SMART HOME

(PARKING SYSTEM)

SUBMITTED BY

JASLIN JENIFER JJ

DEPARTMENT OF ECE

961819106024

INTRODUCTION

This is an Arduino-based collision detection warning system for parking. This kind

of system is the fastest growing safety feature in automotive industries. Such a

system enables vehicles to identify the chances of collision and give visual and

audio warning to driver when parking, so that the driver can take necessary action

to avoid collision during parking. This project idea is based on an Arduino

controller and the whole project will give you very good understanding of how

this system works. The step-by-step method is explained so that you can make

this system. The hardware connection, pin information and Arduino program is

explained clearly.

COMPONENTS

• Computer:

This is required to write program and flash program to the controller. Also, you need to install Arduino IDE which is available free on

the Arduino website download section.

• Controller:

I used Arduino micro-controller. This you can get from an online

seller like Amazon, etc.

• Sensor:

I used HR SC-04 ultrasonic sensor.

• Piezo Buzzer:

I used Piezo buzzer to make the audio warning.

• LED:

There are two colours of LED I used - red and green.

• Wires:

Jumper wires are required to make hardware connections. You need

to use all types of jumper wires like male-male, female-female and female-

male.

CODE:

////parking system////

//// Year 2022////

//// Blesslin benoj D////

const int trigPin = 7;

const int echoPin = 4;

int buzz = 10;

long duration;

int distance;

// the setup function runs once when you press reset or power the board

void setup() {

// initialize digital pin 13 as an output.

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

pinMode(13, OUTPUT);

pinMode(2, OUTPUT);

Serial.begin(9600);

}

// the loop function runs over and over again forever

void loop()

{

// Clears the trigPin

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration = pulseIn(echoPin, HIGH);

// Calculating the distance

distance= duration\*0.034/2;

// Prints the distance on the Serial Monitor

if(distance <= 50 && distance >= 20)

{

digitalWrite(13, HIGH);

// digitalWrite(13, LOW);

// turn the LED on (HIGH is the voltage level)

}

else

{

digitalWrite(13, LOW); // turn the LED off by making the voltage LOW

// wait for a second

}

if(distance <= 20)

{digitalWrite(2, HIGH);

tone(buzz, 2000);

delay(100);

noTone(buzz);

delay(100);

tone(buzz, 2000);

delay(100);

noTone(buzz);

delay(100);

tone(buzz, 2000);

delay(100);

noTone(buzz);

tone(buzz, 2000);

delay(100);

noTone(buzz);

delay(100);

}

else

{

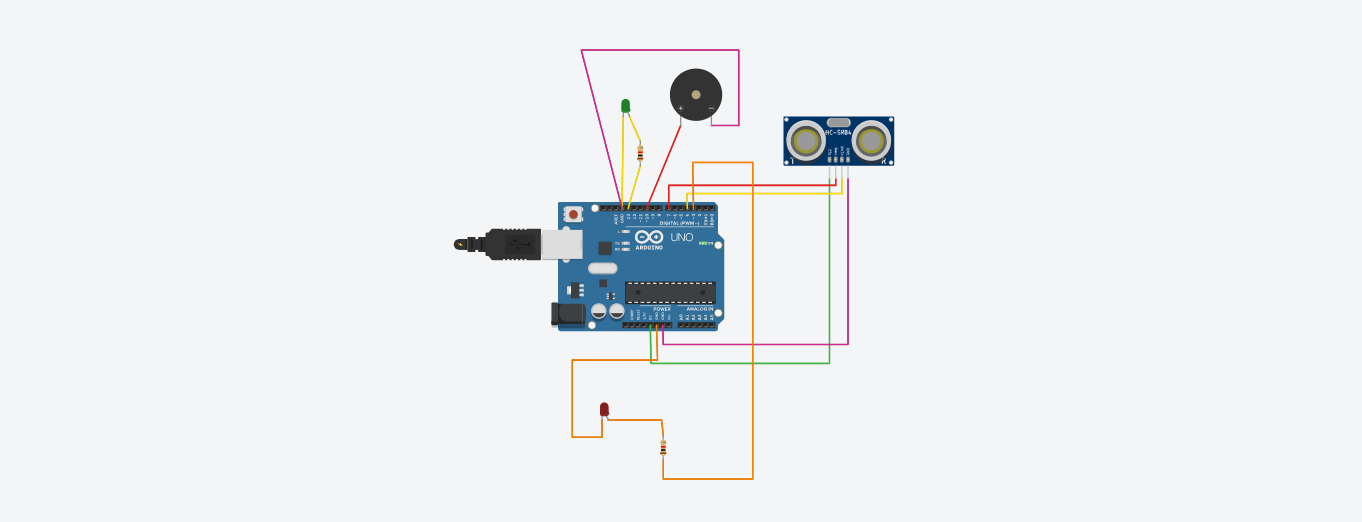
digitalWrite(2, LOW); // turn the LED off by making the voltage LOW

// wait for a second

}

}

FIGURE:



• Zone 1: No warning

• Zone 2: Only visual warning (in this zone, driver has to pay attention)

• Zone 3: Both visual and audio warning (driver has to take necessary action

to avoid collision)

TINKERCAD LINK

https://www.tinkercad.com/things/kYr6chZAvgQ