chairSense Research Paper

Vedant Singhania

Abstract

chairSense is a small, Arduino-powered device designed to assist wheelchair users in navigating tight and/or crowded spaces like narrow hallways or parties, especially when reversing or turning. The few similar devices that exist in markets today are cost-prohibitive to many wheelchair users. As a result, a major goal for this product was to use reliable components and sensors while keeping costs low as possible. Possible future improvements to this product may include a smaller form factor (e.g. a printed circuit board) and additional sensors for more comprehensive obstacle detection. This device is comprised of an Arduino Mega 2650 Rev3 Board, two HC-SR04 ultrasonic sensors, and a 0.96” OLED display that are all wired together via jumper wires on a breadboard, then mounted onto a small plastic board along with a battery pack to power it and tape to secure the battery pack.

Keywords: arduino, microcontroller, disability, wheelchair, ultrasonic sensor, obstacle detection, obstacle sensing, power wheelchair

Contents

chairSense Research Paper

Origin of Idea

When people are confined to wheelchairs, they often lose many aspects of independence, which is often associated with higher levels of depression (Noh et al., 2016). In addition, they are limited in their freedom because it can be hard to navigate crowded or small spaces easily. chairSense aims to slightly alleviate this loss of independence and freedom by helping wheelchair users navigate more freely via sensors that can help to avoid obstacles while backing up or turning. While this device was created for all wheelchair users, it would provide additional help to some who are confined to a wheelchair due to injuries where turning your head/upper body is difficult or impossible (i.e. spinal injury) (Mayo Clinic Staff, 2021).

Current State of Affairs

This isn't the first obstacle detection device created for wheelchairs. In 2016, a company named Braze Mobility came out with blind spot sensors that have a similar feature set but start at $1655, which can be cost-prohibitive to many people (Braze Mobility, n.d.). Additionally, a company named LUCI has created a more comprehensive collision avoidance system (which includes sensors for dropoffs and an anti-tip sensor system) but has a higher price of $8,445 (Coxworth, 2020; LUCI, n.d.). In addition, it may be hard to get insurance providers to approve items like this, so the cost would be more of a limiting factor.

What is the Product?

chairSense is a small device that is designed to be attached above the back wheels of a wheelchair and detects obstacles behind and to the side of the wheelchair using two ultrasonic sensors An 0.96" OLED screen at the front of the wheelchair outputs the distance from nearby objects at the side or back of the wheelchair in centimeters. If the chair gets too close to an object in those areas, a message to stop moving will show up on the OLED display to quickly warn the wheelchair user of a possible collision. By using this device, there will be less of a potential of property damage from backing into or turning into obstacles. In addition, there will be less potential for dangerous injuries from getting run over by wheelchairs, especially powered ones, which can weigh anywhere from around 50 to 400 pounds without a passenger (EZ Lite, n.d.).

Choice of Components

Why Use an Arduino?

Arduino was used as the prototyping platform because of its low cost and high extensibility. In addition to the board being cheap, third-party components are also very affordable, which allows for more trial and error since then, you are not working with expensive parts that could easily break. Because Arduino is open source and highly extensible, there is plenty of documentation for the Arduino and third-party components, which helped to eliminate the extra time of figuring out how each component works (Hunter, 2019). In addition, using Arduino for prototyping has a much smaller learning curve since you don't need to learn how to design an entire circuit. The Arduino Mega 2650 Rev3 Board was used rather than the popular Uno Board in order to not be limited by the memory and amount of pins that could be used so that there was more room for experimentation.

Table 1

Arduino Mega 2560 Rev3 Technical Specifications

Microcontroller	ATmega2560
Operation Voltage	5V7-12V
Input Voltage (Recommended)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16

DC Current Per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB are used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
LED_BUILTIN	13
Length	101.52 mm
Width	53.3 mm
Weight	37 g

Note. Data from "Mega 2560 Rev3 - Arduino Online Shop" by Arduino LLC, (https://store-usa.arduino.cc/products/arduino-mega-2560-rev3?selectedStore=us). Copyright 2022 by Arduino LLC.

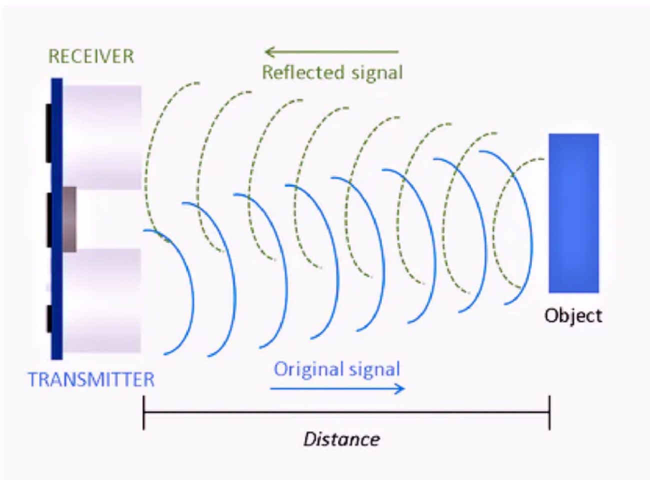
Why Use Ultrasonic Sensors?

Ultrasonic sensors were chosen to detect the obstacles because they are small and not affected by an unwanted source like sunlight or object color like an infrared sensor. In addition, the ultrasonic wave will reflect off of almost any object, which is useful in this case because the sensor needs to be able to reflect off of many objects so the wheelchair user gets an accurate measurement of their distance from an obstacle (Zhmud et al., 2018, 2).

Getting more technical, the ultrasonic sensor used in this project specifically has a transmitter on the left and a receiver on the right. Sound waves are emitted from the transmitter at 40 kHz, and then those sound waves are reflected off of the object and go back to the receiver. The sensor then gives information about the time it took for that sound wave to travel there and back and that data can then be manipulated in order to find the distance away from an object (Zhmud et al., 2018, 2). To convert the data from the sensor to distance, plug in the time from the ultrasonic sensor and the speed of sound (0.034 cm/µs) into the time = distance/speed equation and then solve for distance. After that, divide the measurement by two since the sound wave went to the object and back. This will yield the distance in cm away from the object (Jabbaar, 2019).

Figure **1**

How an ultrasonic sensor works



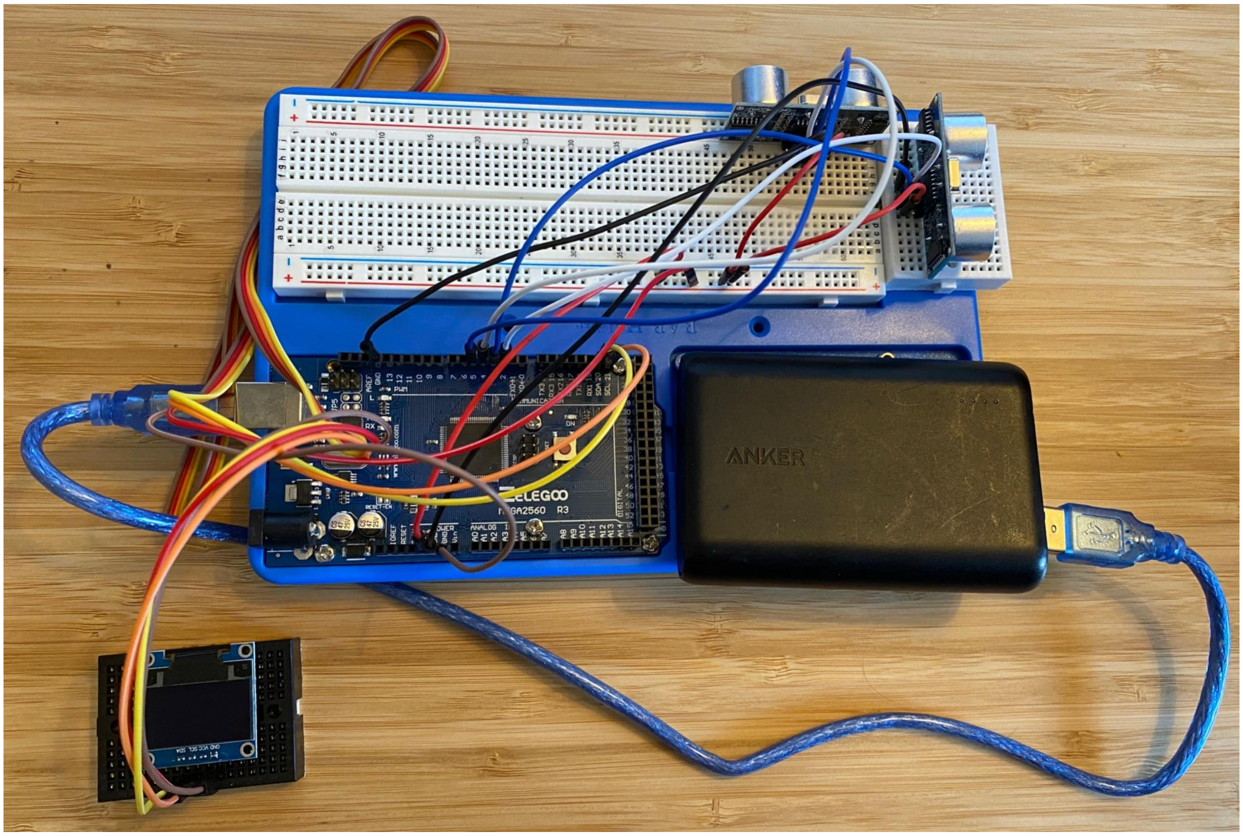
Note. From "Arduino Project Hub" by Arbi Abdul Jabbaar, 2013 (https://create.arduino.cc/projecthub/abdularbi17/ultrasonic-sensor-hc-sr04-with-arduino-tutorial-327ff6).

Product Design

All of the components are mounted on the plastic board, but the OLED display is connected to the Arduino board with a long wire (100 cm) so that even though the rest of the device will be at the back of the chair, the OLED display will be at the front of the chair so the wheelchair user can see it. Because the breadboard pinholes are connected vertically, a separate breadboard turned 90 degrees was used in order to put the second ultrasonic sensor vertically without the pins on the sensor being connected to each other.

Figure 2

Bird's eye view of the device (when not in use)



Part Sourcing and Cost

Table 2

Part Sourcing and Cost

Item	Unit Cost	Quantity	Total Cost
Arduino Mega 2560 Rev3	$17.49	1	$17.49
Full-Size Breadboard	$2.66	1	$2.66
Mini Breadboard	$0.99	2	$1.98
HC-SR04 Ultrasonic Sensor	$2.40	2	$4.80
UCTRONICS 0.96" OLED Display	$6.99	1	$6.99
Jumper Wires	$0.15	14	$2.10
USB Cable	$7.50	1	$7.50
ANKER 10,000 mAh Battery Pack	$25.99	1	$25.99
Plastic Mounting Board	$8.99	1	$8.99

Total	$78.50

Note. Prices are of the actual component purchased from A.

The Code

This device was programmed using the standard Arduino IDE, which uses an extended version of C++ with Arduino-specific functions and code and used a third-party library to easily control the pixels on the OLED screen (Arduino). The overall flow of the code is a setup function to set pins to input or output and to clear the display, then a loop that continuously loops as long as the board is connected to power. Within this loop houses the code that continuously finds the distance away from objects using data from ultrasonic sensors and displays the distance away from a nearby object at the back and side of the chair in cm. The code was made modular by using functions, which greatly helped in the coding process and made the code much more readable. Going more in-depth, first, external libraries are imported in lines 2-4 (figure 2), and then constants and variables are initialized to store the pins where components are plugged in and the distance on lines 13-22 (figure 3). Next, the pins are assigned a number, and whether the pin is input or output, and the display is cleared to ensure no display issues on lines 26-39 (figure 3).

Figure 3

Lines 1 through 40 of the source code of the device

```
1 // Import required libraries
2 #include <ArducamSSD1306.h>    // Modification of Adafruit_SSD1306 for ESP8266 compatibility
3 #include <Adafruit_GFX.h>   // Needs a little change in original Adafruit library (See README.txt file)
4 #include <Wire.h>           // For I2C comm, but needed for not getting compile error
5
6 /*
7 HardWare I2C pins
8 A4   SDA
9 A5   SCL
10 */
11
12 // Pin definitions
13 #define OLED_RESET  16  // Pin 15 -RESET digital signal
14 #define LOGO16_GLCD_HEIGHT 16
15 #define LOGO16_GLCD_WIDTH  16
16
17 const int echoB = 2;
18 const int trigB = 3;
19 const int echoS = 4;
20 const int trigS = 5;
21
22 float durationB, durationS, distanceB, distanceS;
23
24 ArducamSSD1306 display(OLED_RESET); // FOR I2C
25
26 void setup(void)
27 {
28   // Start Serial
29   Serial.begin(115200);
30   pinMode(echoB, INPUT);
31   pinMode(trigB, OUTPUT);
32   pinMode(echoS, INPUT);
33   pinMode(trigS, OUTPUT);
34   // SSD1306 Init
35   display.begin();  // Switch OLED
36   // Clear the buffer.
37   display.setTextSize(2);
38   display.setTextColor(WHITE);
39 }
40
```

Lines 41-65 (figure 4) are a function that calculates the distance from nearby objects from ultrasonic sensor data and prints out that data in a relative position on the OLED display. Finally, in lines 64 - 85 (figure 4), the display is cleared and some basic headings are printed onto the display, and then the previous findDistance function is run in order to populate the display with data continuously (as long the Arduino board has power).

Figure **4**

Lines 41 through 85 of the source code of the device

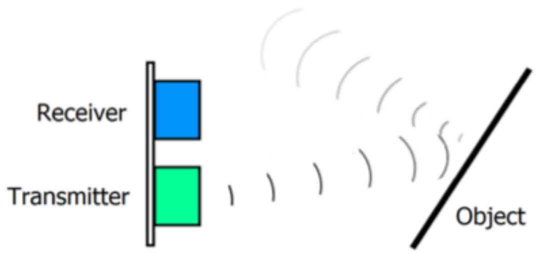
```
41 int findDistance(int duration, int distance, int trig, int echo, int start){
42  digitalWrite(trig, LOW);
43  delayMicroseconds(2);
44  digitalWrite(trig, HIGH);
45  delayMicroseconds(10);
46  digitalWrite(trig, LOW);
47  duration = pulseIn(echo, HIGH);
48  distance = (duration*.0343)/2 - 5;
49  if (distance < 0) {
50   distance = 0;
51  }
52  if (distance <= 30 && distance > 5) {
53   display.setCursor(start,30);
54   display.println(distance);
55  } else if (distance <= 5) {
56   display.setCursor(start,30);
57   display.println("STOP!");
58  }
59
60  if (distance > 5) {
61   display.setCursor(start+15, 30);
62   display.println(" cm");
63  }
64
65 }
66
67
68 void loop() {
69   display.clearDisplay();
70   display.setCursor(5,0);
71   display.println("BACK|RIGHT");
72   display.setCursor(5,15);
73   display.println("    |     ");
74   display.setCursor(5,30);
75   display.println("    |     ");
76   display.setCursor(5,45);
77   display.println("    |     ");
78   display.setCursor(5,60);
79   display.println("    |     ");
80   findDistance(durationB, distanceB, trigB, echoB, 0);
81   findDistance(durationS, distanceS, trigS, echoS, 64);
82   display.display();
83   //5 or 64
84   // +15 px to get from num to cm
85 }
```

Limitations of the Product

In addition to the useful features in this device, there is some current limitation that can hopefully be solved in future iterations of the product. Firstly, because there are only two sensors, there is a greater possibility for inaccurate distance readings. If the angle of the object is too high in comparison to the angle of the ultrasonic sensor, the sound waves from the transmitter will bounce off the object but not return to the receiver (figure 5) (Zhmud et al., 2018, 2).

Figure 5

Movement of the ultrasonic wave with a high angle of the object relative to the sensor



Note. From “Application of ultrasonic sensor for measuring distances in robotics” by V A Zhmud et al., 2018 (https://iopscience.iop.org/article/10.1088/1742-6596/1015/3/032189/pdf).

In addition, the position of the sensors on the wheelchair makes it impossible to detect objects that may be below the sensor. Also, since there are only sensors in the back, the rest of the chair remains unprotected from obstacles by the sensors. The sensor itself has some limitations too: while the sound wave can accurately reflect off of most surfaces, it has trouble with fluffy objects or objects that are very small (Zhmud et al., 2018, 2).

Because of the odd shape and large size of the device, it is currently complex to attach to a wheelchair and requires a lot of tape to attach it, which makes it not easy to put on the chair and not aesthetically pleasing. Finally, since the only way to know whether there is an obstacle is via a display, it is currently not very useful for people with vision impairments.

Future of the Product

In the future, the goal is to solve some of the problems mentioned in the previous section and to keep the product simple and make it more efficient. In order to avoid inaccurate distance readings from objects with high angles of incidence, a solution could be to use a sensor that sends out a larger sound wave in order for there to be a higher chance of the receiver receiving the sound wave (Adafruit Industries, n.d.). Another way to tackle this issue might be to use multiple sensors in close proximity so that there is a higher chance of the receiver receiving the sound wave.

In order for the chair to have more comprehensive obstacle detection, ultrasonic sensors could be added on the sides and front of the wheelchair. They could also be added closer to the bottom of the chair in order to help detect low items like a curb. In addition, the system could be made more encompassing by adding a gyroscope to detect chair tipping, adding an ultrasonic sensor to detect dangerous dips in the ground, and adding an ultrasonic sensor to detect obstacles above the chair. While the idea of this product is to make life in a wheelchair easily, it currently is not well designed for people with visual impairments. A solution for this could be to use audio cues, such as a buzzer that increases in pitch and volume as you get closer to an obstacle.

Another future idea that could be implemented is working with power wheelchair manufacturers such as Permobil to integrate the product onto the power wheelchair by wiring it directly into the wheelchair to eliminate the need for an external battery. Finally, since this device is currently a prototype, size and cost will significantly decrease when the product is converted to a PCB with the sensors and OLED display on it.

Conclusion

For wheelchair users, navigating complex areas, like tight or crowded places, can be difficult, and is much harder for people with injuries that restrict upper body movement. The wheelchair object sensing system solves this by using ultrasonic sensors to detect obstacles at the side of and back of the chair when turning or backing into spaces. With its low cost, it provides similar features to the few other obstacle detection devices for wheelchairs at a fraction of the cost. The Arduino board was cost-efficient, easy to program, and had lots of documentation, so it was a clear option for creating the prototype. The ultrasonic was the obstacle detection sensor used because it isn't restricted by the limitations that other sensors like infrared sensors have. The main limitations of this device are that there is a lack of sensors to detect obstacles on the rest of the chair and that the ultrasonic sensors still have some drawbacks, like not being able to detect objects will high angles of incidence. A future solution could be to add more sensors around the chair and use multiple sensors in close proximity. In terms of the future of this product, sensors could be added in other places like at the front of the wheelchair, and costs and size will go down when the product is made in the form of a PCB.

References

Adafruit Industries. (n.d.). *Large Ultrasonic (Sonar) Sensor with Horn and UART Output*. Adafruit Industries. Retrieved January 16, 2022, from https://www.adafruit.com/product/4664

Arduino. (n.d.). *Arduino Reference*. Arduino. Retrieved January 14, 2022, from https://www.arduino.cc/reference/en/

Arduino LLC. (n.d.). *Mega 2560 Rev3*. Arduino Online Shop. Retrieved January, 2022, from https://store-usa.arduino.cc/products/arduino-mega-2560-rev3?selectedStore=us

Braze Mobility. (n.d.). Braze Mobility | Blind Spot Sensors for Wheelchairs / Mobility Scooters. Retrieved January 13, 2022, from https://brazemobility.com

Coxworth, B. (2020, December 29). *LUCI adds collision avoidance and anti-tip tech to powered wheelchairs*. New Atlas. Retrieved January 14, 2022, from https://newatlas.com/good-thinking/luci-collison-avoidance-powered-wheelchairs/

EZ Lite. (n.d.). *EZ Lite Cruiser ® Standard Model*. EZ Lite Cruiser. Retrieved January 14, 2022, from https://www.ezlitecruiser.com/products/ez-lite-cruiser-standard-model

Hunter, K. (2019). *How to Prototype Electronics - Arduino vs PCB*. MicroType Engineering. Retrieved January 15, 2022, from https://www.microtype.io/how-to-prototype-electronics/

Jabbaar, A. A. (2019, September 17). *Ultrasonic Sensor HC-SR04 with Arduino Tutorial*. Arduino Project Hub. Retrieved January 15, 2022, from https://create.arduino.cc/projecthub/abdularbi17/ultrasonic-sensor-hc-sr04-with-arduino-tutorial-327ff6

LUCI. (n.d.). LUCI Mobility. Retrieved January 14, 2022, from https://luci.com

Mayo Clinic Staff. (2021, October 2). *Spinal cord injury - Symptoms and causes*. Mayo Clinic. Retrieved January 13, 2022, from https://www.mayoclinic.org/diseases-conditions/spinal-cord-injury/symptoms-causes/syc-20377890

Noh, J.-W., Kwon, Y. D., Park, J., Oh, I.-H., & Kim, J. (2016, November 30). *Relationship between Physical Disability and Depression by Gender: A Panel Regression Model*. NCBI. Retrieved January 13, 2022, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5130183/

Purdue Online Writing Lab. (2019). *General APA Format*. Purdue OWL. Retrieved January 16, 2022, from https://owl.purdue.edu/owl/research_and_citation/apa_style/apa_formatting_and_style_guide/general_format.html

Zhmud, V. A., Kondratiev, N. O., Kuznetsov, K. A., Trubin, V. G., & Dimitrov, L. V. (2018). Application of ultrasonic sensor for measuring distances in robotics. *Journal of Physics: Conference Series*, 2. 10.1088/1742-6595/1015/3/032189