

Automated dustbin using Arduino

Problem Statement

The automated dustbin is a sleek and modern solution designed to blend seamlessly into various environments, including homes, offices, and public spaces. The dustbin features a sensor-equipped lid mechanism that opens automatically upon detecting the presence of objects within its proximity, facilitating hands-free operation and minimising contact.

One of its key features is the integrated wet waste detection system, which utilises advanced sensors to identify and segregate wet waste from dry waste effectively. This feature not only helps in preventing odour and leakage issues but also promotes proper waste management practices.

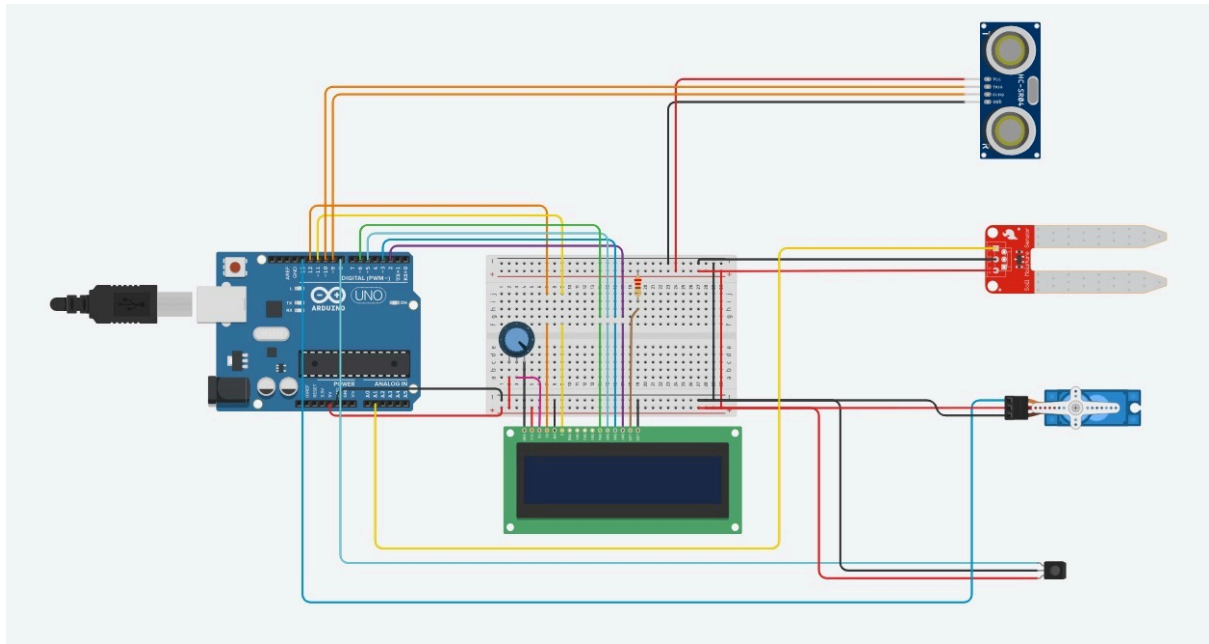
Additionally, the dustbin is equipped with sensors to detect its fill level, providing real-time information on whether it is full or empty. This functionality helps in optimising waste collection schedules, reducing unnecessary trips for waste disposal, and ensuring efficient space utilisation within the dustbin.

Overall, the dustbin represents a sophisticated yet user-friendly solution to modern waste management challenges, offering convenience, hygiene, and environmental responsibility in one compact package.

Components required

- IR obstacle sensor
- Ultrasonic sensor
- Moisture sensor
- Bluetooth module - HC 05
- Bread board
- LCD display
- Jumper wires
- Arduino Uno
- Servo Motor
- Potentiometer (for LCD)

Schematic Diagram



Disclaimer : Bluetooth module was not available in tinkercad and hence we haven't added it in the schematic diagram, The bluetooth module was connected to Digital pins (7,4)

Code

```
#include <LiquidCrystal.h>
#include <Servo.h>
#include <SoftwareSerial.h>

// Setup for servo motor
Servo myservo;
int pos = 0;

// Pins for ultrasonic sensor
int trigPin = 9;
int echoPin = 10;
long duration, cm;

// Pin for moisture sensor
const int sensor_pin = A1;

// Setup for LCD display
LiquidCrystal lcd(12, 11, 6, 5, 3, 2);
const int irSensorPin = 8;

// Setup for Bluetooth communication using Software Serial
SoftwareSerial bluetooth(7, 4); // RX, TX pins for Bluetooth module
```

```

void setup() {
  myservo.attach(13); // Attach the servo motor to pin 13
  lcd.begin(16, 2); // Start the LCD (16 characters, 2 lines)
  pinMode(irSensorPin, INPUT); // Set the IR sensor pin as input
  Serial.begin(9600); // Start serial communication at 9600 baud rate
  bluetooth.begin(9600); // Start Bluetooth communication at 9600 baud rate
  pinMode(trigPin, OUTPUT); // Set the ultrasonic trig pin as output
  pinMode(echoPin, INPUT); // Set the ultrasonic echo pin as input
}

```

```

void loop() {
  int obstacleDetected = digitalRead(irSensorPin);
  lcd.clear(); // Clear the LCD display
  lcd.setCursor(0, 0);
  lcd.print("Not Full");
  lcd.setCursor(0, 1);
  String waste = "Dry waste";
  lcd.print(waste);

  // Trigger the ultrasonic sensor
  digitalWrite(trigPin, LOW);
  delayMicroseconds(5);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Read the duration of the echo signal
  duration = pulseIn(echoPin, HIGH);
  cm = (duration / 2) / 29.1; // Calculate distance in cm

  // Read moisture level from the moisture sensor
  float moisture_percentage;
  int sensor_analog;
  sensor_analog = analogRead(sensor_pin);
  moisture_percentage = (100 - ((sensor_analog / 1023.00) * 100));

  // Check for Bluetooth commands
  if (bluetooth.available() > 0) {
    char command = bluetooth.read();
    if (command == '1') {
      myservo.write(0); // Move servo to 0 degrees
    } else if (command == '0') {
      myservo.write(90); // Move servo to 90 degrees
    }
  }
}

```

```
}  
}
```

```
// Actions when obstacle is detected
```

```
if (obstacleDetected) {  
  myservo.write(0);  
  delay(2000);  
  myservo.write(90);  
  lcd.clear();  
  lcd.setCursor(0, 0);  
  lcd.print("Not Full");  
}
```

```
// Change display if trash is full
```

```
if (cm < 10.0) {  
  lcd.clear();  
  lcd.setCursor(0, 0);  
  lcd.print("Full");  
  lcd.setCursor(0, 1);  
  lcd.print(waste);  
  delay(1000);  
  Serial.print(cm);  
  Serial.print("cm");  
  Serial.println();  
}
```

```
// Change display for wet waste
```

```
if (moisture_percentage > 10) {  
  lcd.setCursor(0, 1);  
  waste = "Wet waste";  
  lcd.print(waste);  
  delay(1000);  
}
```

```
delay(100); // Short delay between loops  
}
```

Output

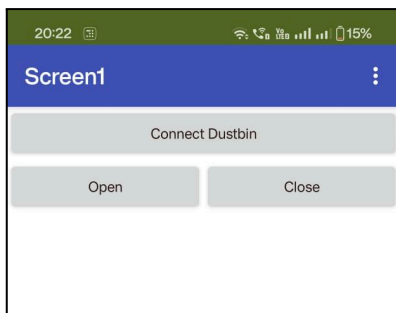


Fig 1. App home screen and bluetooth device

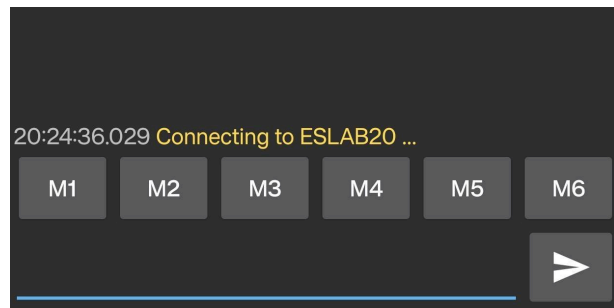


Fig 2. Establishing bluetooth connection



Fig 3. The Automated Dustbin

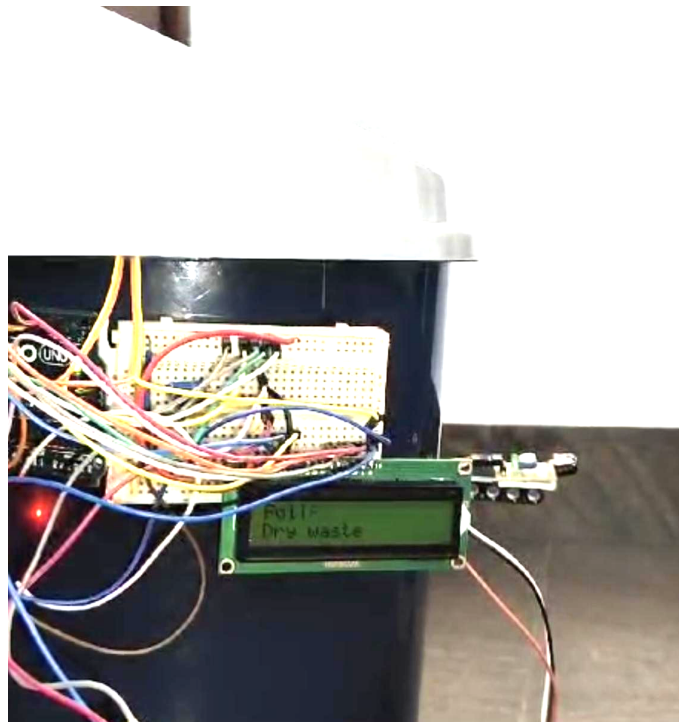


Fig 4. Display indicating that the dustbin is Full

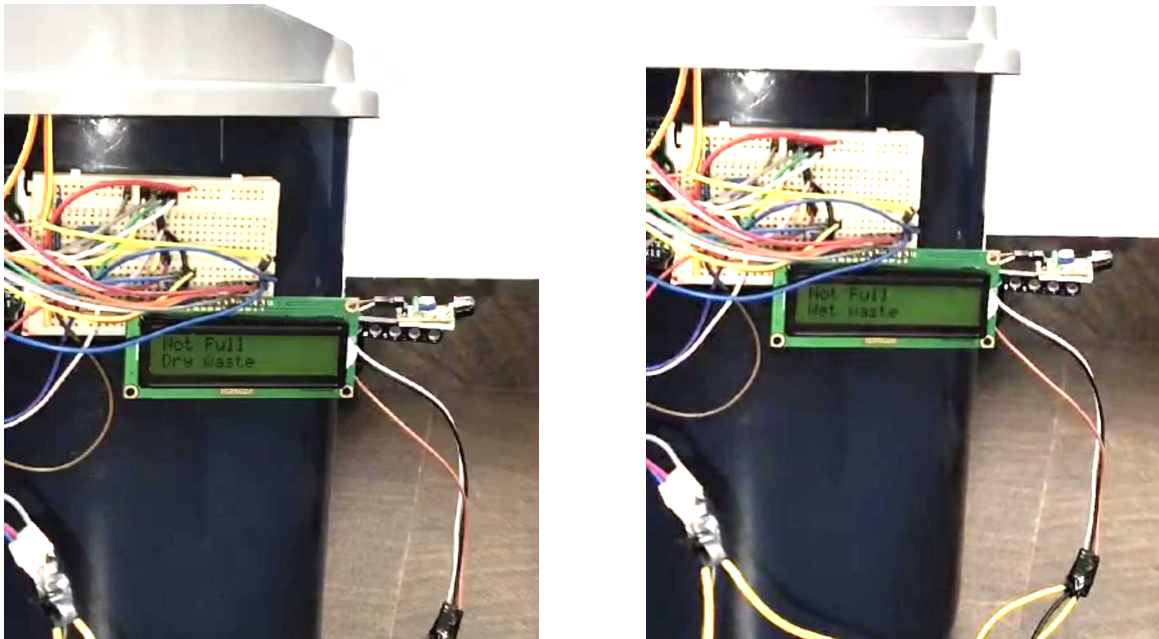


Fig 5. Dry waste and Wet waste detection

Result

In conclusion, the development of the automated dustbin represents a significant step forward in the realm of waste management technology. By addressing the limitations of traditional waste disposal methods, this innovative solution offers a comprehensive approach to handling household and commercial waste efficiently.

By encouraging proper waste management practices and optimising resource utilisation, it contributes to a cleaner and greener future.

Its successful implementation underscores the potential for technology to transform everyday tasks and promote greater efficiency, convenience, and sustainability in our communities.