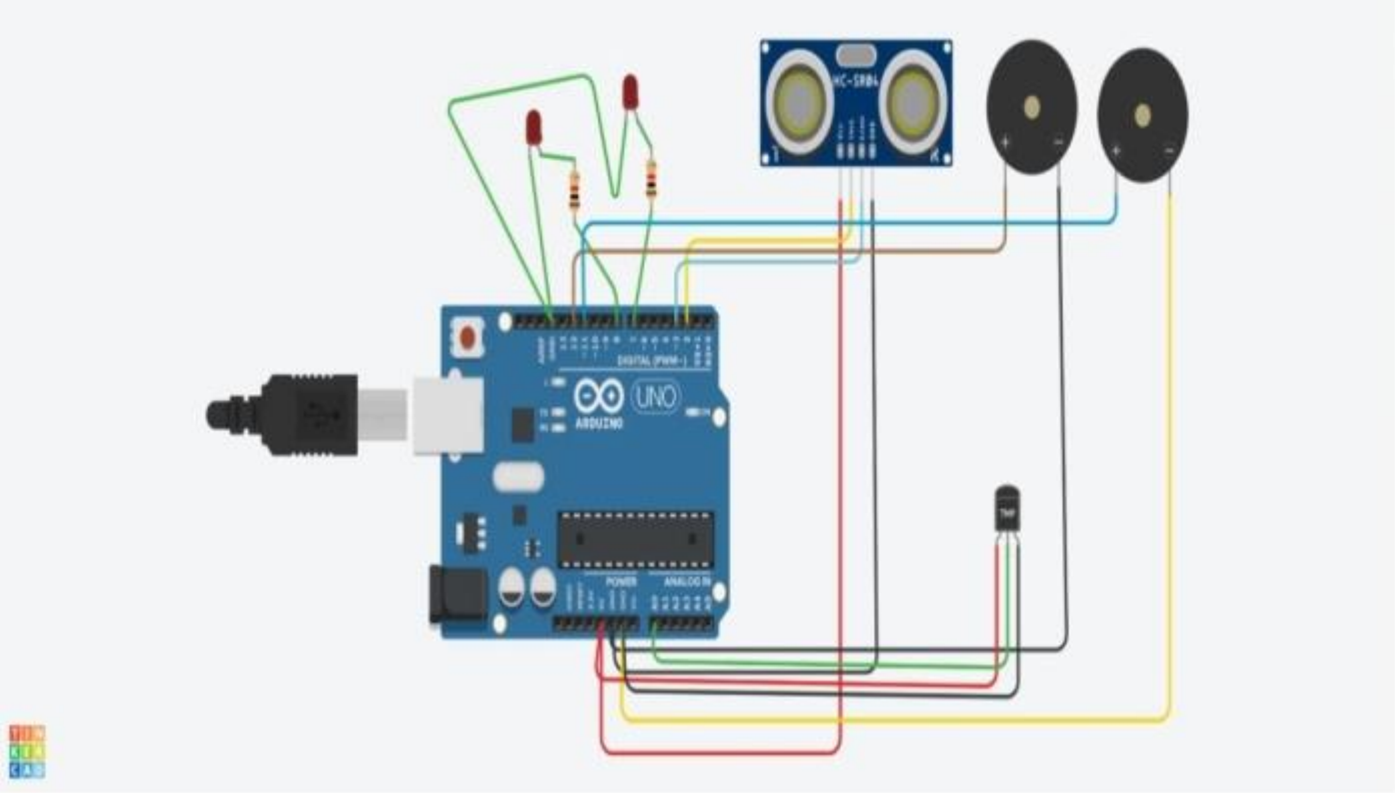
Team ID	PNT2022TMID44448
Submitted by	K.Nithya
Topic	Signs With smart connectivity for better road safety
Assignment 1	Smart home automation using tinker card

CIRCUIT DESIGN :



Code:

```
int
t=2;
        int e=3;
        void setup()
        {
        Serial.begin(9600);
        pinMode(t,OUTPUT);
        pinMode(e,INPUT);
        pinMode(12,OUTPUT);
}
void loop()
{
        //ultrasonic sensor
        digitalWrite(t,LOW);
        digitalWrite(t,HIGH);
        delayMicroseconds(10);
        digitalWrite(t,LOW);
        float dur=pulseIn(e,HIGH); float dis=(dur*0.0343)/2;
        Serial.print("Distance is: ");
        Serial.println(dis);
        //LED ON
        if(dis>=100)
{
        digitalWrite(8,HIGH);
        digitalWrite(7,HIGH);
}
 //Buzzer For ultrasonic Sensor
        if(dis>=100)
        {
 for(int i=0; i<=30000; i=i+10)
{
tone(12,i);
delay(1000);
noTone(12);
delay(1000);
}
}
//Temperate Sensor
double a= analogRead(A0);
double t=(((a/1024)*5)-0.5)*100;
Serial.print("Temp Value: ");
Serial.println(t);
delay(1000);
//LED ON
if(t>=100)
{ digitalWrite(8,HIGH);
digitalWrite(7,HIGH);
}
//Buzzer for Temperature Sensor
if(t>=100)
{
for(int i=0; i<=30000; i=i+10)
{
tone(12,i);
delay(1000);
noTone(12);
delay(1000);
}
}
//LED OFF
if(t<100)
{
digitalWrite(8,LOW);
digitalWrite(7,LOW);
}
}
```

Team ID	PNT2022TMID44448
Submitted by	K.Nithya
Topic	Signs With smart connectivity for better road safety
Assignment 2	Built a python code , assume you get temperature and humidity values (generated with random function to a variable) and write a condition to continuously detect a alarm in case of high temperature

CODING :

```
#define ADC_RESOLUTION 1024.0
#define PIN_LM35       A0
const int TRIG_PIN   = 6; // Arduino pin connected to Ultrasonic Sensor's
TRIG pin
const int ECHO_PIN   = 7; // Arduino pin connected to Ultrasonic Sensor's
ECHO pin
const int BUZZER_PIN = 3; // Arduino pin connected to Piezo Buzzer's pin
const int DISTANCE_THRESHOLD = 50; // centimeters
float duration_us, distance_cm;
void setup() {
  Serial.begin(9600);
  // switch to Internal 1.1V Reference
  pinMode(TRIG_PIN, OUTPUT);   // set arduino pin to output mode
  pinMode(ECHO_PIN, INPUT);    // set arduino pin to input mode
  pinMode(BUZZER_PIN, OUTPUT); // set arduino pin to output mode
  analogReference(INTERNAL);
  pinMode(LED_BUILTIN, OUTPUT);
}
void loop() {
  // get the ADC value from the temperature sensor
  int adcVal = analogRead(PIN_LM35);
  // convert the ADC value to voltage in millivolt
  float milliVolt = adcVal * (ADC_VREF_mV / ADC_RESOLUTION); //
ADC_VREF_mV = 1100 mV
  // convert the voltage to the temperature in Celsius
  float tempC = milliVolt / 10;
  // convert the Celsius to Fahrenheit
  float tempF = tempC * 9 / 5 + 32;
  // print the temperature in the Serial Monitor:
  Serial.print("Temperature: ");
  Serial.print(tempC);   // print the temperature in Celsius
  Serial.print("°C");
  Serial.print("  ~  "); // separator between Celsius and Fahrenheit
  Serial.print(tempF);   // print the temperature in Fahrenheit
  Serial.println("°F");
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the
voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the
voltage LOW
  delay(1000);
  delay(1000);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  // measure duration of pulse from ECHO pin
  duration_us = pulseIn(ECHO_PIN, HIGH);
  // calculate the distance
  distance_cm = 0.017 * duration_us;
  if(distance_cm < DISTANCE_THRESHOLD)
    digitalWrite(BUZZER_PIN, HIGH); // turn on Piezo Buzzer
  else
    digitalWrite(BUZZER_PIN, LOW);  // turn off Piezo Buzzer
  // print the value to Serial Monitor
  Serial.print("distance: ");
  Serial.print(distance_cm);
  Serial.println(" cm");
  delay(500);


}
```

Footer

Footer navigation

T

Team ID	PNT2022TMID44448
Submitted by	K,Nithya
Topic	Signs With smart connectivity for better road safety
Assignment 3	Write python code for blinking LED and traffic lights for Raspberry pi.

CODING:

```
Import RP1.GPIO as GP10
import time
GPIO. setmode(GPIO.BOARD)
GPIO.setup(7, GPIO.OUT) #Green LED
GPIO.setup(11, GPI0.OUT)#Yellow LED
GPIO.setup(13, GPI0.OUT) #Red LED
GPI0.setup(15, GP10.IN, pull_up_down=GPIO.PUD_UP)#Button
def turn_on(pin, seconds):
    GPIO.output (pin,GPIO.HIGH)
    time.sleep(seconds)
def turn_off (pin, seconds):
    GPIO.output (pin, GPIO.LOW)
    time.sleep(seconds)
try:
    while True:
        button_state=GPIO.input (15)
        if button_state== True:
           turn_on(13,2)
           tum_off(13,.1)
           turn_on(7,4)
           turn_off(7,.11)
           turn_on(11,1)
           turn_off(11,1)
         else:
            if button_state== False:
               GPI0.output (7,GPIO.LOW)
               GPIO.output(11,GPIO.LOW)
               GP10.output (13,GPIO.LOW)
               time.sleep(.1)
```

```
except KeyboardInterrupt:
    GPIO.cleanup()
    print("Traffic Light Sequence Done")
```