Innovative Food Safety

Introducing AI Technology

OBJECTIVE

Developing an innovative technology for food safety and quality assurance by leveraging blockchain, IoT, and AI to monitor production, transportation, and storage processes in real-time, minimizing contamination risks, reducing waste, and ensuring consumer health and food supply chain integrity.

Environmental condition

- Proper storage conditions play a crucial role in food preservation.
- Maintain suitable temperature and humidity levels to control decomposition.
- Temperature Danger Zone : Keep storage temperatures between 4°C to 60°C.
- Bacteria multiply rapidly within this range.
- Refrigerate perishable foods promptly.

- Humidity: Aim for 50-55% humidity in food storage areas.
- Prevents food from drying out or becoming too moist.
- Helps maintain food quality.

Food safety **solutions.**

Less food wastage

Improved quality of food will reduce food wastage

Freshness tracking

Methane released by food in ppm

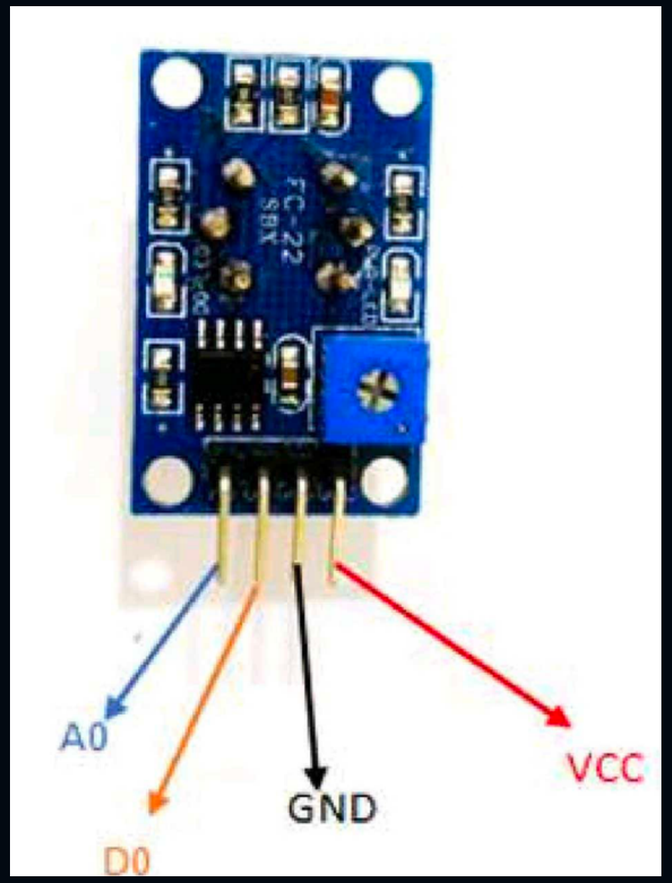
Water Contaminantion detection

ideantifies moisture content in food

pH detection

Tests the impact of acidic or basic nature of food

MQ4 Methane Gas Sensor



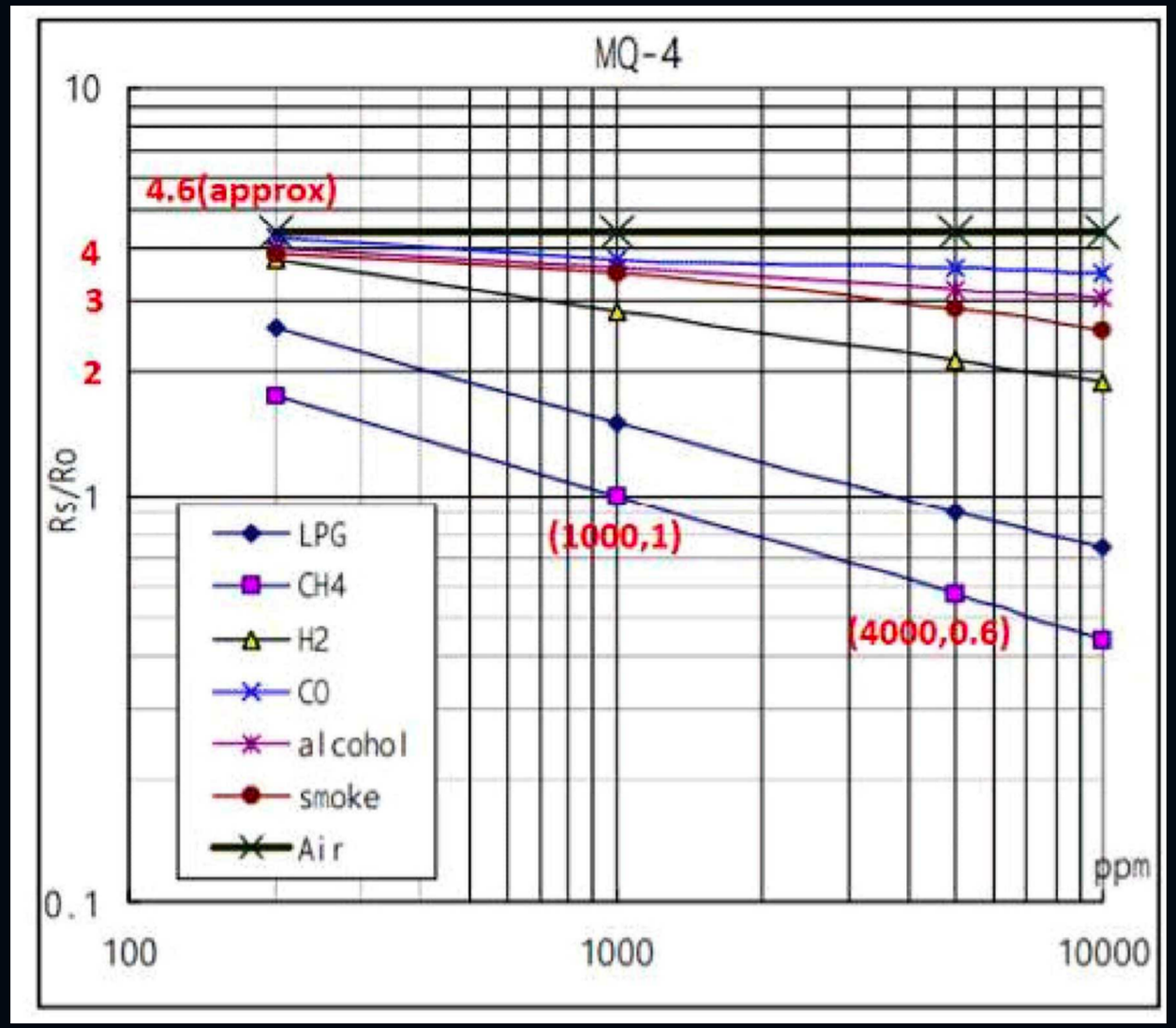
- MQ4 is a gas sensing module, which is used to measure methane gas in the atmosphere. It contains Gas sensing layer, which is made up of SnO2.
- SnO2 is sensitive to gases like LPG, CH4, H2, CO, Alcohol, and smoke. As the decaying food emits methane gas (CH4), the MQ4 sensor can be used to measure this gas monitor food quality.

MQ4 arduino Programming

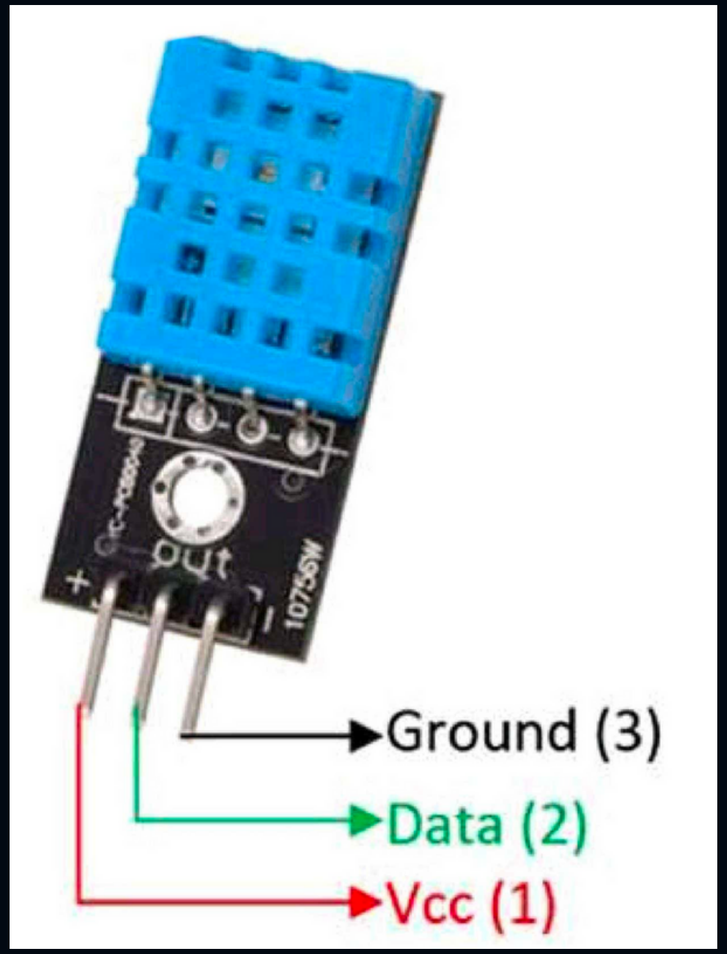
```
tabnine: test | fix | explain | document | ask
void setup() { Serial.begin(9600); //Baud rate

}
tabnine: test | fix | explain | document | ask
void loop() {
float sensor_volt; //Define variable for sensor voltage
float RS_air; //Define variable for sensor resistance
float R0; //Define variable for RO
float sensorValue; //Define variable for analog readings
for (int x = 0; x < 500; x++) //Start for loop
{
sensorValue = sensorValue + analogRead(AB); //Add analog values of sensor 500 times
}
sensorValue = sensorValue / 500.0; //Take average of readings
sensor_volt sensorValue * (5.0/1023.0); //Convert average to voltage
RS_air ((5.0 * 10.0) / sensor_volt) 10.0; //Calculate RS in fresh air
R0=RS_air / 4.6; //Calculate RO
Serial.print("R0 = "); //Display "RO"
Serial.println(R0); //Display value of RO
delay(1000); //Wait 1 second
}

```

MQ4 Output Graph

It Shows the typical sensitivity characteristics of the MQ-4 for several gases. Temp: 20°C. Humidity: 65%. Concentration 21% RL=20kΩ

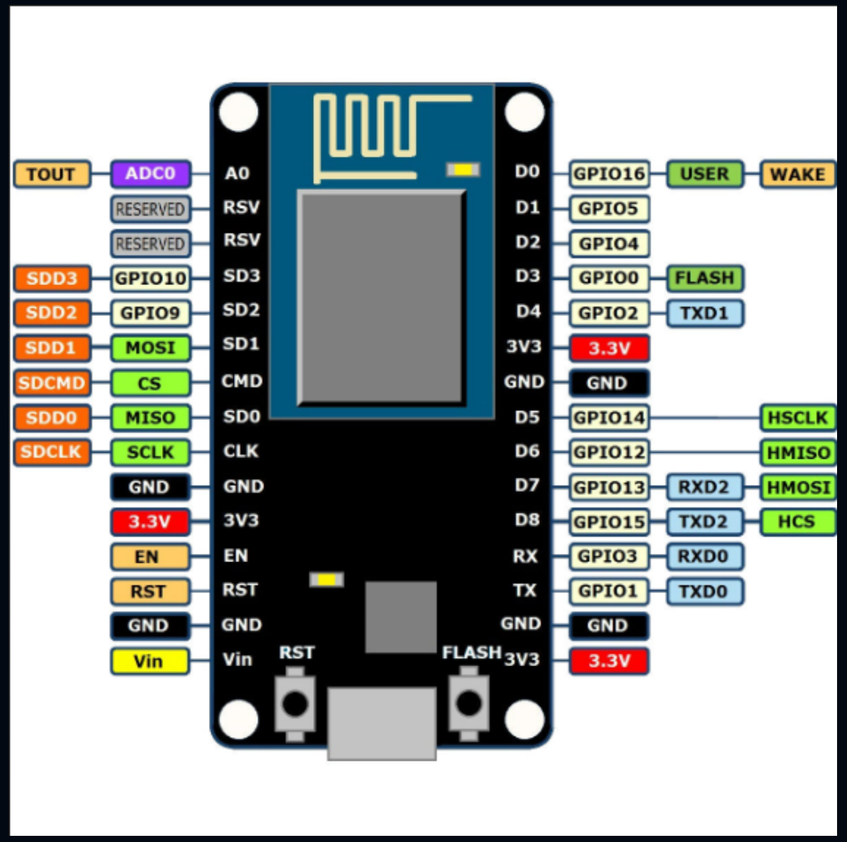


DHT11 Sensor



- The DHT 11 is a low-cost digital sensor used to measure temperature (from 0°C to 50°C with an accuracy of 2°C) and relative humidity (from 20% to 80% with a great accuracy)
- It provides calibrated digital output and is commonly interfaced with microcontrollers like Arduino

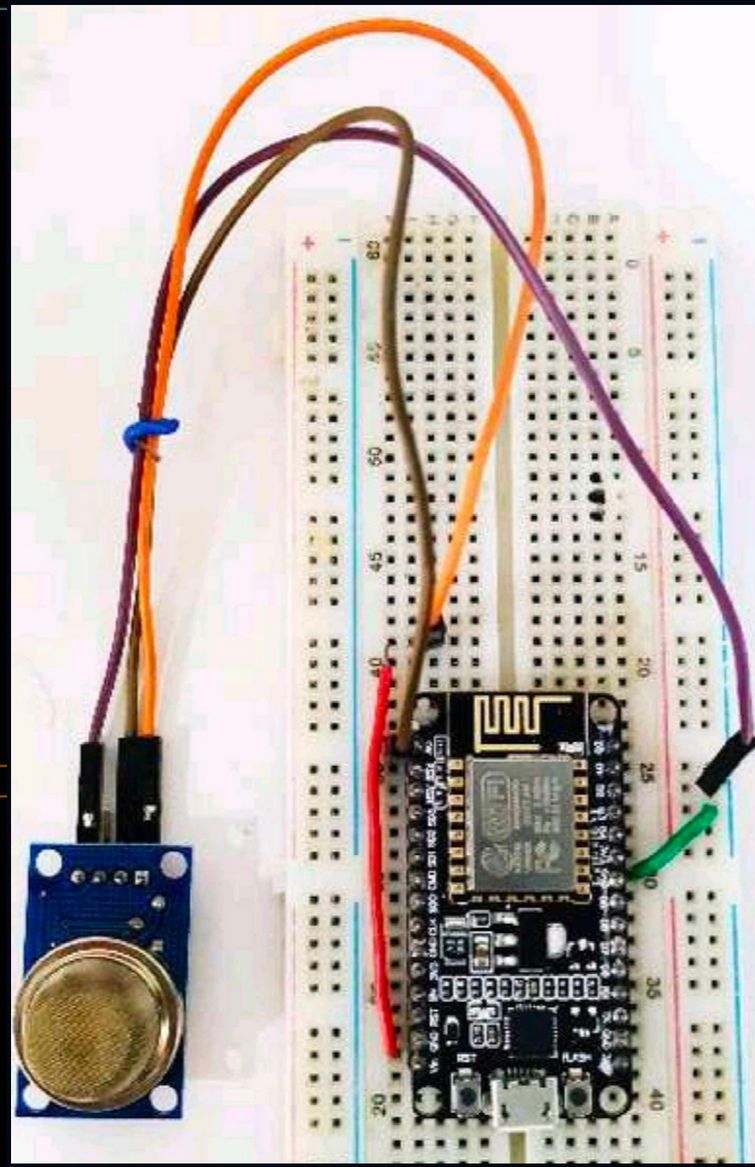
NodeMCU ESP8266



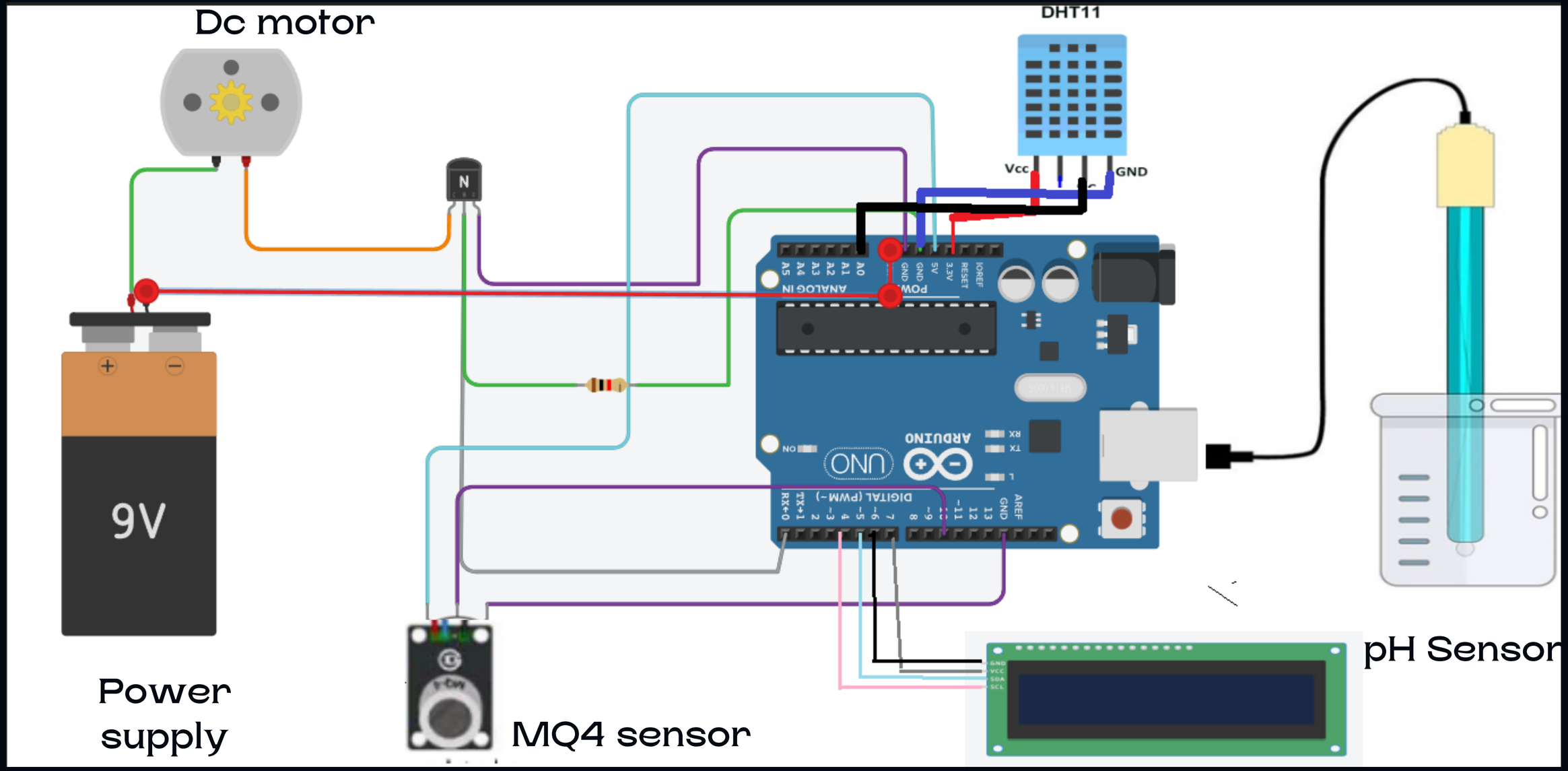
- NodeMCU is used for prototyping and building IoT devices.
- It allows easy integration of Wi-Fi capabilities into various projects.
- Common applications include home automation, sensor networks, and smart devices.

NodeMCU ESP8266

```
11  void setup()
16  {
17  delay(500);
18  Serial.print(".");
19  }
20  Serial.println(WiFi.localIP());
21  server.on("/", []()
22  {
23    page = "<html><head><title>IoT Design</title></head><style type=\"text/css\">";
24    page += "table{border-collapse: collapse;}th {background-color:  green ;color: white;}table,td {border: 4px solid
      black;font-size: x-large;";
25    page += "text-align:center;border-style: groove;border-color: rgb(255,0,0);}</style><body><center>";
26    page += "<h1>Smart Aquaculture Monitoring using IoT</h1><br><br><table style=\"width: 1200px;height: 450px;\"><tr>";
27    page += "<th>Parameters</th><th>Value</th><th>Units</th></tr><tr><td>PH Value</td><td>"+String(data1)+"</td><td>N/A</
      td></tr>";
28    page += "<tr><td>Temperature</td><td>"+String(data2)+"</td><td>Centigrade</td></tr><tr><td>Moisture</td><td>"+String
      (data3)+"</td><td>%</td>";
29    page += "<meta http-equiv=\"refresh\" content=\"3\">";
30    server.send(200, "text/html", page);
31  });
32  server.begin();
33  }
    tabnine: test | explain | document | ask
34  void loop()
35  {
36    StaticJsonBuffer<1000> jsonBuffer;
37    JsonObject& root = jsonBuffer.parseObject(Serial);
38    if (root == JsonObject::invalid())
39    {
40      return;
41      Serial.println("invalid");
42    }
43    data1 = root["a1"];
44    data2 = root["a2"];
45    data3 = root["a3"];
46    Serial.println(data1);
47    Serial.println(data2);
48    Serial.println(data3);
49    server.handleClient();
50  }
```

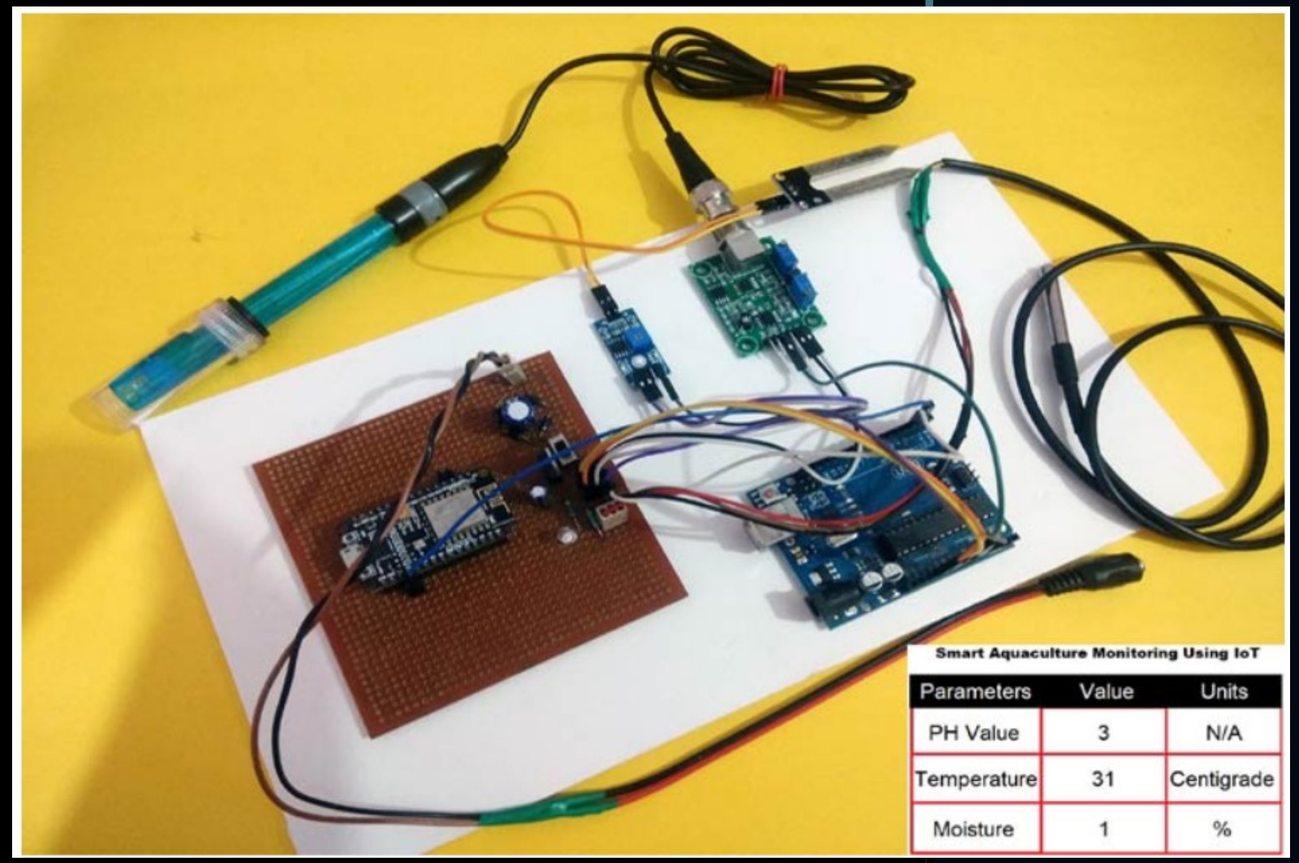


Circuit Diagram



AI to check pH of food

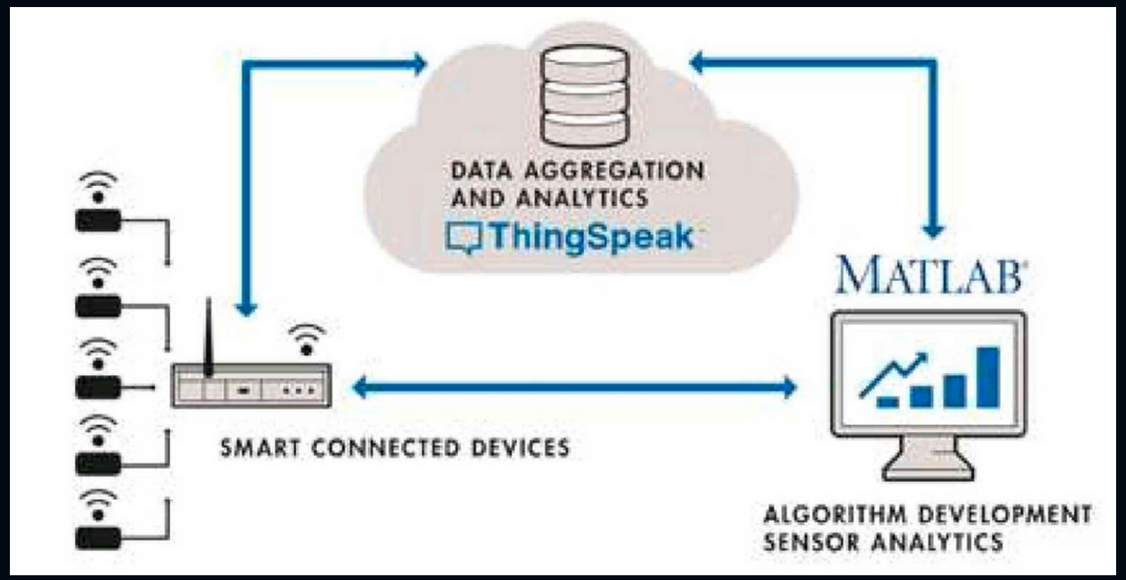
```
11  void setup()
14    temp_sensor.begin();
15  }
16  StaticJsonBuffer<1000> jsonBuffer;
17  JsonObject& root = jsonBuffer.createObject();
    tabnine: test | explain | document | ask
18  void loop() {
19    for (int i = 0; i < 10; i++)
20    {
21      buffer_arr[i] = analogRead(A0);
22      delay(30);
23    }
24    for (int i = 0; i < 9; i++)
25    {
26      for (int j = i + 1; j < 10; j++)
27      {
28        if (buffer_arr[i] > buffer_arr[j])
29        {
30          temp = buffer_arr[i];
31          buffer_arr[i] = buffer_arr[j];
32          buffer_arr[j] = temp;
33        }
34      }
35    }
36    avgval = 0;
37    for (int i = 2; i < 8; i++)
38      avgval += buffer_arr[i];
39    float volt = (float)avgval * 5.0 / 1024 / 6;
40    float ph_act = -5.70 * volt + calibration_value;
41    temp_sensor.requestTemperatures();
42    int moisture_analog=analogRead(A1);
43    int moist_act=map(moisture_analog,0,1023,100,0);
44    root["a1"] = ph_act;
45    root["a2"] = temp_sensor.getTempCByIndex(0);
46    root["a3"] = moist_act;
47    root.printTo(Serial);
48    Serial.println("");
```



Smart Aquaculture Monitoring Using IoT

Parameters	Value	Units
PH Value	3	N/A
Temperature	31	Centigrade
Moisture	1	%

Thingspeak Server



- ThingSpeak is a cloud data service that allows you to collect, store, and analyze data from Wi-Fi-connected devices.
- It integrates seamlessly with MATLAB, enabling real-time analysis and visualization.
- You can send data from any internet-connected device directly to ThingSpeak using REST APIs or MQTT.
- It's commonly used for IoT prototyping and analytics

Food Quality detection by AI

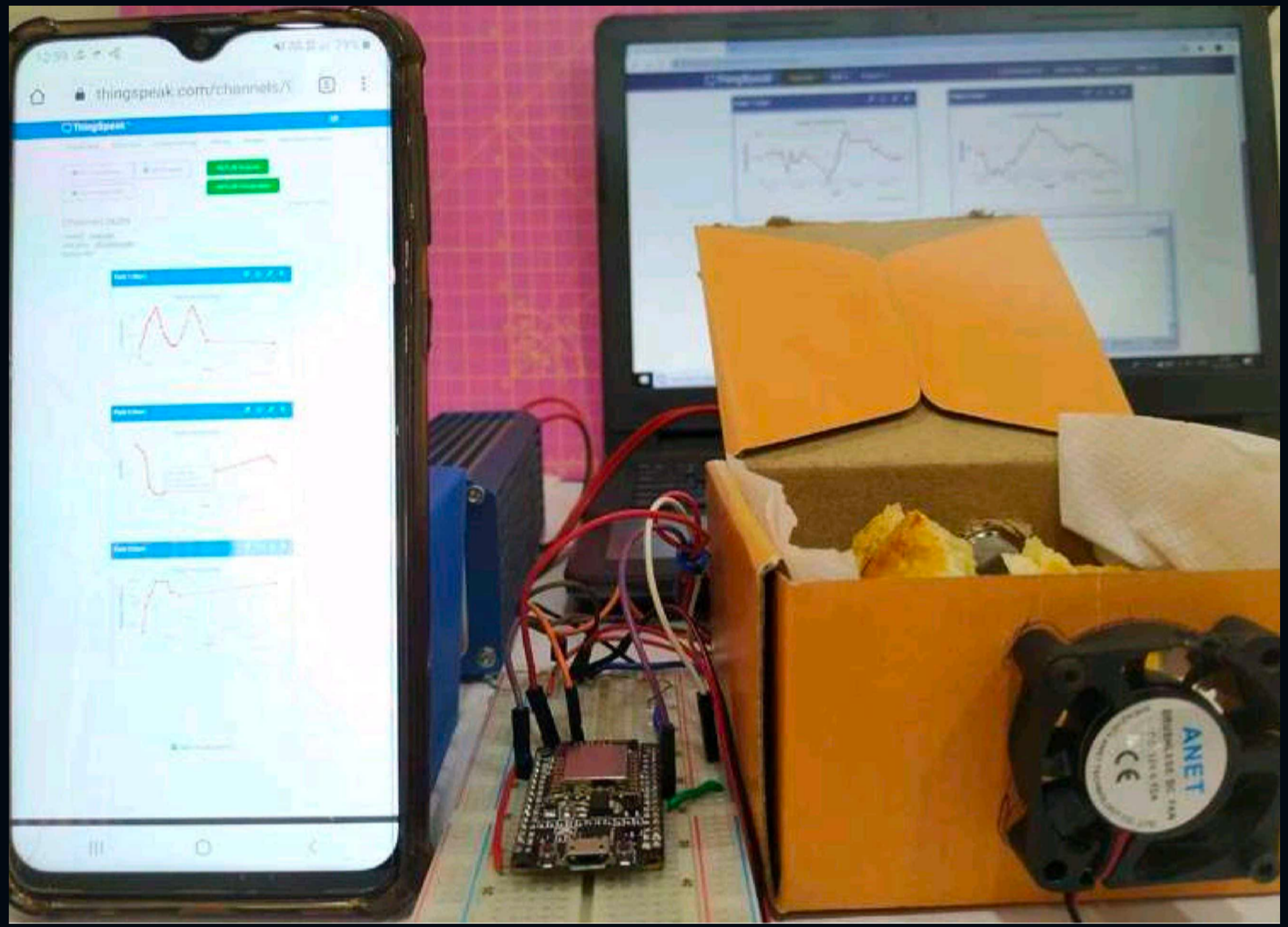
```python
from sklearn.linear_model import LinearRegression

tabnine: test | explain | document | ask
def collect_data():
    methane_ppm = float(input("Enter methane ppm: "))
    humidity = float(input("Enter humidity (%): "))
    temperature = float(input("Enter temperature (°C): "))
    ph_factor = float(input("Enter pH factor: "))
    return methane_ppm, humidity, temperature, ph_factor

methane_ppm, humidity, temperature, ph_factor = collect_data()

methane_threshold = 50
humidity_threshold = 70
temperature_threshold = 10
ph_threshold = 7.0  # pH level considered safe

is_safe = (methane_ppm < methane_threshold) and (humidity < humidity_threshold) and (temperature < temperature_threshold)
and (ph_factor > ph_threshold)

if is_safe:
    print("Food is safe.")
else:
    print("Food is not safe.")

X = [[methane_ppm, humidity, temperature, ph_factor]]
y = [0]

model = LinearRegression()
model.fit(X, y)

y_pred = model.predict(X)

print("Prediction:", y_pred[0])
```

Integrating Cloud API with microprocessor

```
57  void loop() {
64  delay(2000);
65  ThingSpeak.writeField(myChannelNumber, 2,h, myWriteAPIKey);
66  Serial.print("Humidity = ");
67  Serial.println(h);
68  gas = ppm1();
69  delay(2000);
70  if (t > 22){
71    digitalWrite(D0,HIGH);
72    send_event("temp_event");
73    Serial.println("Fan On");
74  }
75  else{
76    digitalWrite(fan,LOW);
77  }
78     if (isnan(h) || isnan(t)|| isnan(gas)) {
79     Serial.println("Failed to read from DHT sensor!");
80      }
81  }
    tabnine: test | explain | document | ask
82  void send_event(const char *event)
83  {
84    Serial.print("Connecting to ");
85    Serial.println(host);
86    // Use WiFiClient class to create TCP connections
87    WiFiClient client;
88    const int httpPort = 80;
89    if (!client.connect(host, httpPort)) {
90      Serial.println("Connection failed");
91      return;
92    }
93    // We now create a URI for the request
94    String url = "/trigger/";
95    url += event;
96    url += "/with/key/";
97    url += privateKey;
98    Serial.print("Requesting URL: ");
99    Serial.println(url);
```

```
82  void send_event(const char *event)
83  {
84    Serial.print("Connecting to ");
85    Serial.println(host);
86    // Use WiFiClient class to create TCP connections
87    WiFiClient client;
88    const int httpPort = 80;
89    if (!client.connect(host, httpPort)) {
90      Serial.println("Connection failed");
91      return;
92    }
93    // We now create a URI for the request
94    String url = "/trigger/";
95    url += event;
96    url += "/with/key/";
97    url += privateKey;
98    Serial.print("Requesting URL: ");
99    Serial.println(url);
100   // This will send the request to the server
101   client.print(String("GET ") + url + " HTTP/1.1\r\n" +
102                "Host: " + host + "\r\n" +
103                "Connection: close\r\n\r\n");
104   while(client.connected())
105   {
106     if(client.available())
107     {
108       String line = client.readStringUntil('\r');
109       Serial.print(line);
110     } else {
111       // No data yet, wait a bit
112       delay(50);
113     };
114   }
115   Serial.println();
116   Serial.println("closing connection");
117   client.stop();
118 }
```

A sample test run to check food quality



Food Safety Using AI

Topic 1: AI and food safety

Artificial intelligence ensures food safety

Topic 2: Faster detection

AI can detect food contamination much faster

Topic 3: Lower risk of outbreaks

Reduced risk of food poisoning outbreaks

Topic 4: Integration of Cloud API

AI can monitor food safety in real-time on my phone

Future Prospect

Collaborating with Culinary Experts:

- Partnering with chefs and food specialists can provide us with qualitative insights that extend beyond mere statistics.
- Delving into Advanced Algorithms, such as deep learning, could reveal fresh data patterns.

Our Vision:

- We aim to merge data science with culinary arts, transforming predictive modeling into a creative aspect of cooking.
- By enhancing our models, we strive to provide chefs and food enthusiasts with tools that enhance dishes and inspire innovation in the culinary realm.

Team Labyrinth Legends
Presents you

The Fresh Palate

Thank You