The BoserBeater: Plan & Milestones

Team Members

Name	SID
Charles Lin	26010321
Kenny Tan	3032225730

Abstract & Features

Egg beating is used in many baking processes. However, this is quite time-consuming and requires someone to manually monitor the process. Thus, the goal of this project is to create an automatic egg beater that is wirelessly controlled. This would free up the baker for other activities, thus cutting time from the overall baking process. In order to accomplish this, the project whisk consist of a whisk attached to a stepper motor via a mount. The motor will then be activated by an ESP32 microcontroller with preset settings. These settings will be sent wirelessly from a computer, allowing us to apply the knowledge that we acquired from EE49 to a real life useful application.

MQTT is to be used for sending commands with potential for upgrade to Bluetooth for increased security.

Sensing

- Temperature readout is given by a thermocouple that senses temperature

Actuation

- Whisk is being actuated by stepper motor

Computing

- Python program on computer that accepts inputs for settings/presets
- MQTT communication to send commands to eggbeater
- Python program on ESP32 that receives MQTT command and drives stepper motor
- Timer

Wireless Communication

- MQTT
    - Computer sends command to ESP32
    - ESP32 communicates task progress and temperature reading to computer

Parts List – Bill of Materials

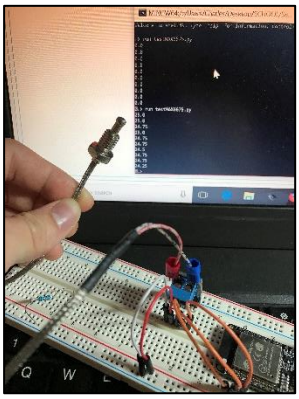
Part	In Lab Kit?	If not, source
ESP32	yes	
DC motor	no	Car kit
Thermocouple sensor: MAX6675	no	Amazon
Motor mount (Laser cut)	no	Jacobs
Whisk mount (3D printed)	no	Jacobs
Whisk	no	Daiso
Power plug & USB cable	no	Amazon

Milestone 1

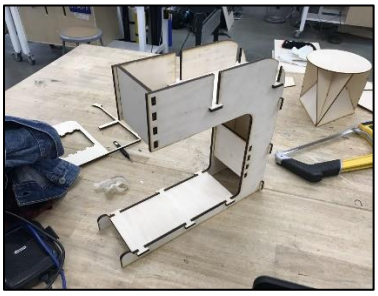
- Objectives
  - Motor Actuation
    - Write Python program to recognize command terminal inputs and settings
    - Establish connectivity using MQTT and command ESP32 in response to the input
    - Write Python program to actuate motor based on input received over MQTT
    - Drive the stepper motor
    - **Put it all together: Use input from computer to wirelessly start, run, and stop the stepper motor.**
  - Temperature Sensing
    - Confirm temperature readings from thermocouple
    - Establish connectivity using MQTT
    - **Put it all together: Communicate temperature readings to computer terminal using MQTT.**
- Accomplishments
  - Materials all purchased
  - Wireless Motor Actuation
    - Wiring diagram is complete
    - boot.py is complete: connects ESP32 to internet
    - main.py is complete: motor is driven and can be controlled by MQTT communication
    - Circuit is wall-powered
  - Mechanical CAD
    - Motor and whisk mount CAD is complete and ready for fabrication
  - Motor Validation
    - We tested the motor speeds and they are sufficient at a planned supply voltage of 5 V. Speed change will occur by changing the PWM.
- Discrepancies
  - Instead of getting temperature sensing capability, we did the mechanical CAD instead. Additionally, we validated the motor this week instead of next week because the motor dimensions drive the motor mount and whisk mount design.
- Plan Changes
  - For Milestone 2, assemble the mechanical CAD. Get the thermocouple working.

Milestone 2

- Objectives
    - Mechanical Design & Assembly
    - Create CAD designs for motor mount and whisk mount.
        - Fabricate mounts by 3D printing using Jacobs 3D printers.
        - **Assemble the entire project.**
    - Motor Validation
        - **Confirm if current stepper motor is sufficient for rotating whisk.**
        - If not, look into more powerful motors and update mount designs as necessary.
- Accomplishments
    - Electrical



        - Wiring diagram for thermocouple is complete
        - Temperature sensing using MAX6675 is operational
    - Mechanical




        - Motor mount fully fabricated and assembled
        - Whisk mount prototype is 3D printed
- Discrepancies
    - Need to continue to refine physical design
        - Whisk mount larger than expected
        - Need to find suitable glue to combine whisk and 3D printed mount
    -
- Plan Changes
    - Continue refining the software functionality and wireless transmission of temperature values

Final Project Presentation and Demo

We are planning on presenting a fully-functional, wall-powered automatic eggbeater.

The demo will consist of mixing Thai tea. Thai tea consists of the brewed tea, sugar, regular milk, and condensed milk. We will prepare the brewed tea. Sugar, regular milk, and condensed milk will be added during the demo and whisked to create Thai tea. We will be mixing a drink instead of raw eggs because we can immediately consume the drink as part of the demo.

The eggbeater will be controlled via computer terminal inputs into a Python program which can send commands.

Demo Overview

We made some changes to our original proposed demo. To simplify the Thai tea mixing, we prepared brewed tea with sugar and a separate mixture of milk and condensed milk. Additionally, instead of controlling the eggbeater using a Python program, we used MQTT services on HobbyQuaker as introduced in class. Finally, instead of using the IoT Eclipse broker as introduced in class, we used the HiveMQ Public Broker for internet consistency.

Overall, the demo went as planned and Thai tea was mixed while we gave the presentation. We successfully served the results of our demo to professors, GSIs, and fellow students.