Arduino RC Car using Tinkercad

Aim: To create a remote-controlled car on Arduino.

1. Turn on and off headlights (use Leds) with the press of a dedicated button.
2. For turning indicators, dedicate 2 buttons for left and right turning and at the press of the turning button, the respective light must blink continuously with an interval of 0.5 seconds and stop when the button is pressed again.
3. Add a horn (buzzer component) that will buzz 2 times when the button is pressed.
4. Building on the aforementioned features, use an ultrasonic sensor (HC-SR04) for: Object detection such that when an object is detected the car is programmed to stop and then steer to turn away and continue to move.

Services used: Tinkercad

The Arduino Uno(R3) is an open-source microcontroller board. Arduino controller is used to control the car. Tinkercad is a platform used to create a design of the circuits and simulate hardware model virtually.

Basically, here we are going to virtually create a robot car which performs the functions as per the instructions or commands given.
Construction

Components used :

1) Arduino uno R3 (1)

2)H-bridge motor driver (1)

3)Piezo (1)

4)Light bulb (4) and LED light (1)

5)DC motor (2)

6)Resistors (1k ohm — 6)

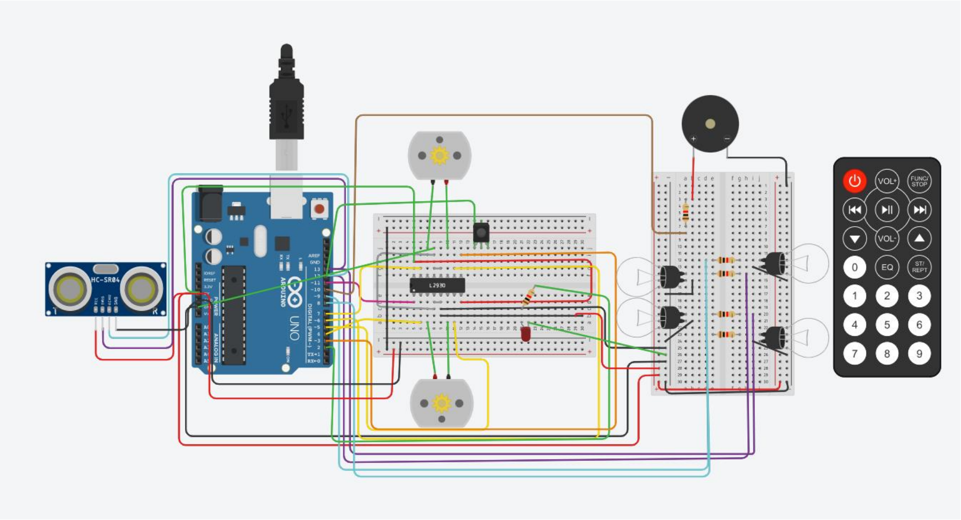
Working

Connect the components as shown in the circuit diagram. Here we have connected 2 DC motors. And based on the commands given in the serial monitor it is controlled. The circuit resembles a car having two headlights and two backlights, and a horn, DC motor for direction and speed control, one led light to indicate the stop position.

- When you send Letter 'F' to serial monitor it indicates car to move forward.
- When you send Letter 'B' to serial monitor it indicates car to move backward
- When you send Letter 'L' to serial monitor it indicates car to move left.
- When you send Letter 'R' to serial monitor it indicates car to move right.
- When you send Letter 'S' to serial monitor it indicates car to stop.
- When you send Letter 'A' to serial monitor it indicates car to accelerate.

- When you send Letter ‘D’ to serial monitor it indicates car to decelerate.
- When you send Letter ‘H’ to serial monitor it indicates blinking of headlights of the car.
- When you send Letter ‘J’ to serial monitor it indicates car to blinking of backlights of the car.
- When you send Letter ‘E’ to serial monitor it indicates horn of the car
- When you send ‘dist’ to serial monitor it indicates the distance of the object or obstacles.

SIMULATION OF THE CIRCUIT



Code explanation

Any Arduino program contains 2 main functions. void setup() and void loop()

```
const int trigPin=13;
const int echoPin=12;
char c;//status
void setup()
{
  Serial.begin(9600);
  pinMode(10,OUTPUT); //buzzer output pin
//for headlights
  pinMode(9, OUTPUT);
  pinMode(8, OUTPUT);

  //for backlights
  pinMode(13, OUTPUT);
  pinMode(12, OUTPUT);

  //for motor(wheels)
  pinMode(7,OUTPUT);//input 4
  pinMode(6,OUTPUT);//input 3
  pinMode(5,OUTPUT);//input 2
  pinMode(4,OUTPUT);//input 1
  pinMode(2,OUTPUT);
  pinMode(3,OUTPUT); //enable 3&4
  pinMode(11,OUTPUT);// enable 1&2
  delay(500);
  analogWrite(11, 255);
  analogWrite(3, 255);
 int distance;
  int duration;
}
void loop()
{
  if(Serial.available())
  { c=Serial.read();
   Serial.println(c);
  }
    if(c=='H')
    {
Serial.println("headlights Blinking");
                              digitalWrite(9,HIGH);
                       digitalWrite(8,HIGH);
```

```
                    delay(1000);

                    digitalWrite(9,LOW);

                    digitalWrite(8,LOW);

  delay(10);


}

if(c=='J')

{

        Serial.println("backlights blinking");

                            digitalWrite(13,HIGH);

                    digitalWrite(12,HIGH);

  delay(10);

   digitalWrite(13,LOW);

                    digitalWrite(12,LOW);

    delay(10);

}


  if(c=='F')

  {

   Serial.println("Forward");
```

```
            digitalWrite(7,LOW);

            digitalWrite(6,HIGH);

            digitalWrite(5,LOW);

            digitalWrite(4,HIGH);

                         digitalWrite(2,LOW);

    delay(10);

  }

 if(c=='L')

 {

   Serial.println("Left");

            digitalWrite(7,HIGH);

            digitalWrite(6,LOW);

            digitalWrite(5,LOW);

            digitalWrite(4,HIGH);

                         digitalWrite(2,LOW);

   delay(10);

 }

 if(c=='S')

 {

Serial.println("Stop");
```

```
                    digitalWrite(7,HIGH);

                    digitalWrite(6,HIGH);

                    digitalWrite(5,HIGH);

                    digitalWrite(4,HIGH);

                    digitalWrite(2,HIGH);

    delay(10);

}

if(c=='R')

{

    Serial.println("Right");

                    digitalWrite(7,LOW);

                    digitalWrite(6,HIGH);

                    digitalWrite(5,HIGH);

                    digitalWrite(4,LOW);

                    digitalWrite(2,LOW);

    delay(10);

}

if(c=='B')

{

   Serial.println("Back");
```

```
             digitalWrite(7,HIGH);

             digitalWrite(6,LOW);

             digitalWrite(5,HIGH);

             digitalWrite(4,LOW);

            digitalWrite(2,LOW);

   delay(10);

  }

  if(c=='E')

  { Serial.println("Buzzer");

  tone(10, 220, 100);

  delay(200);

  }

  if(c=='A'){

   Serial.println("accelerate");

 // Accelerate from zero to maximum speed

for (int i = 0; i < 256; i++) {

analogWrite(11, i);

analogWrite(3, i);

delay(10);

}
```

```
       pinMode(7,LOW);
  pinMode(6,LOW);
  pinMode(5,LOW);
  pinMode(4,LOW);
    }
    if(c=='D'){
      Serial.println("deaccelerate");
// Decelerate from maximum speed to zero
for (int i = 255; i >= 0; --i) {
analogWrite(11, i);
analogWrite(3, i);
delay(100);
}
if(c=='dist'){
        Serial.println("find distance");
        digitalWrite(trigPin,LOW);
  delayMicroseconds(2);

  digitalWrite(trigPin,HIGH);
  delayMicroseconds(10);
```

```
    }

    }

}
```

Using pinMode we can define the GPIO pins to be OUTPUT pins.

```
if ( Serial.available())
  {
    char c = Serial.read();
```

Serial. available() returns the number of characters (i.e.; bytes of data)

which have arrived in the serial buffer and that are ready to be read.

The value which is read from Serial is stored in a variable named c.

In void loop() i have defined the conditions which the car has to perform the actions based on the instructions given.

Tinkercad link :

https://www.tinkercad.com/things/4UpJDOhgQxw-brave-leelo-krunk/editel?sharecode=nC0WLBaIDsVe3DRAmym5_1ra95aWKC-HTTY3v4iY7t8