Product Design Specification

Team 3: Kai Han, Luke Hoskam, Luis Nadora, Zheng Zhang
Professor Greenberg
ECE411
10/20/2021

Executive Summary / Concept of Operations

Many apartment dwellers would like to have smart devices in their house but cannot because of the relative permeance of some devices. A heater in an apartment, for instance, may only present a user with a dial style thermostat knob to set an abstract temperature. This would leave no option to attach a smart thermostat. This would force an energy conscious renter to adjust the knob manually every time he left his apartment and when he returned.

Our device aims to fix that by attaching non-destructively to the thermostat knob and allow the user to program temperature settings though their local intranet.So people who are limited by the relative permeance of some devices but also would like to make the heater smarter will use it. The device will host a small webpage to allow the user to program either a constant temperature or set on-off times, for instance, automatically turn off when the user is at work.

Basically, this device turns your standard knob controlled heater into a programmable thermostat without having to remove or damage parts of the building or equipment owned by the landlord.

Brief Market Analysis

The intended customers are apartment dwellers who would like a smart heater but already had a dumb heater and renters who have a limited budget but want to maintain a constant temperature better the knob is capable. The device will non-destructively attach to the already installed heater knob and manipulate its inputs.

For now, our competitors are mainly manufacturers with complete replacement heating systems. If landlords replace the heater system with a smart heater, it will affect our sales of our project. The best aspect of our project is that we do not need to replace the original heater (which is not the property of the tenant), which can reduce waste. Replacing the heater will not only damage the walls or facilities in the original apartment, but also create a lot more waste. Thus, we will create a device that allows people to non-destructively convert an existing heater with a thermostat knob to a smart heater. The networked heater will also be controlled ‘smartly’ on the intranet. It not only saves electricity but also makes it easier and more fun to control the heater.

Another type of competition is portable heaters, which are controlled/programmed by a smartphone app.

The price range for the permanently installed and portable smart devices is anywhere from $60-$250 and we are positive that we can build and sell our device within that price range.

Requirements

Must requirements:

- Be lightweight
- Non-destructively attach to a heater knob
- Interface with the device through website
- Allow users to program on-off times through the website.
- Have a case
- Have a manufactured PCB

Should requirements:

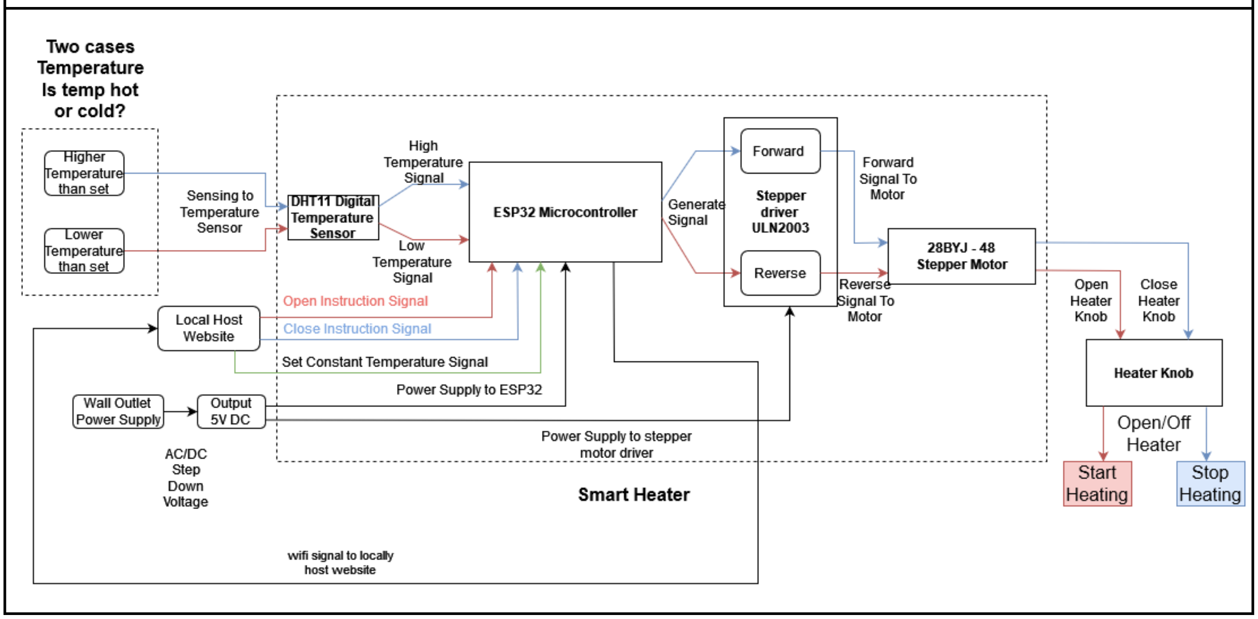
- Powered by a battery
- Have an attractively designed case.
- Interface with buttons to control the motor.

May requirements:

- Realize the internet to control the temperature.
- Connect wirelessly to auxiliary thermostats to maintain an average set temperature.
- Have a display

System Architecture

Level 1 Block Diagram for the Smart Heater

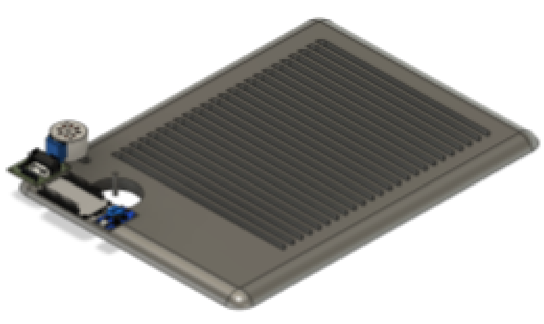


Design Specification

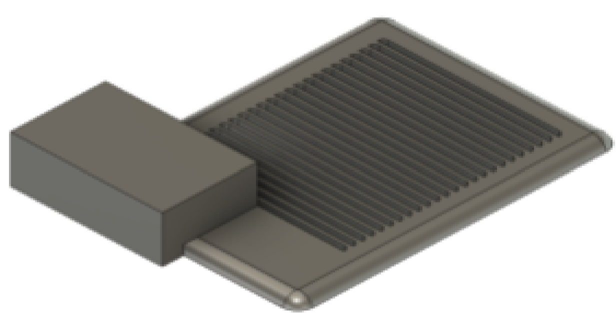
- Processor: ESP Wroom 32 Devkit V1. 4
  - The ESP Wroom 32 is a low-cost development board with a Dual-core CPU Xtensa LX6 clocked at 250MHz. It has: 25 digital I/O pins, 6 Analog inputs, and 2 digital-to-analog output pins, and 4MB of storage.
  - Our project will use the ESP32 to process the signal from the DHT11 and, when appropriate, signal the stepper driver to manipulate the heater knob.

- Sensor: Temperature Sensor (DHT11) x 1.
  - The DHT11 is a basic, ultra low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and outputs a digital signal on the data pin (no analog input pins needed).
  - The project uses DHT11 sensors to detect the room temperature and then compare that temperature to the set temperature on the ESP32.
- Actuator: Stepper motor(28BYJ - 48) and Driver (ULN2003)x 1.
  - Powered with 5v from the AC-DC wall adapter, the ULN2003 is a 16-pin IC. It has seven Darlington Pairs inside, where each can drive loads up to 50V and 500mA.
  - 28BYJ-48 is a small, lightweight, unipolar, 5V stepper motor that inputs alternating pulse signals and rotates by converting these input signals into mechanical rotation by selectively turning on and off specific electromagnets.
  - Our project will have 5v DC to the ULN2003 common pin. Then, when the ESP32 sends an HIGH signal to a particular pin on the ULN 2003 will output a clean 5v to the stepper motor. The stepper motor will connect to the heater knob through gears to turn the heater on and off.
- Power: E178074 AC-DC wall adapter.
  - Input: 100-240AC,
  - Output: 5v 1A wall adapter.
- Mechanical Design: Designed in Fusion 360 and printed using a 3D Printer.
  - The software we will use to design the gears and the case will be Fusion 360. We will print the parts on one of three printers owned by group members.
  - The case will include an opening for the DHT11 to extend outside the case. The circuit board, motor, gears and microcontroller will be inside the case.
- Development environment: Visual Studio Code, PlatformIO and GIT.
  - We will use Visual Studio Code with the PlatformIO extension to program the ESP32.
  - The group will stay connected and up to date with the project using GIT through github.com (https://github.com/lnadora/ECE411-Team3)

<u>Primitive CAD drawing</u>



Simple design idea of placement of the parts



Simple design of how the case will fit on the heater