ECE 445

Spring 2024

Design Doc:
Stick-On Proximity Sensor

Shrijan Sathish, Raunak Bathwal, Aryan Damani

Problem:

Many older cars lack proximity sensors that let the user know how close their car is to various obstacles, whether it be their garages, parking garage walls, or even curbs. Though this can be handled through driver experience, as well as various tricks of knowing where to look in the rearview or side mirrors to know where the front, sides, or back of the car is with respect to walls and other obstacles, it is always better to be sure. We aim to solve this inconvenience that comes with older model cars. Given this is an accessory most modern cars come with, and something users find useful, we believe that providing an easy, convenient, and relatively cheap solution to this issue will be beneficial.

Solution:

Our solution involves using 4 proximity sensors that can be placed on each corner of the car, with a receiver that can be placed inside the car. These will communicate through bluetooth and radio frequency, and the receiver itself will also contain 4 lights, one on each of its corners, each corresponding to a sensor or sensor cluster. This will correspond with each sensor placed, and light up as well as produce an auditory cue (small “beeps”) to alert the user how close they are to an obstacle and where it is. The closer you are to an obstacle, the faster the frequency of the beeps and flashes of light. Below is a visual aid to clarify our proposed solution.

Visual Aid:

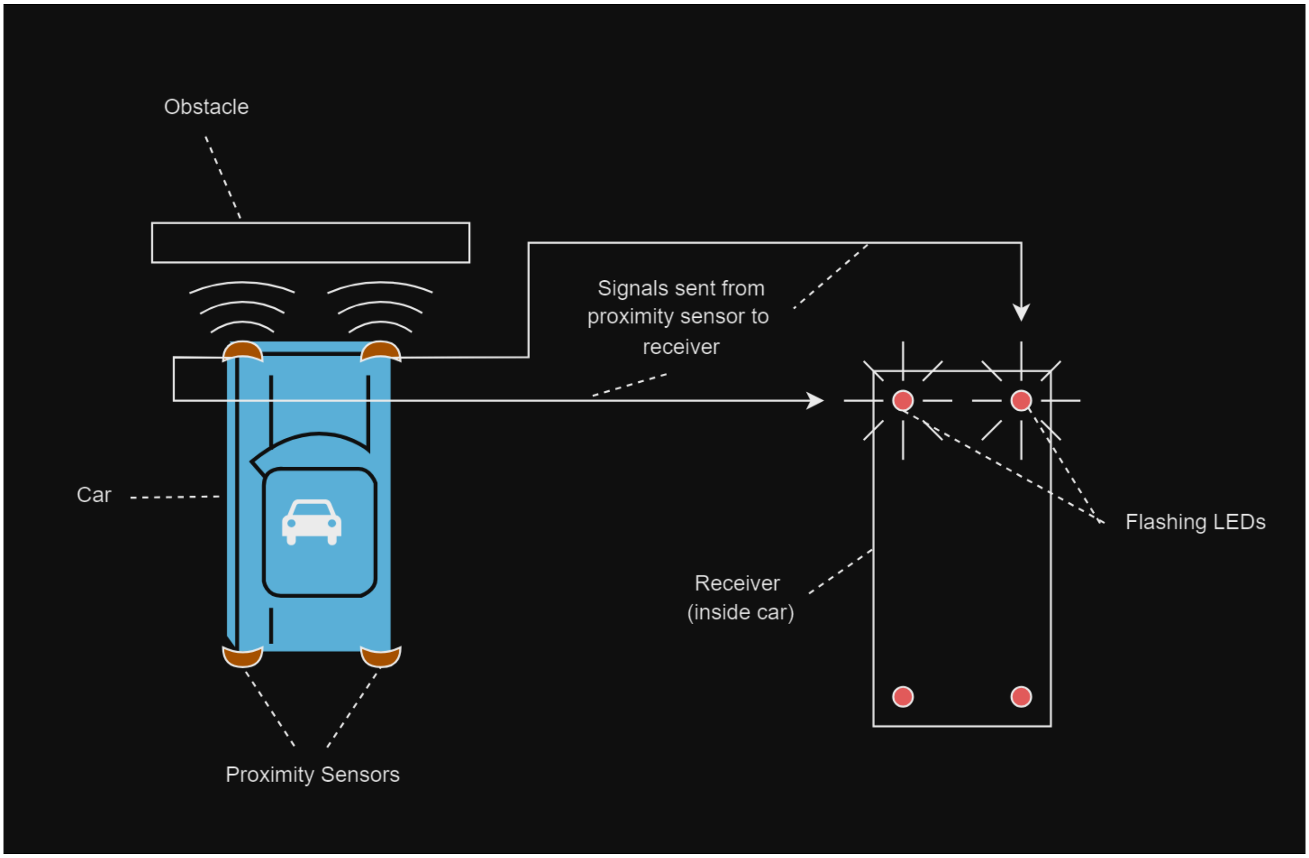


Figure 1: Visual Aid depicting how the sensors will interact with the receiver in the car

Block Diagram:

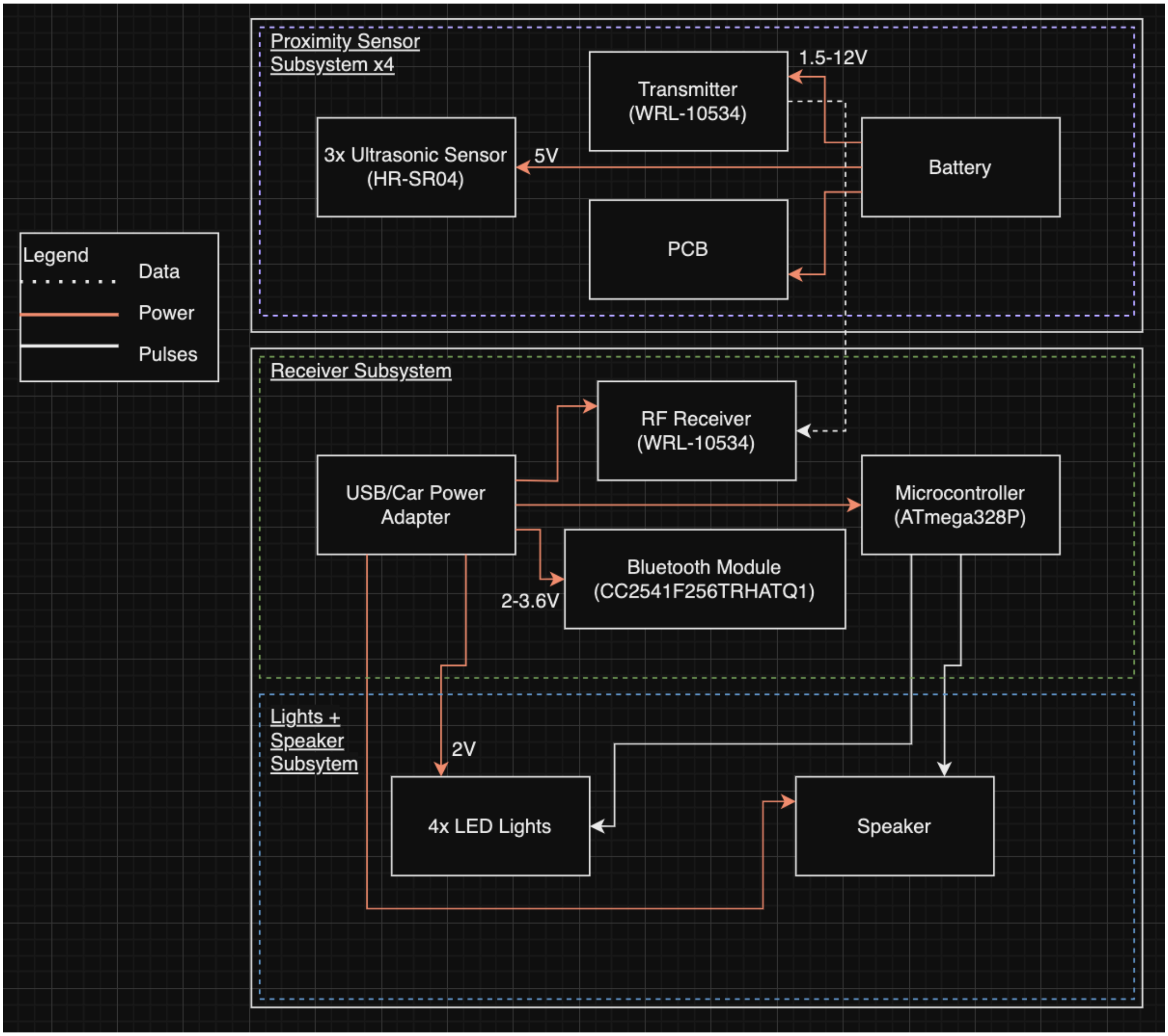


Figure 2: Block Diagram outlining the various subsystems and components of our design

High Level Requirements:

- The user must be able to stop the car/avoid obstacles without using any aids other than the proximity sensor (e.g. back facing camera, internal sensors that may be in more modern cars that we may use for testing).

- The receiver must accurately beep and flash the corresponding LED when an object is detected within range

- The four proximity sensors must be accurately mapped to their respective positions on the receiver system LEDs.

Subsystem Overview:

Subsystem 1: Proximity Sensor (x4)
The first, and main system, will be the sensors placed all around the car. Each module will be the same, regardless of where on the car it is placed. Each module will consist of 3 ultrasonic sensors(HC-SR04) based on their predicted placement on the vehicle, our custom PCB, a small lithium ion battery, and a wireless RF transceiver (WRL-10534). The module will constantly transmit distance data to the receiver module located within the vehicle to make sure the driver is aware of how close they may be to any potential obstacles. These will all be mountable to vehicles, and will have instructions on how to be placed for accurate obstacle detection. Each module will also have a small plastic housing to hold the pieces in place and to attach to the vehicle. The module will have one or two modules depending on placement on the vehicle, and for 2 sensors the data from the one transmitting an object as closer will be sent through the receiver.

Requirements:
- The proximity sensor must be able to detect objects that come within 0-3 feet of the car continuously.
- It must be able to consistently update the receiver module about distance from obstacles.

Verification:
- No drops in communication between sensor and receiver for more than 500ms at a time
- Delay between sensor and receiver is minimal (less than human perceivable)
- Sensor reading is continuous, and is able to detect within ~22 degrees on each side of the sensor, horizontally and vertically
- Sensor can read up to 3 ft in front of it

Subsystem 2: Receiver

The receiver subsystem will be located within the vehicle, consisting of an RF receiver (WRL-10534) to communicate with the above proximity sensors, a power adapter to get power from the USB/car power, and a microcontroller(ATtiny85) or raspberry pi to read input from proximity sensors, and output signals to control the lights and speakers and if the vehicle is too close to an object. This is where all calculations regarding the time data of the ultrasonic sensor will be processed, allowing for use of threshold “distances" in order to change the frequency of the beeps and flashes of the speaker and lights. This is also where the identification of which sensor is tripped is done, so the corresponding light is triggered.

Requirements:
- The receiver inside the car must be able to receive signals from the proximity sensor consistently and update the microcontroller with the new data
- It must be able to receive power from the car without overheating as it will be consistently plugged in.
- Must properly output to LEDs and speaker output with updating frequency according to distance

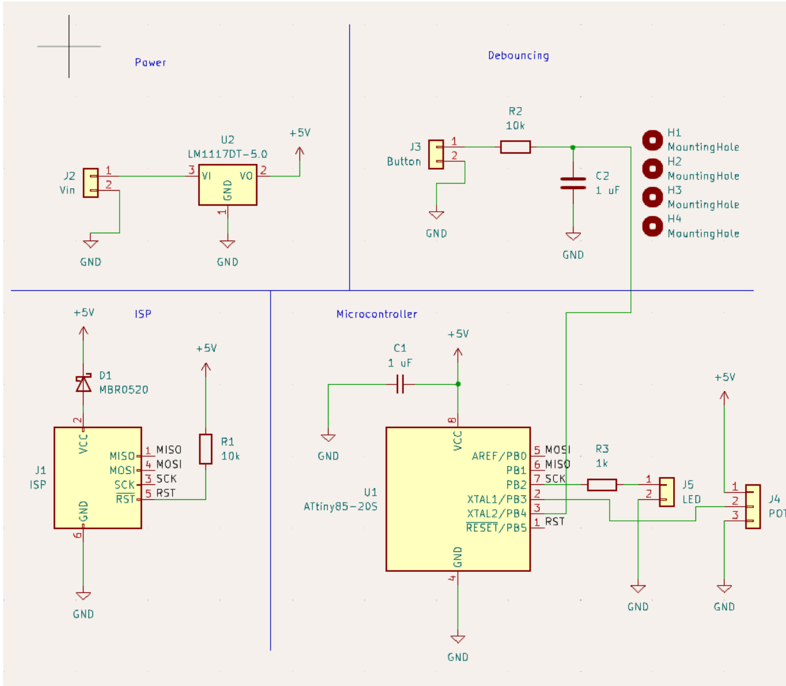
Verification:
- No drops in communication between sensor and receiver for more than 500ms at a time
- Delay between sensor and receiver is minimal (less than human perceivable)
- Delay between sensor reading and LED/speaker output must be less than 500ms
- Speaker Output frequency must vary with distance input consistently

Subsystem 3: Lights + Speaker

The light and speaker system will consist of a small speaker that will change frequency based on how close an object is, combined with a set of red LED diodes to represent which sensor is being triggered so the driver knows which direction to avoid.

Requirements:
- The LED on each corner (four total) must light up with its corresponding proximity sensor as a visual cue to the user
- There must be audible and increased frequency beeps as the car gets increasingly closer to an obstacle, giving an audio cue for distance

Verification:
- Delay between sensor reading and LED/speaker output must be less than 500ms



To clarify what we plan to do, the above image is the kicad schematic for the mini-encabulator lab. We found that this is very similar to the function we want from the lights and speaker subsystem, so we plan to use a modified version of this setup in our project. Instead of taking an input from the potentiometer to control the frequency of the lights/beeping, we will take the current updated distance from an obstacle as the input. Approaching an obstacle will be the same as twisting the potentiometer to achieve a higher frequency and vice versa when the obstacle is exiting the vicinity around a car where it could be potentially dangerous.

Tolerance Analysis:

One part of our design that is critical to its success is the proximity sensors' ability to detect objects in front of the car. Since each ultrasonic sensor has a detection cone of 30 degrees, we believe that having 3 per unit, each offset by 30 degrees will be a good way to sense 90 degrees, from the side of a given car to the front bumper of the car.

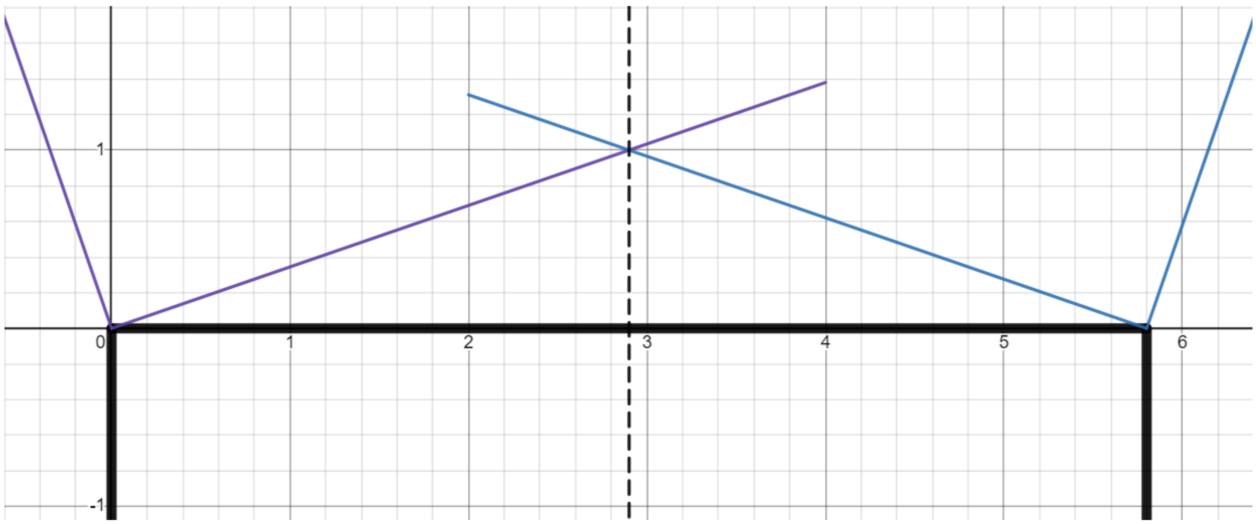


Figure 3: Graph plot outlining detection angles for 3 ultrasonic sensors placed together, with a minimum detection distance of 1 foot from the midpoint of the car

Using a graphing application, and assuming an average car width of 5.8 feet, we can see that sensors placed at an angle such that they start detecting at a distance of 1 foot in front of the car allows us some leeway at the edges of the car as well, which could help with object detection. We would simply have to warn the user that once the "beeps" coming from the receiver reach a frequency that makes it sound like a constant tone, that an object is closer than one foot from the car.

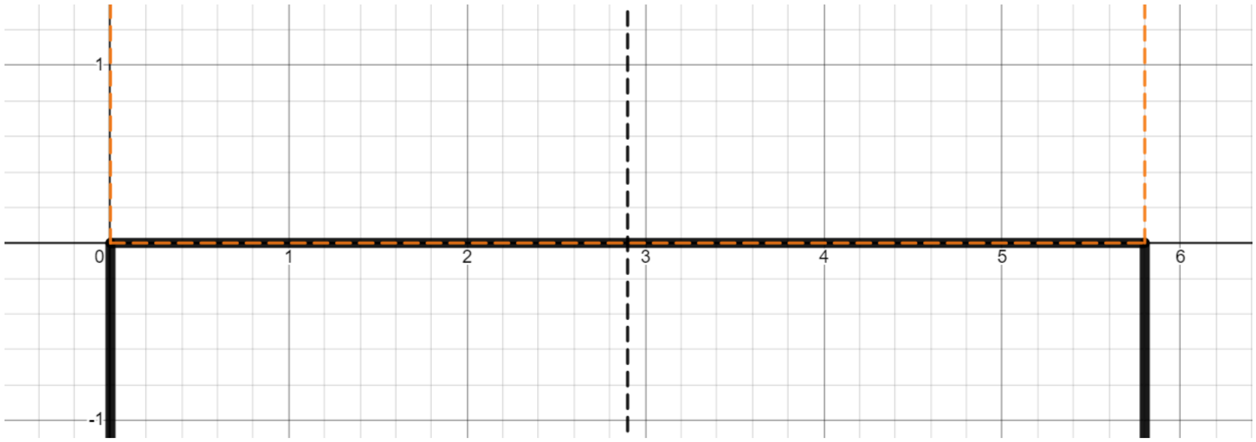


Figure 4: Detection angle (orange dashes) if ultrasonic sensors are aligned with the bumper of the car

If we sacrifice that little extra distance away from the edge of the car to detect objects that are strictly in front of it, the 90 degree detection area will allow us to measure objects right up to the bumper, but also only strictly in front of the car (as seen in the graph above).
This provides us with enough confirmation that regardless of the detection angle we choose, we can use three ultrasonic sensors per detection module in a way such that we will be able to detect obstacles in front of or behind the vehicle at any given time.

According to the HC-SR04 datasheet, the formula to calculate distance based on the time recorded by the ultrasonic sensor is as follows: Centimeters = ((Microseconds / 2) / 29). Using the graph seen in figure 4, we can calculate an example of what a threshold distance might look like. If we take one of the sensor modules placed on the left side of the car (relative to figure 4), and one of the ultrasonic sensors closest to the bumper, we can calculate the max distance ahead of the car it can detect with a 30 degree cone.
Using $tan(30) = 0.57735$, we can calculate the slope of a 30 degree angle and plot it on the graph:

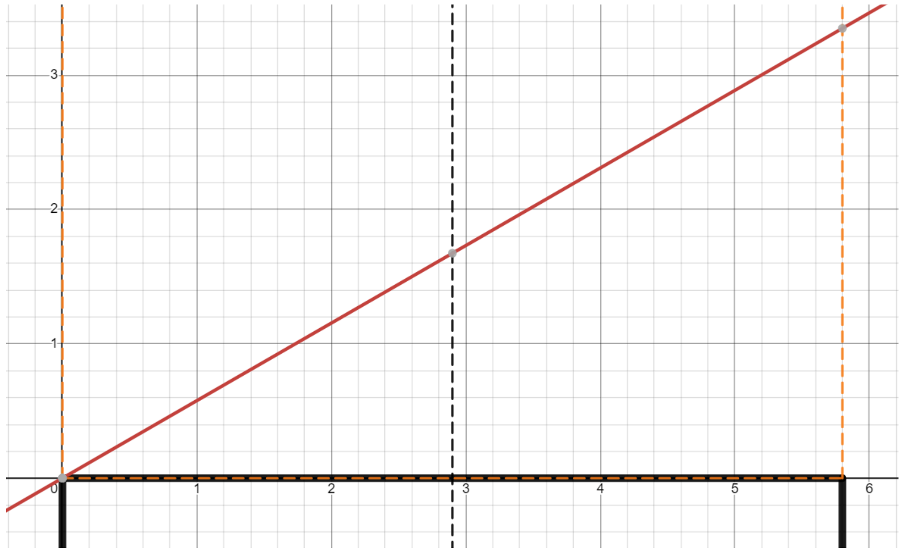


Figure 5: Max angle of detection of ultrasonic sensor aligned with bumper of car

As seen in figure 5, we have 2 intersection points at the midsection line of the car and the opposite edge of the car. By hovering over these points, we get a distance from the car of 1.674 feet at the midpoint, and 3.349 feet at the opposite end. Let's use the midpoint distance of 1.674 feet as an example, and convert this to 51.02352 cm. Using the formula from the datasheet, we can calculate that 51.02352 = ((Microseconds / 2) / 29), meaning that the threshold for microseconds would be 2959.36416. This time data can be used to trigger the microcontroller to send signals to our LEDs and speaker to flash and beep at a certain frequency, which ensures that our design will work as intended.

Cost and Schedule:

Labor:

The average salary for a computer engineering graduate from UIUC is $109,176, which equates to roughly $52 an hour. We are planning to dedicate 10 hrs/wk per person for the following 9 weeks, which will give ($52/hr) * 2.5 * (10hrs/wk * 9wk * 3 people) = $35100 total for labor.

Schedule:

Week 7	Design Review with TAs, Start PCB Design, Review part list
Week 8	Review part order and submit, Continue PCB design
Week 9	Spring Break
Week 10	Finalize PCB Design, Start Receiver build with RasPi for test
Week 11	Create Housing, Start module creation with PCB, Start Microcontroller programming for module
Week 12	Write receiver logic for output to lights and speakers, Complete receiver
Week 13	Test thoroughly and debug any functionality
Week 14	Add extra functionality if necessary
Week 15	Practice for demo

Parts:

Part	Price
HC-SR04 (Ultrasonic Sensor) x 4	9.99 x 4
WRL-10534 (Transceiver) x 2	5.50 x 2
Raspberry Pi	35.00 x (1)
Red LED	6.99 for 100 pack (if not taken from ECEB lab room)
Speaker	2.10
Battery/Batteries	5.50 x 4,
ATTINY85-20PU (microcontroller)	1.66
Total:	118.71

The parts list above is a maximum estimate, as we believe that some of them can be found in the ECEB lab room. Thus we believe we will be well below the $150 allocated budget for this project.

Ethics and Safety:

In accordance with the IEEE Code of Ethics (7.8, I - III), we guarantee that our group will uphold the highest standard of integrity and safety when designing and testing our proximity sensor.

Firstly, as a team we will ensure to divide work equally and adhere to IEEE's Code of Ethics concerning treating all persons involved with fairness and respect. We will enlist the help of TAs and professors accordingly and promise a harassment-free workplace when engaging in the design and engineering process (IEEE II, 7-9).

As our design obviously involves real vehicles, safety is quite a large concern and will require rigorous testing to make sure that our design will not fail in any condition. The biggest safety concern arises from our sensor's ability to detect and transmit accurate distance data to the microcontroller in the receiver, which we promise to be fully transparent about (ACM 1.3). If the design does not meet standards, we will make sure to only run simulation tests where no person or property can be damaged (ACM 1.2). An example of such a test would be to keep the sensors stationary and at a height and distance that would mimic a car, and move objects slowly towards them. Until we pass trials of these simulation tests 100% of the time, we will not move on to using an actual vehicle to validate our design. This is to ensure that we uphold ethical practices in guaranteeing public safety and lawful practices.

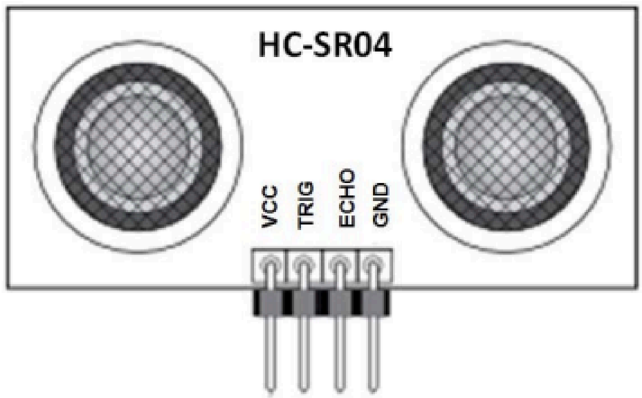
Another concern that arises from our project will be the use of batteries/drawing power from the usb port of various cars. Since usb ports/power from cars are already regulated in terms of overcurrent, overvoltage, and overheating, we believe that this will not be an issue in our design. However, when using a battery in our proximity sensor module, we will need to ensure that we account for all of the previous mentioned things. This will involve ensuring things like the operating temperatures to range from -30 C to 80 C, and an optimal operating range from 5 C to 45 C [3].

Ultimately, we plan to uphold the highest standards of design, safety, and integrity throughout this project, as seen in the IEEE Code of Ethics, the ACM Code of Ethics, and laws/regulations pertaining to the practical use of obstacle detection sensors on cars. We aim to produce a product that delivers an extra edge of safety and convenience to owners of older model cars while adhering to rigorous ethical and safety standards.

Diagrams:

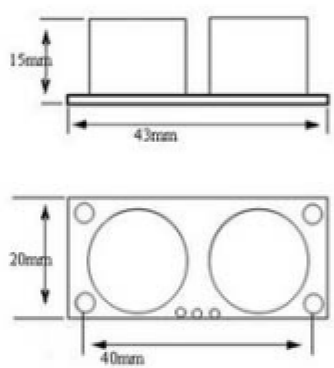
Ultrasonic Sensor(HC-SR04):

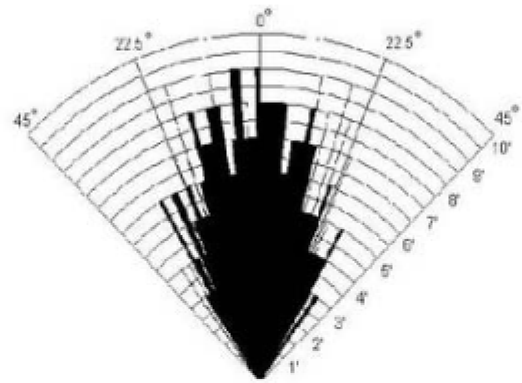
4. Hardware Information



VCC	3.3v ~ 5V
TRIG	Triggering Input Pin. 10uS TTL Pulses
ECHO	TTL Logic Output Pin. Proportional to distance
GND	Ground Pin

4.1 Mechanical Dimension





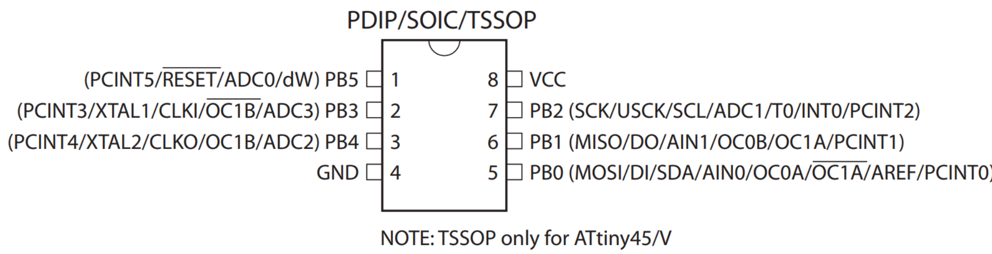
Practical test of performance,

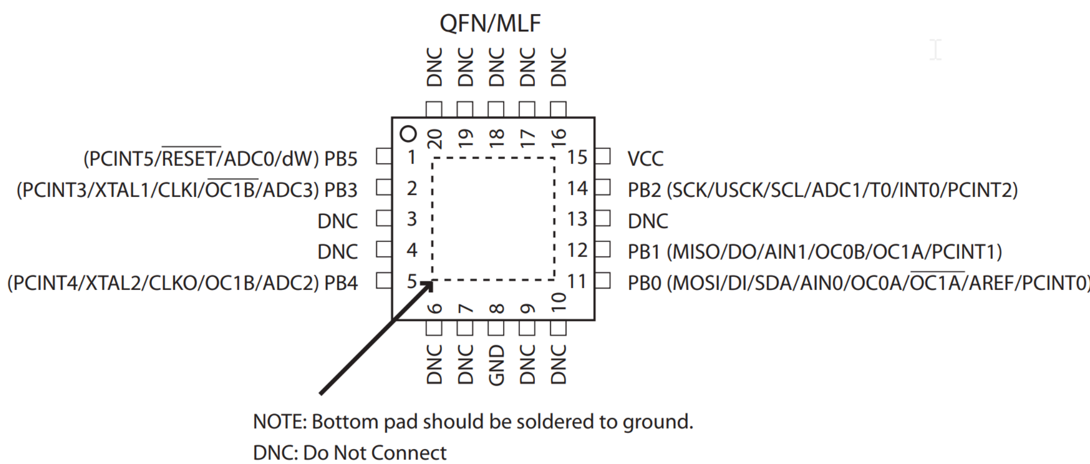
Best in 30 degree angle

Microcontroller(ATTiny85):

1. Pin Configurations

Figure 1-1. Pinout ATtiny25/45/85





References:

1. "IEEE Code of Ethics." *IEEE*, www.ieee.org/about/corporate/governance/p7-8.html. Accessed 8 Feb. 2024.
2. *Code of Ethics*, www.acm.org/code-of-ethics. Accessed 9 Feb. 2024.
3. "Why You Should Not Charge a Lithium Battery below 32 Degrees." *REDARC*, 19 Sept. 2022, www.redarcelectronics.com/us/resources/chargers-isolators-faqs/do-not-charge-lithium-battery-below-32-degrees/#:~:text=Lithium%20battery%20manufacturers%20often%20state,model%2C%20consult%20your%20manufacturer).