

# Smart Heater Control System

Embedded Systems Intern Assignment – upliance.ai

Author: Rajashekar Miryala

Platform: ESP32 on Wokwi

Date: August 2025

---

## 1. Project Overview

This project demonstrates a Smart Heater Control System built using the ESP32 microcontroller. The system simulates a temperature-controlled heating unit using an LM35 analog sensor, an LED as the heater, and a buzzer for overhear alerts. The simulation is built and tested using the Wokwi online simulator.

---

## 2. Hardware Components

ESP32 DevKit V1

LM35 Temperature Sensor (simulated using a potentiometer)

LED (represents the heater)

Buzzer (used for overheating alert)

220-ohm Resistor

Jumper Wires

---

3. Circuit Connections

Component	ESP32 Pin	Description
LM35 (VCC) / Potentiometer VCC	3.3V	Power Supply
LM35 (GND) / Potentiometer GND	GND	Ground
LM35 (OUT) / Potentiometer SIG	GPIO 34	Analog Input
LED Anode (+) GPIO 23		Heater Control Output
LED Cathode (–)	GND via 220Ω resistor	Current Limiting
Buzzer (+)	GPIO 22	Overheat Alert
Buzzer (–)	GND	Ground

Note: Power and ground connections are made independently to avoid signal merging.

---

4. System Behavior

The control system is designed as a state machine that switches states based on current temperature readings:

Temperature Range	System State	Heater Status	Buzzer Status
-------------------	--------------	---------------	---------------

< 25°C	IDLE	OFF	OFF
--------	------	-----	-----

25°C – 30°C	HEATING	ON	OFF
-------------	---------	----	-----

30°C – 44°C	STABILIZING	ON	OFF
-------------	-------------	----	-----

~45°C (±1°C margin)	TARGET	OFF	OFF
---------------------	--------	-----	-----

> 60°C	OVERHEAT	OFF	ON
--------	----------	-----	----

---

## 5. Code Summary

```
#define TEMP_SENSOR_PIN 34
```

```
#define HEATER_PIN 23
```

```
#define BUZZER_PIN 22
```

```
void setup() {
```

```
    Serial.begin(115200);
```

```
    pinMode(HEATER_PIN, OUTPUT);
```

```
    pinMode(BUZZER_PIN, OUTPUT);
```

```
    digitalWrite(HEATER_PIN, LOW);
```

```
    digitalWrite(BUZZER_PIN, LOW);
```

```
}
```

```

void loop() {
    float temperature = readTemperature();
    HeaterState nextState;

    if (temperature < 25) nextState = IDLE;
    else if (temperature < 30) nextState = HEATING;
    else if (temperature < 44) nextState = STABILIZING;
    else if (temperature < 60)
        nextState = (abs(temperature - 45) <= 1.0) ? TARGET : STABILIZING;
    else nextState = OVERHEAT;

    updateHeaterOutput(nextState);
    printSystemStatus(temperature, nextState);
    delay(1000);
}

```

---

## 6. Summary

This Smart Heater Control System project showcases fundamental embedded concepts such as analog data reading, GPIO control, buzzer/LED interfacing, and implementing a state machine in real-time applications. It simulates an industrial heating application with overheat protection and state feedback via serial monitor.

---

## 7. Submission Info

Name: Rajashekar Miryala

Internship Role: Embedded Systems Intern

Submitted To: upliance.ai

Submission Date: August 2025