SMART DUSTBIN

A mini-project report submitted for
Wireless Network(Semester VI)

by

Clint Dmello(8484)

Selvin Tuscano (8437)

Cyrus Dbritto(8483)

Under the guidance of

Prof. Vaibhav Godbole

(sign with date)

MOULDING ENGINEERS WHO CAN BUILD THE NATION

DEPARTMENT OF INFORMATION TECHNOLOGY
Fr. Conceicao Rodrigues College of Engineering
Bandra (W), Mumbai - 400050
University of Mumbai

Approval Sheet

Project Report Approval

This project report entitled by **Smart Dustbin using** by **Clint Dmello, Selvin Tuscano,Cyrus Dbritto and Veena Gaurea** is approved as mini project in Third year Engineering, Information Technology**.**

Examiners

1._______________________

2._______________________

Date:

Place:

‘

Abstract

The system smart waste-bin that can managed the waste in a smart city project. The system consist of sensors to measure the volume(distance) of waste and the level of waste inside the bin. The system also adapt with network environment, to manage all information from waste management. As the result we proposed a prototype of smart waste-bin that suitable for many kind of conventional waste-bin. cause. In this project, smart bin is built on a microcontroller based platform Arduino Mega board which is interfaced with Ultrasonic sensor. Ultrasonic sensor is placed at the top of the dustbin which will measure the stature of the dustbin. Arduino will be programmed in such a way that when the dustbin is being filled, the remaining height from the threshold height will be displayed.. Once these smart bins are implemented on a large scale, by replacing our traditional bins present today, waste can be managed efficiently as it avoids unnecessary lumping of wastes on roadside.A smart dustbin solves the problem of monitoring by sending a notification to the user about the garbage levels after which he can empty the dustbin.

Table of Content

Lab Outcomes

LO1: Identify the requirements for the real world problems.

LO2: Conduct a survey of several available literatures in the preferred field of study.

LO3: Study and enhance software/hardware skills.

LO4: Demonstrate and build the project successfully by hardware requirements, coding, emulating and testing.

LO5: To report, present and demonstrate an ability to work in teams and manage the conduct of the research study.

Rubrics for the assessment (LO1):

Indicator	Very Poor	Poor	Average	Good	Excellent
Timeline (2)	More than a 3session late (0)	More than a 2session late (0.5)	More than a 2session late (1)	More than a 1session late (1.5)	Early or on time (2)
Identify real world problems(4)	NA (0)	Very poor identification of real world problems(1)	Poor identification of real world problems(2)	Good identification of real world problems.(3)	Accurate identification of the real world problems. (4)
Design the problem solution (4)	No requirement analysis is done(0)	Very poor requirement analysis is done(1)	Poor requirement analysis is done(2)	Good requirement analysis is done(3)	Requirement analysis done with best solution design(4)

Marks:

Rubrics for the assessment (LO2):

Indicator	Very Poor	Poor	Average	Good	Excellent
Timeline (2)	More than a 3session late (0)	More than a 2session late (0.5)	More than a 2session late (1)	More than a 1session late (1.5)	Early or on time (2)
Selection of Sources(8)	No information is gathered from a range of sources.(0)	Information is gathered from a limited number of sources.(2)	Sources rely heavily on a small number of sources and are not considered to be from authoritative sources(4)	Information is gathered from a range of sources but do not entirely reflect the breadth of the debate(6)	Information is gathered from a wide range of journals, books and related authoritative research materials.(8)

Formatting and Presentation of Assignment (4)	Document contains many errors in formatting, punctuation and writing was incoherent. (0)	Document contains many errors in punctuation and formatting. Referencing is not consistent with chosen style guide. Writing style lacks clarity.(1)	Document contains few errors in formatting and punctuation. Style of referencing is generally consistent with chosen style guide. Writing style is coherent(2)	Document contains few errors in formatting and punctuation. Style of referencing is generally consistent with chosen style guide. Writing style is coherent(3)	Document is professionally presented with virtually no errors in punctuation and is in the correct format. The style of referencing is consistent with chosen style guide. Writing style is clear and engaging(4)

Marks:

LO3: Study and enhance software/hardware skills.

Rubrics for the assessment:

Indicator	Very Poor	Poor	Average	Good	Excellent
Timeline (2)	More than a 3session late (0)	More than a 2session late (0.5)	More than a 2session late (1)	More than a 1session late (1.5)	Early or on time (2)
Installation of Arduino IDE/Raspbian OS(4) Programming	NA(0)	Installation not done(1)	Installation With some drivers(2)	Installation without drivers(3)	Installation with drivers done(4)
Interfacing of sensors to Arduino/Raspberry board	NA(0)	Unable to do connection and but required output not obtained.(1)	Able to do connection and but required output not obtained(2)	Able to do connection and required output is obtained and no libraries are installed(3)	Able to do connection and required output is obtained and libraries are installed(4)
Interfacing of sensors to Arduino/Raspberry board	NA(0)	Unable to do connection and but required output not obtained.(1)	Able to do connection and but required output not obtained(2)	Able to do connection and required output is obtained and no libraries are installed(3)	Able to do connection and required output is obtained and libraries are installed(4)
Sending data on ThingSpeak ,Analysis of Data	NA(0)	No data sent onthingspeak(1)	Data sent on thingspeak and no analysis done(2)	Data sent onthingspeak and some analysis notdone(3)	Data sent on thingspeak and analysis done(4)

Marks:

LO4: Demonstrate and build the project successfully by hardware requirements, coding, emulating and testing.

Rubrics for the assessment (LO4):

Indicator	Very Poor	Poor	Average	Good	Excellent
Timeline (2)	More than a 3session late (0)	More than a 2session late (0.5)	More than a 2session late (1)	More than a 1session late (1.5)	Early or on time (2)
Code design(4)	NA (0)	Very poor code design with no comments and indentation(1)	Poor code design with very comments and indentation (2)	Design with good coding standards (3)	Accurate design with better coding standards (4)
Demo	No system set up was shown(0)	Incomplete System set up was shown.(1)	Partially Complete set up shown with working(2)	Almost Complete set up shown with working(3)	Complete set up shown with working(4)

Marks :

LO5: To report, present and demonstrate an ability to work in teams and manage the conduct of the research study.

Rubrics for the assessment (L04):

Indicator	Very Poor	Poor	Average	Good	Excellent
Timeline (2)	More than a 3session late (0)	More than a 2session late (0.5)	More than a 2session late (1)	More than a 1session late (1.5)	Early or on time (2)
Teamwork and cooperation(4)	The project appears to have been carried out by only minimal (1-2) members for different tasks. (0)	) The project appears to have been carried out by only by 2) members	. The project was carried out by most (3-4) members	The project was carried out by most (3-4) members	The project was carried out by all members.

Formatting andPresentation of Report (4)	Document contains many errors in formatting, punctuation and writing was incoherent. (0)	Document contains many errors in punctuation and formatting. Referencing is not consistent with chosen style guide. Writing style lacks clarity.(1)	Document contains few errors in formatting and punctuation. Style of referencing is generally consistent with chosen style guide. Writing style is coherent(2)	Document contains few errors in formatting and punctuation. Style of referencing is generally consistent with chosen style guide. Writing style is coherent(3)	Document is professionally presented with virtually no errors in punctuation and is in the correct format. The style of referencing is consistent with chosen style guide. Writing style is clear and engaging(4)

Marks**:**

Chapter 1

Introduction

We are living in an age where tasks and systems are fusing together with the power of IOT to have a more efficient system of working and to execute jobs quickly! With all the power at our finger tips this is what we have come up with.The Internet of Things (IoT) shall be able to incorporate transparently and seamlessly a large number of different systems, while providing data for millions of people to use and capitalize. Building a general architecture for the IoT is hence a very complex task, mainly because of the extremely large variety of devices, link layer technologies, and services that may be involved in such a system. One of the main concerns with our environment has been solid waste management which impacts the health and environment of our society. The detection, monitoring and management of wastes is one of the primary problems of the present era. The traditional way of manually monitoring the wastes in waste bins is a cumbersome process and utilizes more human effort, time and cost which can easily be avoided with our present technologies. This is our solution, a method in which waste management is automated. This is our IoT Garbage Monitoring system, an innovative way that will help to keep the cities clean and healthy.

The Applications of IoT are :-

Wearables:-

Wearable technology is a hallmark of IoT applications and probably is one of the earliest industries to have deployed the IoT at its service. We happen to see Fit Bits,heart rate monitors and smartwatches everywhere these days [5]

Health Care

IoT applications can turn reactive medical-based systems into proactive wellness-based systems.The resources that current medical research uses, lack critical real-world information. It mostly uses leftover data, controlled environments, and volunteers for medical examination. IoT opens ways to a sea of valuable data through analysis, real-time field data, and testing.

Agriculture

Statistics estimate the ever-growing world population to reach nearly 10 billion by the year 2050. To feed such a massive population one needs to marry agriculture to technology and obtain best results. There are numerous possibilities in this field. One of them is the **Smart Greenhouse**.

A greenhouse farming technique enhances the yield of crops by *controlling environmental parameters*. However, manual handling results in production loss, energy loss, and labor cost, making the process less effective.

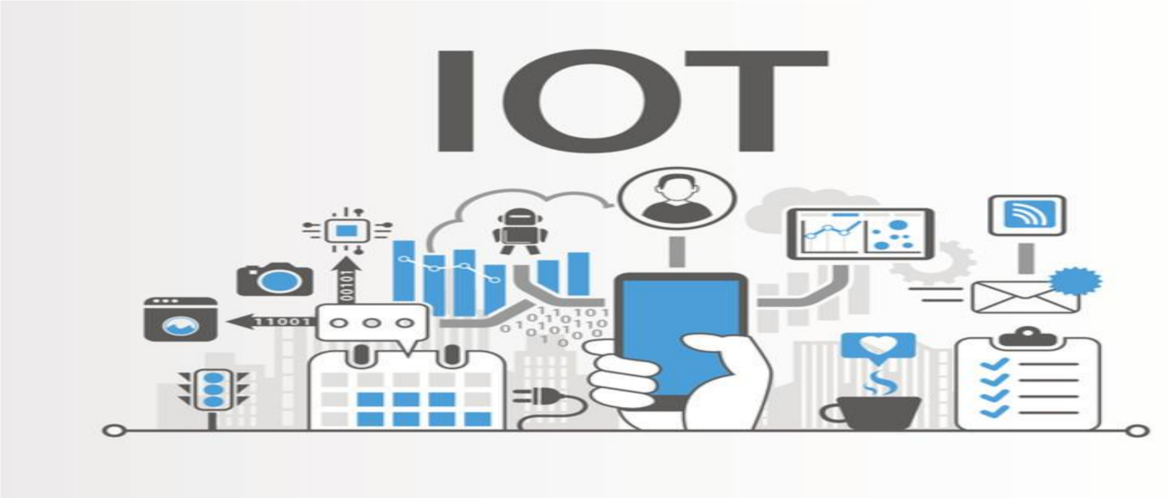
A greenhouse with embedded devices not only makes it easier to be monitored but also, enables us to control the climate inside it. Sensors measure different parameters according to the plant requirement and send it to the cloud. It, then, processes the data and applies a control action.

Industrial Internet

Industrial Internet is the new buzz in the industrial sector, also termed as Industrial Internet of Things (IIoT). It is empowering industrial engineering with sensors, software and big data analytics to create brilliant machines.

According to Jeff Immelt, CEO, GE Electric, IIoT is a “beautiful, desirable and investable” asset. The driving philosophy behind IIoT is that, smart machines are more accurate and consistent than humans in communicating through data. And, this data can help companies pick inefficiencies and problems sooner.

IIoT holds great potential for quality control and sustainability. Applications for tracking goods, real time information exchange about inventory among suppliers and retailers and automated delivery will increase the supply chain efficiency. According to GE the improvement industry productivity will generate $10 trillion to $15 trillion in GDP worldwide over next 15 years.



Chapter2

Problem Definition

The smart dustbin uses Arduino Mega and 2 ultrasonic sensors (one ultrasonic sensor for the detection of the object near the dustbin and the other one for monitoring the garbage levels in the dustbin). The lid of the dustbin opens when the object is at a certain distance from the sensor, Servo motor is used to lift the lid of the dustbin. After the dustbin is filled at a certain level, the lid of the dustbin opens automatically . Data for garbage levels is store at Cloud via Thingspeak. Also when the garbage in the bin reaches a certain level the bulb lights up.

Chapter3

Literature Survey

In paper[1], quantitative analysis between existing dustbins and their serving population. The study first analyses the spatial distribution of dustbins in some areas of Dhaka city using average nearest neighbor functions of GIS. Remarkably, the spatial circulation of the current dustbins has appeared to be dominatingly in clustered pattern. Next, an optimal number of additional dustbins were calculated. It is shown that the number of existing dustbins is insufficient in the study area. The extent of pollution caused by the existing dustbins was calculated using spatial analyst functions of GIS. It is found that all the dustbins are burnt with wastes and causing pollution to the environment. [6]

In paper[2],the smart bins with ultrasonic sensors which measure the level of dustbin being filled up. The container is divided into three levels of garbage being collected in it. Every time the garbage crosses a level the sensors receives the data of the filled level. This data is further sent to the garbage analyzer as instant message using GSM module. Placing three ultrasonic sensors at three different levels of the container may be a disadvantage as the cost of the dustbin increases due to the sensors and also the sensors can be damaged due to the rough action by the users. An IoT-based smart garbage system(SGS) is proposed to reduce the amount of food waste.

In paper[3], battery-based smart garbage bins (SGBs) exchange information with each other using wireless mesh networks, and a router and server collect and analyze the information for service provisioning. Furthermore, the SGS includes various IoT skills considering user convenience and increases the battery lifetime through two types of energy-efficient operations of the SGBs: stand-alone operation and

cooperation based operation. The proposed SGS had been functioned as a pilot project in Gangnam district, Seoul, Republic of Korea, for a one-year period. The test demonstrated that the normal measure of food waste could be decreased by 33%.

In paper[4] , framework in which a Camera will be set at each garbage collection point alongside load cell sensor at base of the trash can. The camera will take continuous snapshots of the garbage can. A threshold level is set which compares the output of camera and load sensor. The comparison is done with help of microcontroller. After analyzing the image an idea about level of garbage in the can and from the load cell sensor, weight of garbage can be known. Accordingly, information is processed that is controller checks if the threshold level is exceeded or not. This is convenient to use but economically not reliable.

Chapter 4

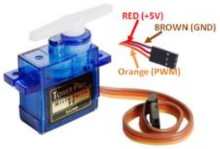
Hardware and Software Components used in Project

Hardware Components:-

- Breadboard :- A breadboard is a solderless device for temporary prototype with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate. The breadboard has strips of metal underneath the board and connect the holes on the top of the board. The metal strips are laid out as shown below. Note that the top and bottom rows of holes are connected horizontally and split in the middle while the remaining holes are connected vertically.

- Ultrasonic Sensors:- An Ultrasonic sensor is a device that can measure the distance to an object by using sound waves. It measures distance by sending out a sound wave at a specific frequency and listening for that sound wave to bounce back. By recording the elapsed time between the sound wave being generated and the sound wave bouncing back, it is possible to calculate the distance between the sonar sensor and the object.

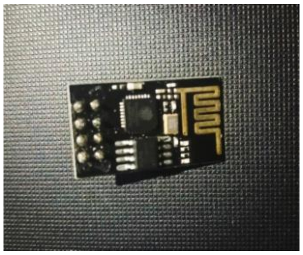
- Servo Motor :- A **servo motor** is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use servo motor. It is just made up of simple motor which run through **servo mechanism**. If motor is used is DC powered then it is called DC servo motor, and if it is AC powered motor then it is called AC servo motor. We can get a very high torque servo motor in a small and light weight packages.



- Jumper Wires:- Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed. Fairly simple. In fact, it doesn’t get much more basic than jumper wires.
- 5V Battery:-

Software Components:-

- Arduino : Arduino is an open source,computer hardware and software company ,project ,and user community that designs and manufactures microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world.
  Arduino board designs use a variety of microcontrollers and controllers. The boards are equipped with sets of digital and analog (I/O) pins that may be interfaced to various expansion boards or breadboards (*shields*) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers. The microcontrollers can be programmed using C and C++ programming languages. In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environment (IDE) based on the Processing language project.
  .in this project we have used Arduino Mega .
- Wifi-Module:



The ESP8266 WiFi Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor. This module comes with AT commands firmware which allows you to get functionality like arduino wifi, however, you can load different firmwares to make your own application on the modules' memory and processor. Its a very economic module and has a huge and growing community support.

This module has onboard 80Mhz low power 32-bit processor which can be used for custom firmware. This also means that you can host small webpages without any external controller. The ESP8266 supports APSD for VoIP applications and Bluetooth co-existence interfaces, it contains a self-calibrated RF allowing it to work under all operating conditions and requires no external RF parts.

Hardware Components	Software Components
	Arduino Mega
Ultrasonic Sensor ()	Wifi-module
Servo Motor ()	
Breadboard ()	
5V Battery()	
LED Light ()	

Cost Of Components:

Components	Cost
Arduino Mega	750 Rs
Ultrasonic Sensor x 2	200 Rs
Servo Motor	250 Rs
Breadboard	350 Rs
Jumper Wires	180 Rs
Dustbin	100 Rs
Battery	50 Rs
LED light	5 Rs
Total	**2485 Rs**

Project Implementation

The interfacing diagram, explaination and connection

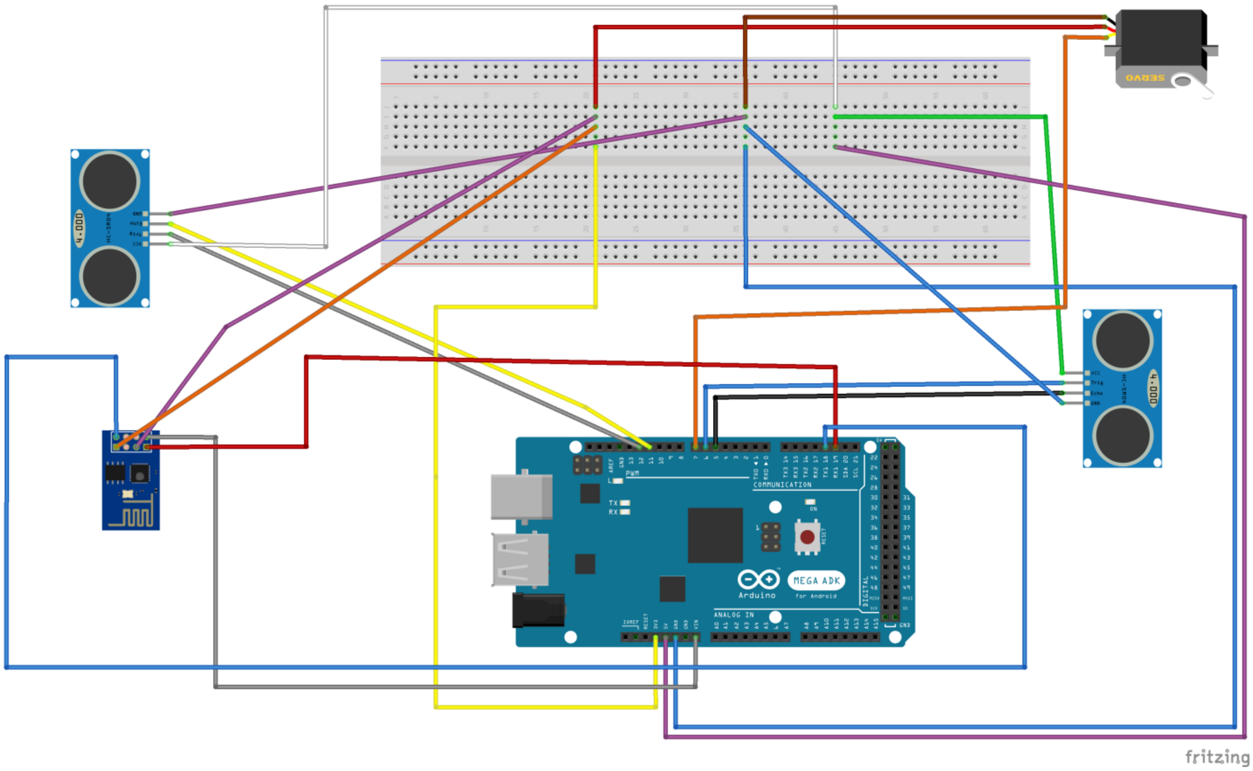


Fig.(5.1) Circuit Diagram

Chapter 6

Testing & Debugging

Tested Components:

1 .Ultrasonic Sensor:

For testing the Ultrasonic Sensor ,the Sensor was interfaced with Arduino Board and the motor and the program ran successfully .the sensor would sense garbage approaching and it would make the motor move.

2 .Ultrasonic Sensor for Monitoring Distance:

Ultrasonic sensor for displaying Level(distance) interfaced with Arduino and was connected via breadboard first the program ran unsuccessfully but later it was successfully implemented. This Sensor is placed at the top of the Dustbin It detects the distance or the level to which Garbage has been filled.

3 .Arduino:

Initially we used Arduino UNO for implementation ,It proved successful for implementing Servo motor but it failed when we started to implement the program for monitoring distance .So later we tried. Arduino MEGA and we got the results ,both the Ultrasonic Sensor ran successfully.

4 .WiFi Module :

The Module was tested by trying to send data coming from the already interfaced Ultrasonic Sensor for monitoring distance to the Thingspeak server and after many trials and errors in the WiFi code the module ran perfectly and the send the data to the Thingspeak Cloud .Initially due to failure of WiFi module we used the Blynk android app due to easy UI and instructions provided in the app. But later after all the issues were sorted we were able to send the on the Thingspeak Cloud successfully.

5. Breadboard:

We used Breadboard because we couldn’t connect both sensors to Arduino .Also connections were made more easier using the Breadboard.

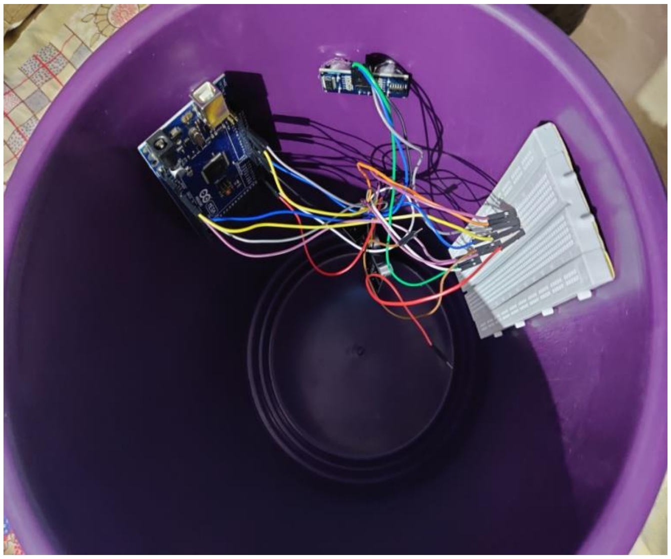
6.Servo Motor:

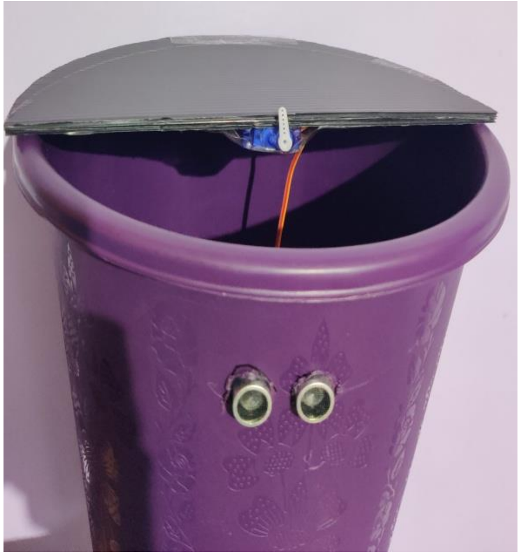
We used Servo Motor for open the lib of the dustbin.We initially faced problems for connecting the motor to the arduino due to the short length of the wire with help of jumper wires ie: female to male to increase the length of wire and we successfully connected motor to the arduino.

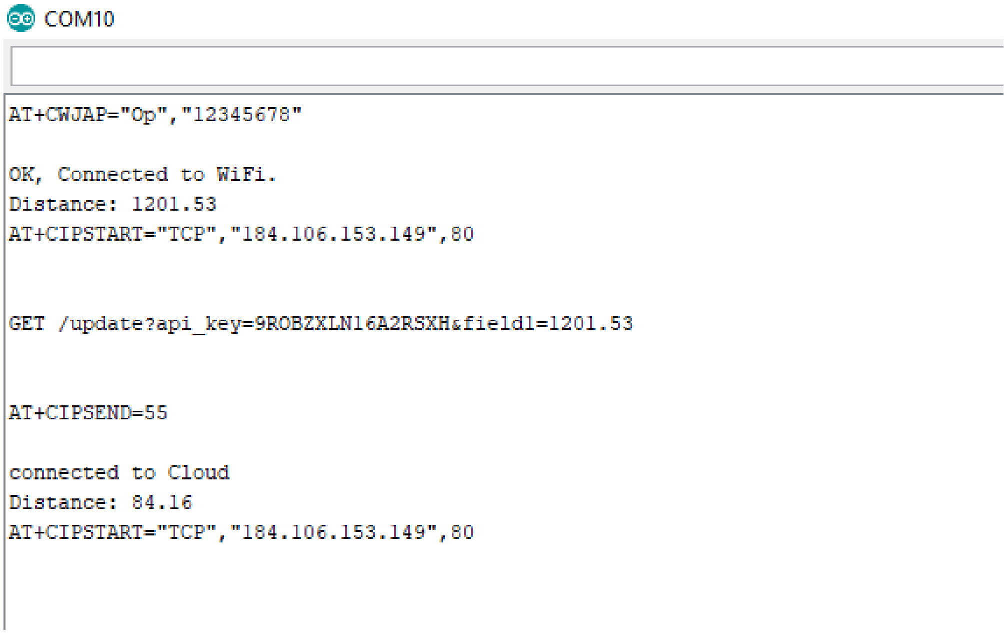
7. LED Light : We used LED Light for notification ,when dustbin is full and indicates as notification light.

Chapter 7

Results

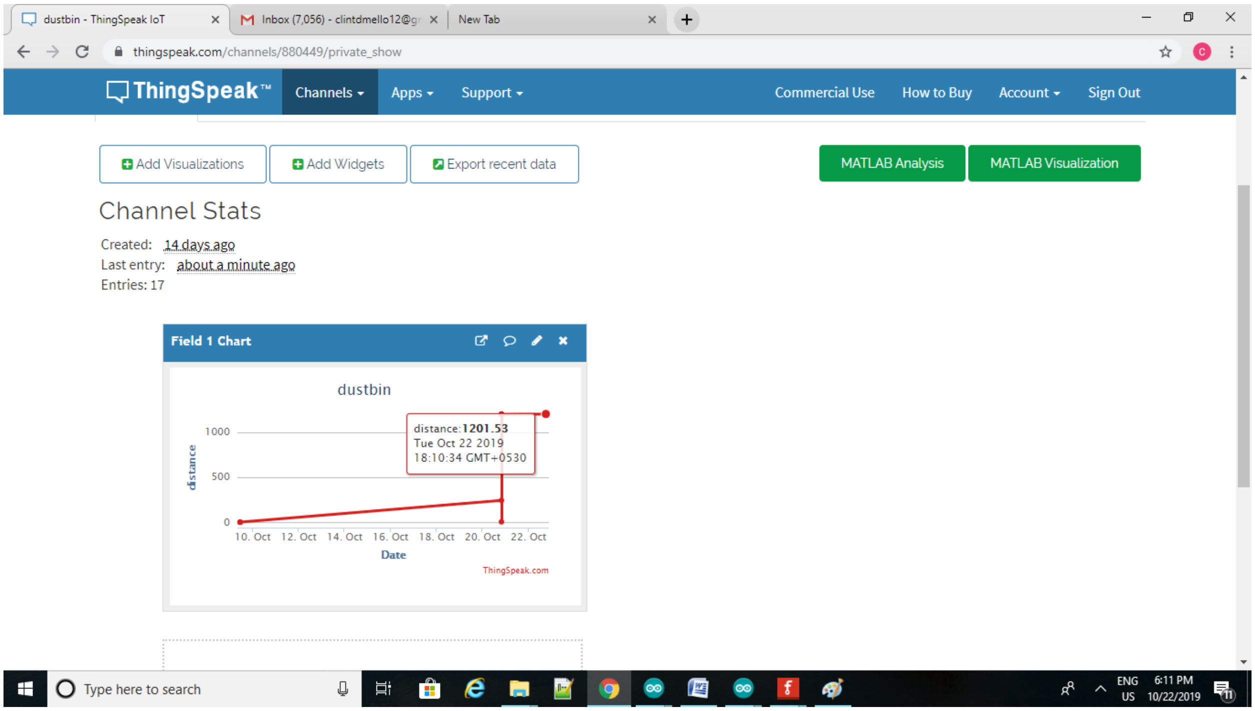






```
AT+CWJAP="Op","12345678"

OK, Connected to WiFi.
Distance: 1201.53
AT+CIPSTART="TCP","184.106.153.149",80


GET /update?api_key=9ROBZXLN16A2RSXH&field1=1201.53


AT+CIPSEND=55

connected to Cloud
Distance: 84.16
AT+CIPSTART="TCP","184.106.153.149",80
```

Fig.(7.2) Readings connected to Cloud showing Distance



Channel Stats

Created: 14 days ago
Last entry: about a minute ago
Entries: 17

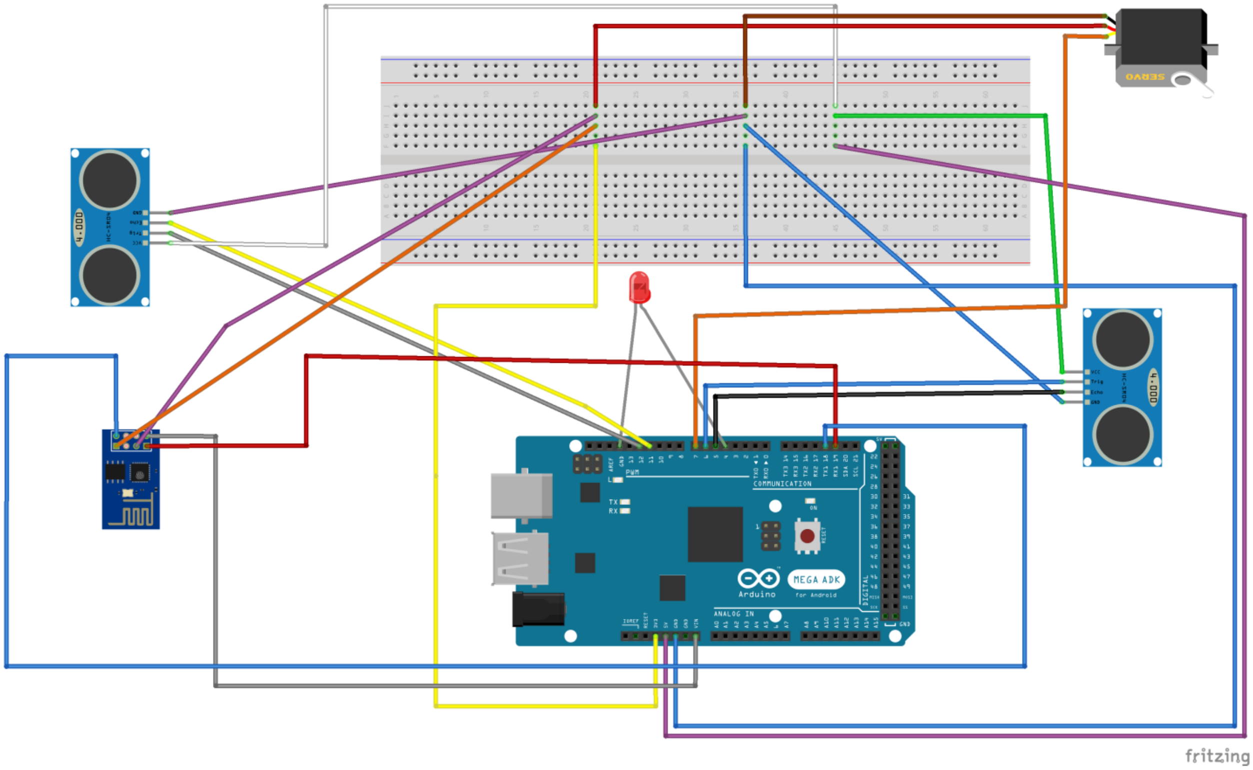
Fig.(7.3) Graph for Level of Garbage

Chapter 8.

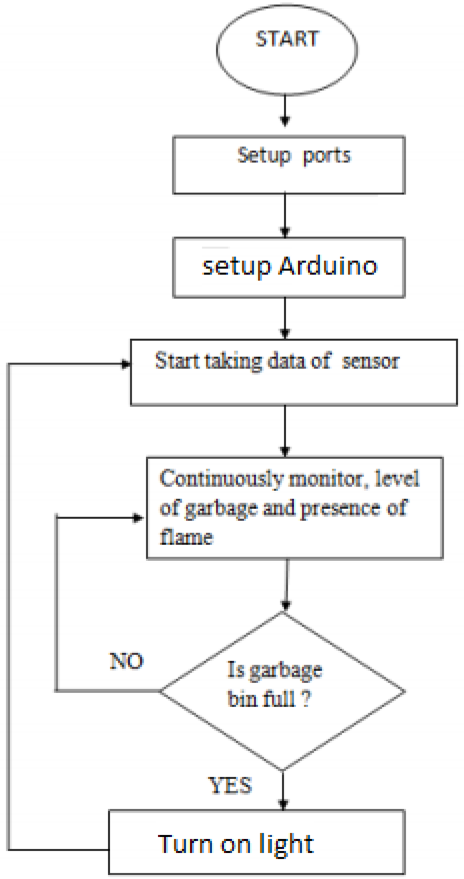
Proposed modification

Until now the project we used ,it is able able to open the lid as ultrasonic sensor detects the person approaching towards the bin with garbage in his hand and when the garbage is thrown in the bin the lid is closed after 3 seconds . And it would send the data on cloud to the user as how much the dusbin is filled with the garbage .

Now we have attached a red led bulb to the dustbin so that when dustbin is full or when the garbage reaches certain level of distance the red led bulb will glow ,so that any person nearby would know that dustbin is completely filled with the garbage



Upgraded Diagram of the circuit


START
Setup ports
setup Arduino
Start taking data of sensor
Continuously monitor, level of garbage and presence of flame
Is garbage bin full ?
NO
YES
Turn on light

FLOWCHART

Chapter 9

Futurework

Automatic Garbage Fill Alerting system helps us to reduce the pollution. Many times garbage dustbin is overflow and many animals like dog or cow enters inside or near the dustbin. Also some birds are also trying to take out garbage from dustbin. This project can avoid such situations. And the message can be sent directly to the cleaning vehicle instead of the contractor's office. Apart from this, differentiation can be made between dry trash bin and wet trash bin collecting plastic dry waste and biodegradable waste respectively. To implement this methane and smell sensors can be used. This helps in distinguishing the waste at the source and hence reducing the requirement of manpower. To enhance it further, an automated system can be developed which is able to pick up waste in and around the bin, segregate them and put them in respective bins.

Smart dustbin helps us to reduce the pollution. Many times garbage dustbin overflows and many animals like dog or rat enter inside or near the dustbin. This creates a bad scene. Also some birds are also trying to take out garbage from dustbin. This project can avoid such situations. And the message can be sent directly to the cleaning vehicle instead of the contractor's office.

Chapter 10

Reference

[1]. Alexey Medvedev, Petr Fedchenkov, ArkadyZaslavsky, Theodoros,

Anagnostopoulos Sergey Khoruzhnikov, “Waste Management as an IoT- Enable Service in Smart Cities.” –International journal of Engineering, June 27 2015.

[2]. Meghana K C, Dr. K R Nataraj, “IOT Based Intelligent Bin for Smart Cities.” –International journal on Research and Innovation trends in Computing and Communication,13 Feb 2017

[3].KasliwalManasi H., SuryawanshiSmitkumar B, “A Novel Approach to Garbage Management Using Internet of Things for Smart Cities”.-International Journal of Trends in Scientific Research and Devlopment ,11 Nov 2017

[4].Vishesh Kumar Kurrel, “Smart Garbage Collection Bin Overflows Indicator using Internet of Things.”- International Journal of Innovative Research in Computer and Communication Engineering 5 May 2016)

[5]. [https://www.analyticsvidhya.com]

[6]. http://ijesc.org

Appendix

```
const int trigPin = 12;
const int echoPin = 11;
int servoPin = 7;
const int TrigPin = 6;
const int EchoPin = 5;
float initialdistance = 40; // insert initial distance or distance between sensor and empty base tank
float surfacearea = 10; // enter value of surface area of the rectangular tank base by multiplying width and length
long duration1, dist, average;
long aver[3];   //array for average
float duration, distance, heightofwater;

String apiKey = "9ROBZXLN16A2RSXH";  //API key

void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  servo.attach(servoPin);
  pinMode(TrigPin, OUTPUT);
pinMode(EchoPin, INPUT);
  Serial.begin(9600);
  Serial1.begin(115200);   // Arduino to ESP Communication
  servo.write(0);         //close cap on power on
  delay(100);
  servo.detach();
  connectWiFi();           // To connect to Wifi
}

void measure()
{



digitalWrite(10,HIGH);
digitalWrite(TrigPin, LOW);
delayMicroseconds(5);
digitalWrite(TrigPin, HIGH);
delayMicroseconds(15);
digitalWrite(TrigPin, LOW);
pinMode(EchoPin, INPUT);
duration = pulseIn(EchoPin, HIGH);
dist = (duration1/2) / 29.1;    //obtain distance
}
void loop() {
  for (int i=0;i<=2;i++) {   //average distance
    measure();
   aver[i]=dist;
    delay(10);              //delay between measurements
  }
```

```
 dist=(aver[0]+aver[1]+aver[2])/3;
if ( dist<50 ) {
//Change distance as per your need
 servo.attach(servoPin);
  delay(1);
 servo.write(0);
 delay(3000);
 servo.write(150);
 delay(1000);
 servo.detach();
}



digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);
  distance = (duration*.0343)/2;
  //heightofwater = initialdistance-distance;
  //volume = heightofwater*surfacearea;
  Serial.print("Distance: ");

  Serial.println(distance);

  delay(1000);

  //Serial.print("Volume: ");

//Serial.println(volume);

  //delay(1000);




  Serial1.println("AT+CIPMUX=0\r\n");      // To Set MUX = 0

  delay(2000);                              // Wait for 2 sec


If(distance < 10)

 {

digitalWrite(13, HIGH); // sets the digital pin 13 on delay(1000); // waits for a second
digitalWrite(13, LOW); // sets the digital pin 13 off delay(1000); // waits for a second

}


  // TCP connection

  String cmd = "AT+CIPSTART=\"TCP\",\"";   // TCP connection with thingspeak server

  cmd += "184.106.153.149";                // IP addr of api.thingspeak.com

  cmd += "\",80\r\n\r\n";                  // Port No. = 80


  Serial1.println(cmd);                    // Display above Command

  Serial.println(cmd);                     // Send above command to Rx1, Tx1
```

```
delay(20000);                              // Wait for 20 Sec


if(Serial1.find("ERROR"))                  // If returns error in TCP connection
{
  Serial.println("AT+CIPSTART error");   // Display error msg
  //return;
}
// prepare GET string
String getStr = "GET /update?api_key=";
getStr += apiKey;
getStr +="&field1=";
getStr += distance;
getStr += "\r\n\r\n";


Serial.println(getStr);                  // Display GET String


cmd = "AT+CIPSEND=";                     // send data length
cmd += String(getStr.length());
cmd+="\r\n";


Serial.println(cmd);                     // Display Data length
Serial1.println(cmd);                    // Send Data length command to Tx1, Rx1
delay(20000);                            // wait for 20sec


if(Serial1.find(">"))                    // If prompt opens verify connection with cloud
```

```
  {
    Serial.println("connected to Cloud");  // Display confirmation msg
    Serial1.print(getStr);                 // Send GET String to Rx1, Tx1
  }
  else
  {
    Serial1.println("AT+CIPCLOSE\r\n");    // Send Close Connection command to Rx1, Tx1
    Serial.println("AT+CIPCLOSE");         // Display Connection closed
  }

delay(16000);                            // wait for 16sec



}
boolean connectWiFi() {               // Connect to Wifi Function

  Serial1.println("AT+CWMODE=1\r\n"); // Setting Mode = 1
  delay(100);                         // wait for 100 mSec

  String cmd = "AT+CWJAP=\"";         // Connect to WiFi
  cmd += "Op";                        // ssid_name
  cmd += "\",\"";
  cmd += "12345678";                  // password
  cmd += "\"\r\n";
Serial.println(cmd);                  // Display Connect Wifi
```

```
  Serial1.println(cmd);               // send Connect WiFi command to Rx1, Tx1


  delay(10000);                       // wait for 10 sec


  Serial1.println("AT+CWJAP?");       // Verify Connected WiFi


  if(Serial1.find("+CWJAP"))
  {
    Serial.println("OK, Connected to WiFi.");         // Display Confirmation msg ......first time
    return true;
  }
  else
 {
    Serial.println("Can not connect to the WiFi.");   // Display Error msg
    return false;
  }
 }
```

//FOR SERVO MOTOR//

```
#include <Servo.h>   //servo library
Servo servo;
int trigPin = 5;
int echoPin = 6;
int servoPin = 7;
int led= 10;
long duration, dist, average;
long aver[3];   //array for average

void setup() {
    Serial.begin(9600);
    servo.attach(servoPin);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    servo.write(0);         //close cap on power on
    delay(100);
    servo.detach();
}

void measure() {
 digitalWrite(10,HIGH);
digitalWrite(trigPin, LOW);
delayMicroseconds(5);
digitalWrite(trigPin, HIGH);
delayMicroseconds(15);
digitalWrite(trigPin, LOW);
pinMode(echoPin, INPUT);
duration = pulseIn(echoPin, HIGH);
dist = (duration/2) / 29.1;    //obtain distance
}
void loop() {
  for (int i=0;i<=2;i++) {   //average distance
    measure();
aver[i]=dist;
    delay(10);              //delay between measurements
  }
 dist=(aver[0]+aver[1]+aver[2])/3;

if ( dist<50 ) {
//Change distance as per your need
 servo.attach(servoPin);
```

```
  delay(1);
 servo.write(0);
 delay(3000);
 servo.write(150);
 delay(1000);
 servo.detach();
}
Serial.print(dist);
}
```