Smart Dustbin for Waste Segregation Using Arduino and Azure:

Introduction:

The Smart Dustbin project aims to develop an intelligent waste segregation system that distinguishes between dry and wet waste. This system leverages an Arduino microcontroller, servo motor, ultrasonic sensor, soil moisture sensor, and Azure cloud technologies to enhance waste management efficiency and sustainability.

Objectives:

1.Automate Waste Segregation:Implement sensors to detect and classify waste as dry or wet automatically.

2. Data Collection and Monitoring: Use Azure cloud services for data storage, processing, and remote monitoring.

3. Promote Environmental Sustainability: Improve waste management practices to support recycling and reduce landfill waste.

Components and Technologies:

1. Hardware:

- Arduino Uno:The central microcontroller managing sensor data and controlling the servo motor.
- Servo Motor:Operates the dustbin lid and directs waste into designated compartments.
- Ultrasonic Sensor:Detects the presence of waste near the dustbin.
- Soil Moisture Sensor:Measures moisture content to classify waste.

2. Azure Cloud Services:

- Azure IoT Hub: Central hub for bi-directional communication between the Arduino and the cloud.
- Azure Stream Analytics: Real-time data stream processing service.
- Azure Storage (Blob Storage):Stores large amounts of unstructured data.
- Azure SQL Database:Manages structured data storage.
- Power BI:Provides interactive visualizations and business intelligence capabilities.

System Design:

1. Waste Detection:

- The ultrasonic sensor detects the proximity of waste when brought near the dustbin.
- Upon detection, the Arduino signals the servo motor to open the dustbin lid.

2. Waste Classification:

- Waste falls onto a platform where the soil moisture sensor measures its moisture content.
- The Arduino processes this data to classify the waste as dry or wet.

3. Waste Segregation:

- Based on the classification, the Arduino instructs the servo motor to direct the waste into the appropriate compartment within the dustbin (dry or wet).

4. Data Logging and Monitoring:

- Sensor data, including waste type and amount, is sent to Azure IoT Hub.
- Azure Stream Analytics processes this data in real-time.
- Power BI dashboards provide real-time and historical data visualization for monitoring purposes.

Implementation Plan:

1. Hardware Assembly:

- Connect the ultrasonic sensor and soil moisture sensor to the Arduino.
- Install the servo motor to control the dustbin lid and waste segregation mechanism.

2. Software Development:

- Develop Arduino code to read sensor data, classify waste, and control the servo motor.
- Implement scripts to send data from the Arduino to Azure IoT Hub using appropriate IoT protocols.

3. Azure Integration:

- Set up Azure IoT Hub to receive data from the Arduino.

- Configure Azure Stream Analytics to process incoming data.
- Use Azure Storage and Azure SQL Database for data storage.
- Create Power BI dashboards for real-time monitoring and visualization.

4. Testing and Calibration:

- Test the system with various types of waste to ensure accurate segregation.
- Calibrate sensors and fine-tune Arduino code to optimize performance.

5. Deployment:

- Deploy the system in a real-world environment and monitor its performance.
- Collect feedback and make necessary adjustments for improvement.

Expected Outcomes:

- Efficient Waste Segregation:Automated separation of dry and wet waste with minimal human intervention.
- Real-Time Monitoring: Ability to monitor waste data remotely through Azure cloud services.
- Scalability:Potential to scale the solution for use in larger waste management systems or smart cities.

Conclusion:

The Smart Dustbin project demonstrates the integration of IoT and cloud technologies to address environmental challenges. By automating waste segregation and providing real-time monitoring capabilities, the system promotes efficient waste management and sustainability.

Code

```
#include <Servo.h>
Servo servo1;
const int trigPin = 12;
const int echoPin = 11;
long duration;
int distance=0;
int potPin = A0; //input pin
int soil=0;
int fsoil;
void setup()
{
  Serial.begin(9600);
//Serial.print("Humidity");
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
servo1.attach(8);
}
void loop()
{


  int soil=0;
  for(int i=0;i<2;i++)
  {
digitalWrite(trigPin, LOW);
```

```
delayMicroseconds(7);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
delayMicroseconds(10);
duration = pulseIn(echoPin, HIGH);
distance= duration*0.034/2+distance;
delay(10);


  }
  distance=distance/2;
  Serial.println(distance);
if (distance <15 && distance>1)
{
  delay(1000);
for(int i=0;i<3;i++)
{
soil = analogRead(potPin) ;
soil = constrain(soil, 485, 1023);
fsoil = (map(soil, 485, 1023, 100, 0))+fsoil;
delay(75);
}
```

```
fsoil=fsoil/3;
Serial.println(fsoil);
Serial.print("%");
if(fsoil>3)
{delay(1000);
  Serial.print("WET ");
  servo1.write(180);
delay(3000);}
else{ delay(1000);
  Serial.print("dry ");
servo1.write(0);
delay(3000);}


servo1.write(90);}
distance=0;
fsoil=0;delay(1000);
}
```