EEE 174 / CpE 185 Final Project Report

SQUISHY'S REVENGE: THE SEARCH FOR AN EAR DEVICE

Team Squishy

Team Members: Diep La, John Jimenez, Kyle Santoyo, Isai Flores

Team members and responsibilities

This section should list out each team member and what they are responsible for.

- Hardware/Wiring: Diep La
- Coding: Diep La, John Jimenez, Kyle Santoyo, Isai Flores
- Project Description: Isai Flores
- Finite State Machine: John Jimenez
- Test Cases: Kyle Santoyo

Add link to Bitbucket repository
https://bitbucket.org/team-squishy/src/src/master/

Project Description

This should contain a brief description of your project and its functionality. Include any features or necessary components of your project here.

We will be using the STM32F303 Nucleo32 board to manipulate the LEDs in this project. There are three LEDs which are red, green and blue and all these LEDs will act upon the sound sensor module.
- Here are the list that describes the behavior of our LEDs based on the order of sound it detect:

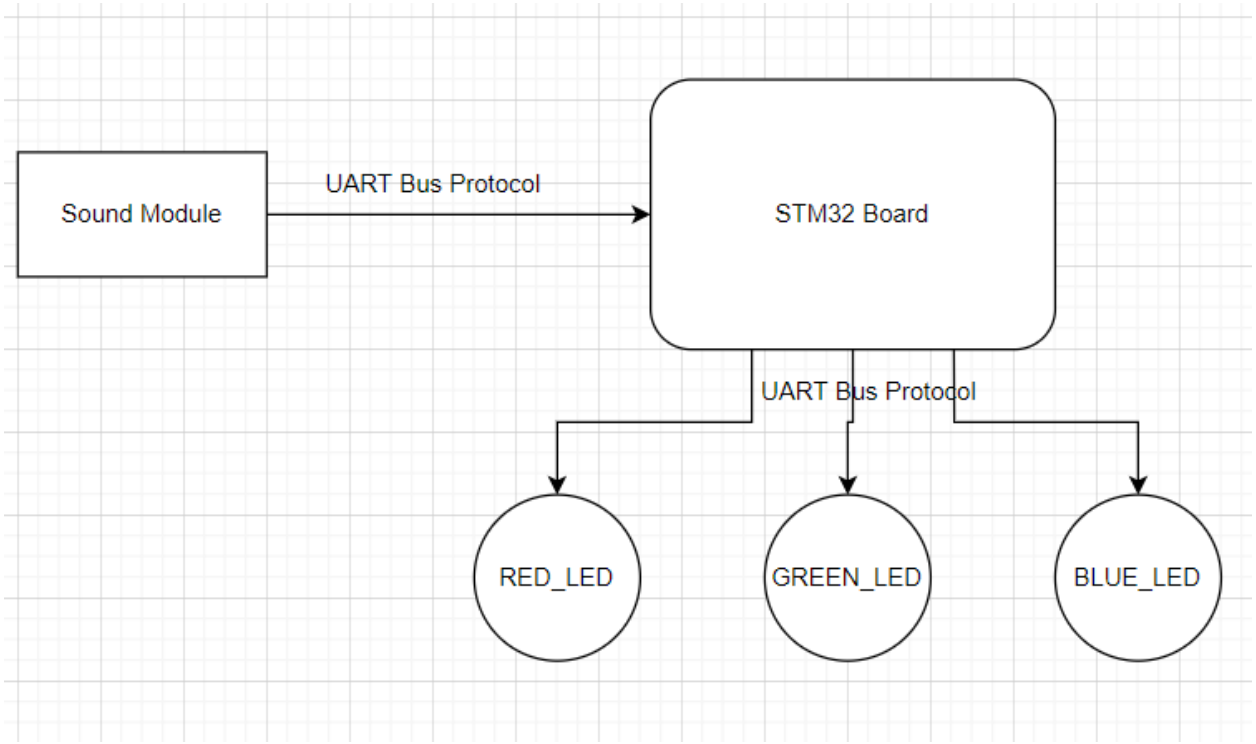
- If the first sound is detected, the cycle begins and the red LED light will be turn on
- The second time a sound is detected, the green LED light will be turn on
- The third sound detected will turn on the blue LED light
- Last sound will turn off all the LEDs and finish the cycle.

Block Diagram

Include a block diagram that shows each piece of hardware you plan to use in your project.

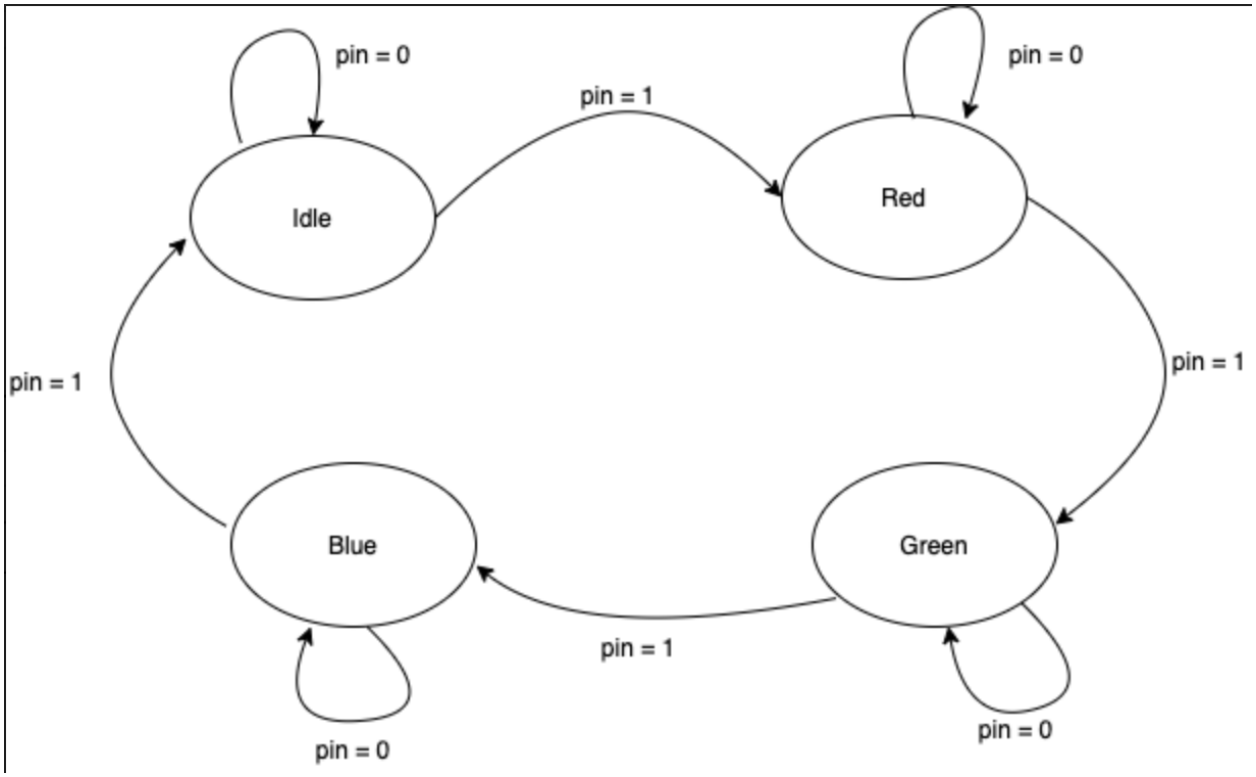
This should include which hardware platforms (STM, Microchip, and/or RPI) you plan to use as well as, and how they will be connected to each other or various sensors.

Include any bus protocols that may be used in the diagram. For example, I2C, UART, SPI, and CAN bus protocols.



FSM Diagram

Finite State Machine Diagram. This should reflect your final design.



Current State	Input Pin	Next State	Output
Idle	0	Idle	No LEDs
	1	S1	RED
S1	0	S1	RED
	1	S2	RED & GREEN
S2	0	S2	RED & GREEN
	1	S3	RED & GREEN & BLUE
S3	0	S3	RED & GREEN & BLUE
	1	Idle	No LEDs

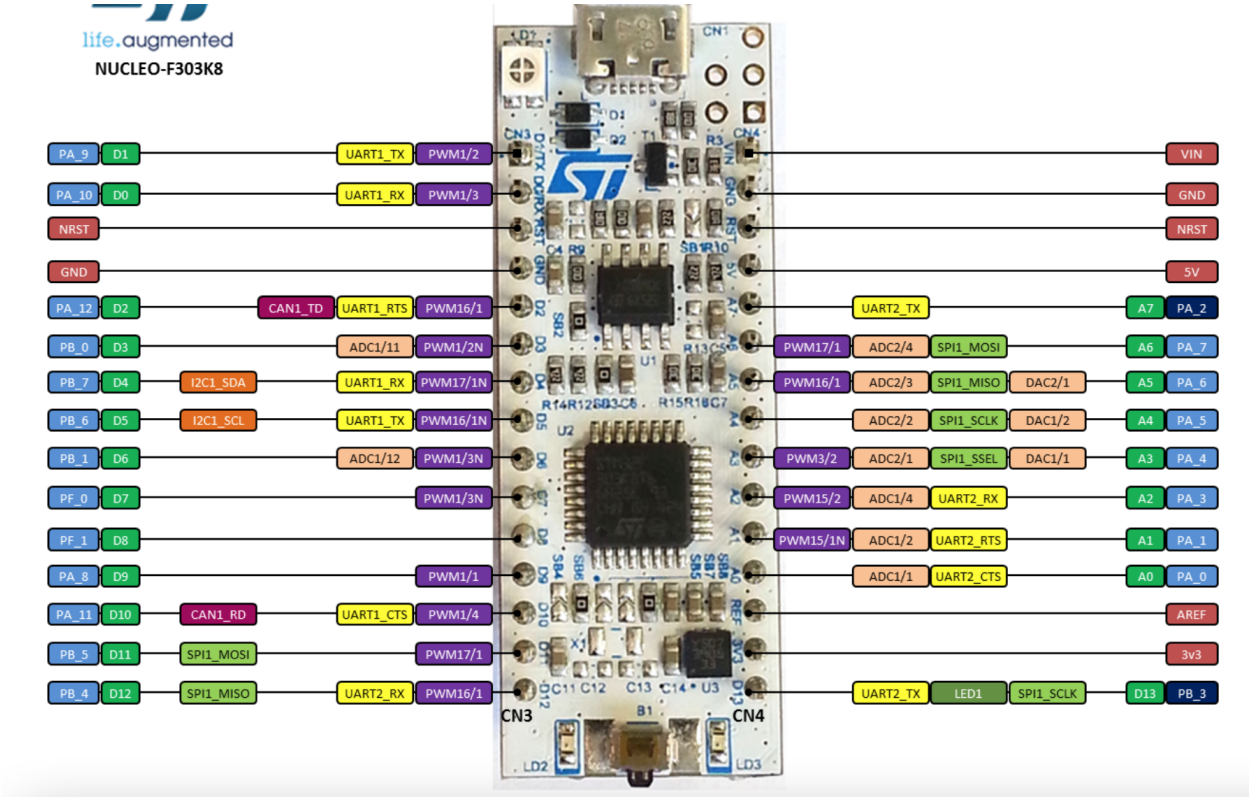
Background

This section should go over sensors or electrical mechanical devices (i.e. motors).

- In this project, we mainly utilize the STM32 board and the sound sensor detector. They will both interact with each other to produce the expected outputs as we have in our FSM above.

[STM32 Board]

This section should go over one sensor or electrical mechanical device. Include images detailing your sensor/electrical device.

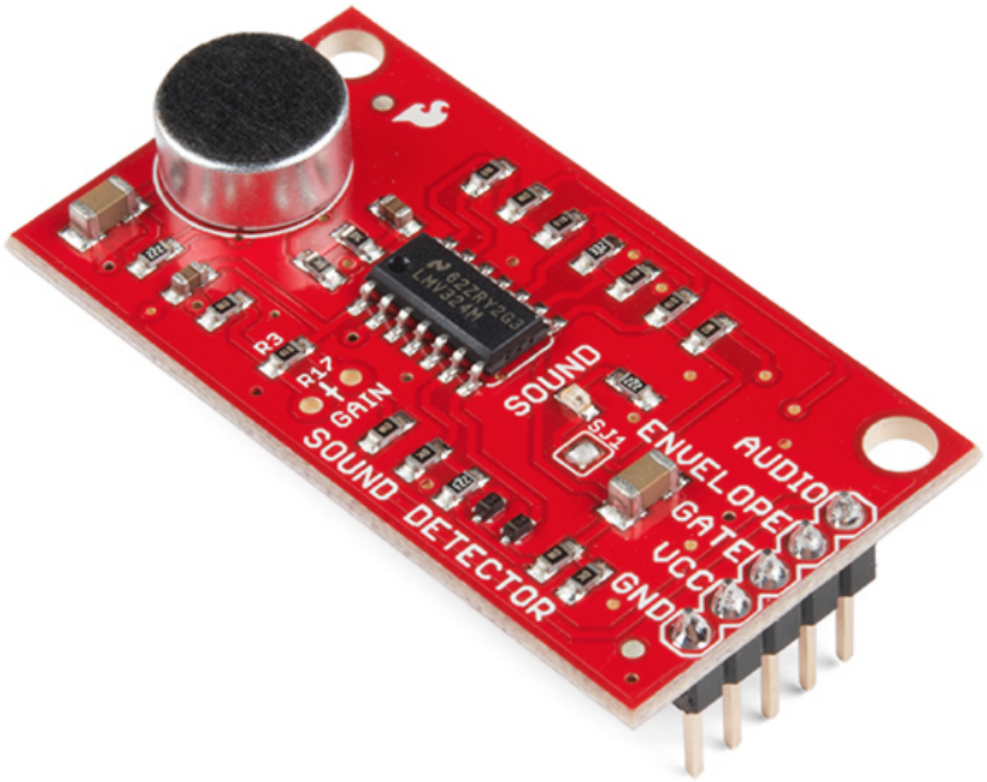


Source: https://os.mbed.com/platforms/ST-Nucleo-F303K8/

- The STM32 board will be used to get signals from the sound detector and give output signals to three LEDs red, green and blue. Depending on how the time it gets the signal, the LEDs will react respectively as shown in our FSM diagram and table.

[Sound Module]

This section should go over one sensor or electrical mechanical device. Include images detailing your sensor/electrical device.



Source: https://www.sparkfun.com/products/14262

- The sound detector that we utilize can give off three output signals through gate, envelope and audio pins. In this project, we are going to use the gate pin as an output signal to our STM32 board.

Results

- Here is the link to see the outputs of our project as expected https://www.youtube.com/watch?v=Hos_WqOHPD4

Test Cases

Any empirical data that is captured, and test cases done to verify functionality of device.

- The order in which the LEDs light up; Idle Stage - All LEDs are off, Stage 1 - Red LED is on, Stage 2 - Red and Green LEDs are on, Stage 3 - All three LEDs are on

- Test that all LEDs are off when Sound 4 is received; Return to Idle Stage

- Testing that the counter that stores the input at each stage of the program resets to zero after Sound 4 so that the program can be run any number of times as dictated by the user;

- Check if the microphone has a reverb effect that could count for additional unintended input(s); We added a buffer to mitigate this issue.

- Test the rate at which it detects input

Test Cases we were unable to test in time:
- If background noise is detected outside the intended input sound, if so how to completely mitigate it

- The range of detectable sound for the sensor(s) to take as inputs in order to confirm what our working range of inputs are

Conclusion

Sum up state of project, possibly discuss where the project can be done further. Reflect on things that could be improved in the project.

The state of the project was a working model that executes all of the base functions that we intended to implement. We successfully implemented 4 stages: Idle, Stage 1, Stage 2, and Stage 3 in which it would cycle through the LEDs, turning each one on with every input and then shutting them all off upon the sound module hearing the 4th sound. We managed to test the basic functions of the project, but areas that it can be taken further are preventing background noises from being picked up by the sound module, checking the range of detectable inputs by the sound module, and continue testing for different scenarios of additional unintended inputs that cause the program to advance sooner than intended.

References

Any outside references or resources used for the project.
Web links to reference sites used, (data sheets, sensors, open software sites, etc.)

https://microcontrollerslab.com/sound-detection-module-interfacing-with-pic-microcontroller/ ← Source of idea