# EXPERIENTIAL ENGINEERING EDUCATION FABRICATION / MODEL DEVELOPMENT

**REPORT ON**

## (WATER POLLUTION AND POLLUTANT MONITORING)

*A Report submitted*

***by***

M. SAI NIKETH,20951A04F5

P. VIJAYVARDHAN,20951A04P2

B. SAHITH,20951A04E4

A.SAI KOUSHIK,20951A04F2

IARE-New-Logo_Black.png

Name of the Department

## INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad – 500 043, Telangana

June, 2022

EXPERIENTIAL ENGINEERING EDUCATION FABRICATION / MODEL DEVELOPMENT

**Title of your Idea : WATER POLLUTION AND POLLUTANTS MONITORING.**

**Thrust Area / Sector : EMBEDDED SYSTEMS AND ENVIRONMENT**

**Branch & Section : ECE-B**

**Year / Semester :2nd,4-SEMESTER**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Name of the Student** | **Roll Number** | **Mobile Number** |
| **1** | M. SAI NIKETH | 20951A04F5 | 9640927942 |
| **2** | A. SAI KOUSHIK | 20951A04F2 | 7993884866 |
| **3** | B. SAHITH | 20951A04E4 | 9866279402 |
| **4** | P.VIJAY VARDHAN | 20951A04P2 | 8555064070 |

**Abstract of your Idea:**

* We have designed our own DIY IOT Based Water pH Meter & Pollutant Monitoring using pH Sensor&ESP8266 WiFi Module.
* In this project, we used an advanced pH Sensor, which gives analog linear pH readings in 0pH to 14pH range. We will send the data to both the Serial monitor & Thing Speak Server.
* The Serial Monitor provides the pH readings directly from NodeMCU giving us the pH values and pollutants present at that pH level on the Laptop Screen via the Arduino application.
* And ThingSpeak provides instant visualizations of data posted by our device (i.e ESP8266). Since ESP8266 is a 12-bit controller with built-in ADC, so it can measure data precisely.
* So, the total abstract of our project is about the pH Monitoring the pH levels of Water Bodies present in our Surroundings, and identifying the harmful and edible pollutants present in the Water Bodies.
* Making a Report on that Water Bodies on their Conditions totally based upon the readings given by the pH Sensor.
* And Further, we will look into, whether that water body needs treatment or not, by taking suitable steps. And consult experts for further developments.

**Objectives & Significance:**

* The objective of this project is to measure the pH levels of water bodies in the surrounding areas and their present condition. so, that they can be used for the people around it or need to be treated and make it suitable for aquatic life and general activities for the local people.
* So, we take the readings collected from the sample of that water body, where the values are read from the pH Meter.
* Based on the readings or values we have set some statements in the programmable code of the NodeMCU. By the commands given to the NodeMCU, the pH Values and Pollutants present at that pH Level are displayed.
* By this way, we made a report based on the synopsis given by the pH Meter and came to some scientific assumptions.
* The Scientific significance of our project is unique and built on the combination of embedded components, PCBs (printed circuit boards), and microcontrollers.
* Our project is significant because it is based on the concept of ADC (Analog to Digital converters) and the response is displayed on IOT Based applications and also in Arduino applications.
* In this project, an advanced pH sensor kit has been used, in which a sensitive pH

electrode has been used for the accurate readings. With a pH sensor for processing the

voltage variations given by the pH electrode.

* Those values are directed to the microcontrollers for further, processing of the values

and displaying the respective values.

**Background of the Idea:**

* The background of our idea is taken from the environmental pollution, in which our project is based on the presence of contaminants and pH levels of water bodies present in our surroundings.
* The quality of water plays a crucial role in the health of animals and human beings, water quality monitoring of the water bodies requires a lot of effort and should be checked the entire lake.
* Water quality checking is a difficult process with the old equipment, where the monitoring is totally based on the wires connectivity system and the values are not accurate, all these drawbacks can be overcome by the new technology-based IoT water quality monitoring.
* pH value plays a major role in supporting the aquatic life of a water body. And a well-maintained and good-conditioned water body supports the people around, whose lives are dependent on it.
* So, maintaining the water bodies, will support our lives and fulfill our daily needs. All these well-being conditions can be created by taking care of the water bodies and treating them from time to time.
* By using this project we can do the maintenance of the source accordingly to the requirements, which has to be treated only in certain parts and it doesn’t need for us to treat the entire lake.

**Detailed Problem Description:**

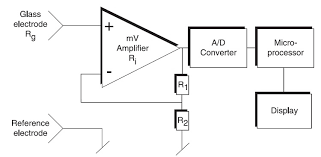
* The quality of water plays a crucial role in the health of animals and human beings, water quality monitoring of the water bodies requires a lot of effort and should be checked the entire lake.
* Water quality checking is a difficult process with the old equipment, where the monitoring is totally based on the wires connectivity system and the values are not accurate, all these drawbacks can be overcome by the new technology-based IoT water quality monitoring.
* pH value plays a major role in supporting the aquatic life of a water body. And a well-maintained and good-conditioned water body supports the people around, whose lives are dependent on it.
* So, maintaining the water bodies, will support our lives and fulfill our daily needs. All these well-being conditions can be created by taking care of the water bodies and treating them from time to time.
* The major problem is the hat pH of the lake can vary throughout the year but it is important that it stays within a range that is healthy for the fish.
* The amount of total dissolved solids or chlorophyll in the water will affect the pH.
* High pH causes water pipes to become encrusted deposits and it depresses the effectiveness of the disinfection of chlorine, thereby causing the need for additional chlorine or dissolve metals high.
* Low pH water will corrode or dissolve metals and other substances.

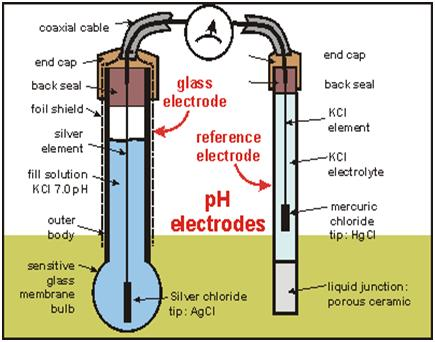
**Proposed Innovative Solution and Methodology:**

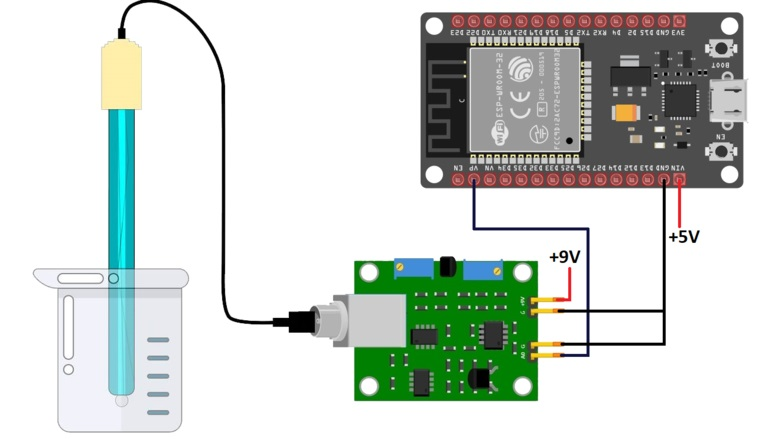
* To check the varying pH of the water body throughout the year, we have designed an advanced pH sensor reading prototype with the extension of IoT.
* This prototype helps to check the water quality of vast water bodies with ease. Its main is to check the pH level and identify the pollutants(industrial or natural) of the water source.
* The pollutants which are acidic and basic in nature can easily be identified by a pH meter without any additional equipment needed for generating the values.
* Our main motto of quality checking and remotely accessing the values moving around the water body is done by all the sensors, and motherboard connected to the pH meter.
* Thus by using our prototype water quality monitoring can be done with ease in all types of water bodies and reservoirs, compared to previous equipment or systems.
* Which have been used in the past or adopted by them. Our product has a better and more accurate performance than the old pH meters.

**Design and Modeling (Software or Hardware) of the proposed solution:**

**INTERFACING OF PH SENSOR WITH ESP8266:**







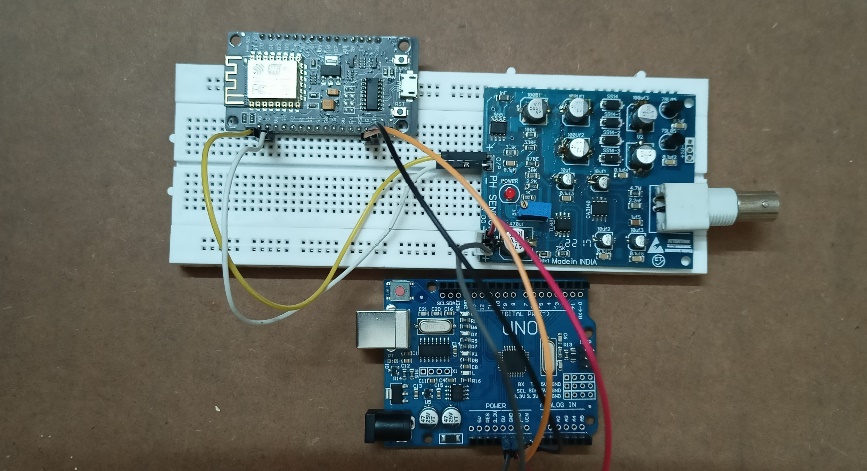


Figure 1: PH Sensor Circuit

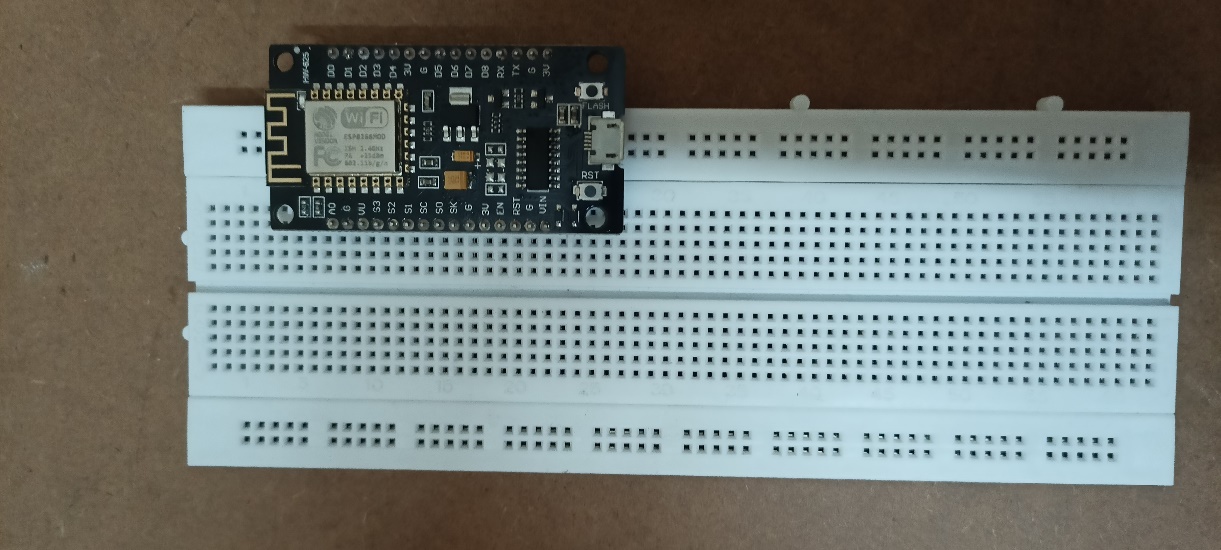


Figure 2: Wi-Fi Module (ESP8266)

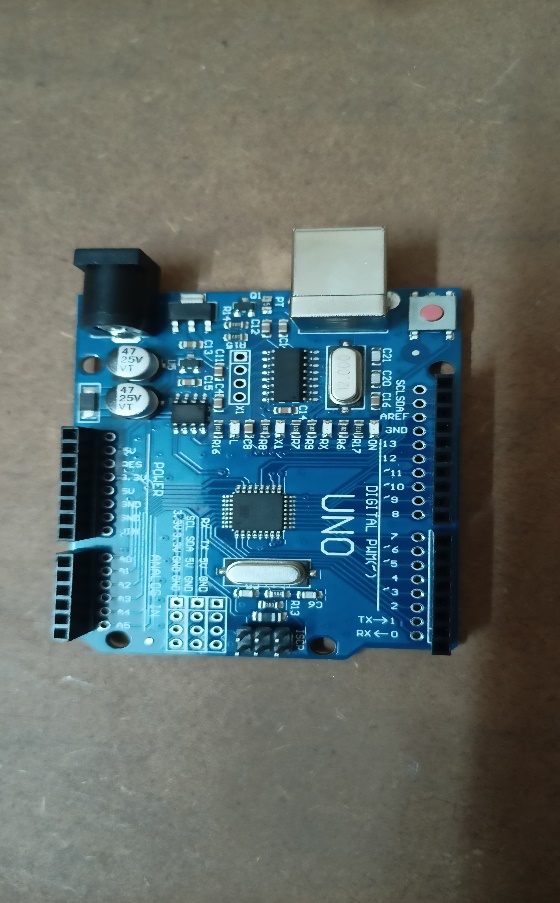


Figure 3:Arduino has Power supply

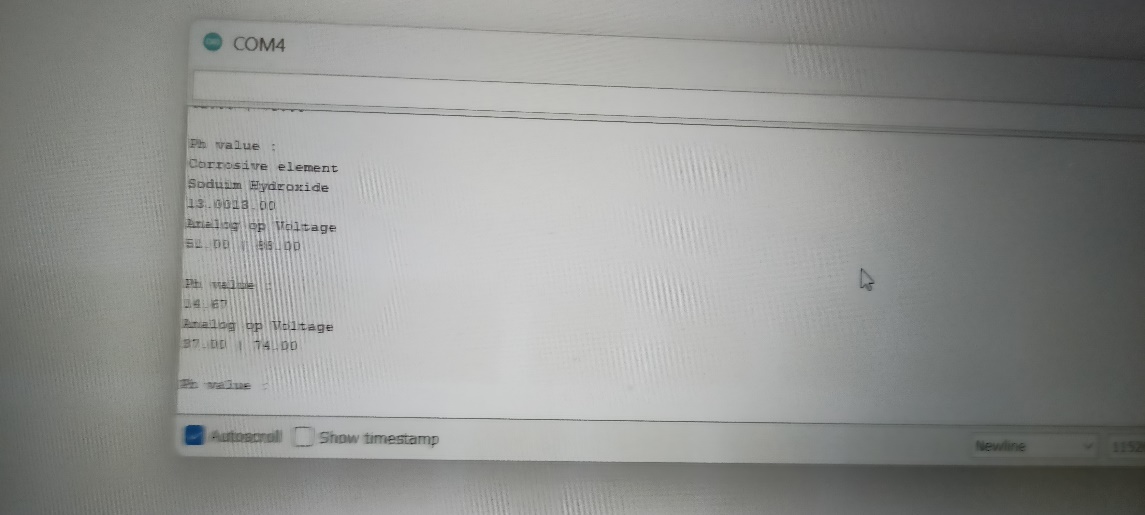


Figure 4: Output in Serial Monitor

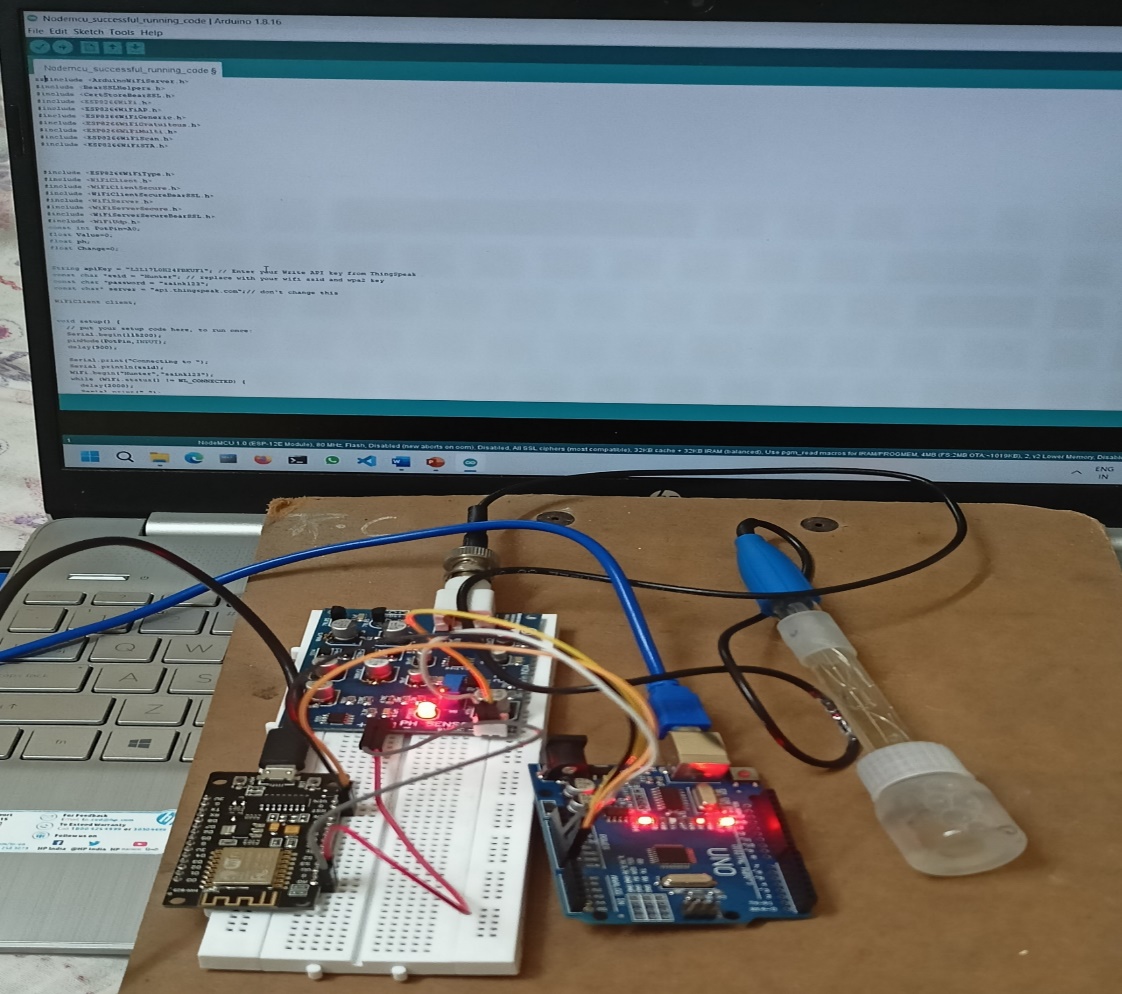


Figure 5:Entire equipment.



Figure 6:PH electrode

**

Figure 7:PH Sensor PCB

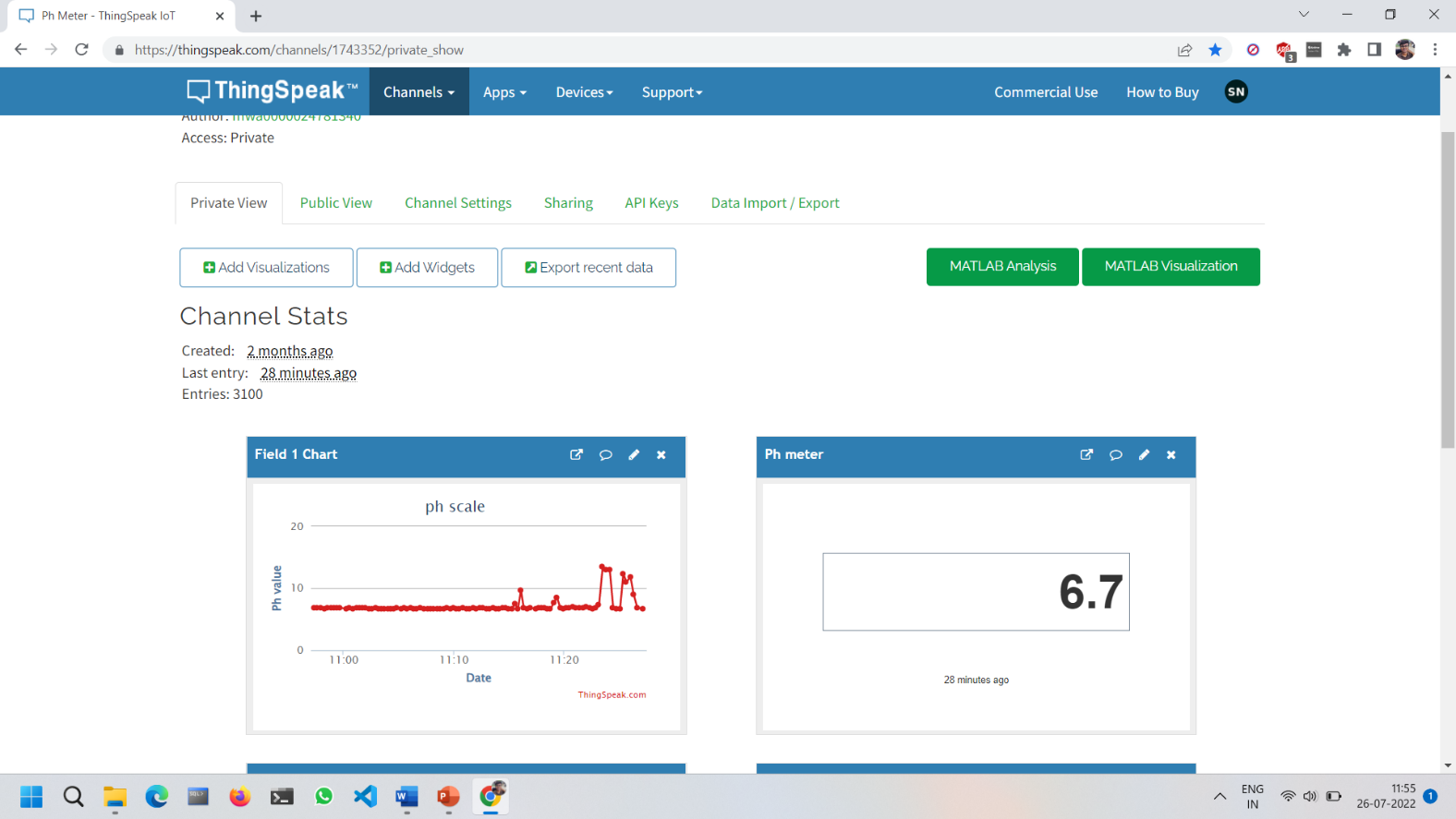


Figure 8: IoT interface

**Algorithm for PH Sensor Code:**

* **Step 1**: Install all supporting libraries for esp826(IoT device) and PH Sensor libraries. (#Importing Libraries)
* **Step 2**: Initializing the variables. (#Initials)
* **Step 3:** Provide the Link between IoT device and web-view. (#Setting up IoT)
* **Step 4**: Provide all the constraints needed for IoT Wi-Fi Connectivity like API key,

Wi-Fi Name, Password, And Provide logic code, server link into the Wi-Fi Client block code. (#Connectivity)

* **Step 5**: Read outputs from PH Sensor. (#To obtain PH values)

Under Set up loops.

* **Step 6**: Use the If, Else Conditions to Display the Chemicals and PH levels identified by the PH sensor and Logic Code. (#Conditioning)
* **Step 7**: Give commands to the Wi-Fi Client to display the corresponding output unto the web-view IoT Server. (#Display)
* (**Note**: The IoT Server Client displays the info in Visuals or Graphical Manner and can export the data into an EXCEL sheet.)

**Arduino IDE Libraries:**

* **#Include<ESP8266WiFi.h>:** ESP8266 Wi-Fi Library for Arduino provides a function for easily Wi-Fi communication using ESP8266 from sketch via the serial on such as Arduino UNO.
* Also this library has a debug output facility can monitor the transmitted and received data.
* This library provides ESP8266 Specific Wi-Fi routines that we are calling to connect to the network. The actual connection to Wi-Fi is initialized by calling: begin (“network-name”,” pass-to-network”).
* With this library you can instantiate servers, Clients and send/receive UDP packets through Wi-Fi. The shield can connect either to open or encrypted networks. The IP address can be assigned statically or through a DHCP.
* **BASIC TEST CODE:**

const int potPin=A0;

float ph;

float Value=0;

void setup() {

// put your setup code here, to run once:

Serial.begin(115200);

pinMode(potPin,INPUT);

delay(1000);

}

void loop(){

Value= analogRead(potPin);

Serial.print(Value);

Serial.print(" | ");

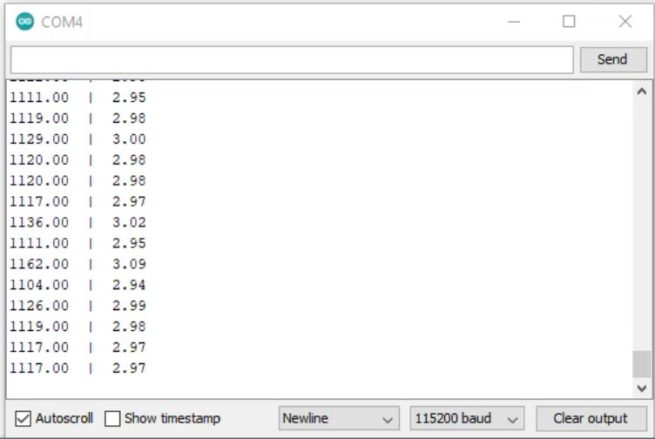
float voltage=Value\*(3.3/4095.0);

ph=(3.3\*voltage);

Serial.println(ph);

delay(500);

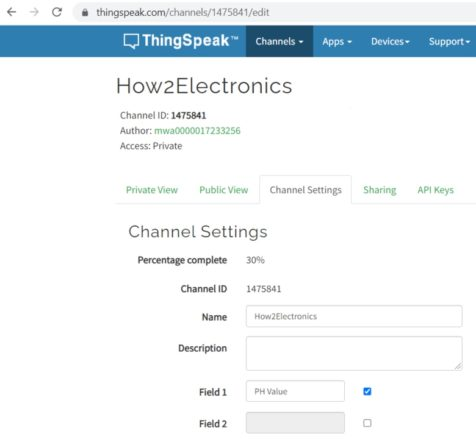
}

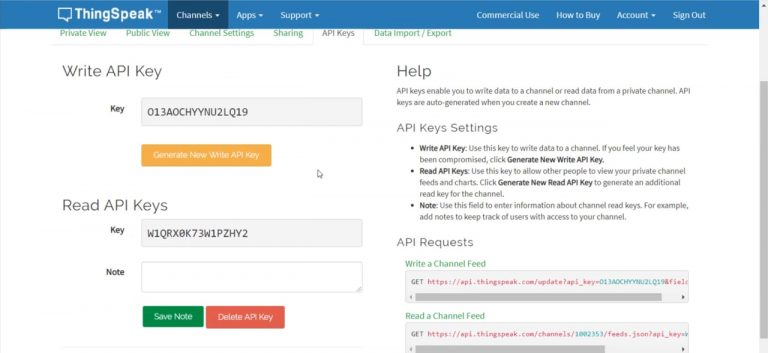
****

**IoT INTERFACE:**

* **ThingSpeak** provides a very good tool for IoT-based projects. By using the ThingSpeak site, we can monitor our data and control our system over the Internet, using the Channels and web pages provided by ThingSpeak. So first you need to sign up for ThingSpeak. So visit [https://thingspeak.com](https://thingspeak.com/) and create an account.

****

****

****

**IoT USING NODEMCU CODE:**

* **#include <ArduinoWiFiServer.h>**
* **#include <BearSSLHelpers.h>**
* **#include <CertStoreBearSSL.h>**
* **#include <ESP8266WiFi.h>**
* **#include <ESP8266WiFiAP.h>**
* **#include <ESP8266WiFiGeneric.h>**
* **#include <ESP8266WiFiGratuitous.h>**
* **#include <ESP8266WiFiMulti.h>**
* **#include <ESP8266WiFiScan.h>**
* **#include <ESP8266WiFiSTA.h>**
* **#include <ESP8266WiFiType.h>**
* **#include <WiFiClient.h>**
* **#include <WiFiClientSecure.h>**
* **#include <WiFiClientSecureBearSSL.h>**
* **#include <WiFiServer.h>**
* **#include <WiFiServerSecure.h>**
* **#include <WiFiServerSecureBearSSL.h>**
* **#include <WiFiUdp.h>**
* **const int PotPin=A0;**
* **float Value=0;**
* **float ph;**
* **float Change=0;**
* **String apiKey = "L2L17L0H24FBKUF1"; // Enter your Write API key from ThingSpeak**
* **const char \*ssid = "Hunter"; // replace with your wifi ssid and wpa2 key**
* **const char \*password = "saink123";**
* **const char\* server = "api.thingspeak.com";// don't change this**
* **WiFiClient client;**
* **void setup() {**
* **// put your setup code here, to run once:**
* **Serial.begin(115200);**
* **pinMode(PotPin,INPUT);**
* **delay(900);**
* **Serial.print("Connecting to ");**
* **Serial.println(ssid);**
* **WiFi.begin("Hunter","saink123");**
* **while (WiFi.status() != WL\_CONNECTED) {**
* **delay(2000);**
* **Serial.print(".");**
* **}**
* **// Print local IP address and start web server**
* **Serial.println("");**
* **Serial.println("WiFi connected.");**
* **Serial.println("IP address: ");**
* **Serial.println(WiFi.localIP());**
* **}**
* **void loop() {**
* **// put your main code here, to run repeatedly:**
* **Value= analogRead(PotPin);**
* **Serial.print("\n");**
* **Serial.println("Analog op Voltage"),Serial.print(Value);**
* **Serial.print(" | ");**
* **Change=(Value+37);**
* **Serial.println(Change);**
* **float voltage=Change\*(5.0/150.0);**
* **Serial.print("\n");**
* **Serial.println("Ph value :"),ph=(5.0\*voltage);**
* **delay(1000);**
* **if(ph<7.0 and ph>6.7){**
* **Serial.println("Minute urea,Pigments,Inorganic Salts");**
* **Serial.print(ph);**
* **}**
* **if(ph<1.3 and ph>0.8){**
* **Serial.println("Corrosive element");**
* **Serial.println("Hydrochloric acid");**
* **Serial.print(ph);**
* **}**
* **if(ph<=0.5 and ph>0.3){**
* **Serial.println("Corrosive element");**
* **Serial.println("Sulfuric acid");**
* **Serial.print(ph);**
* **}**
* **if(ph<1.3 and ph>1.0){**
* **Serial.println("Corrosive element");**
* **Serial.println("Nitric acid");**
* **Serial.print(ph);**
* **}**
* **if(ph<=3.3 and ph>2.8){**
* **Serial.println("Corrosive element");**
* **Serial.println("Chromic acid");**
* **Serial.print(ph);**
* **}**
* **if(ph<=2.4 and ph>2.0){**
* **Serial.println("Corrosive element");**
* **Serial.println("Acetic acid");**
* **Serial.print(ph);**
* **}**
* **if(ph<=10.9 and ph>9.5){**
* **Serial.println("Corrosive element");**
* **Serial.println("Ammonium Hydrocide");**
* **Serial.print(ph);**
* **}**
* **if(ph<=13 and ph>12){**
* **Serial.println("Corrosive element");**
* **Serial.println("Soduim Hydroxide");**
* **Serial.print(ph);**
* **}**
* **if(ph<=1.0 and ph>0.0){**
* **Serial.println("sulfuric acid");**
* **Serial.println("Water is Hazardous");**
* **Serial.print(ph);**
* **}**
* **if(ph <5.8 and ph>5.5){**
* **Serial.println("Copper");**
* **Serial.print("Water is unedible");**
* **Serial.print(ph);**
* **}**
* **if(ph > 7.0 and ph <7.5){**
* **Serial.println("Calcuim & Lead");**
* **Serial.println("Water is Safe");**
* **Serial.print(ph);**
* **}**
* **if(ph > 5.5 and ph<6.5){**
* **Serial.println("Magnesium & Lead ");**
* **Serial.print("Water is Safe");**
* **Serial.print(ph);**
* **}**
* **if(ph < 12.1 and ph>11.90){**
* **Serial.println("Potassium hydroxide");**
* **Serial.println("Water is Unsafe");**
* **Serial.print(ph);**
* **}**
* **if(ph <14.0 and ph>13.0){**
* **Serial.println("Sodium hydroxide");**
* **Serial.print("Water is Unsafe");**
* **Serial.print(ph);**
* **}**
* **if(ph < 11.4 and ph>11.0){**
* **Serial.println("Ammonia");**
* **Serial.println("Water is more Base nature");**
* **Serial.print(ph);**
* **}**
* **if(ph <2.0 and ph>1.0){**
* **Serial.println("citrus acid");**
* **Serial.println("Acidic level is High");**
* **Serial.print(ph);**
* **}**
* **if(ph < 9.3 and ph>9.0){**
* **Serial.println("Copper");**
* **Serial.print(ph);**
* **}**
* **if(ph<8.0 and ph>7.0){**
* **Serial.println("minute ammonium phosphate");**
* **Serial.println("Water is eligble");**
* **Serial.print(ph);**
* **}**
* **if(ph <8.5 and ph > 8.3){**
* **Serial.println("Sodium Bicarbonate");**
* **Serial.println("Water is eligble");**
* **Serial.print(ph);**
* **}**
* **if (ph<5.0 and ph >4.0){**
* **Serial.println("citric acid");**
* **Serial.println("Water is acidic");**
* **Serial.print(ph);**
* **}**
* **else{**
* **Serial.print(ph);**
* **}**
* **if (client.connect(server, 80)) // "184.106.153.149" or api.thingspeak.com**
* **{**
* **String postStr = apiKey;**
* **postStr += "&field1=";**
* **postStr += String(ph);**
* **postStr += "\r\n";**
* **client.print("POST /update HTTP/1.1\n");**
* **client.print("Host: api.thingspeak.com\n");**
* **client.print("Connection: close\n");**
* **client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n");**
* **client.print("Content-Type: application/x-www-form-urlencoded\n");**
* **client.print("Content-Length: ");**
* **client.print(postStr.length());**
* **client.print("\n\n");**
* **client.print(postStr);**
* **}**
* **}**
* **Detailed description of the Prototype or Product:**
* COMPONENTS REQUIRED:

|  |  |  |  |
| --- | --- | --- | --- |
| S.NO | COMPONENTS NAME | DESCRIPTION | QUANTITY |
| 1. | ESP8266 BOARD | ESP8266(DEVELOPMENT BOARD ESP8266 12E-MODULE) | 1 |
| 2. | PH SENSOR | PH SENSOR KIT FOR WATER | 1 |
| 3. | BATTERY | ARDUINO FOR POWER SUPPLY | 1 |
| 4. | CONNECTING WIRES | JUMPER WIRES | 15 |
| 5. | BREAD BOARD | -------------------------- | 1 |

#### **PH Sensor Specifications**

1. Module Power: DC 9.00V 1A
2. Measuring Range: 0-14PH
3. Accuracy: ± 0.1pH (24 )
4. Response Time: 1min
5. Industry pH Electrode with BNC Connector
6. Power Indicator LED
7. Output: Analog values in the range of (0.5V to 3V)
8. Alkali Error: 0.2PH
9. Internal Resistance: 250MO

Working principle:

* Before proceeding for ****IoT pH Meter****, let us do a basic interfacing and testing of pH sensor with the ESP8266WiFi Module. This is a simple connection diagram.
* To power the pH Sensor use an external ****9V battery**** or a 9V DC Supply. Connect the output pin of the pH Sensor signal board to A0 of ESP8266 which can be used as an Analog pin. The output of the sensor ranges between 0.5V to 3V so the sensor can be used with analog pins of ESP8266.
* The pH sensor is an analog sensor. So we will be converting the analog output voltage into a pH value. Here is a simple ESP8266 & pH Sensor Code.
* Now code the program on the editor of Arduino IDE. Select the ESP8266-12E from Board Manager and also select the COM port. Now you can upload the code.
* After uploading the code, open the Serial Monitor, and the sensor values must be visible.
* There is no need for calibration of the sensor if you haven’t changed the probe which was already provided with the Ph sensor Kit. Otherwise, you can do that by tuning the 5k resistor on the Ph module.

IoT Procedure:

* Now let us write another code to build an IoT Based pH Meter. Using this code we can monitor the pH value from any part of the world. We will be using the Thingspeak server for online monitoring of the pH data.
* **ThingSpeak** provides a very good tool for IoT-based projects. By using the ThingSpeak site, we can monitor our data and control our system over the Internet, using the Channels and web pages provided by ThingSpeak. So first you need to sign up for ThingSpeak. So visit [https://thingspeak.com](https://thingspeak.com/) and create an account.
* Then create a new channel and set up a widget for pH Value. Then create the **API**

**keys**. This key is required for programming modifications and setting your data.

* Then create the **API keys**. This key is required for programming modifications and setting your data.
* The following is the code for IoT Based pH Meter using pH Sensor & ESP8266.

String apiKey = "\*\*\*\*\*\*\*\*\*\*\*\*\*";

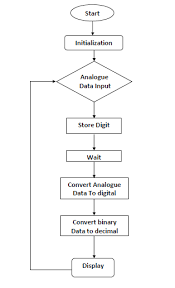
const char \*ssid = "\*\*\*\*\*\*\*\*\*\*\*\*\*";

const char \*password = "\*\*\*\*\*\*\*\*\*\*\*\*\*";

* You need to update the API key by pasting the API Key copied previously from Thingspeak Dashboard. Also update the Wi-Fi credentials, Wi-Fi name in SSID, and its password in password variable.
* Once the code is uploaded, the ESP32 will try connecting to the WiFi Network. Every 15 seconds the data will be uploaded to the Thinkspeak server.
* We can view the uploaded values in the private view section on the Thinkspeak dashboard as in the image.

Working Plan:

* Flow Chart:



* We have taken the pH electrode from pH sensor kit, and cleansed with ionized solution, after that the probe of the pH electrode is connected to port of pH module.
* The calibration value is set built in within the module,it can be varied by tunig the 5k resistor present on module.
* Then with respect to the circuit, we connected the connections started measuring the values with the help of program given to the nodemcu.

Work division:

* The work is equally divided among the group members as follows:
* Member1 & Member2:-Assembling and testing of hardware componets, like microcontrollers and sensors working. Calibration adjustment and buying of componets, collecting samples from water bodies. And finally taking report of that surround area and water body.
* Member3 & Member4:-Programming of Code needed for the microcontrollers, Document preparation through MS word and values checking needed for the idea and Research needed for the idea.

Components:

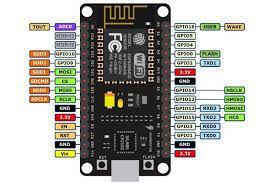
PH SENSOR KIT

* The PH sensor measures the hydrogen-ion activity in water-based solutions, we usually use it to measure the PH of a liquid. It is widely used in the chemical industry, the pharmaceutical industry, the dye industry, and scientific research where acidity and alkalinity testing is required. The drive board in this kit support both 3.3V and 5V system. And with the stander BNC probe interface and Grove connector, it is very convenient to work with Arduino and Raspberry Pi.



NODEMCU-ESP8266

* The NodeMCU (Node MicroController Unit) is an open source software and hardware development environment that is built around a very inexpensive System-on-a-Chip (SoC) called the ESP8266.It is an open source IoT platform.It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware, which is based on the ESP-12 module. The term “NodeMCU” by default refers to the firmware rather than the dev kits. The firmware uses the Lua scripting language.



**Details of the deployment of product:**

* **O**ur Prototype is at it’s developing stage, more research and changes need to done and further extensions to make the Prototype as product.
* Which is helpful for the people for sustaining the lives, understanding the environment.
* Our protoype need more development and further research, to make it durable and accurate in response.

**Outcomes & Scope for the future extension:**

* Outcomes of our idea are the pH Values and pollutants present at that pH levels of that particular waterbody and are displayed through serial monitoring or IoT web server.
* These outcomes are totally based upon the setup of the WiFi module(nodemcu) with pH Meter via with suitable programming.
* The Future Extensions of idea, are installing this pH module to an RC boat and another addition of component called Turbidity Sensor. For further understanding nature of the water body and to every corner in the water body with help of RC boat for accurate response.
* All though our idea is advance and most efficient, by further extensions the idea is made more Efficient and Accurate in values and understanding.

**References:**

* [**https://how2electronics.com/diy-iot-water-ph-meter-using-ph-sensor-esp32/**](https://how2electronics.com/diy-iot-water-ph-meter-using-ph-sensor-esp32/)
* [**https://www.pantechsolutions.net/water-quality-monitoring-using-nodemcu-esp8266**](https://www.pantechsolutions.net/water-quality-monitoring-using-nodemcu-esp8266)
* [**https://circuitdigest.com/microcontroller-projects/arduino-ph-meter**](https://circuitdigest.com/microcontroller-projects/arduino-ph-meter)
* [**https://how2electronics.com/ph-meter-using-ph-sensor-arduino-oled/**](https://how2electronics.com/ph-meter-using-ph-sensor-arduino-oled/)