Smart Bin Monitoring System

Submitted by

Sakshi Asudhani
(Enroll No.: IU1741090033)

Sushant Menon
(Enroll No.: IU1741090013)

In partial fulfillment of the requirement for the degree of Bachelor of Technology
in the
Department of Electronics and Communication Engineering

Under the Guidance of

Internal Guide
Abhishek Vaghela
Assistant Professor
Electronics and Communication
Indus University

External Guide
Parthraj Gohil
CEO
Embedded Systems and IoT
CoreFragment

INDUS INSTITUTE OF TECHNOLOGY AND ENGINEERING
INDUS UNIVERSITY

May, 2021

Smart Bin Monitoring System

Submitted by

Sakshi Asudhani
(Enroll No.: IU1741090033)

Sushant Menon
(Enroll No.: IU1741090013)

In partial fulfillment of the requirement for the degree of Bachelor of Technology
in the
Department of Electronics and Communication Engineering

Under the Guidance of

Internal Guide
Abhishek Vaghela
Assistant Professor
Electronics and Communication
Indus University

External Guide
Parthraj Gohil
CEO
Embedded Systems and IoT
CoreFragment

INDUS INSTITUTE OF TECHNOLOGY AND ENGINEERING
INDUS UNIVERSITY

May, 2021

DECLARATION

By the B.Tech Student

I hereby declare that the Report of the U.G. Project Work entitled ***Smart Bin Monitoring System*** which is being submitted to the **IITE, Indus University**, in partial fulfillment of the requirements for the award of the Degree of **Bachelor of Technology** in **Electronics and Communication Engineering** in the department of Electronics and Communication Engineering, *is a bonafide report of the work carried out by me.* The material contained in this Report has not been submitted to any University or Institution for the award of any degree.

Sakshi Asudhani,
Department of Electronics and Communication Engineering

Sushant Menon,
Department of Electronics and Communication Engineering

Place: IITE, Indus University.
Date: 10th May, 2021

BONAFIDE CERTIFICATE

This is to certify that the U.G. project work report entitled **Smart Bin Monitoring System** submitted by **Sakshi Asudhani**(Enroll No: IU1741090033) and **Sushant Menon**(Enroll No: IU1741090013) as the record of the work carried out by him, is accepted as the U.G. project work report submission in partial fulfillment of the requirements for the award of degree of **Bachelor of Technology** in **Electronics and Communication Engineering** in the Department of **Electronics and Communication Engineering of IITE, Indus University** during the academic year 2020-2021.

(Abhishek Vaghela)
Project Guide

(Dr. Vrushank Shah)
HOD,EC

ACKNOWLEDGEMENTS

We would like to acknowledge Professor Abhishek Vaghela for his valuable and constructive suggestions during the planning and development of this project work. His willingness to give his time so generously has been very much appreciated.We would also like to thank Parthraj Gohil for his patient guidance, enthusiastic encouragement and useful critiques of this project work. We would also like to thank the staff of CoreFragment for enabling us to visit their office to observe their daily operations.

Sakshi Asudhani and **Sushant Menon**

ABSTRACT

The Main aim of this project is to develop an Intelligent System which can monitor the bins through sensors and gives the information in detail through the internet. Sensors will measure and calculate the content of the bin and information will be sent to the AWS server. By This approach we can get information of bin by using an android app or any other device given that it has access to the Internet.

Contents

List of Figures

Chapter 1

INTRODUCTION

[1] In Smart Cities Generation of waste is increasing due to rapid growth of people and industries in urban areas and the biggest problem to authorities is Collection of wastage from different locations i,e Houses, Public Places and Industries. Due to the lack of proper information an amount of 85 percent of the total municipal solid waste budget is spent on waste collection and transportation. To tackle this problem we need an intelligence to monitor waste and gives the complete information to authorities.

1.1 Components Used

1. **ESP32**

   ESP32 is the main processing unit of the System. ESP32 can perform as a complete standalone system or as a slave device to a host MCU, reducing communication stack overhead on the main application processor. ESP32 can interface with other systems to provide Wi-Fi and Bluetooth functionality through its SPI / SDIO or I2C / UART interfaces.In this System we have interfaced Touch Sensing Module, Weight Sensing Module and I2C LCD Display with the ESP32.

2. **Weight Sensor (HX711 Module)**

   [2] HX711 is a precision 24-bit analogto-digital converter (ADC) designed for weigh scales and industrial control applications to interface directly with

a bridge sensor.

3. **I2C LCD Display**

   This is a 16x2 LCD display screen with I2C interface. It is able to display 16x2 characters on 2 lines, white characters on blue background.

   Usually, Arduino LCD display projects will run out of pin resources easily, especially with Arduino Uno. And it is also very complicated with the wire soldering and connection. This I2C 16x2 Arduino LCD Screen is using an I2C communication interface. It means it only needs 4 pins for the LCD display: VCC, GND, SDA, SCL. It will save at least 4 digital/analog pins on Arduino.

4. **LED's and connecting Wires**

   LED's are being used to notify the user the status of the Smart Bin. For example Connecting to WiFi, Connected to WiFi, Going to sleep and Threshold level Crossed.

5. **Touch Module**

   This allows the user to see the state of any bin by simply touching the Wake-up Touch pad without having to actually lifting any of the bin lids.

1.2 AWS IoT Core

AWS IoT Core lets you select the communication protocol most appropriate for your use case to connect and manage IoT devices. AWS IoT Core supports MQTT (Message Queuing and Telemetry Transport), HTTPS (Hypertext Transfer Protocol - Secure), MQTT over WSS (WebSockets Secure), and LoRaWAN (low-power long-range wide-area network). For devices and clients that use MQTT, MQTT over WSS, and HTTP protocols, AWS IoT Core Message Broker provides a high-throughput publish/subscribe message broker to transmit messages to and from all of your IoT devices and applications.

Chapter 2

BLOCK DIAGRAM

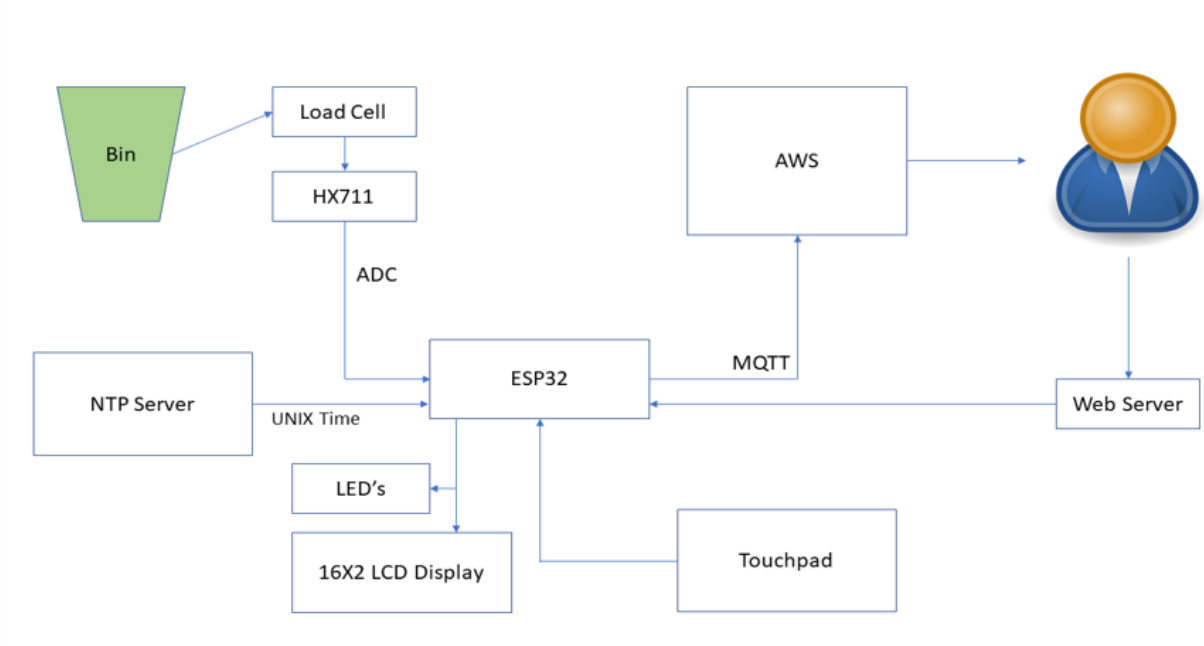

Figure 2.1: Block diagram of the System

The System will initially try to connect to a WiFi stored in it's EEPROM. If it does not find any credentials it will start a access point to which the user can connect to. The user is then required to connect with the System via that access point. Once the user is connected they will be taken to a Captive portal where they will have to enter the credentials of their WiFi which they want the system to be connected to. These credentials will then be saved in the EEPROM of the ESP32 so that the next time the system will automatically be connect to the WiFi.

After establishing a successful WiFi connection, the system will connect to an AWS Server (AWS IoT Core). After successfully connecting to the Server the system will record the weights present in the bin using the Loadcell and HX711 Module. After recording these weights it will publish those records to the AWS Server where it will be stored. The protocol followed between the ESP32 and AWS Server is known as MQTT Protocol which works on a Subscribe/Publish Model. To minimize power consumption once the records are sent to the server the system will go to a deep sleep mode. The system can be waken up using three different modes.

- Wake-up using Timer
- Wake-up using touch module
- Wake-up using lid-off detection

These modes are explained in detail in the next section.

The time format used in our system is UNIX time. This format is being fetched from the NTP Server which is freely available for use.

One additional feature of the lid-off detection is when the lid is put back on, and the weight of the bin is taken. If it is below the warning or critical warning threshold, the Master Warning LED, as well as the appropriate warning LED for that particular bin, is turned on for five seconds. Since a weight change can only occur when the lid of a bin is lifted, this five-second warning allows the user to see that the weight of the bin has dropped below its threshold. It also lights up the appropriate warning LED's as well as the Master Warning LED if threshold conditions are met.

To display the Weight of the bin an I2C 16X2 LCD Display is also interfaced with the ESP32.

Chapter 3

CIRCUIT DIAGRAM

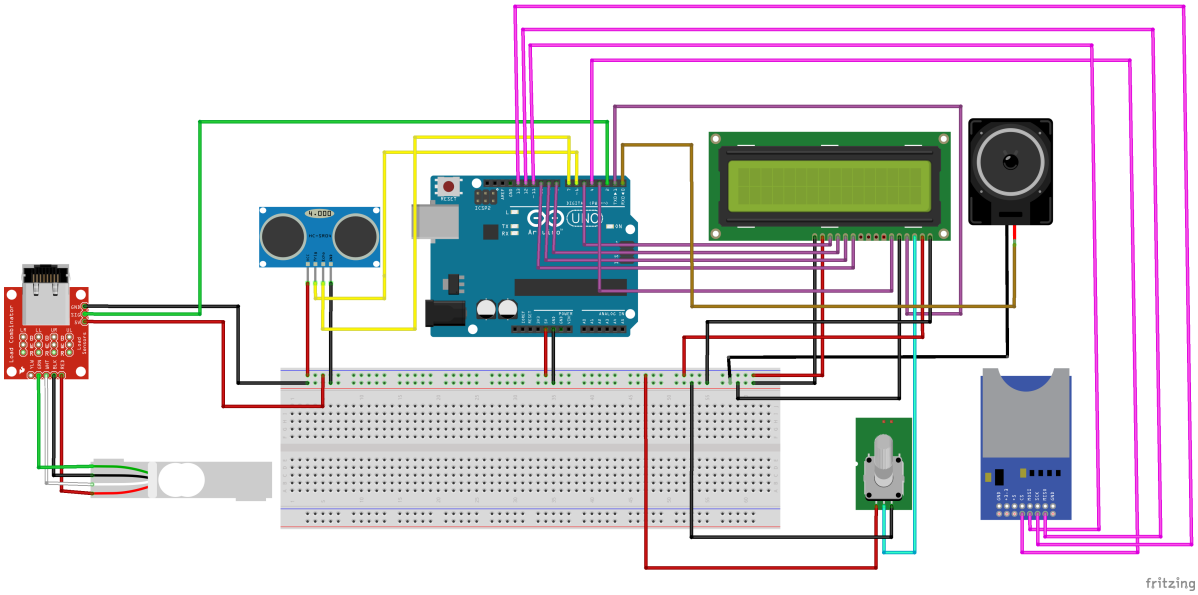

Figure 3.1: Circuit Diagram of the System

Chapter 4

WORKING

4.1 Internet Connectivity

As this is an IoT device it needs to be connected to the internet We will be connecting the device to the Internet using Wi-Fi If the Wi-Fi credentials are stored in the EEPROM of the ESP32 it will connect automatically or else it will create it's own Wi-Fi access point After the Access point is created the user can connect to it and he/she will be directed to webpage where he/she can enter his/her Wi-Fi and personal credentials These credentials will the be stored in EEPROM and SPIFF inside the ESP32 so that it can connect to the it automatically the next time it reboots. The system simply reports the weights of each bin everyday at midnight. First, as soon as the system is setup by the user, it will use the Wi-Fi SSID/Password to connect to Epoch time. Then every night, while uploading to AWS, it will re-adjust the time so as to eliminate any accumulated inaccuracies. This is done automatically, and is transparent to the user. Normally, the system is in deep sleep. After a normal bootup, and assuming the time is properly Setup, it will immediately read the current time and adjust its wake-up timer to wake it up a sometime close to midnight to read and report the bin weights to AWS. Since the internal timer is not very accurate, the actual reporting could be not exactly at midnight, and that is just fine. A few minutes of inaccuracy is acceptable for this application. The system also wakes up due to an exception. These are described in the next several sections.

4.2 Wakeup on Lid-off Detection

If any of the lids is lifted. The system wakes up. It will immediately take to weight of the bin whose lid has been lifted and the time. These are locally stored. Then these additional readings are sent to AWS at around midnight, along with the weights of all the bins as usual. One additional feature of the lid-off detection is when the lid is put back on, and the weight of the bin is taken. If it is below the warning or critical warning threshold, the Master Warning LED, as well as the appropriate warning LED for that particular bin, is turned on for five seconds. Since a weight change can only occur when the lid of a bin is lifted, this five-second warning allows the user to see that the weight of the bin has dropped below its threshold. Since the user should still be around at the very moment when the lid was put back on, five seconds is more than ample time for him to see/hear the warning.

4.3 Wakeup on Touch

At any time, the user can touch the Wake-up Touch pad, and wake up the system. In this case, the system uses the locally stored weights, and displays them on the LCD screen for five seconds. It also lights up the appropriate warning LED's as well as the Master Warning LED if threshold conditions are met. This allows the user to see the state of any bin by simply touching the Wake-up Touch pad without having to actually lifting any of the bin lids.

Bibliography

[1] downtoearth website, *https://www.downtoearth.org.in/blog/waste/india-s-challenges-in-waste-management-56753*.

[2] 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales, *https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711$_e$nglish.pdf*.