International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-06, Issue-04, July 2020

Pollution Monitoring System Using IoT and LabVIEW

Rucha Golwalkar[1], Kajal Shivgan[2], Parul Sindhwad[3]*

K.J. Somaiya College of Engineering[123], Mumbai[123], India[123]

rucha.g@ somaiya.edu [1], kajal.shivgan@ somaiya.edu [2], parul.sindhwad@somaiya.edu[3]

Abstract —**Air Pollution in India has become an alarming problem in current day scenario. The pollution levels have increased to such an extent that it has already started showing adverse effects on the human life and other living organisms. We have developed Pollution Monitoring System based on two strategies that will measure the level of harmful gases such as CO2, CO, smoke along with humidity and temperature of the surrounding environment. One of the two methods involves use of IoT which enables the user to monitor the pollution levels remotely. The system includes essential components to measure the parameters such as carbon monoxide, carbon dioxide, smoke level, humidity and temperature in the surrounding area of the system. Another method developed uses LabVIEW, where the pollution levels can be monitored automatically with the help of sensors.**

Keywords — Arduino, Blynk, Internet of Things, IoT Applications, LabVIEW, NodeMCUESP8266, Pollution Monitoring

I. INTRODUCTION

Many factors such as industrialization, urbanization, increase in population, etc. cause rise in the pollution levels. The environmental pollution has led to deterioration of the atmosphere, degradation of land, acid rains, global warming and many other severe problems. Out of all the categories of environmental pollution, air pollution can be highly disastrous as it can rapidly disseminate over large areas [7]. Thus, it has become the need of the hour to constantly monitor the levels of harmful gases in air such as carbon dioxide, carbon monoxide, smoke, NH3, etc. which contribute in deterioration of the air quality. One of the advantages of monitoring the real-time data measured using sensors is that the acquired data can be analyzed in order to take precautionary measures to reduce or control air pollution. In this project, not only is the real-time data displayed on the LCD display and the user's phone using sensors and Arduino, but also the data is stored into a Spreadsheet for future reference. The real-time as well as the stored data can be represented graphically to furthermore help the user to analyze it. Moreover, there is a buzzer connected to the system which acts as an alarm when the smoke level goes up. A notification is sent on the Blynk App of user's phone to make him alert about the risen smoke level in the environment surrounding the system. Thus, this system utilizes the technology of IoT for pollution monitoring and uses LabVIEW to further automate the system.

II. MOTIVATION

Continuous degradation in the quality of air has not only resulted in adverse effects on the health of human beings, but also has an effect on the climatic conditions. The Project based learning encouraged us to design a system that will monitor the levels of pollutant gases in the environment so that required actions can be taken in order to bring the pollution under control. The comprehensive approach of Project based learning enabled us to apply knowledge and skills to develop a portable system that can be used to alert the user whenever the harmful pollutants in the air rise beyond a permissible level.

III. RELATED WORK

A prototype has been designed by Meo Vincent C. Caya et al. to detect the air pollution using Raspberry Pi, dust and gas sensors. The project is designed to send an email notification to the registered email address whenever the pollution levels exceed a certain threshold. The email address to be registered can be entered in the Particulate Matter App which consists of Dust sensor Reading references. The email notification consists of the pollutant levels along with a warning message indicating the high level of gases in air [12].

Arnab Kumar Saha et al. have proposed a Raspberry Pi based pollution monitoring system to monitor the air and sound pollution levels along with temperature and humidity. The system enables cloud-based monitoring and includes SMS-based alerting system to alert the authorities regarding the undesirable increase in any of the parameters being monitored [6].

In order to monitor the Air Quality, a project has been developed by Poonam Pal et al. which will detect the pollutant gases in air with the help of Arduino, ESP8266

 DOI : 10.35291/2454-9150.2020.0520

and gas sensor. The IOT based system triggers an alarm in case the pollution levels rise beyond the critical level [13].

IV. PROPOSED SYSTEM

In this project we are going to make an IoT Based Air Pollution Monitoring System in which we will monitor the Air Quality over the Blynk Application using internet and will trigger an alarm when the harmful gases in air rise above a certain level, which means that there are large amount of harmful gases like CO_2, smoke, CO present in the air. It will show the air quality in PPM (Parts per million) on the LCD as well as on a mobile application so that we can monitor it very easily from anywhere. Another approach includes LabVIEW which is a system-design platform that has been used to store the acquired values for future reference and to represent the data graphically.

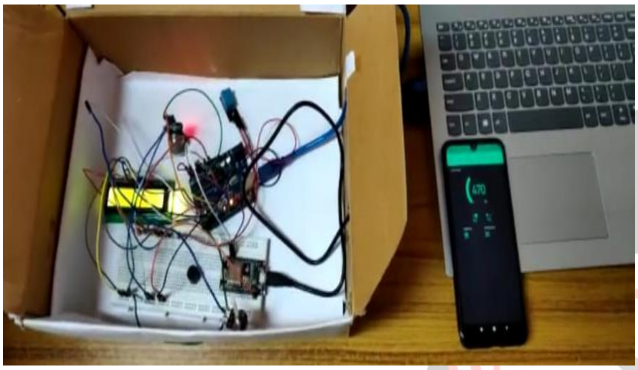


Figure 1. Developed System using IoT

Figure 1 shows the proposed system that has been implemented using IoT. The system consists of Arduino to which the LCD display, MQ7, MQ135 and DHT11 sensors along with a buzzer are connected. The Arduino is connected to the NodeMCUESP8266 for transferring data which then sends it to the user's mobile phone.

V. METHODOLOGY

In the developed system, the pollution monitoring strategies and techniques involve the implementation of Internet of Things and LabVIEW.

A. Calibration

The MQ-7 sensor is highly sensitive to Carbon Monoxide and thus, was a preferred choice to measure the level of carbon monoxide in the air using the proposed system. However, the sensor does not give precise readings until and unless it is manually calibrated. The MQ7 sensor needs to be preheated for minimum 48 hours at the voltage of 1.4-5 V [2]. The output of the sensor is dependent on the values of surface resistance Rs and load resistance R_L. The value of R_L is different depending on various kinds and various concentration gases. Thus, the sensor was calibrated to adjust the sensitivity by changing the value of load resistance by preheating. Similarly, the MQ135 sensor was used to measure the Carbon dioxide and smoke in air which also needs to be calibrated by preheating it at 1.4-5V for over 24 hours. The output of the two sensors was tested by checking the state of the Buzzer. The Buzzer turned on when the sensor was placed in a container filled with sensitive gas. On the other hand, the buzzer would turn off if the sensor was placed in clean air. The DHT11 sensor was the preferred choice for measuring humidity and temperature as it gives almost accurate values with a tolerance of±1℃.

B. Internet of Things

Internet of Things enables interconnection between smart devices which allows the devices to communicate and transfer data with one another without any human intervention [4]. Each and every device connected to form a wireless sensor network has a unique identifier. The interrelated devices can share the data acquired by the sensors in the system and send the collected data to the cloud using IoT gateways for further analysis or can locally analyze the acquired values. Thus, the technology makes it possible to bridge the gap between operation technology and information technology which gives more opportunities for improvements by analyzing the stored data.

C. Implementation of Internet of Things

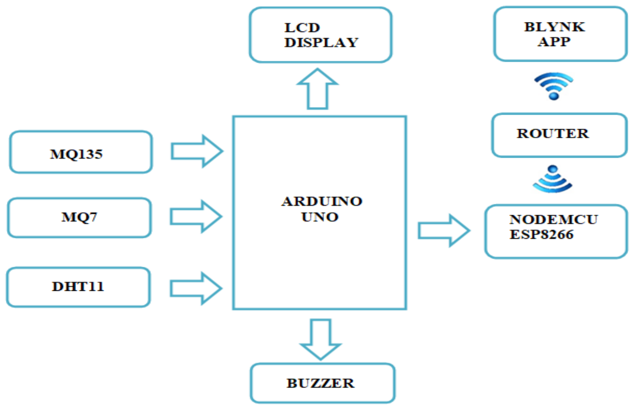


Figure 2. Block diagram of system using IOT

Figure 2 shows the block diagram of the proposed pollution monitoring system implemented using IoT. Arduino is used along with the sensors to display the measured data on the LCD and mobile phone thereby implementing the technology of IoT. The MQ-135 Gas sensor is used to measure NH_3, NOx, Alcohol, Benzene, Smoke, and CO_2 in air [1], [9]. The MQ135 sensor is connected to the analog pin A0 of Arduino UNO to measure carbon dioxide in PPM and smoke in the air while the MQ7 sensor is connected to the analog pin A1 of the Arduino which senses carbon monoxide in the environment. The range of CO which the MQ-7 sensor can measure is from 20 to 2000 PPM. Both the sensors have high sensitivity and fast response time which make it suitable for monitoring the air quality continuously. The DHT11 sensor is connected to the analog pin A2 of the Arduino which senses the percentage of humidity in air and the temperature in Celsius. All the data acquired by the MQ135, MQ7 and DHT11 sensors is

 DOI : 10.35291/2454-9150.2020.0520

displayed on a 16*2 LCD display. The buzzer is connected to the digital pin of the Arduino and whenever the smoke level rises above the threshold level the buzzer is turned on so that the user is made aware of the rising pollution level. The NodeMCUESP8266 is connected with the Arduino for the implementation of IoT. ESP8266 Wi-Fi module gives the system access to Wi-Fi or internet. Arduino UNO sends these values to NodeMCUESP8266 using I2C protocol [5].

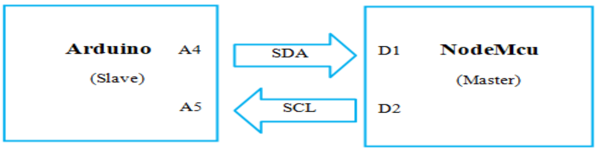


Figure 3. Implementation of I2C protocol

In Figure 3, I2C (Inter Integrated Circuit) protocol has been implemented by connecting the Arduino to NodeMCUESP8266. I2C is a two wire interface protocol which supports multi master and multi slave communication. Master provides clock signal to slave via SCL line thereby enabling synchronous communication. It is a single ended protocol using which Arduino sends values to the NodeMCUESP8266 via SDA line which further sends the values to the user's phone using Wi-Fi. Thus, the real-time values of gases along with humidity and temperature can be viewed on the user's phone and the air quality can be monitored continuously. When MQ135 sensor senses smoke, it sends a signal due to which the buzzer is turned on and the user is alerted about the rising smoke levels. Additionally, the user gets a notification through the Blynk Application on the user's mobile phone regarding the smoke detected in the area near the system so that one can take the required action.

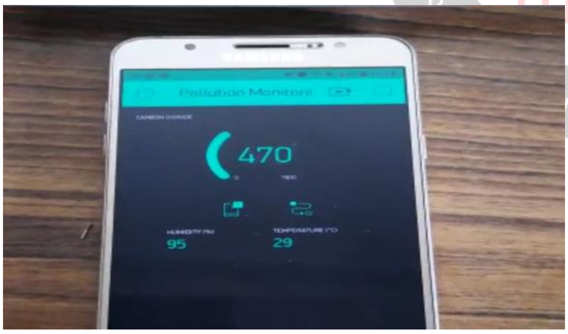


Figure 4. Result on Blynk application

In Figure 4, the Blynk application on a mobile phone shows humidity as 95% and temperature as 29°C.The real-time values are sent to the app using Internet. A notification is sent to the App alerting the user in case the system senses smoke or the pollution in air increases above a certain level which can be harmful for human health.

Figure 5. Result on LCD Display

Figure 5 shows a 16*2 LCD display with the temperature displayed on it in Celsius. The temperature is measured using DHT11 sensor and the acquired value is displayed on the LCD display using Arduino.

D. LabVIEW

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) is a system-design platform and development environment from National Instruments which is used for visual programming. There are three components of LabVIEW namely, front panel, block diagram, connector pane. LabVIEW enables graphical programming using a block diagram which contains the graphical source code and the blocks are placed as objects on the front panel which is a user interface. This development environment can be easily integrated with hardware components such as Arduino which can be used to design systems for a variety of applications.

E. Implementation of LabVIEW

This method to monitor the air quality involves creating an interface between Arduino and LabVIEW where in the acquired values are stored in an Excel sheet and plotted on a graph thereby making it easier for analysis of the pollution levels in the environment.

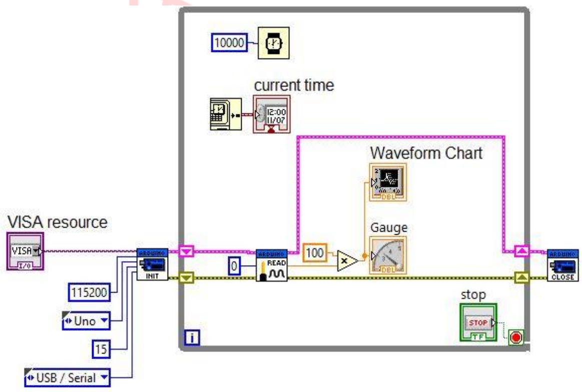


Figure 6. Block diagram of system implemented using LabVIEW used to acquire real- time data from Sensors

Figure 6 shows the block diagram of the system developed using LabVIEW. We start by uploading the code consisting LIFA (LabVIEW Interface for Arduino) commands for the interfacing Arduino UNO with LabVIEW into the Arduino UNO board [14]. The level of Carbon Monoxide in the air is measured using the MQ7 sensor connected to the Adruino board. The 'Arduino init' block is used to initialize the Arduino UNO board with the LabVIEW environment. After selecting the appropriate COM Port, the Baud rate is set to 115200. The third input to the 'Arduino init' block indicates the Board type while the fourth input indicates the number of bytes per packet. The data input line of 'Arduino Read' block is further connected to the data output line of the previous block which is used to read the analog values measured using the MQ7 sensor. The second pin of this

 DOI : 10.35291/2454-9150.2020.0520

block indicates the pin number i.e. analog A0 pin of the Arduino UNO to which the sensor is connected. A 'while loop' is applied to blocks to enable the system to read the input values from the sensor continuously. The Error control lines of all the blocks are connected together which display the type of error in case any error occurs. The measured values are displayed on the Front Panel of LabVIEW on the Gauge as well as on the Graph every 10 seconds which is set as the sampling rate. The sampling rate can be increased or decreased according to the requirement of the user. Thus, the variations in the real-time levels of gases can be easily viewed on the graph. The readings acquired using the sensor are saved every 10 seconds in an Excel file along with the corresponding date and time at which the values were measured.

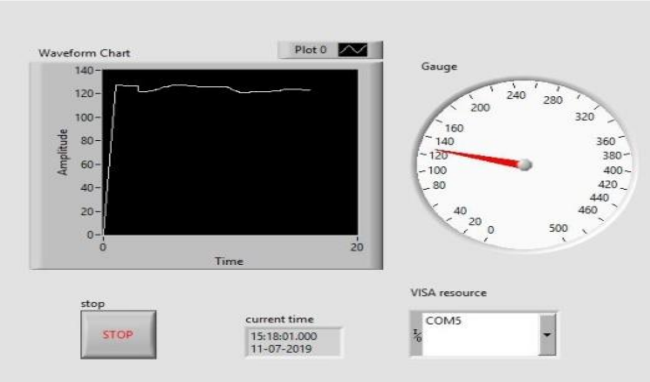
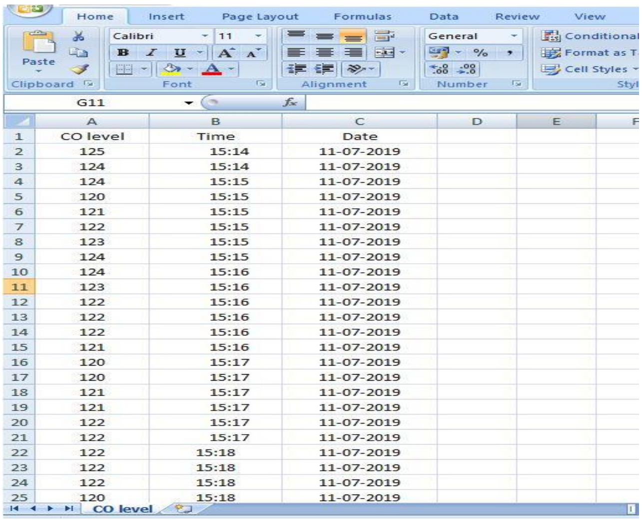


Figure 7. Result on Front panel of LabVIEW

In Figure 7, the real-time value of Carbon Monoxide measured by MQ7 sensor is displayed on the Waveform Graph and Gauge on the Front Panel of LabVIEW. Current date and time is also displayed on the Front Panel to make the analysis easier.

The data acquired using MQ7 sensor can be sent to an Excel file by right clicking on the Front panel and selecting the 'Export data to Excel' option under 'Export' [3]. Use of this function enables the system to export the data plotted on the graph to the Excel sheet.



	A	B	C
1	CO level	Time	Date
2	125	15:14	11-07-2019
3	124	15:14	11-07-2019
4	124	15:15	11-07-2019
5	120	15:15	11-07-2019
6	121	15:15	11-07-2019
7	122	15:15	11-07-2019
8	123	15:15	11-07-2019
9	124	15:15	11-07-2019
10	124	15:16	11-07-2019
11	123	15:16	11-07-2019
12	122	15:16	11-07-2019
13	122	15:16	11-07-2019
14	122	15:16	11-07-2019
15	121	15:16	11-07-2019
16	120	15:17	11-07-2019
17	120	15:17	11-07-2019
18	121	15:17	11-07-2019
19	121	15:17	11-07-2019
20	122	15:17	11-07-2019
21	122	15:17	11-07-2019
22	122	15:18	11-07-2019
23	122	15:18	11-07-2019
24	122	15:18	11-07-2019
25	120	15:18	11-07-2019

Figure 8. Excel sheet where acquired values are stored

Figure 8 shows the values of CO in air acquired using MQ7 sensor, saved in the Excel sheet. Next to the CO level are the corresponding date and time at which the carbon monoxide was measured by the sensor. Thus, one can refer to the Excel file for future analysis.

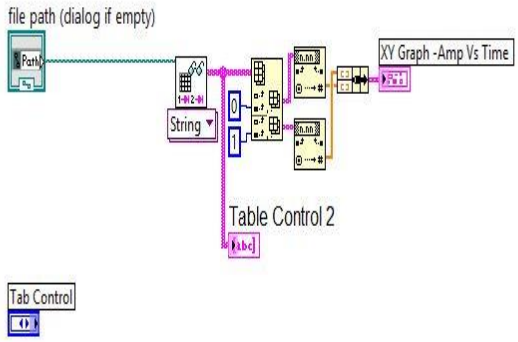


Figure 9. Block diagram of system using LabVIEW used to plot the stored data

Figure 9 shows the block diagram designed in LabVIEW to plot the data in the spread sheet on the Graph with respect to date and time to analyze the pollution levels. The path of the Excel file whose data is to be plotted on the graph is copied in the 'File path' block. The next block is used to load the transposed file. 'Table Control' is used to create a table on the Front Panel of LabVIEW consisting of values from the Spreadsheet. The Amplitude i.e. CO level is plotted with respect to Time on the XY Graph. Thus, the stored values from the Spreadsheet can be retrieved easily to analyze the CO levels and the hour of the day when the pollution was highest, can be noted by observing the waveform on the graph.

VI. RESULT ANALYSIS

The conventional methods for pollution monitoring include using the analytical instruments which provide precise analysis. However, they are of large size, heavy weight and expensive [7].This makes the installation of such systems in industrial or urban areas difficult.

Table 1. Comparison between Analytical instruments and Gas sensors [7], [8].

Features	Analytical Instruments	Gas sensors
Cost	Very high	Low
Size	Bulky	Compact
Rigidity	Fragile	Rigid
Mass Production	Difficult	Easy
Measurement	Instantaneous	Continuous

Table 1 shows the comparison between Analytical instruments and Gas sensors. It is clear that using MQ7 and MQ135 sensors for measuring CO and CO2 respectively, makes the system more compact, light-weight and less expensive. Thus, the system can be installed easily at any location.

 DOI : 10.35291/2454-9150.2020.0520

IoT based system makes it possible to monitor the air quality remotely on user's mobile phone using Internet as shown in Figure 4. A notification is sent on Blynk Application if the CO or CO2 levels rise above the threshold or if the system senses smoke. An alarm is triggered at the same time indicating the excess amount of harmful gases present in the environment surrounding the system.

LabVIEW provides a virtual environment to monitor the pollutant gases in the environment simply by observing the Front panel shown in Figure 7. Interface of Arduino with LabVIEW is easy and the compactness of the system is advantageous. The reduced cost of the system especially gives it an edge over the conventional methods.

Table 2. Comparison between measured and desired values of parameters monitored by the proposed system

Parameters	Measured values	Desired values
Temperature (°C)	30	29
Humidity (%)	95	85
Carbon Monoxide (PPM)	120	112

Table 2 shows the comparison between the values measured by the proposed system and the desired values acquired from 'Google weather' at 15:18 on 11/07/2019. Thus, the system developed measures the amount of Carbon monoxide and humidity in air and the temperature in the surrounding area with sufficient accuracy which will enable in monitoring the air quality, alerting the user in case the pollution levels rise and storing as well as retrieving data for analysis.

VII. FUTURE SCOPE

Integration of several different sensors with the Arduino will enable the system to measure more number of parameters such as sound, dust, etc. A wireless sensor network can be developed for measuring the air quality which will provide a large geographical coverage. The sensor data can be sent and stored on ThingSpeak cloud using IoT. LabVIEW can also be used to send notification to the user in case smoke is sensed in the surrounding environment of the system. Machine learning can be implemented which would help us to predict the weather conditions and forecast the environmental data well in advance. Implementing a machine learning-based system can enable designing of a model which can differentiate between industrial areas and cities. Thus, the algorithm can be used to process the collected data in a distributed fashion.

VIII. CONCLUSION

Air pollution being such an important issue in India has made it mandatory to continuously monitor the air quality in industrial as well as urban areas in order to avoid side effects on human health. We have successfully designed a low-cost portable system using IoT as well as LabVIEW to measure the concentration of harmful gases in air along with temperature and humidity. Both the methods provide quick-response systems and enable continuous monitoring of the data. IoT enables the user to view the pollution levels of the area where the system is installed from anywhere. Furthermore, LabVIEW reduces the hardware requirement by enabling visual programming. The real-time as well as stored data can analyzed to know the air quality in the nearby area of the system.

REFERENCES

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

[2] https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf.

[3] http://read.pudn.com/downloads522/sourcecode/labview/2164424/labview%20to%20excel.pdf.

[4] https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT#:~:text=The%20internet%20of%20things%2C%20or,human%2Dto%2Dcomputer%20interaction

[5] https://www.electronicwings.com/nodemcu/nodemcu-i2c-with-arduino-ide

[6] Arnab Kumar Saha, Sachet Sircar, Priyasha Chatterjee, Souvik Dutta, Anwesha Mitra, Aiswarya Chatterjee, Soummyo Priyo Chattopadhyay, Himadri Nath Saha, 'A Raspberry Pi Controlled Cloud Based Air and Sound Pollution Monitoring System with Temperature and Humidity Sensing'.

[7] Duk-Dong Lee, Dae-Sik Lee, 'Environmental Gas Sensors', IEEE SENSORS JOURNAL, VOL. 1, NO. 3, OCTOBER 2001.

[8] Gagan Parmar, Sagar Lakhani, Manju K. Chattopadhyay, "An IoT Based Low Cost Air Pollution Monitoring System.", International conference on Recent Innovations is Signal Processing and Embedded Systems (RISE-2017) 27-29 October, 2017.

[9] Harsh N. Shah, Zishan Khan, Abbas Ali Merchant, Moin Moghal, Aamir Shaikh, Priti Rane , 'IOT Based Air Pollution Monitoring System', International Journal of Scientific & Engineering Research Volume 9, Issue 2, February-2018.

[10] Kiruthika.R, A.Umamakeswari, 'Low Cost Pollution Control and Air Quality Monitoring System using Raspberry Pi for Internet of Things', International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017).

[11] L.Ezhilarasi, K.Sripriya, A .Suganya, K.Vinodhini, ' A System For Monitoring Air And Sound Pollution Using Arduino Controller With Iot Technology' , International Research Journal in Advanced Engineering and Technology (IRJAET).

[12] Meo Vincent C. Caya; Angeline P. Babila; Alyssa Moya M. Bais; Seoi Jin V. Im; Rafael Maramba, 'Air Pollution and Particulate Matter Detector Using Raspberry Pi with IoT Based Notification".

[13] Poonam Pal, Ritik Gupta, Sanjana Tiwari, Ashutosh Sharma, 'IOT BASED AIR POLLUTION MONITORING SYSTEM USING ARDUINO', International Research Journal of Engineering and Technology (IRJET).

[14] Tommy Sugiarto, Pusat Teknologi Penerbangan, Lembaga Penerbangan, Antariksa Nasional, "LabVIEW-Arduino Interfacing for Data Acquisition and Telemetry System".

 DOI : 10.35291/2454-9150.2020.0520