# POLLUTANT DETECTION IN AQUATIC ENVIRONMENTS USING IOT

[1]Gogula Dinesh Yadav[2]Gulladurthi Naveen Kumar Reddy[3]Isanaka Chareesh Reddy[4]Dasari Sam Aman Raj[5]Guvvala Jagadeesh[6] R Prashanthi[7]S Hari Krishna

*[1]UG scholar, [2]UG scholar, [3]UG scholar,[4]UG scholar, [5]UG scholar,*
*[6]Assistant Professor, [7]Assistant Professor*
*ECE Department, Narayana Engineering College, Nellore, India*

***Abstract:*** *The main objective of this project is to determine the quality of water continuously monitoring using IoT devices like Node MCU. This system makes use of a Water quality monitoring sensor, cloud and Web UI, IOT Node MCU ESP8266, Temperature sensor, PH sensor, turbidity and TDS sensors, Arduino, Fast alerts and Buzzer. In India, water contamination is a serious environmental issue. Untreated sewage is the main cause of water contamination in India. Most Indian rivers, lakes, and surface waters are polluted by uncontrolled small-scale industries. In conventional systems, the monitoring procedure include manually gathering sample water from various places, followed by testing and analysis in a lab. This method is inefficient because it is laborious, takes a longtime, and doesn't yield results right away. Instead, create a low-cost system for real-time water monitoring using wireless sensors network (WSN). Node MCU has a connected built-in Wi-Fi module that enables internet access and sends sensor data measurements to the systems. To evaluate the quality of water from aquatic bodies, a number of sensors are employed to monitor a variety of factors like Ph value, TDS, Turbidity, temperature and water level to determine if the water is appropriate or not, the results are saved in the cloud. A warning message is sent to the master kit from WSN for monitoring and also PWD department if dirty water is found.*

***Keywords:*** *NODEMCU-ESP8266; WSN; DHT11*

## I. INTRODUCTION

In recent years, there have been a tremendous number of flood instances due to the altered rainfall pattern. Flood catastrophes cause harm to the economy and to people's lives. Floods afflict millions of people each year. To reduce flood casualties, the government was compelled by this to develop a flood forecasting method. The installation of alert systems near any significant body of water or water region gives vital wealth of knowledge that can both save lives and protect property. Some places are more prone to flooding than other places. Of course, the most effective flood warning systems cost a lot of money, require a lot of upkeep and require highly skilled personnel to operate them. One of the natural disasters that cannot be prevented is flooding. It happened too quickly and had a huge impact on both people and property. Flooding is a natural occurrence that sparks interest around the world. Both lives are lost and the ecosystem is severely damaged as a result. Flooding is caused by heavy rains, faulty structures and a variety of human factors. Precipitation levels and rates, topography, geology, land use and pre-existing moisture conditions all play a role in floods. Prior to this, the majority of the systems that have been developed only concentrated on a few areas A better response to

flood threats is required given the growing number of flood related deaths as well as financial losses seen every year around the world. It's interesting that in the past ten years, there have been numerous academic endeavors looking at how camera footage and wireless sensor data from IoT networks could improve flood management. The presented work sheds light on the potential for creating an alarm system to reduce the risk of flooding. It is able to estimate how quickly the flood will occur by sending the user an alarm message along with the timing and pace of the water rise. Additionally, it has undergone testing in a controlled setting to gauge performance.

## II. FUNCTIONAL OVERVIEW

As in [1], a novel architecture for the transceiver is proposed in order to increase the service range of IEEE802.11ah, which is necessary for the long-range IoT communication of emergency messages in emergency situations. Experimental results show that the presented architecture is suitable for the long-range IoT communication of emergency alert messages.

A wireless sensor network system could remotely monitor the real time data of water condition in the identified areas. To monitor the water conditions such as water level, flow and precipitation level, wireless sensor network system is developed.[2]

GRAB, [3] designed for robust data delivery in face of unreliable nodes and failable wireless link. GRAB forwards data along band of interleaved mesh.

In [4], in Wireless Sensor Networks (WSN) the user requirements are often desired to be evolvable, whether driven by changes of the monitored parameters or WSN properties of configuration, structure, communication capacities, node density and energy among many others.

As in [5], the Functionality is supported by the reflective and component-based Grid Kit middleware, which provides support for both WSN and Grid application domains.

## III. DESIGN

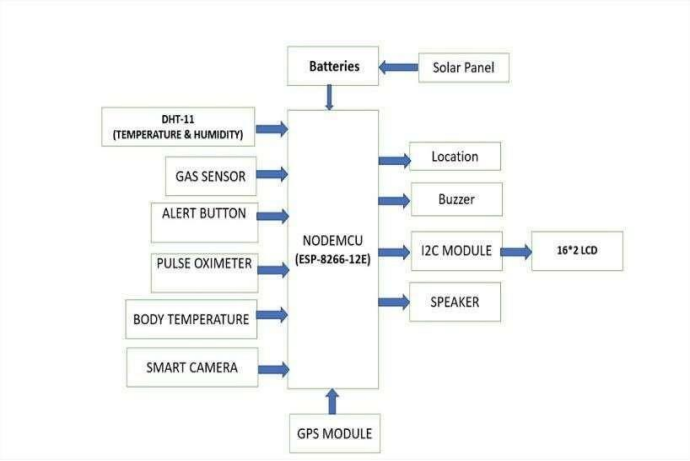


Fig.1 Block Diagram of Pollutant Detection in Aquatic Environments using IOT

### *A. NODEMCU (ESP-8266 12E) Wi-Fi Module*

ESP-8266 12E Wi-Fi module is developed by Ai- thinker Team. core processor ESP8266 in smaller sizes of the module encapsulates Ten silica L106 integrates industry- leading ultra-low power 32-

bit MCU micro, with the 16-bit short mode, Clock speed support 80 MHz, 160 MHz, supports the RTOS, integrated Wi-Fi MAC/BB/RF/PA/LNA, on- board antenna. The module supports standard IEEE802.11 b/g/n agreement, complete TCP/IP protocol stack. Users can use the add modules to an existing device networking, or building a separate network controller. ESP8266 is high integration wireless SOCs, designed for space and power constrained mobile platform designers. It provides unsurpassed ability to embed Wi-Fi capabilities within other systems, or to function as a standalone application, with the lowest cost, and minimal space requirement. The ESP8266 has seen a wide adoption as a cost-effective solution for IOT and Wi-Fi-capable devices.

### *B. TEMPERATURE AND HUMIDITY M DHT11*

DHT11 digital temperature and humidity sensor is a calibrated digital signal output of the temperature and humidity combined sensor. It uses dedicated digital modules capture technology and the temperature and humidity sensor technology to ensure that products with high reliability and excellent long-term stability. Sensor includes a resistive element and a sense of wet NTC temperature measurement devices and with a high-performance 8-bit microcontroller connected. The DHT11 sensor measures ambient temperature and humidity. DHT11 output calibrated digital signal. It utilizes exclusive digital-signal-collecting-technique and humidity sensing technology, assuring its reliability and stability.

### *C. ULTRASONIC SENSOR*

The ultrasonic sensor is a transducer which converts electrical energy into sound waves and vice versa. These sound waves fall above the normal range of human hearing and hence it is known as ultrasonic waves. These types of waves are above the frequency of about 18000 Hz. An ultrasonic sensor transmits ultrasonic waves into the air and detects reflected waves from an object. There are many applications for ultrasonic sensors, such as in intrusion alarm systems, automatic door openers and backup sensors for automobiles. Accompanied by the rapid development of information processing technology, new fields of application, such as factory automation equipment and car electronics, are increasing and should continue to do so. Ultrasonic sensors are devices that use electrical–mechanical energy transformation, the mechanical energy being in the form of ultrasonic waves, to measure distance from the sensor to the target object. Ultrasonic waves are longitudinal mechanical waves which travel as a succession of compressions and rarefactions along the direction of wave propagation through the medium.

### *D. DS18B20 TEMPERATURE SENSOR*

The DS18B20 Digital Thermometer provides 9 to 12-bit (configurable) temperature readings which indicate the temperature of the device. Information is sent to/from the DS18B20 over a 1-Wire interface, so that only one wire (and ground) needs to be connected from a central microprocessor to a DS18B20. Power for reading, writing, and performing temperature conversions can be derived from the data line itself with no need for an external power source. Because each DS18B20 contains a unique silicon serial number, multiple DS18B20s can exist on the same 1-Wire bus. This allows for placing temperature sensors in many different places. Applications where this feature is useful include HVAC environmental controls, sensing temperatures inside buildings, equipment or machinery, and process monitoring and control.

### *E. I2C MODULE (INTER INTEGRATED CIRCUIT)*

The two wires, or lines are called Serial Clock (or SCL) and Serial Data (or SDA). The SCL line is the clock signal which synchronize the data transfer between the devices on the I2C bus and it's generated by the master device. The other line is the SDA line which carries the data. The two lines are

"open-drain" which means that pull up resistors needs to be attached to them so that the lines are high because the devices on the I2C bus are active low. Commonly used values for the resistors are from 2K for higher speeds at about 400 kbps, to 10K for lower speed at about 100 kbps.

### *F.* LCD (Liquid Crystal Display)

A liquid crystal display or LCD draws its definition from its name itself. It is combination of two states of matter, the solid and the liquid. LCD uses a liquid crystal to produce a visible image. Liquid crystal displays are super-thin technology display screen that are generally used in laptop computer screen, TVs, cell phones and portable video games. LCD's technologies allow displays to be much thinner when compared to cathode ray tube (CRT) technology. Liquid crystal display is composed of several layers which include two polarized panel filters and electrodes.

### *G. RF TRANSCIEVERS*

Generally, an RF module is a small size electronic device that is used to transmit or receive radio signals between two devices. The main application of RF module is an embedded system to communicate with another device wirelessly. This communication may be the accomplished through radio frequency communication. For various applications the medium of choice is radio frequency since it does not need line of sight. The applications of RF modules mainly involve in low volume and medium volume products for consumer applications like wireless alarm systems, garage door openers, smart sensor applications, wireless home automation systems and industrial remote controls.

#### *G.1 RF Transmitter*

An RF transmitter module is a small size PCB capable of transferring a radio wave and modulating radio wave to carry data. RF transmitter modules are usually applied along with a micro controller, which will offer data to the module which can be transmitted.

#### *G.2. RF Receiver*

An RF receiver module takes the modulated RF signal to demodulate it. There are two kinds of RF receiver modules, namely the super-regenerative receivers and super-heterodyne receivers. Usually, super-regenerative modules are low power designs and low cost using a series of amplifiers to remove modulated data from a carrier wave.

### *H. BUZZER*

The electric buzzer was invented in 1831 by Joseph Henry. They were mainly used in early doorbells until they were phased out in the early 1930s in favor of musical chimes, which had a softer tone.

#### *H.1 Piezoelectric Buzzer*

Piezoelectric buzzers, or piezo buzzers, as they are sometimes called, were invented by Japanese manufacturers, and fitted into a wide array of products during the 1970s to 1980s. This advancement mainly came about because of cooperative efforts by Japanese manufacturing companies. In 1951, they established the Barium Titanate Application Research Committee, which allowed the companies to be "competitively cooperative" and bring about several piezoelectric innovations and inventions.

*H.2 Electromechanical*

Early devices were based on an electromechanical system identical to an electric bell without the metal gong. Similarly, a relay may be connected to interrupt its own actuating current, causing the contacts to buzz. Often these units were anchored to a wall or ceiling to use it as a sounding board. The word "buzzer" comes from the rasping noise that electromechanical buzzers made.

*H.3 Mechanical*

A joy buzzer is an example of a purely mechanical buzzer and they requires drivers. These buzzers typically consist of an electromagnet, a vibrating diaphragm, and a contact. When an electric current passes through the coil of the electromagnet, it generates a magnetic field that attracts the contact, causing the diaphragm to vibrate and produce the sound. Attached to the armature is a flexible diaphragm, often made of metal or plastic. When the armature moves towards the electromagnet, it pushes or pulls the diaphragm, causing it to vibrate.

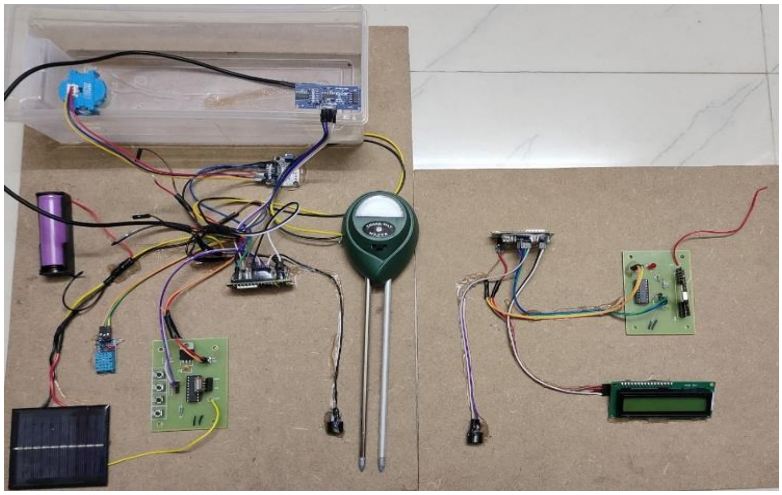

*Fig 2: Here are the connections within the NODEMCU and difference types of sensors*

## IV. INSTALLATION

After learning about the main parts of the Arduino UNO board, we are ready to learn how to set up the Arduino IDE. Once we learn this, we will be ready to upload our program on the Arduino board. In this section, we will learn in easy steps, how to set up the Arduino IDE on our computer and prepare the board to receive the program via USB cable.

**Step 1**

First you must have your Arduino board (you can choose your favorite board) and a USB cable. In case you use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable (A plug to B plug), the kind you would connect to a USB printer as shown in the following image.

**Step 2: Download Arduino IDE Software**

You can get different versions of Arduino IDE from the Download page on the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.

**Step 3: Power up your board**

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks.

**Step 4: Launch Arduino IDE**

After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Double-click the icon to start the IDE.

**Step 5: Open your first project**

Once the software starts, you have two options

- Create a new project.
- Open an existing project example.

To create a new project, select File --> New.

**Step 6: Select your Arduino board**

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer. Go to Tools -> Board and select your board.

**Step 7: Select your serial port**

Select the serial device of the Arduino board. Go to Tools -> Serial Port menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.

**Step 8: Upload the program to your board**

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.

**BLYNK APP**

Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things.

Blynk is a platform with iOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet. It's a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets. It's really simple to set everything up and you'll start tinkering in less than 5 mins. Blynk is not tied to some specific board or shield. Instead, it's supporting hardware of your choice. Whether your Arduino or Raspberry Pi is linked to the Internet over Wi-Fi, Ethernet or this new ESP8266 chip, Blynk will get you online and ready for the Internet Of Your Things.

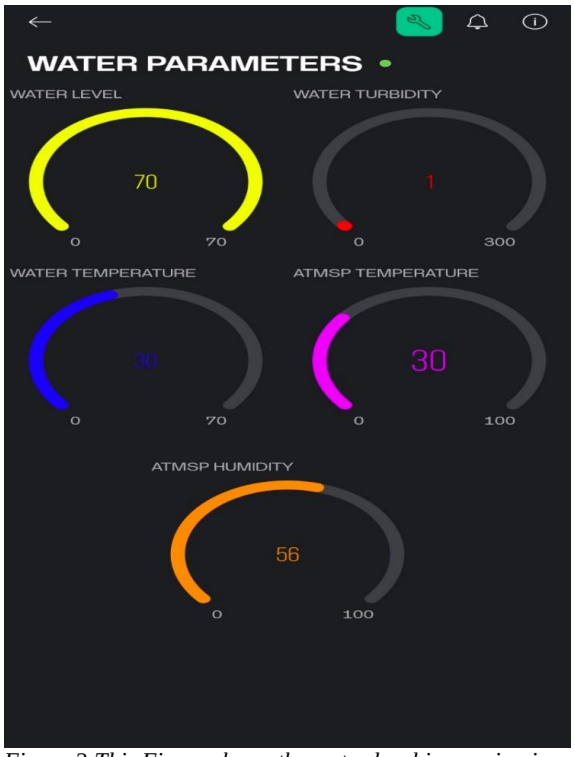


*Figure 2:This Figure shows the water level increasing in the river and then sign for the Emergency Alert*

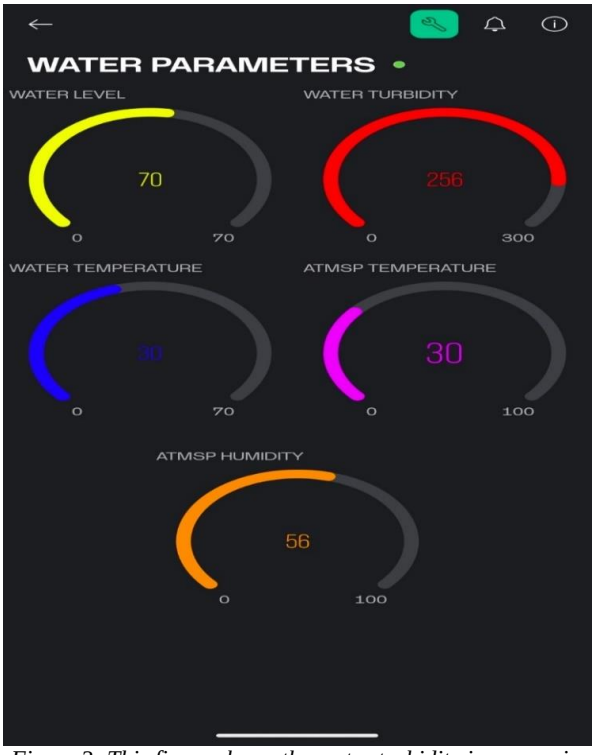


*Figure 3: This figure shows the water turbidity increases in water, water temperature, atmospheric temperature and atmospheric humidity*

## V. CONCLUSION

The potential to create an alarm system that will mitigate the risk of flooding is highlighted in this project. In the case of a flood or other dangerous natural disaster, it might also support a variety of governmental bodies or authority, which might aid society and humanity. The suggested model has been tried out and is functioning as described in this project. It will keep an eye on all potential sources of flooding. It will quickly transmit a warning if the water level increases along with the speed. Additionally, it makes dealing with and recovering from this disastrous situation more accessible. In conclusion, it will aid the neighbourhood in making quick judgments and preparing for this tragedy.

## VI. REFERENCES

[1] K. S. Adu-Manu, N. Adam, C. Tapparello, H. Ayatollahi, and W. Heinzelman, ‘‘Energy-harvesting wireless sensor networks (EHWSNs): A review,’’ ACM Trans. Sensor Netw., vol. 14, no. 2, pp. 1–50, Apr. 2018.S. Scataglini, G. Andreoni, J. Gallant, "A Review of Smart Clothing in Military", Workshop on Wearable Systems and Applications, 2015.

[2] Z. Asif, Z. Chen, C. An, and J. Dong, ‘‘Environmental impacts and challenges associated with oil spills on shorelines,’’ J. Mar. Sci. Eng., vol. 10, no. 6, p. 762, May 2022.

[3] A.-L. Balogun, S. T. Yekeen, B. Pradhan, and K. B. W. Yusof, ‘‘Oil spill trajectory modelling and environmental vulnerability mapping using GNOME model and GIS,’’ Environ. Pollut., vol. 268, Jan. 2021, Art. no. 115812.

[4] A. M. Bernabeu, M. Plaza-Morlote, D. Rey, M. Almeida, A. Dias, and A. P. Mucha, ‘‘Improving the preparedness against an oil spill: Evaluation of the influence of environmental parameters on the operability of unmanned vehicles,’’ Mar. Pollut. Bull., vol. 172, Nov. 2021, Art. no. 112791.

[5] M. Bhuyan, Intelligent Instrumentation: Principles and Applications. Boca Raton, FL, USA: CRC Press, 2013.

[6] C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), 1st ed. Springer, 2006.

[7] H. Denkilkian, A. Koulakezian, R. Ohannessian, M. S. Chalfoun, M. K. W. Joujou, A. Chehab, and I. H. Elhajj, ‘‘Wireless sensor for continuous real-time oil spill thickness and location measurement,’’ IEEE Trans. Instrum. Meas., vol. 58, no. 12, pp. 4001–4011, Dec. 2009.

[8] R. C. Eberhart and Y. Shi, Computational Intelligence: Concepts to Implementations, 1st ed. San Mateo, CA, USA: Morgan Kaufmann, 2007.

[9] M. Fingas and C. Brown, ‘‘A review of oil spill remote sensing,’’ Sensors, vol. 18, no. 2, p. 91, Dec. 2017.

[10] D. D. Gajski, G. S. S. Abdi, and A. Gerstlauer, Embedded System Design: Modeling, Synthesis and Verification, 1st ed. Springer, 2009.