

# **Smartphone Controlled Object Detector**

Submitted by

Asfia Kawnine [C151273]

Arafa Ferdous [C151275]



Department of Computer Science & Engineering

International Islamic University Chittagong

# **Smartphone Controlled Object Detector**

Submitted by-

Asfia Kawnine [C151273]

Department of Computer Science & Engineering

International Islamic University Chittagong

Arafa Ferdous [C151275]

Department of Computer Science & Engineering

International Islamic University Chittagong

Submitted to-

Ahmed Imteaj

Lecturer

Department of Computer Science & Engineering

International Islamic University Chittagong

---

# Contents

## Topics:

- Objectives
- System Model
  - Connection between Bluetooth module and Arduino Uno
  - Connection between Ultra Sonic Sensor and Arduino Uno
  - Connection among Buzzer, Ultra Sonic Sensor and Arduino Uno
  - Connection among Gear Motors, Motor Shield and Battery
- System Description
  - Arduino Uno
  - Motor Shield
  - Ultra Sonic Sensor
  - Bluetooth module
  - Buzzer
  - Breadboard
  - Chassis board, Gear motor, Battery Holder, Jumper wire, Wheel
  - Working Process
- Program Code
- System Implementation
- Discussion

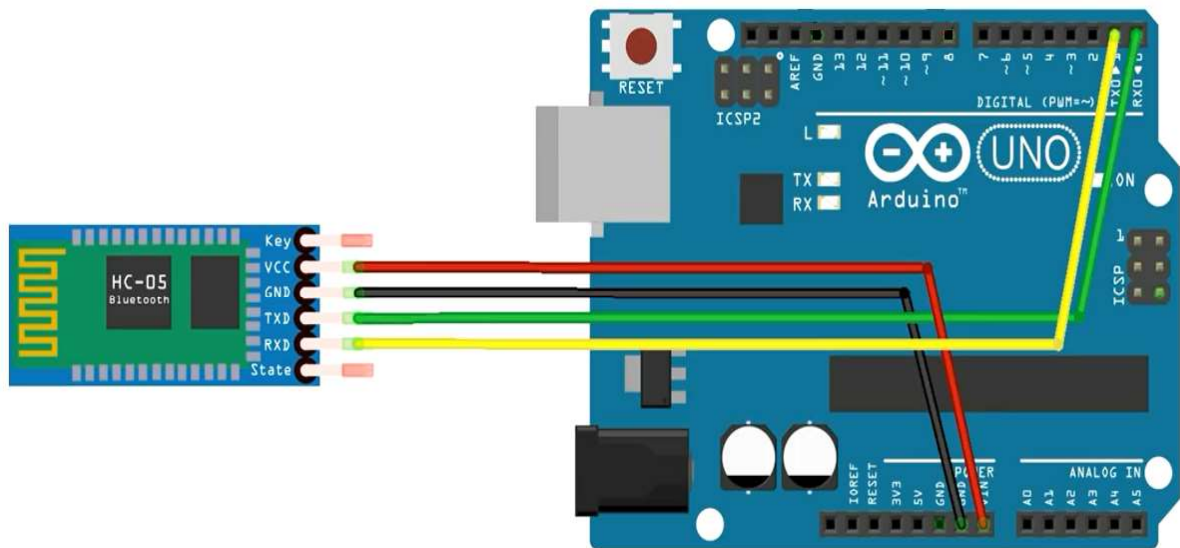
## **OBJECTIVES:**

- The device will detect objects by ultra sonic sensor.
- A fixed length can be declared in the code for detecting objects.
- The device will buzz when it will detect any object.
- The device can be controlled by Smartphone.
- Smartphone will send signal to the Bluetooth module attach with the device and thus the device will move.

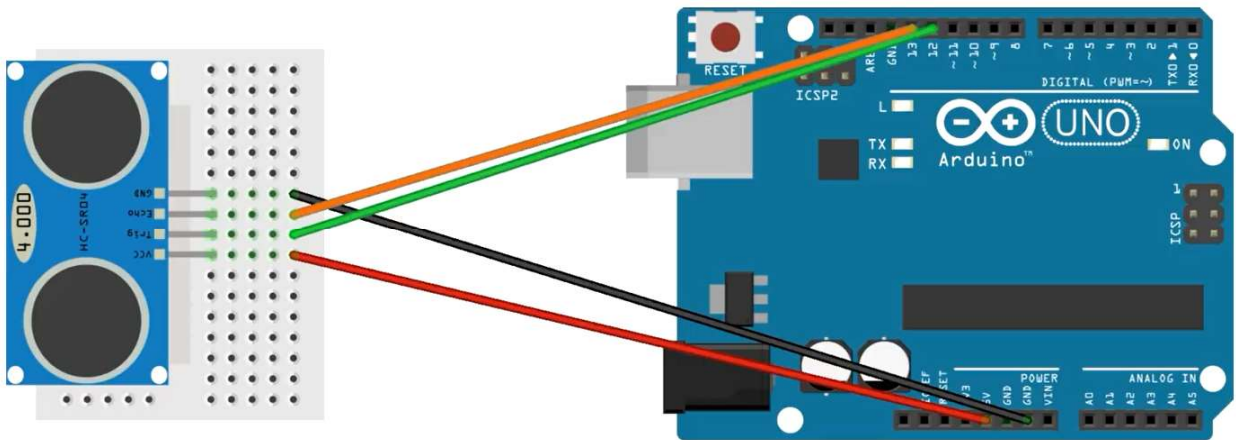
## **SYSTEM MODEL:**

The connections in our project-

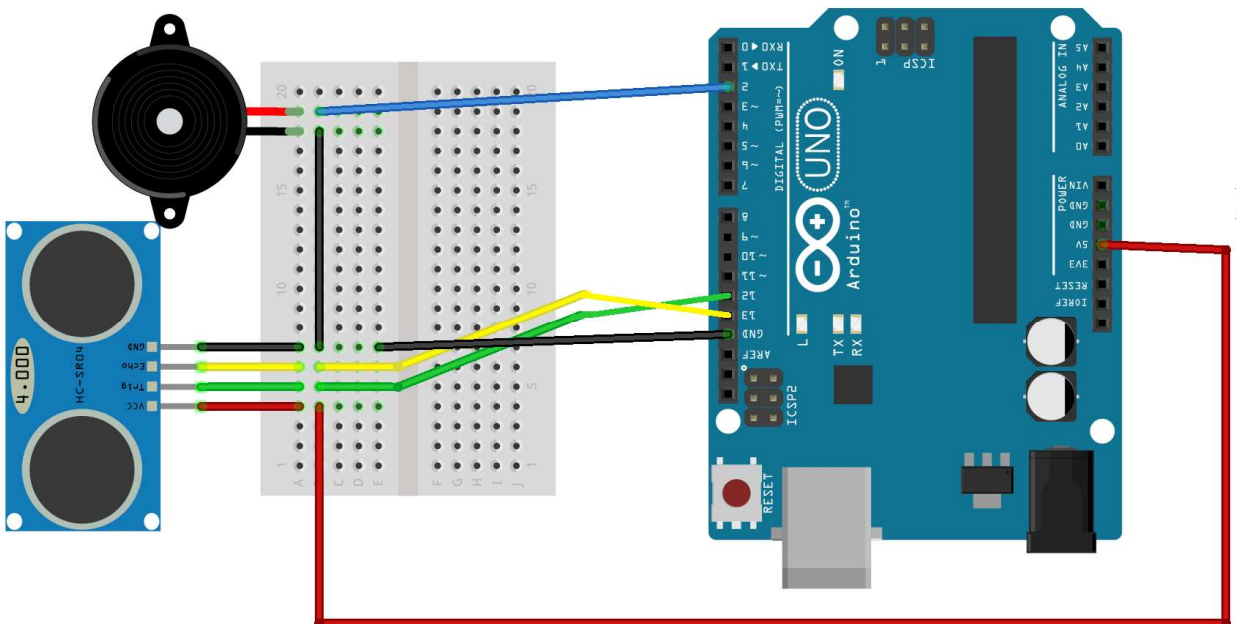
- Connection between Bluetooth module and Arduino Uno:



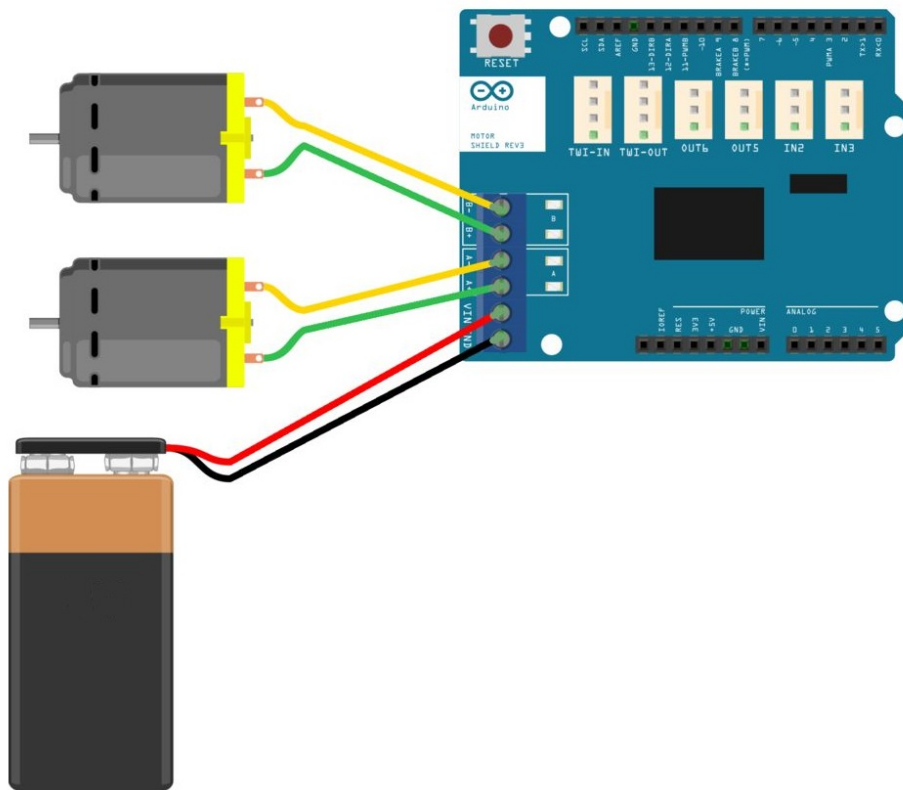
- Connection between Ultra Sonic Sensor and Arduino Uno:



- Connection among Buzzer, Ultra Sonic Sensor and Arduino Uno:



- Connection among Gear Motors, Motor Shield and Battery:



## SYSTEM DESCRIPTION:

Materials we used for this project are -

- *Arduino Uno:*

**Arduino Uno** is a microcontroller board based on the ATmega328P ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a



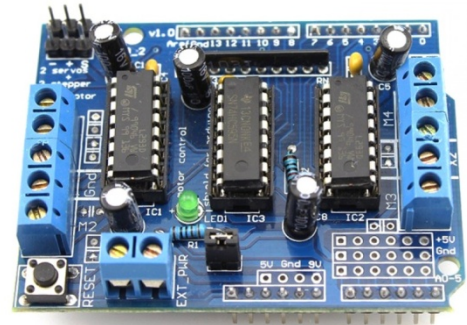
USB cable or power it with a AC-to-DC adapter or battery to get started.

The Arduino Uno can be programmed with the (Arduino Software (IDE)).

- ***Motor Shield:***

The **Arduino Motor Shield** is based on the L298 ([datasheet](#)), which is a dual full-bridge driver designed to drive inductive loads such as relays, solenoids, DC and stepping motors.

It lets you drive two DC motors with your Arduino board, controlling the speed and direction of each one independently. You can also measure the motor current absorption of each motor, among other features. The shield is TinkerKit compatible, which means you can quickly create projects by plugging TinkerKit modules to the board.



- ***Ultra Sonic Sensor:***

Basic Ultrasonic Sensor Operation

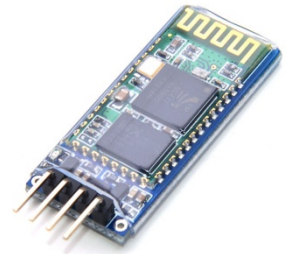
Ultrasonic sensors emit ultrasonic pulses that travel in a cone-shaped beam by using a vibrating device known as a transducer, generating the ultrasonic wave. The frequency of vibration of the transducer determines the range of an ultrasonic sensor. The sound waves transmit for progressively shorter distances along with the frequency increases. Conversely, the sound waves transmit for progressively longer distances as the frequency decreases. Therefore, short-range



ultrasonic sensors work best at higher frequencies, while long-range ultrasonic sensors work best at lower frequencies.

- ***Bluetooth module:***

- Bluetooth protocol: Bluetooth Specification v2.0+EDR
- Frequency: 2.4GHz ISM band
- Modulation: GFSK(Gaussian Frequency Shift Keying)
- Emission power: =4dBm, Class 2
- Sensitivity: =-84dBm at 0.1% BER
- Speed: Asynchronous: 2.1Mbps(Max) / 160 kbps, Synchronous: 1Mbps/1Mbps
- Security: Authentication and encryption
- Profiles: Bluetooth serial port
- Power supply: +3.3VDC 50mA
- Working temperature: -20 ~ +75 Centigrade
- Dimension: 26.9mm x 13mm x 2.2 mm



- ***Buzzer***







## **Working Process of the Device -**

- At first Smartphone has to be connected with the Bluetooth Module of the device.
- An app is used to control the movement of the device.
- When the device will move by the Smartphone, if it gets any object in 50cm distance it will activate the buzzer.
- Sensor calculates the distance by the time that the ultra sonic wave returns, thus the work is done.

## **PROGRAM CODE:**

```
#include <AFMotor.h>
```

```
AF_DCMotor right_motor(1, MOTOR12_8KHZ);
```

```
AF_DCMotor left_motor(2, MOTOR12_8KHZ);
```

```
int const trigPin = 12;
```

```
int const echoPin = 13;
```

```
int const buzzPin = 2;
```

```
String readString;
```

```
void setup() {
```

```
    Serial.begin(9600);
```

```
    right_motor.setSpeed(250);
```

```
    left_motor.setSpeed(250);
```

```
    pinMode(trigPin, OUTPUT);
```

```
    pinMode(echoPin, INPUT);
```

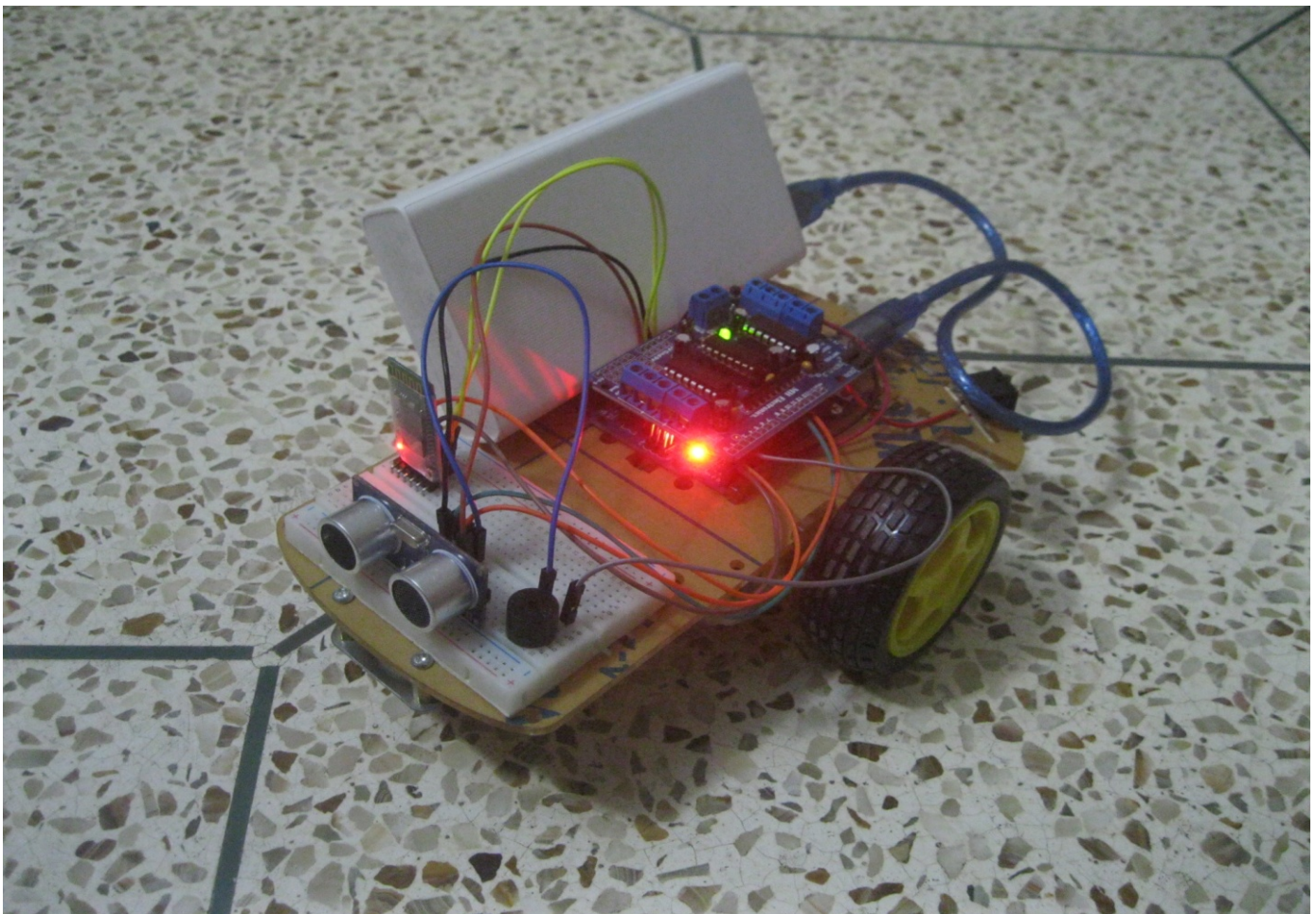
```
pinMode(buzzPin, OUTPUT);  
}  
void loop() {  
  while(Serial.available()){  
    delay(50);  
    char c=Serial.read();  
    readString+=c;  
  }  
  if(readString.length()>0){  
    Serial.println(readString);  
    if (readString == "BACKWARD"){  
      right_motor.run (FORWARD);  
      left_motor.run (FORWARD);  
      delay(500);  
    }  
    if (readString == "FORWARD"){  
      right_motor.run (BACKWARD);  
      left_motor.run (BACKWARD);  
      delay(500);  
    }  
    if (readString == "LEFT"){  
      right_motor.run (FORWARD);  
      left_motor.run (BACKWARD);
```

```
    delay(500);
}
if (readString == "RIGHT"){
    right_motor.run (BACKWARD);
    left_motor.run (FORWARD);
    delay(500);
}
if (readString == "STOP"){
    right_motor.run (RELEASE);
    left_motor.run (RELEASE);
    delay(500);
}
readString="";
}
int duration, distance;
digitalWrite(trigPin, HIGH);
delay(1);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = (duration/2) / 29.1;
if (distance <= 50 && distance >= 0) {
    digitalWrite(buzzPin, HIGH);
}
```

```
else {  
    digitalWrite(buzzPin, LOW);  
}  
delay(60);  
}
```

## **System Implementation:**

After connecting everything and implementing the code-



## **Discussion:**

- Sensor is very sensitive, so we had to handle it carefully.
- Bluetooth modulo can be damaged easily, so we had to disconnect it from the Arduino before uploading the code.
- Any connections with Arduino's TX and RX had to be disconnected, so that the Arduino doesn't get damaged.
- While uploading code other battery source has to be turned off or removed from the Arduino.
- We had to connect two separate power source (one with Arduino & another with Motor Shield) for better performance.

-----