NAVI

Navigation Assist for the Visually Impaired

ELE 392: Digital Signal Processing and Control Systems Laboratory
Junior Design Project

Henry C. Duisberg
Jenna M. Stapleton
Gilberto E. Ruiz

Supervised by Jennifer W. Graham, Ph. D

Table of Contents

Introduction

The objective of this project was to design and implement a real life application of a control system. First we tried to brainstorm some real life problems. The issue that came to us was that there are people who are visually impaired and need a cane or guide dog to help them get around. Navigation without sight is a challenge. NAVI, navigational assist for the visually impaired, aims to help.

Our junior design project is the solution for this issue. Our project, NAVI, will allow the visually impaired to wear a pair of earmuffs that have sensors (front, back, right, and left) on them and will alert them when they get close to an object . It will emit 'pip' tones with increasing frequency depending on how close they get.

Teensy 3.6 M4 is a microcontroller that has Arduino IDE as its primary programming environment and consists of a total of 64 input and output pins such as PWM pins, DAC pins, Analog GND pins, and regular Analog pins. We needed multiple analog pins because we had to wire a total of four proximity sensors to the microcontroller.

Proximity sensor VL53L0X, was our first choice for a distance sensor which measures the range to a target object up to 2m away. It uses time-of-flight measurements of infrared pulses for accurate readings depending on the color and surface. The measurements are read through a digital I2C interface which led to difficulties of having four of them running simultaneously so we decided to switch them for the SDS02A sensor.

Proximity sensor SDS02A, is a distance sensor from Sharp with a wide detection range from 4” to 60” (10cm to 150cm) that indicates its distance reading through analog voltage. It’s very easy to set up and program with the SharpIR library so it was the perfect sensor to use for our project.

As a result of our junior design project, we discovered new information. We were able to continue learning about control systems, and implement the controller and sensors together in the Arduino IDE. We also needed to build a circuit that would connect all four sensors to the controller (Arduino Teensy 3.5/3.6 M4). In addition, we learned how to connect all the components together.

Design and Implementation Plan

Design

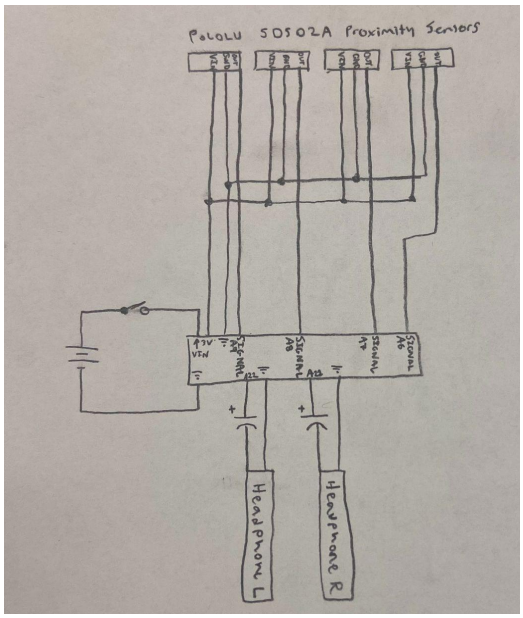


Image 1: Circuit Design of our Project

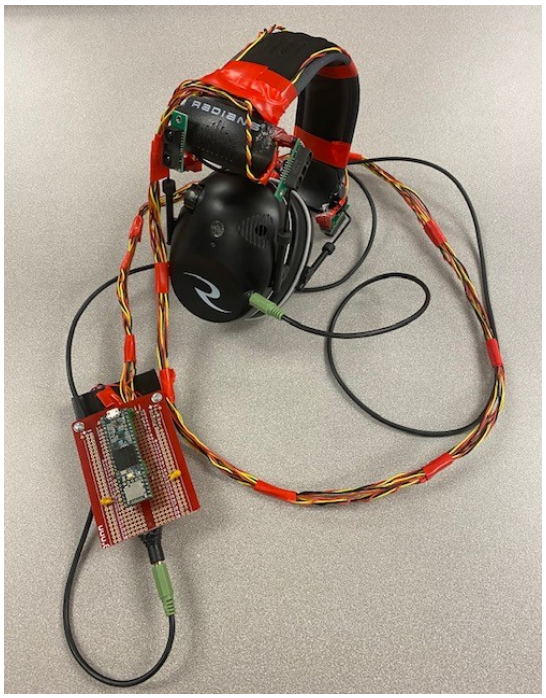


Image 2: Final Product

Equipment & Materials:

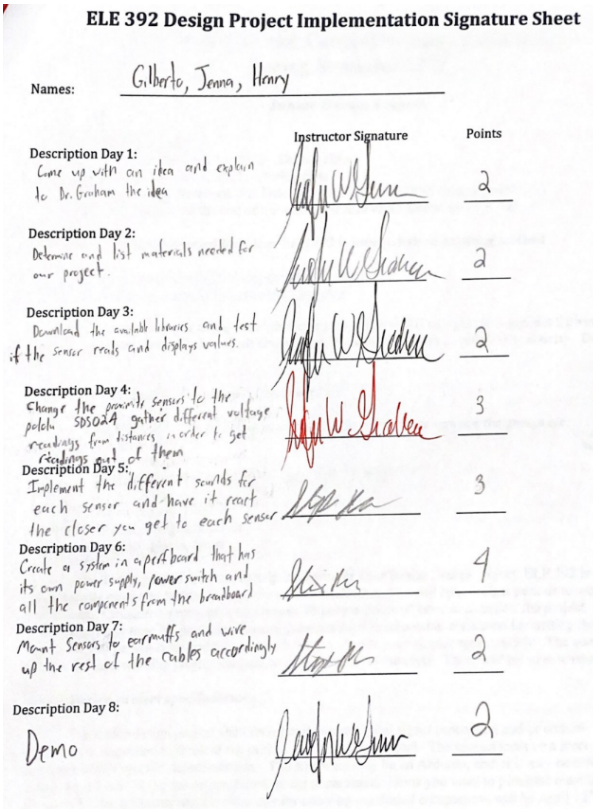
Hardware:

- Pololu SDS02A infrared proximity sensors (x4)
- Earmuffs/Headphones
- Electrical Tape
- Perfboard
- Auxiliary Audio Cable (3.5mm male to 3.5mm male)
- Wires
- Capacitors
- Teensy 3.6 M4
- Breadboard

Software:

- Analog Discovery
- Arduino IDE
    - Serial Monitor
    - Serial Plotter
    - Arduino SharpIR Library for Infrared Sensors

Implementation Plan



ELE 392 Design Project Implementation Signature Sheet

Names: Gilberto, Jenna, Henry

	Instructor Signature	Points
Description Day 1: Come up with an idea and explain to Dr. Graham the idea		2
Description Day 2: Determine and list materials needed for our project.		2
Description Day 3: Download the available libraries and test if the sensor reads and displays values.		2
Description Day 4: Change the pinouts, sensors to the potch SDSO2A gather different voltage readings from distances in order to get readings out of them		3
Description Day 5: Implement the different sounds for each sensor and have it react the closer you get to each sensor		3
Description Day 6: Create a system in a perfboard that has its own power supply, power switch and all the components from the breadboard		4
Description Day 7: Mount Sensors to earmuffs and wire up the rest of the cables accordingly		2
Description Day 8: Demo		2

Image 3: Implementation Signature Sheet

Description Day 1:

Day one's signature was to come up with a junior design project idea that we could potentially accomplish within the remaining weeks of the semester. Our idea is to create a sensor "cane" to help the visually impaired. We plan to put four sensors on a pair of headphones that will notify the person using them when they are coming in close proximity to something around them (front, back, left, right).

Today, everyone was present in the lab session. As a team, we brainstormed different ideas and asked each other if they liked the idea or not. We had three ideas, but the third idea just kept coming back to us. We just kept trying to improve that idea once it was in our heads. Jenna did some research on the types of sensors we will need. Gilberto and Henry went around the lab room to see if they could find any components that we could use in our project, else they wrote down what we might need to buy. The notebook was not open yet during class time, but we plan on writing what we did today after class. Our next steps for this project are to research a little more into this idea and come up with a list of materials we need to buy.

Description Day 2:

The demonstration for this signature was a list of all the materials needed for our junior design project. We decided that this signature was for 2 points.

As a group we discussed what materials we needed for the project and came to the conclusion that we don't need to order anything online for our project because all the hardware is provided here in the lab. Henry started working on the Arduino code to test each proximity sensor, Jenna started the lab notebook and Gilberto made an excel sheet with all the materials needed for this project.

Description Day 3:

Due to our group not having the signing sheet till today, we had to demonstrate for 3 signatures today. The first one was our idea for the junior design project, a list of materials needed for the project as the second signature, and lastly, to install all the necessary libraries in Arduino IDE and get one proximity sensor to read out data and form a noise. All those three signatures were 2 points each, leaving us with 6 out of 20 for now.

We were very successful today where not only were we the first group to get signatures, but we managed to get three signatures. In today's lab session, Jenna started the lab notebook and wrote all the details and steps we've taken until today. Gilberto brainstormed the design of the project and mentioned that we should have the sensors to be removed easily, like a "plug and play" sort of thing. Henry opened Arduino and tested a proximity sensor to make noise when you put anything a foot away from the sensor.

Description Day 4:

The demonstration that we planned on showing today was to have multiple proximity sensors running and detecting objects at the same time for 3 points. We planned on doing it after demonstrating that we installed a potentiometer for volume control, and changed the wiring from the M0 Itsy Bitsy to the Teensy 3.6 M4 board.

For today's lab, we unfortunately ran into a problem that put us back into square one of our project. We discovered that the proximity sensor we were using, the VL53L0X, can only have one sensor running. Meaning that we can't have 4 of those sensors running at the same time due to it being an I2C sensor. To solve this issue, we changed the proximity sensor to a different kind of proximity sensor, an infrared one that has an analog signal. As for group performance, Jenna updated the notebook today, Henry was working on the code in order to get the same response in sound for one of the new sensors, and Gilberto crimped 4 sets of wires together in order to create a cable connection between the Teensy board and the new proximity sensor.

Description Day 5:

For today's demonstration, we showed that we replaced all the old proximity sensors and installed the new polou SDS02A sensors. In addition to that, 4 sensors display a reading from detecting our hands. This demonstration almost puts us back to where we were before, and we decided that this signature was for 3 points.

Unfortunately, one of our team members, Jenna, fell ill and was not able to attend the lab in person. However, Gilberto called her on Facetime so that she was able to keep up with what the group had been working on. Jenna was able to use this time to log the group's progress in the team's online notebook. Gilberto wired all the proximity sensors to the board and twisted the cables to better organize them. Henry set up the Arduino code where the sensor displays voltage readings depending on how far away the object is. Gilberto and Henry also made the effort of putting a measuring tape on the floor, pointing one sensor facing the wall 160cm away (with the tape as a guide) and getting closer to the wall every 10 cm to put on an excel file the different voltage readings the sensor gives out. Once all the readings were put on excel, a scatter plot was made which would demonstrate the behavior of the sensor as it got closer to the wall. We made good progress today but we are planning on meeting again tomorrow for an extra session to work on installing the sensors to the earmuffs we ordered on Amazon.

Description Day 6:

For the last lab session, we planned to demonstrate that all 4 proximity sensors were giving readings together and it was outputting an audio signal. This demonstration was listed to be a total of 3 points, but we ran into trouble from the audio signal output and couldn't demonstrate on time.

Since Jenna still couldn't come to the lab she worked on the poster for our project. Gilberto worked a little on the notebook and made a wiring for two summation amplifiers on the circuit to get an audio signal output. Henry coded in Arduino a new library called SharpIR to get the Infrared Sensors to read and display distance.

Ethical Justification of the Project

The concept behind this project was to invent a device which could provide those with severe visual impairment a way to more easily and safely navigate their surroundings. More specifically, this product has the potential to reduce head injuries caused by the visually impaired running into objects positioned at or near the level of the head, where a traditional cane would not be effective at detecting these objects.

This has obvious ethical implications, first and foremost that regardless of disability, humans ought to have a right to safety. No human being chooses to be born with, or without, a disability, and few, if any, would choose to develop one later in life. Therefore, we believe it is our duty to help those with disabilities lead happy, healthy, and safe lives.

Description of Design Results

Day 1:

We spent the first day of the junior design project exploring potential project ideas. We had to think of ideas that weren't too easy or too hard to create within the three weeks we had. When we were brainstorming ideas, we kept returning to one concept. Since each new idea we had was just improving our initial concept, we decided to pursue the idea of creating a navigation aid. We began investigating the components of the product after we had settled on our concept. We researched the sensors that we could potentially use. We also talked with the professors about our project to bounce ideas off each other for what components we could potentially use. In addition, we had to look around the lab to examine what materials we already had. We made a list of everything we needed and divided it into sections of components to buy and components we already had.

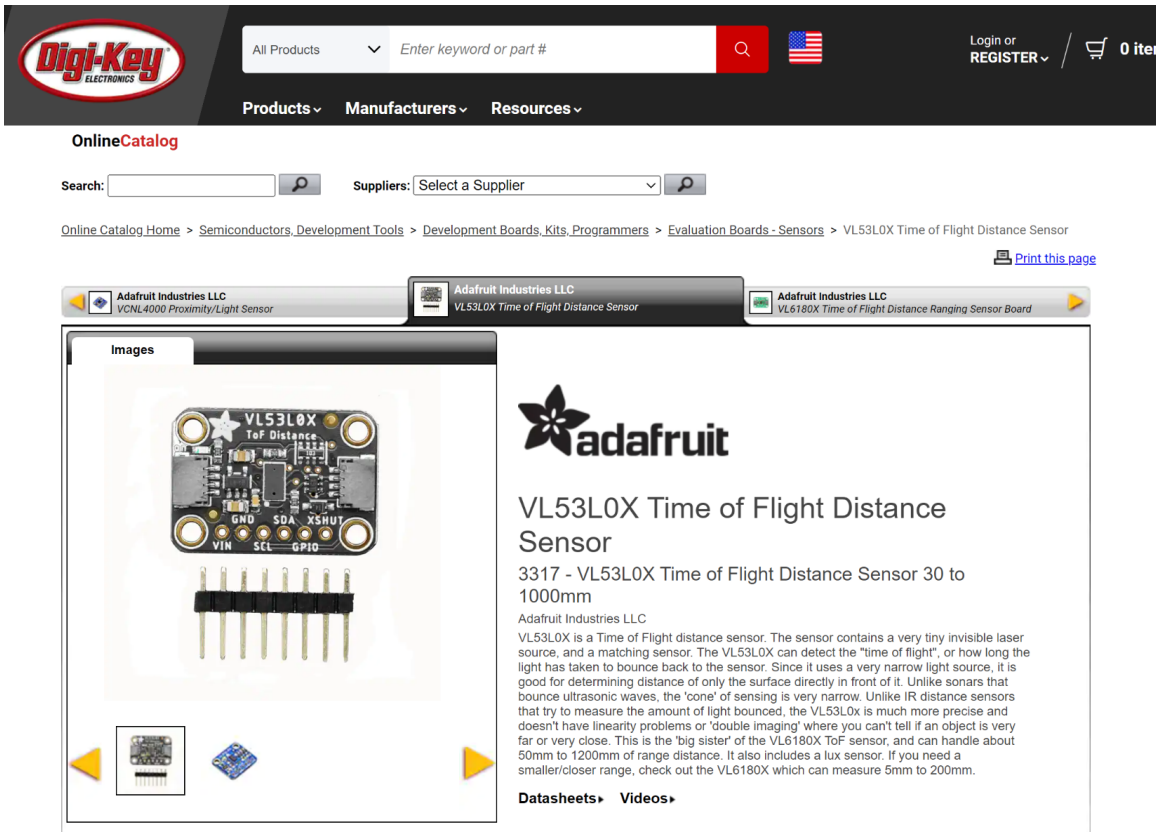


Image 4: First Proximity Sensor We Were Originally Going to Use

Day 2:

On day two, we finalized our list of materials for our junior design project. Once we looked at the list of materials we needed to make our project, we came to the conclusion that we didn't need to buy anything except the base we were going to use. It took us a couple of days to figure out what we were going to use for the base of the product. Our original idea for the base was a baseball cap but then we discussed a little more and decided that not everyone wears hats and some places don't allow hats indoors. Then we came up with the idea of using a headband as the base but realized that it wouldn't be beneficial if we were going to play a sound when detecting how close you were to an object. Our final idea was to use earmuffs as the base because then you could hide the headphones. In our case, the earmuffs we bought were also headphones so it worked out perfectly. So in the end, the only components we bought were the earmuffs. We also ended up buying batteries.

After researching the different proximity sensors, we chose to use the VL53L0X. We talked with the professors about what microchip would work best with the sensor, and they told us the Teensy M4 would work best. Once we showed the final list of materials we needed to buy to the professors, they told us that we didn't need to buy anything because the lab room had all the materials on the list. In addition, they told us that we didn't need to 3D print anything and we could just mount the sensors to the earmuffs. Our original idea was to 3D print a mount for the sensors.

Vendor	Item Number	Web Link	Unit Price	Quantity	Total Price	Description
Pololu	VL53L0X	https://www.pololu.com/product/2490	$12.95	4	$51.80	This is the proximity sensor that will detect any wall or object coming towards you
Digikey	DEV-14056	link	$31.25	1	$31.25	Teensy 3.5 Development Board
duracell				1		9 volt battery
walmart				1		Ear Muffs
Sci tech (it's here)				1		Headphones

Image 5: List of Materials Needed For Project

Day 3:

On day three, we created a circuit with one of the proximity sensors and we started arduino code to test the sensor. We were able to get the one proximity sensor to work with our arduino code. But then started researching how to have multiple sensors working at the same time. We researched that we could change the address of the sensor because using the same sensor, they would all have the same addresses. We could change the addresses in the arduino code with the code lines:

- Lox.begin(0x30)
- Lox.setAddress(0x30)

But then we found out that even though we could connect multiple sensors to the I2C ports, they wouldn't work simultaneously. We would have to “shut down” the other three sensors to allow the one to work. So we had to figure out a different solution. We talked with the professors and decided to use a different proximity sensor that would allow us to use multiple sensors at the same time. The sensor we chose was the SDS02A. It still worked with the Teensy M4.

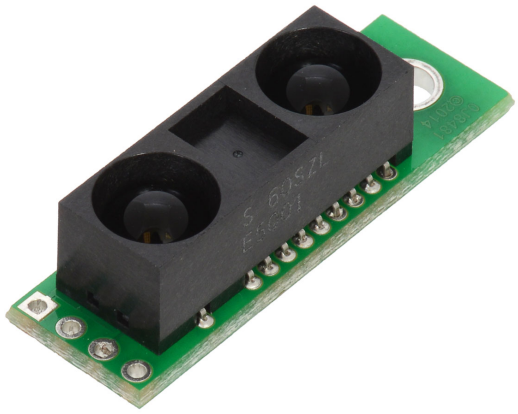


Image 5: Pololu SDS02A Proximity Sensor (Sensor we ended up using)

Day 4:

On day four, before we could work on the project, we had to demonstrate to the class our proposal and inform them what our idea was. After presenting our proposal, we now had to transfer our working circuit from the Itsy Bitsy M0 to our actual Teensy 3.6 M0 board like planned, but before that, we drew a diagram of the connections between the sensor and the Itsy Bitsy in order to use it as a reference while changing them, this is the diagram:

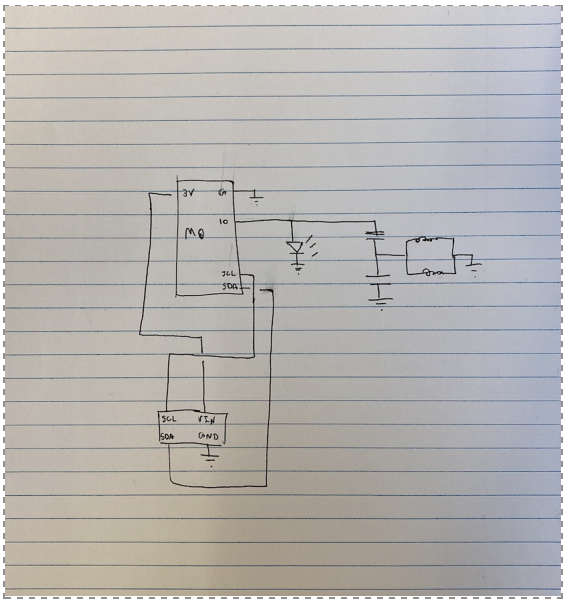


Image 6: Wiring Diagram of First Design

To make our main circuit, since we still haven't gotten our ear muffs, we decided to put them all on breadboards and wire them up. It's also a way to double check if the system is working before permanently soldering them together.

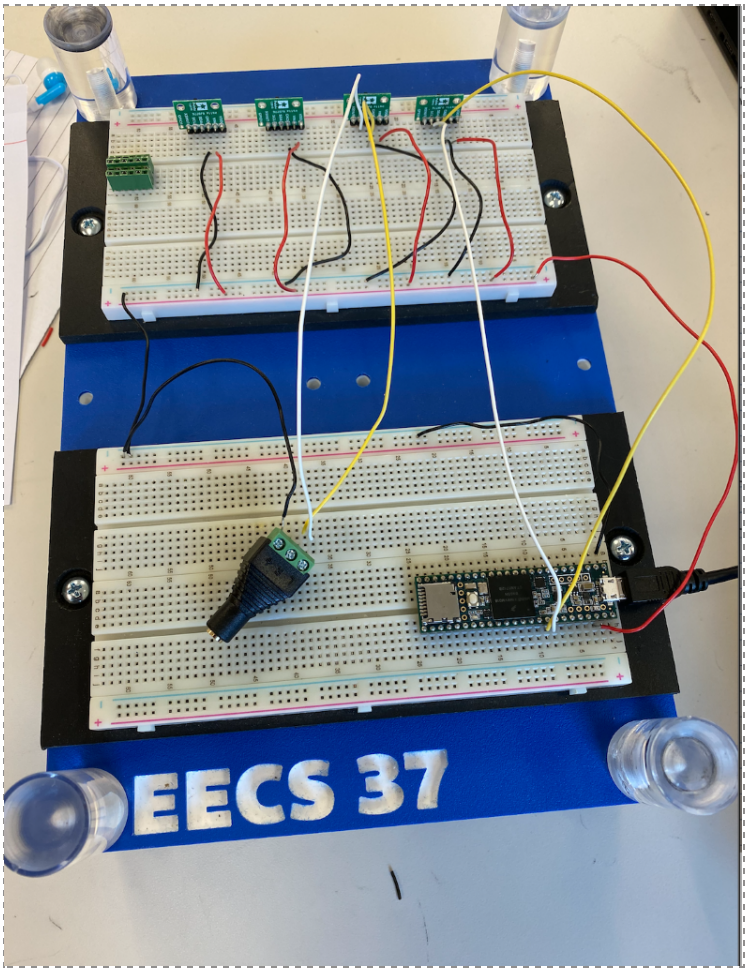


Image 7: Wiring of the Teensy 3.6 with Proximity Sensors

After rewiring and getting one of the sensors to respond with the Teensy, we added an led light to help us visualize that the audio signal is being sent instead of having to constantly listen to the earbuds. In addition to that, we added a potentiometer as a temporary “volume control” because the audio signal being sent would sometimes be too loud so in order to keep our project ethical, we implemented it. We had planned to get all four sensors working at the same time but time ran out because we were running into some complications with the code.

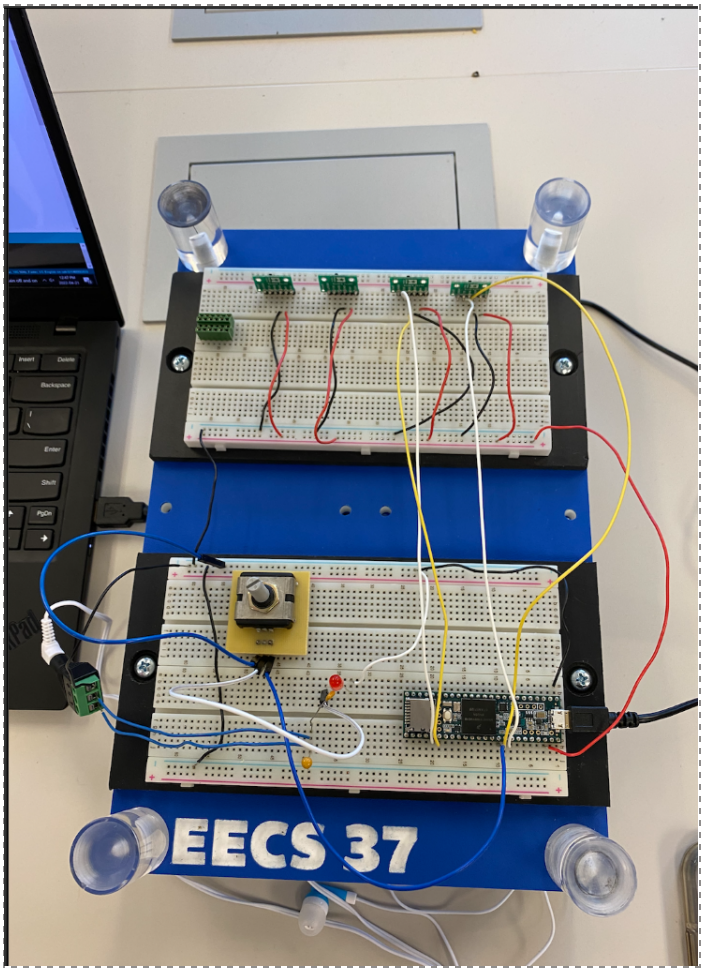


Image 8: Wiring with Potentiometer and Led Bulb

Day 5:

For the fifth lab session, we developed our code and tried to get all four sensors running (Figure 1 of the Additional Figures section demonstrates what we had in code). Unfortunately, we researched how to implement two sensors to work at the same time and made the discovery that there can't be multiple VL53L0X sensors working at the same time with each other. According to the information we gathered, I2C sensors can only allow one address-per-device so you have to make sure each I2C device has a unique address but the other sensors have to be in shutdown. This set us back to square one on our project and we had to rewrite our signature steps as well. Since we can't use the VL53L0X sensors, we made the decision to replace them with the SDS02A Infrared proximity sensor, which allows us to use multiple ones of them at the same time.

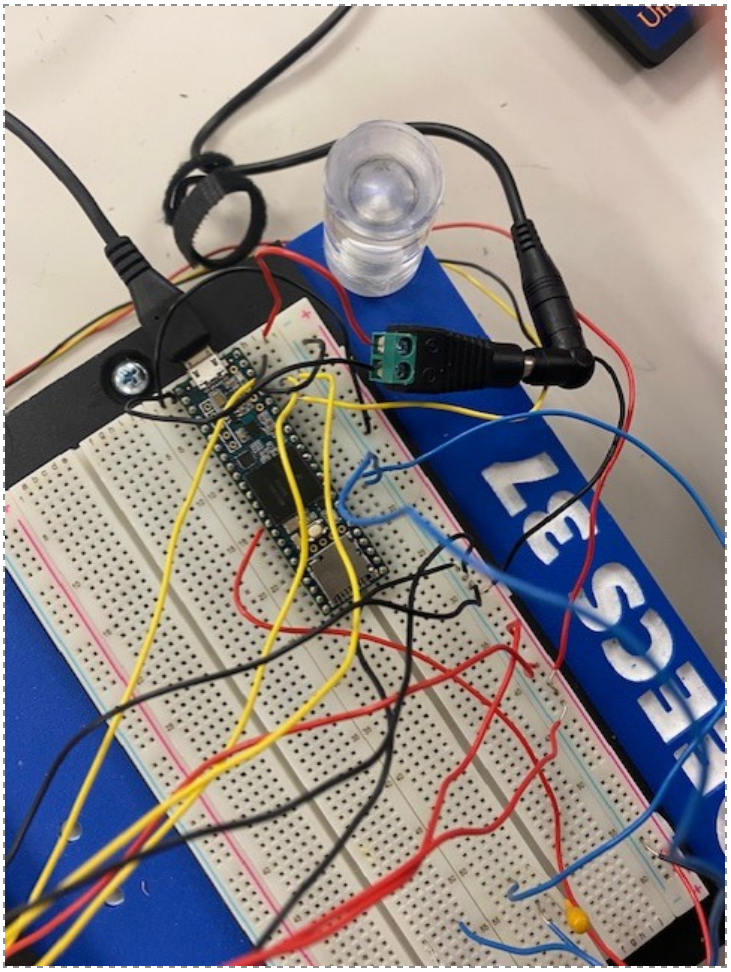


Image 9: New Wiring of Teensy with 4 SDS02A Sensors

Day 6:

Now that the sensors are wired to the board, we now had to write code to get them to read distances. We came up with multiple ideas on how to do it but we decided on setting up the arduino code to get different voltage readings at different distances. We characterized the sensor by taking measurements from 10 to 160cm in increments of 10cm, where one of us held the sensor 10 cm from the floor and pointed it to the wall. As we held the sensor and got the readings on the serial monitor, we would pause it for every increment of 10cm towards the wall and then write the data into Microsoft Excel for analysis. Once we got all the measurements, we generated a scatter plot that would show the behavior of the sensor as it got closer to the wall.

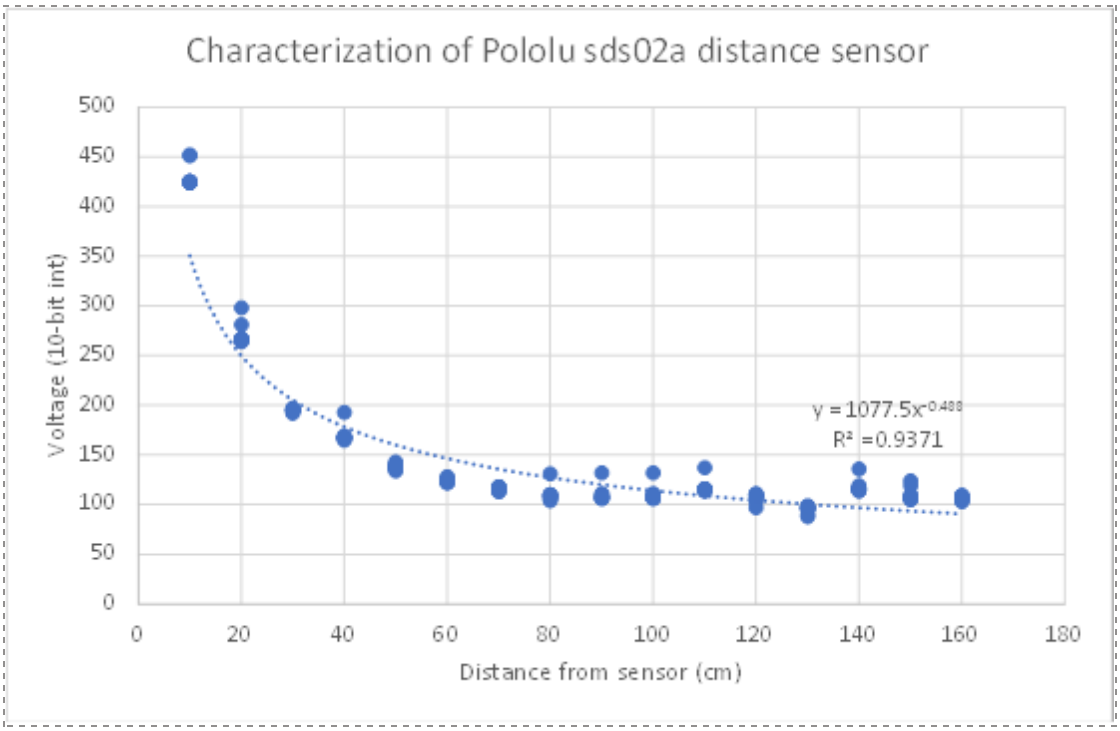


Image 10: Scatter Plot of Infrared Sensor Readings

After getting the plot, we now needed to implement different sounds for each sensor and have it react the closer and object gets to them. For this we found the inverse function of y=1077.5x^-0.488 using Symbolab. Another development in the project is that we noticed that the function to go from a voltage came out to a ridiculous root that would be impractical for the Teensy to calculate in real time: that function is this:

$$\sqrt[61]{\frac{1077.5^{125}}{x^{125}}}$$

It led to some confusion on what the next step would be so we decided to scrap that idea and use a lookup table for each “case” of how fast the sensors should beep based on the voltage reading from each sensor.

Day 7:

After many attempts on cleaning the signal readings from each sensor, we came to a conclusion that our method wasn’t going to work too well. We realized that we were overcomplicating ourselves by trying to calculate distance from voltage readings in code because we discovered an

Arduino library that does all of that for us, it's called SharpIR. With this library, we got a working reading of the distance for all four proximity sensors. In addition to that, we added tone methods for each of the sensors in order to get different sounds for them.

Day 8:

For the final day before our demonstration, we now had to mount the sensors to the earmuffs and solder every cable and components to a perf board. The components in the perf board consisted of:

- Headout pins that holds the Teensy M4
- Wires
- The aux port to connect any headphone to
- Capacitors
- A battery pack (4 AA batteries)

We also finally bought the earmuffs we wanted at a Walmart store.

Image 11: Earmuff Use for Final Project

Before soldering anything, to be sure that we knew where each pin of the teensy and every component was going to be in the right place on the perf board, we organized a table to use as a reference while soldering:

The left side of the teensy (all pins from the left side) will go on the column C of the perfboard while the right side of the teensy will go on the column g of the perfboard.

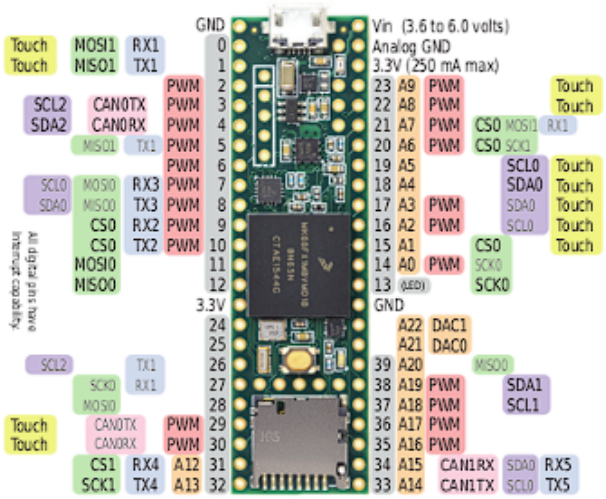


Image 12: Pinout of the Teensy (Reference)

Specific Function	Teensy 3.6 port (From the pinout)	Perfboard pin
Power to Sensors	3.3V pin	Left 15 >>(+)rail
Front Sensor: Yellow cable Red cable Black cable	A6 X X X	X Right 7 (+)rail (-)rail
Back Sensor: Yellow cable Red cable Black cable	A7 X X X	X Right 6 (+)rail (-)rail
Left Sensor: Yellow cable Red cable Black cable	A8 X X X	X Right 5 (+)rail (-)rail
Right Sensor: Yellow cable Red cable Black cable	A9 X X X	X Right 4 (+)rail (-)rail
Left Ear Sound	A22	Right 16
Right Ear Sound	A21	Right 17

(>> means that the previous port is connected to the next one through a wire)
(X means that it doesn't exist or should just disregard it)

Image 13: Teensy on the Perfboard

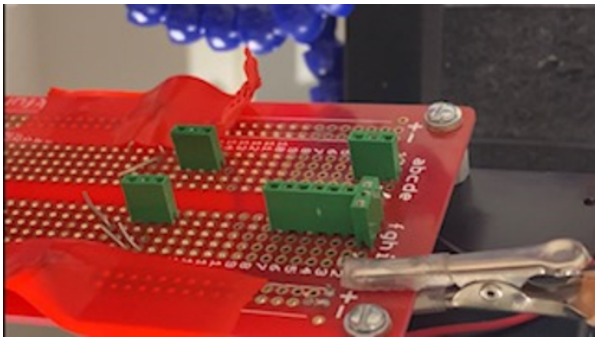


Image 14: Attaching Wires to Perfboard

With the table, it was a smooth transition from the breadboards to the final perf board model.

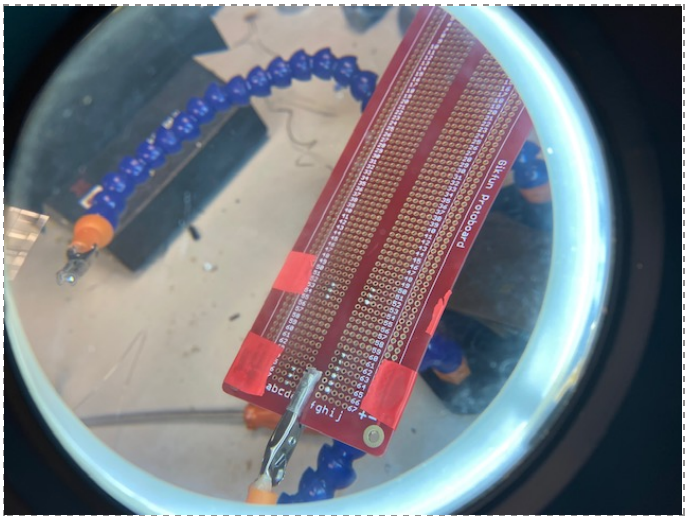


Image 15: Soldering the Header Pins to the Perfboard (Close Up)

Image 16: Soldering the Header Pins to the Perfboard

A discovery we made from reading more into the Teensy 3.6 is that the board can work just as fine up to 6V so we decided to just put a four AA battery pack in the back of the perfboard with screws and standoffs and solder the pack into the Teensy's positive and negative pins in order to make the design portable.

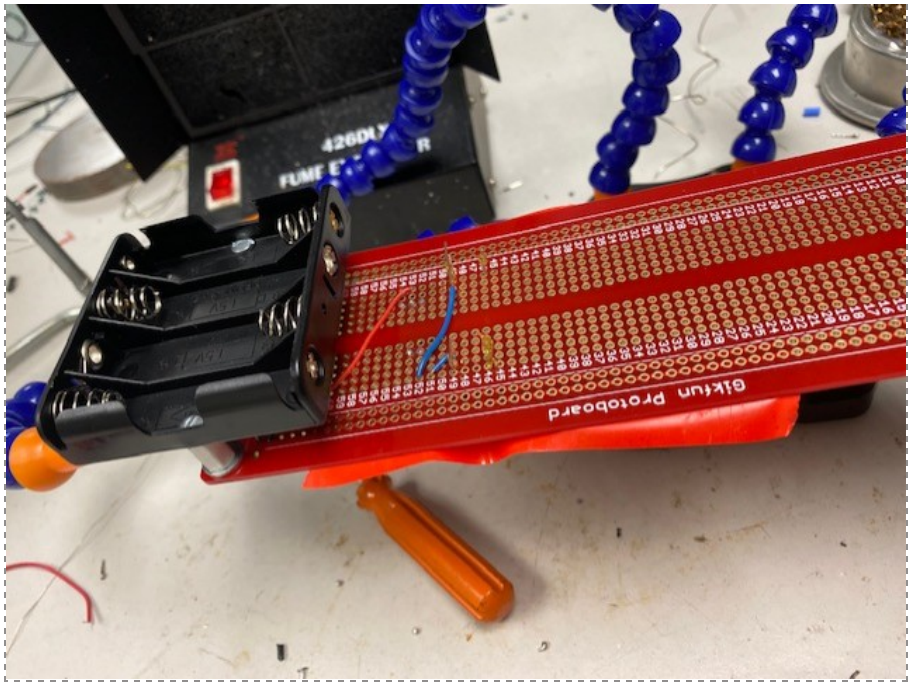


Image 17: The New Voltage Source of the System (Batteries)

After soldering for hours and mounting the sensors to the earmuffs, all that was left to do is organize the wires so that the user of our project won't be tangled with them.

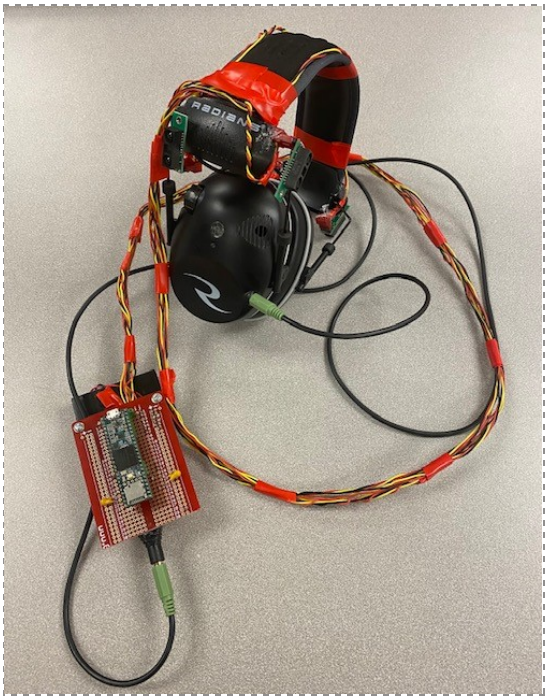


Image 18: NAVI Project Design With Wires

Day 9:

We gave a presentation and demonstration of NAVI on the final day of our junior design project. Our team explained to the audience how we created and executed NAVI's final product. We described how we came up with the idea for this project and the components we needed to make it. Our first presentation was Henry demonstrating how our device worked, and our second demonstration involved having a volunteer try our product.

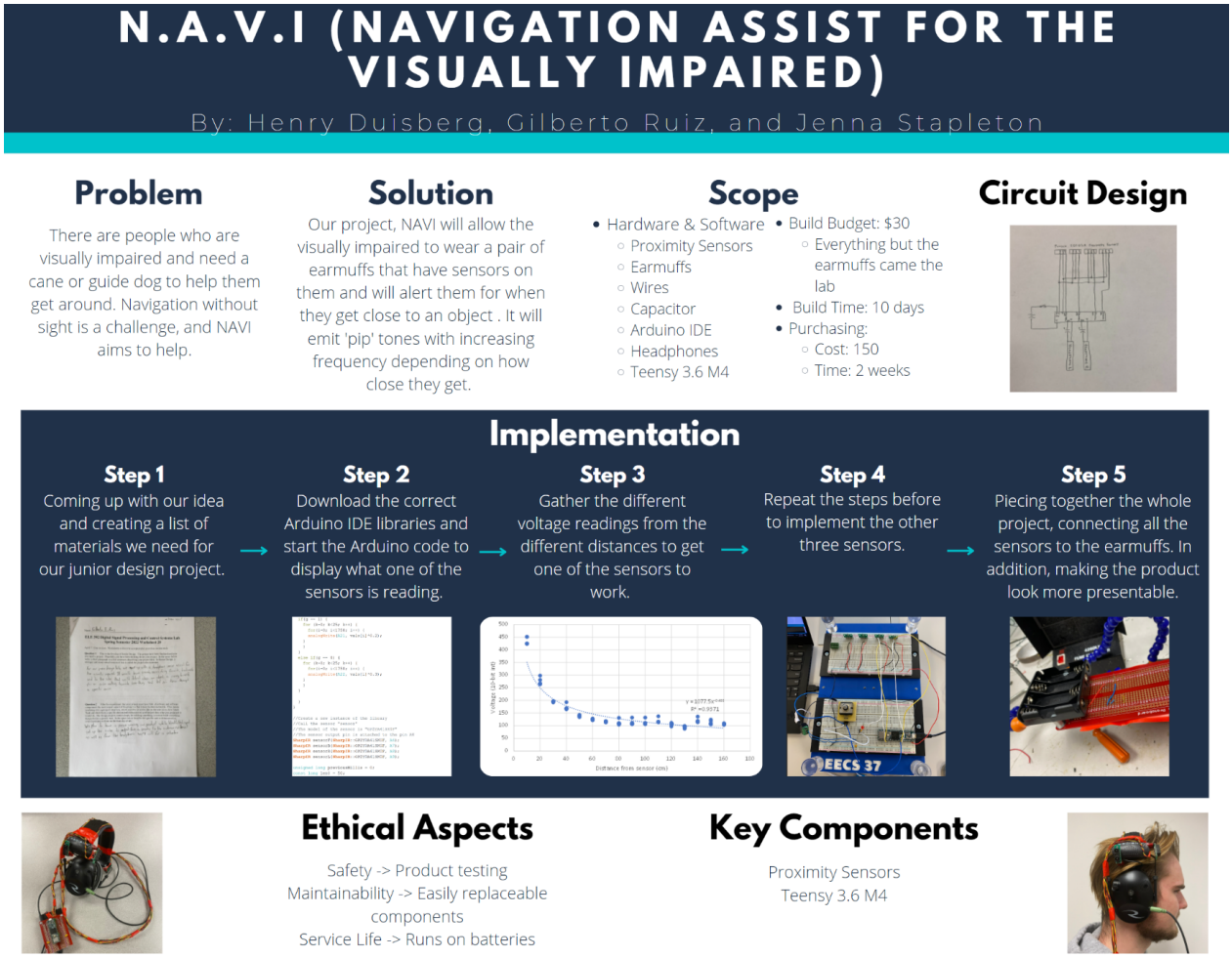


Image 19: Project Poster

Image 20: Final Product on User

Assessment of Experimental Results

We learned how to create and implement our own project concept during the course of this assignment. This project proved challenging at almost every turn. The most challenging elements for us were working out the concept, selecting components, and using the SharpIR library.

We learned even more about control systems as a result of this assignment. Using control loops, a control system manager, commands, directs, or regulates the behavior of other devices or systems. We also learned about product development, the engineering process, group work, maintaining proper work notes, problem solving and debugging, and more. This project puts what we've learned during this semester into practice. Aside from the main contents of the project, we continued learning how important it was to preserve project files, which included keeping our notebook up to date. Our notebook has all the project details which includes what we accomplished, our challenges, our procedures and methods, etc. It also includes the arduino code used to to program the board and data plots from the sensors.

This project was different from the other projects because we didn’t have to create the project details ourselves. Before, the professor gave us the project outcome, the components, and hints throughout the project in our worksheets on how to complete it. Since we completed three projects before, we were able to use our previous knowledge to complete this project. The first step of this project, coming up with a project idea was definitely a difficult part. It took us a whole day and a half to figure out an idea and the components we needed to produce our idea. Once we had the idea, the next step of figuring out what components we needed was also challenging. It was trial and error with the sensors we were using. But once we finalized what sensors to use, it was pretty easy to piece everything together. The only other issue we ran into was with trying to get our code for the sensors to work before learning about the SHARPIR library. At first, we tried to calculate our own distance readings through the voltage readings but it led to an overcomplicated code that just didn’t work. Luckily, we learned about the SHARPIR library that does it for us and made it possible to finish the project. In the end, we were able to present and demonstrate the final version of our working product to an audience.

This design project allowed us (students) to show the different real life products one can create using control systems and or digital signal processing. In addition, we learned more about the engineering process, how to break down and resolve an issue, and how to work together efficiently.

Additional figures

Figure 1: Version of the code which implements a VL53L0X distance sensor

```
#include "Adafruit_VL53L0X.h"
#include <Wire.h>

#define LOX0_ADDRESS 0x30
#define LOX1_ADDRESS 0x31

int sensor0;
int sensor1;

Adafruit_VL53L0X lox0 = Adafruit_VL53L0X();
Adafruit_VL53L0X lox1 = Adafruit_VL53L0X();

void setup() {
  Serial.begin(115200);

  // wait until serial port opens for native USB devices
  while (! Serial) {
   delay(1);
  }

  Serial.println("Adafruit VL53L0X 0 test");
  if (!lox0.begin()) {
   Serial.println(F("Failed to boot VL53L0X 0"));
   while(1);
  }
  if (!lox1.begin()) {
   Serial.println(F("Failed to boot VL53L0X 1"));
   while(1);
  }
  // power
  Serial.println(F("VL53L0X API Simple Ranging example\n\n"));
  pinMode(17, OUTPUT);

  Wire.setSCL(19);
  Wire.setSDA(18);
  Wire1.setSDA(38);
  Wire1.setSCL(37);
  Wire2.setSCL(3);
  Wire2.setSDA(4);
//  Wire3.setSCL(7);
```

```
//  Wire3.setSDA(8);
  Wire.begin();
  analogWriteFrequency(17, 588);
}

double distance0;
double distance1;
double distance2;

void loop() {
  VL53L0X_RangingMeasurementData_t measure0;
  VL53L0X_RangingMeasurementData_t measure1;

//  Serial.print("Reading a measurement... ");
  lox0.rangingTest(&measure0, false); // pass in 'true' to get debug data printout!
  lox1.rangingTest(&measure1, false); // pass in 'true' to get debug data printout!

  if (measure0.RangeStatus != 4) {  // phase failures have incorrect data
   Serial.print("Distance (mm): "); Serial.println(measure0.RangeMilliMeter);
  } else {
   Serial.println(" out of range 0");
  }
  if (measure1.RangeStatus != 4) {  // phase failures have incorrect data
   Serial.print("Distance (mm): "); Serial.println(measure1.RangeMilliMeter);
  } else {
   Serial.println(" out of range 1");
  }

  distance0 = measure0.RangeMilliMeter;
  distance1 = measure1.RangeMilliMeter;

  if(distance0 < 200) {
   if(distance1 < 200) {
   analogWrite(16,128);
   analogWrite(17,128);
   }
  }
  else if (distance0 < 100 && distance0 >= 50) {
   analogWrite(17,128);
   delay(50);
   analogWrite(17,0);
   delay(10);
  }
  else if (distance0 < 200 && distance0 >= 100) {
```

```
  analogWrite(17,128);
  delay(50);
  analogWrite(17,0);
  delay(25);
 }
 else if (distance0 < 400 && distance0 >= 200) {
  analogWrite(17,128);
  delay(50);
  analogWrite(17,0);
  delay(25);
 }
 else if (distance0 < 800 && distance0 >= 400) {
  analogWrite(17,128);
  delay(50);
  analogWrite(17,0);
  delay(100);
 }
 else if (distance0 < 1200 && distance0 >= 800) {
  analogWrite(17,128);
  delay(50);
  analogWrite(17,0);
  delay(200);
 }
 else if (distance0 < 1600 && distance0 >= 1200) {
  analogWrite(17,128);
  delay(50);
  analogWrite(17,0);
  delay(400);
 }
 else if (distance0 < 2000 && distance0 >= 1600) {
  analogWrite(17,128);
  delay(50);
  analogWrite(17,0);
  delay(1000);
 }
 else if (distance0 >= 2000) {
  analogWrite(17,0);
 }
 delay(100);
}

//int sensorCase(int front, int back, int left, int right){
//  int front = distance0;
//  int back  = distance1;
```

```
//  int left  = distance2;
//  int right = distance3;
//
//
if(front<2000&&front>=2000&&back<2000&&back>=2000&&left<2000&&left>=2000&&right
<2000&&right>=2000){
// return 0000;
//  }
//  else if(){
// return 0001;
//  }
//
//}
```

Figure 2: Final edition of the code on the Teensy 3.6 M4

```
//import the library in the sketch
#include <SharpIR.h>

void playSineTone1(bool cond) {
  analogWriteResolution(12);
  int i;
  int k;
  int vals [2000] = {0x800,0x806,0x80c,0x813,0x819,0x820,0x826,0x82d,
0x833,0x839,0x840,0x846,0x84d,0x853,0x85a,0x860,
0x866,0x86d,0x873,0x87a,0x880,0x886,0x88d,0x893,
0x89a,0x8a0,0x8a7,0x8ad,0x8b3,0x8ba,0x8c0,0x8c7,
0x8cd,0x8d3,0x8da,0x8e0,0x8e7,0x8ed,0x8f3,0x8fa,
0x900,0x907,0x90d,0x913,0x91a,0x920,0x926,0x92d,
0x933,0x939,0x940,0x946,0x94c,0x953,0x959,0x960,
0x966,0x96c,0x973,0x979,0x97f,0x985,0x98c,0x992,
0x998,0x99f,0x9a5,0x9ab,0x9b2,0x9b8,0x9be,0x9c4,
0x9cb,0x9d1,0x9d7,0x9dd,0x9e4,0x9ea,0x9f0,0x9f6,
0x9fd,0xa03,0xa09,0xa0f,0xa16,0xa1c,0xa22,0xa28,
0xa2e,0xa35,0xa3b,0xa41,0xa47,0xa4d,0xa53,0xa5a,
0xa60,0xa66,0xa6c,0xa72,0xa78,0xa7e,0xa84,0xa8b,
0xa91,0xa97,0xa9d,0xaa3,0xaa9,0xaaf,0xab5,0xabb,
0xac1,0xac7,0xacd,0xad3,0xad9,0xadf,0xae5,0xaeb,
0xaf1,0xaf7,0xafd,0xb03,0xb09,0xb0f,0xb15,0xb1b,
0xb21,0xb27,0xb2d,0xb33,0xb38,0xb3e,0xb44,0xb4a,
0xb50,0xb56,0xb5c,0xb61,0xb67,0xb6d,0xb73,0xb79,
0xb7e,0xb84,0xb8a,0xb90,0xb96,0xb9b,0xba1,0xba7,
0xbac,0xbb2,0xbb8,0xbbe,0xbc3,0xbc9,0xbcf,0xbd4,
0xbda,0xbe0,0xbe5,0xbeb,0xbf0,0xbf6,0xbfc,0xc01,
0xc07,0xc0c,0xc12,0xc17,0xc1d,0xc22,0xc28,0xc2d,
0xc33,0xc38,0xc3e,0xc43,0xc49,0xc4e,0xc53,0xc59,
0xc5e,0xc64,0xc69,0xc6e,0xc74,0xc79,0xc7e,0xc84,
0xc89,0xc8e,0xc94,0xc99,0xc9e,0xca3,0xca9,0xcae,
0xcb3,0xcb8,0xcbd,0xcc3,0xcc8,0xccd,0xcd2,0xcd7,
0xcdc,0xce1,0xce6,0xcec,0xcf1,0xcf6,0xcfb,0xd00,
0xd05,0xd0a,0xd0f,0xd14,0xd19,0xd1e,0xd23,0xd27,
0xd2c,0xd31,0xd36,0xd3b,0xd40,0xd45,0xd4a,0xd4e,
0xd53,0xd58,0xd5d,0xd61,0xd66,0xd6b,0xd70,0xd74,
0xd79,0xd7e,0xd82,0xd87,0xd8c,0xd90,0xd95,0xd9a,
0xd9e,0xda3,0xda7,0xdac,0xdb0,0xdb5,0xdb9,0xdbe,
0xdc2,0xdc7,0xdcb,0xdd0,0xdd4,0xdd8,0xddd,0xde1,
0xde6,0xdea,0xdee,0xdf3,0xdf7,0xdfb,0xdff,0xe04,
0xe08,0xe0c,0xe10,0xe14,0xe19,0xe1d,0xe21,0xe25,
```

0xe29,0xe2d,0xe31,0xe35,0xe39,0xe3d,0xe41,0xe45,
0xe49,0xe4d,0xe51,0xe55,0xe59,0xe5d,0xe61,0xe65,
0xe69,0xe6d,0xe70,0xe74,0xe78,0xe7c,0xe7f,0xe83,
0xe87,0xe8b,0xe8e,0xe92,0xe96,0xe99,0xe9d,0xea1,
0xea4,0xea8,0xeab,0xeaf,0xeb2,0xeb6,0xeb9,0xebd,
0xec0,0xec4,0xec7,0xecb,0xece,0xed1,0xed5,0xed8,
0xedb,0xedf,0xee2,0xee5,0xee8,0xeec,0xeef,0xef2,
0xef5,0xef8,0xefc,0xeff,0xf02,0xf05,0xf08,0xf0b,
0xf0e,0xf11,0xf14,0xf17,0xf1a,0xf1d,0xf20,0xf23,
0xf26,0xf29,0xf2b,0xf2e,0xf31,0xf34,0xf37,0xf39,
0xf3c,0xf3f,0xf42,0xf44,0xf47,0xf4a,0xf4c,0xf4f,
0xf51,0xf54,0xf57,0xf59,0xf5c,0xf5e,0xf61,0xf63,
0xf66,0xf68,0xf6a,0xf6d,0xf6f,0xf72,0xf74,0xf76,
0xf79,0xf7b,0xf7d,0xf7f,0xf82,0xf84,0xf86,0xf88,
0xf8a,0xf8c,0xf8f,0xf91,0xf93,0xf95,0xf97,0xf99,
0xf9b,0xf9d,0xf9f,0xfa1,0xfa3,0xfa4,0xfa6,0xfa8,
0xfaa,0xfac,0xfae,0xfaf,0xfb1,0xfb3,0xfb5,0xfb6,
0xfb8,0xfba,0xfbb,0xfbd,0xfbf,0xfc0,0xfc2,0xfc3,
0xfc5,0xfc6,0xfc8,0xfc9,0xfcb,0xfcc,0xfce,0xfcf,
0xfd0,0xfd2,0xfd3,0xfd4,0xfd6,0xfd7,0xfd8,0xfda,
0xfdb,0xfdc,0xfdd,0xfde,0xfdf,0xfe1,0xfe2,0xfe3,
0xfe4,0xfe5,0xfe6,0xfe7,0xfe8,0xfe9,0xfea,0xfeb,
0xfeb,0xfec,0xfed,0xfee,0xfef,0xff0,0xff0,0xff1,
0xff2,0xff3,0xff3,0xff4,0xff5,0xff5,0xff6,0xff7,
0xff7,0xff8,0xff8,0xff9,0xff9,0xffa,0xffa,0xffb,
0xffb,0xffb,0xffc,0xffc,0xffc,0xffd,0xffd,0xffd,
0xffe,0xffe,0xffe,0xffe,0xffe,0xfff,0xfff,0xfff,
0xfff,0xfff,0xfff,0xfff,0xfff,0xfff,0xfff,0xfff,
0xfff,0xfff,0xfff,0xfff,0xffe,0xffe,0xffe,0xffe,
0xffe,0xffd,0xffd,0xffd,0xffc,0xffc,0xffc,0xffb,
0xffb,0xffb,0xffa,0xffa,0xff9,0xff9,0xff8,0xff8,
0xff7,0xff7,0xff6,0xff5,0xff5,0xff4,0xff3,0xff3,
0xff2,0xff1,0xff0,0xff0,0xfef,0xfee,0xfed,0xfec,
0xfeb,0xfeb,0xfea,0xfe9,0xfe8,0xfe7,0xfe6,0xfe5,
0xfe4,0xfe3,0xfe2,0xfe1,0xfdf,0xfde,0xfdd,0xfdc,
0xfdb,0xfda,0xfd8,0xfd7,0xfd6,0xfd4,0xfd3,0xfd2,
0xfd0,0xfcf,0xfce,0xfcc,0xfcb,0xfc9,0xfc8,0xfc6,
0xfc5,0xfc3,0xfc2,0xfc0,0xfbf,0xfbd,0xfbb,0xfba,
0xfb8,0xfb6,0xfb5,0xfb3,0xfb1,0xfaf,0xfae,0xfac,
0xfaa,0xfa8,0xfa6,0xfa4,0xfa3,0xfa1,0xf9f,0xf9d,
0xf9b,0xf99,0xf97,0xf95,0xf93,0xf91,0xf8f,0xf8c,
0xf8a,0xf88,0xf86,0xf84,0xf82,0xf7f,0xf7d,0xf7b,
0xf79,0xf76,0xf74,0xf72,0xf6f,0xf6d,0xf6a,0xf68,
0xf66,0xf63,0xf61,0xf5e,0xf5c,0xf59,0xf57,0xf54,

0xf51,0xf4f,0xf4c,0xf4a,0xf47,0xf44,0xf42,0xf3f,
0xf3c,0xf39,0xf37,0xf34,0xf31,0xf2e,0xf2b,0xf29,
0xf26,0xf23,0xf20,0xf1d,0xf1a,0xf17,0xf14,0xf11,
0xf0e,0xf0b,0xf08,0xf05,0xf02,0xeff,0xefc,0xef8,
0xef5,0xef2,0xeef,0xeec,0xee8,0xee5,0xee2,0xedf,
0xedb,0xed8,0xed5,0xed1,0xece,0xecb,0xec7,0xec4,
0xec0,0xebd,0xeb9,0xeb6,0xeb2,0xeaf,0xeab,0xea8,
0xea4,0xea1,0xe9d,0xe99,0xe96,0xe92,0xe8e,0xe8b,
0xe87,0xe83,0xe7f,0xe7c,0xe78,0xe74,0xe70,0xe6d,
0xe69,0xe65,0xe61,0xe5d,0xe59,0xe55,0xe51,0xe4d,
0xe49,0xe45,0xe41,0xe3d,0xe39,0xe35,0xe31,0xe2d,
0xe29,0xe25,0xe21,0xe1d,0xe19,0xe14,0xe10,0xe0c,
0xe08,0xe04,0xdff,0xdfb,0xdf7,0xdf3,0xdee,0xdea,
0xde6,0xde1,0xddd,0xdd8,0xdd4,0xdd0,0xdcb,0xdc7,
0xdc2,0xdbe,0xdb9,0xdb5,0xdb0,0xdac,0xda7,0xda3,
0xd9e,0xd9a,0xd95,0xd90,0xd8c,0xd87,0xd82,0xd7e,
0xd79,0xd74,0xd70,0xd6b,0xd66,0xd61,0xd5d,0xd58,
0xd53,0xd4e,0xd4a,0xd45,0xd40,0xd3b,0xd36,0xd31,
0xd2c,0xd27,0xd23,0xd1e,0xd19,0xd14,0xd0f,0xd0a,
0xd05,0xd00,0xcfb,0xcf6,0xcf1,0xcec,0xce6,0xce1,
0xcdc,0xcd7,0xcd2,0xccd,0xcc8,0xcc3,0xcbd,0xcb8,
0xcb3,0xcae,0xca9,0xca3,0xc9e,0xc99,0xc94,0xc8e,
0xc89,0xc84,0xc7e,0xc79,0xc74,0xc6e,0xc69,0xc64,
0xc5e,0xc59,0xc53,0xc4e,0xc49,0xc43,0xc3e,0xc38,
0xc33,0xc2d,0xc28,0xc22,0xc1d,0xc17,0xc12,0xc0c,
0xc07,0xc01,0xbfc,0xbf6,0xbf0,0xbeb,0xbe5,0xbe0,
0xbda,0xbd4,0xbcf,0xbc9,0xbc3,0xbbe,0xbb8,0xbb2,
0xbac,0xba7,0xba1,0xb9b,0xb96,0xb90,0xb8a,0xb84,
0xb7e,0xb79,0xb73,0xb6d,0xb67,0xb61,0xb5c,0xb56,
0xb50,0xb4a,0xb44,0xb3e,0xb38,0xb33,0xb2d,0xb27,
0xb21,0xb1b,0xb15,0xb0f,0xb09,0xb03,0xafd,0xaf7,
0xaf1,0xaeb,0xae5,0xadf,0xad9,0xad3,0xacd,0xac7,
0xac1,0xabb,0xab5,0xaaf,0xaa9,0xaa3,0xa9d,0xa97,
0xa91,0xa8b,0xa84,0xa7e,0xa78,0xa72,0xa6c,0xa66,
0xa60,0xa5a,0xa53,0xa4d,0xa47,0xa41,0xa3b,0xa35,
0xa2e,0xa28,0xa22,0xa1c,0xa16,0xa0f,0xa09,0xa03,
0x9fd,0x9f6,0x9f0,0x9ea,0x9e4,0x9dd,0x9d7,0x9d1,
0x9cb,0x9c4,0x9be,0x9b8,0x9b2,0x9ab,0x9a5,0x99f,
0x998,0x992,0x98c,0x985,0x97f,0x979,0x973,0x96c,
0x966,0x960,0x959,0x953,0x94c,0x946,0x940,0x939,
0x933,0x92d,0x926,0x920,0x91a,0x913,0x90d,0x907,
0x900,0x8fa,0x8f3,0x8ed,0x8e7,0x8e0,0x8da,0x8d3,
0x8cd,0x8c7,0x8c0,0x8ba,0x8b3,0x8ad,0x8a7,0x8a0,
0x89a,0x893,0x88d,0x886,0x880,0x87a,0x873,0x86d,

0x866,0x860,0x85a,0x853,0x84d,0x846,0x840,0x839,
0x833,0x82d,0x826,0x820,0x819,0x813,0x80c,0x806,
0x800,0x7f9,0x7f3,0x7ec,0x7e6,0x7df,0x7d9,0x7d2,
0x7cc,0x7c6,0x7bf,0x7b9,0x7b2,0x7ac,0x7a5,0x79f,
0x799,0x792,0x78c,0x785,0x77f,0x779,0x772,0x76c,
0x765,0x75f,0x758,0x752,0x74c,0x745,0x73f,0x738,
0x732,0x72c,0x725,0x71f,0x718,0x712,0x70c,0x705,
0x6ff,0x6f8,0x6f2,0x6ec,0x6e5,0x6df,0x6d9,0x6d2,
0x6cc,0x6c6,0x6bf,0x6b9,0x6b3,0x6ac,0x6a6,0x69f,
0x699,0x693,0x68c,0x686,0x680,0x67a,0x673,0x66d,
0x667,0x660,0x65a,0x654,0x64d,0x647,0x641,0x63b,
0x634,0x62e,0x628,0x622,0x61b,0x615,0x60f,0x609,
0x602,0x5fc,0x5f6,0x5f0,0x5e9,0x5e3,0x5dd,0x5d7,
0x5d1,0x5ca,0x5c4,0x5be,0x5b8,0x5b2,0x5ac,0x5a5,
0x59f,0x599,0x593,0x58d,0x587,0x581,0x57b,0x574,
0x56e,0x568,0x562,0x55c,0x556,0x550,0x54a,0x544,
0x53e,0x538,0x532,0x52c,0x526,0x520,0x51a,0x514,
0x50e,0x508,0x502,0x4fc,0x4f6,0x4f0,0x4ea,0x4e4,
0x4de,0x4d8,0x4d2,0x4cc,0x4c7,0x4c1,0x4bb,0x4b5,
0x4af,0x4a9,0x4a3,0x49e,0x498,0x492,0x48c,0x486,
0x481,0x47b,0x475,0x46f,0x469,0x464,0x45e,0x458,
0x453,0x44d,0x447,0x441,0x43c,0x436,0x430,0x42b,
0x425,0x41f,0x41a,0x414,0x40f,0x409,0x403,0x3fe,
0x3f8,0x3f3,0x3ed,0x3e8,0x3e2,0x3dd,0x3d7,0x3d2,
0x3cc,0x3c7,0x3c1,0x3bc,0x3b6,0x3b1,0x3ac,0x3a6,
0x3a1,0x39b,0x396,0x391,0x38b,0x386,0x381,0x37b,
0x376,0x371,0x36b,0x366,0x361,0x35c,0x356,0x351,
0x34c,0x347,0x342,0x33c,0x337,0x332,0x32d,0x328,
0x323,0x31e,0x319,0x313,0x30e,0x309,0x304,0x2ff,
0x2fa,0x2f5,0x2f0,0x2eb,0x2e6,0x2e1,0x2dc,0x2d8,
0x2d3,0x2ce,0x2c9,0x2c4,0x2bf,0x2ba,0x2b5,0x2b1,
0x2ac,0x2a7,0x2a2,0x29e,0x299,0x294,0x28f,0x28b,
0x286,0x281,0x27d,0x278,0x273,0x26f,0x26a,0x265,
0x261,0x25c,0x258,0x253,0x24f,0x24a,0x246,0x241,
0x23d,0x238,0x234,0x22f,0x22b,0x227,0x222,0x21e,
0x219,0x215,0x211,0x20c,0x208,0x204,0x200,0x1fb,
0x1f7,0x1f3,0x1ef,0x1eb,0x1e6,0x1e2,0x1de,0x1da,
0x1d6,0x1d2,0x1ce,0x1ca,0x1c6,0x1c2,0x1be,0x1ba,
0x1b6,0x1b2,0x1ae,0x1aa,0x1a6,0x1a2,0x19e,0x19a,
0x196,0x192,0x18f,0x18b,0x187,0x183,0x180,0x17c,
0x178,0x174,0x171,0x16d,0x169,0x166,0x162,0x15e,
0x15b,0x157,0x154,0x150,0x14d,0x149,0x146,0x142,
0x13f,0x13b,0x138,0x134,0x131,0x12e,0x12a,0x127,
0x124,0x120,0x11d,0x11a,0x117,0x113,0x110,0x10d,

0x10a,0x107,0x103,0x100,0xfd,0xfa,0xf7,0xf4,
0xf1,0xee,0xeb,0xe8,0xe5,0xe2,0xdf,0xdc,
0xd9,0xd6,0xd4,0xd1,0xce,0xcb,0xc8,0xc6,
0xc3,0xc0,0xbd,0xbb,0xb8,0xb5,0xb3,0xb0,
0xae,0xab,0xa8,0xa6,0xa3,0xa1,0x9e,0x9c,
0x99,0x97,0x95,0x92,0x90,0x8d,0x8b,0x89,
0x86,0x84,0x82,0x80,0x7d,0x7b,0x79,0x77,
0x75,0x73,0x70,0x6e,0x6c,0x6a,0x68,0x66,
0x64,0x62,0x60,0x5e,0x5c,0x5b,0x59,0x57,
0x55,0x53,0x51,0x50,0x4e,0x4c,0x4a,0x49,
0x47,0x45,0x44,0x42,0x40,0x3f,0x3d,0x3c,
0x3a,0x39,0x37,0x36,0x34,0x33,0x31,0x30,
0x2f,0x2d,0x2c,0x2b,0x29,0x28,0x27,0x25,
0x24,0x23,0x22,0x21,0x20,0x1e,0x1d,0x1c,
0x1b,0x1a,0x19,0x18,0x17,0x16,0x15,0x14,
0x14,0x13,0x12,0x11,0x10,0xf,0xf,0xe,
0xd,0xc,0xc,0xb,0xa,0xa,0x9,0x8,
0x8,0x7,0x7,0x6,0x6,0x5,0x5,0x4,
0x4,0x4,0x3,0x3,0x3,0x2,0x2,0x2,
0x1,0x1,0x1,0x1,0x1,0x0,0x0,0x0,
0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,
0x0,0x0,0x0,0x0,0x0,0x1,0x1,0x1,0x1,
0x1,0x2,0x2,0x2,0x2,0x3,0x3,0x3,0x4,
0x4,0x4,0x5,0x5,0x6,0x6,0x7,0x7,
0x8,0x8,0x9,0xa,0xa,0xb,0xc,0xc,
0xd,0xe,0xf,0xf,0x10,0x11,0x12,0x13,
0x14,0x14,0x15,0x16,0x17,0x18,0x19,0x1a,
0x1b,0x1c,0x1d,0x1e,0x20,0x21,0x22,0x23,
0x24,0x25,0x27,0x28,0x29,0x2b,0x2c,0x2d,
0x2f,0x30,0x31,0x33,0x34,0x36,0x37,0x39,
0x3a,0x3c,0x3d,0x3f,0x40,0x42,0x44,0x45,
0x47,0x49,0x4a,0x4c,0x4e,0x50,0x51,0x53,
0x55,0x57,0x59,0x5b,0x5c,0x5e,0x60,0x62,
0x64,0x66,0x68,0x6a,0x6c,0x6e,0x70,0x73,
0x75,0x77,0x79,0x7b,0x7d,0x80,0x82,0x84,
0x86,0x89,0x8b,0x8d,0x90,0x92,0x95,0x97,
0x99,0x9c,0x9e,0xa1,0xa3,0xa6,0xa8,0xab,
0xae,0xb0,0xb3,0xb5,0xb8,0xbb,0xbd,0xc0,
0xc3,0xc6,0xc8,0xcb,0xce,0xd1,0xd4,0xd6,
0xd9,0xdc,0xdf,0xe2,0xe5,0xe8,0xeb,0xee,
0xf1,0xf4,0xf7,0xfa,0xfd,0x100,0x103,0x107,
0x10a,0x10d,0x110,0x113,0x117,0x11a,0x11d,0x120,
0x124,0x127,0x12a,0x12e,0x131,0x134,0x138,0x13b,
0x13f,0x142,0x146,0x149,0x14d,0x150,0x154,0x157,

```
0x15b,0x15e,0x162,0x166,0x169,0x16d,0x171,0x174,
0x178,0x17c,0x180,0x183,0x187,0x18b,0x18f,0x192,
0x196,0x19a,0x19e,0x1a2,0x1a6,0x1aa,0x1ae,0x1b2,
0x1b6,0x1ba,0x1be,0x1c2,0x1c6,0x1ca,0x1ce,0x1d2,
0x1d6,0x1da,0x1de,0x1e2,0x1e6,0x1eb,0x1ef,0x1f3,
0x1f7,0x1fb,0x200,0x204,0x208,0x20c,0x211,0x215,
0x219,0x21e,0x222,0x227,0x22b,0x22f,0x234,0x238,
0x23d,0x241,0x246,0x24a,0x24f,0x253,0x258,0x25c,
0x261,0x265,0x26a,0x26f,0x273,0x278,0x27d,0x281,
0x286,0x28b,0x28f,0x294,0x299,0x29e,0x2a2,0x2a7,
0x2ac,0x2b1,0x2b5,0x2ba,0x2bf,0x2c4,0x2c9,0x2ce,
0x2d3,0x2d8,0x2dc,0x2e1,0x2e6,0x2eb,0x2f0,0x2f5,
0x2fa,0x2ff,0x304,0x309,0x30e,0x313,0x319,0x31e,
0x323,0x328,0x32d,0x332,0x337,0x33c,0x342,0x347,
0x34c,0x351,0x356,0x35c,0x361,0x366,0x36b,0x371,
0x376,0x37b,0x381,0x386,0x38b,0x391,0x396,0x39b,
0x3a1,0x3a6,0x3ac,0x3b1,0x3b6,0x3bc,0x3c1,0x3c7,
0x3cc,0x3d2,0x3d7,0x3dd,0x3e2,0x3e8,0x3ed,0x3f3,
0x3f8,0x3fe,0x403,0x409,0x40f,0x414,0x41a,0x41f,
0x425,0x42b,0x430,0x436,0x43c,0x441,0x447,0x44d,
0x453,0x458,0x45e,0x464,0x469,0x46f,0x475,0x47b,
0x481,0x486,0x48c,0x492,0x498,0x49e,0x4a3,0x4a9,
0x4af,0x4b5,0x4bb,0x4c1,0x4c7,0x4cc,0x4d2,0x4d8,
0x4de,0x4e4,0x4ea,0x4f0,0x4f6,0x4fc,0x502,0x508,
0x50e,0x514,0x51a,0x520,0x526,0x52c,0x532,0x538,
0x53e,0x544,0x54a,0x550,0x556,0x55c,0x562,0x568,
0x56e,0x574,0x57b,0x581,0x587,0x58d,0x593,0x599,
0x59f,0x5a5,0x5ac,0x5b2,0x5b8,0x5be,0x5c4,0x5ca,
0x5d1,0x5d7,0x5dd,0x5e3,0x5e9,0x5f0,0x5f6,0x5fc,
0x602,0x609,0x60f,0x615,0x61b,0x622,0x628,0x62e,
0x634,0x63b,0x641,0x647,0x64d,0x654,0x65a,0x660,
0x667,0x66d,0x673,0x67a,0x680,0x686,0x68c,0x693,
0x699,0x69f,0x6a6,0x6ac,0x6b3,0x6b9,0x6bf,0x6c6,
0x6cc,0x6d2,0x6d9,0x6df,0x6e5,0x6ec,0x6f2,0x6f8,
0x6ff,0x705,0x70c,0x712,0x718,0x71f,0x725,0x72c,
0x732,0x738,0x73f,0x745,0x74c,0x752,0x758,0x75f,
0x765,0x76c,0x772,0x779,0x77f,0x785,0x78c,0x792,
0x799,0x79f,0x7a5,0x7ac,0x7b2,0x7b9,0x7bf,0x7c6,
0x7cc,0x7d2,0x7d9,0x7df,0x7e6,0x7ec,0x7f3,0x7f9};
  if (cond == true) {
          for (k=0; k<25; k++) {
          for(i=0; i<2000; i++) {
          analogWrite(A21, vals[i]*0.3);
          analogWrite(A22, vals[i]*0.3);
```

```
          }
          }
  }
}

void playSineTone2(bool cond) {
  analogWriteResolution(12);
  int i;
  int k;
  int vals[1500] = {0x800,0x808,0x811,0x819,0x822,0x82a,0x833,0x83c,
0x844,0x84d,0x855,0x85e,0x866,0x86f,0x878,0x880,
0x889,0x891,0x89a,0x8a2,0x8ab,0x8b3,0x8bc,0x8c4,
0x8cd,0x8d6,0x8de,0x8e7,0x8ef,0x8f8,0x900,0x909,
0x911,0x91a,0x922,0x92b,0x933,0x93c,0x944,0x94c,
0x955,0x95d,0x966,0x96e,0x977,0x97f,0x988,0x990,
0x998,0x9a1,0x9a9,0x9b2,0x9ba,0x9c2,0x9cb,0x9d3,
0x9db,0x9e4,0x9ec,0x9f4,0x9fd,0xa05,0xa0d,0xa16,
0xa1e,0xa26,0xa2e,0xa37,0xa3f,0xa47,0xa4f,0xa57,
0xa60,0xa68,0xa70,0xa78,0xa80,0xa89,0xa91,0xa99,
0xaa1,0xaa9,0xab1,0xab9,0xac1,0xac9,0xad1,0xad9,
0xae1,0xae9,0xaf1,0xaf9,0xb01,0xb09,0xb11,0xb19,
0xb21,0xb29,0xb31,0xb38,0xb40,0xb48,0xb50,0xb58,
0xb60,0xb67,0xb6f,0xb77,0xb7e,0xb86,0xb8e,0xb96,
0xb9d,0xba5,0xbac,0xbb4,0xbbc,0xbc3,0xbcb,0xbd2,
0xbda,0xbe1,0xbe9,0xbf0,0xbf8,0xbff,0xc07,0xc0e,
0xc15,0xc1d,0xc24,0xc2b,0xc33,0xc3a,0xc41,0xc49,
0xc50,0xc57,0xc5e,0xc65,0xc6d,0xc74,0xc7b,0xc82,
0xc89,0xc90,0xc97,0xc9e,0xca5,0xcac,0xcb3,0xcba,
0xcc1,0xcc8,0xccf,0xcd5,0xcdc,0xce3,0xcea,0xcf1,
0xcf7,0xcfe,0xd05,0xd0b,0xd12,0xd19,0xd1f,0xd26,
0xd2c,0xd33,0xd39,0xd40,0xd46,0xd4d,0xd53,0xd5a,
0xd60,0xd66,0xd6d,0xd73,0xd79,0xd7f,0xd86,0xd8c,
0xd92,0xd98,0xd9e,0xda4,0xdaa,0xdb0,0xdb6,0xdbc,
0xdc2,0xdc8,0xdce,0xdd4,0xdda,0xde0,0xde6,0xdeb,
0xdf1,0xdf7,0xdfd,0xe02,0xe08,0xe0d,0xe13,0xe19,
0xe1e,0xe24,0xe29,0xe2f,0xe34,0xe39,0xe3f,0xe44,
0xe49,0xe4f,0xe54,0xe59,0xe5e,0xe64,0xe69,0xe6e,
0xe73,0xe78,0xe7d,0xe82,0xe87,0xe8c,0xe91,0xe96,
0xe9b,0xe9f,0xea4,0xea9,0xeae,0xeb2,0xeb7,0xebc,
0xec0,0xec5,0xec9,0xece,0xed2,0xed7,0xedb,0xee0,
0xee4,0xee8,0xeed,0xef1,0xef5,0xef9,0xefe,0xf02,
0xf06,0xf0a,0xf0e,0xf12,0xf16,0xf1a,0xf1e,0xf22,
0xf26,0xf29,0xf2d,0xf31,0xf35,0xf38,0xf3c,0xf40,
0xf43,0xf47,0xf4a,0xf4e,0xf51,0xf55,0xf58,0xf5c,
```

0xf5f,0xf62,0xf66,0xf69,0xf6c,0xf6f,0xf72,0xf75,
0xf79,0xf7c,0xf7f,0xf82,0xf84,0xf87,0xf8a,0xf8d,
0xf90,0xf93,0xf95,0xf98,0xf9b,0xf9d,0xfa0,0xfa3,
0xfa5,0xfa8,0xfaa,0xfad,0xfaf,0xfb1,0xfb4,0xfb6,
0xfb8,0xfba,0xfbd,0xfbf,0xfc1,0xfc3,0xfc5,0xfc7,
0xfc9,0xfcb,0xfcd,0xfcf,0xfd0,0xfd2,0xfd4,0xfd6,
0xfd7,0xfd9,0xfdb,0xfdc,0xfde,0xfdf,0xfe1,0xfe2,
0xfe4,0xfe5,0xfe6,0xfe8,0xfe9,0xfea,0xfeb,0xfed,
0xfee,0xfef,0xff0,0xff1,0xff2,0xff3,0xff4,0xff5,
0xff6,0xff6,0xff7,0xff8,0xff9,0xff9,0xffa,0xffa,
0xffb,0xffb,0xffc,0xffc,0xffd,0xffd,0xffe,0xffe,
0xffe,0xffe,0xfff,0xfff,0xfff,0xfff,0xfff,0xfff,
0xfff,0xfff,0xfff,0xfff,0xfff,0xffe,0xffe,0xffe,
0xffe,0xffd,0xffd,0xffc,0xffc,0xffb,0xffb,0xffa,
0xffa,0xff9,0xff9,0xff8,0xff7,0xff6,0xff6,0xff5,
0xff4,0xff3,0xff2,0xff1,0xff0,0xfef,0xfee,0xfed,
0xfeb,0xfea,0xfe9,0xfe8,0xfe6,0xfe5,0xfe4,0xfe2,
0xfe1,0xfdf,0xfde,0xfdc,0xfdb,0xfd9,0xfd7,0xfd6,
0xfd4,0xfd2,0xfd0,0xfcf,0xfcd,0xfcb,0xfc9,0xfc7,
0xfc5,0xfc3,0xfc1,0xfbf,0xfbd,0xfba,0xfb8,0xfb6,
0xfb4,0xfb1,0xfaf,0xfad,0xfaa,0xfa8,0xfa5,0xfa3,
0xfa0,0xf9d,0xf9b,0xf98,0xf95,0xf93,0xf90,0xf8d,
0xf8a,0xf87,0xf84,0xf82,0xf7f,0xf7c,0xf79,0xf75,
0xf72,0xf6f,0xf6c,0xf69,0xf66,0xf62,0xf5f,0xf5c,
0xf58,0xf55,0xf51,0xf4e,0xf4a,0xf47,0xf43,0xf40,
0xf3c,0xf38,0xf35,0xf31,0xf2d,0xf29,0xf26,0xf22,
0xf1e,0xf1a,0xf16,0xf12,0xf0e,0xf0a,0xf06,0xf02,
0xefe,0xef9,0xef5,0xef1,0xeed,0xee8,0xee4,0xee0,
0xedb,0xed7,0xed2,0xece,0xec9,0xec5,0xec0,0xebc,
0xeb7,0xeb2,0xeae,0xea9,0xea4,0xe9f,0xe9b,0xe96,
0xe91,0xe8c,0xe87,0xe82,0xe7d,0xe78,0xe73,0xe6e,
0xe69,0xe64,0xe5e,0xe59,0xe54,0xe4f,0xe49,0xe44,
0xe3f,0xe39,0xe34,0xe2f,0xe29,0xe24,0xe1e,0xe19,
0xe13,0xe0d,0xe08,0xe02,0xdfd,0xdf7,0xdf1,0xdeb,
0xde6,0xde0,0xdda,0xdd4,0xdce,0xdc8,0xdc2,0xdbc,
0xdb6,0xdb0,0xdaa,0xda4,0xd9e,0xd98,0xd92,0xd8c,
0xd86,0xd7f,0xd79,0xd73,0xd6d,0xd66,0xd60,0xd5a,
0xd53,0xd4d,0xd46,0xd40,0xd39,0xd33,0xd2c,0xd26,
0xd1f,0xd19,0xd12,0xd0b,0xd05,0xcfe,0xcf7,0xcf1,
0xcea,0xce3,0xcdc,0xcd5,0xccf,0xcc8,0xcc1,0xcba,
0xcb3,0xcac,0xca5,0xc9e,0xc97,0xc90,0xc89,0xc82,
0xc7b,0xc74,0xc6d,0xc65,0xc5e,0xc57,0xc50,0xc49,
0xc41,0xc3a,0xc33,0xc2b,0xc24,0xc1d,0xc15,0xc0e,
0xc07,0xbff,0xbf8,0xbf0,0xbe9,0xbe1,0xbda,0xbd2,

0xbcb,0xbc3,0xbbc,0xbb4,0xbac,0xba5,0xb9d,0xb96,
0xb8e,0xb86,0xb7e,0xb77,0xb6f,0xb67,0xb60,0xb58,
0xb50,0xb48,0xb40,0xb38,0xb31,0xb29,0xb21,0xb19,
0xb11,0xb09,0xb01,0xaf9,0xaf1,0xae9,0xae1,0xad9,
0xad1,0xac9,0xac1,0xab9,0xab1,0xaa9,0xaa1,0xa99,
0xa91,0xa89,0xa80,0xa78,0xa70,0xa68,0xa60,0xa57,
0xa4f,0xa47,0xa3f,0xa37,0xa2e,0xa26,0xa1e,0xa16,
0xa0d,0xa05,0x9fd,0x9f4,0x9ec,0x9e4,0x9db,0x9d3,
0x9cb,0x9c2,0x9ba,0x9b2,0x9a9,0x9a1,0x998,0x990,
0x988,0x97f,0x977,0x96e,0x966,0x95d,0x955,0x94c,
0x944,0x93c,0x933,0x92b,0x922,0x91a,0x911,0x909,
0x900,0x8f8,0x8ef,0x8e7,0x8de,0x8d6,0x8cd,0x8c4,
0x8bc,0x8b3,0x8ab,0x8a2,0x89a,0x891,0x889,0x880,
0x878,0x86f,0x866,0x85e,0x855,0x84d,0x844,0x83c,
0x833,0x82a,0x822,0x819,0x811,0x808,0x800,0x7f7,
0x7ee,0x7e6,0x7dd,0x7d5,0x7cc,0x7c3,0x7bb,0x7b2,
0x7aa,0x7a1,0x799,0x790,0x787,0x77f,0x776,0x76e,
0x765,0x75d,0x754,0x74c,0x743,0x73b,0x732,0x729,
0x721,0x718,0x710,0x707,0x6ff,0x6f6,0x6ee,0x6e5,
0x6dd,0x6d4,0x6cc,0x6c3,0x6bb,0x6b3,0x6aa,0x6a2,
0x699,0x691,0x688,0x680,0x677,0x66f,0x667,0x65e,
0x656,0x64d,0x645,0x63d,0x634,0x62c,0x624,0x61b,
0x613,0x60b,0x602,0x5fa,0x5f2,0x5e9,0x5e1,0x5d9,
0x5d1,0x5c8,0x5c0,0x5b8,0x5b0,0x5a8,0x59f,0x597,
0x58f,0x587,0x57f,0x576,0x56e,0x566,0x55e,0x556,
0x54e,0x546,0x53e,0x536,0x52e,0x526,0x51e,0x516,
0x50e,0x506,0x4fe,0x4f6,0x4ee,0x4e6,0x4de,0x4d6,
0x4ce,0x4c7,0x4bf,0x4b7,0x4af,0x4a7,0x49f,0x498,
0x490,0x488,0x481,0x479,0x471,0x469,0x462,0x45a,
0x453,0x44b,0x443,0x43c,0x434,0x42d,0x425,0x41e,
0x416,0x40f,0x407,0x400,0x3f8,0x3f1,0x3ea,0x3e2,
0x3db,0x3d4,0x3cc,0x3c5,0x3be,0x3b6,0x3af,0x3a8,
0x3a1,0x39a,0x392,0x38b,0x384,0x37d,0x376,0x36f,
0x368,0x361,0x35a,0x353,0x34c,0x345,0x33e,0x337,
0x330,0x32a,0x323,0x31c,0x315,0x30e,0x308,0x301,
0x2fa,0x2f4,0x2ed,0x2e6,0x2e0,0x2d9,0x2d3,0x2cc,
0x2c6,0x2bf,0x2b9,0x2b2,0x2ac,0x2a5,0x29f,0x299,
0x292,0x28c,0x286,0x280,0x279,0x273,0x26d,0x267,
0x261,0x25b,0x255,0x24f,0x249,0x243,0x23d,0x237,
0x231,0x22b,0x225,0x21f,0x219,0x214,0x20e,0x208,
0x202,0x1fd,0x1f7,0x1f2,0x1ec,0x1e6,0x1e1,0x1db,
0x1d6,0x1d0,0x1cb,0x1c6,0x1c0,0x1bb,0x1b6,0x1b0,
0x1ab,0x1a6,0x1a1,0x19b,0x196,0x191,0x18c,0x187,
0x182,0x17d,0x178,0x173,0x16e,0x169,0x164,0x160,

```
0x15b,0x156,0x151,0x14d,0x148,0x143,0x13f,0x13a,
0x136,0x131,0x12d,0x128,0x124,0x11f,0x11b,0x117,
0x112,0x10e,0x10a,0x106,0x101,0xfd,0xf9,0xf5,
0xf1,0xed,0xe9,0xe5,0xe1,0xdd,0xd9,0xd6,
0xd2,0xce,0xca,0xc7,0xc3,0xbf,0xbc,0xb8,
0xb5,0xb1,0xae,0xaa,0xa7,0xa3,0xa0,0x9d,
0x99,0x96,0x93,0x90,0x8d,0x8a,0x86,0x83,
0x80,0x7d,0x7b,0x78,0x75,0x72,0x6f,0x6c,
0x6a,0x67,0x64,0x62,0x5f,0x5c,0x5a,0x57,
0x55,0x52,0x50,0x4e,0x4b,0x49,0x47,0x45,
0x42,0x40,0x3e,0x3c,0x3a,0x38,0x36,0x34,
0x32,0x30,0x2f,0x2d,0x2b,0x29,0x28,0x26,
0x24,0x23,0x21,0x20,0x1e,0x1d,0x1b,0x1a,
0x19,0x17,0x16,0x15,0x14,0x12,0x11,0x10,
0xf,0xe,0xd,0xc,0xb,0xa,0x9,0x9,
0x8,0x7,0x6,0x6,0x5,0x5,0x4,0x4,
0x3,0x3,0x2,0x2,0x1,0x1,0x1,0x1,
0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,
0x0,0x0,0x0,0x0,0x1,0x1,0x1,0x1,0x2,
0x2,0x3,0x3,0x4,0x4,0x5,0x5,0x6,
0x6,0x7,0x8,0x9,0x9,0xa,0xb,0xc,
0xd,0xe,0xf,0x10,0x11,0x12,0x14,0x15,
0x16,0x17,0x19,0x1a,0x1b,0x1d,0x1e,0x20,
0x21,0x23,0x24,0x26,0x28,0x29,0x2b,0x2d,
0x2f,0x30,0x32,0x34,0x36,0x38,0x3a,0x3c,
0x3e,0x40,0x42,0x45,0x47,0x49,0x4b,0x4e,
0x50,0x52,0x55,0x57,0x5a,0x5c,0x5f,0x62,
0x64,0x67,0x6a,0x6c,0x6f,0x72,0x75,0x78,
0x7b,0x7d,0x80,0x83,0x86,0x8a,0x8d,0x90,
0x93,0x96,0x99,0x9d,0xa0,0xa3,0xa7,0xaa,
0xae,0xb1,0xb5,0xb8,0xbc,0xbf,0xc3,0xc7,
0xca,0xce,0xd2,0xd6,0xd9,0xdd,0xe1,0xe5,
0xe9,0xed,0xf1,0xf5,0xf9,0xfd,0x101,0x106,
0x10a,0x10e,0x112,0x117,0x11b,0x11f,0x124,0x128,
0x12d,0x131,0x136,0x13a,0x13f,0x143,0x148,0x14d,
0x151,0x156,0x15b,0x160,0x164,0x169,0x16e,0x173,
0x178,0x17d,0x182,0x187,0x18c,0x191,0x196,0x19b,
0x1a1,0x1a6,0x1ab,0x1b0,0x1b6,0x1bb,0x1c0,0x1c6,
0x1cb,0x1d0,0x1d6,0x1db,0x1e1,0x1e6,0x1ec,0x1f2,
0x1f7,0x1fd,0x202,0x208,0x20e,0x214,0x219,0x21f,
0x225,0x22b,0x231,0x237,0x23d,0x243,0x249,0x24f,
0x255,0x25b,0x261,0x267,0x26d,0x273,0x279,0x280,
0x286,0x28c,0x292,0x299,0x29f,0x2a5,0x2ac,0x2b2,
0x2b9,0x2bf,0x2c6,0x2cc,0x2d3,0x2d9,0x2e0,0x2e6,
```

```
0x2ed,0x2f4,0x2fa,0x301,0x308,0x30e,0x315,0x31c,
0x323,0x32a,0x330,0x337,0x33e,0x345,0x34c,0x353,
0x35a,0x361,0x368,0x36f,0x376,0x37d,0x384,0x38b,
0x392,0x39a,0x3a1,0x3a8,0x3af,0x3b6,0x3be,0x3c5,
0x3cc,0x3d4,0x3db,0x3e2,0x3ea,0x3f1,0x3f8,0x400,
0x407,0x40f,0x416,0x41e,0x425,0x42d,0x434,0x43c,
0x443,0x44b,0x453,0x45a,0x462,0x469,0x471,0x479,
0x481,0x488,0x490,0x498,0x49f,0x4a7,0x4af,0x4b7,
0x4bf,0x4c7,0x4ce,0x4d6,0x4de,0x4e6,0x4ee,0x4f6,
0x4fe,0x506,0x50e,0x516,0x51e,0x526,0x52e,0x536,
0x53e,0x546,0x54e,0x556,0x55e,0x566,0x56e,0x576,
0x57f,0x587,0x58f,0x597,0x59f,0x5a8,0x5b0,0x5b8,
0x5c0,0x5c8,0x5d1,0x5d9,0x5e1,0x5e9,0x5f2,0x5fa,
0x602,0x60b,0x613,0x61b,0x624,0x62c,0x634,0x63d,
0x645,0x64d,0x656,0x65e,0x667,0x66f,0x677,0x680,
0x688,0x691,0x699,0x6a2,0x6aa,0x6b3,0x6bb,0x6c3,
0x6cc,0x6d4,0x6dd,0x6e5,0x6ee,0x6f6,0x6ff,0x707,
0x710,0x718,0x721,0x729,0x732,0x73b,0x743,0x74c,
0x754,0x75d,0x765,0x76e,0x776,0x77f,0x787,0x790,
0x799,0x7a1,0x7aa,0x7b2,0x7bb,0x7c3,0x7cc,0x7d5,
0x7dd,0x7e6,0x7ee,0x7f7};
  if (cond == true) {
          for (k=0; k<25; k++) {
          for(i=0; i<1500; i++) {
          analogWrite(A21, vals[i]*0.3);
          analogWrite(A22, vals[i]*0.3);
          }
          }
  }

}

void playSineTone3(bool cond, int g) {
  int i;
  int k;
  int vals [1751] = {0x800,0x807,0x80e,0x816,0x81d,0x824,0x82c,0x833,
0x83a,0x842,0x849,0x850,0x858,0x85f,0x866,0x86e,
0x875,0x87c,0x884,0x88b,0x892,0x89a,0x8a1,0x8a8,
0x8b0,0x8b7,0x8be,0x8c6,0x8cd,0x8d4,0x8dc,0x8e3,
0x8ea,0x8f2,0x8f9,0x900,0x907,0x90f,0x916,0x91d,
0x925,0x92c,0x933,0x93a,0x942,0x949,0x950,0x957,
0x95f,0x966,0x96d,0x974,0x97c,0x983,0x98a,0x991,
0x998,0x9a0,0x9a7,0x9ae,0x9b5,0x9bc,0x9c4,0x9cb,
0x9d2,0x9d9,0x9e0,0x9e7,0x9ee,0x9f6,0x9fd,0xa04,
```

0xa0b,0xa12,0xa19,0xa20,0xa27,0xa2e,0xa35,0xa3c,
0xa44,0xa4b,0xa52,0xa59,0xa60,0xa67,0xa6e,0xa75,
0xa7c,0xa83,0xa8a,0xa91,0xa98,0xa9f,0xaa5,0xaac,
0xab3,0xaba,0xac1,0xac8,0xacf,0xad6,0xadd,0xae4,
0xaea,0xaf1,0xaf8,0xaff,0xb06,0xb0c,0xb13,0xb1a,
0xb21,0xb28,0xb2e,0xb35,0xb3c,0xb43,0xb49,0xb50,
0xb57,0xb5d,0xb64,0xb6b,0xb71,0xb78,0xb7e,0xb85,
0xb8c,0xb92,0xb99,0xb9f,0xba6,0xbac,0xbb3,0xbba,
0xbc0,0xbc7,0xbcd,0xbd3,0xbda,0xbe0,0xbe7,0xbed,
0xbf4,0xbfa,0xc00,0xc07,0xc0d,0xc13,0xc1a,0xc20,
0xc26,0xc2d,0xc33,0xc39,0xc3f,0xc46,0xc4c,0xc52,
0xc58,0xc5e,0xc64,0xc6b,0xc71,0xc77,0xc7d,0xc83,
0xc89,0xc8f,0xc95,0xc9b,0xca1,0xca7,0xcad,0xcb3,
0xcb9,0xcbf,0xcc5,0xccb,0xcd1,0xcd6,0xcdc,0xce2,
0xce8,0xcee,0xcf3,0xcf9,0xcff,0xd05,0xd0a,0xd10,
0xd16,0xd1b,0xd21,0xd27,0xd2c,0xd32,0xd38,0xd3d,
0xd43,0xd48,0xd4e,0xd53,0xd59,0xd5e,0xd64,0xd69,
0xd6e,0xd74,0xd79,0xd7e,0xd84,0xd89,0xd8e,0xd94,
0xd99,0xd9e,0xda3,0xda9,0xdae,0xdb3,0xdb8,0xdbd,
0xdc2,0xdc7,0xdcc,0xdd2,0xdd7,0xddc,0xde1,0xde6,
0xdeb,0xdef,0xdf4,0xdf9,0xdfe,0xe03,0xe08,0xe0d,
0xe11,0xe16,0xe1b,0xe20,0xe24,0xe29,0xe2e,0xe32,
0xe37,0xe3c,0xe40,0xe45,0xe49,0xe4e,0xe52,0xe57,
0xe5b,0xe60,0xe64,0xe69,0xe6d,0xe71,0xe76,0xe7a,
0xe7e,0xe83,0xe87,0xe8b,0xe8f,0xe94,0xe98,0xe9c,
0xea0,0xea4,0xea8,0xeac,0xeb0,0xeb4,0xeb8,0xebc,
0xec0,0xec4,0xec8,0xecc,0xed0,0xed4,0xed7,0xedb,
0xedf,0xee3,0xee7,0xeea,0xeee,0xef2,0xef5,0xef9,
0xefc,0xf00,0xf04,0xf07,0xf0b,0xf0e,0xf11,0xf15,
0xf18,0xf1c,0xf1f,0xf22,0xf26,0xf29,0xf2c,0xf2f,
0xf33,0xf36,0xf39,0xf3c,0xf3f,0xf42,0xf45,0xf48,
0xf4b,0xf4e,0xf51,0xf54,0xf57,0xf5a,0xf5d,0xf60,
0xf63,0xf66,0xf68,0xf6b,0xf6e,0xf71,0xf73,0xf76,
0xf79,0xf7b,0xf7e,0xf80,0xf83,0xf85,0xf88,0xf8a,
0xf8d,0xf8f,0xf92,0xf94,0xf96,0xf99,0xf9b,0xf9d,
0xf9f,0xfa1,0xfa4,0xfa6,0xfa8,0xfaa,0xfac,0xfae,
0xfb0,0xfb2,0xfb4,0xfb6,0xfb8,0xfba,0xfbc,0xfbe,
0xfc0,0xfc1,0xfc3,0xfc5,0xfc7,0xfc8,0xfca,0xfcc,
0xfcd,0xfcf,0xfd0,0xfd2,0xfd4,0xfd5,0xfd6,0xfd8,
0xfd9,0xfdb,0xfdc,0xfdd,0xfdf,0xfe0,0xfe1,0xfe3,
0xfe4,0xfe5,0xfe6,0xfe7,0xfe8,0xfe9,0xfea,0xfeb,
0xfec,0xfed,0xfee,0xfef,0xff0,0xff1,0xff2,0xff3,
0xff4,0xff4,0xff5,0xff6,0xff6,0xff7,0xff8,0xff8,
0xff9,0xff9,0xffa,0xffa,0xffb,0xffb,0xffc,0xffc,

0xffd,0xffd,0xffd,0xffe,0xffe,0xffe,0xffe,0xffe,
0xfff,0xfff,0xfff,0xfff,0xfff,0xfff,0xfff,0xfff,
0xfff,0xfff,0xfff,0xfff,0xffe,0xffe,0xffe,0xffe,
0xffe,0xffd,0xffd,0xffd,0xffc,0xffc,0xffb,0xffb,
0xffa,0xffa,0xff9,0xff9,0xff8,0xff8,0xff7,0xff6,
0xff6,0xff5,0xff4,0xff4,0xff3,0xff2,0xff1,0xff0,
0xfef,0xfee,0xfed,0xfec,0xfeb,0xfea,0xfe9,0xfe8,
0xfe7,0xfe6,0xfe5,0xfe4,0xfe3,0xfe1,0xfe0,0xfdf,
0xfdd,0xfdc,0xfdb,0xfd9,0xfd8,0xfd6,0xfd5,0xfd4,
0xfd2,0xfd0,0xfcf,0xfcd,0xfcc,0xfca,0xfc8,0xfc7,
0xfc5,0xfc3,0xfc1,0xfc0,0xfbe,0xfbc,0xfba,0xfb8,
0xfb6,0xfb4,0xfb2,0xfb0,0xfae,0xfac,0xfaa,0xfa8,
0xfa6,0xfa4,0xfa1,0xf9f,0xf9d,0xf9b,0xf99,0xf96,
0xf94,0xf92,0xf8f,0xf8d,0xf8a,0xf88,0xf85,0xf83,
0xf80,0xf7e,0xf7b,0xf79,0xf76,0xf73,0xf71,0xf6e,
0xf6b,0xf68,0xf66,0xf63,0xf60,0xf5d,0xf5a,0xf57,
0xf54,0xf51,0xf4e,0xf4b,0xf48,0xf45,0xf42,0xf3f,
0xf3c,0xf39,0xf36,0xf33,0xf2f,0xf2c,0xf29,0xf26,
0xf22,0xf1f,0xf1c,0xf18,0xf15,0xf11,0xf0e,0xf0b,
0xf07,0xf04,0xf00,0xefc,0xef9,0xef5,0xef2,0xeee,
0xeea,0xee7,0xee3,0xedf,0xedb,0xed7,0xed4,0xed0,
0xecc,0xec8,0xec4,0xec0,0xebc,0xeb8,0xeb4,0xeb0,
0xeac,0xea8,0xea4,0xea0,0xe9c,0xe98,0xe94,0xe8f,
0xe8b,0xe87,0xe83,0xe7e,0xe7a,0xe76,0xe71,0xe6d,
0xe69,0xe64,0xe60,0xe5b,0xe57,0xe52,0xe4e,0xe49,
0xe45,0xe40,0xe3c,0xe37,0xe32,0xe2e,0xe29,0xe24,
0xe20,0xe1b,0xe16,0xe11,0xe0d,0xe08,0xe03,0xdfe,
0xdf9,0xdf4,0xdef,0xdeb,0xde6,0xde1,0xddc,0xdd7,
0xdd2,0xdcc,0xdc7,0xdc2,0xdbd,0xdb8,0xdb3,0xdae,
0xda9,0xda3,0xd9e,0xd99,0xd94,0xd8e,0xd89,0xd84,
0xd7e,0xd79,0xd74,0xd6e,0xd69,0xd64,0xd5e,0xd59,
0xd53,0xd4e,0xd48,0xd43,0xd3d,0xd38,0xd32,0xd2c,
0xd27,0xd21,0xd1b,0xd16,0xd10,0xd0a,0xd05,0xcff,
0xcf9,0xcf3,0xcee,0xce8,0xce2,0xcdc,0xcd6,0xcd1,
0xccb,0xcc5,0xcbf,0xcb9,0xcb3,0xcad,0xca7,0xca1,
0xc9b,0xc95,0xc8f,0xc89,0xc83,0xc7d,0xc77,0xc71,
0xc6b,0xc64,0xc5e,0xc58,0xc52,0xc4c,0xc46,0xc3f,
0xc39,0xc33,0xc2d,0xc26,0xc20,0xc1a,0xc13,0xc0d,
0xc07,0xc00,0xbfa,0xbf4,0xbed,0xbe7,0xbe0,0xbda,
0xbd3,0xbcd,0xbc7,0xbc0,0xbba,0xbb3,0xbac,0xba6,
0xb9f,0xb99,0xb92,0xb8c,0xb85,0xb7e,0xb78,0xb71,
0xb6b,0xb64,0xb5d,0xb57,0xb50,0xb49,0xb43,0xb3c,
0xb35,0xb2e,0xb28,0xb21,0xb1a,0xb13,0xb0c,0xb06,
0xaff,0xaf8,0xaf1,0xaea,0xae4,0xadd,0xad6,0xacf,

0xac8,0xac1,0xaba,0xab3,0xaac,0xaa5,0xa9f,0xa98,
0xa91,0xa8a,0xa83,0xa7c,0xa75,0xa6e,0xa67,0xa60,
0xa59,0xa52,0xa4b,0xa44,0xa3c,0xa35,0xa2e,0xa27,
0xa20,0xa19,0xa12,0xa0b,0xa04,0x9fd,0x9f6,0x9ee,
0x9e7,0x9e0,0x9d9,0x9d2,0x9cb,0x9c4,0x9bc,0x9b5,
0x9ae,0x9a7,0x9a0,0x998,0x991,0x98a,0x983,0x97c,
0x974,0x96d,0x966,0x95f,0x957,0x950,0x949,0x942,
0x93a,0x933,0x92c,0x925,0x91d,0x916,0x90f,0x907,
0x900,0x8f9,0x8f2,0x8ea,0x8e3,0x8dc,0x8d4,0x8cd,
0x8c6,0x8be,0x8b7,0x8b0,0x8a8,0x8a1,0x89a,0x892,
0x88b,0x884,0x87c,0x875,0x86e,0x866,0x85f,0x858,
0x850,0x849,0x842,0x83a,0x833,0x82c,0x824,0x81d,
0x816,0x80e,0x807,0x800,0x7f8,0x7f1,0x7e9,0x7e2,
0x7db,0x7d3,0x7cc,0x7c5,0x7bd,0x7b6,0x7af,0x7a7,
0x7a0,0x799,0x791,0x78a,0x783,0x77b,0x774,0x76d,
0x765,0x75e,0x757,0x74f,0x748,0x741,0x739,0x732,
0x72b,0x723,0x71c,0x715,0x70d,0x706,0x6ff,0x6f8,
0x6f0,0x6e9,0x6e2,0x6da,0x6d3,0x6cc,0x6c5,0x6bd,
0x6b6,0x6af,0x6a8,0x6a0,0x699,0x692,0x68b,0x683,
0x67c,0x675,0x66e,0x667,0x65f,0x658,0x651,0x64a,
0x643,0x63b,0x634,0x62d,0x626,0x61f,0x618,0x611,
0x609,0x602,0x5fb,0x5f4,0x5ed,0x5e6,0x5df,0x5d8,
0x5d1,0x5ca,0x5c3,0x5bb,0x5b4,0x5ad,0x5a6,0x59f,
0x598,0x591,0x58a,0x583,0x57c,0x575,0x56e,0x567,
0x560,0x55a,0x553,0x54c,0x545,0x53e,0x537,0x530,
0x529,0x522,0x51b,0x515,0x50e,0x507,0x500,0x4f9,
0x4f3,0x4ec,0x4e5,0x4de,0x4d7,0x4d1,0x4ca,0x4c3,
0x4bc,0x4b6,0x4af,0x4a8,0x4a2,0x49b,0x494,0x48e,
0x487,0x481,0x47a,0x473,0x46d,0x466,0x460,0x459,
0x453,0x44c,0x445,0x43f,0x438,0x432,0x42c,0x425,
0x41f,0x418,0x412,0x40b,0x405,0x3ff,0x3f8,0x3f2,
0x3ec,0x3e5,0x3df,0x3d9,0x3d2,0x3cc,0x3c6,0x3c0,
0x3b9,0x3b3,0x3ad,0x3a7,0x3a1,0x39b,0x394,0x38e,
0x388,0x382,0x37c,0x376,0x370,0x36a,0x364,0x35e,
0x358,0x352,0x34c,0x346,0x340,0x33a,0x334,0x32e,
0x329,0x323,0x31d,0x317,0x311,0x30c,0x306,0x300,
0x2fa,0x2f5,0x2ef,0x2e9,0x2e4,0x2de,0x2d8,0x2d3,
0x2cd,0x2c7,0x2c2,0x2bc,0x2b7,0x2b1,0x2ac,0x2a6,
0x2a1,0x29b,0x296,0x291,0x28b,0x286,0x281,0x27b,
0x276,0x271,0x26b,0x266,0x261,0x25c,0x256,0x251,
0x24c,0x247,0x242,0x23d,0x238,0x233,0x22d,0x228,
0x223,0x21e,0x219,0x214,0x210,0x20b,0x206,0x201,
0x1fc,0x1f7,0x1f2,0x1ee,0x1e9,0x1e4,0x1df,0x1db,
0x1d6,0x1d1,0x1cd,0x1c8,0x1c3,0x1bf,0x1ba,0x1b6,

0x1b1,0x1ad,0x1a8,0x1a4,0x19f,0x19b,0x196,0x192,
0x18e,0x189,0x185,0x181,0x17c,0x178,0x174,0x170,
0x16b,0x167,0x163,0x15f,0x15b,0x157,0x153,0x14f,
0x14b,0x147,0x143,0x13f,0x13b,0x137,0x133,0x12f,
0x12b,0x128,0x124,0x120,0x11c,0x118,0x115,0x111,
0x10d,0x10a,0x106,0x103,0xff,0xfb,0xf8,0xf4,
0xf1,0xee,0xea,0xe7,0xe3,0xe0,0xdd,0xd9,
0xd6,0xd3,0xd0,0xcc,0xc9,0xc6,0xc3,0xc0,
0xbd,0xba,0xb7,0xb4,0xb1,0xae,0xab,0xa8,
0xa5,0xa2,0x9f,0x9c,0x99,0x97,0x94,0x91,
0x8e,0x8c,0x89,0x86,0x84,0x81,0x7f,0x7c,
0x7a,0x77,0x75,0x72,0x70,0x6d,0x6b,0x69,
0x66,0x64,0x62,0x60,0x5e,0x5b,0x59,0x57,
0x55,0x53,0x51,0x4f,0x4d,0x4b,0x49,0x47,
0x45,0x43,0x41,0x3f,0x3e,0x3c,0x3a,0x38,
0x37,0x35,0x33,0x32,0x30,0x2f,0x2d,0x2b,
0x2a,0x29,0x27,0x26,0x24,0x23,0x22,0x20,
0x1f,0x1e,0x1c,0x1b,0x1a,0x19,0x18,0x17,
0x16,0x15,0x14,0x13,0x12,0x11,0x10,0xf,
0xe,0xd,0xc,0xb,0xb,0xa,0x9,0x9,
0x8,0x7,0x7,0x6,0x6,0x5,0x5,0x4,
0x4,0x3,0x3,0x2,0x2,0x2,0x1,0x1,
0x1,0x1,0x1,0x0,0x0,0x0,0x0,0x0,
0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x1,
0x1,0x1,0x1,0x1,0x2,0x2,0x2,0x3,
0x3,0x4,0x4,0x5,0x5,0x6,0x6,0x7,
0x7,0x8,0x9,0x9,0xa,0xb,0xb,0xc,
0xd,0xe,0xf,0x10,0x11,0x12,0x13,0x14,
0x15,0x16,0x17,0x18,0x19,0x1a,0x1b,0x1c,
0x1e,0x1f,0x20,0x22,0x23,0x24,0x26,0x27,
0x29,0x2a,0x2b,0x2d,0x2f,0x30,0x32,0x33,
0x35,0x37,0x38,0x3a,0x3c,0x3e,0x3f,0x41,
0x43,0x45,0x47,0x49,0x4b,0x4d,0x4f,0x51,
0x53,0x55,0x57,0x59,0x5b,0x5e,0x60,0x62,
0x64,0x66,0x69,0x6b,0x6d,0x70,0x72,0x75,
0x77,0x7a,0x7c,0x7f,0x81,0x84,0x86,0x89,
0x8c,0x8e,0x91,0x94,0x97,0x99,0x9c,0x9f,
0xa2,0xa5,0xa8,0xab,0xae,0xb1,0xb4,0xb7,
0xba,0xbd,0xc0,0xc3,0xc6,0xc9,0xcc,0xd0,
0xd3,0xd6,0xd9,0xdd,0xe0,0xe3,0xe7,0xea,
0xee,0xf1,0xf4,0xf8,0xfb,0xff,0x103,0x106,
0x10a,0x10d,0x111,0x115,0x118,0x11c,0x120,0x124,
0x128,0x12b,0x12f,0x133,0x137,0x13b,0x13f,0x143,
0x147,0x14b,0x14f,0x153,0x157,0x15b,0x15f,0x163,

```
0x167,0x16b,0x170,0x174,0x178,0x17c,0x181,0x185,
0x189,0x18e,0x192,0x196,0x19b,0x19f,0x1a4,0x1a8,
0x1ad,0x1b1,0x1b6,0x1ba,0x1bf,0x1c3,0x1c8,0x1cd,
0x1d1,0x1d6,0x1db,0x1df,0x1e4,0x1e9,0x1ee,0x1f2,
0x1f7,0x1fc,0x201,0x206,0x20b,0x210,0x214,0x219,
0x21e,0x223,0x228,0x22d,0x233,0x238,0x23d,0x242,
0x247,0x24c,0x251,0x256,0x25c,0x261,0x266,0x26b,
0x271,0x276,0x27b,0x281,0x286,0x28b,0x291,0x296,
0x29b,0x2a1,0x2a6,0x2ac,0x2b1,0x2b7,0x2bc,0x2c2,
0x2c7,0x2cd,0x2d3,0x2d8,0x2de,0x2e4,0x2e9,0x2ef,
0x2f5,0x2fa,0x300,0x306,0x30c,0x311,0x317,0x31d,
0x323,0x329,0x32e,0x334,0x33a,0x340,0x346,0x34c,
0x352,0x358,0x35e,0x364,0x36a,0x370,0x376,0x37c,
0x382,0x388,0x38e,0x394,0x39b,0x3a1,0x3a7,0x3ad,
0x3b3,0x3b9,0x3c0,0x3c6,0x3cc,0x3d2,0x3d9,0x3df,
0x3e5,0x3ec,0x3f2,0x3f8,0x3ff,0x405,0x40b,0x412,
0x418,0x41f,0x425,0x42c,0x432,0x438,0x43f,0x445,
0x44c,0x453,0x459,0x460,0x466,0x46d,0x473,0x47a,
0x481,0x487,0x48e,0x494,0x49b,0x4a2,0x4a8,0x4af,
0x4b6,0x4bc,0x4c3,0x4ca,0x4d1,0x4d7,0x4de,0x4e5,
0x4ec,0x4f3,0x4f9,0x500,0x507,0x50e,0x515,0x51b,
0x522,0x529,0x530,0x537,0x53e,0x545,0x54c,0x553,
0x55a,0x560,0x567,0x56e,0x575,0x57c,0x583,0x58a,
0x591,0x598,0x59f,0x5a6,0x5ad,0x5b4,0x5bb,0x5c3,
0x5ca,0x5d1,0x5d8,0x5df,0x5e6,0x5ed,0x5f4,0x5fb,
0x602,0x609,0x611,0x618,0x61f,0x626,0x62d,0x634,
0x63b,0x643,0x64a,0x651,0x658,0x65f,0x667,0x66e,
0x675,0x67c,0x683,0x68b,0x692,0x699,0x6a0,0x6a8,
0x6af,0x6b6,0x6bd,0x6c5,0x6cc,0x6d3,0x6da,0x6e2,
0x6e9,0x6f0,0x6f8,0x6ff,0x706,0x70d,0x715,0x71c,
0x723,0x72b,0x732,0x739,0x741,0x748,0x74f,0x757,
0x75e,0x765,0x76d,0x774,0x77b,0x783,0x78a,0x791,
0x799,0x7a0,0x7a7,0x7af,0x7b6,0x7bd,0x7c5,0x7cc,
0x7d3,0x7db,0x7e2,0x7e9,0x7f1,0x7f8,0x800};
if (cond == true) {
  if(g == 1) {
          for (k=0; k<25; k++) {
          for(i=0; i<1750; i++) {
          analogWrite(A21, vals[i]*0.2);
          }
          }
  }
  else if(g == 0) {
          for (k=0; k<25; k++) {
```

```
          for(i=0; i<1750; i++) {
          analogWrite(A22, vals[i]*0.3);
          }
          }
  }

  }
}

//Create a new instance of the library
//Call the sensor "sensor"
//The model of the sensor is "GP2YA41SK0F"
//The sensor output pin is attached to the pin A0
SharpIR sensorF(SharpIR::GP2Y0A41SK0F, A6);
SharpIR sensorB(SharpIR::GP2Y0A41SK0F, A7);
SharpIR sensorR(SharpIR::GP2Y0A41SK0F, A8);
SharpIR sensorL(SharpIR::GP2Y0A41SK0F, A9);

unsigned long previousMillis = 0;
const long len0 = 50;
bool truetoneF = false;
bool truetoneB = false;
bool truetoneR = false;
bool truetoneL = false;



void setup() {
  Serial.begin(115200);
  pinMode(A21, OUTPUT);
  pinMode(A22, OUTPUT);
  pinMode(A6, INPUT);
  pinMode(A7, INPUT);
  pinMode(A8, INPUT);
  pinMode(A9, INPUT);
}


void loop()
{
  unsigned long currentMillis = millis();

  int dF = sensorF.getDistance(); //Calculate the distance in centimeters and store the value in a variable
  int dB = sensorB.getDistance();
```

```
  int dR = sensorR.getDistance();
  int dL = sensorL.getDistance();


  if (currentMillis - previousMillis >= len0) {
        previousMillis = currentMillis;
// CASE FRONT
        if (dF<15 && dF >= 10) {
        if(truetoneF) {
        truetoneF = false;
        }
        if(!truetoneF) {
        truetoneF = true;
        playSineTone1(truetoneF);
        }
        }

        else if(dF<10 && dF >= 0) {
        playSineTone1(truetoneF);
        }

// CASE BACK
  if (dB<15 && dB >= 10) {
        if(truetoneB) {
        truetoneB = false;
        }
        if(!truetoneB) {
        truetoneB = true;
        playSineTone2(truetoneB);
        }
        }

        else if(dB<10 && dB >= 0) {
        playSineTone2(truetoneB);
        }

// CASE RIGHT
  if (dR<10 && dR >= 5) {
        if(truetoneR) {
        truetoneR = false;
        }
        if(!truetoneR) {
        truetoneR = true;
        playSineTone3(truetoneR, 0);
```

```
        }
        }

        else if(dR<5 && dR >= 0) {
        playSineTone3(truetoneR, 0);
        }
        }

// CASE LEFT
  if (dL<10 && dL >= 5) {
        if(truetoneL) {
        truetoneL = false;
        }
        if(!truetoneL) {
        truetoneL = true;
        playSineTone3(truetoneL, 1);
        Serial.println("true");
        }
        }

        else if(dL<5 && dL >= 0) {
        playSineTone3(truetoneL, 1);
        Serial.println("TRUE");
        }
}
```