

IOT BASED GARBAGE MONITORING SYSTEM FOR SMART CITY

G. SUJATHA, ASSISTANT PROFESSOR, DIET.

CH.DINESH, SR.ASST.PROFESSOR,DIET

Y. DINESH KUMAR, ASSISTANT PROFESSOR, DIET.

INTRODUCTION

ABOUT IoT:

The Internet of Things is a novel paradigm shift in IT arena. The phrase “Internet of Things” which is also shortly well-known as IoT is coined from the two words i.e. the first word is “Internet” and the second word is “Things”. Internet is a global network that provides variety of information and facilitates communication among people. IoT is the ability to transfer the data over a network without human to human or human to computer interaction.

SMART GARBAGE MONITORING

EXISTING SYSTEM:

In the existing system, garbage is collected by Municipal corporation by 2 days once. In these two days, the garbage in the garbage bin might be overflow well in advance before the commencement of the next cleaning process which creates unhygienic condition for the people who live in and around and also adds ugliness to the place by leaving bad smell. The smell will be heavy and produces air pollution and also spreads diseases. The dogs and other animals of that street eat the waste food and spread it all over the area and creates dirty environment.

Disadvantages of existing system:

- Time consuming and less effective.
- Employees are unaware of the status of a particular location.
- High costs.
- Unhygienic Environment and look of the city.
- Bad smell spreads and may cause illness to human beings.
- More traffic and Noise.

PROPOSED SYSTEM

- In this system, a 24x7 garbage monitoring system is designed for monitoring dumpsters
- In this dumpster, there are other two built in dustbins.
- One dustbin holds dry waste and the other holds wet waste.
- Unique ID will be provided for every dustbin to track the location using GPS.
- The ultrasonic sensor is used for measuring the level of waste in the dumpster.
- IR sensor and moisture sensor is used for separating wet and dry waste.
- All these sensors are connected to an Arduino Uno board.
- If either of the containers is full then an alert message is sent from the dumpster.
- In turn, employees can clear the corresponding dumpster.

Advantages

- Real time information on the level of waste filled in the dustbin.
- Deployment of dustbin based on the actual needs.
- Cost Reduction and resource optimization.
- Improves Environment quality -Fewer smells - Cleaner cities.
- System can be accessed anytime and from anywhere.
- Effective usage of dustbins.

SEGREGATION OF WASTE

Waste segregation is the separation of wet waste and dry waste. The generation of waste is unavoidable, and the materials carried in this waste impacts human and environmental health. Naturally, waste management is something that must be carried out, and one way to do this is segregation of wet and dry waste, so that dry waste can be recycled and wet waste can be composted. When we segregate waste, there is reduction of waste that reaches landfills and occupies space.



Waste Segregation

1.2.3.1 DRY WASTE

Dry waste consists of waste that doesn't decay. It is free of organic material such as food. It is also known as non-biodegradable waste. Dry waste recycling helps materials find second lives. For example, glass can be recycled hundreds of times into new jars and bottles.

Examples of dry waste:

- Plastic
- Glass
- Metal
- Timber
- Sawdust
- Textiles
- Cardboard and other mixed paper
- Home or office furniture
- Household junk



Dry Waste

WET WASTE

Wet waste is biodegradable. Recycling diverts food waste from landfill and turns it into compost.

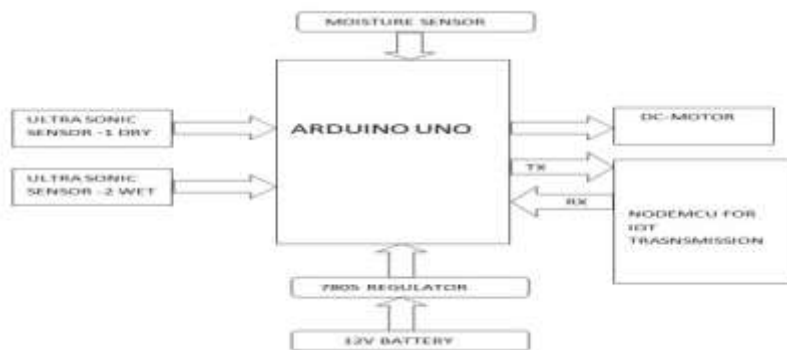
Examples of Wet waste:

- Organic food matter
- Cooking fats

- Oil
- Grease
- Liquid waste

WORKING PROCESS

IoT Based Smart Garbage and Waste Collection Bin creates unhygienic conditions for people as well as ugliness to that place leaving bad smell. To avoid all such situations we are going to implement a project called IoT Based Smart Garbage Monitoring and Waste Collection bins. This project on the working of the Wi-Fi module; essential for its implementation. The main aim of this project is to reduce human resources and efforts along with the enhancement of a smart city vision. By using sensor and GSM for smart cities environment clean and ensures hygienic surrounding. Improper disposal and improper maintenance of the domestic waste creates issues in public health and environmental pollution. This model consists of an Arduino controller, a few garbage bins loaded with sensors and monitored continuously through a web. This system also has a scope for citizen participation, wherein any grievances from citizens related to waste management is heard. The proposed e-monitoring system is an embedded system that comprises of IOT technology interfaced with Arduino micro-controller and a web base which is completely computerized. By employing the proposed system, the municipal authority could monitor the waste collection status effectively.



SYSTEM SPECIFICATION

HARDWARE SPECIFICATIONS

ATMEGA & ARDUINO UNO

Description

Any microcontroller based board which follows the standard Arduino schematic and is flashed with the Arduino boot loader can be called an Arduino board. The Arduino is referred to as open source hardware, since the standard schematic is open to everyone and anybody can make their own version of Arduino board following the standard schematic.

Arduino is a single board microcontroller, intended to make the application of interactive objects or environments more accessible. The hardware consists of an open source hardware board designed around an 8-bit Atmel AVR microcontroller, or a 32-bit Atmel ARM. Pre-programmed into the on-board microcontroller chip is a boot-loader that allows uploading programs into the microcontroller memory without needing a chip /device programmer.

Arduino started in 2005 as a project for students at the Interaction Design Institute Ivrea in Italy. The core Arduino developer team is composed of Massimo Banzi, David Cuartielles, and David Mellis. Arduino family consists of UNO, LILYPAD, DIECIMILA, NANO, and DUEMILANOVE.

Hardware

An Arduino board consists of an Atmel 8-bit microcontroller with complementary components to facilitate programming and incorporation into other circuits. Official Arduino have used the mega AVR series of chips, specifically the ATmega8, ATmega168, ATmega328, ATmega1280, and ATmega2560. Most boards include a 5 volt linear regulator and a 16 MHz crystal oscillator or ceramic resonator in some variants.

An Arduino microcontroller is also pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory, compared with other devices that typically need an external programmer. This allows an Arduino to be used by novices and experts alike without having to go through the difficulties first faced by many when using electronics by allowing the use of an ordinary computer as the programmer. At a conceptual level, when using the Arduino software stack, all boards are programmed over an RS-232 serial connection, but the way this is implemented varies by hardware version.

Current Arduino boards are programmed via USB, implemented using USB-to-serial adapter chips such as the FTDI FT232. When used with traditional microcontroller tools instead of the Arduino IDE, standard AVR ISP programming is used. Arduino board provides 14 digital I/O pins, six of which can produce pulse-width modulated signals, and other six analog inputs. The output or inputs can be

taken from the boards or given to the board using convenient connectors. Both digital and analog inputs and outputs are available in all arduino boards. The arduino boards can also communicate with other devices using standard communication ports like USART, IIC, and USB etc.



Arduino Board

Pin Description:

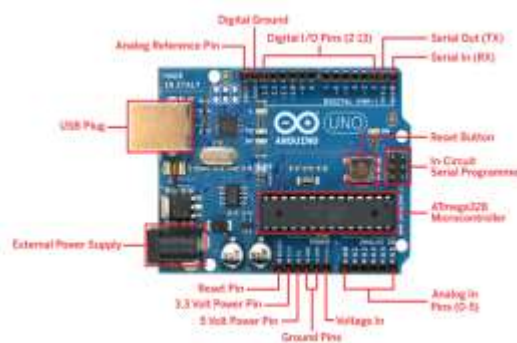


Fig No. 4.2 Arduino Board representing all Components

Serial: 0 (RX) and 1 (TX)

Used to receive (RX) and transmit (TX) TTL serial data. On the Arduino Diecimila, these pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip. On the Arduino BT, they are connected to the corresponding pins of the WT11 Bluetooth module. On the Arduino Mini and LilyPad Arduino, they are intended for use with an external TTL serial module.

Digital pins:

In addition to the specific functions listed below, the digital pins on an Arduino board can be used for general purpose input and output via the `pinMode()`, `digitalRead()` and `digitalWrite()` commands. Each pin has an internal pull-up resistor which can be turned on and off using `digitalWrite()`. When the pin is configured as an input. The maximum current per pin is 40 mA.

External Interrupts: 2 and 3

These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the `attachInterrupt()` function for details.

PWM: 3, 5, 6, 9, 10, and 11

Provide 8-bit PWM output with the `analogWrite()` function. On boards with an ATmega8, PWM output is available only on pins 9, 10, and 11.

BT Reset: 7

(Arduino BT-only) Connected to the reset line of the Bluetooth module.

SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK)

These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.

LED 13:

On the Diecimila and LilyPad, there is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

Analog pins

In addition to the specific functions listed below, the analog input pins support 10-bit analog-to-digital conversion using the `analogRead()` function. Most of the analog inputs can also be used as digital pins: analog input 0 as digital pin 14 through analog input 5 as digital pin 19. Analog inputs 6 and 7 -present on the Mini and BT cannot be used as digital pins. I²C: 4 and 5. Support I²C communication using the Wire library.

There are a couple of other pins on the board:

AREF:

Reference voltage for the analog inputs. Used with `analogReference()`.

Reset :

Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Power:

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically. External power can come either from an AC-to-DC adapter or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

1. VIN:

The input voltage to the Arduino board when it's using an external power source as opposed to 5 volts from the USB connection or other regulated power source. You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

2. 3V3:

A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

3. GND:

Ground pins.

4. 5V:

This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack 7 - 12V, the USB connector 5V, or the VIN pin of the board 7-12V. Supplying voltage via the 5V or 3.3V pins bypasses the regulator.

5.IOREF:

This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

Memory:

The ATmega328 has 32 KB with 0.5 KB used for the boot loader. It also has 2 KB of SRAM and 1 KB of EEPROM which can be read and written with the EEPROM library.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL 5V serial communication, which is available on digital pins 0 for RX and 1 for TX. An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an *.inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer but not for serial communication on

pins 0 and 1. A Software Serial library allows for serial communication on any of the

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 Ma
DC Current for 3.3V Pin	50 Ma
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)

Uno's digital pins.

EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Key Features of Arduino

Using Arduino IDE:

The most basic way to use the ESP8266 module is to use serial commands, as the chip is basically a WiFi/Serial transceiver. However, this is not convenient. What we recommend is using the very cool Arduino ESP8266 project, which is a modified version of the Arduino IDE that you need to install on your computer. This makes it very convenient to use the ESP8266 chip as we will be using the well-known Arduino IDE. Following the below step to install ESP8266 library to work in Arduino IDE environment.

Install the ESP8266 Board Package

Enter http://arduino.esp8266.com/stable/package_esp8266com_index.json into Additional Board Manager URLs field in the Arduino v1.6.4+ preferences.

Ultrasonic Sensor

An ultrasonic sensor is an instrument that measures the distance to an object using ultrasonic sound waves. An ultrasonic sensor uses a transducer to send and receive ultrasonic pulses that relay back information about an object's proximity. High-frequency sound waves reflect from boundaries to produce distinct echo patterns.



Ultrasonic Sensor

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.



IR Sensor

Moisture sensor

This Moisture Sensor can read the amount of moisture present in the water surrounding it. This Moisture Sensor can be used to detect the moisture of soil or judge if there is water around the sensor. Soil moisture sensors measure the lumetric water content in soil. 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.



Moisture Sensor

SOFTWARE SPECIFICATION

ARDUINO SOFTWARE

Introduction

- Arduino IDE is an open source software that is mainly used for Writing and compiling the code into the arduino module.
- It is used for making code compilation too easy that even a common person with no prior knowledge can get their feet wet with learning process.
- It is easily available for operating systems like MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment.
- A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more.
- Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code.
- The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
- The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.
- This environment supports both C and C++ languages.

BLYNK

Introduction

Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other things.

- **Blynk** is a new platform that allows you to quickly build interfaces for controlling and monitoring your hardware projects from your iOS and Android device.
- After downloading the **Blynk app**, you can create a project dashboard and arrange buttons, sliders, graphs, and other widgets onto the screen.
- **Blynk Server** is an Open-Source Net based Java **server**, responsible for forwarding messages between **Blynk** mobile application and various microcontroller boards.

Hardware platforms

Blynk supports hardware platforms such as Arduino, Raspberry Pi, and similar microcontroller boards to build hardware for your projects.

Connection types

Blynk supports the following connection types to connect the microcontroller board (hardware) with the Blynk Cloud and Blynk's personal server:

- Ethernet
- Wi-Fi
- Bluetooth
- Cellular
- Serial

Blynk architecture

There are three major components in the platform:

- **Blynk App** - allows to you create amazing interfaces for your projects using various widgets we provide.
- **Blynk Server** - responsible for all the communications between the smartphone and hardware. You can use our Blynk Cloud or run your private Blynk server locally. It's open source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.
- **Blynk Libraries** - for all the popular hardware platforms - enable communication with the server and process all the incoming and outgoing commands.



Blynk Architecture

CONCLUSION

By using this system, there is a possibility of waste segregation and we can also constantly check the level of the garbage in the dustbins which are placed in various parts of the city. If a particular dustbin has reached the maximum level then the corresponding employees are informed to empty it as soon as possible. The employees can also check the status of these bins anytime and from anywhere. So, this system helps to maintain the level of cleanliness in the city and form an environment which is better for living which ultimately leads to a 'Smarter City'.

REFERENCES

- [1]. Dr.N.Sathish kumar, B.Vijayalakshmi, R.Jenifer Prarthana, A.Shankar, "IOT Based Smart Garbage alert system using Arduino UNO" 978-1-5090-2597-8/16/\$31.00c 2016 IEEE, pp.1024-1034
- [2]. Narayan Sharma, Nirman Sinha, Tanmoy Dutta, "Smart Bin Implementation for Smart Cities" , International Journal of Scientific & Engineering Research, vol-6, Issue-9, 2015,pp.787-789H. Poor, An Introduction to Signal Detection and Estimation. New York: Springer-Verlag, 1985, ch. 4.
- [3]. Twinkle Sinha, Mugesh Kumar, P. Saisharan," Smart Dustbin", International Journal of Industrial Electronics and Electrical Engineering, SRM University, India,vol-3, Issue -5, 2015,pp.101 -104.
- [4]. Vikrant Bhor,"Smart Garbage Management System International Journal of Engineering Research & Technology(IJERT), Vol 4 Issue 03, March-201552000.
- [5]. [5] Yusuf AbdullahiBadamasi," The Working Principle of An Arduino, Electronics, Computer and Computation (ICECCO)", 2014 11th International Conference on 29 Sept.-1 Oct. 2014.