

Write-Up

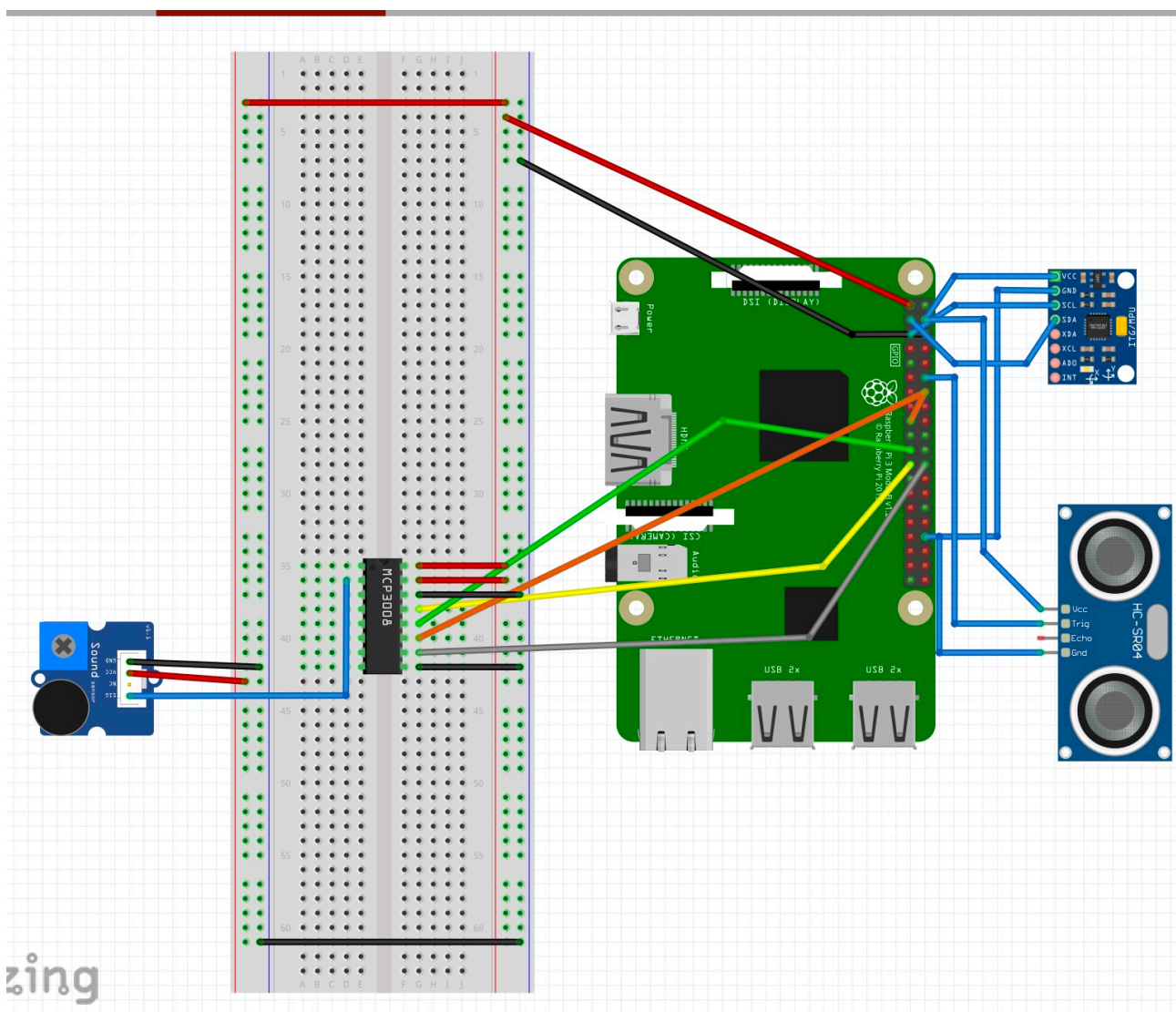
Team Member: Linus Lei, Way Zheng

Project Name: RiderGuard: Intelligent Crash Detection System for bicycles and scooters

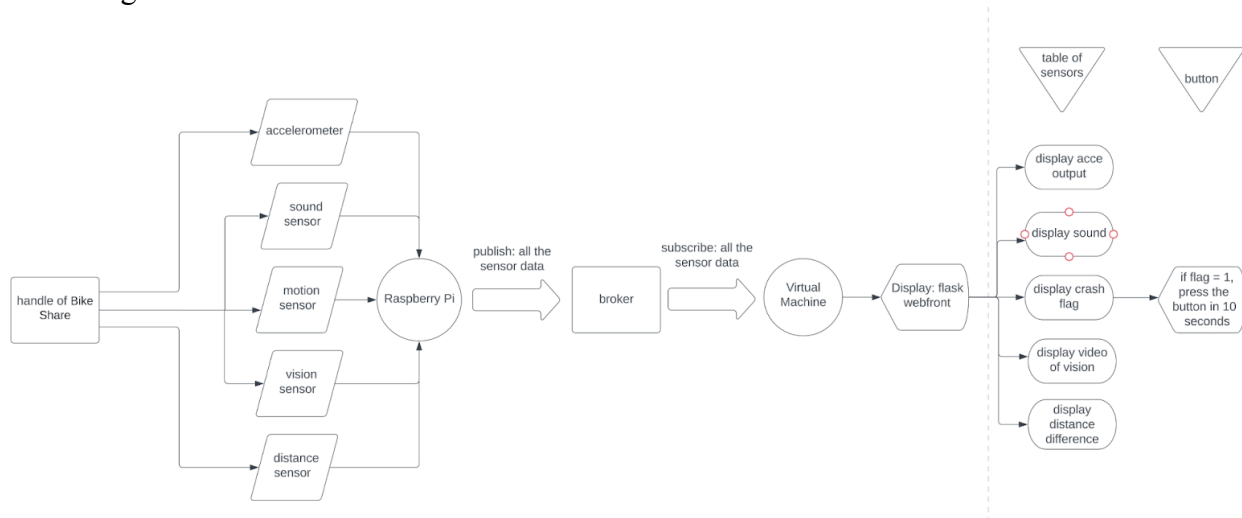
Project Description:

This project is dedicated to making a Crash Detection System based on MPU6050 Accelerometer and Raspberry Pi Sound Sensor. A crash is detected when both sensors detects a high value in reading. A Raspberry Pi camera module is used to record the accident after being triggered to preserve evidence. When a crash is detected, based on the readings from the two sensors, as well as change in distance from HC-SR04 Ultrasonic sensor and the Weather API to give out a severity score that indicates how urgent would help be needed, as well as a confidence score indicating to what degree that this is an actual accident. It also initiates a prompt system that allows the user to choose whether they need help, if no response is made in 10 seconds, it will automatically define this as an actual accident and needs urgent help. In this case, a message containing both score will be sent to host machine using MQTT server.

Hardware Building Instructions:



Block Diagram:



Limitations:

-Precision: The current triggering algorithm is based on a preset absolute value that is not scientific. Hence it could miss or falsely trigger many accidents. In order to improve, ML techniques that bases on many real life accidents could be used to solve this issue

-Video Recording: Due to technical issues, we are unable to record anything prior to the accident. Also, the video is stored locally on Raspberry Pi, and due to the nature of MQTT server, we are unable to transmit this video to host. A possible improvement would be to have the camera constantly on and re-writing itself and upload the 10s video to a cloud server.

-Lack of specific information: Due to time and technical constraints we were unable to incorporate GPS and cellular module. It would be much better to have the message include GPS location and automatically calls 911.

- Front End: We didn't create a front-end portal. It would be much better to incorporate everything in a front-end portal.

Additional Technical Details:

1.The accelerometer uses I2C protocol for communicating with RPi