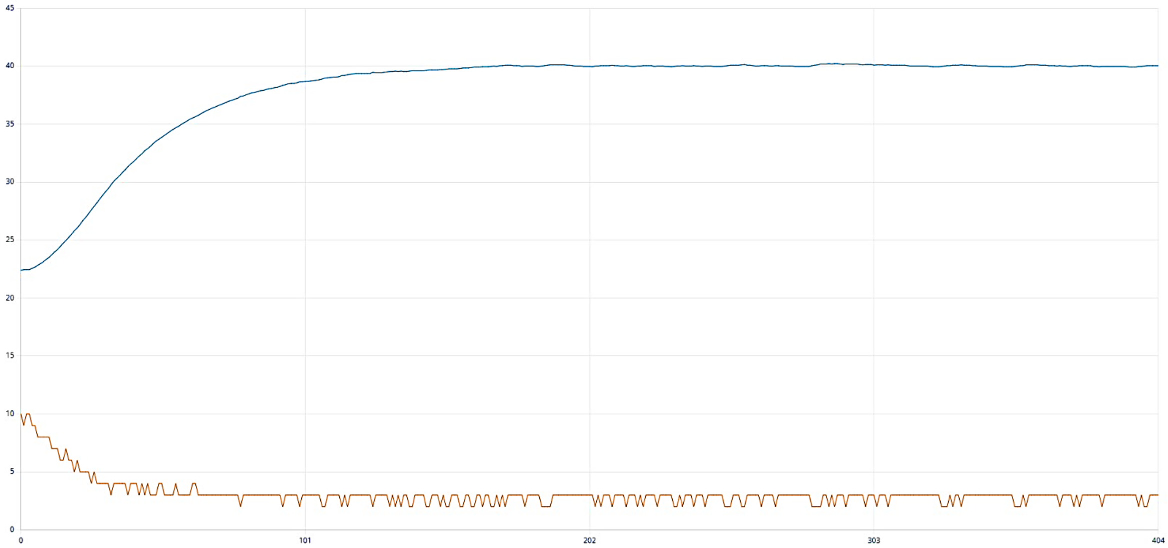

Figure 4.4.6. PID Control Operation

SlowPWM Control Operation: For a set value of 40°C, the temperature-time graph of the system is as shown in the figure. The system provides very precise control with very small losses of approximately -/+ 0.6°C. (The blue graph shows the temperature-time graph, and the orange graph shows the PID output value as a percentage.)

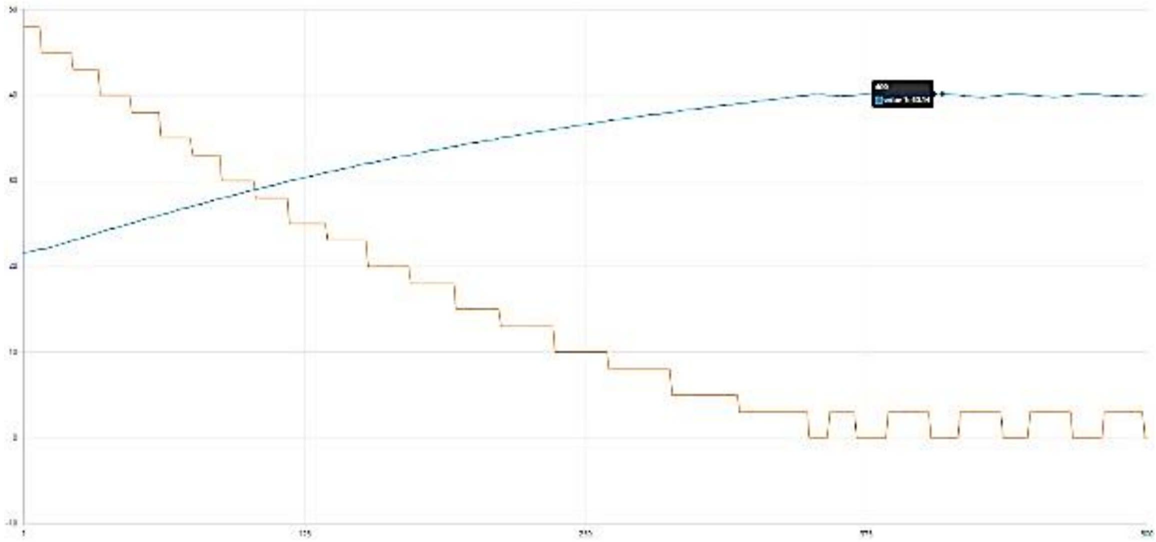

Figure 4.4.7. SlowPWM Control Operation

5. EXPERIMENTAL STUDIES

5.1. General Information

This section provides an overview of the experimental studies conducted during the project implementation phase. Detailed explanations are given regarding how the established system was realized, the challenges encountered, and the conveniences achieved. Additionally, safety precautions and necessary markings to be observed during practical use are discussed.

Since the system is designed for educational and laboratory environments, it was installed and prepared in compliance with safety standards. All critical connections, control elements, and hazardous points on the system are clearly marked to ensure users operate the system correctly and safely.

As part of the experimental studies, connection schematics, printed circuit board layouts, and photographs taken during operation are presented to demonstrate the physical configuration of the system to the reader.

5.2. System Setup and Integration

All hardware components used in the project were assembled compatibly to create the system prototype. The PT100 sensor was connected to the MAX31865 converter module, which was integrated with the Arduino Uno via the SPI communication protocol. The Arduino read the temperature values and controlled the heater through an SSR relay according to the software-implemented control algorithms. For the user interface, a 16x2 LCD display was connected to the Arduino using the I2C protocol, and push-buttons were appropriately positioned for menu navigation. All cables, power supplies, and signal lines were carefully arranged, and suitable connection diagrams were created.

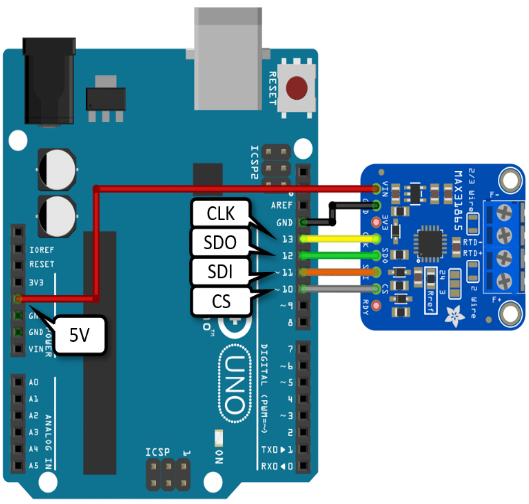

Figure 5.2.1. Arduino – MAX31865 Connection

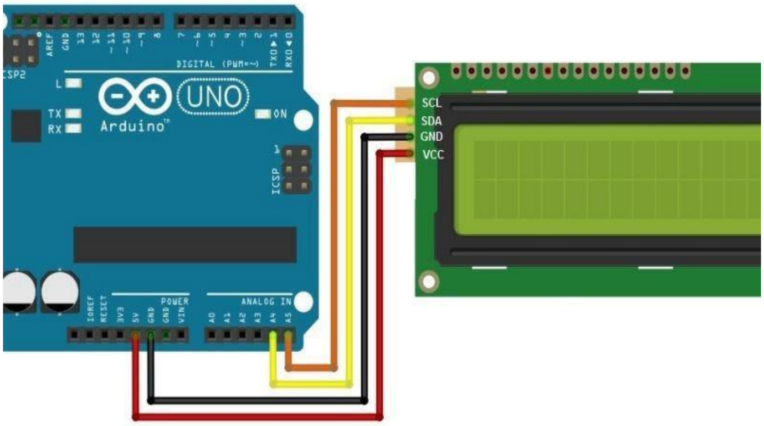

Figure 5.2.2. Arduino-LCD with I2C Connection

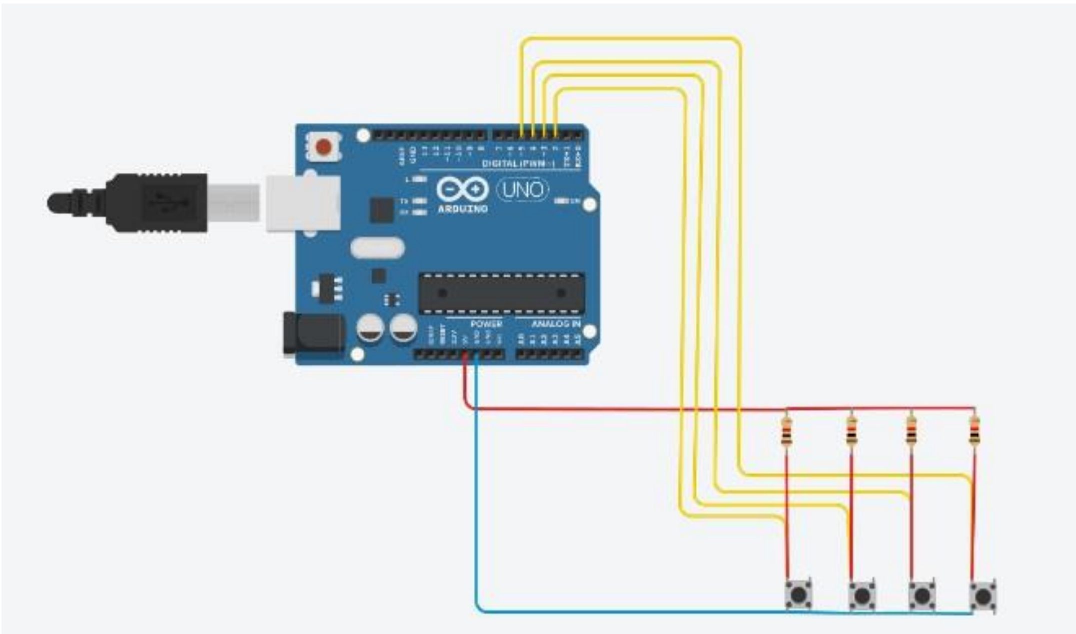

Figure 5.2.3 Arduino-Push Buttons Connection

5.3. Implementation of Interface Elements

A menu-based interface was designed to facilitate interactive system usage. Using buttons, the user was able to easily modify the temperature setpoint, PID parameters, and control modes.

The LCD screen provided real-time temperature information and displayed menu options, enabling continuous monitoring of the system status. The operational principles of the buttons, the use of digital input pins on the Arduino, and the function of pull-up resistors were explained to detail the technical foundation of the interface.

Figure 5.3.1. Device Main Screen

Figure 5.3.2. Control Function Menu

(PID Control, On-Off and SlowPWM Control are selected from this menu.)

5.4. Performed Tests

Various tests were performed to verify whether the designed and prototyped temperature control system functions in accordance with the intended operation. In these tests, the response of the system in different control modes (ON-OFF, PID, SlowPWM) was examined and performance evaluations were performed.

Figure 5.4.1 Set Temperature on PID Control

Figure 5.4.2. Set PID Coefficients (1)

Figure 5.4.3. Set PID Coefficients (2)

PID coefficients are set to optimum, but the desired value can still be entered here. (With increase-decrease buttons)

Figure 5.4.4. PID Control Operating Screen

After the desired values are set and saved, the device operation screen looks like this. The instantaneous temperature value and PID output appear as % (percentage) on the screen.

Figure 5.4.5. Set Temperature on On-Off Control

Figure 5.4.6. Set Hysteresis on On-Off Control

The set value temperature and Hysteresis for On-Off control are entered from the screen.

Figure 5.4.7. On-Off Control Operating Screen

After the desired values are set and saved, the device operation screen looks like this. The instantaneous temperature value and Relay status appear as “OPEN” or “CLSD” on the screen.

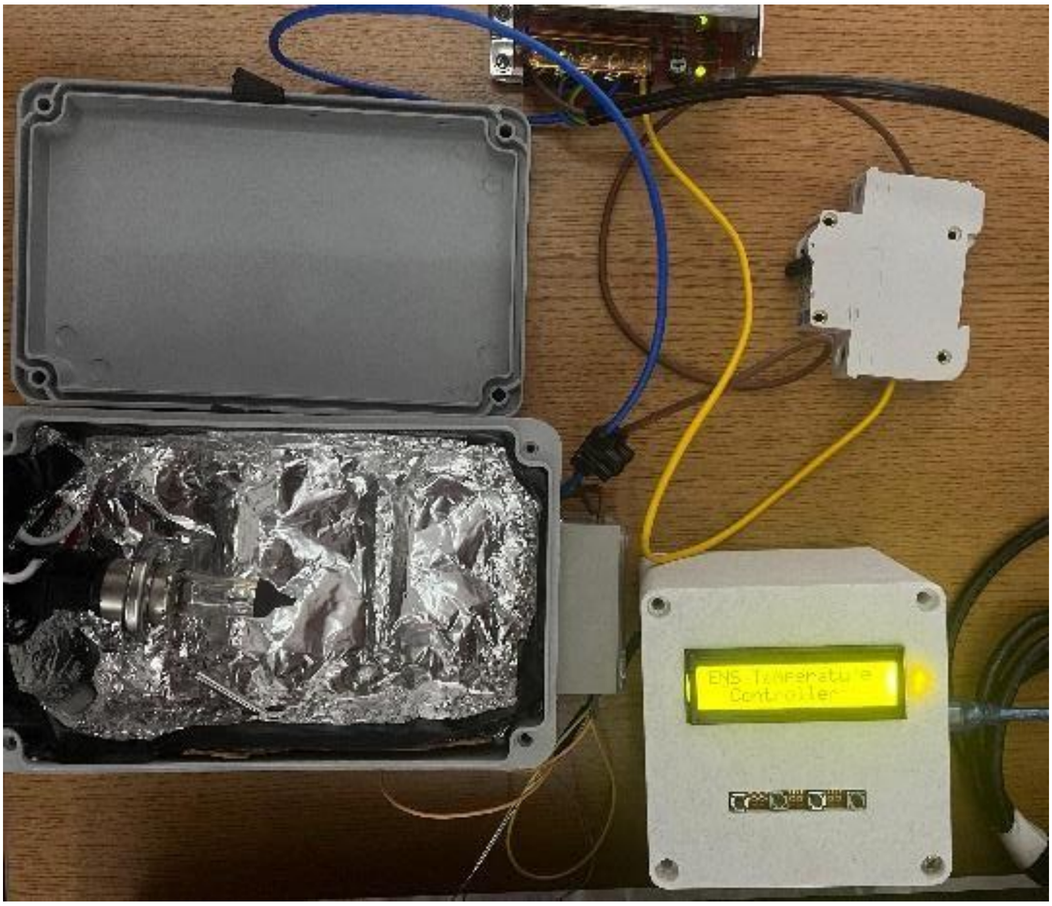

Figure 5.4.8. Experiment Setup-ON

Figure 5.4.9. Experiment Setup-OFF

Test Conditions and Considered Factors

Environmental Conditions: Tests were conducted in a laboratory environment with a stable and controlled ambient temperature. The influence of sudden external factors on test results was minimized.

Power Supply: All system components were powered by a 12V DC power supply and a 5V output provided by the Arduino. Voltage fluctuations in the power supply were stabilized prior to testing.

Sensor Calibration: Pre-calibration was performed to ensure the accurate operation of the PT100 sensor, and sensitivity settings of the MAX31865 module were checked.

Assumptions: For system accuracy, connections between sensors and actuators were assumed to be ideal. Electromagnetic interference and cable resistances were maintained at minimal levels during testing.

Safety Precautions: Electrical safety standards were implemented during testing, and system warnings for high temperature and short circuit conditions were monitored.

Test Methods and Connection Diagrams:

Tests were performed using predefined temperature setpoints. In each control mode, the system was operated until the setpoint was reached, with temperature variations and control outputs recorded.

The test system connection diagram illustrates all prototype components and communication lines. This diagram includes the sensor signal reaching the Arduino, the execution of control algorithms, and the transmission of commands to the heater.

6. RESULTS

6.1. General Remarks

This section evaluates whether the project has achieved its intended objectives based on numerical and graphical outputs. The obtained results reflect the success of the control algorithms in reaching the specified temperature setpoints as well as the overall system stability. The axes and units are clearly indicated in the graphs, enabling an objective analysis of the system’s behavior.

6.2. Simulation Results

Simulation studies were conducted in the MATLAB Simulink environment. The temperature–time graphs obtained from PID tuning and control algorithm simulations on the model show that the system reached the desired temperature quickly and stably. The simulation outputs also include performance comparisons among different control modes, highlighting the superiority of the PID control algorithm in parameters such as overshoot and settling time. These simulation results form the basis for the expected performance in experimental studies.

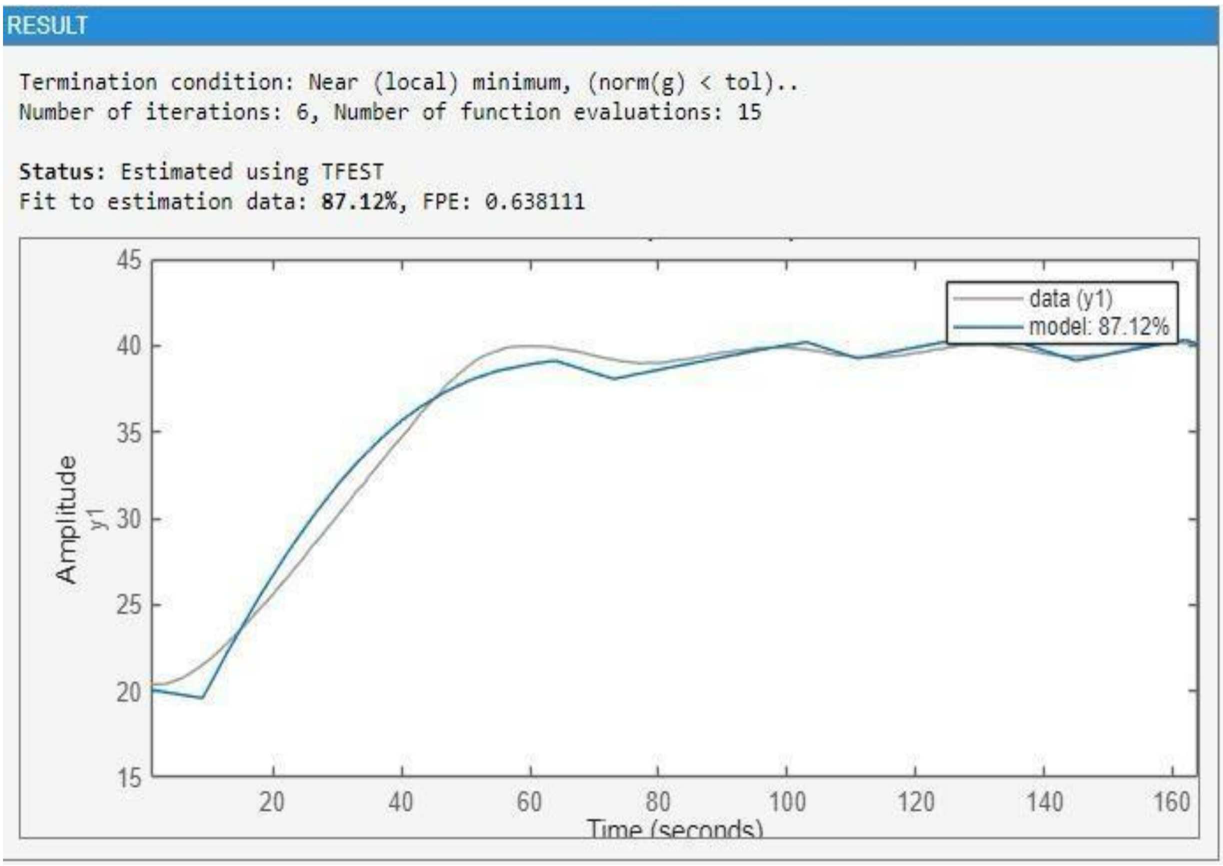

Figure 6.2.1. PID Transfer Function-Realtime Model Comparison

PID Control Simulation Results: (The black graph represents the experimental (measured) temperature–time curve, while the blue graph represents the theoretical (mathematically modeled) temperature–time curve.) The similarity between the simulation and experimental results is 87.12%.

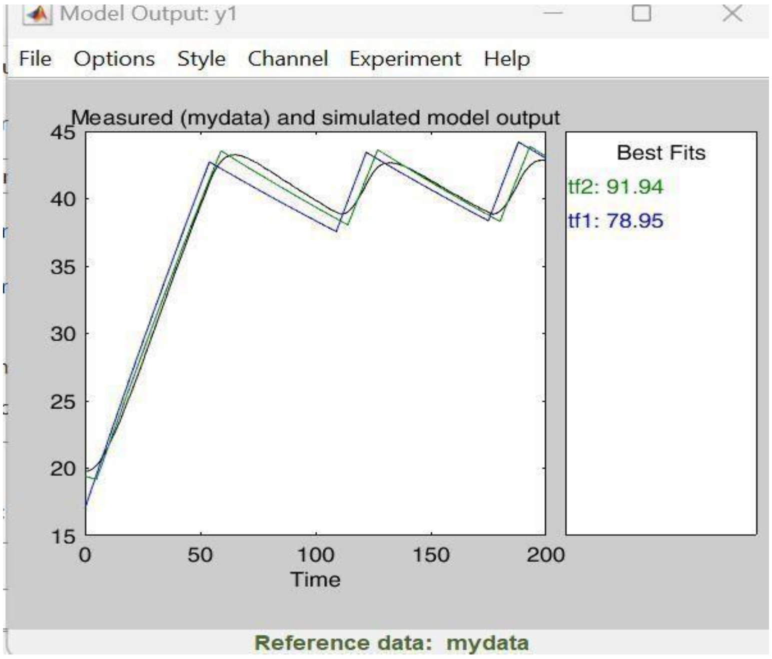

Figure 6.2.2. On-Off Transfer Function-Realtime Model Comparison

On-Off Control Simulation Results: (The black graph shows the experimental (measured) temperature-time graph, and the green graph shows the theoretical (mathematically modeled) temperature-time graph.) There is a 91.94% similarity rate between the simulation result and the experimental result.

6.3. Experimental Results

In tests conducted on real hardware, the system was tested at different temperature set points and the measured temperature-time graphs were compared with the simulation results. In the tests, the system reached the target temperature values within the specified tolerances and maintained these values stably.

The small differences between the experimental and simulation results were due to sensor sensitivity, environmental effects and tolerances of the hardware components. These differences show that the system works successfully under real-world conditions.

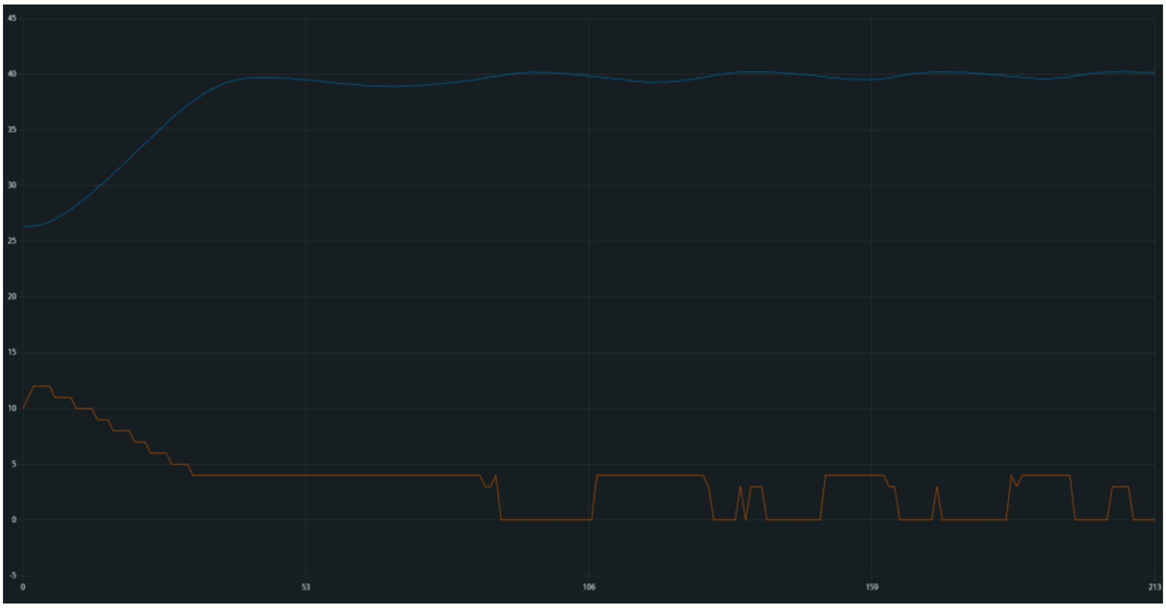

Figure 6.3.1. PID Control Experiment Results

PID Control Experiment Results (Blue graph shows the temperature-time graph, orange graph shows the PID output value as a percentage.) The system provides very precise control with very small losses of approximately -/+ 0.4°C.

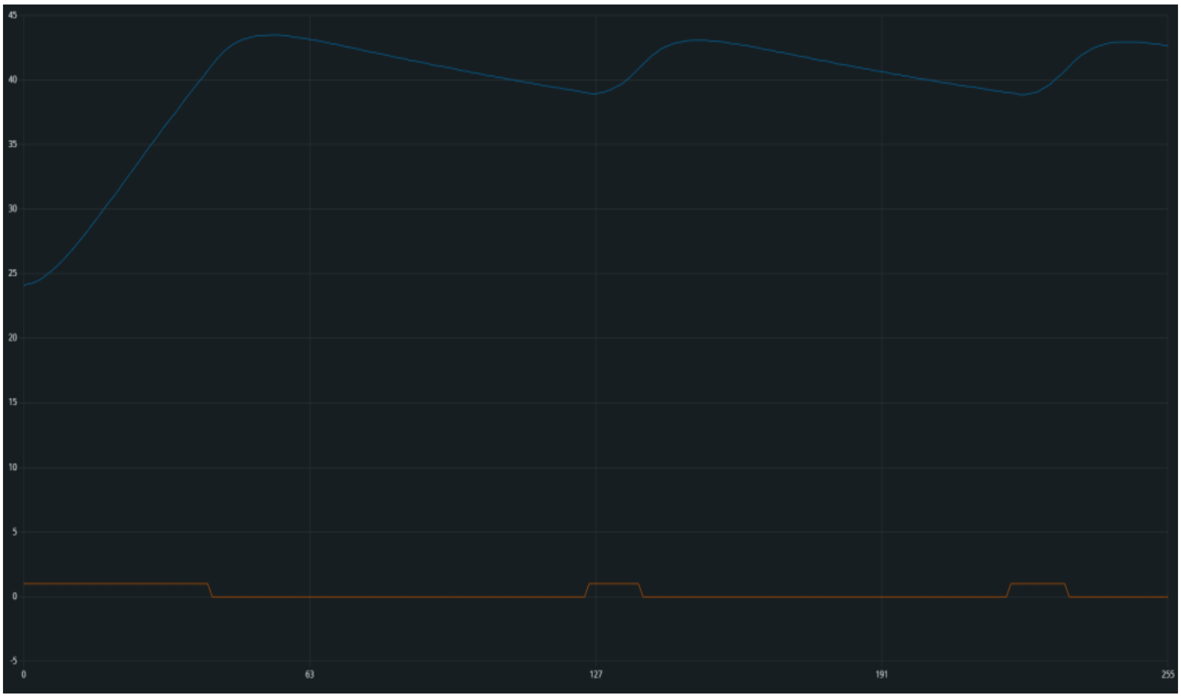

Figure 6.3.2. On-Off Control Experiment Results

On-Off Control Experiment Results (Blue graph shows the temperature-time graph, orange graph shows the relay on or off (on-off status as 1-0).) The system provides precise control with a loss of approximately 1.2°C (excluding the hysteresis value).

7. CONCLUSIONS

The temperature control system prototype developed in this study has provided significant benefits, particularly in the field of education. The core objective of the project to practically experience the theoretical concepts of control algorithms was achieved through a modular and user-friendly design. Offering different control methods such as PID, ON-OFF, and SlowPWM on a single platform has enhanced flexibility and scope in educational processes.

The prototype addresses the lack of practical applications commonly encountered in industrial automation and control systems education, thereby improving students' comprehension and application skills. Consequently, the quality of engineering education is elevated, and graduates' adaptation to the industry is facilitated.

The primary issues addressed by this work include simplifying complex control algorithms, filling the gap in hands-on educational materials, and developing a low-cost hardware platform. Additionally, the user-friendly interface has facilitated practical parameter adjustments.

The target users of the prototype include engineering faculties of universities, vocational schools, and technical high schools. Moreover, educational institutions and research centers requiring control systems training are also potential users.

Some discrepancies were observed between the projected budget during the design phase and the actual costs incurred during implementation. These differences arose from market price fluctuations, additional safety measures, and labor costs during production. However, overall costs remained within planned limits, and the project proved economically sustainable.

Future improvements may include integration of wireless communication, incorporation of different sensor types, use of advanced PID algorithms, and compliance with industrial standards. Furthermore, commercialization of the prototype and its wider dissemination to educational institutions are also planned objectives.

8. REFERENCES

[1]. Yılmaz, h., kaya, m., öztürk, e., "PID kontrol uygulaması ile sıcaklık regülasyonu", ulusal otomatik kontrol sempozyumu, 2020, ss. 45-52.

[2]. Zhao, l., chen, w., sun, m., "IoT based PID temperature control system", ieee international conference on control and automation, 2019, beijing, china, ss. 112-117.

[3]. Aras, f., demirtaş, a., "Eğitim amaçlı düşük maliyetli PID kiti tasarımı", tÜbİtak turkish journal of electrical engineering & computer sciences, cilt 26, sayı 3, 2018, ss. 1234-1245.

[4]. Singh, r., patel, k., sharma, v., "Simulation of temperature control using PID in embedded systems", international journal of scientific research in engineering and management, cilt 6, sayı 8, 2020, ss. 556-563.

[5]. Güzel, a., "Endüstriyel sıcaklık kontrol sistemlerinde PID ve bulanık mantık kıyaslaması", yüksek lisans tezi, ostim teknik üniversitesi, elektrik elektronik mühendisliği bölümü, 2021.

[6]. Yan, z., jiang, h., zhang, z., liu, b., "Application of a self-tuning PID controller in temperature control system", ieee 4th advanced information technology, electronic and automation control conference (IAEAC), 2019, shanghai, china, ss. 234-239.

[7]. Gräber, m., "Practical PID tuning guide", tlk energy, 16 mart 2021, https://tlk-energy.de/blog-en/practical-pid-tuning-guide, erişim tarihi: 2025.

[8]. Kristiansson, b., "Evaluation and simple tuning of PID controllers with high-frequency robustness", journal of process control, cilt 15, sayı 6, 2005, ss. 587-598, https://doi.org/10.1016/j.jprocont.2005.02.002.

[9]. Lenfromtoronto, "PID temperature control of a small thermal chamber", arduino project hub, https://projecthub.arduino.cc/lenfromtoronto/pid-temperature-control-of-a-small- thermal-chamber-df6567, erişim tarihi: 2025.

[10]. Arduino Forum, "PWM using millis()", https://forum.arduino.cc/t/pwm-using-millis/570432, erişim tarihi: 2025.

[11]. MATLAB, "System Identification Toolbox Documentation", the mathworks inc., https://www.mathworks.com/help/ident/, erişim tarihi: 2025.

9. APPENDIX

APPENDIX-1 IEEE Etik Kuralları (Türkçe) ve IEEE Code of Ethics (İngilizce)

IEEE Code of Ethics

We, the members of the IEEE, in recognition of the importance of our technologies in affecting the quality of life throughout the world, and in accepting a personal obligation to our profession, its members, and the communities we serve, do hereby commit ourselves to the highest ethical and professional conduct and agree:

I. To uphold the highest standards of integrity, responsible behavior, and ethical conduct in professional activities.

1. to hold paramount, the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, to protect the privacy of others, and to disclose promptly factors that might endanger the public or the environment;
2. to improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems;
3. to avoid real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist;
4. to avoid unlawful conduct in professional activities, and to reject bribery in all its forms;
5. to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, to be honest, and realistic in stating claims or estimates based on available data, and to credit properly the contributions of others;
6. to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;

II. To treat all persons fairly and with respect, to avoid harassment or discrimination, and to avoid injuring others.

7. to treat all persons fairly and with respect, and to not engage in discrimination based on characteristics such as race, religion, gender, disability, age, national origin, sexual orientation, gender identity, or gender expression;
8. to not engage in harassment of any kind, including sexual harassment or bullying behavior;
9. to avoid injuring others, their property, reputation, or employment by false or malicious actions, rumors, or any other verbal or physical abuses;

III. To strive to ensure this code is upheld by colleagues and co-workers.

10. to support colleagues and co-workers in following this code of ethics, to strive to ensure the code is upheld, and to not retaliate against individuals reporting a violation.

Adopted by the IEEE Board of Directors June 2020

IEEE Advancing Technology for Humanity

www.ieee.org

APPENDIX-2. Interdisciplinary Study

The theoretical foundation for the control algorithms applied in this project was established through the Control Systems and Intelligent Control Systems courses taken during my undergraduate education. These courses provided in-depth knowledge on topics such as system modeling, feedback mechanisms, sensor integration, control system block diagrams, and fundamental control algorithms.

In particular, the ON-OFF control with hysteresis and the PID (Proportional-Integral-Derivative) control algorithm implemented in this project are directly based on the concepts covered in these courses. The control structure, relying on temperature feedback, ensured that the system remained stable within a defined temperature range.

Furthermore, system modeling was carried out using the transfer function approach, and a mathematical representation of the physical system was developed in MATLAB Simulink. Based on the identified model, PID parameters were optimized, allowing for a close match between the theoretical model and the actual system behavior.

This project serves as a concrete example of how control theory, learned throughout the engineering education, can be applied in an interdisciplinary system design that combines both theoretical understanding and practical implementation.