Smart Waste Management System using ARDUINO

GITHUB LINK: https://github.com/rex223/IOT_project

Md Rehan Farooque[1]
Department of AIML Engineering
Netaji Subhash Engineering College, Kolkata,India

Raquib[2]
Department of AIML Engineering
Netaji Subhash Engineering College, Kolkata,India

Anuska Saha[3]
Department of AIML Engineering
Netaji Subhash Engineering College, Kolkata,India

[4]Abhishreya Das
Department of AIML Engineering
Netaji Subhash Engineering College, Kolkata,India

Anwesha Guha[5]
Department of AIML Engineering
Netaji Subhash Engineering College, Kolkata,India

[6]Shreyan Ghosh
Department of AIML Engineering
Netaji Subhash Engineering College, Kolkata,India

***Abstract*— A system is introduced to manage waste in big cities effectively without having to monitor the parts 24x7 manually. Here the problem of unorganized and non-systematic waste collection is solved by designing an embedded IoT system which will monitor each dumpster individually for the amount of waste deposited. Here an automated system is provided for segregating wet and dry waste. A mechanical setup can be used for separating wet and dry waste into separate containers here sensors can be used for separating wet and dry.For detecting the presence of any waste wet or dry can be detected using Moisture sensor. In this process, Servo motor is used to Segregate the waste. Both these containers are embedded with ultrasonic sensors at the top, the ultrasonic sensor is used for measuring height of waste. This makes it possible to measure the amount of waste in the containers if one of the containers is full then alert message will be sent to the corresponding personal.**

I. INTRODUCTION

Today big cities around the world are facing a common problem, managing the city waste effectively without making city unclean. Today's waste management systems involve a large number of employees being appointed to attend a certain number of dumpsters this is done every day periodically. This leads to a very inefficient and unclean system in which some dumpsters will be overflowing some dumpsters might not be full. This is caused by variation in population in the density in the city or some other random factor this makes it impossible to determine which part needs immediate attention. Here a waste management system is introduced in which each dumpster is embedded in a monitoring system which will notify the corresponding personal if the dumpster is full. In this system, it is also possible to separate wet and dry waste for both beginners and experienced electronics enthusiasts, into two separate containers. This system provides an effective empowering them to bring their creative ideas to fruition with solution to waste management problem confidence and ease.

II. EXISTING SYSTEM

- Manual systems in which employees clear the dumpsters periodically.
- No systematic approach towards clearing the dumpsters.
- Unclear about the status of a particular location
- Employees are unaware of the need for a particular location
- Very less effective in cleaning city

Module Description

1.

1.1. List of Modules

1.1.1. Processor
1.1.2. Sensors
1.1.3. Softwares
1.1.4. DC Motor

1.1.1 Processor: *1.1.1.1. ARDUINO UNO:* The Arduino Uno is a cornerstone of the Arduino platform, beloved for its simplicity, versatility, and accessibility in the world of electronics prototyping and DIY projects. At its heart lies the Atmega328P microcontroller chip, which serves as the brain of the board, executing programmed instructions and controlling connected hardware. he Arduino Uno's blend of affordability, and flexibility makes it an ideal choice for both beginners and experienced electronics enthusiasts, into two separate containers. This system provides an effective solution to waste management problem and ease.

IOREF: This stick on the Arduino/Genuino board gives the voltage reference with that the microcontroller works. Associate in Nursing suitably organized defend will examine the IOREF pin voltage and choose the fitting power source or empower voltage interpreters on the outputs to figure with the 5V or 3.3V.

VIN: the input voltage to the Arduino / Genuino board once it's utilizing an external power supply (instead of five volts from the USB association or alternative managed power source). We will be able to provide voltage through this pin, or, if providing voltage by means that of the power jack, get to that through this pin

LED: there's a worked in crystal rectifier driven by advanced 13 pin. At the purpose once the pin is high in terms of value, the crystal rectifier is on, once the pin is low, it's off.

5V: This pin outputs a managed 5V from the controller on the board. The board is given power either from the DC power jack (7 - 20V), the USB connecter (5V), or the VIN stick of the board (7-20V). Providing voltage by means that of the 5V or 3.3V pins sidesteps the controller, and might damage the board.

Reset: generally wont to add a reset catch to shields that sq. the one on the board

GND: This is a ground pin

3V3: 50mA of maximum current is drawn and on- board regulator generates 3.3 volts of supply

Fig 1.1.1.1-Module Diagram for Arduino Uno Processor

1.1.2. Sensors:

1.1.2.1. Ultrasonic Sensor:-

Ultrasonic (US) sensor is a 4-pin module, whose pin names are Vcc, Trigger, Echo and Ground respectively. This detector could be a very-about detector utilized This detector could be utilized in several applications wherever mensuration distance or sensing objects are needed. The module has 2 eyes like accompanies like robot at the front that frames the ultra-supersonic transmitter and recipient. The locator works with the simple secondary school recipe that

Distance = Speed × Time

The Ultrasonic transmitter broadcast a supersonic wave, this wave goes in air and when it gets questioned by any material it gets reflected back toward the sensor this reflected wave is seen by the Ultrasonic beneficiary module as appeared in the image beneath Now, to figure the separation utilizing the above recipe, we should know Speed and time. Since we tend to utilize the supersonic wave we as a whole know all inclusive speed of wave at region conditions that is 330m/s. The hardware inbuilt on the module will compute the time taken for the US wave to return and turns on the reverberation stick high for that equivalent specific measure of your time, along these lines we can likewise realize the time taken. Presently just figure the separation utilizing a microcontroller or small scale chip. Likewise, this nondeterministic mapping case (i.e., one-to-many mapping) happens even after we normalize all parameter values to extract the structures of the web requests and queries. Since the mapping can appear differently in different cases, it becomes difficult to identify all of the one-to-many mapping patterns for each web request. Moreover, when different operations occasionally overlap at their possible query set, it becomes even harder for us to extract the one-to-many mapping for each operation by comparing matched requests and queries across the sessions.

Fig 1.1.2.1 Module Diagram for Ultrasonic Sensor

1.1.2.2. IR Sensor:

An infrared sensor is an electronic device that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion. These types of sensors measures only infrared radiation rather than emitting it that is called as a passive IR sensor.Usually in the infrared spectrum, all the objects radiate some form of thermal radiations. These types of radiations are invisible to our eyes that can be detected by an infrared sensor. The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode which is sensitive to IR light of the same wavelength as that emitted by the IR LED. When IR light falls on the photodiode, the resistances and these output voltages, change in proportion to the magnitude of the IR light received.

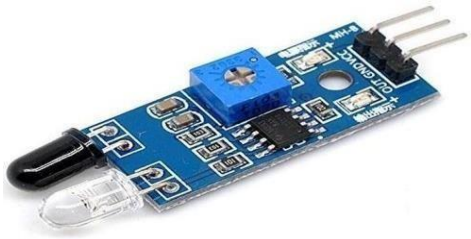


Fig 1.1.2.2 Module Diagram for IR Sensor

1.1.2.3. Moisture Sensor:

Moisture sensors measure the volumetric water content in soil. Since the direct gravimetric measurement of free soil moisture requires removing, drying, and weighing of a sample, soil moisture sensors measure the volumetric water content indirectly by using some other property of the soil, such as electrical resistance, dielectric constant, or interaction with neutrons, as a proxy for the moisture content.

The relation between the measured property and soil moisture must be calibrated and may vary depending on environmental factors such as soil type, temperature, or electric conductivity. Reflected microwave radiation is affected by the soil moisture and is used for remote sensing in hydrology and agriculture. Portable probe instruments can be used by farmers or gardeners.

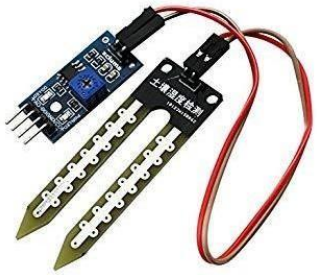


Fig 1.1.2.3 Module Diagram for Moisture Sensor

1.1.3. Softwares:

1.1.3.1. Arduino IDE

The Arduino integrated development environment (IDE) is a cross-stage application (for Windows, macOS, Linux) that is written in the programming language Java. It is utilized to compose and transfer programs to Arduino compatible boards, yet in addition, with the assistance of outsider centres, other seller advancement sheets.

The source code for the IDE is discharged under the GNU General Public License. The Arduino IDE underpins the dialects C and C++ utilizing uncommon guidelines of code organizing. The Arduino IDE supplies a product library from the Wiring venture, which gives numerous normal information and yield methodology. Client composed code just requires two essential capacities, for beginning the sketch and the principle program circle, that are aggregated and connected with a program stub fundamental () into an executable cyclic official program with the GNU toolchain, additionally included with the IDE distribution. The Arduino IDE utilizes the program avrdude to change over the executable code into a book record in hexadecimal encoding that is stacked into the Arduino board by a loader program in the board's firmware

The primary code, otherwise called a sketch, made on the IDE platform will eventually produce a Hex File which is then moved and transferred in the controller on the board.

The IDE condition for the most part contains two essential parts: Editor and Compiler where previous is utilized for composing the required code and later is utilized for assembling and transferring the code into the given Arduino Module.

This environment supports both C and C++

Fig 1.1.3.1 Module Screenshot for Arduino IDE

1.1.3.2. ThingSpeak

ThingSpeak is an IoT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak. With the ability to execute MATLAB code in ThingSpeak you can perform online analysis and processing of the data as it comes in. ThingSpeak is often used for prototyping and proof of concept IoT systems that require analytics.

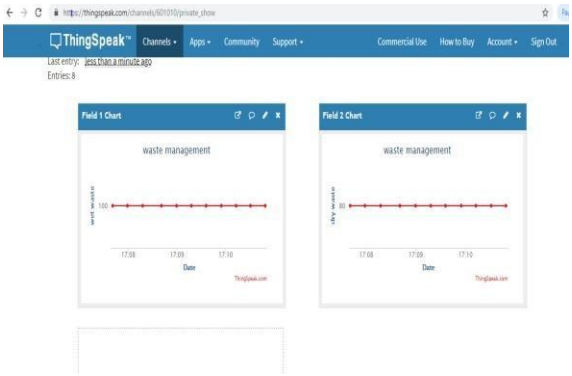


Fig 1.1.3.2 Module Screenshot for BLYNK App

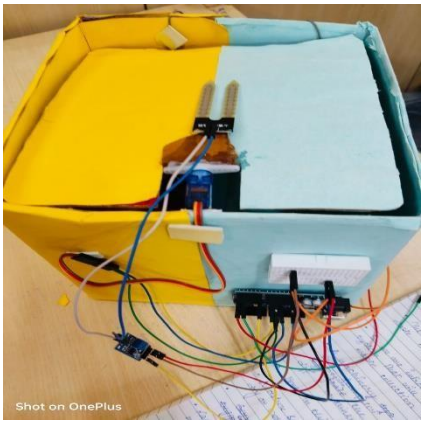
1.1.4. Servo Sg90 Motor:

The SG90 servo motor is a compact and lightweight device widely utilized in hobbyist electronics and robotics projects. Operating typically between 4.8V to 6V, it offers a balance of torque and speed suitable for controlling various lightweight mechanisms. With torque outputs ranging from 1.5 kg/cm at 4.8V to 1.8 kg/cm at 6V, it's capable of powering small robot

arms, RC vehicles, or other moving parts in DIY endeavors. Controlled via pulse-width modulation (PWM) signals, it responds to frequencies around 50Hz, with pulse widths varying between 1ms to 2ms to achieve different angular positions within its range. Its construction, comprising a DC motor, gear set, potentiometer, and control circuit, ensures precise Overall, the SG90 servo motor's compact size, moderate torque, and ease of use make it an excellent choice for hobbyists, educators, and makers looking to add motion control to their projects. Its affordability and availability also contribute to its widespread popularity in the DIY electronics community.

positioning. Compatible with popular microcontroller platforms like Arduino and Raspberry Pi, the SG90 servo motor stands out for its simplicity, versatility, and affordability, making it a staple component in the arsenal of electronics enthusiasts and makers worldwide.

Fig 1.1.4. Module Diagram for DC Motor



Prototype of the Waste Management system

2. Circuit diagram

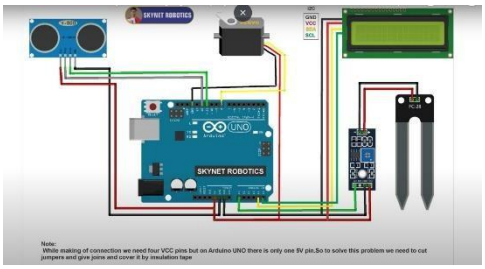


Fig 3.3-Circuit Diagram for Smart Waste Management System

Here, the figure represents an integration of Smart Waste Management System with a 3-tier sensor processor device system.

- Ultrasonic sensor measure distances by using ultrasonic waves. The sensor emits an ultrasonic wave and receives the reflected wave back from the target.
- IR Sensor emits in order to sense some aspects of the surroundings.
- Moisture Sensor measures the volumetric water content in the soil. Reflected microwave radiation is affected by the soil moisture and is used for remote sensing hydrology and agriculture.

- Servo motor which is connected to the digital pins of Arduino

- We are using serial monitor for the display.

Program code

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>

LiquidCrystal_I2C lcd(0x27, 16, 2); // Address for the LCD display
Servo servo; // Servo motor object

const int trigPin = 12; // Ultrasonic sensor trigger pin
const int echoPin = 11; // Ultrasonic sensor echo pin
const int soilMoisturePin = A0; // Soil moisture sensor analog pin

bool isWasteDetected = false; // Flag to track waste detection
bool isServoStable = true; // Flag to track servo stability
bool isFirstDetection = true; // Flag to track first waste detection
int wetWasteThreshold = 1000; // Threshold for wet waste detection
int lastMoistureLevel = 0; // Variable to store the last moisture level
int maxWasteHeight = 20; // Maximum waste height in centimeters

void setup() {
  Serial.begin(9600);
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Waste Segregator");
  delay(2000);
  lcd.clear();

  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);

  servo.attach(9);
  servo.write(90); // Set servo to initial position
}

void loop() {
  // Run waste detection only once
  if (isFirstDetection) {
    detectWaste();
    isFirstDetection = false;
  }

  // Continuous monitoring for changes
  monitorChanges();
}

void detectWaste() {
  lcd.setCursor(0, 0);
  lcd.print("Measuring...");

  int distance = measureDistance();
  int moistureLevel = measureMoisture();

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Distance: ");
  lcd.print(distance);
  lcd.print(" cm");

  lcd.setCursor(0, 1);
  lcd.print("Moisture: ");
  lcd.print(moistureLevel);
  // Check for initial waste detection
  if (!isWasteDetected) {
    if (moistureLevel > wetWasteThreshold) {
      lcd.clear();
      lcd.print("Dry Waste Detected");
segregateWaste(true); // Rotate servo clockwise for dry waste
      isWasteDetected = true;
    } else {
      lcd.clear();
      lcd.print("Wet Waste Detected");
      segregateWaste(false); // Rotate servo anticlockwise for wet waste
      isWasteDetected = true;
    }
  }
  // Check for waste height
  if (distance < maxWasteHeight) {
    lcd.setCursor(0, 2);
    lcd.print("Waste Full!");
  }
```

```
else {
```

```
    lcd.setCursor(0, 2);
lcd.print("Waste Height: ");
lcd.print(distance);     lcd.print("
cm");
  }

  delay(2000); // Delay after initial detection
}

void  monitorChanges()  {      //
Continuous monitoring loop
  while (true) {
    int moistureLevel = measureMoisture();

    // Check for significant changes in moisture level
if (abs(moistureLevel - lastMoistureLevel) >= 50) {
lcd.clear();                          lcd.setCursor(0,     1);
lcd.print("Moisture: ");
      lcd.print(moistureLevel);

      if       (moistureLevel       >       wetWasteThreshold)       {
lcd.setCursor(0, 0);
       lcd.print("Dry Waste Detected");
       segregateWaste(true); // Rotate servo clockwise for dry waste
isWasteDetected = true;
      } else {
       lcd.setCursor(0, 0);
       lcd.print("Wet Waste Detected");
       segregateWaste(false); // Rotate servo anticlockwise for wet waste
isWasteDetected = true;
      }

      lastMoistureLevel = moistureLevel; // Update last moisture level
delay(2000); // Delay after detection
    }

    delay(1000); // Delay between monitoring cycles
  }
}

int          measureDistance()          {
digitalWrite(trigPin,             LOW);
delayMicroseconds(2);
digitalWrite(trigPin,   HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
  long duration = pulseIn(echoPin, HIGH);
int distance = duration * 0.034 / 2;   return
distance;
}

int measureMoisture() {
  int     moisture     =     analogRead(soilMoisturePin);
return moisture;
}

void segregateWaste(bool isDryWaste) {
  if (isDryWaste) {
    servo.write(180); // Rotate servo clockwise for dry waste
  } else {
    servo.write(0); // Rotate servo anticlockwise for wet waste
  }
  delay(2000);
  servo.write(90); // Reset servo position
}
```

IV.FUTURE SCOPE

Every project is always has scope for improvement, perhaps the most pressing issue of separation of waste is when their dispose simultaneously. The waste segregator can be improvised to include the separation of paper and plastic, safe segregation of biomedical waste generated at home, compact and aesthetic Mechanical design.

V. REFERENCES

[1] 2017 5th International Conference on Instrumentation, Control, and Automation (ICA) Yogyakarta, Indonesia, August 9-11, 2017

[2] IOT Based Smart Garbage alert system using Arduino UNO Date and journal: 2016 IEEE Region 10 Conference (TENCON) Author: Dr.N.SATHISH KUMAR, B.VIJAYALAKSHMI, R. JENIFER PRARTHANA, A .SHANKAR (S.R.M., Coimbatore, INDIA)

[3] Solid waste management based upon IoT or Smart city Date and Journal: ICICCS 2017 Author: Krishna Nirde, Prashant S. Mulay, UttamM.Chaskar (C.O.E. Pune).

[4] SMART WASTE MANAGEMENT SYSTEM Date and journal: 2015, ICCES Author: MS. NIRMALA Y BARIKER, MR. JASON VINOD D'SOUZA (MIT, Mangalore)

[5] SVASTHA: An effective solid waste management system in Android OS Date and journal: 2013 by IEEE Global humanitarian technology Author: Issac and akshai

[6] Smart bin using Arduino and other sensors Date and journal: 2013 by FCIS Author: Yusuf et al

[7] SMART DUSTBIN MANAGEMENT SYSTEM Date and journal: 7 may 2018 (IJESRT) Author: Swati Sharma and sarabjeet Singh

[8] WIRELESS DUST BIN MONITORING AND ALERT SYSTEM USING ARDUINO JOURNAL:-978-1-5090-3239-6/17/$31.00©2017IEEE P.Siva Nagendra Reddy, R.Naresh Naik, A.Amareshwar Kumar, S.NandaKishor

[9] IOT Based Smart Bin JOURNAL AND DATE:- Volume: 04 Issue: 09 | Sep -2017 ISO 9001:2008