Smart Dust Bin

Kiranmai Arva, Student, University of Regina

Abstract

This project introduces the design and development of a Smart Dustbin. Waste disposal strategy is crucial to the environment. This project is implemented on Arduino and Particle Photon. It integrates two Ultrasonic Sensors. one sensor is used to monitor the bin level while the other sensor is used to open and close of the lid when an object is brought near to the bin. When the bin is full a buzzer rings to notify the user. In this project signal communication between Arduino and smartphone is achieved in two ways one is by using a smartphone, buzzer can be stopped by interfacing the Bluetooth module. The other mode of communication is using a Photon particle a notification is sent to the mobile.

Keywords: Ultrasonic sensor,Servo Motor,Buzzer,Photon

1. Introduction

This project contains four modules. The first module is done using a servomotor and ultrasonic sensor when an object is detected at the lid of the bin it will open automatically. In the second module using the Ultrasonic sensor, the level of the bin is detected. Once the bin is full a buzzer rings and displayed on the LCD. In the third module, the user can stop the buzzer using the smartphone via a Bluetooth module. In the final module, using a Photon Particle a notification is sent to the user's smartphone about the bin level and also about the garbage pickup days. The detailed design is explained in the design description.

2. Motivation

Garbage releases toxic gases which cause air pollution. Garbage disposal is very important to society, it not only causes air pollution but also contaminates water and leads to several health issues, so garbage management is vital for the general wellbeing of the public. Technology improved drastically that nowadays in every aspect technology is applied. The traditional method of garbage disposal is no more a part of society as it requires a lot of human power. Automated machinery has come to lift bins and dispose of the garbage. The concept of smart bin came into existence however smart the system is people forget to dispose of their dustbins properly in the home. They constantly need provocations or alarms when the bin is filled and get acknowledged with the garbage pick off days so that they can keep their bins outside. All the remainders are notified and can be controlled using smartphones as today's everything is smartphone mechanized.

3. Background

In [1] this is IoT based smart dustbin using the ESP8266 module. The main functionality of this project is to track the amount of waste inside the bin and the dump vehicle can plan the route accordingly and decide which part of the cities dump to clean first. Using this device, the dump vehicle can plan the route in an optimized manner. The components used in this project are Arduino Nano, ESP8266, HC-SR04 Ultrasonic Sensor, 9v Battery,3.3v Regulator. The ultrasonic sensor is used to measure the bin waste level, to interact with the internet ESP8266 is used. The only

disadvantage of this project is it is cost effective, installing this device in each public bin is a tedious task.

In [2] this project, a Smart bin is made using the Ultrasonic sensor and servomotor. This smart bin can open and close the lid of the dustbin when an object is brought near to the lid. The shaft of the servomotor is connected to the thread and this thread is connected to the lid so that when the motor rotates the lid will open.

4. Design Description

4.1 Overview

The design and description of smart Bin are described in the following sessions. A total of four modules are explained in detailed. Each module is important to the next higher layer modules. Below Illustrates the different components involved in this project their functionalities and why those components are used and also challenges faced while integrating all the modules.

4.2 Components Required

- Arduino Uno
- Ultrasonic sensor
- Dustbin
- Servo motor
- Buzzer
- LED
- Smartphone
- Bluetooth Module
- Particle Photon
- LCD Screen

4.3 Software Installations Required

- Arduino IDE
- Particle IDE
- Bluetooth terminal Application in smartphone
- IFTT Application

4.4 Module 1

Components used in this module are

- Ultrasonic Sensor
- Servomotor
- Lid of the bin

4.4.1 Description:

When an object is brought near to the lid of the dustbin, the sensor detects the object and sends signals to Arduino, according to the code written in the microcontroller. It sends the signal to the servo motor and motor rotates its knob given an angular. The servomotor is attached to the lid of the bin when the servo motor receives the signal from Arduino it rotates its shaft as well as the lid of the bin and allows the user to place the waste into the bin and closes automatically. The user's hand will not contact with the dustbin for throwing the waste into the bin.

4.4.2 Component description:

Ultrasonic sensor: The model number used in this project is HC-SR04. It can measure the distance from 2cm to 400cm. It contains a transmitter, receiver and a control circuit. The transmitter transmits ultrasonic sound when these rays encounter object rays reflect and reache the receiver. The distance is calculated by the time taken for the sound to travel forward and backward from the sensor to object times the speed of the sound. It contains four pins namely Trig pin, Echo pin, Vcc, Ground. The main reason behind choosing this component is it can calculate the distance of the object and perform certain actions. In this project the trig pin of the Ultrasonic sensor is connected to the digital pin 5 and echo pin is connected to the digital pin 6.

Servo motor: [3] Servomotor works on the principle of pulse width modulation. It works in a closed loop mechanism and contains three parts Controlled device, Output sensor, the Feedback system. Encoder controls the position feedback. It can rotate up to 180 degrees. Precise control of the shaft can be achieved. Servomotor contains three wires one is positive second is the negative third to the digital pin. In this project, the servomotor is attached to the digital pin 11. When Arduino sends the signal to the servomotor the shaft of the motor moves according to the angular position given. So, with the above features, only the servomotor can rotate the lid of the bin. FIG1 :The Block Diagram of module 1 is in the appendix

4.5 Module 2

Components used in this module are

- Ultrasonic Sensor
- Buzzer
- LCD

4.5.1 Description:

The sensor continuously monitors the waste level of the bin and displays the amount of waste the bin is full on the LCD screen. When the bin is full a message is displayed on the LCD screen and a buzzer rings to indicate the bin owner that its time to empty the bin. A buzzer is arranged because some lazy human beings require constant provocation to empty the bin. Fig5:shows the Block diagram of Module2

4.6 Two modes of communication

- Module 3 explains the communication between the Arduino and smartphone using Bluetooth module
- Module 4 explains the communication between the Arduino and smartphone using a photon particle.

4.6.1 Module 3

Components used in this module are

- Bluetooth Module
- Smartphone
- Buzzer

Bluetooth Module: In [4] The Bluetooth module uses radio waves for communication. In this project, the Hc05 module is used. This can range up to ten meters. It contains a total of six pins the first state pin which shows the status by blinking the led if Bluetooth is connected to the device. Second and third pins are RX and TX pins they help in interfacing the communication. Vcc, Ground, and key are the other pins. It also contains the button it allows the user to choose the modes between command and application. TX and RX pins are connected to digital pins 3 and 4.

Bluetooth Terminal:Bluetooth terminal is an application that can connect to the Bluetooth module via smartphone Bluetooth and can send signals to the Bluetooth module. As it establishes serial communication, so it can exchange the data bidirectionally [5]. The buzzer and led both can be controlled using this app by sending input signals from this app to the Bluetooth module. Finally, the Bluetooth module sends signals to the Arduino.

4.6.2 Module 4

The Components used in this module are

- Photon
- Buzzer
- Ultrasonic sensor
- IFTT App

This module functions similar to module 3 that is the ultrasonic sensor monitors the waste level of the bin when the bin is full buzzer is rung and led flashes. Both buzzer and LED can be controlled using a smartphone. The only difference is all the actives are performed using particle photon instead of using Arduino. A notification is also sent to the IFTT app and buzzer is also controlled using this app.

Particle photon: Particle photon is a microcontroller with a built-in Wi-Fi chip. Same as Arduino it also contains 3.3 volts power, Digital pins, and analog pins. The photon has a web IDE where the code is verified and flashed into the microcontroller of the photon. Before using the photon particle, the device should be claimed, only if the device is claimed it shows the status of the particle in the IDE and one can flash the code into it. The photon can be controlled using the IFTT app in the smartphone from anywhere in the world. The only criteria are the photon particle should be in the same network used while claiming the device.

4.7 Difference between the two modes of communication mentioned above

In the first mode, the communication between the Arduino and smartphone is achieved using an interface called Bluetooth module. In the second mode, there is no interface device for communication between the particle photon and smartphone. In the first mode, the communication is only within a particular range whereas in the second mode of communication one can control the particle photon from anywhere in the world till the photon is in the registered network. In both cases, two-way communication is possible in the first case the communication through Bluetooth is a bit difficult compared to the second case. Of both cases, the second one is the best communication mode compared to the first if the usage of the particle is acquainted.

5. Build Description of the Circuit

Following are the steps to build the circuit

1. Take an Arduino Uno, Two Ultrasonic sensors, Servo motor, Piezo Buzzer, Bluetooth Module and Particle photon.
2. First Ultrasonic sensor Trig and Echo pins are connected to digital pins 5 and 6 respectively, the other two pins are ground and Vcc.
3. Connect servo motor yellow pin (signal pin) to Arduino digital pin no. 11.
4. Connect servo +Vcc pin to Arduino +5V pin.
5. Connect servo GND pin to Arduino GND pin.
6. Second Ultrasonic sensor Trig and Echo pins are connected to digital pins 10 and 11 respectively, the other two pins are ground and Vcc.
7. Positive pin of the digital pin is connected to the Digital pin no. 2
8. The Negative pin of the buzzer is connected to the ground.
9. Bluetooth module Transmitter (Tx) is connected to the Receiver pin (Rx) of Arduino.
10. Bluetooth module Receiver pin (Rx) is connected to the Transmitter pin of Arduino.
11. Ground pin and cathode pins of LCD are connected to the ground.
12. Vcc and Anode pins of LCD are connected to the 5volts.
13. Contrast pin is connected to the signal pin of the potentiometer.
14. The other pins of LCD are connected to the digital pins.
15. The potentiometer Vcc is connected to 5 volts and Grnd to ground.

Steps to be followed while connecting the particle photon

1. Claim the device by signing in to the particle Account.
2. Enter the Wi-Fi credentials when it asks while claiming the device.
3. Once the particle is claimed successfully, the led of the particle blinks with cyan color.
4. In the particle Web IDE under devices one can see the device after claiming the device.
5. Insert the particle into the breadboard.
6. Ultrasonic sensor Trig and Echo pins are connected to digital pins 5 and 6 respectively, the other two pins are ground and Vcc.
7. Positive pin of the digital pin is connected to the Digital pin no. 2
8. The Negative pin of the buzzer is connected to the ground.

6. Description of the Code Build process

6.1 Libraries

Libraries used in building this project are

- Servo.h
- SoftwareSerial.h
- LiquidCrystal.h

a) In order to turn the shaft of the servo motor, it has to receive a high pulse signal. Servo library is used to set up the signal, While the shaft rotates meanwhile it can check the other part of the code. Servo library continuously runs background [6].

"myservo.attach(11);" //Initializing the signal to pin 11

"myservo.write(dis+160);" //if the given condition is true it has to execute the Myservo.write it turns the shaft to 160 degrees.

b) Built-in support for serial communication is present in Arduino for the pins 0 and 1. The other pins do not function as RX and TX the other digital pins to function as serial pins it has to use the software serial library. SoftwareSerial BTserial(3, 4) // Here we are declaring the digital pins 3 and 4 for serial communication, BT is Bluetooth.

c) LiquidCrystal Library this library helps Arduino to communicate with the LCD display. LiquidCrystal lcd(12, 11, 6, 7, 8, 2) // declaring the pins connected to the arduino. lcd.print(Opened) // using this function one can display the required message on LCD.

6.2 Functions used in this project are

- void firstsensor()
- void secondsensor()
- void blue()

a) void firstsensor() : The main role of this function is to open the lid of dust bin if an object is detected. This is achieved by writing the condition on the distance measured by the Ultrasonic sensor. The Distance of the ultrasonic sensor is calculated using the below code.

$$duration = pulseIn(echoPin, HIGH)$$

Duration is the amount of time it takes to reach the obstacle plus the reflected rays back to the sensor

$$Distance = 0.034 * (duration/2)$$

Distance is calculated using the duration.

$$if(distance < 27)$$

By mentioning the conditions on distance if the condition is true it enters the loop and executes the code in the loop. Here if the distance is less than the twenty-seven it rotates the shaft of the servo motor else it does not rotate the shaft of the motor.

b) Void SecondSensor(): The main role of this function is to ring the buzzer if the bin is full and display the same on the LCD screen. The function of the second Ultrasonic sensor here again the distance is calculated using different echo pin declared initially. if distance is 0.034*(duration/2)

if above is the case then the buzzer should ring and display on LCD that the bin is full digitalWrite(buzzPin, HIGH); digitalWrite(ledPin, HIGH); lcd.print(Bin is full).

c) Void Blue (): This function is used for Bluetooth communication. Using this function, a buzzer is stopped using an app Bluetooth terminal in a smartphone. Using a smartphone, a string is sent to the Bluetooth module which in turn sends the data to Arduino if the condition is true then it stops the buzzer. if(Serial.available()) // check if any data is available while (Serial.available()¿) // if data is available then enter the loop. (inputString == "b") // if the available data is the string B then turn off the buzzer.

7. USER MANUAL

1. This smart bin can be used in both homes and the public.
2. The user has to set up a Bluetooth terminal app on their smartphone.
3. Turn on the smartphone Bluetooth.
4. Open the app, under devices select the appropriate Bluetooth device.
5. Once connected it shows the status connected.
6. If buzzer rings open the app and send the input string b to stop the buzzer.

If the communication is through photon particle

1. Install the IFTT app in the smartphone.
2. Connect the photon particle.
3. All the available services are displayed.
4. Add the service and check in the available photon services for the connected photon.
5. Now one can control the photon form Mobile.
6. All the notifications are sent to the mobile from particle photon.

8. Evaluation

8.1 Testing

The Bluetooth module uses radio waves for communication. In this project, the Hc05 module is used. This can range up to ten meters. It contains a total of six pins the first state pin which shows the status by blinking the led if Bluetooth is connected to the device. Second and third pins are RX and TX pins they help in interfacing the communication. Vcc, Ground, and key are the other pins. It also contains the button it allows the user to choose the modes between command and application. TX and RX pins are connected to digital pins 3 and 4.

8.2 Challenges and Issues Encountered

1. Data transfer issue while sending data to Bluetooth Module: Initially when the data is sent from the app from the smartphone to the Bluetooth module the required action to be performed is to stop the buzzer which is not achieved. When given an input using serial monitor the buzzer stopped so there is no problem in communication between the Bluetooth module and Arduino. The problem is Bluetooth is not receiving the signal from the phone. The issue is resolved by adding Serial.read() in the code part. Got this issue resolved from the Internet [7].

2. Problems with RX and TX pins: Initially the Rx pins of Bluetooth module is connected to the Rx pin of Arduino and similarly, the TX pin of the Bluetooth module is connected to the Tx pin of Arduino but later realized that the pins are connected erroneously. The TX pin of Bluetooth module is linked to the RX of Arduino correspondingly the RX pin of Bluetooth module is connected to the TX of the Arduino because the Tx pin acts as transmitter pin for Bluetooth module. Sending the signal to the Arduino, it acts as a receiver as Bluetooth acts as Transmitter TX pin is connected to RX. Arduino TX pin is connected to the Bluetooth module RX pin here the signal from Arduino is sent to the Bluetooth module, so TX is connected to RX.

3. Error while Uploading the code to Arduino: avrdude: stk500 getsync (): not in sync: resp=0x00 encountered this error while uploading the code to the Arduino. This is due to RX and TX pins of Arduino are linked to the Bluetooth module. The issue is fixed by disconnecting the Wires of RX and TX pins of Arduino while the code is uploaded. Once done with uploading pins are connected back.

4. Problem while claiming the photon device: Once the device is claimed under one user the device cannot be claimed by others. if one tries to reclaim the same Photon Particle The following error message is displayed, the device has already been claimed by another user. The photon should be unclaimed before reclaiming the particle photon. Un claim can be done using the Particle app. In the settings an option unclaim is present. Once the device is unclaimed, one can register or claim the device using their credentials. The device setup can be done easily.

5. Network problem with the particle photon: While setting up the photon particle the device asks for the Wi-Fi network credentials. For the device to work effectively the Photon should be in the same network, which is used while setting up the particle. One can control the particle photon from anywhere in the world if the particle is on the registered network. If the particle is connected to the other network and tries to control the particle from a phone particle will not function. Instead, it shows the device status offline.

9. Milestones

A total of five milestones is mentioned in the project proposal. The first milestone is to identify and gather all the required devices for the project. All the devices are gathered appropriate time, except the particle photon. The second milestone is to build the circuit and this goal is reached. The third milestone is to work on code and buzzer part, this task is accomplished within a given time. Milestone four and five are interchanged compared to the proposal. Milestone four is to stop the buzzer using a smartphone via Bluetooth module which is attained timely. The final milestone is to send the notification from Arduino to the smartphone which initially tried by the Bluetooth module later realized that for sending the notification message wi-fi is required. which is tough to attain using the Bluetooth module so I started working on the Particle photon which is a new microcontroller with a new interface. Initially faced a problem with the device later I am able to claim the device and flash the code into the Particle photon and send the notifications to mobile finally. The final milestone has accomplished but not on time. All the roles like Gathering the components, Building the Hardware, Programming, Documentations are done by me.Below is the link of the Github Repository: https://github.com/Kiranmai95/Smart-Bin

This is my link: Smart Bin.

10. Future work

The whole project can be done without using Arduino only by using a photon particle as it is also a microcontroller with built in WI-FI.Using the latest technologies like IoT devices and Cloud Servers this project can be taken to the next higher level. When bins are placed in public it's the responsibility of the government to handle the trash as handling the trash is very important to the environment. In the future, the dustbins are designed in a way that when waste is placed in the bin. The bin will automatically separate the recyclable and nonrecyclable waste. when the trash is thrown in the bin it reaches the transient layer of the bin. where the image of the scrap is captured and send to the cloud, it then decides whether it is recyclable or nonrecyclable. Once the decision is made by the server it separates the waste this is just done in a fraction of seconds. Using sensors it also detects the vapor content in the bin and garbage level of the bin and infers the respective persons.

11. Conclusion

Finally, Smart Bin is designed with the following features. When an object is brought near the dustbin the lid of the bin will open automatically. The waste level of the bin is monitored continuously and displayed on the LCD screen when the bin is full then a buzzer rings and also displayed on the screen. Here in this project, two modes of communication The first method is based on Bluetooth module and smartphone one can turn off the buzzer. In the second method, a particle photon which is a microcontroller with inbuilt wifi capability is used so in this method an Ultrasonic sensor and buzzer are used when the bin is full it can send a notification to phone and also about the garbage pickup days. Apart from this particle photon can be controlled from the phone using an app IFFTT.

Appendix A.

Arduino Code

```
#include <Servo.h>
#include <LiquidCrystal.h>
char cache;
Servo myservo;
#include <SoftwareSerial.h> //.h files are header files

SoftwareSerial BTserial(3, 4); //Initializing the pins
LiquidCrystal lcd(12, 11, 6, 7, 8, 2);

int dis = 20;
const int trigPin = 5;
const int echoPin = 6;
long duration;
float distance;
int const trigPin1 = 10;
int const echoPin1 = 9;
int const buzzPin = 2;
const int ledPin = 13;
long duration1;
float distance1;
String inputString="";
void setup()
```

```
{
   Serial.begin (9600);
   pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(ledPin, OUTPUT);
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
   pinMode(buzzPin, OUTPUT);
    myservo.attach(11);
    lcd.begin(16, 1);
  myservo.write(dis);
}

void firstsensor() //First function
{

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  duration = pulseIn(echoPin, HIGH);
  distance = 0.034*(duration/2); //Calculating the distance
   if (distance < 27) //checking the condition on distance
  {

    myservo.write(dis+160); //servomotor rotates
    lcd.print(  Opened  );
    delay(1000);
    lcd.print(  Closed  );
  }
  else
  {

      myservo.write(dis);
  }
  delay(300);
}
void secondsensor() //Second function to measure waste leel in bin
{
  digitalWrite(trigPin1, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin1, HIGH);
  delayMicroseconds(10);

duration1 = pulseIn(echoPin1, HIGH);
 distance1 = 0.034*(duration1/2);

if (distance1 <= 10 && distance1 >= 0) //Condition based on distance
{
// Buzz

digitalWrite(buzzPin, HIGH); // if condition is true buzz pin is high
digitalWrite(ledPin, HIGH);
lcd.print( Bin  is  full ).
```

```
} else {
// Don't buzz
digitalWrite(buzzPin, LOW);
digitalWrite(ledPin, LOW);
}
// Waiting 60 ms won't hurt any one
delay(60);

}
void blue() //function for bluetooth module
{
char inChar = (char)Serial.read();
inputString += inChar;
}
Serial.println(inputString);
while (Serial.available()>0)
{
cache = Serial.read();
}

if(inputString == "a")
{
//digitalWrite(13,HIGH);
digitalWrite(buzzPin, HIGH);

}
else if(inputString == "b") //if b is given as input the buzzer will stop
{
//digitalWrite(13,LOW);
digitalWrite(buzzPin, LOW);
delay(10000000);
}
delay(10000);
inputString = "";
delay(100000000);
}
//delay(100000000);
}

void loop() {
Serial.println("\n");
firstsensor();
secondsensor();
blue();

}
```

Photon Particle code

```
const int trigPin = D5;
const int echoPin = D6;
const int buzzer = D2;
const int ledPin = 13;

// defines variables
```

```
long duration;
int distance;
int safetyDistance;


void setup() {
pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin, INPUT); // Sets the echoPin as an Input
pinMode(buzzer, OUTPUT);
pinMode(ledPin, OUTPUT);
Serial.begin(9600); // Starts the serial communication
}


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;

safetyDistance = distance;
if (safetyDistance <= 5){
  digitalWrite(buzzer, HIGH);
  digitalWrite(ledPin, HIGH);
}
else{
  digitalWrite(buzzer, LOW);
  digitalWrite(ledPin, LOW);
}

// Prints the distance on the Serial Monitor
Serial.print("Distance: ");
Serial.println(distance);
}
```

Appendix B.

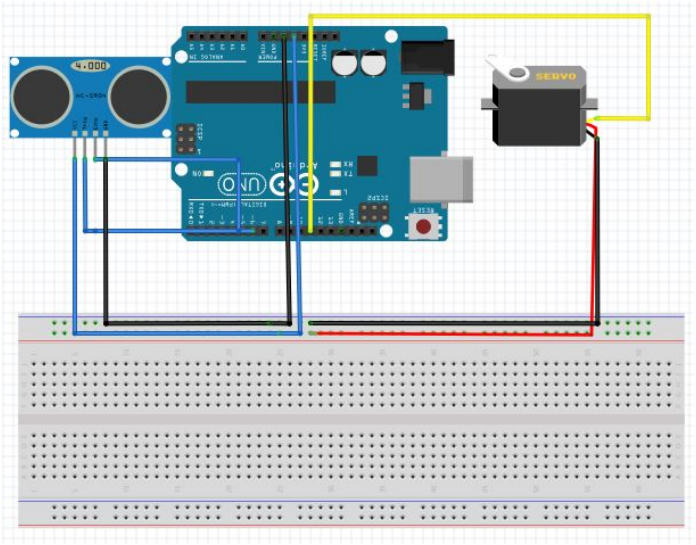

Figure 1: Module1 Circuit Diagram

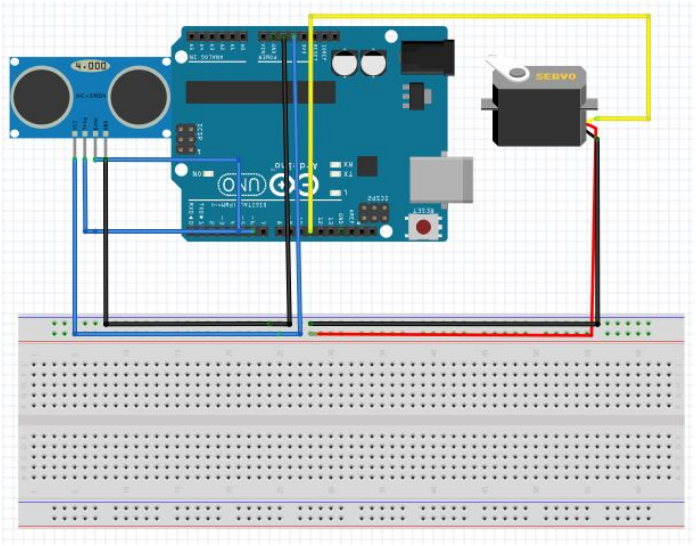

Figure 2: Module2 Circuit Diagram

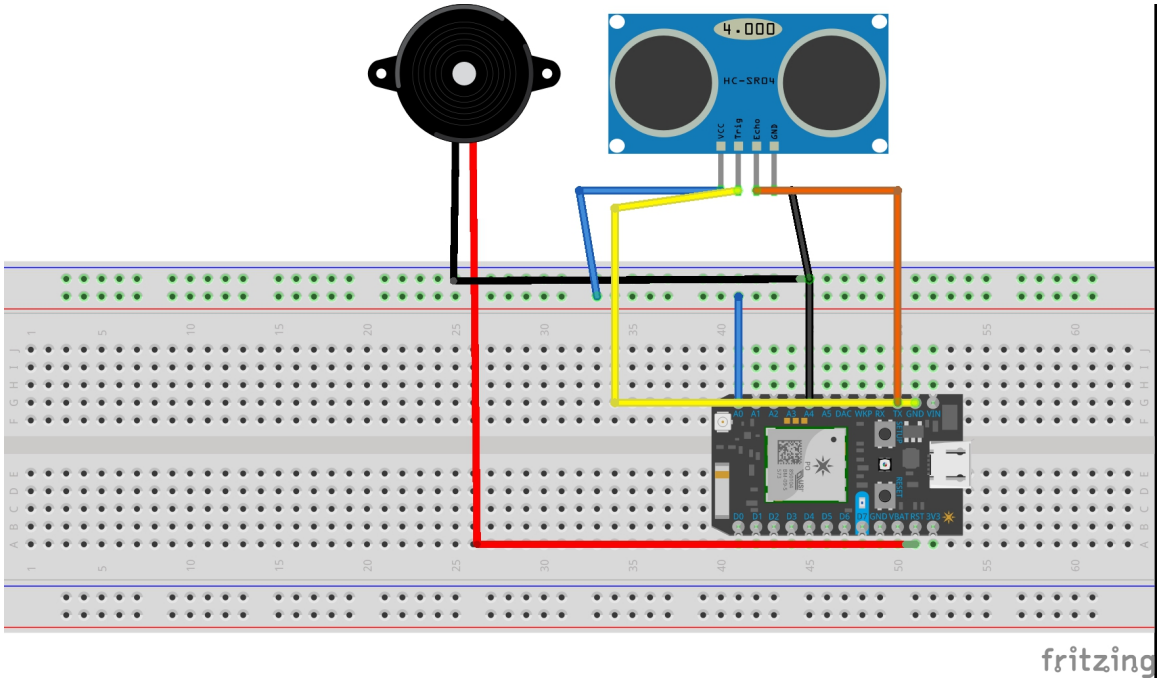

Figure 3: Particle photon Circuit diagram

Figure 4: Block Diagram of Module1

References

[1] onlineHackster.io,title=IOT Smart Dustbin using Arduino Nano and ESP8266,url=https://www.hackster.io/arnabdasbwn/iot-smart-dustbin-using-arduino-nano-and-esp8266-2187e4 2

[2] onlineArduino Project Hub,title=Smart Dustbin,url=https://create.arduino.cc/projecthub/msr048/smart-dustbin-064003 3

[3] onlineEn.wikipedia.org Hub,title=Servomotor,url=https://en.wikipedia.org/wiki/Servomotor 4

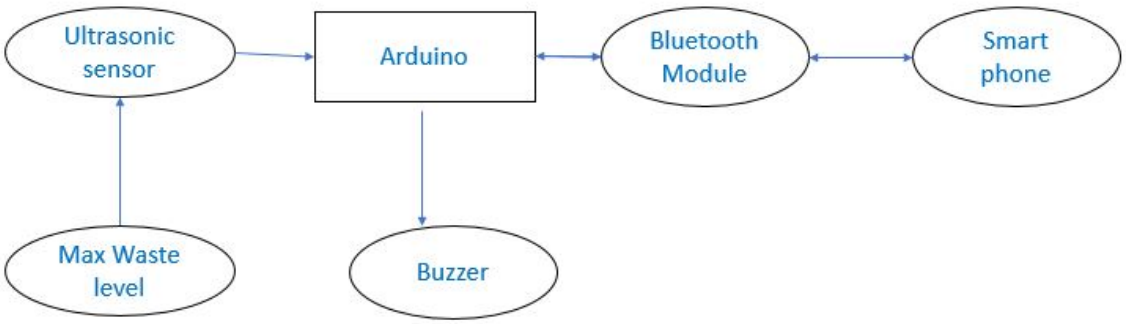

Figure 5: Block Diagram of Module2

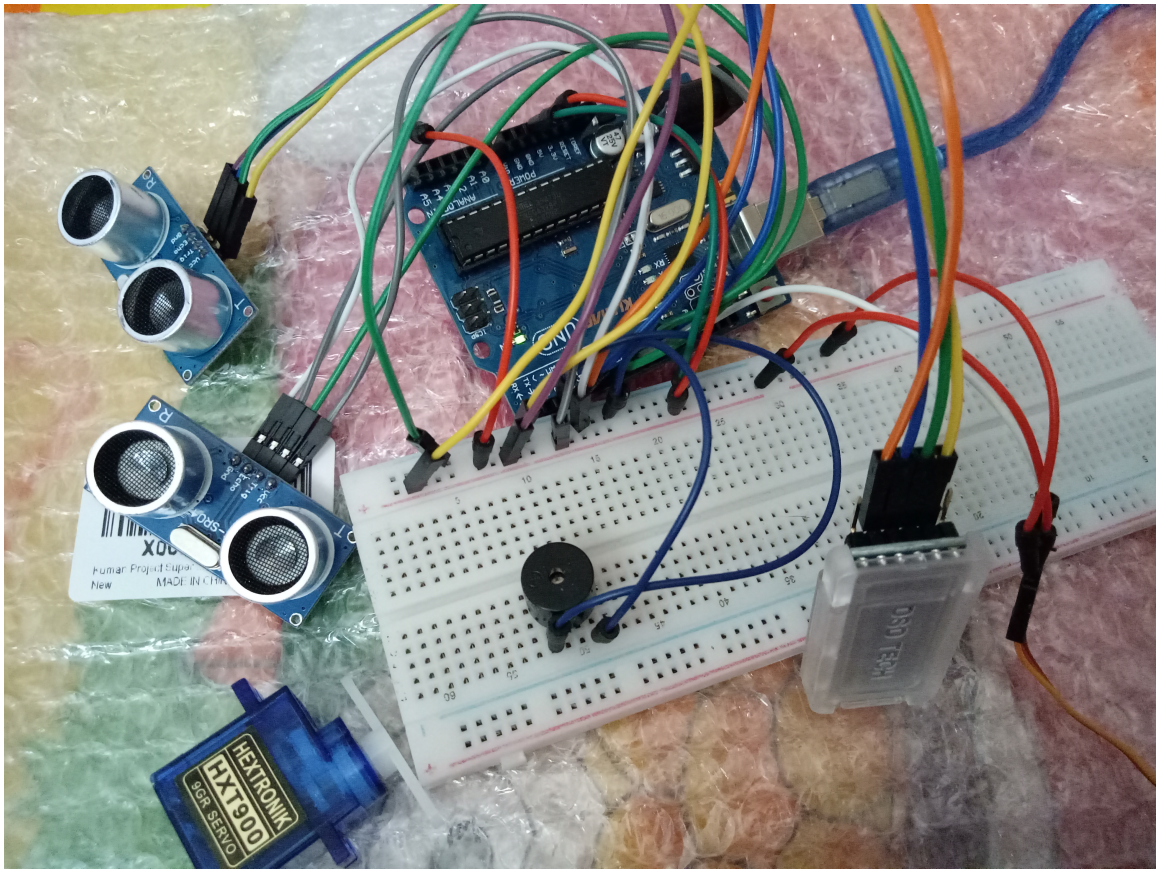

Figure 6: Circuit diagram of Arduino

[4] online Hobbyist.co.nz,title=Bluetooth Module (Device),url= https://www.hobbyist.co.nz/?q=bluetooth-module-device 5

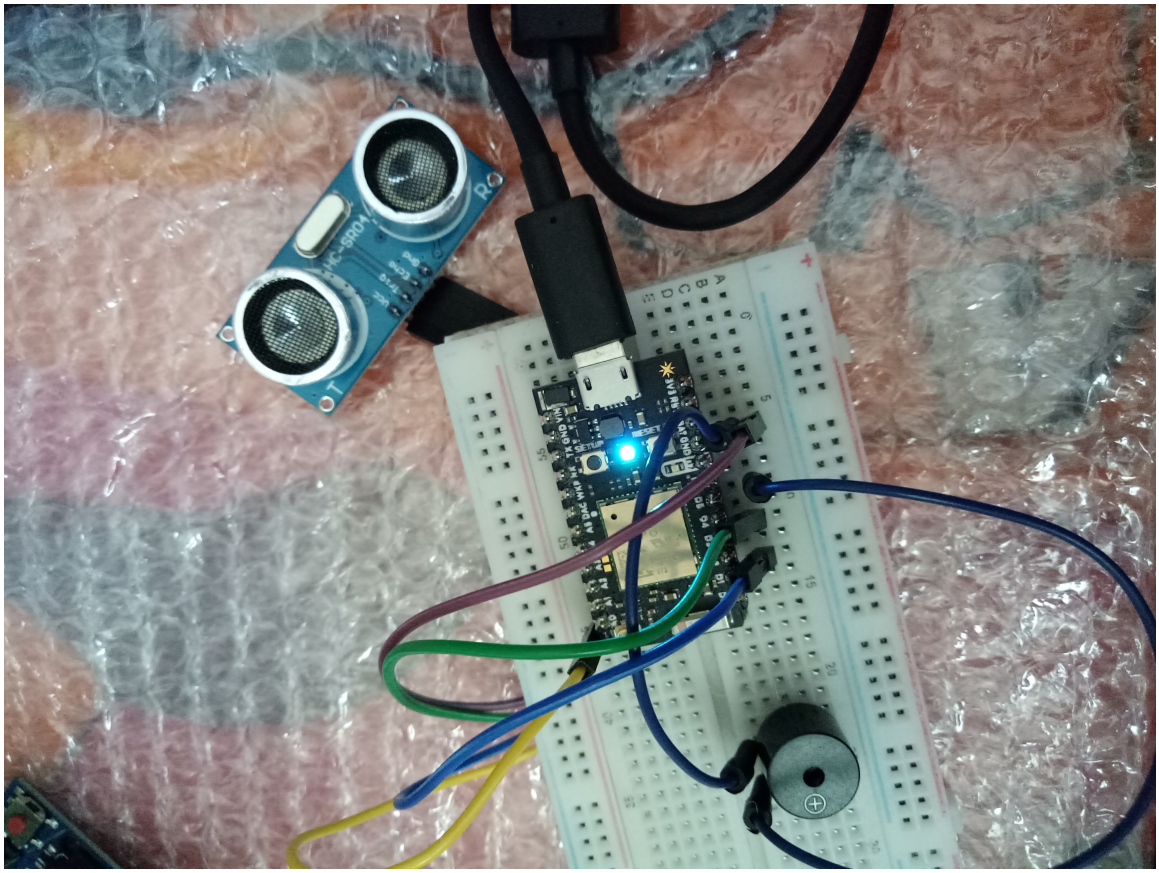

Figure 7: Circuit diagram of Photon

[5] online npm ,title=web-bluetooth-terminal,url=https://www.npmjs.com/package/web-bluetooth-terminal 6

[6] online Learn.parallax.com,title=How to Use the Arduino Servo Library — LEARN.PARALLAX.COM,url= https://learn.parallax.com/tutorials/robot/shield-bot/robotics-board-education-shield-arduino/chapter-2-shield-lights-servo-11 7

[7] onlineArduino Stack Exchange,title=Problem with hc-05 bluettoth module,url= https://arduino.stackexchange.com/questions/6882/problem-with-hc-05-bluettoth-module