

# AN EFFECTIVE INVESTIGATION USING ADVANCED SENSORS FOR INTELLIGENT INDUSTRIAL MONITORING SYSTEM USING IOT TECHNOLOGY

## Abstract

Today, smart grid, smart homes, smart water networks, intelligent transportation, are infrastructure systems that connect our world more than we ever thought possible. The common vision of such systems is usually associated with one single concept, the Internet of Things (IoT), where through the use of sensors, the entire physical infrastructure is closely coupled with information and communication technologies; where intelligent monitoring and management can be achieved via the usage of networked embedded devices. These devices will connect to internet to share different types of data. We have proposed an Industrial Monitoring System using WIFI module and sensing based applications for internet of things. By detecting the values of sensors it can easily find out the Temperature, humidity, and gas present in the industrial area.

**Key words:** Arduino Uno, LCD Display, sensor, gas sensor, temperature sensor, DHT22, LM35, fire sensor, Node MCU, LCD.

## Authors

### **Dr. B. Nageshwar Rao**

Associate professor  
Department of IoT, School of Engineering  
Malla Reddy University,  
Hyderabad, Telangana, India.  
nageshwarb@mallareddyuniversity.ac.in

### **V. Vijayakumar Dasari**

Assistant professor  
Department of CS, School of Engineering  
Malla Reddy University,  
Hyderabad, Telangana, India.  
vijaydasari@mallareddyuniversity.ac.in

### **Pulukuri Lakshmi Naga Bramham**

III<sup>rd</sup> IoT student,  
Department of IoT  
School of Engineering  
Malla Reddy University  
Hyderabad, Telangana, India.  
pbramham9809@gmail.com

### **Dr. G. Anand Kumar**

Associate professor & HOD  
Department of IoT  
School of Engineering  
Malla Reddy University  
Hyderabad, Telangana, India.  
nageshwarb@mallareddyuniversity.ac.in

## I. INTRODUCTION

In its simplest form, the Internet of Things is a network of physical objects that can exchange data with the use of sensors, electronics, software, and connectivity. There is no need for human involvement with these systems. The micro-controller known as Node MCU receives the signal from many sensors, including the smoke, temperature, and gas sensor. This is then transmitted to the IOT platform via the microcontroller. In the event of a fire, the smoke sensor and temperature sensor would communicate information to the Node MCU upon detecting the presence of smoke and changes in temperature. Wi-Fi, a buzzer, and an exhaust fan are all connected to the microcontroller. The backup light will switch on if there are any changes in the lighting conditions, according to the information sent to Node MCU.

The Node MCU is programmed to turn on the buzzer when the temperature sensor reads a temperature exceeding a threshold value, turn on the exhaust when the gas sensor reads a smoke or gas higher than the threshold value, and turn on the backup light if the light sensor reads a light reading lower than the threshold value. As required, this value can be configured. The sensor readings will be transmitted simultaneously to the website (Blynk). Since the client is using an IOT platform to monitor the data LIVE, quick action may be made.

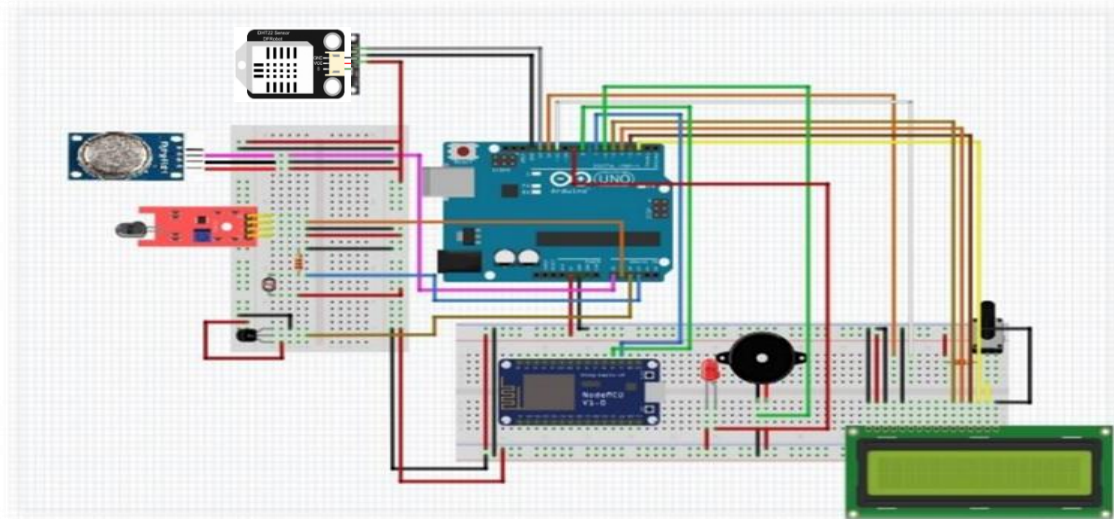
To find LPG gas leaks, gas sensors are employed. In the event of a gas leak, the sensor would alert the Node MCU and instruct it to switch on the exhaust fan while simultaneously transmitting the same information over IOT. The detection of lighting conditions uses a light sensor. The sensor would detect any unusual lighting conditions, such as reduced lighting in the workspace, and send a signal to the Node MCU, which would turn on the backup light.

## II. MOTIVATION

Quick action may be taken because the customer is using an IOT platform to monitor the data LIVE. The use of gas sensors to detect LPG gas leakage. The sensor would notify the Node MCU of a gas leak and give it instructions to turn on the exhaust fan while simultaneously sending the same information through IOT. A light sensor is used to determine the lighting conditions. The sensor would send a signal to the Node MCU, which would turn on the backup light, if it noticed any unexpected lighting conditions, such as decreased lighting in the workstation.

The primary goal of this project is to develop an affordable, simple-to-build system that monitors an industry's environment and alerts staff members even if there is a remote possibility of an accident due to a small fire, gas leak, abrupt change in temperature, or loss of light. The approach is designed to enable industry workers to take preventative action before these minute factors cause a serious accident.

### III. DESIGN AND MODELING OF IoT INDUSTRY POTECTION

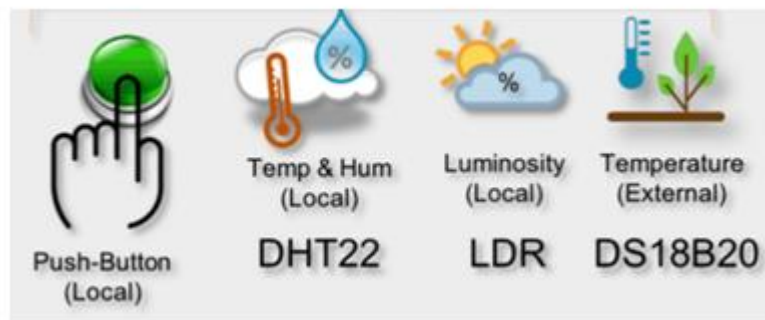


**Figure 1:** IoT Industry protection

- 1. Hardware Setup:** Recognize and collect the required hardware parts, such as the Arduino, NodeMCU, DHT sensor, gas sensor, fire sensor, light sensor, LCD display, and buzzer. Connect the parts in accordance with their pin layouts and specifications. Ensure that the sensors, buzzer, LCD display, and microcontrollers are all properly wired and connected.
- 2. Software Configuration:** Install the necessary libraries for Arduino and NodeMCU development, including the Blynk, LiquidCrystal\_I2C, Software Serial, Adafruit\_Sensor, DHT, and Wire libraries. Install the development environments for Node MCU and Arduino on your computer. Create an account on the Blynk app by downloading it to your smartphone. To receive the authentication token required for Blynk integration, create a new project in the Blynk app.
- 3. Code Implementation:**
  - Launch the Arduino IDE, and then start a fresh sketch.
  - Enter the provided code by copying and pasting it into the drawing.
  - Check the code to make that all the necessary constants, pin assignments, Wi-Fi login information, Blynk authentication tokens, and emergency contact information are configured correctly.
  - To check for syntax issues or warnings, compile the code. Fix any problems that surface.
  - Using the proper technique (such as a USB cord), upload the code to the Arduino or Node MCU board.
  - To confirm that the board is successfully linked to the Wi-Fi network and the Blynk cloud platform, monitor the serial output in the Arduino IDE.

#### 4. Testing and Calibration:

- Turn on the hardware setup and watch how it functions.
  - Make that the temperature and humidity values on the LCD display are accurate.
  - To make sure they offer reliable readings, test the gas, fire and light sensors.
  - Test that an emergency alert is sent via Blynk by initiating emergency conditions (for example, by exposing the gas sensor to a high concentration of gas).
  - By checking the temperature and humidity levels in real-time on the Blynk app, you can test the Blynk integration. If necessary, calibrate the sensors to increase their accuracy and responsiveness.
5. **Performance Assessment:** To assess the system's stability and dependability, keep an eye on it for a while. Examine the sensors' response times and the emergency alert system's efficiency. To determine the sensor readings' accuracy, compare them to standard measurements. Gather opinions on the system's functionality and usefulness from users or other stakeholders.

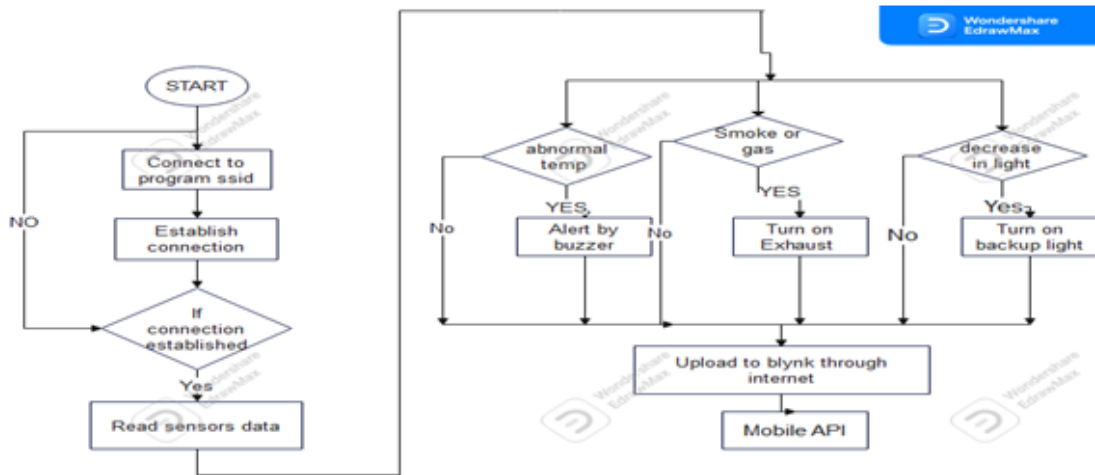


**Figure 2:** Different sensors used in IoT Industry protection

#### IV. PROPOSED IOT INDUSTRY PROTECTION ARCHITECTURE

1. Start the programme, connect it to the SSID, and if the SSID is verified, a connection will be made.
2. We've now entered the program's main loop. We read sensor values for the supplied temperature, gas, light and flame sensors and update the Blynk app with the real-time data.
3. Look for any emergency situations.
4. If a flame is detected, the temperature exceeds the threshold value, the gas concentration exceeds the threshold value, the light intensity is below the threshold value, or any of the aforementioned conditions apply.
5. "Send Emergency SMS ()" will send an SMS to the mobile whenever any ambiguous situation occurs in any Sensor function.
6. This function also identifies the mobile sensor that is having problems.
7. The programme will also provide an LCD display for alerts in the business.
8. To notify the staff, this programme is connected to a buzzer.
9. If no problem is found, the emergency alarm feature of the programme will be disabled.
10. The buzzer will also be turned off by the programme.

11. Following a specified delay, the loop will then repeat itself.



**Figure 3:** IoT Industry Protection Architecture

## V. TODAY'S SCENARIOS

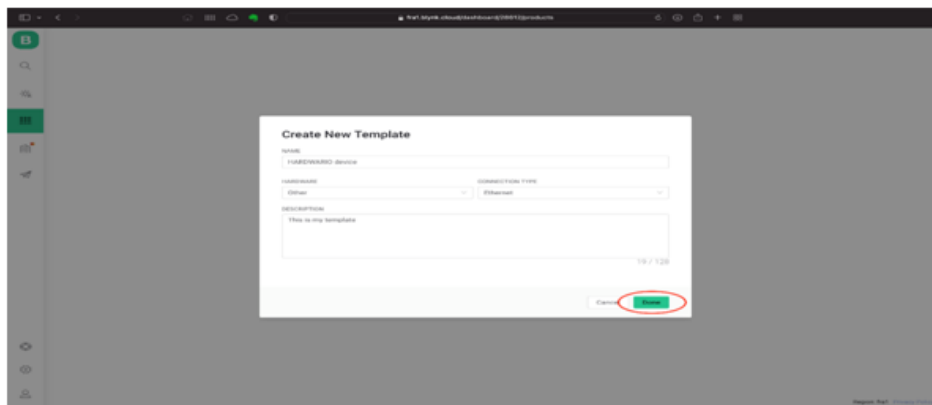
The analysis of accidents showed that information about the disaster was only made public once it had affected infrastructure and had a direct impact on people's lives. These accidents could have been avoided if there had been prior or before impact information available prior to the disaster, which was relatively simple to do. It should be emphasized that in the majority of fire mishaps, the fire was not discovered until it had spread over the area and was obvious from the exterior of the structure. Therefore, the goal of this initiative is to notify people in advance about the possibility of a disaster in order to limit significant financial losses for the sector.

## VI. CONNECTING TO CLOUD

The Blynk app account creation process is the first step. Even if you've used a prior version of the app, you need a new account.

To create a new account, go to [blynk.cloud](https://blynk.cloud) in your browser and click it. Once you have entered your email, a message with a link to generate a password will be sent to you.

Click + Create template on the right after choosing Templates in the left menu. Enter the name of the template in the appropriate field; at this point, it is irrelevant whether hardware or connection type you use. To complete building the template, click the Done button.



**Figure 4: Step1**

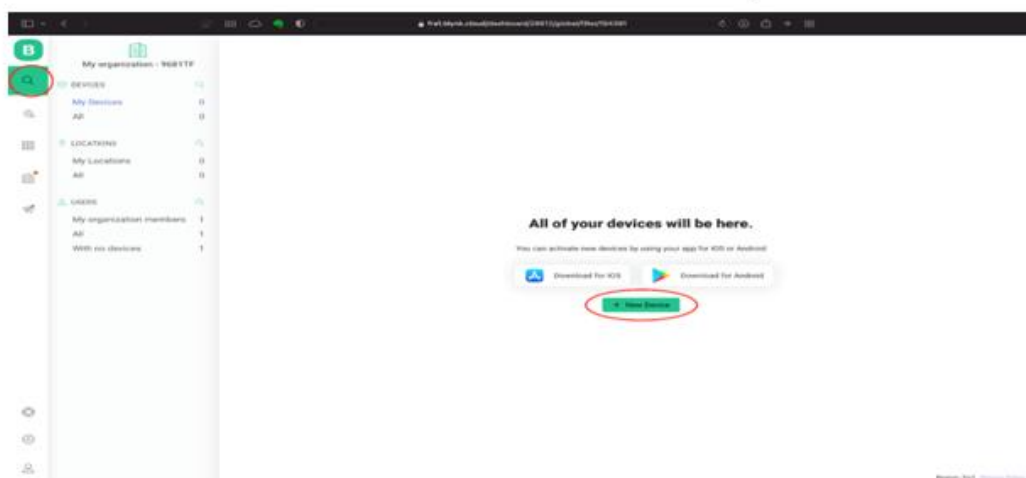
You must make a data stream for each value you want to write to Blynk. The virtual pins principle, which was used in the earlier iteration of Blynk, will be applied in this handbook.

Select DataStream's from the menu on the built template's open page. After selecting the Virtual Pin option, click the Add New Datastream button.

Enter the name, color, number (each DataStream will have a unique number), datatype, and, if applicable, the unit and other characteristics for the data stream. Finally, click Create to add.

Repeat the process if you wish to add more data streams.

Click on the magnifying glass on the left to go to Blynk's home page. To open the window for adding a new device, click the Add New Device button.



**Figure 5: Step2**

AN EFFECTIVE INVESTIGATION USING ADVANCED  
SENSORS FOR INTELLIGENT INDUSTRIAL MONITORING SYSTEM USING IOT TECHNOLOGY

Pick From template from the creation method drop-down menu. Choose your template and give the gadget a name in the following form. To complete the procedure, click Create.

On the right, you should see a notification that contains details about the new device. They include details regarding TEMPLATE ID and AUTH TOKEN, which you will shortly need to load in Playground.



Figure 6: Step3

Through the Node Mcu wifi module and Blynk application, the same precise data is transferred to the chosen user's phone. With the exception of the message produced by the fire sensor, all of the display units are identical to those on the LCD screen. When a fire is detected, the fire sensor in the Blynk programme indicates 0; otherwise, it indicates.



Figure7: Step4

## VII. DATA MANAGEMENT

1. Data Cleaning
2. Statistical Summaries
3. Visualizations
4. Correlation Analysis
5. Anomaly Detection
6. Data Segmentation

## VIII. RESULTS AND DISCUSSION

All of the linked sensors have analogue inputs, and real-time data is received and shown on the LCD that is coupled to the Arduino board. On the LCD, the temperature measured by the temperature sensor LM 35 is displayed in Fahrenheit. The gas density sensed by the MQ2 module and the light intensity sensed by the LDR are both presented in percentage form. A display message indicates whether or not fire has been detected. The Node MCU WIFI module uses the Blynk application to send the same precise data to the desired user's phone. With the exception of the message produced by the fire sensor, all of the display units are identical to those on the LCD screen. The Blynk application does not produce any messages.



**Figure 8:** LCD Display showing results

## IX. CONCLUSION

This paper outlines the development of an Arduino-based IOT-based industrial security system. The system has the capacity to gather sensor data and act deftly under various circumstances. Based on the use of wireless communication, it was created. It is ideal for real-time and effective high-speed data collecting system requirements in an IOT context. The use of ARDUINO makes the design of peripheral circuits much simpler and increases the system's flexibility and extensibility. As long as they are connected to the system, several sensor kinds can be used. This dissertation describes the primary design process for the reconfigurable smart sensor interface device. Lastly, using the example of monitoring industrial safety metrics in an IOT setting,



## **X. FUTURE SCOPE**

The threshold values of the sensor are reconfigured by the suggested approach. There is a lot of room for developing an environment that can be automatically controlled based on the alerts from the monitoring system, but this project was unable to do so since the Arduino (UNO R3) board's output voltage was insufficient. As a safety measure, this system can be connected to a GSM module that, in the event that the Blynk program is closed or malfunctioning, will text the user to alert them of the emergency. Additionally, this system can be made to take wise precautions that will shield the industry from losses, including turning on the ventilation fan in the event of a gas leak detection or activating the industry's fire suppression system when a large flame is detected based on the temperature and flame sensor readings.

## **REFERENCES**

- [1] S.P. Lalwani, MKKKhurana , S.J. Khandare, O.U.R. Ansari , Dr. S.B. Pokle, "IOTBased Industrial Parameters Monitoring and Alarming System using Arduino - A Novel Approach," International Journal of Engineering Science and Computing, vol. 8, no. 4, pp.17305-17308, April. 2018
- [2] Li Da Zu" Internet of Things in industries: A Survey" IEEE transaction on Industrial on Industrial informatica, vol,no, November 2014
- [3] Y. Fan, Y. Yin, L. Xu, Y. Zeng, and F. Wu, "IoT based smart rehabilitation system," IEEE Trans. Ind. Informat., vol. 10, no. 2, pp. 1568–1577, 2014.
- [4] L. Wang, L. D. Xu, Z. Bi, and Y. Xu, "Data cleaning for RFID and WSN integration," IEEE Trans. Ind. Informat., vol. 10, no. 1, pp. 408–418, Feb. 2014.
- [5] He, G. Yan, and L. Xu, "Developing vehicular data cloud services in the IoT environment," IEEE Trans. Ind. Informat., vol. 10,no. 2, pp. 1587–1595, 2014.
- [6] Ahmad El Kouche"Towards a Wireless Sensor Network Platform for the Internet of Things", IEEE ICC 2012 - Adhoc and Sensor Networking Symposium

