CYCLE BUDDY

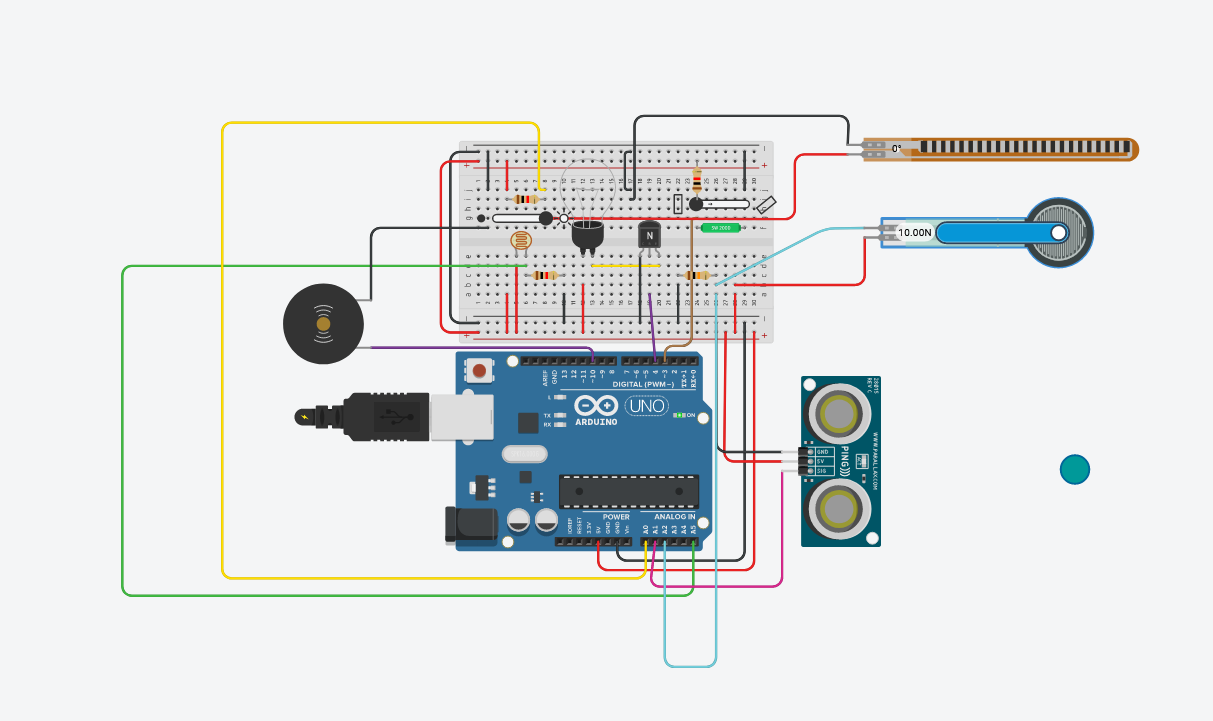
**By ~ Ashish Anand**

**ABSTRACT OF THE PROJECT:**

The main aim of the project is to prevent cyclist from accidents, so Cycle Buddy is a system that takes care of safety of the cycle rider in dangerous situations. It helps the rider to enjoy his ride stress-free without worrying about accidents. It turns on the headlights at night automatically for the rider for better visibility and in events of any accident it sounds the buzzer as and when necessary. It also keeps a note of nearby vehicles on the route and when the proximity of nearby vehicles is below the threshold value it sounds the buzzer automatically. We made the prototype by using hardware components like LDR Sensor, Force Sensor,

Ultrasonic sensor, Piezo buzzer, etc.

**CIRCUIT DIAGRAM:**

****

**Components Used**

**1) LDR SENSOR – For turning on the headlights as and when required based on the ambient surrounding light.**

**2)** **ULTRASONIC SENSOR – For maintaining appropriate distance from other vehicles on the road. This sounds the buzzer automatically when the distance of other objects is less than the threshold distance (40cm here).**

**3)** TILT SENSOR – This helps others around the rider on the road by sounding the buzzer whenever the rider has lost his/her balance for more than 3 seconds (threshold value).

**4)** FORCE SENSOR – This sensor detects if a major accident has happened and then sounds the buzzer in a high pitched and easily detectable tone which can’t be turned off without physically removing the circuit, this ensures that the rider (if unconscious) is found after the accident.

**5)** FLEX SENSOR – This sensor’s job is to prevent accidents from happening by sounding the buzzer in bursts of 2 seconds each time when the break has been pressed for longer than 2 seconds (threshold value).

**ARDUINO CODE:**

int ldr=A5;

int ldr\_value;

int light\_pin=4;

//int signal\_pin = 3;

int buzzer\_pin = 10;

int force\_pin = A2;

int distance;

int force\_value=0;

int accident =0;

int tilt\_pin = 3;

int tilt\_value=0;

int flex\_pin = A0;

int flex\_value = 0;

long readUltrasonicDistance(int triggerPin, int echoPin)

{

pinMode(triggerPin, OUTPUT); // Clear the trigger

digitalWrite(triggerPin, LOW);

delayMicroseconds(2);

// Sets the trigger pin to HIGH state for 10 microseconds

digitalWrite(triggerPin, HIGH);

delayMicroseconds(10);

digitalWrite(triggerPin, LOW);

pinMode(echoPin, INPUT);

// Reads the echo pin, and returns the sound wave travel time in microseconds

return pulseIn(echoPin, HIGH);

}

void setup()

{

Serial.begin(9600);

pinMode(light\_pin, OUTPUT); //LDR

pinMode(ldr, INPUT); //LDR

pinMode(buzzer\_pin, OUTPUT); //common

pinMode(force\_pin, INPUT); //Force

pinMode(tilt\_pin, INPUT); //Tilt

pinMode(flex\_pin, INPUT); // Flex

}

void loop()

{

digitalWrite(buzzer\_pin, LOW);

//Ultrasonic sensor

distance = 0.01723 \* readUltrasonicDistance(A1, A1);

if (distance < 40) {

digitalWrite(10, HIGH);

//Serial.println("Alert :");

//Serial.println(distance);

}

else {

//Serial.println("Safe :");

//Serial.println(distance);

digitalWrite(10, LOW);

}

//LDR sensor

ldr\_value=analogRead(ldr);

if (ldr\_value>512)

digitalWrite(light\_pin, LOW);

else

digitalWrite(light\_pin, HIGH);

//Force sensor

force\_value = analogRead(force\_pin);

//Serial.println(force\_value);

if(force\_value > 640){

accident = force\_value;

}

if(accident > 0){

//digitalWrite(buzzer\_pin, HIGH);

tone(buzzer\_pin,accident);

}

//Tilt sensor

tilt\_value = digitalRead(tilt\_pin);

//Serial.println(tilt\_value);

delay(3000);

tilt\_value = digitalRead(tilt\_pin);

noTone(buzzer\_pin);

if(tilt\_value == 0){

tone(buzzer\_pin,2000);

//digitalWrite(buzzer\_pin, HIGH);

}

//Flex sensor

flex\_value = analogRead(flex\_pin);

//noTone(buzzer\_pin);

if(flex\_value > 1000){

delay(2000);

flex\_value = analogRead(flex\_pin);

if(flex\_value > 1000){

digitalWrite(buzzer\_pin, HIGH);

delay(2000);

}

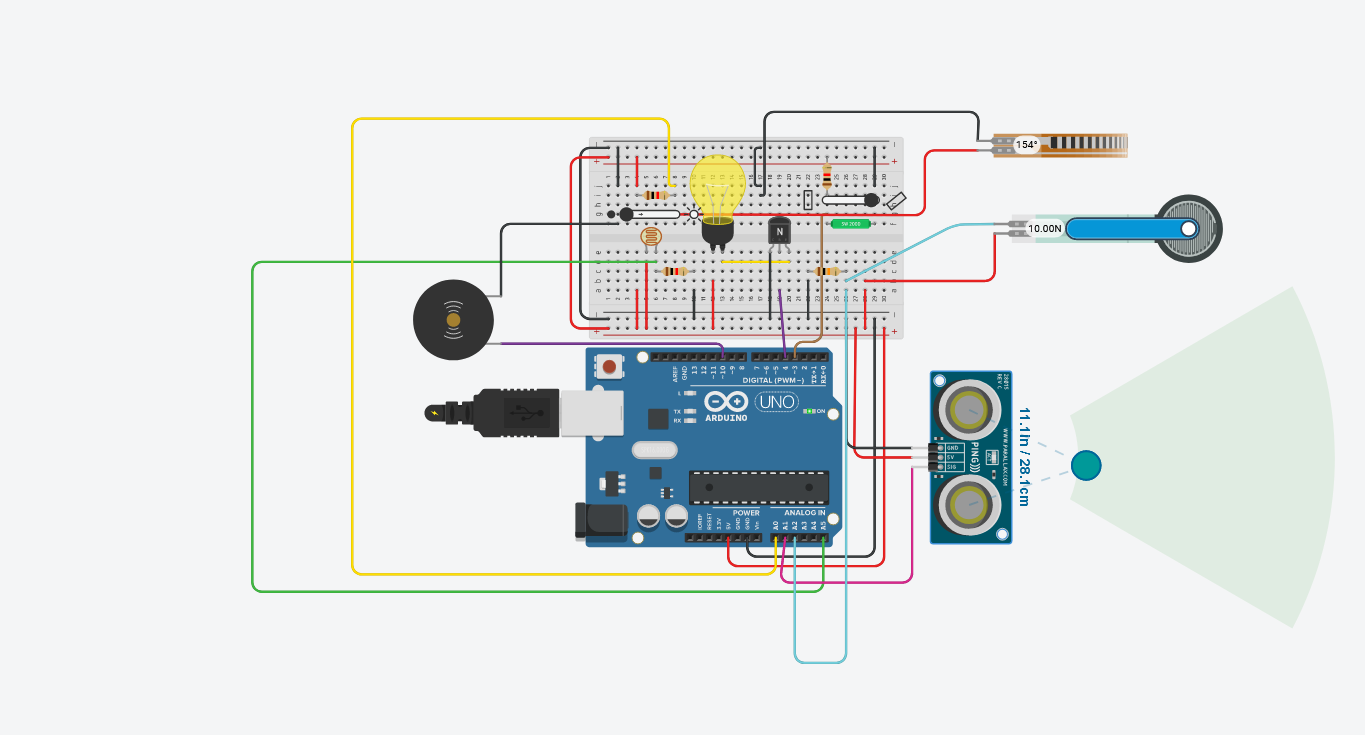
}

//Serial.println(flex\_value);

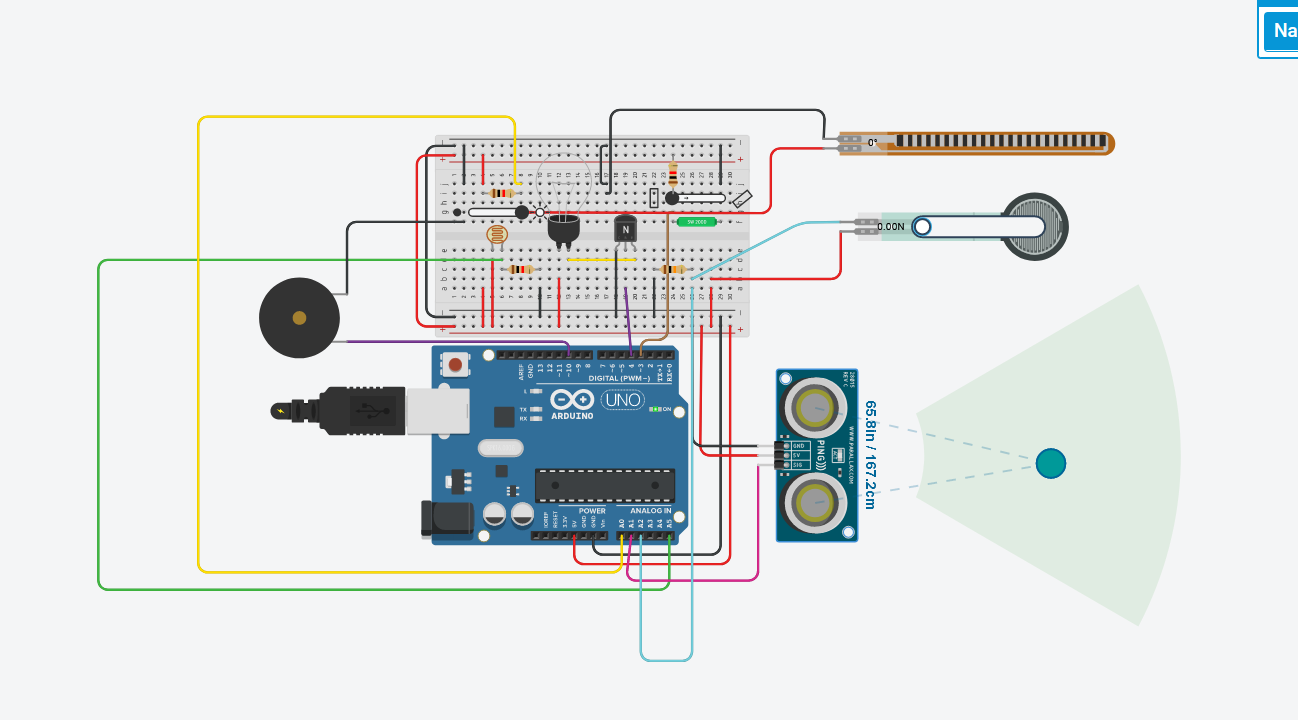
//delay(500);

}

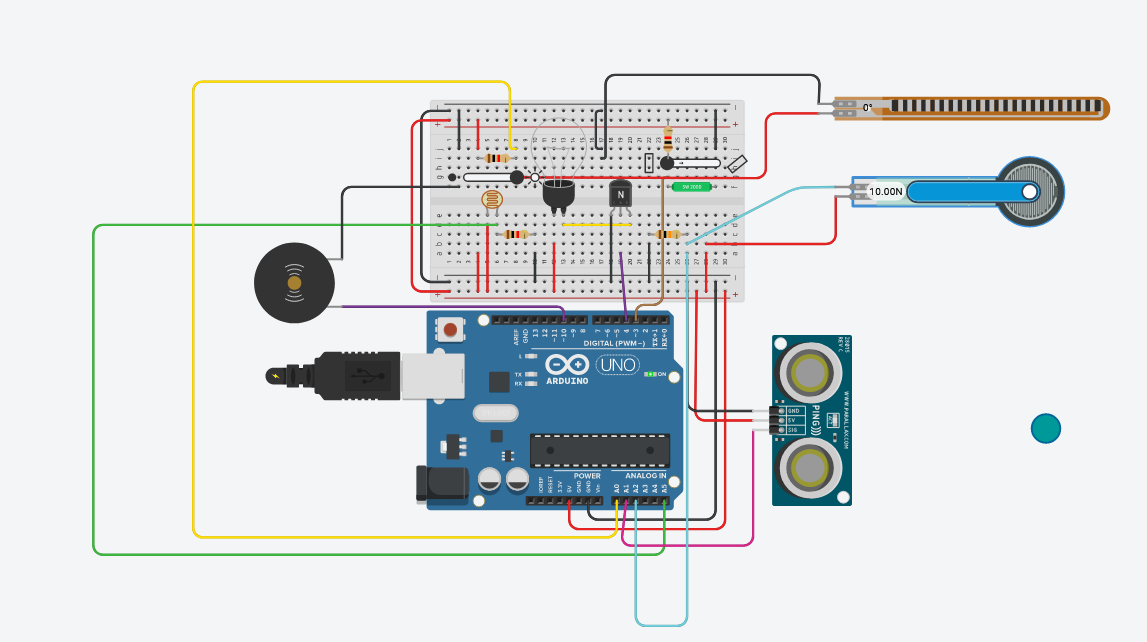
**SCREEN SHOTS OF THE OUTPUT:**



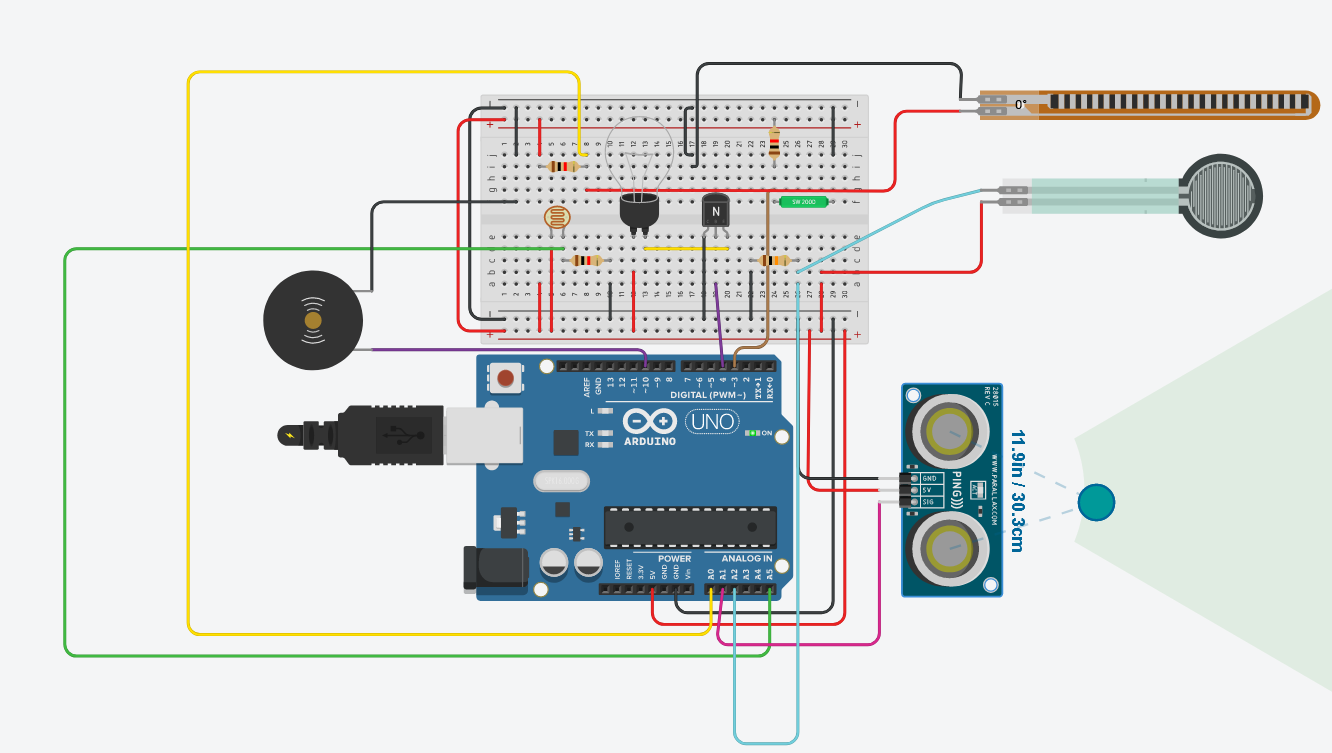
This photo shows the state of the circuit when all the sensors have been activated at the same time.



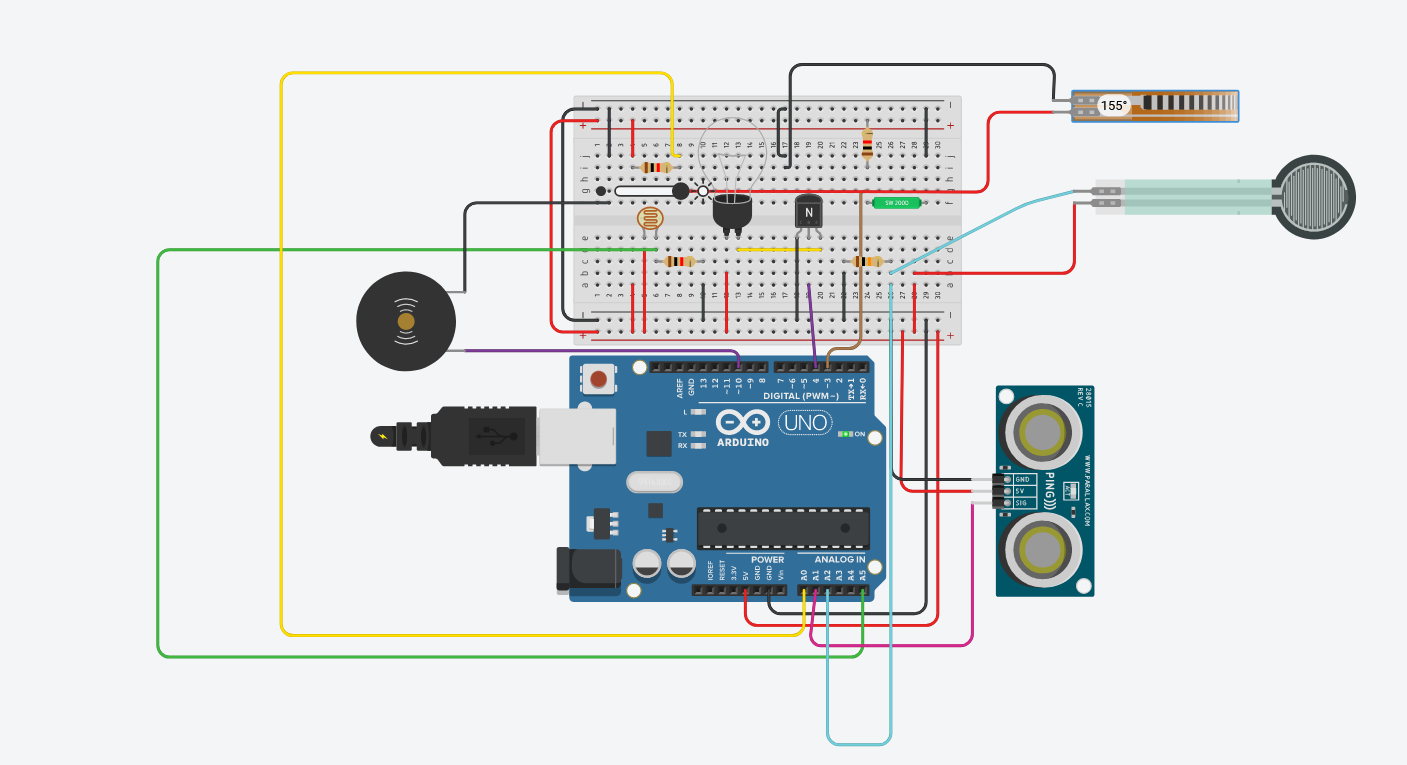
This image shows the state of the circuit when there is enough ambient sunlight is the surroundings. Notice here the bulb is in the “OFF” position.



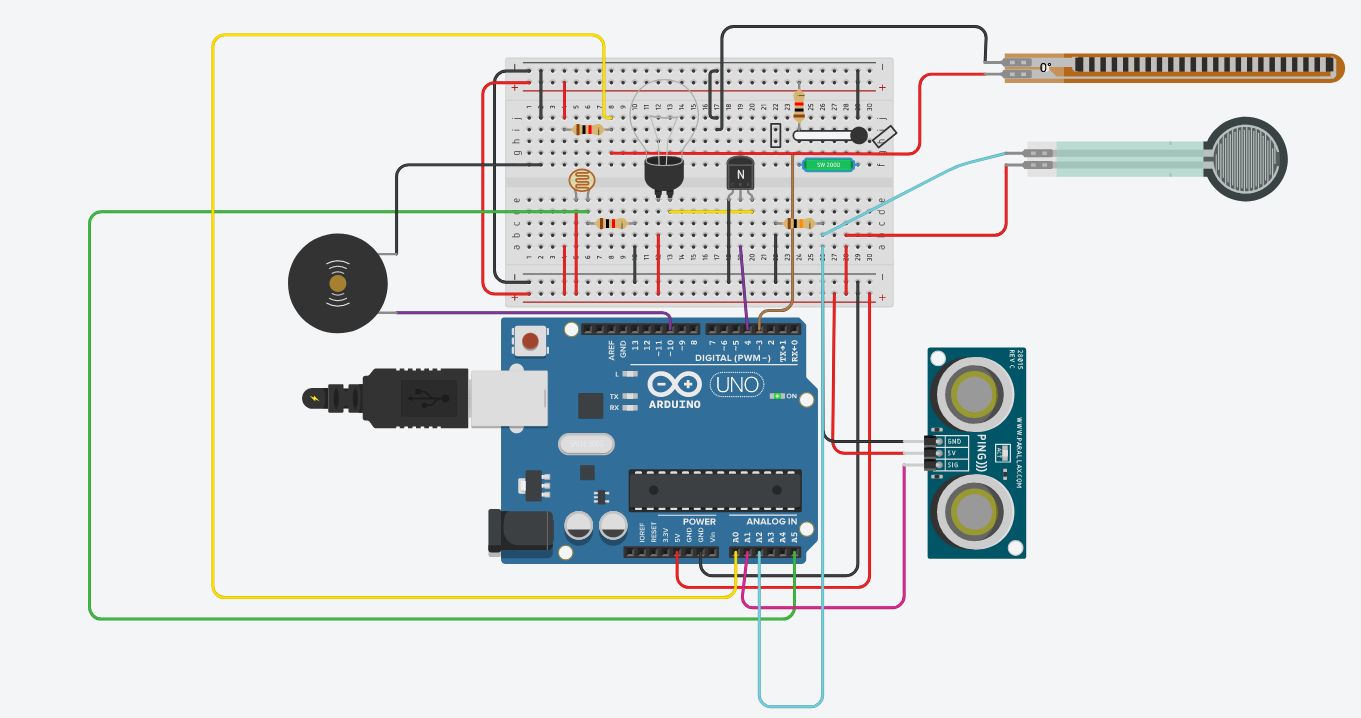
This image shows the condition of accident, here the force sensor has been activated and so the buzzer is ringing continuously.



This image shows the condition of a vehicle being too close to the rider, here the ultrasonic sensor has been activated and so the buzzer is ringing. The buzzer will stop ringing as soon as the rider gets away from the vehicle (distance > 40cm).



This image shows the condition of breaks being pressed for too long by the rider, here the flex sensor has been activated and so the buzzer is ringing in bursts of 2 seconds.



This image shows the condition of the rider losing balance for too long and hence the tilt sensor has been activated and the buzzer will ring till the rider gains back his/her balance.

**REFERENCES**

* [**https://www.youtube.com/watch?v=lv6Uhk1mRBA**](https://www.youtube.com/watch?v=lv6Uhk1mRBA)
* [**https://www.learnelectronicsindia.com/post/smart-street-light-project-using-arduino-uno-ldr-and-led**](https://www.learnelectronicsindia.com/post/smart-street-light-project-using-arduino-uno-ldr-and-led)
* [**https://www.youtube.com/watch?v=RjH\_BmHJkWY**](https://www.youtube.com/watch?v=RjH_BmHJkWY)
* [**https://www.youtube.com/watch?v=tbrhUIa1ZsM**](https://www.youtube.com/watch?v=tbrhUIa1ZsM)
* [**https://www.youtube.com/watch?v=12ueXJyrQVU&t=16s**](https://www.youtube.com/watch?v=12ueXJyrQVU&t=16s)
* [**https://www.arduino.cc/en/Tutorial/BuiltInExamples/Ping**](https://www.arduino.cc/en/Tutorial/BuiltInExamples/Ping)