**Ensuring Food Quality based on IoT Module**

**S.Syedakbar¹,S.SankaraNarayanan²,B.Seralathan²,F.Allwin Raj²**

**¹Assistant Professor, Department of Electronics and Communication Engineering,**

**K Ramakrishnan College of Technology, Trichy-621112**

**Mail I'd:** [**syedakbar.s@krct.ac.in**](mailto:syedakbar.s@krct.ac.in)**,** [**Sankara7134@gmail.com**](mailto:Sankara7134@gmail.com)**,** [**seramaz7@gmail.com**](mailto:seramaz7@gmail.com)**,** [**allwinraj10071007@gmail.com**](mailto:allwinraj10071007@gmail.com)

**Abstract**

This project aims to develop a food quality sensor system using the Arduino platform, along with the DHT11 sensor for temperature and humidity measurement and the MQ-3 sensor for detecting alcohol vapours. The system provides a cost-effective solution for monitoring the freshness and safety of perishable food items. The DHT11 sensor is employed to measure the temperature and humidity levels inside a food storage container. By monitoring these parameters, the system can determine whether the storage conditions are suitable for maintaining the quality of the food. If the temperature or humidity deviates from the desired range, the system can trigger an alert to notify the user. The MQ-3 sensor is utilized to detect alcohol vapors, which can be an indicator of food spoilage or contamination. When certain foods undergo decomposition, they release volatile compounds such as ethanol. By incorporating the MQ-3 sensor into the system, it becomes possible to identify the presence of alcohol vapours and raise an alert if detected above a predefined threshold. The Arduino microcontroller serves as the central processing unit for collecting data from the sensors and executing the necessary algorithms. It interfaces with the DHT11 and MQ-3 sensors to retrieve sensor readings. These readings are then processed to determine the food quality status. The Arduino can be programmed to send notifications via SMS, email, or other communication methods to inform the user about any potential issues. Overall, this food quality sensor system provides a cost-effective and easily implementable solution for monitoring the freshness and safety of food items. It can be utilized in various settings,

including homes, restaurants, or food storage facilities, to ensure that the stored food remains within optimal conditions and to prevent the consumption of spoiled or contaminated items

Keywords: DTH11, MQ3, ALCOHOL VAPOUR

**INTRODUCTION**

Food quality is a crucial aspect of our daily lives, as the consumption of contaminated or spoiled food can lead to health issues. To ensure food safety, the use of food quality sensors has become increasingly important. In this context, utilizing Arduino, DHT11, and MQ-3 sensors can provide an effective solution for monitoring and maintaining the quality of food. Arduino is an open-source electronics platform widely used for prototyping and creating interactive projects. It consists of a microcontroller board and a development environment that allows users to write and upload code to the board. Arduino boards are versatile and can be easily integrated with various sensors to perform specific tasks. The DHT11 sensor is a low-cost digital temperature and humidity sensor that provides accurate readings. It can measure ambient temperature from 0 to 50 degrees Celsius with an accuracy of ±2 degrees Celsius, and humidity from 20% to 90% with an accuracy of ±5%. These features make the DHT11 sensor suitable for monitoring the environmental conditions affecting food quality. The MQ-3 sensor, on the other hand, is a gas sensor that is primarily used for detecting alcohol vapor. It can detect alcohol concentrations in the range of 0.05 to 10 mg/L. Since the presence of alcohol vapors can indicate food spoilage or contamination, the MQ-3 sensor can be employed to determine the freshness and quality of certain food products.



**Fig 1 : Food with humidity and temperature**

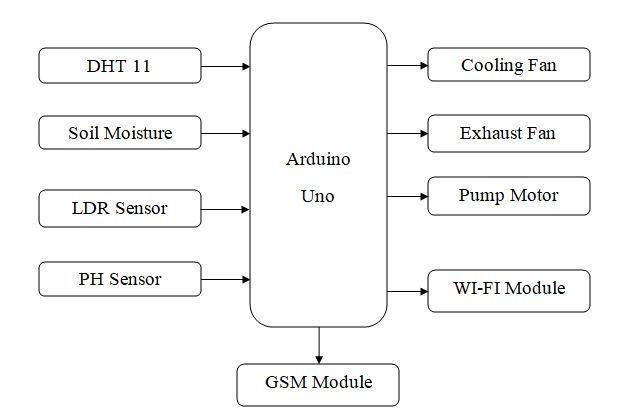
By combining the Arduino board, DHT11 sensor, and MQ-3 sensor, we can develop a food quality monitoring system. The DHT11 sensor measures temperature and humidity, while the MQ-3 sensor detects alcohol vapors. These sensor readings can be processed by the Arduino board, which can then trigger appropriate actions or provide feedback based on predefined thresholds. For example, if the temperature exceeds a certain limit or if alcohol vapors are detected above acceptable levels, the system can generate an alert or activate a cooling mechanism to preserve the food. Additionally, the collected data from the sensors can be logged and analyzed to identify patterns, optimize storage conditions, or predict potential food quality issues. In conclusion, utilizing an Arduino board, DHT11 sensor, and MQ-3 sensor can facilitate the development of a food quality sensor system. This system can monitor temperature, humidity, and alcohol vapour levels to ensure the freshness and safety of food products. By leveraging these sensors, we can take proactive measures to prevent food contamination and maintain high-quality standards. The proposed work involves developing a food quality sensor using the DTH11 sensor and MQ-3 sensor in conjunction with an Arduino microcontroller. The goal is to create a device that can assess the freshness and safety of food products based on various parameters. Hardware setup is Connect the DTH11 sensor and MQ-3 sensor to the Arduino R3 board appropriately. The DTH11 sensor will be used to measure temperature and humidity, while the MQ-3 sensor will be utilized for detecting alcohol vapor, which can indicate food spoilage. Calibration and data acquisition: Calibrate the sensors to ensure accurate measurements. Develop a program using Arduino programming language to read data from the sensors at regular intervals. This program should take into account the specific requirements of each sensor and ensure proper data acquisition. Data processing and analysis is Use the acquired sensor data to assess the quality of the food. For example, the temperature and humidity readings from the DTH11 sensor can be compared against predefined thresholds to determine if the food is stored under suitable conditions. The MQ-3 sensor data can be analyzed to identify the presence of alcohol vapors, indicating potential spoilage. Alert system implements a mechanism to alert the user if the food quality falls below acceptable levels. This can be achieved through visual indicators (such as LEDs), audible alarms, or even notifications sent to a connected device. UI is Design and develop a user interface to display the sensor readings and status. This can be a simple LCD display connected to the Arduino or a more advanced graphical interface using additional components like an OLED screen or a computer interface. Prototyping and testing is Build a functional prototype of the food quality sensor system. Test the device with various food samples under different storage conditions to evaluate its effectiveness and reliability.



**Fig2: Vapour in hot drinks**

Make necessary adjustments to the hardware or software as required. Documentation and reporting is Document the entire development process, including the circuit diagram, wiring details, and program code. Prepare a report summarizing the findings, limitations, and potential improvements of the food quality sensor system. It's worth noting that the scope of the project can be expanded based on specific requirements. For example, additional sensors can be incorporated to measure parameters like pH or specific gas concentrations for a more comprehensive analysis of food quality. To develop a food quality sensor using the MQ3 sensor and DHT11 sensor with Arduino Uno R3, you can follow the following methodology.System Design is Define the overall system requirements and functionality of your food quality sensor. Determine what parameters you want to measure, such as alcohol concentration (MQ3 sensor) and temperature/humidity (DHT11 sensor).Arduino Programming is Write the code for your Arduino board to read data from the sensors and process it. This can be done using the Arduino IDE or any other compatible programming environment. Here's a basic example to get you started is UI. If desired, you can create a User Interface to display the sensor readings or provide an output to the user. This can be done using an LCD display, LEDs, or even a graphical interface on a computer or mobile device.Testing and Optimization is Test your food quality sensor system with various food samples and scenarios to ensure its accuracy and reliability. Make any necessary adjustments to the hardware or software to optimize the performance.Remember to consult the sensor datasheets, Arduino documentation, and relevant resources for detailed information on the sensors and their usage with Arduino. Additionally, feel free to customize and expand upon this methodology based on your specific project requirements.

**METHODOLOGY**



**Fig 3: Block diagram of food quality sensor using arduino**

Arduino UNO: The Arduino UNO is a microcontroller board based on the ATmega328. One of the most widely used prototype boards is it. The board already has an embedded Arduino boot loader. It has 6 PWM pins, 6 Analog inputs, 14 GPIO pins, an on-board resonator, a reset button, mounting holes for pin headers, an on-board resonator, and on-board UART, SPI, and TWI interfaces. The board can be linked to a PC via a USB port and powered by USB while being programmed. The 32 KB Flash, 1 KB EEPROM, and 2 KB SRAM of the Arduino UNO are all expandable. For networking via Ethernet, Bluetooth, Wi-Fi, Zigbee, or Cellular networks, the board can be connected to several Arduino Shields. It can also be connected to the majority of IoT platforms.

**Calibration:** Calibrate the sensors to establish a baseline for normal conditions. This involves exposing the sensors to known environments. The NTC temperature sensor (or Thermistor) and the humidity sensing component are the two primary components of the DHT11 sensor. Thermistors actually function as variable resistors, changing their resistance in response to variations in temperature. Both of them detect the local temperature and humidity, then transmit their findings to the IC (located on the sensor's rear side). The four pins on the sensor are VCC, Ground, data Out, and NC. The common VCC and Grounds are connected to, respectively, the VCC and Ground pins. The sensor's Data Out pin is linked to the Arduino board's PB0 pin. The MQ3 sensor, which uses SnO2 as its sensitive material because of its decreased conductivity in clean air, is used to detect the presence of ethanol. As the amount of ethanol gases in the atmosphere grows, so does its conductivity.Both an analog and a digital sensor, MQ-3. Food that contains ethanol vapours has deteriorated. Therefore, the MQ3 sensor can determine whether food has begun to rot. The sensor has four pins: Ground, VCC, Analog Out, and Digital Output. The common VCC and Ground are connected to the VCC. The digital output pin is left unconnected because it is not in use. The analog output pin, which is attached to the sensor, provides the output. LDR Sensor: The LDR detects the level of light intensity. The sensor is attached to the Arduino board's A1 pin connected to the output. The sensor is wired up in a circuit with a potential divider. The built-in ADC converts the analog voltage provided by the LDR into a digital reading.16X2 LCD - To connect the 16X2 LCD display to the Arduino board, connect its data pins to pins 2 through 5. The Arduino board's pins 10 and 9 are connected to the LCD's RS and E pins, respectively. The LCD's RW pin is grounded. The ESP8266 Wi-Fi Module is a self-contained SOC that has an integrated TCP/IP protocol stack and can connect to a wireless network. The ESP8266 is capable of offloading all Wi-Fi networking tasks to another application processor or hosting an application. An AT command set firmware is pre-programmed into each ESP8266 module. There are two models of the module available: ESP-01 and ESP-12. ESP-01 only has 8 usable pins while ESP-12 has 16 accessible for interface.

**Data acquisition:** Install an Internet of Things (IoT) gadget based on Arduino in a grocery store. It begins reading data from the interfaced sensors, including the MQ3 Sensor, LDR Sensor, and DHT-11 Temperature and Humidity Sensor, once it has been correctly installed and turned on. A digital sensor with an integrated capacitive humidity sensor and thermometer is the DHT11 Temperature and Humidity Sensor. Every two seconds, it transmits a real-time temperature and humidity reading. The sensor can read temperatures between 0°C and 50°C and relative humidity levels between 20% and 95% while operating on a 3.5 to 5.5 V supply. 80 microseconds of LOW followed by 80 microseconds of HIGH were wed. The pin is set to digital after transmitting the start signal. The start signal consists of a LOW for 18 milliseconds, a HIGH for 20 to 40 microseconds, a LOW once more for 80 microseconds, and a HIGH once more for 80 microseconds. The pin is set to digital output after receiving the start signal, and 40-bit data including the temperature and humidity reading is latch-out. The first two bytes of the five-bit data are an integer and a decimal reading of the relative humidity, the third and fourth bytes are an integer and a decimal measurement of the temperature, and the final byte is a checksum byte.

A standard library for the DHT-11 sensor is already available for Arduino. Calling the read11() function of the DHT class will quickly prepare the data from the sensor. The LDR sensor inputs a voltage at the controller's analog input pin using a potential divider circuit connection. ADC channel incorporated into the device is used to read and digitize the voltage.

The MQ3 sensor picks up on ethanol-related gas emissions. Food and fruits release ethanol-type fumes when they begin to spoil. Such gases are detected using the MQ3 sensor, which produces an analog voltage proportional to the gas concentration. The Arduino's analog pin receives the analog output and converts it to a digital value using an internal ADC. The Arduino gathers information from every sensor and transforms the values into strings. The character LCD receives the sensor data wrapped as appropriate strings and displays it. The data is uploaded to ThingSpeak Server by the Arduino and the ESP8266 Wi-Fi module. Either a digital dashboard or a data broker are required for displaying and managing data that has been submitted to the ThingSpeak server. In this project, the sensor data is visually tracked online using a digital dashboard called Freeboard.io. The data from ThingSpeak is shown using JASON files by Freeboard.io. It provides three components for creating a dashboard.

1) Data Sources - Data sources obtain their information from outside sources. These external sources could be JavaScript programs, JSON files that get material from an HTTP server, or data broker services. A JSON file that gets data from the ThingSpeak server serves as the project's data source.

2) Widgets - The Widgets support the textual or graphical display of data. Freeboard.io offers a wide variety of widgets, including text, graphs, gauges, and more.

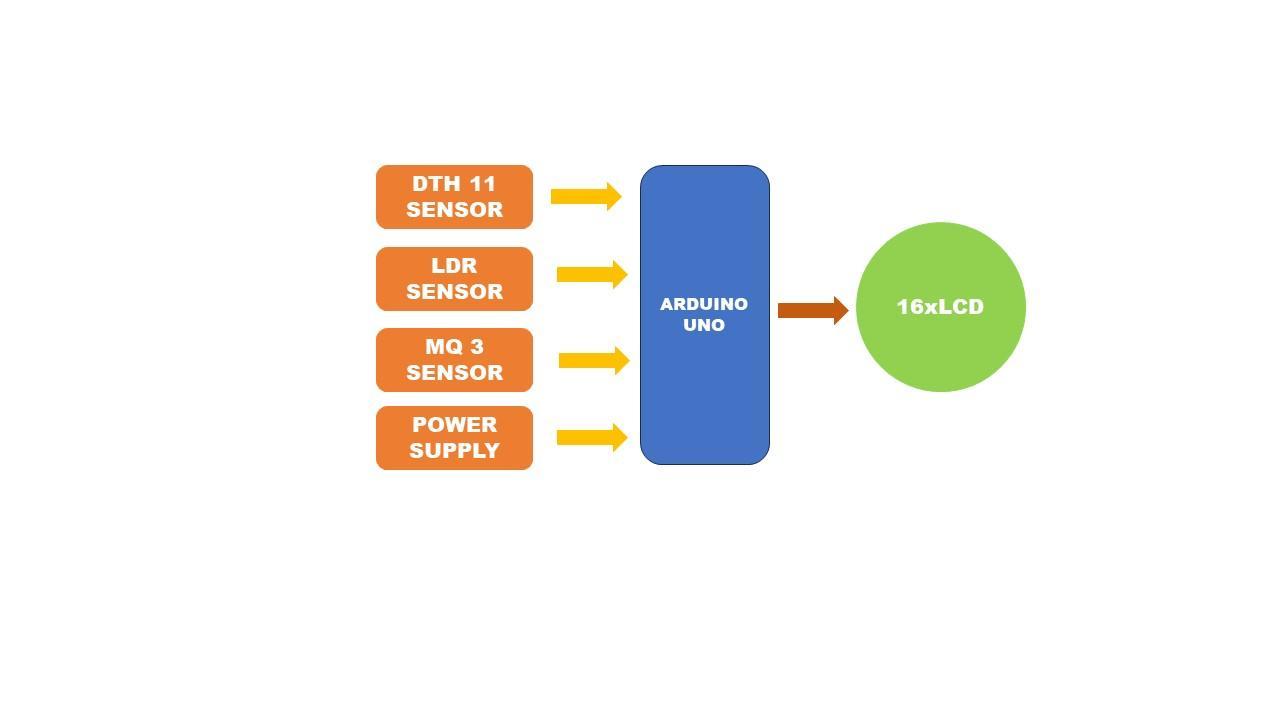
3) Panes - Widgets are arranged using this.

**Data processing and analysis:** The Arduino program is able to gather data from the MQ3, DHT-11, and LDR sensors, stringify the sensor values, display the strings on a character LCD, and send the data to the ThingSpeak server. Before importing the software Serial library for serial connection with the Wi-Fi module, the standard open-source Arduino library for interfacing DHT11 must first be loaded. Initialization is done on the variables that represent the pin connections used to read sensor data. When the setup() function is used, the serial communication's baud rate is set to 9600 for use in communication with the Wi-Fi modem. With some delays, the AT commands are used to establish the Wi-Fi mode and network connectivity. According to how long it takes to connect to the network, the delay should be indicated. The serial communication's baud rate is set to 9600 when the setup() function is used to communicate with the Wi-Fi modem. The Wi-Fi mode and network connexion are established via AT instructions, but with significant delays. The delay should be displayed based on how long it takes to connect to the network. The AT commands for establishing a TCP connection are submitted to the esp8266() function, and then the ThingSpeak API key is supplied to transmit data to the designated channel. The data is updated on the ThingSpeak channel every 16 seconds.

**Alert system:** The sensors used in the food quality are specialized for their own uses for example DTH11 used for humidity and temperature sensing of the food . The MQ3 Sensor used to sense the gas particle and vapour presented in closed container food.These values will be displayed in the LCD display

It's important to note that the specific implementation details may vary depending on the requirements and constraints of your project. Additionally, consider incorporating additional sensors or features as needed to enhance the food quality monitoring capabilities.

Here the proposed method



**Fig 4 : Proposed Method**

Creating a food quality sensor using the MQ-3 alcohol gas sensor and the DHT11 temperature and humidity sensor without using a NodeMCU involves interfacing these sensors with a microcontroller or microprocessor and implementing a suitable algorithm to analyze the data. Below is a general methodology to help you get started:

Methodology:

The MQ-3 alcohol gas sensor, the DHT11 temperature and humidity sensor, and a NodeMCU can be used to create a food quality sensor by connecting these sensors to a microcontroller or CPU and creating an appropriate algorithm to interpret the data. Listed below is a generic process to get you started:

Components required are:

1. Alcohol Gas Sensor MQ-3

2.DHT11 Temperature and Humidity Sensor

3. Microcontroller (such as the Raspberry Pi and Arduino Uno)

4. Jumper wires and a breadboard

5. Power supply (batteries or USB power, for example)

Methodology:

Sensor Connections:

Attach the MQ-3 sensor to the correct microcontroller pins. Since the MQ-3 sensor normally provides an analog output,and used to sense the gas volume of the food in closed container and attach its analog pin to one of the microcontroller's analog input pins. Attach the DHT11 sensor to a digital device which is used to sense humidity and temperature of the food in closed container. LDR sensor are the component which is used sense the data in light or open sunlight to show value of the food

Power Supply:

Power up the MQ-3 and DHT11 sensors simultaneously. Be sure to provide consistent power and adhere to each sensor's voltage specifications.

MQ-3 calibration:

For the MQ-3 sensor to convert the analog output to alcohol concentration, calibration is necessary. The sensor must be exposed to a known alcohol concentration, and the analog output must be recorded. Create a calibration curve that connects sensor output to alcohol concentration using the data from this step.

Data Collection:

Utilize the analog-to-digital converter (ADC) of the microcontroller to read analog output from the MQ-3 sensor. Utilizing the calibration curve, convert this analog number to an alcohol concentration.Utilize digital communication protocols to read the DHT11 sensor's data on temperature and humidity.

Data Processing:

Assess the quality of the food using the collected data. For instance, temperature and alcohol concentration are crucial aspects to take into account. Given that some foods are sensitive to moisture, you might also wish to take humidity levels into consideration.

Alerts and Thresholds:

Set parameters that indicate what constitutes acceptable food quality, such as alcohol content, temperature, and humidity. Alerts should be set off if any of these numbers fall outside of the allowed range.

Alert System:

Implement an alert system, such as LED blinking, a buzzer sounding, or email or text message notifications.

(Optional) User interface To display readings and alerts in real time, you can optionally link a display (such an LCD) to the microcontroller.

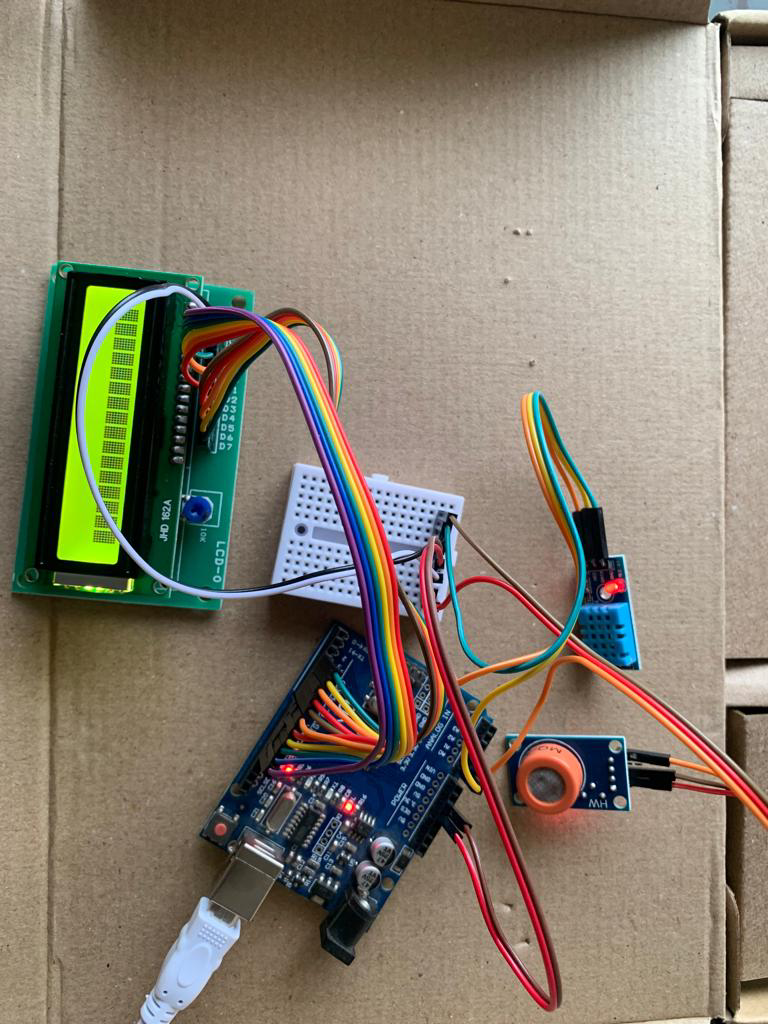
Evaluation and Improvement:

To ensure accurate readings and pertinent notifications, test the sensor system with a variety of foods and environmental factors while checking food quality by dth 11 and mq3 sensors inorder to prevent from any disease causes.

Power Control (Battery-Powered Configuration):

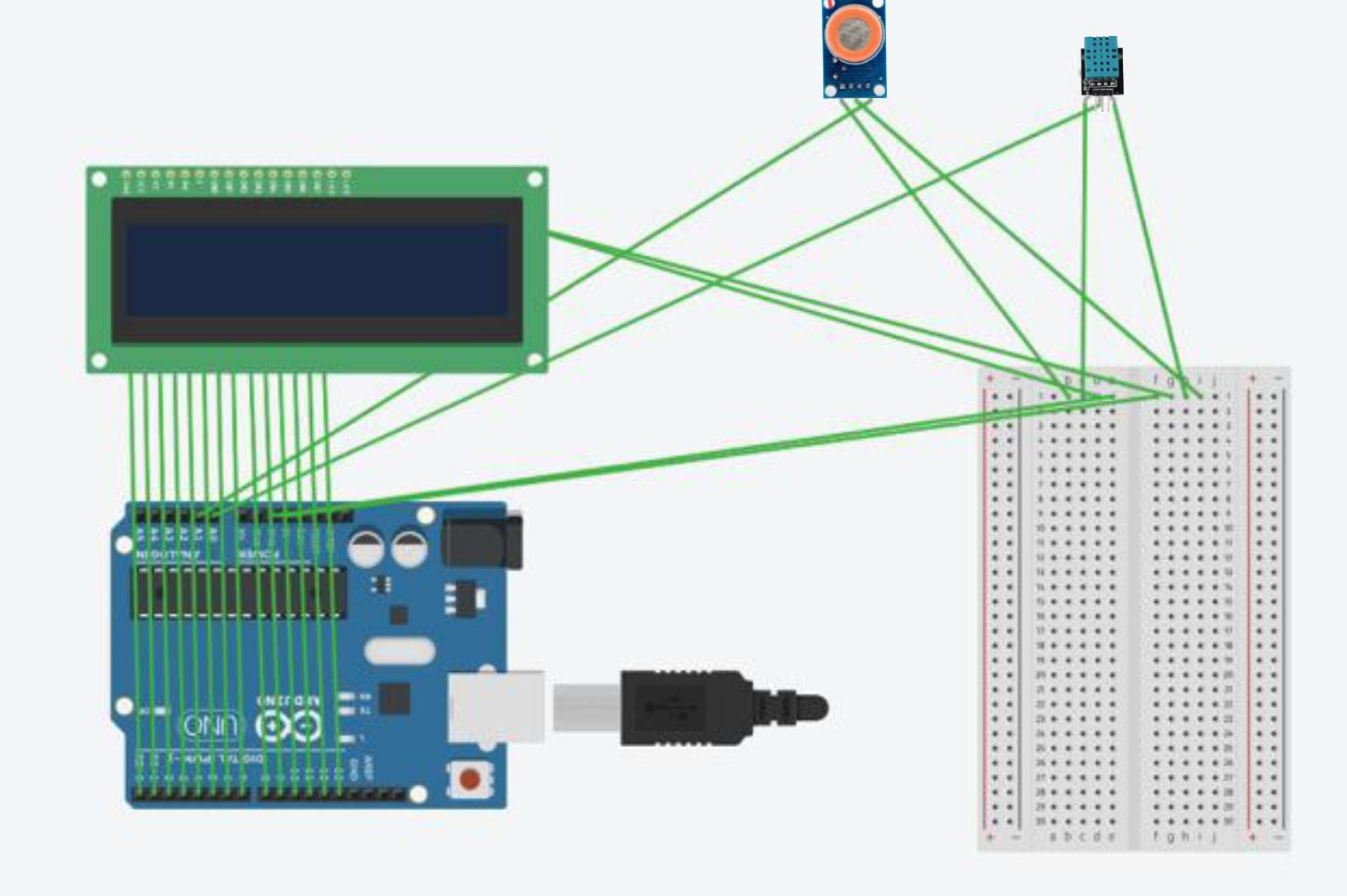
Use power-saving techniques to increase battery life if you're using batteries. This can entail sleeping the microcontroller and waking it up to take readings at set intervals.

**RESULTS AND DISCUSSION**



**Fig 5: Hardware Module of Proposed Methodology**

We, the undersigned, hereby declare the development of a Food Quality Sensor utilizing the DHT11 and MQ-3 sensors. This sensor aims to enhance the assessment and monitoring of food quality parameters, including temperature, humidity, and alcohol content, in various food products. The purpose of this sensor is to provide a reliable and cost-effective solution for individuals and businesses involved in the food industry to ensure the safety and freshness of food products. By accurately measuring and monitoring critical parameters, the sensor will assist in preventing spoilage, detecting contamination, and promoting food safety.Additionally, the MQ-3 sensor, based on gas-sensing technology, is integrated into the system to measure alcohol content. This feature is particularly useful for assessing the freshness and quality of fermented food and beverages. The MQ-3 sensor can detect the presence of alcohol vapor in the surrounding environment, allowing for real-time monitoring and ensuring compliance with quality standards.The Food Quality Sensor system is designed with the following key features as follows Accurate measurement of temperature, humidity, and alcohol content, Real-time monitoring and display of sensor readings, Alarm notifications for critical threshold breaches, Data logging capability for analysis and record-keeping, User-friendly interface for easy operation and configuration,Compact and portable design for versatile application.The result of the food quality sensing system using Arduino Uno R3, DHT11 sensor, MQ-3 sensor, and an LCD display will depend on the specific implementation and the thresholds set for determining the food quality.

****

**FIG 6: SCHEMATIC VISULATION**

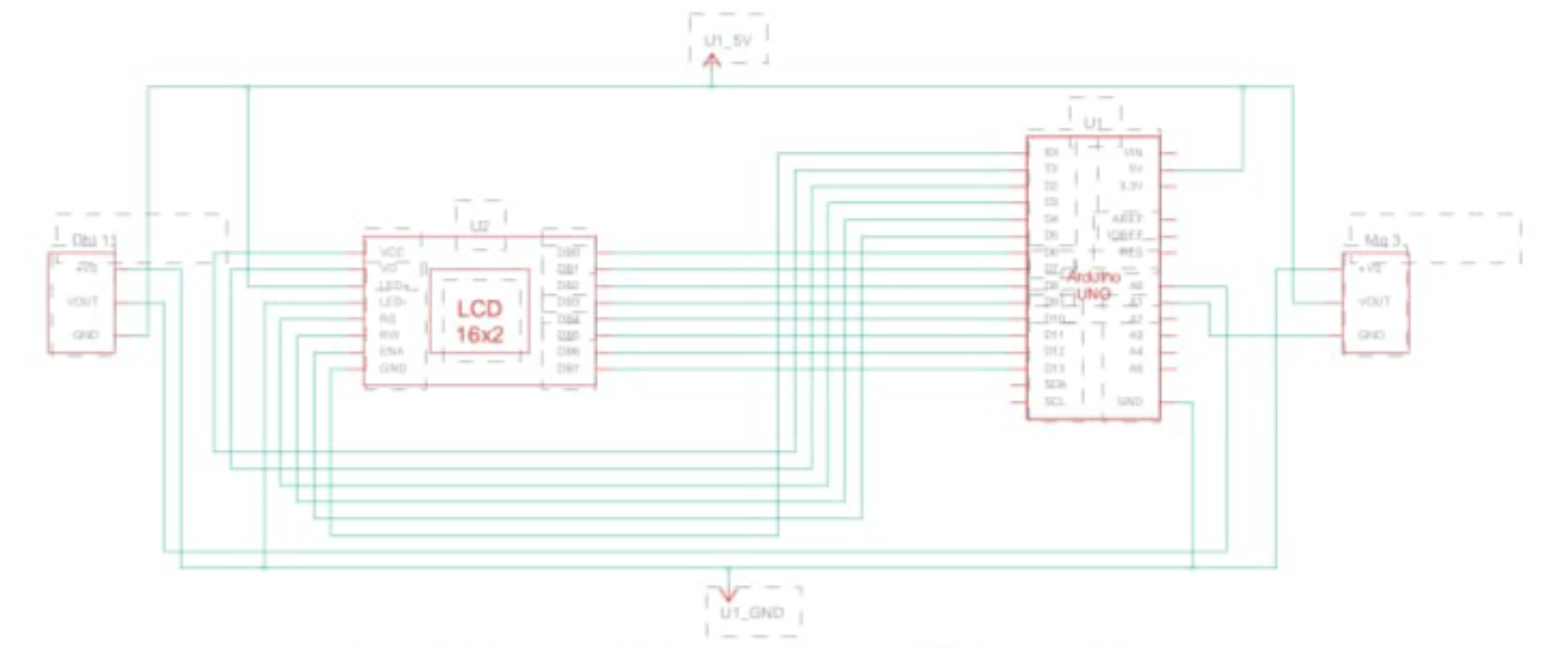
In general, the system can provide the following information.

**Temperature and Humidity:** The DHT11 sensor measures temperature and humidity. The system can display the current temperature and humidity values on the LCD display.

**Gas Detection:** The MQ-3 sensor measures alcohol and other gases. It can detect the presence of gases that may indicate food spoilage or contamination. The system can display the gas value on the LCD display.

**Threshold Violations:** The system can compare the temperature and gas values with predefined thresholds. If the temperature exceeds a certain threshold, it can display a warning message on the LCD display indicating high temperature. Similarly, if the gas value exceeds a specified threshold, it can display a warning message indicating the presence of high gas levels.

**Quality Assessment:** Based on the temperature and gas readings and the threshold violations, the system can provide an overall assessment of the food quality. For example, if the temperature and gas values are within acceptable ranges, it can display a message indicating that the food quality is "OK." If any threshold is violated, it can display a warning message indicating a potential issue with the food quality.



**FIG 7: CIRCUIT DIAGRAM**

Remember that the specific implementation and interpretation of the results may vary depending on your requirements and the algorithms you use to analyze the sensor data. The example code provided earlier in the conversation can serve as a starting point for developing your own food quality sensing system using the mentioned components.

**CONCLUSIONS**

Food wastage is one of the crucial crises in the world. One of the main reasons for food wastage is improper warehouse management and this is a solvable problem to an extent with the current technological advancement. Over referring to different researches and solutions to this problem, we have come to a realization that the field of IoT can provide a very efficient solution to this problem. Therefore, we have discussed a food quality monitoring system based on IoT that will control different environmental factors such as light intensity, humidity and temperature that are necessary to be maintained at a threshold value to prevent the food from spoilage. It also provides a user friendly through an app where they can monitor the light intensity parameters and at the same time get alerts when the food is spoiled or if there is a fire hazard. The combination of Arduino R3, DHT11 sensor, MQ3 sensor, and an LCD display provides a robust and efficient solution for food quality sensing. The Arduino R3 acts as the central processing unit, receiving data from the DHT11 sensor for temperature and humidity monitoring, as well as data from the MQ3 sensor for detecting the presence of harmful gases.The DHT11 sensor accurately measures temperature and humidity, allowing for precise monitoring of the food environment. The MQ3 sensor plays a crucial role in detecting harmful gases, such as carbon dioxide or ammonia, which could indicate spoilage or contamination. By continuously monitoring the air quality surrounding the food, potential health hazards can be identified early, allowing for timely interventions to prevent foodborne illnesses.The LCD display serves as a user-friendly interface, providing real-time feedback on temperature, humidity, and gas levels. It enables easy interpretation of the sensor data, making it convenient for both professionals and consumers to assess the quality and safety of food.Overall, the integration of these components offers a comprehensive food quality sensing system, empowering individuals and businesses to take proactive measures to maintain food freshness, prevent spoilage, and ensure consumer safety. The Arduino R3, DHT11 sensor, MQ3 sensor, and LCD display together provide an effective solution for food quality monitoring in various settings, such as homes, restaurants, and food processing facilities.

**REFERENCES**

1. Ajay Doltade, Ankita Kadam, Sayali Honmore, Sanjeev Wagh, “Intelligent Grain Storage Management System based on IoT”, International Journal of Science and Research (IJSR), March 2019,Vol. 8 Issue 3.
2. Amrita Srivastava, Ankita Gulati, “ITrack: IoT framework for Smart Food Monitoring System”, International Journal of Computer Applications, August 2016, Vol. 148 - No.12.
3. Atkare Prajwal, Patil Vaishali, Zade payal, Dhapudkar Sumit, “Food Quality Detection and Monitoring System”, IEEE International Student’s ‘Conference on Electrical, Electronics and Computer Science 2020
4. Bin Yu, Ping Zhan, Ming Lei, Fang Zhou & Peng Wang, “Food Quality Monitoring System based on Smart Contracts and Evaluation Models”, IEEE, 13 Jan 2020, Vol. 8.
5. Fu Ying, Li Fengquan, “Application of internet of things to the Monitoring System for Food Quality safety”, 2013 fourth international conference on digitalmanufacturing & automation, 16 September 2013.
6. Jun Liu, Yue Shen, Cheng’an Wang, Jiahui Li, “Design of Fruits Warehousing Monitoring and Control System Based on Wifi ”, IEEE, 2015.
7. Dr. Keshavmurty, Megaha shree.M, Meghamala M, Mariyam Steffi J, “Automatized Food Quality Detection & System using Neural Network”, 4th international conference on recent trends on electronics, information, communication & technology, 18 May
8. Maen Taruri, Abubakar, Noora Alnaqbi, Hessa Al Shehhi, Abdul Halim M Jallad & Amine Bermak , “AMachine Learning Approach Monitoring using a DoFP Sensor”,IEEE, 17 Aug 2020, Vol. 8
9. Meo Vincent C.Caya,Febus Reidj G.Cruz,Carolle Marian N.Fernando,Wen yaw chung, “Monitoring and Detection of Fruits & Vegetable Spoilage in the Refrigerator using Electronic Nose Based on Principal Component Analysis”, IEEE, 15 Jun 2020.
10. Naveed Shahzad, Usman Khalid, Atif Iqbal, Meezan-Ur- Rahman, “eFresh – a Device to Detect Food Freshness”, International Journal of Soft Computing and Engineering (IJSCE), September 2018, Vol. 8 Issue 3.
11. Ravi Chander B, P.A. Lovina & G. Shiva Kumari, “Food Quality Monitoring System by using Arduino”, Journal of engineering sciences, Apr 2020, Vol. 11 Issue 4.
12. Venkatesh.A, Saravanakumar.T, Vairamsrinivasan S, Vigneshwar A, Santhosh Kumar M, “A Food Monitoring System Based on Bluetooth Low Energy and Internet of Things”, International Journal of Engineering Research and Application, March 2017,Vol. 7 Issue 3.