Sothern New Hampshire University

CS 350 Module seven final Project Thermostat Lab

# Smart Thermostat System Using Raspberry Pi

## Objective

The primary aim of this module was to develop a smart thermostat system based on Raspberry Pi with other real-time components such as the LCD display, LED indicators, input buttons, and temperature and humidity sensor (AHT20). The solution is a basic heating-cooling controller, enhanced with visual feedback and user interaction by a series of buttons.

## System Overview

The whole thermostat setup is designed using the Raspberry Pi and a 16x2 character LCD which can display date, time, temperature, and system state. Heating is represented with a red LED, while a blue LED is used to indicate cooling. The three push-button switches allow the user to toggle system modes (OFF, HEAT, COOL) and set the set point temperature. The sensors used to record the room ambient temperature at the startup are called AHT20; this temperature will be used as a reference throughout the operation of the system.

A circuit board with wires and other wires

AI-generated content may be incorrect.

A circuit board with wires

AI-generated content may be incorrect.

## Technical Implementation

Key components and features:

* AHT20 Sensor (I2C): Reads room temperature straight away when the system is initialized.
* System States: Controlled through a button (GPIO 21) with OFF, HEAT, and COOL modes.
* Temperature Control: Two buttons (GPIO 22 and GPIO 27) can be used for simulating room temperature (for testing purposes).

LED Feedback:

* HEAT mode: the Red LED is turned on but will fade as room temperature is lower than the set point, becoming solid once above or equal to it.
* COOL mode: the Blue LED behaves similarly, indicating that it is cooling.

LCD display: First row displays real-time clock while the second-row toggles between temperature and system state with the set point.

UART logging: Logs the system state room temperature and set point every 30 seconds for the monitoring purposes.

## Reflections

This project entailed a lot of direct engagement in combining digital and PWM inputs and outputs, time-basis logic, and I2C communications through Python with Raspberry Pi.

Using real-time conditions (such as reading sensors and simulating heating/cooling states) reinforced state machine concepts and embedded control logic. Interaction with buttons and LED transitions greatly aided in visualizing system behavior. Furthermore, modular function usage alongside condition-based logic is good programming practice for embedded systems.

## Conclusion

The purpose of this module is the successful demonstration of sensor-driven control systems using Raspberry Pi, making a basic thermostat functionality in an embedded system. Future developments may include implementation of continuous temperature readout, decision based on humidity, and IoT access from anywhere.

## 

## References

<https://forums.raspberrypi.com/viewtopic.php?t=191336>

<https://forums.raspberrypi.com/viewtopic.php?t=359930>

https://ve2zaz.net/RasTherm/RasTherm.htm