*IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)*
*e-ISSN: 2278-1676,p-ISSN: 2320-3331, Volume 14, Issue 3 Ser. I (May. – June. 2019), PP 70-79*
*www.iosrjournals.org*

# Development of Pollution Monitoring and Control System Using Lab view

Chandni Mariann Thomas[1], Bijily Rose Varghese[2]
*[1](Electrical and Electronics Engineering, MGM College of engineering and Technology, India)*
*[2](Electrical and Electronics Engineering, MGM College of Engineering and Technology, India)*
*Corresponding Author: Chandni Mariann Thomas1*

***Abstract:*** *Pollution monitoring systems are widely applied in industries. One of the greatest challenges in pollution is to control the parameters that causes pollution and deteriorates the natural and industrial environment pattern hence this has received high concern from industries. The main objective of this paper is to design a systematic and robust system to monitor and control the parameters causing pollution and to minimize the effect of these parameters without influencing the plant or natural environment. The proposed method is to model a system to read and monitor pollution parameters and to inform the required authorities when any of these factors goes higher than the standards. This paper introduces a LabVIEW based program for monitoring and controlling industrial pollution caused by untreated disposal of waste. Thus, observed polluting parameters can be monitored online from anywhere. This paper proposes the use of an Arduino Uno board which fetches the temperature and humidity parameter from the DHT -11 sensor, measures air quality using MQ-135 sensor and collects acid and base value from a hardware compilation of pH sensor. These observed values are monitored and controlled using a LabVIEW based program and are updated to cloud data storage using ThingSpeak IoT analytics service platform through which the values can be monitored online. To increase the reliability and flexibility an android application using App Inventor is also developed. The overall industrial sector can undergo a rudimentary and pre-eminent change similar to the industrial revolution with the development of Internet of Things (IoT) based technologies. Online monitoring solutions invoked by IoT techniques of environmental polluting parameters help us to congregate the parameter values such as pH, temperature, humidity and air quality. These monitoring techniques using sensors will enable us to maintain a keen control on the environmental pollution caused by the industries.***

***Keywords:*** *Arduino UNO, DHT-11 sensor, IoT, LabVIEW, MQ-135 sensor.*

---

Date of Submission: 06-05-2019 Date of acceptance: 20-05-2019

---

## I. Introduction

Industrialization has increased the extent of automation but at the same time it has increased the pollution by releasing the undesirable parameters into the environment especially in industrial areas. Industrial pollution has adversely affected our bio-diversity. In order to control industrial pollution, we can adopt two approaches: -

i. Reducing consumption or usage of polluting product.
ii. Treatment of wastes, discharges and disposal of pollutant.

For the past two decades treatment of wastes, discharges and disposal of pollutant approach was implemented across industries. But this approach has certain disadvantages like uncertainty about the extent of effect of pollutant discharge, waste treatment in one medium results in contamination of other medium and waste treatment is highly costly. Due to these disadvantages industries are now adopting the approach of reducing consumption or usage of polluting product. The advantages of pollution prevention methods are improving the effectiveness of managing reduced waste streams, minimize the uncertainty associated with the environmental impact of released pollutants, avoiding cross-media transfers of released pollutants and protecting natural resources. Hence, there aroused a necessity for a system to monitor and assess the industrial pollution. Special attention is given to elements which may affect human health and the biodiversity of the natural system itself. Industrial monitoring is carried out at regular intervals of time by collecting information from different locations and the collected data is used to define current circumstances. Large variations are found between industries due to the complexity of various parameters [1]. With recent technological advances; monitoring, measuring and controlling of different parameters have become technically and economically feasible for industries. An efficacious method is to check or assess the condition if any parameters under consideration (e.g. air quality, pH, humidity and temperature) goes than the formulated level. This paper proposes the use of an Arduino Uno board which fetches the temperature and humidity parameter from the DHT -11 sensor, measures air quality using MQ-

135 sensor and collects acid and base value from a hardware compilation of pH sensor. These observed values are monitored and controlled using a LabVIEW based program and are updated to cloud data storage using ThingSpeak IoT analytics service platform through which the values can be monitored online. To increase the reliability and flexibility an android application using App Inventor is also developed. LabVIEW (Laboratory Virtual Instrument Engineering Workbench) is a platform for system design and is a development environment for a visual programming language from National Instruments. The most powerful feature that LabVIEW offers engineers and scientists is its graphical programming environment. By creating a graphical user interface on the computer screen, we can design customized virtual instruments using LabVIEW. This will support us to operate the instrumentation program, control selected hardware, analyze acquired data and display results [9]. The Internet of Things (IoT) is a framework of things or physical objects entrenched with sensors, software's, electronics, and internet connectivity, which allow the connected objects to summon and reciprocate data. Virtually endless opportunities are possible through various connections in IoT platform. On a broader spectrum, the IoT can be applied to things like home appliances, transportation networks etc. Opportunities like "smart cities" can be developed using IoT which can help us reduce waste and improve efficiency in areas such as energy use; thus, helping us in understanding and improving the way we work and live. This paper also presents the implementation of a simple IOT system within an industry for monitoring purpose. This increases the safety level of workers as well as the working environment from possible work hazards. Any miscellaneous activity within the industry can be monitored easily from anywhere in the world by any authorized person [10].

## II. System Design

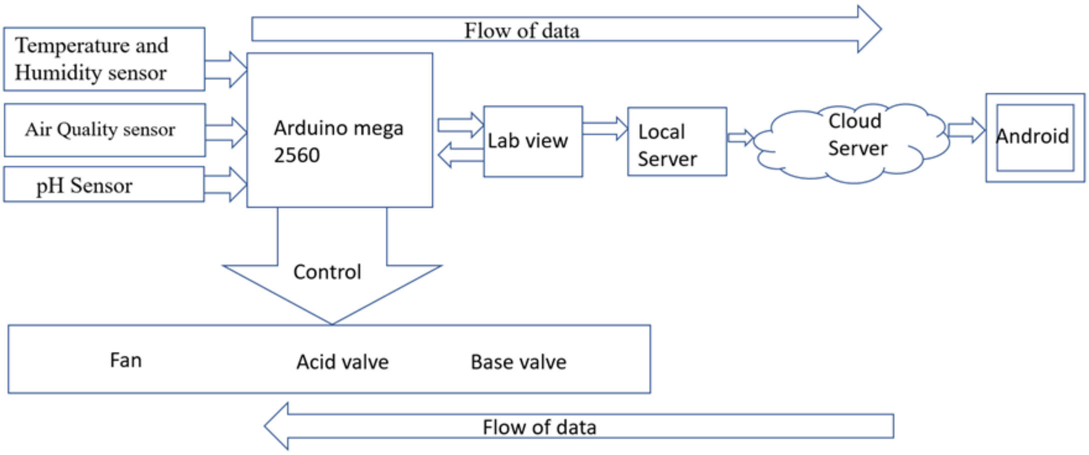


**Figure 1.** block diagram of the proposed system

The model block diagram of the proposed system is shown in the Fig 1. The block diagram describes the overall working of the proposed work. There is a two-way flow of data in the proposed system. In case of monitoring the flow of information is from left to right. After the successful interfacing of all the hardware, the system starts with the Arduino UNO board fetching data from the DHT-11, MQ-135 sensors and data is sent to the LabVIEW. These monitored values can be stored in the cloud database and can be displayed using ThingSpeak service platform. These fetched data can also be displayed on mobile phones using App Inventor application software. The flow of information will be from right to left for the controlling section. The control signals are sent from LabVIEW and are decoded by Arduino and it sets the fan and various control valves on or off. The individual description about each block is given in the following paragraphs.

## III. Measurement Of Parameters

### 3.1 Arduino UNO

The Arduino UNO is an open-source microcontroller board based on the microchip AT mega 328P microcontroller and was developed by Arduino.cc. It has digital and analog input/output (I/O) pins that can be interfaced to various expansion boards (shields) and other circuits. The board has 14 digital pins and 6 analog pins. It can be programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9-volt battery. It operates for voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. The ATmega328 on the Arduino Uno comes preprogramed with a bootloader that allows uploading new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. The Uno also differs from all preceding boards as it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter. The Arduino/Genuino Uno has a number of facilities for communicating with a computer, another Arduino/Genuino board, and other microcontrollers. The ATmega328 provides UART TTL (5V) for serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware is used for the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A software serial library allows serial communication on any of the Uno's digital pins. Rather than requiring a physical press of the reset button before an upload, the Arduino/Genuino Uno board is designed in such a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip [8].

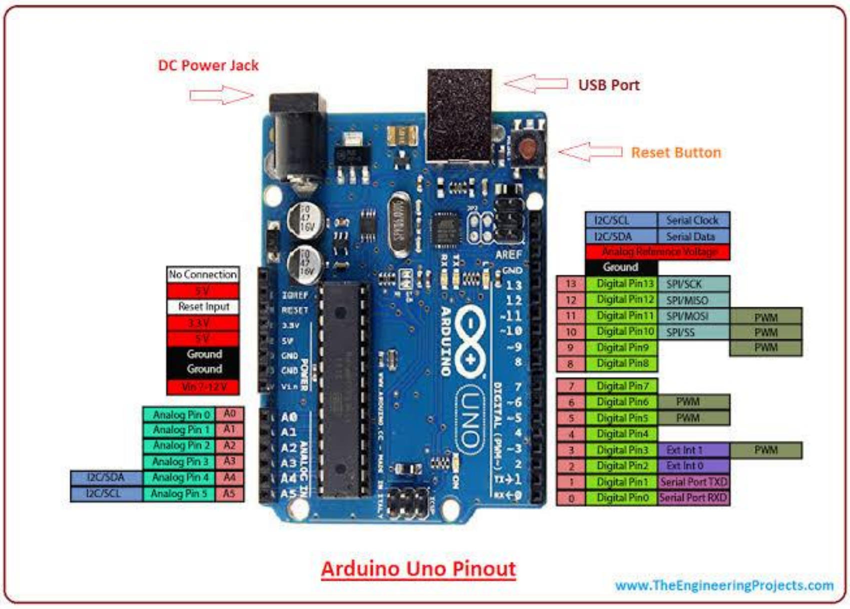


**Figure 2.** Arduino UNO board

### 3.2 DHT-11 temperature and humidity sensor

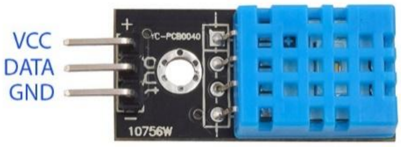


**Figure 3.** DHT-11 sensor

The DHT11 sensor is commonly used as a temperature and humidity sensor. The sensor has a dedicated NTC to measure temperature. The sensor is also factory calibrated and can be easily interfaced with other microcontrollers. The sensor is used to measure temperature from 0°C to 50°C and humidity from 20% to 90% with an accuracy of ±1°C and ±1%. The DHT11 sensor is factory calibrated and uses an 8-bit microcontroller to output the temperature and humidity values serial data and hence it is highly easy to set it up. The connection diagram for this sensor is as shown in Fig.4.

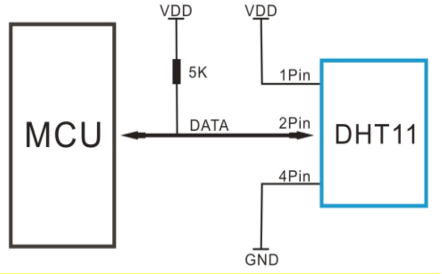


**Figure 4.** DHT-11 sensor block diagram

The data pin is connected to an I/O pin of the MCU and a 5K pull-up resistor is used. This data pin outputs the value of both temperature and humidity as serial data. The available ranges and accuracy of the DHT11 are:- humidity range: 20-90% RH, humidity accuracy: ±5% RH, temperature range: 0-50 °C, temperature accuracy: ±2% °C, operating voltage: 3V to 5.5V [10].

### 3.3 MQ-135 air quality sensor

MQ-135 sensors are used in air quality control equipments and are suitable for detecting or measuring of $NH_{3,}$ $NO_x$, alcohol, benzene, smoke, $CO_2$ etc. MQ-135 gas sensor applies SnO2 which has a lower conductivity in the clear air as a gas-sensing material. In an atmosphere where there may be polluting gas, the conductivity of the gas sensor raises as the concentration of the polluting gas increases. MQ-135 performs a good detection to smoke and other harmful gas, especially to ammonia, sulphide and benzene steam. Its ability to detect various harmful gas, its long life and low cost make MQ-135 an ideal choice for different gas detection applications. This module can be applied to harmful gas detecting devices. It has detection range: 10 - 300 ppm NH3, 10 - 1000 ppm benzene, 10 - 300 alcohol, heater voltage: 5.0V dimensions: 18mm diameter, 17mm high excluding pins, pins - 6mm high. Its common applications are domestic air pollution detector, industrial air pollution detector and portable air pollution detector [4].

**Figure 5.** MQ-135 air quality sensor

### 3.4 4 Channel 5V relay module with optocoupler isolation

This relay module facilitates to combine the processing power of the Arduino to devices that use higher current and voltage. It does so by providing four relays that are rated for 7A at either 28V DC or 10A at 125V AC. Each relay has a Normally Open (NO) and a Normally Closed (NC) contact. With these relays we can control appliances, motors, lights and other relays. For relay module inputs, the module is supplied with power via the pin labelled VCC and ground via the pin labelled GND. The relays are energized with low inputs to the IN1, IN2, IN3 and IN4 inputs. For relay module outputs, there are four relays and each of them provide dry contact outputs. Each relay provides a common (COM), normally open (NO) and a normally closed (NC) terminal [5].

**Figure 6.** 4 Channel 5V relay module with optocoupler isolation

### 3.5 Thing Speak

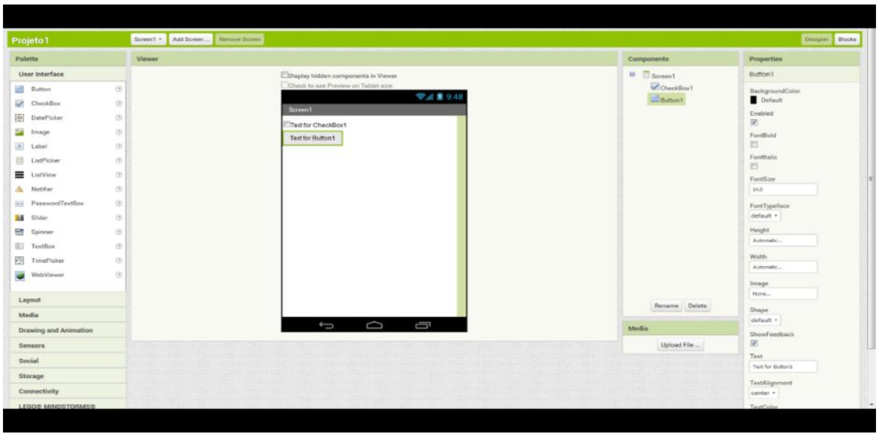
ThingSpeak is an IoT analytics platform service that allows us to aggregate, visualize and analyse live data streams in the cloud. We can send data to ThinkSpeak from our devices, create instant visualizations of live data and send alerts using web services like Twitter and Twilio. With MATLAB analytics inside ThingSpeak, we can write and execute MATLAB code to perform pre-processing, visualizations and analyses. ThingSpeak enables engineers and scientists to prototype and build IoT systems without setting up servers or developing web software. Key capabilities of ThingSpeak include:

- Configure devices to send data to ThingSpeak using a REST API or MQTT.
- Aggregate data on-demand from devices and third – party sources.
- Get instant visualizations of live or historical sensor data.
- Pre-process and analyze your collected data using integrated MATLAB
- Run your IoT analytics automatically based on schedules or events.
- Act on the data and communicate using third – party services like Twilio or Twitter

In order to send data to ThingSpeak using an Arduino, we require an Arduino with network connectivity either onboard or with a shield. We have an official library for ThingSpeak. This library needs to be installed and used by the Arduino device in order to send data to ThingSpeak. ThingSpeak requires a user account and a channel. A channel is where you send data and where ThingSpeak stores data. Each channel has up to 8 data fields, location fields, and a status field. You can send data every 15 seconds to ThingSpeak, but most applications work well every minute [6].

3.6 App Inventor for Android

App Inventor for Android is an open-source web application originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT). The original author is Google, Inc.. The developers include MIT Media Lab, MIT computer science and artificial intelligence lab. It is written in Java, Kawa, Scheme. The operating system is Android.it is an application software developer which allows newcomers to computer programming to create software applications for the Android operating system. It uses a graphical interface, very similar to Scratch and the StarLogo TNG user interface, which allows users to drag and drop visual objects to create an application that can run on Android devices. In creating App Inventor, Google drew upon significant prior research in educational computing as well as work done within Google on online development environments [7].



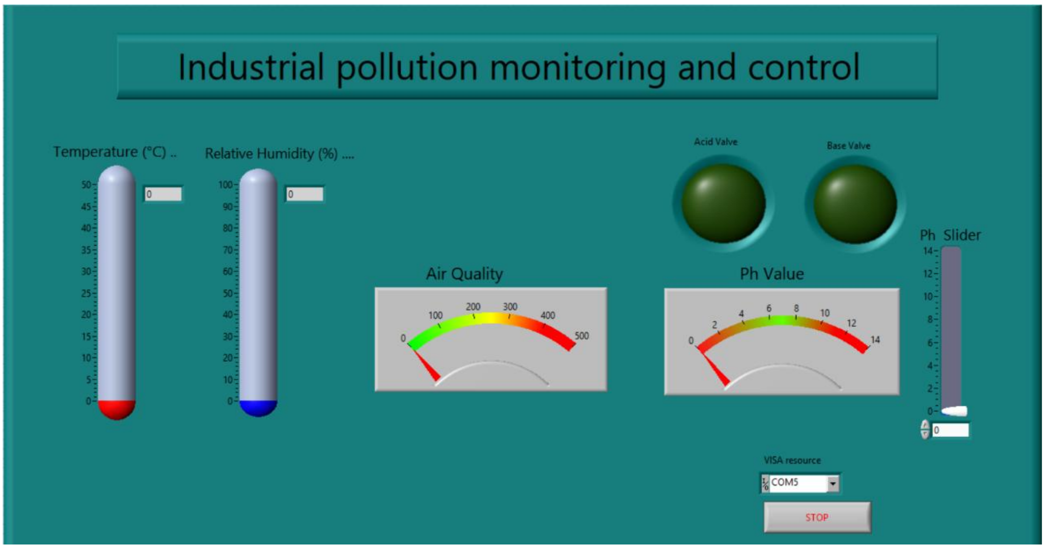
□ Display hidden components in Viewer
□ Check to see Preview on Tablet size.
□ Text for CheckBox1

Button1
BackgroundColor: ■ Default
Enabled: ☑
FontBold: □
FontItalic: □
FontSize: 14.0
FontTypeface: default ▾
Height: Automatic...
Width: Automatic...
Image: None...
Shape: default ▾
ShowFeedback: ☑
Text: Text for Button1
TextAlignment: center ▾

**Figure 7.** App Inventor for Android

### 3.7 LabVIEW

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) is a graphical programming language that uses icons to create applications. LabVIEW uses dataflow programming and the flow of data determines its execution order. LabVIEW also includes several applications to help us to quickly configure the computer-based instruments and build applications. We can build a user interface in LabVIEW by using a set of tools and objects. Code is then added by using graphical representations of functions to control the front panel objects. This code is contained in the block diagram. The block diagram resembles a flowchart in some ways. When the program is running, we use front panel for interaction. By controlling the program, we can change inputs, and see data updated in real time. A corresponding terminal is available on the block diagram for every front panel control or indicator. When a VI is activated, values from controls flow through the block diagram, where they are used as the functions on the diagram, and the results are passed into other functions or indicators through wires. National Instruments extends a variety of third-party devices to be interfaced with the LabVIEW software which includes an Arduino UNO. The VISA (Virtual Instrument Software Architecture) driver part of a LabVIEW enables us to connect with the Arduino UNO board with PC [9]. LabVIEW is used to fetch the temperature, humidity, air quality and pH parameters through Arduino. After the fetching session the following data are stored in the cloud database through ThingSpeak application service. The values can also be displayed on our mobile phones through App Inventor applications installed in the phones.



Industrial pollution monitoring and control

VISA resource: COM5

**Figure 8.** front panel view of LabVIEW

### 3.8 Hardware Setup

Fig. 9 shows the hardware setup of the proposed system. For temperature sensor the standard temperature has been set as $34^0$C. If the temperature goes higher than $34^0$C, the cooling fan is used to control the temperature until it comes down to the set temperature. For hardware assembled manually simulated pH sensor the normal range has been set as 6.5ppm to 7.5ppm. If the range goes lower than 6.5pm or higher than 7.5ppm, the corresponding solenoid valve is opened

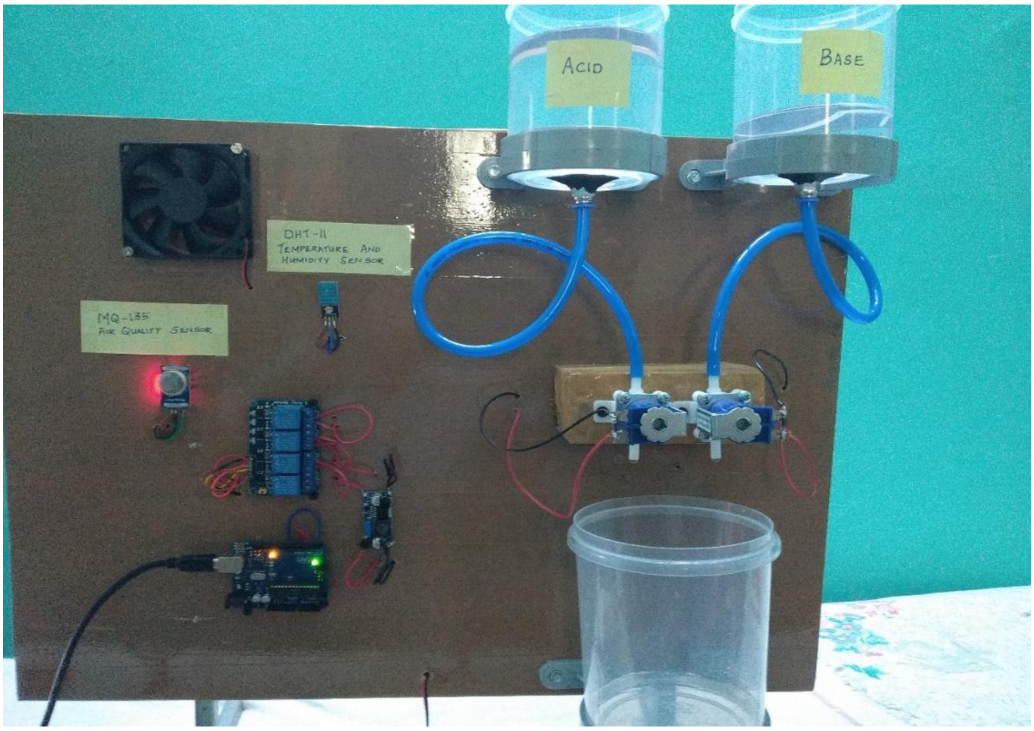


**Figure 9**. hardware setup

## IV. Results

The whole program is placed in a While loop with a STOP control. Multiple frames are created in sequence to place the programs created for particular parameters. Simultaneous processes can be monitored and controlled in LabVIEW. The Complete Front Panel design of the work is shown in figure 8. In this paper we have discussed the step by step procedure espoused to design the hardware and software of the project. The hardware has been implemented for acquisition of real time temperature, humidity, air quality and pH. Using LabVIEW relevant software has been developed for real time monitoring and controlling of the mentioned parameters. The results of the measured and studied parameters are displayed on the front panel which act as the graphical user interface, developed using user friendly indicators like thermometers, Boolean indicators and string outputs, etc. This work has a low-cost custom-built monitoring and controlling system [2]. However, this system can be considered as a low cost, customized system for industrial applications. Thus, this work can be modified to suit any industrial requirements. The proposed work starts with the monitoring section. The data is fetched from the sensors when request in form of string datatype is send from LabVIEW to Arduino UNO board. The Sensors are connected to the Arduino UNO board. In the coding section, when the serial input is given to the Arduino UNO board, it fetches the data from the sensor. The serial input is provided to the Arduino UNO board by LabVIEW [3]. Next, the data collected from the Arduino UNO board is sent to cloud storage database using the ThingSpeak analytical tool. And then using the App Inventor application the fetched data is displayed on mobile phone. In the controlling section: in case of DHT-11 sensor the set temperature is 340C, when the observed temperature is above it the fan is switched on by the signal received from LabVIEW. In case of manually simulated hardware assembled pH sensor the neutral range is kept between 6.5ppm and 7.5ppm and when the observed value is anywhere below 6.5ppm and above 7.5ppm, corresponding solenoid valves are opened and closed according to the control signals from LabVIEW. When the pH value is greater than 7.5ppm, the acid valve is opened. Fig. 10 shows the front panel view of LabVIEW when the acid valve is open. When the pH value is lesser than 6.5ppm, base valve is opened. Figure 11 shows the front panel view of LabVIEW when base valve is open. Once the parameter values are observed they are sent to the cloud database through ThingSpeak platform. The observed values can be viewed through this platform. Figure 12 sows the humidity values taken on a particular date. Fig. 13 shows the temperature values taken. Fig. 14 and Fig. 15 shows the pH and air quality values respectively.

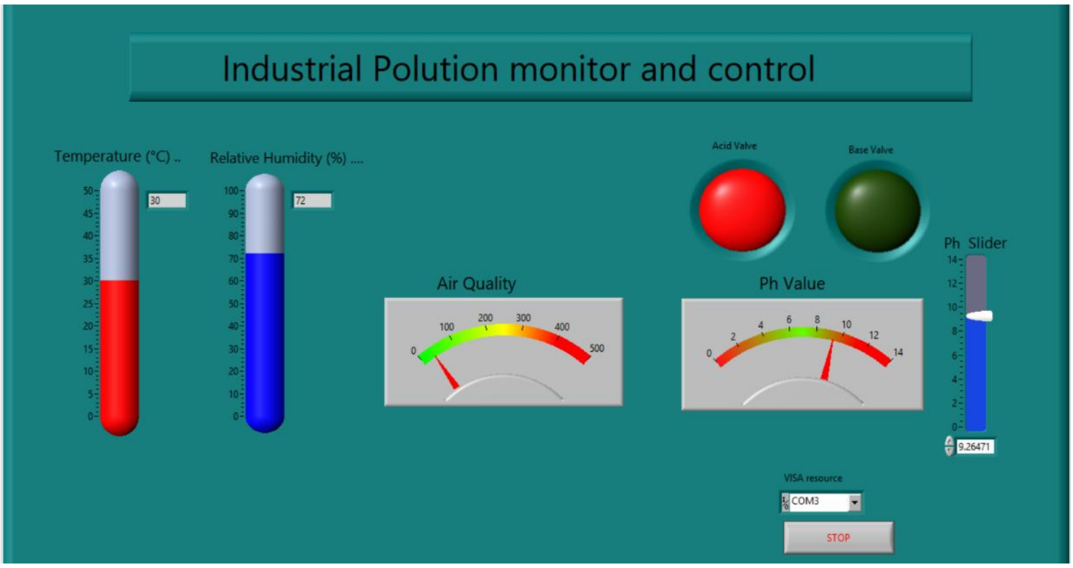


**Figure 10.** front panel view when acid valve is open

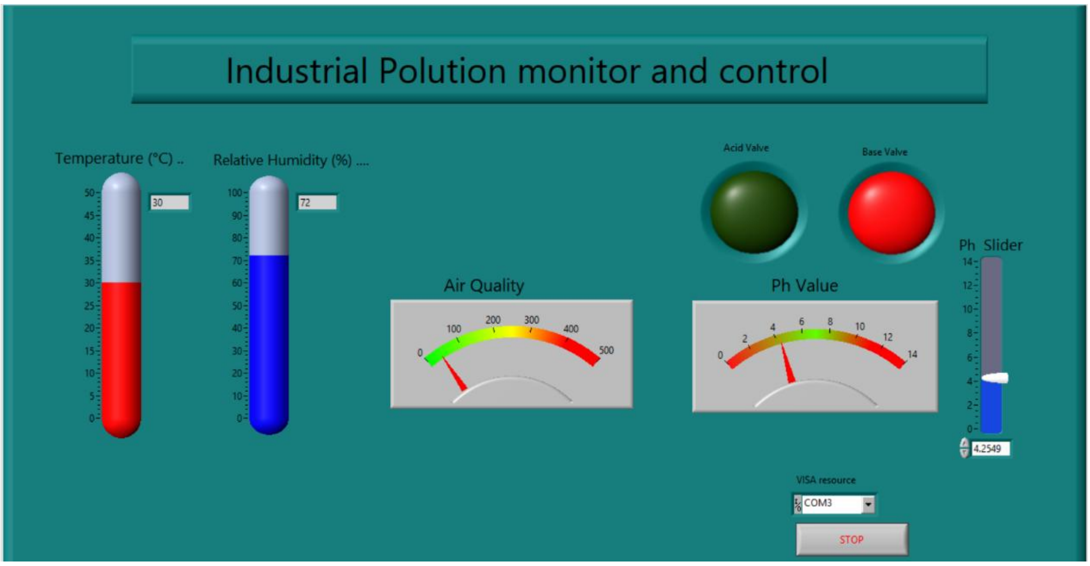


**Figure 11.** front panel view when base valve is open

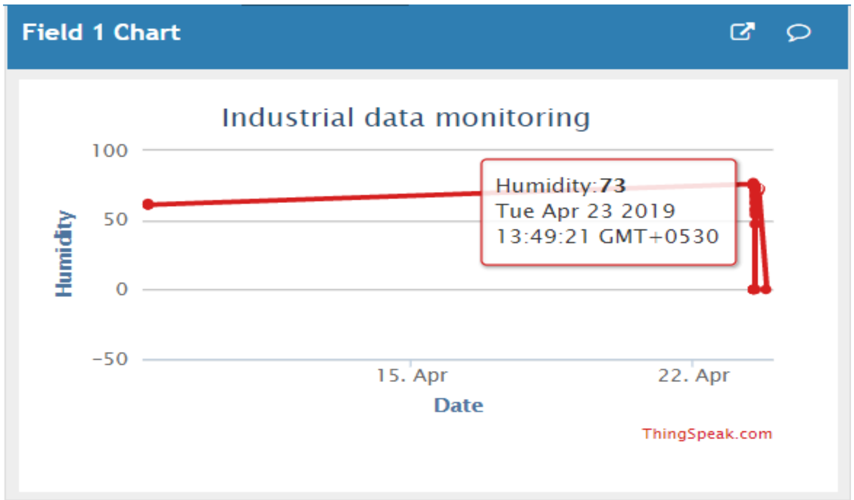


**Figure 12.** humidity value monitoring

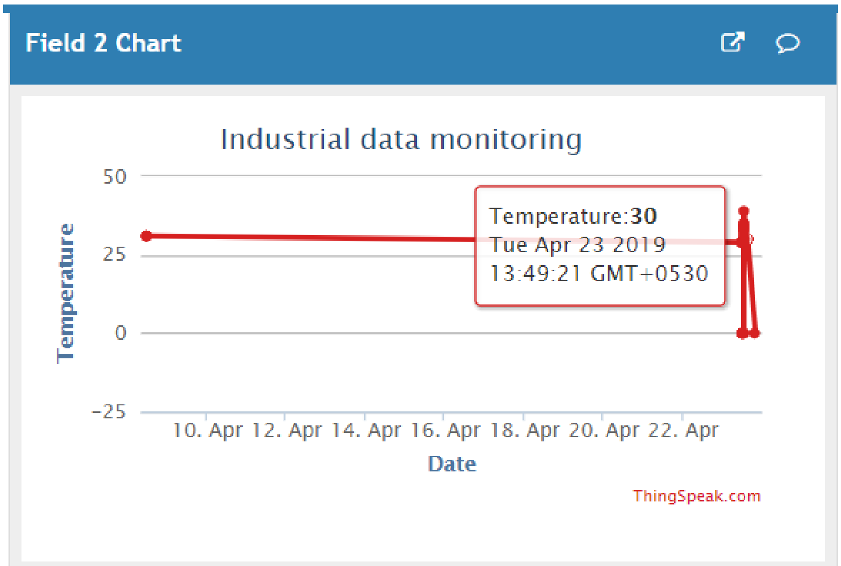


**Figure 13**. temperature value monitoring

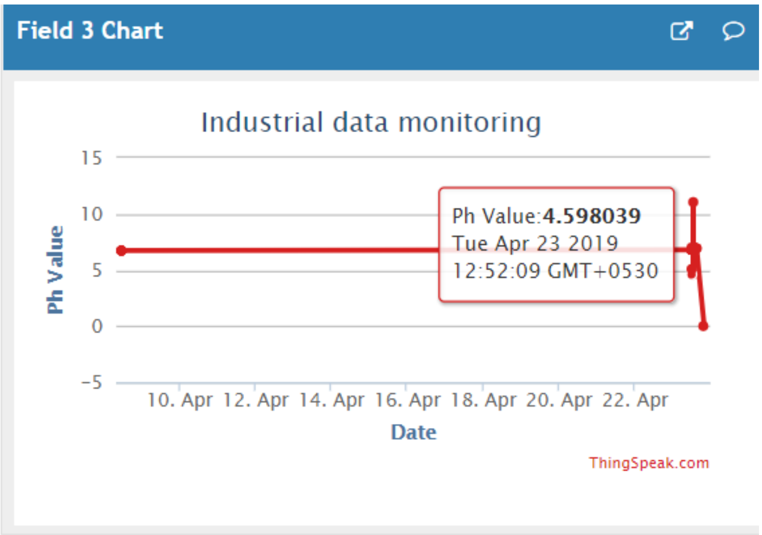


**Figure 14.** pH value monitoring

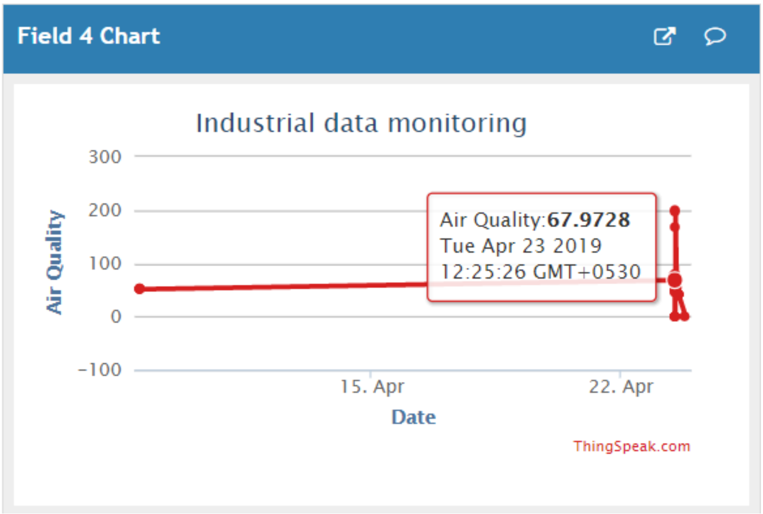


**Figure 15.** air quality value monitoring

## V. Conclusion And Future Scope

The domain of pollution monitoring and control is very substantial and this work is an experiment to minimize the problem of cost and regular inspections. Advanced LabVIEW is used for relieving these problems. The performance of the pollution monitoring and control system can be further developed by implementing sensors for controlling various pollutant parameters, thereby developing the industrial and natural environment. With this it can also prevent any hazardous situations in the industries and provides an effective control over the environmental pollution. LabVIEW software enables a practical applicability of using IoT with an easy graphical user interface platform for the designers and the users. The execution cost of this work is very economical as the sensors and the Arduino UNO board are easily available. As this system is monitoring only four parameters it can be expanded by considering more parameters that cause pollution especially by the industries. This proposed system gives opportunity of viewing the sensor outputs through internet. In future it can be made to control the pollutant parameters by giving commands from distance. Many pollutants do not have known sensors that sense them and if available they are very expensive. Therefore, building sensors for different parameters might be a very challenging task for future.

## References

### Journal Papers:

[1]. N.Kularatna and B.H.Sudantha, An environmental air pollution monitoring system based on the IEEE 1451 standard for low cost requirements. IEEE Sensors Journal, volume 8, April 2008 pp. 415422.

[2]. Yingxiang Qiu and Qijun Chen, The realization of cyclic serial port communication based on LabVIEW IEEE International Conference on System Science and Engineering (ICSSE), Shanghai,2014 pp. 46-49.

[3]. R. Kazi and G. Tiwari, IoT based Interactive Industrial Home wireless system, Energy management system and embedded data acquisition system to display on web page using GPRS, SMS & E-mail alert. International Conference on Energy Systems and Applications, Pune,2015 pp. 290-295.

[4]. https://components101.com/sensors/mq135-gas-sensor-for-air-quality

[5]. https://hacktronics.co.in/4-channel-relay-module/4-channel-5v-relay-module-with-optocoupler-isolation

[6]. https://thingspeak.com

[7]. https://en.wikipedia.org/wiki/App_Inventor_for_Android

[8]. https://www.arduino.cc/en/Guide/HomePage.

[9]. http://www.ni.com/en-in.html

[10]. https://www.electronicwings.com/sensors-modules/dht11

IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE) is UGC approved Journal with Sl. No. 4198, Journal no. 45125.

Chandni Mariann Thomas[1] Development of Pollution Monitoring and Control System Using Labview." IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE) 14.3 (2019): 70-79.