IoT BASED FRUIT QUALITY MONITORING SYSTEM

Mini Project Report Submitted by

NISHITHA RAJEEVAN (TLY21EC061)
POORNIMA KP (TLY21EC064)
SIDHARTH PRADEEP (TLY21EC073)
MUHAMMED ADNAN (TLY21EC051)

Towards the partial fulfillment of the requirement for the award of B. Tech. degree in

Electronics and Communication Engineering

Department of Electronics and Communication Engineering

College of Engineering Thalassery, kannur - 670107

May 2023-2024

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

COLLEGE OF ENGINEERING THALASSERY

CERTIFICATE

This is to certify that the project report entitled **IoT BASED FRUIT QUALITY MONITORING SYSTEM** is a bonafide record of the work done by **POORNIMA KP (TLY21EC064) NISHITHA RAJEEVAN(TLY21EC061) SIDHARTH PRADEEP (TLY21EC073) MUHAMMED ADNAN YAKOOB (TLY21EC051)**, towards the partial fulfillment of the requirement for the award of the Degree of Bachelor of Technology in Electronics and Communication Engineering under APJ Abdul Kalam Technological University during the academic year 2023-2024.

Ms Safoora OK
Project Supervisor
Department of ECE

Mr. Jinesh S
Project Co-ordinator
Department of ECE

DR. Sudheer V R
Head of the department
Department of ECE

DECLARATION

We, the undersigned, hereby solemnly declare that this project report titled IoT BASED FRUIT QUALITY MONITORING SYSTEM , submitted for the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering from APJ Abdul Kalam Technological University is a bonafide record of our own work carried out under the supervision of Ms Safoora O K.

Wherever we have used materials (data, theoretical analysis, and text) from other sources, we have adequately and accurately cited the original sources.

NISHITHA RAJEEVAN (TLY21EC061)
POORNIMA KP (TLY21EC064)
SIDHARTH PRADEEP (TLY21EC073)
MUHAMMED ADNAN (TLY21EC051)

PLACE : THALASSERY
DATE : 06-05-2024

ACKNOWLEDGMENTS

I would like to take this opportunity to extend my sincere thanks to people who helped me to make this project possible. This project would be incomplete without mentioning all the people who helped me to make it real. We are extremely grateful to the Principal DR. Rajeev P , College of Engineering Thalassery, for providing the necessary facilities. We would like to express our sincere gratitude to DR. Sudheer VR, Head of Department of Electronics and Communications for giving us all the support and confidence to work with project. We express our sincere gratitude to our project guide Ms Safoora O K and project co-ordinator Mr. Jinesh S for their guidance and support throughout the duration of the project.

We would like to thank all our teaching and non-teaching staff of College of Engineering, Thalassery for their valuable help in the successful completion of our project. We extent our thanks to family members and especially to our friends for encouraging and helping us in critical situations and make the project successful. We also wish to thank our friends who helped us in performing certain tests and gave us various suggestions.

ABSTRACT

Food safety and hygiene are among the key concerns in order to prevent the wastage of food. However, for lack of technology and ignorance about the effects of humidity, temperature, exposure to light and alcohol content on foods, food safety is not maintained well enough. This has led to massive losses in many food stores resulting from food decay. Currently, majority of food stores and ware- houses still rely on manual monitoring of the atmospheric factors related to food quality. These conventional food inspection technologies are limited to weight, volume, colour and aspect inspection and as a result do not provide a lot of information needed on quality of food. The quality of the food needs to be monitored and it must be prevented from rotting and decaying by the atmospheric factors like temperature, humidity and dark. This project is focused on such a food monitoring system which suggests systematic use of various sensors to perform quality monitoring . More precisely, this system consists of gas, temperature and humidity sensors, which provide the essential information needed for evaluating the quality of the packed or stored product. This information is transmitted wirelessly to a computer system providing an interface where the user can observe the evolution of the product quality over time using the Internet of Things technology.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Food contamination can occur in the production process, but also a large part caused by the inefficient food handling because of inappropriate ambient conditions when the food is being transported and stored. There are many factors leading to food poisoning, typically changes in light intensity, temperature, alcohol content and humidity are important factors. A monitoring system capable of measuring temperature and humidity variability during transport and storage is of prime importance. Today almost everybody is getting affected by the food they consume, it's not only about the junk food, but all the packed foods, vegetables, products consumed and used in daily life, as all of them do not offer quality since their temperature, moisture, oxygen content vary from time to time. Majority of consumers only pay attention to the information provided on the packaging, i.e. the amount of ingredients used and their nutritional value but they forget that they are blindly risking their health by ignoring the environmental conditions to which these packets are subjected. Every product making firm just want to attract more and more costumers towards them and their main motive is to sell the product anyhow like by adding more flavours, colouring chemicals and preservatives to increase the taste and appearance but they forget that these money making tactics are actually affecting the consumers' health. High temperature and relative humidity favour the development of post-harvest decay organisms. More acidic tissue is generally attacked by fungi, while fruits and vegetables having pH above 4.5 are more commonly attacked by bacteria. A wide variety of foods can also undergo changes in colour, flavour, and nutrient composition when exposed to light. The extent of these changes depends on many factors including the composition of the food and the light source. Light exposure could result in colour and vitamin loss. Light also may be responsible for the oxidation of fats. Some types of yeasts can also lead to spoilage. True yeast metabolizes sugar producing alcohol and carbon dioxide gas. This process is known as fermentation. Hence by checking on alcohol quantity content one can detect the quality of food[1].

The system proposed is intended for food quality monitoring. In this project, a similar food quality monitoring device will be designed that will keep watch of environmental factors like temperature, humidity, alcohol content.

1.1 Design Objectives

The proposed system should be able to;

1.Read temperature and relative humidity in the food store.

2.Detect the emission of ethanol type of gases.

3.Collect data from all the sensors and pass to LCD for display.

4.Monitor the sensor data visually online.

1.2 Motivation

Existing systems have been unable to provide food safety guarantees. Currently the performances and analysis of routine measurements, aimed at detecting changes in the nutritional or health status of the food does not guarantee that. This project proposes a system to analyze the ambient conditions under which the fruit is being stored and transported. The proposed solution senses the temperature, humidity, alcohol content and light parameters of surrounding environment as these parameters affect nutritional values of fruits.

LITERATURE REVIEW

In recent years, there has been a growing interest in leveraging IoT (Internet of Things) technologies for various agricultural applications, including fruit quality monitoring. This literature review synthesizes existing research and developments related to an IoT-based fruit quality monitoring system utilizing MQ-3, MQ-4, and DHT11 sensors, an ESP8266 Wi-Fi module, LCD display with I2C interface, Arduino Uno microcontroller, and ThingSpeak platform.

IoT technologies offer significant advantages in agriculture by enabling real-time monitoring and management of environmental conditions, thereby improving crop yield, quality, and resource utilization. Fruit quality monitoring is crucial for ensuring produce meets market standards, reducing post-harvest losses, and enhancing consumer satisfaction.

CHAPTER 2

PROPOSED SYSTEM

2.1 Block Diagram

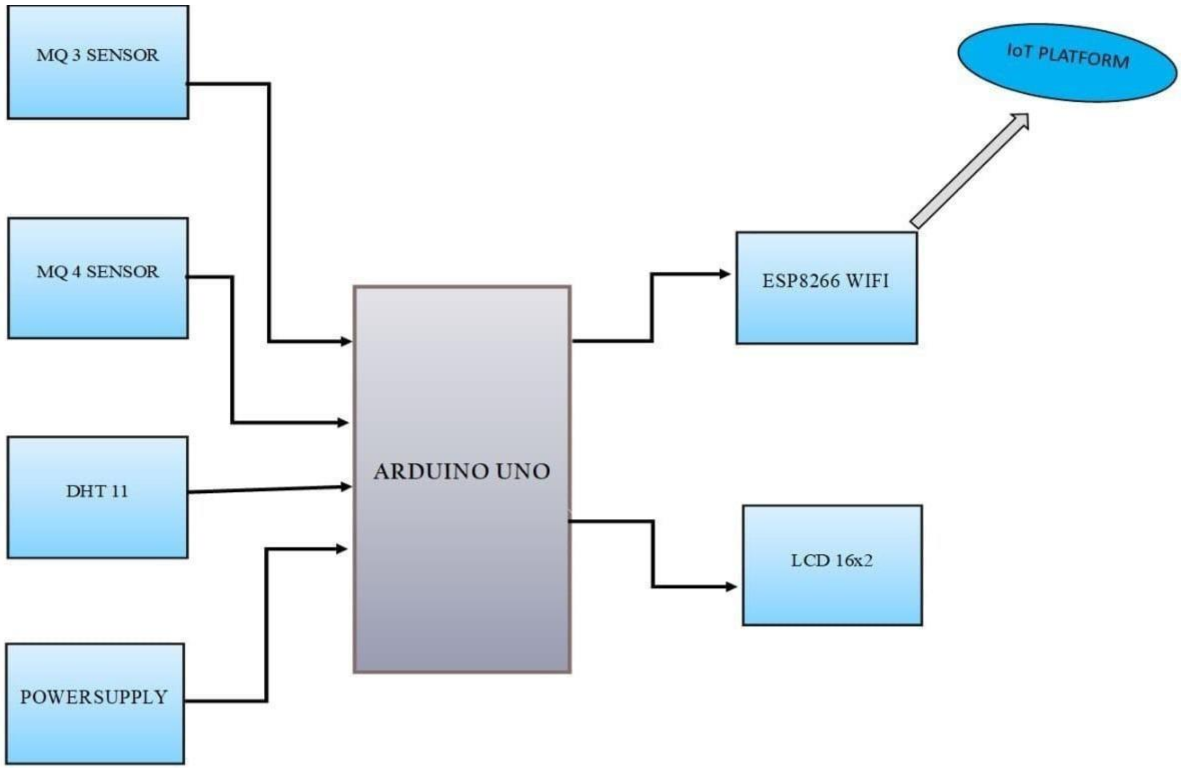

Fig. 2.1. Block Diagram

2.1.1 Arduino Uno

The Arduino Uno serves as the core controller in an IoT-based fruit quality monitoring system, managing data acquisition from MQ-3 and MQ-4gas sensors to assess the fruit's environmental conditions and gas quality. It interfaces with IoT modules to transmit this data, enabling real-time monitoring and ensuring optimal fruit storage conditions.

2.1.2 Power Supply

In an IoT-based fruit quality monitoring system using MQ-3 and MQ-4 gas sensors, a stable and reliable power supply is crucial to ensure uninterrupted operation. Typically, this system requires a regulated power source, such as a DC adapter or a battery with appropriate voltage levels, to power the Arduino and sensors while considering power efficiency for extended monitoring periods.

2.1.2.1 LCD Display

In an IoT-based fruit quality monitoring system utilizing MQ-3 and MQ-4 gas sensors, an LCD display provides a real-time visual interface, showing gas concentration and environmental data for easy monitoring. This LCD display enhances user interaction, allowing stakeholders to assess fruit quality conditions at a glance.

2.1.3 DHT 11

In an IoT-based fruit quality monitoring system, alongside MQ-3 and MQ-4 gas sensors, provides critical environmental data such as temperature and humidity levels, contributing to a comprehensive assessment of fruit storage conditions. This combined sensor setup ensures that both gas quality and environmental factors are monitored to maintain optimal fruit quality and freshness.

2.1.4 MQ-4

The MQ-4 gas sensor, alongside the MQ-3, enhances the IoT-based fruit quality monitoring system by detecting a wider range of gases relevant to fruit storage, such as methane and natural gas, providing a more comprehensive assessment of environmental conditions. This dual-sensor setup enables precise monitoring and
timely interventions to maintain fruit quality and safety

2.1.5 MQ-3

The MQ-3 gas sensor in an IoT-based fruit quality monitoring system, along- side the MQ-4 sensor, detects and measures gases like ethanol and methane, helping assess fruit storage conditions. These sensors play a crucial role in ensuring the freshness and safety of fruits by monitoring gas levels that can affect fruit quality.

2.1.6 ESP8266 Wi-Fi Module

The ESP8266 Wi-Fi module acts as the communication bridge in an IoT-based fruit quality monitoring system, facilitating data transmission from MQ-3 and MQ-4 gas sensors to a central platform, allowing remote monitoring of fruit storage condi- tions and gas quality. Its wireless connectivity ensures seamless integration and data accessibility for efficient fruit quality management.

2.2 Algorithm

1. Initialize the hardware.
2. Analyse and store the data collected using sensors.
3. Display the collected data on the LCD display.
4. Transmit the collected data onto an Iot platform(Thinkspeak)and display it.
5. Compare the data collected using sensors with that of the threshold values set for each data.
6. Display ’fruit is good’ if the conditions are satisfied and display ’fruit is bad’ if it doesn’t satisfy them.

CHAPTER 3

HARDWARE DESCRIPTION

3.1 Arduino Uno

Fig. 3.1. Arduino Uno R3

A basic Arduino consists of a simple base board that has the microcontroller and its support circuitry with connectors to connect to plug in modules and a USB interface to download code from the PC. The Arduino Uno shown in Fig 3.1 is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 Analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button. On the peripherals there will be 2x 8-bit Timer/Counter with a dedicated period register and compare channels ,1x 16-bit Timer/Counter with a dedicated period register, input capture and compare channels, 1x USART with fractional baud rate generator and start-of-frame detection ,1x controller/peripheral Serial Peripheral Interface (SPI) ,1x Dualmode controller/peripheral I2C, 1x Analog Comparator (AC) with a scalable reference in- put, Watchdog Timer with separate on-chip oscillator ,Six PWM channels ,Interrupt and wake-up on pin change. You can also find a USB port 6 on the board, which could both be a power and data port, a barrel jack for the power supply, a LED power indicator, a reset button, a voltage regulator, and TX/RX LEDs. There is also a set of labelled pins for 5V, 3.3V, GND, Analog, Digital, PWM, and AREF. These pins are mainly used to attach expansion cards, or SHIELDS, to Arduino for extra functionalities like network connection, LCD, and joysticks. The pins can also be used to attach components from a breadboard for prototyping. Arduino boards are available at low cost and they are super easy to learn. There is no need for any external hardware, multiple projects can be made using single Arduino boards which cut overall project cost by a great margin. While most of the microcontrollers can only be programmed using Windows Arduino is not only limited to Windows it is also available across multiple platforms like Linux and macOS. Arduino's accessibility, easy-to-understand hardware design, and simple software make it appealing to different types of users. programming an off-the-shelf microcontroller is often messy, and the code is not easy to comprehend, especially for newbies. Arduino provides a simple and easy-to-understand coding platform through the Arduino IDE. It is friendly to students and to those who are still new in electronics projects, simplifying coding and compiling and uploading codes to the board, eliminating the need for an external programmer or burner. Even professionals and experienced programmers are using Arduino IDE because of its uncomplicated interface

3.2 LCD Display

The LCD (Liquid Crystal Display) shown in Fig 3.2 is a type of display that uses the liquid crystals for its operation. Here, we will accept the serial input from the computer and upload the sketch to the Arduino. The characters will be displayed on the LCD. The library that allows us to control the LCD display is called Liquid Crystal Library, The Liquid Crystal Display has a parallel interface. It means that the

Fig. 3.2. LCD Display

microcontroller operates several pins at once to control the LCD display. The 16-pins present on the LCD display are RS-The Register Select (RS) pin controls the memory of the LCD in which we write the data. We can select either the data register or the instruction register. The LCD looks for the upcoming instruction, which is present in the instruction register.R/W-The Read/Write pin selects the reading or writing mode. The Enable (E) mode is used to enable the writing to the registers. It sends the data to the data pins when the mode is HIGH. There are eight data pins numbered as D0, D1, D3, D3, D4, D5, D6, and D7. We can set the state of the data pin either HIGH or LOW. Pin 1 of the LCD is the Ground pin, and pin 2 is the VCC or the voltage source pin. The pin 3 of the LCD is the VEE or the contrast pin. For example, we can connect the potentiometer's output to the VEE and can adjust the contrast of the LCD. The A and K pins are also called as Backlight pins (Bklt+ and Bklt-). The process includes putting the data (to be displayed on the LCD screen) into the data registers. The instructions in the Register Select are kept in the instruction register.

The liquid crystal library has simplified process to display the characters on the LCD.

3.3 MQ-4

The MQ4 methane gas sensor shown in Fig 3.3 is extremely used for detecting gas leakage at home or in industries like Methane (CH_4) CNG Gas. This gas sensor is highly responsive in very little time, so based on the sensitivity requirements; it can be adjusted through a potentiometer. This is an Analog output sensor, used like a CNG (compressed natural gas) sensor within the series of MQ sensors. So this sensor is suitable for detecting the concentration of natural gas like methane within the air. For this sensor, if the gas concentration increases then the output voltage will be increased. This sensor works with 5V DC and draws 750 mW around. This article discusses an overview of the MQ4 methane gas sensor and its working with applications. MQ4 methane gas sensor is a MOS (metal oxide semiconductor) type

Fig. 3.3. MQ-4

sensor, used to detect the methane gas concentration within the air at either home or industries generates output like Analog voltage by reading it. Here, the range of concentration for sensing ranges from 300 pm – 10,000 ppm which is appropriate for the detection of a leak. This gas sensor mainly includes a detecting element like ceramic based on Aluminium-oxide (AlO), coated with Tin dioxide (SnO_2) and arranged within a stainless-steel mesh. When methane gas and detecting elements get in contact with each other then the resistivity of the detecting element will be changed. After that, the change is measured to get the methane gas concentration. The ignition of Methane gas is extremely exothermal which means it generates a huge amount of heat once ignited.

3.4 MQ-3

Fig. 3.4. MQ-3

The MQ3 alcohol shown in Fig 3.4 gas sensor is a module used for detecting alcohol, CH4, benzene, gasoline, hexane, CO, and LPG. It has a sensitive material SnO2 for alcohol gas detection, with lower electrical conductivity in the fresh air. It is a semiconductor alcohol gas sensor that detects or monitors the presence or absence of alcohol. It is also known as chemiresistors because sensing of the sensitive material depends on the resistance change when the sensor is exposed to alcohol gas. When the sensor is pointed closer to the alcohol gas, the SnO2 conductivity increases. The increase in sensor conductivity is directly proportional to the alcohol concentration. Therefore, the alcohol concentration is measured by any microcontroller very easily. The MQ3 alcohol gas sensor is very fast and has a high sensitivity to alcohol, smoke, and gasoline. An Alcohol detector can be made using this alcohol sensor. The concentration of alcohol gas sensing range in fresh air or atmosphere by the MQ3 sensor is 0.04mg/L-4 mg/L, which is acceptable for breathanlayzers. It consumes 150 mA and operates with a 5V power supply at -10°C to 50°C temperature.

3.5 DHT 11

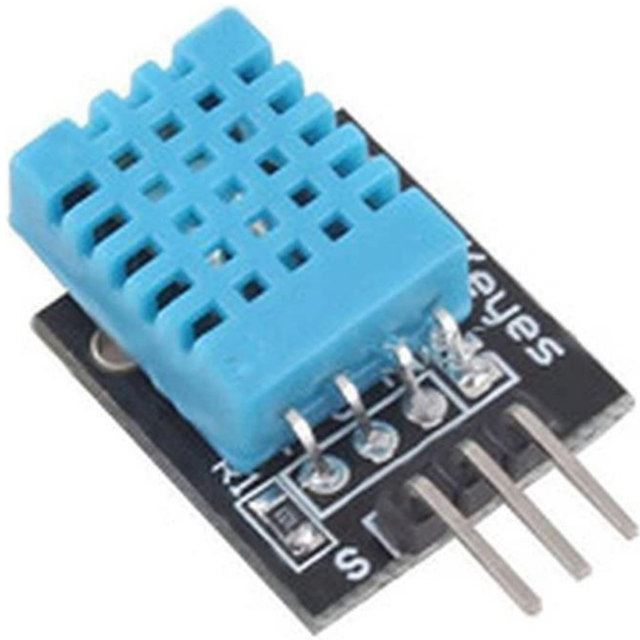

Fig. 3.5. DHT 11

Fig 3.5shows DHT 11 sensor. These sensors are very popular for electronics hobbyists because there are very cheap but still providing great performance. We have two versions of the DHT sensor, they look a bit similar and have the same pinout, but have different characteristics a) DHT 11 Sensor b) DHT 22 Sensor The DHT22 is the more expensive version which obviously has better specifications. Its temperature measuring range is from -40 to +125 degrees Celsius with +-0.5 degrees' accuracy, while the DHT11 temperature range is from 0 to 50 degrees Celsius with +-2 degrees' accuracy. Also the DHT22 sensor has better humidity measuring range, from 0 to 100accuracy, while the DHT11 humidity range is from 20 to 80rate which for the DHT11 is 1Hz or one reading every second, while the DHT22 sampling rate is 0,5Hz or one reading every two seconds and also the DHT11 has smaller body size. The operating voltage of both sensors is from 3 to 5 volts, while the max current used when measuring is 2.5mA. DHT11 / DHT22 Working Principle They consist of a humidity sensing component, a NTC temperature sensor (or thermistor) and an IC on the back side of the sensor. For measuring humidity, they use the humidity sensing component which has two electrodes with moisture holding substrate between them. So as the humidity changes, the conductivity of the substrate changes or the resistance between these electrodes changes. This change in resistance is measured and processed by the IC which makes it ready to be read by a microcontroller. On the other hand, for measuring temperature these sensors use a NTC temperature sensor or a thermistor. A thermistor is actually a variable resistor that changes its

resistance with change of the temperature [2]. These sensors are made by sintering of semi conductive materials such as ceramics or polymers in order to provide larger changes in the resistance with just small changes in temperature. The term
"NTC" means "Negative Temperature Coefficient", which means that the resistance decreases with increase of the temperature

3.6 THINGSPEAK

ThingSpeak is a managed open-source platform used for prototyping that enables systems and devices to upload data to the Internet and perform data analysis on the uploaded data. It uses HTTPS and MQTT protocols to store and retrieve data from devices and systems. Data can be sent to ThingSpeak from a device or a system, a real-time visualization of the data created, and alerts sent using web services like Twitter and Twilio. (ThingSpeak, 2019) In ThingSpeak platform, devices can be easily configured to send data to the website, the uploaded data can then be aggregated, analyzed using MATLAB and visualized in real time. The data analysis can be done automatically based on schedules [2]. The systems can be connected to the Internet without setting up a server or developing a software.

3.7 ESP8266 Wi-Fi MODULE

An ESP8266 Wi-Fi module shown in Fig 3.6 is a SOC microchip mainly used for the development of end-point IoT (Internet of things) applications. It is referred to as a standalone wireless transceiver, available at a very low price. It is used to enable the internet connection to various applications of embedded systems. Express if systems designed the ESP8266 Wi-Fi module to support both the TCP/IP capability and the microcontroller access to any Wi-Fi network. It provides the solutions to meet the requirements of industries of IoT such as cost, power, performance, and de- sign. It can work as either a slave or a standalone application. If the ESP8266 Wi-Fi

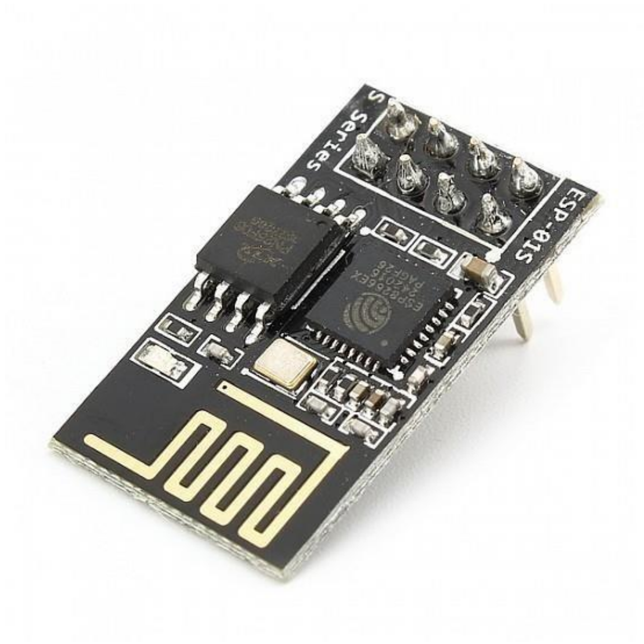

Fig. 3.6. ESP8266 Wi-Fi Module

runs as a slave to a microcontroller host, then it can be used as a Wi-Fi adaptor to any type of microcontroller using UART or SPI. If the module is used as a standalone application, then it provides the functions of the microcontroller and Wi-Fi network.

The ESP8266 Wi-Fi module is highly integrated with RF balun, power mod- ules, RF transmitter and receiver, Analog transmitter and receiver, amplifiers, filters, digital baseband, power modules, external circuitry, and other necessary com- ponents. The ESP8266 Wi-Fi module is a microchip shown in the figure below.

A set of AT commands are needed by the microcontroller to communicate with the ESP8266 Wi-Fi module. Hence it is developed with AT commands software to allow the Arduino Wi-Fi functionalities, and also allows loading various software to design the own application on the memory and processor of the module

CHAPTER 4

CIRCUIT DIAGRAM

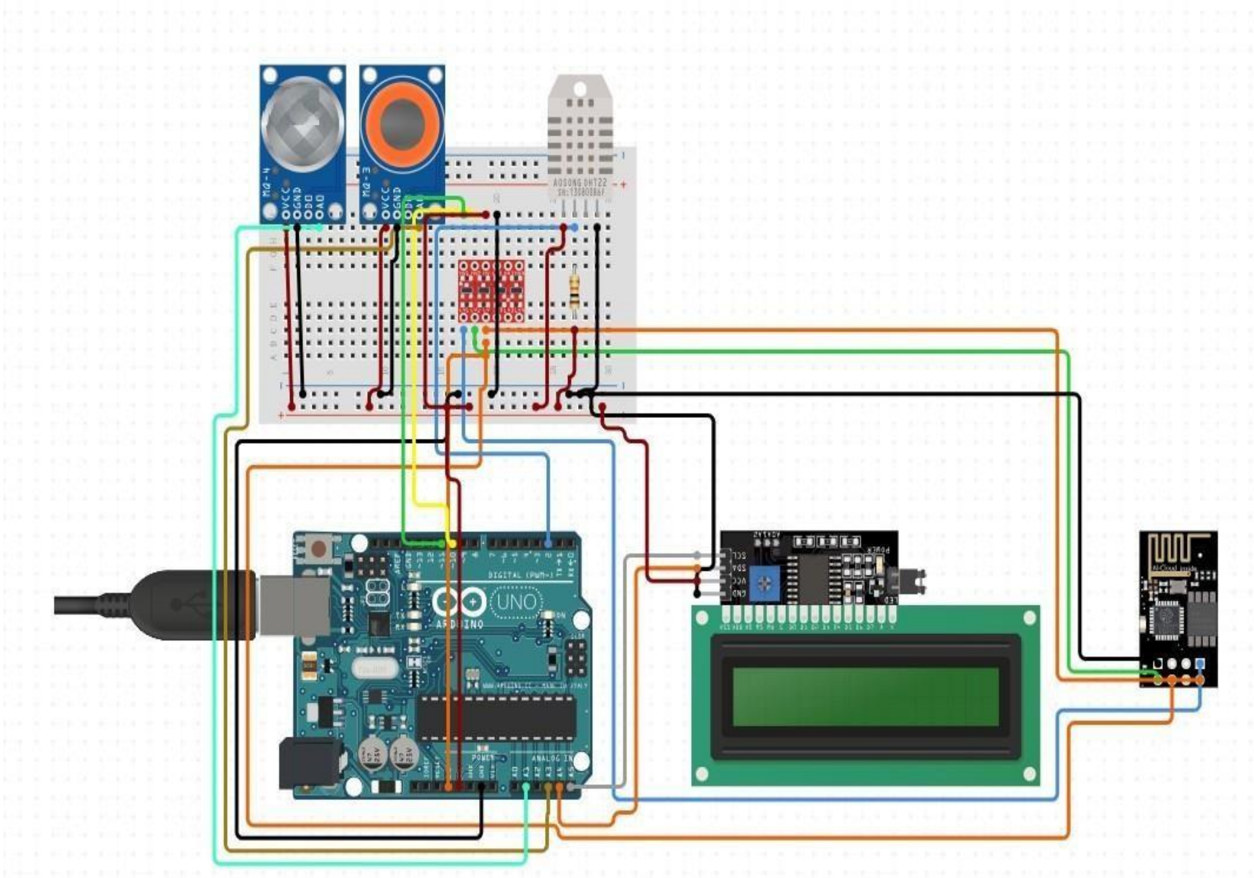

Fig 4.1 Fruit quality monitoring system

CIRCUIT EXPLANATION

MQ-3 Alcohol Sensor and MQ-4 Methane Sensor:

Connect the VCC pin of both sensors to the 5V output from Arduino.
Connect the GND pin of both sensors to the ground (GND) of Arduino.
Connect the Analog output pins (AO) of each sensor to Analog input pins (A0 and A1) of Arduino respectively.

DHT11 Temperature and Humidity Sensor:

Connect the VCC pin to 5V.
Connect the GND pin to GND.
Connect the data pin to a digital input/output pin (e.g., D2 on Arduino).

ESP8266 Wi-Fi Module:

Connect VCC to 3.3V (important: do not connect to 5V as it may damage the module).
Connect GND to GND.
Connect TX (transmit) to RX (receive) pin of Arduino (e.g., RX pin of ESP8266 to TX pin of Arduino).
Connect RX (receive) to TX (transmit) pin of Arduino.

LCD Display with I2C Interface:

Connect VCC to 5V.
Connect GND to GND.
Connect SDA to A4 (data line) on Arduino (for I2C communication).
Connect SCL to A5 (clock line) on Arduino.

Arduino Connections:

Arduino Uno to Sensors and Display:
Ensure all ground connections are common.
Connect Arduino Uno to your computer via USB for programming and power.

PROGRAM CODE:

```
#include <DHT.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
//#include <ESP8266wifi.h>
//#include <ThingSpeak.h>
#include <SPI.h>
LiquidCrystal_I2C lcd(0x27,16,2);


// WiFi settings
/*const char* ssid = "good";
const char* password = "good123";

// ThingSpeak settings const char*
thingSpeakApiKey = "api.thingspeak.com";
const unsigned long channelNumber = 2535649;*/

// Pin definitions (for sensors)
#define DHT_PIN 2       // Digital pin for DHT11 sensor
#define MQ3_ANALOG_PIN A0 // Analog pin for MQ-3 alcohol sensor
#define MQ4_ANALOG_PIN A1 // Analog pin for MQ-4 methane sensor

// Initialize DHT sensor
DHT dht(DHT_PIN, DHT11);

// Initialize LCD (I2C interface)


void setup() {
Serial.begin(9600);
dht.begin();
  SPI.begin(); // Init SPI bus
  // Initialize WiFi connection
  /* WiFi.begin(ssid, password);

  //  Wait  for  WiFi  connection       while
(WiFi.status()  !=  WL_CONNECTED)  {
delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi.");*/

  // Initialize ThingSpeak
  //ThingSpeak.begin(client); // Initialize ThingSpeak with WiFi client

  // Initialize LCD (I2C interface)
  lcd.init();
lcd.backlight();
lcd.setCursor(0, 0);
lcd.print("Smart Fruit");
```

```
lcd.setCursor(0, 1);
lcd.print("Quality
Check");   delay(2000);
}

void loop() {
  // Read temperature and humidity from DHT11
float temperature = dht.readTemperature();
  float humidity = dht.readHumidity();

// Read methane and alcohol content from sensors   int mq3Value =
analogRead(MQ3_ANALOG_PIN); // Alcohol sensor   int mq4Value
= analogRead(MQ4_ANALOG_PIN); // Methane sensor

  // Display sensor readings on
LCD   lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Temp: ");
lcd.print(temperature);
lcd.print(" C");

  lcd.setCursor(0, 1);
lcd.print("Humidity: ");
lcd.print(humidity);
lcd.print("%");
delay(3000);

  lcd.clear();
lcd.setCursor(0,0);
lcd.print("Alcohol: ");
lcd.print(mq3Value);
lcd.print("%");

  lcd.setCursor(0,1);
lcd.print("Methane: ");
lcd.print(mq4Value);
lcd.print("%");
delay(3000);

  // Determine fruit quality based on sensor readings   bool
isGoodFruit = (temperature > 20 && temperature < 35 &&
humidity > 40 && humidity < 100 &&
mq3Value < 500  && mq4Value < 400);

  // Display fruit quality result on LCD
lcd.setCursor(0, 2);

if (isGoodFruit) {
lcd.print("Good Fruit!");
Serial.print("Good Fruit");
   Serial.print(" ");
lcd.clear();
lcd.setCursor(0, 0);
```

```
lcd.print("Good Fruit");
delay(2000);

  } else {
    lcd.print("Bad Fruit!");
Serial.print("Bad Fruit");
    Serial.print("     ");
lcd.clear();
lcd.setCursor(0,   0);
lcd.print("Bad
Fruit");
delay(2000);
  }

  // Update ThingSpeak with sensor data
  /*ThingSpeak.writeField(channelNumber, 1, temperature, thingSpeakApiKey);
  ThingSpeak.writeField(channelNumber, 2, humidity, thingSpeakApiKey);
  ThingSpeak.writeField(channelNumber, 3, mq3Value, thingSpeakApiKey); // Alcohol sensor data
ThingSpeak.writeField(channelNumber, 4, mq4Value, thingSpeakApiKey);*/ // Methane sensor
data

  // Print status to Serial monitor
  Serial.print("Humdity: ");
  Serial.println(humidity);
  Serial.print("Temperature: ");
  Serial.println(temperature);
  Serial.print("Alcohol: ");
  Serial.println(mq3Value);
  Serial.print("Methane: ");
  Serial.println(mq4Value);

 delay(10000); // Delay before next reading and update (10 seconds)
}
```

CHAPTER 5

RESULT ANALYSIS

The project ’IOT BASED FRUIT QUALITY MONITORING SYSTEM’ is developed using Arduino module. We developed the code for the project to analyse the quality of the fruit with the help of the data collected from sensors which is then displayed on an LCD display. And then the values are displayed on Thingspeak platform. When certain conditions fails to satisfy, then LCD will display the message ”fruit is bad” or else when the conditions satisfy ,then the LCD will display the message ”fruit is good”. Fig 4.1 indicates the condition when the quality of fruit is good and Fig 4.2 indicates condition when the quality of fruit is bad. Fig 4.3 and Fig 4.4 indicates the quality of fruit as good or bad respectively . Fig 4.5 and Fig 4.6 shows the values collected from the sensors being displayed on Thingspeak platform.

.

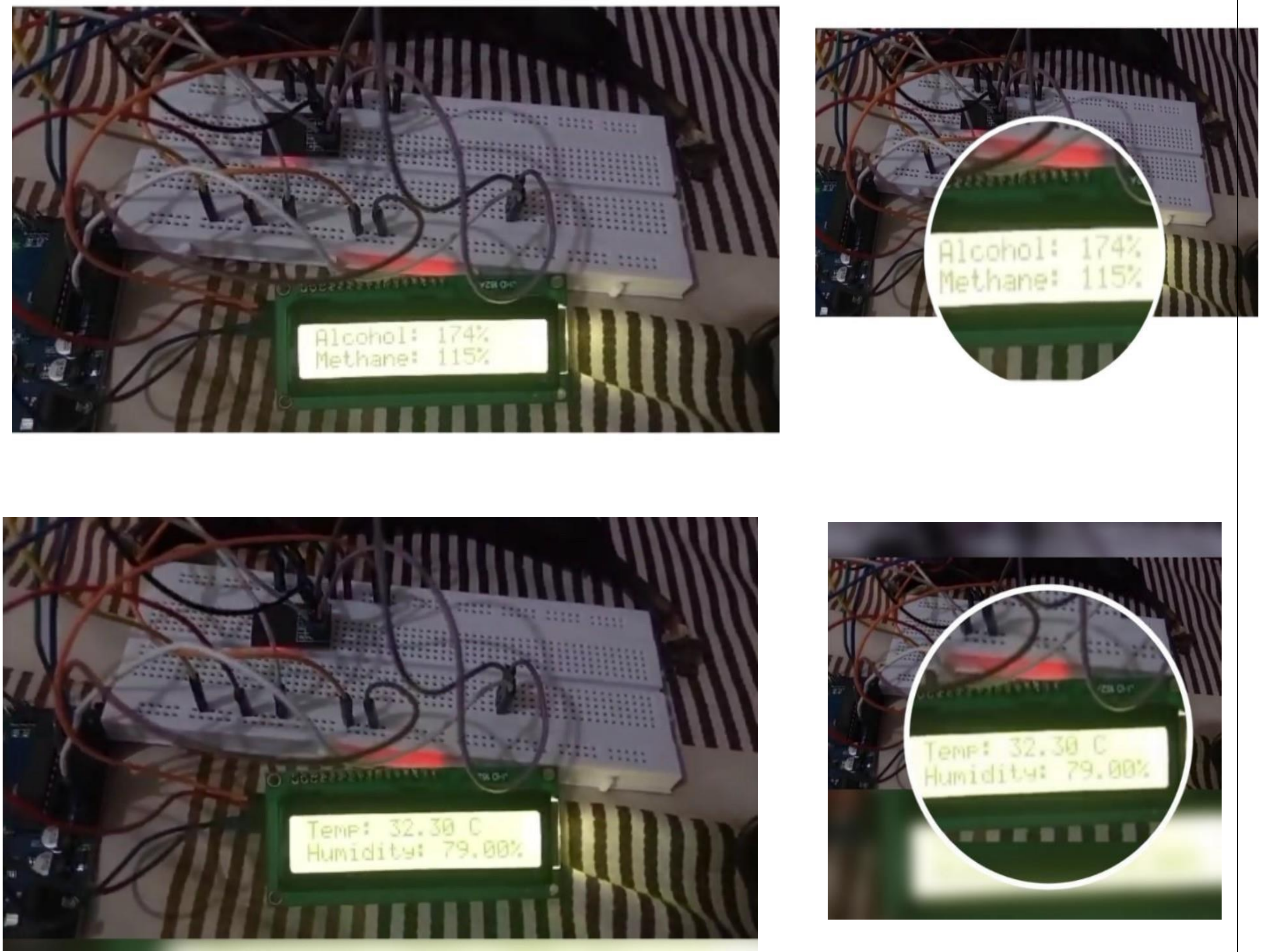

Fig. 5.1. Displaying values for bad fruit

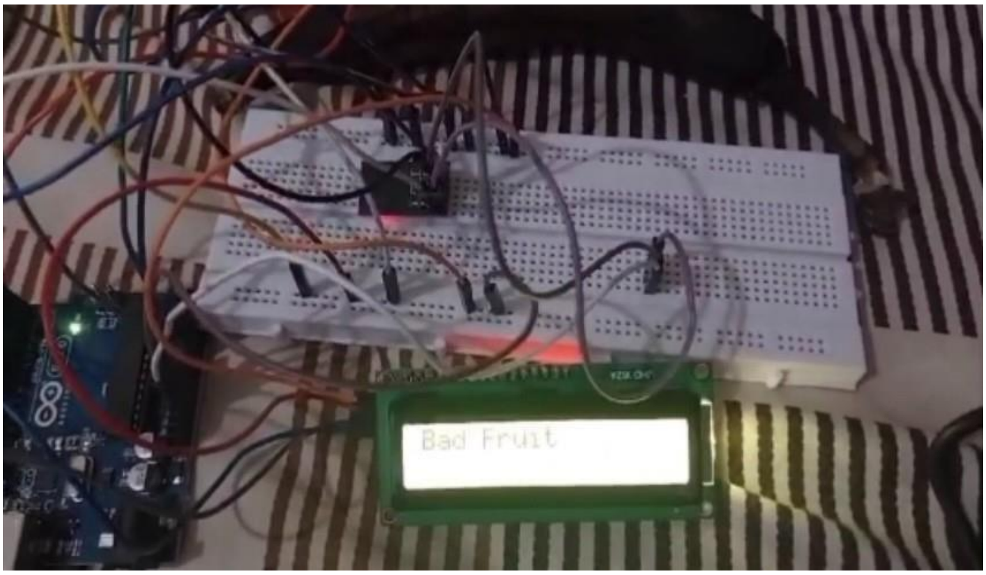

Fig. 5.2. Displaying the quality

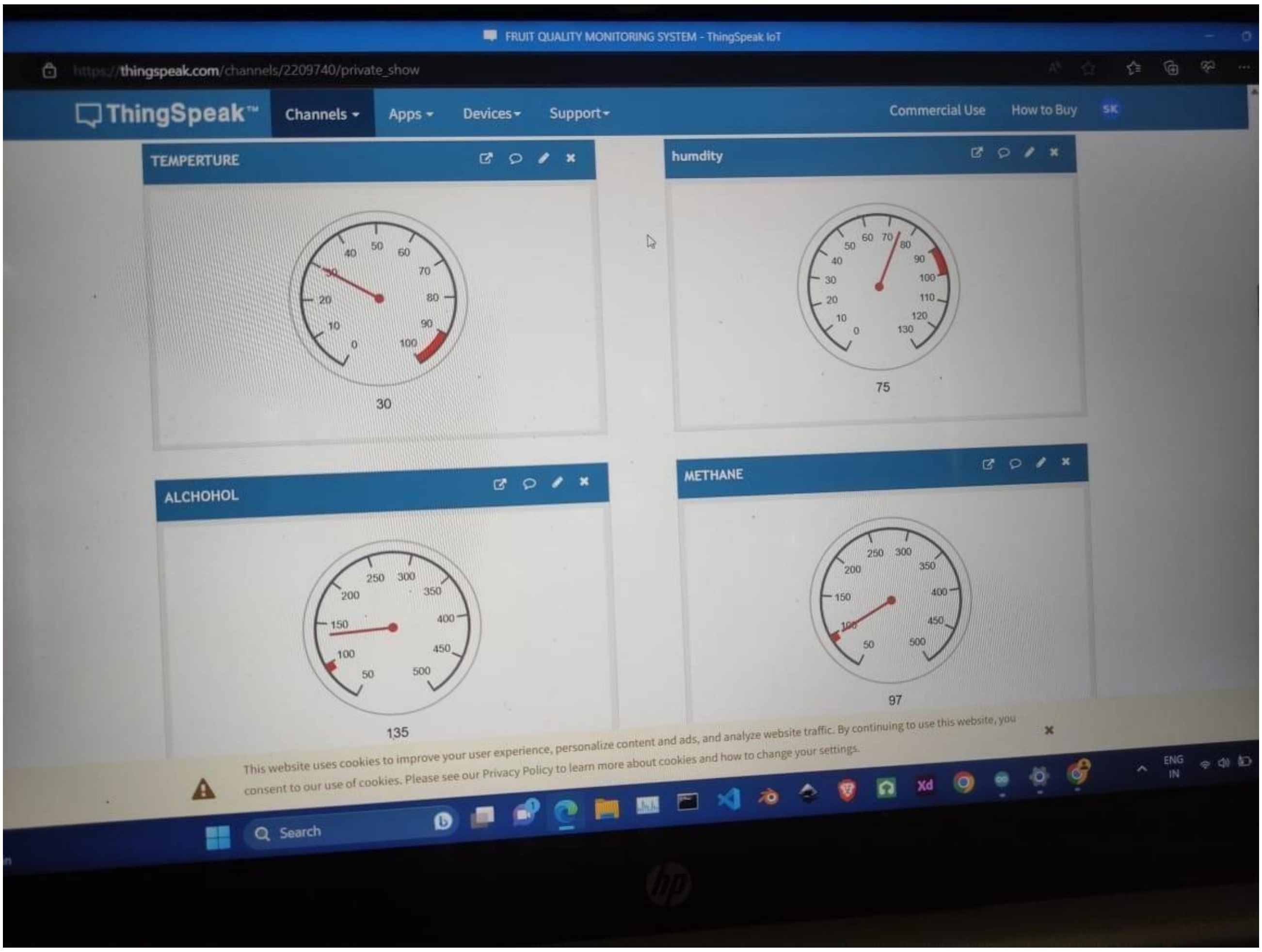

Fig. 5.3. Displaying on Thinkspeak

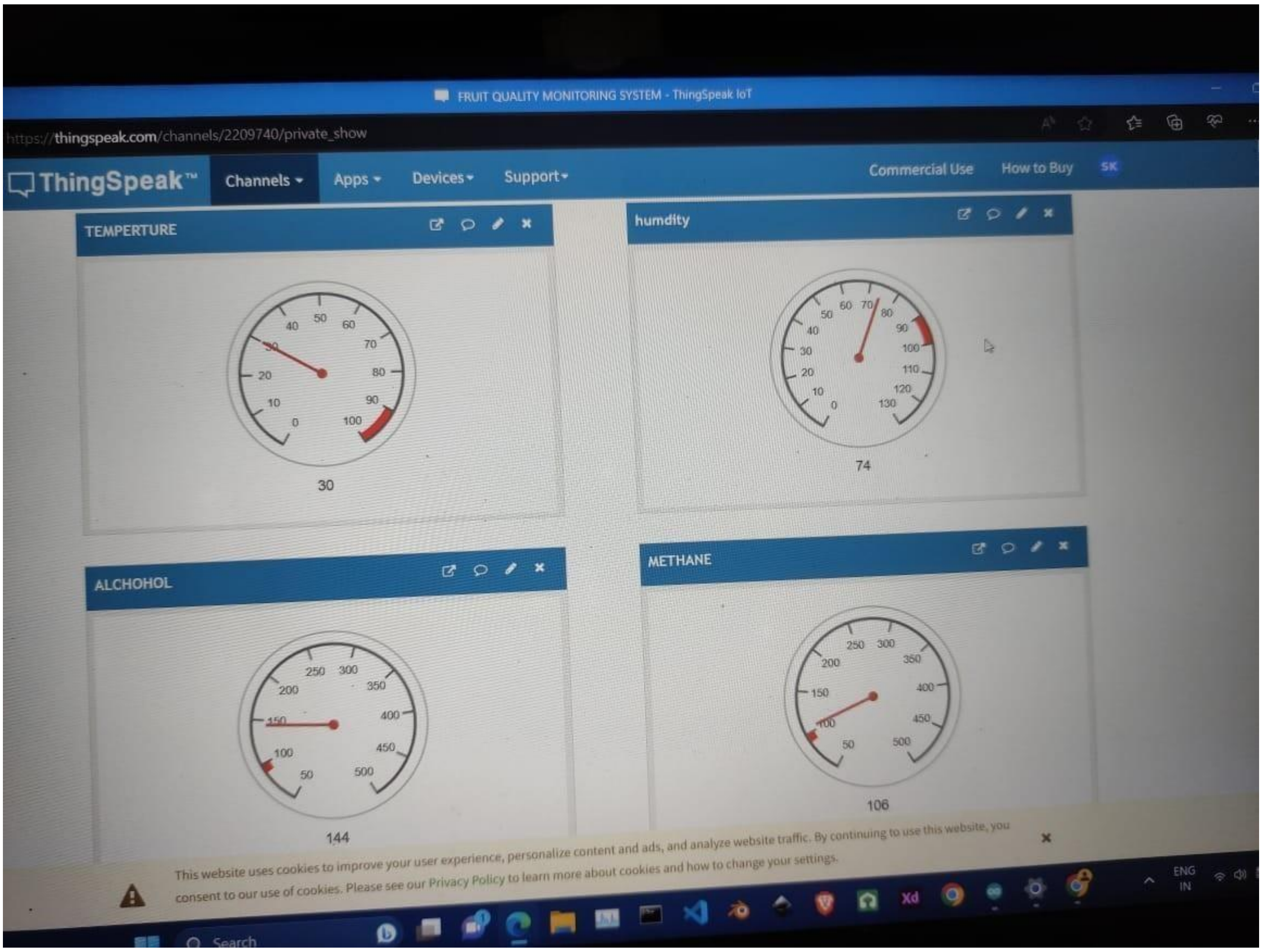

Fig. 5.4. Displaying on Thinkspeak

ADVANTAGES

1 Real-time Monitoring: IoT enables continuous monitoring of various parameters such as

temperature, humidity, and gas levels in storage facilities or during transportation. This real-time data helps in maintaining optimal conditions for fruit quality.

2 Remote Access: Stakeholders can access the data remotely via mobile apps or web interfaces, allowing for timely interventions and adjustments, regardless of location.

3 Predictive Analytics: By analyzing historical data, IoT systems can predict potential issues or quality degradation before they occur, enabling proactive measures.

4 Quality Assurance: Ensures consistent quality by maintaining ideal storage conditions throughout the supply chain, reducing spoilage and wastage.

5 Efficiency: Automation of monitoring and data collection processes reduces human error and labour costs associated with manual monitoring.

6 Traceability: IoT systems can track fruit from farm to fork, providing transparency and traceability which is crucial for food safety standards and consumer trust.

DISADVANTAGES

1. Initial Costs: Setting up an IoT infrastructure, including sensors, network - connectivity, and data management systems, can be expensive, especially for small-scale farmers or businesses.

2. Complexity: Managing and integrating IoT devices, networks, and data streams requires technical expertise. Maintenance and troubleshooting may also be challenging.

3. Data Security: IoT systems collect and transmit sensitive data, making them potential

targets for cyberattacks. Robust security measures are essential to protect against data breaches.

4. Reliability: Dependence on IoT technology means that any network disruptions or technical failures could impact data accuracy and system functionality.

5. Compatibility Issues: Integrating IoT systems with existing infrastructure or other technologies (such as ERP systems) may pose compatibility challenges.

6. Privacy Concerns: Collecting data on fruit quality may raise privacy concerns, especially if it involves personal or sensitive information about farmers or consumers.

CHAPTER 6

CONCLUSION

The food quality monitoring system is applicable in areas such as; supermarket warehouses, supermarket shelf lives, farm output food stores and warehouses, food production industries, food shipment and containers and other food storage facilities. This study has enabled us to gain knowledge on gas, temperature, humidity and light sensor technologies and how they are used. The study it has also enabled us to get experience in construction and testing of electronics. Moreover, we were able to understand the challenges faced during coming up with an idea for developing an electronic system and the best way to go about it.

The food quality monitoring system was able to read temperature and relative humidity in the food store, sense the intensity of light in the food store and detect the emission of ethanol type of gases. It was also able collect data from all the sensors and pass to LCD for display and lastly monitor the sensor data visually online.

CHAPTER 7

FUTURE SCOPE

As much as this project was able to meet the set objectives, in future works food quality monitoring system can be improved by using better sensors than can measure extreme conditions like for refrigerated vacuum-packed foods. The system can also be improved by miniaturizing and help elderly and disabled persons by alarming them if the food starts deteriorating. Further, a GSM module can be included in the system to alert the user in case there is no Internet connection.

REFERENCES

[1] C.-J. Du and D.-W. Sun, “Learning techniques used in computer vision for food quality evaluation: a review,”

[2] T. J. Ross *et al.*, “, fuzzy logic with engineering applications,” *Wiley Online Library*, vol. 2, 2004.

[3] L. Z. Y. Gu, W. Han and B. Jin, “Using iot technologies to resolve the food safety problem–an analysis based on chinese food standards,” *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 4, 2012.

[1–3]

APPENDIX

DATASHEETS

1 Arduino Uno R3 Specifications

Board	Name	Arduino Uno
	SKU	A000066
Microcontroller	ATmega328P	
USB connector	USB-B	
Pins	Built-in LED Pin	13
	Digital I/O Pins	14
	Analog input pins	6
	PWM pins	6
Communication	UART	Yes
	I2C	Yes
	SPI	Yes
Power	I/O Voltage	5V
	Input voltage (nominal)	7-12V
	DC Current per I/O Pin	20 mA
	Power Supply Connector	Barrel Plug
Clock speed	Main Processor	ATmega328P 16 MHz
	USB-Serial Processor	ATmega16U2 16 MHz
Memory	ATmega328P	2KB SRAM, 32KB FLASH, 1KB EEPROM
Dimensions	Weight	25 g
	Width	53.4 mm
	Length	68.6 mm

Table .1. Specifications for Arduino Uno