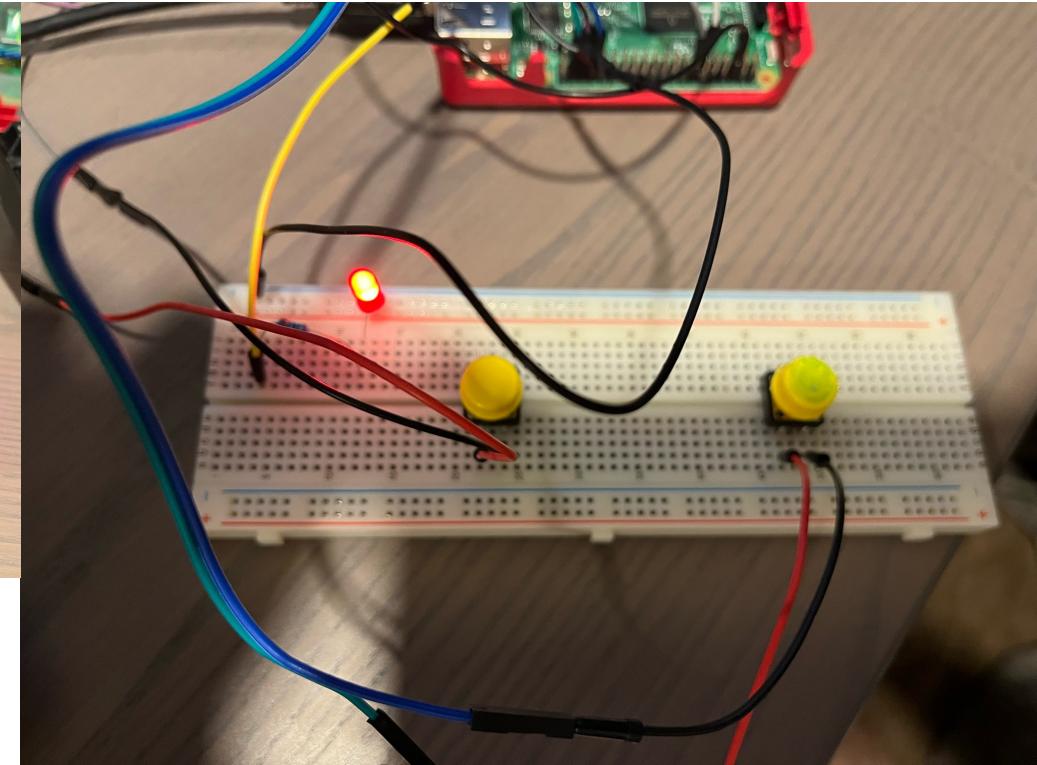
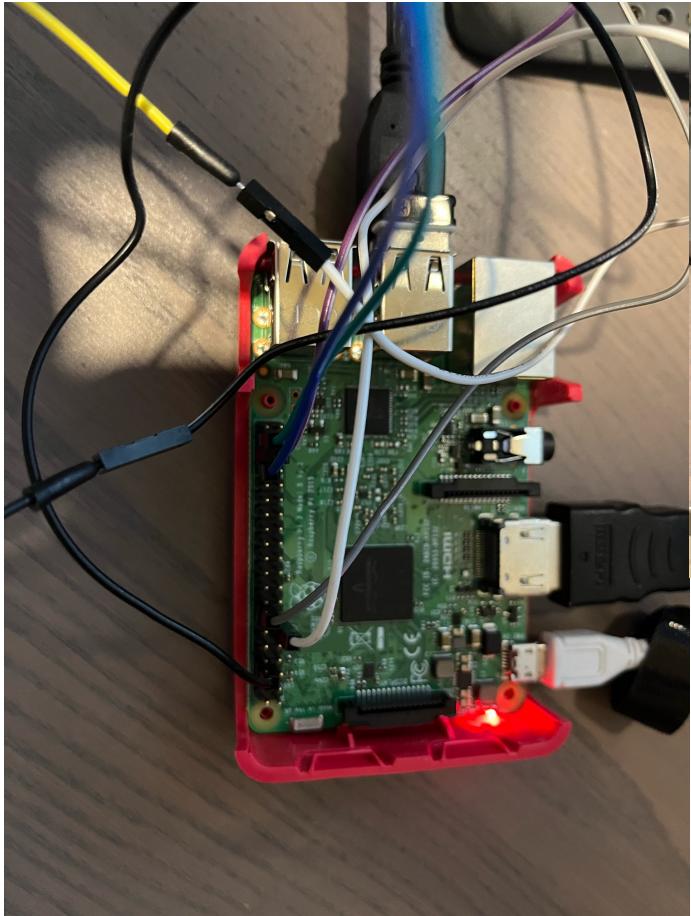


Raspberry Pi Alarm Clock

By: Codie Tamida

Hardware



Software

- Python
- Raspberry Pi OS
- Bluetooth

How it works

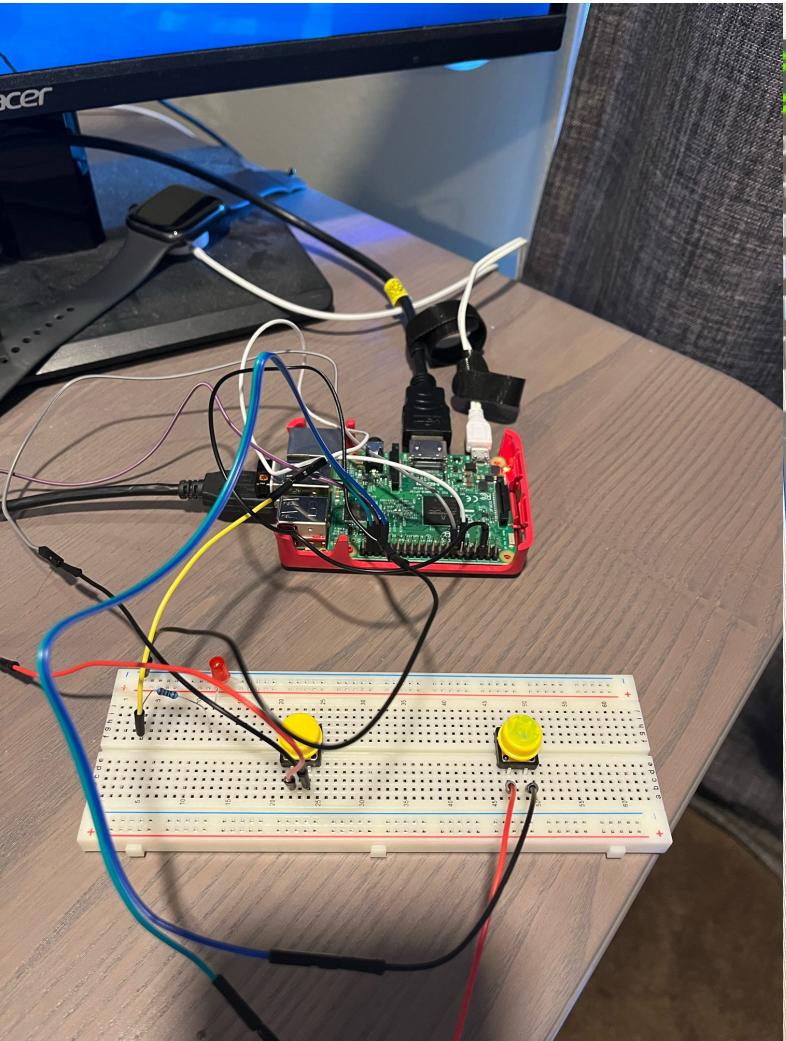
- I was able to connect two buttons on the breadboard to the Pi GPIO pins and when clicked they perform different functions. A light was added using a transistor and breadboard/GPIO pin connection was added for fun
- Through an external monitor I was able to connect the Pi to a Bluetooth speaker
- The Alarm has to be manually set with in the script using:

```
53  
54     scheduled_time = datetime.now().replace(hour =22, minute = 40, second = 20, microsecond = 0)  
55
```

How it Works

- Once the time matches the scheduled time in the python script, a song will play from the Bluetooth speaker
- If the user presses the button on the left (yellow) then the alarm will be snoozed for 15 seconds (Adjustable by the user within the script)
- If the user presses the button on the right (Slightly blue) then the alarm will stop completely

Additional Pictures



```
File Edit Tabs Help
sodie@raspberrypi:~ $ cd Desktop/RaspberryPi-alarm
sodie@raspberrypi:~/Desktop/RaspberryPi-alarm $ python3 button.py
pygame 2.1.2 (SDL 2.26.5, Python 3.11.2)
Hello from the pygame community. https://www.pygame.org/contribute.html
The Alarm is Snoozed
The Alarm is Snoozed
The Alarm is Snoozed
The alarm has been shut off
```

A screenshot of a terminal window titled "raspberrypi-alarm". The window shows the command `python3 button.py` being run. The output of the script is displayed, indicating the alarm status: "The Alarm is Snoozed" three times, followed by "The alarm has been shut off". The terminal is part of a desktop environment, with a blurred background showing a person's hand and a monitor.