Indonesian Journal of Electrical Engineering and Computer Science
Vol. 10, No. 1, April 2018, pp. 43~49
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v10.i1.pp43-49

Implementation of Monitoring System for Air Quality using Raspberry PI: Experimental Study

Abdulrahman Abdullah Alkandari[1], Samer Moein[2]
[1]Computer Department, Basic Education College, PAAET, Kuwait
[2]Electrical and Computer Engineering, University of Victoria, Canada

Article Info

Article history:

Received Jan 9, 2018
Revised Mar 2, 2018
Accepted Mar 18, 2018

Keywords:

Air Quality
Environment Monitoring System
Raspberry PI

ABSTRACT

Because of rising dependency on fossil fuels, and rising amounts of toxic gases in the environment, it found that people are in need of a way to ensure the safety specifically those that live in cities. An approach is suggested in this paper, that is economical yet affords good detection, and can give accurate readings that can be analyzed and manipulated, and can even provide warnings through sending emails. These requirements are found in the Raspberry PI when it hooked up to the sensors. This paper was focused on few dangerous gases such as Carbon Monoxide (CO), Nitrogen Dioxide (NO2) and other gases. The results in this paper showed that some gases, specifically CO, may be a problem in Kuwait as it is always slightly below the warning level. The success with the Raspberry PI and the results were encouraging to open the way for much improvement in the future.

Corresponding Author:

Abdulrahman Abdullah Alkandari,
Computer Department, Basic Education College,
PAAET, Kuwait,
Email: Aam.alkandari@paaet.edu.kw

1. INTRODUCTION

Nowadays, the Wireless Sensor Network (WSN) is considered to be an essential technology that is used in many fields and projects such as monitoring the water quality, engine emissions and air pollution and metrological. It is made of nodes, every node or more than one node is connected to one sensor. The WSN had many advantages which are sufficient data, temporal accuracy, flexibility, low power consumption, less implantation cost and so on [1]. The wireless sensor network can be an excellent device to observe air quality. It gathers air quality data automatically.

Due to depending on industrial and fossil fuel in over the world, toxins and unhealthy gases and radiation that in many cases cannot be detected by smell or sight surround by humans. These gases and radiations have many potential dangers such as lung damage, skin damage, and even cancer and death, not to mention the possibility of explosive gas damage [2].

This paper presented an approach to prevent such dangers in a reliable and affordable way. Many hardware can be used to implement this work such as the Arduino, PandaBoard ES, Cubieboard, Hackberry, Raspberry Pi and so on. This project aims to prevent such dangers by using the Raspberry Pi because of its features and characteristics, which are:

1) Relatively low-cost.
2) Easy to program and understand.
3) Portable computer and pocket size device.
4) Many sensor nodes can be hooked up to it.

This paper is organized as follows. The second section is the problem description. The third section is a description of some related works briefly. The fourth section views the design of the system. Then, the fifth section defines the system implementation. The sixth section shows the results of the project. Finally, the seventh section will conclude the paper and talk about some future work.

2. PROBLEM DESCRIPTION

Many gases in our environment are harmful and even deadly. Most of them can be prevented easily and cost-effectively. This project focuses on a few of these gases.

One of the most immediately dangerous gases is Carbon Monoxide (CO), which is odorless, tasteless and colorless gas. There are many sources of CO such as Internal Combustion Engines, ovens, and stoves; it is also produced by the incomplete burning of fuels. Carbon Monoxide is one of the most common types of deadly gas poisoning in the world; it combines with red blood cells making them unable to carry oxygen to the rest of the body and even very low concentrations can lead to seizures and fatalities.

Another dangerous gas is Nitrogen Dioxide (NO2), which is a common air pollutant with millions of tons produced every year. While the acrid smell of the gas makes it easy to avoid, it is nonetheless dangerous because it can have long-term effects even from short-term exposure. Exposure to this gas anywhere from 30 minutes to 24 hours can have adverse health effects on respiratory airways such as airway inflammation and increased asthma symptoms in people that suffer from it.

Other gasses are dangerous because they are flammable and explosive, such as Methane, which is odorless in low concentration and it is highly explosive and could lead to suffocation. Another flammable gas is Hydrogen, which can be very explosive at specific concentrations.

These gasses are possible threats facing every household, but protection from these gases (through companies that offer devices and sensors for protection) are costly and are out of the reach of many people. That leads to present this project to find a solution for this problem by protecting against these dangerous gases and making sure houses are safe, while at the same time keeping costs low and affordable.

3. RELATED WORK

Hussein in his research has explained that the goal of the project is to use common and available sensors to create engine emission detecting computer and give a judgment on whether an area is above or below the required emission levels. The main method of measurement is using a Carbon Monoxide sensor [2].

Arizaga, Calleja, Hernandez, and Benitez presented a project, which is built around an automated control system that is used to ensure sterilization of biological material. This system is separated into four modules that are flame detection, pressure sensor, gas control and PU modules. The system is controlled centrally by an Arduino board using the C++ language. The system was tested to ensure that it is able to maintain the different environmental variables needed for proper sterilization [3].

Sayantani, Sridevi, and Pitchiah explained in their paper that the project's stated aim was to develop a wireless solution for indoor air quality monitoring using Java. The proposed system is to integrate environmental parameters like temperature, humidity, gaseous pollutants and aerosol/particulate matter into a controlling HVAC (Heating, Ventilation, and Air Conditioning) system in a smart building [4].

In 2013, Preethichandra has demonstrated that the research objective is the development of an ultra-low power microcontroller for sensor nodes to gather information on indoor air quality. The system measures Carbon Dioxide, Carbon Monoxide, Propane and methane using industrial grade gas sensors. The research was verified with actual measurements under real-life situations [5].

In 2014, Ferdoush and Xinrong designed an environmental monitoring system based on Arduino and Raspberry Pi. It is an inexpensive and scalable wireless sensor network system that can be used in many environmental monitoring applications [6].

In 2016, Sonali and Venkatasubramanian presented a Raspberry-pi based on IOT system that is able to monitor the environmental parameters, such as carbon dioxide, temperature, humidity, and carbon monoxide. The system uploads the recorded parameters to the cloud for the user, who can track the condition of the environment [7]. As for Balasubramaniyan and Manivannan, they described a monitoring system for Air quality that uses IoT and Raspberry Pi. The system goal is to remotely monitor the air quality in a specific area of interest, and it is a user-oriented system [8]. In addition, Jadhav et al. built a Raspberry-pi based environment monitoring system. It is a universal and scalable system that can monitor the environmental parameters such as humidity CO2 concentration in air, temperature, and more [9].

By 2017, Rohani et al. used Arduino microcontroller with Open Platform Communications (OPC) in designing a monitoring system to control and monitor the CO2 emission in the industrial environment. System testing was performed in a lab under real-time CO2 emissions measurement, and system implementation indicated a successful application [10].

A real-time monitoring system for agriculture's weather was proposed by Susanto et al., which used parallel processing, Arduino, and Raspberry Pi. The system was faster and 50% efficient than the systems with single processor [11]. Furthermore, Mekki and Abdallah implemented a monitoring system for greenhouse controlling using Wireless Sensor Networks (WSN). Temperature, humidity, and soil parameters were controlled by the proposed system [12].

Abd Allah et al. presented a data logger system that can be used in applications of environmental monitoring. The system is universal, it is built using Arduino and LabView software, and can be used as a standalone device. It can monitor and record a massive amount of data [13].

Baharun et al. used the SICK sensor in conducting a study on measuring and analyzing the air quality inside the Meru Menora Tunnel. The study helped in controlling the ventilation fan system, which operates efficiently depending on the gases concentration level in the tunnel and thus eliminates the power consumption [14]. Montanaro et al. presented an air pollution monitoring system for the city, called SmartBike. The system exploited a network of bicycles to provide several services to citizens, such as bikes location detection, antitheft, traveled distance, and monitoring of air pollution [15].

4. SYSTEM DESIGN

4.1. Hardware

The proposed system is designed to upload readings provided by many sensors connected to a Raspberry Pi, providing data analysis and early warning. In this system, various software and hardware are used.

1) Raspberry Pi: A very small computer (single-board computer) can be useful in different electronics projects. It also can be used like a desktop PC, which can perform many things such as games, spreadsheets, processing and plays high-definition videos. It provides many input and output ports such as LAN, GPIO, HDMI, and USB [16].
2) Breadboard: A solder-less electronic board used for prototyping circuits. It is reusable, making it ideal for creating temporary circuit designs to experiment on. It consists of two main areas, the first area is the area of the board where sensors and components can insert on it, and the second area is the top and bottom strips used to distribute the positive and negative voltages that power the components.
3) T –cobbler: A piece of hardware used to connect the Raspberry Pi to the breadboard easily, allowing easy and labeled access to the Raspberry Pi's GPIO ports. It acquires some soldering.
4) MCP3008: An analogue-digital converter (ADC) used with some of the project's sensors to convert their signal to digital form.
5) Sensors:
   a) DHT22 - Humidity & Temperature sensor. An ADC is not required for this sensor because it is digital.
   b) TGS 2600 - General Air Quality: This sensor consists of a mixture of gases, which are methane, isobutane, ethanol, and hydrogen. The output of this sensor is analogue, so an ADC is necessary.
   c) MICS-2710 – NO2 Concentration: The output of this sensor is analogue; as a result, an ADC is required. The concentration of NO2 can be dignified by using this sensor.
   d) MICS-5525 - CO Concentration: This sensor might be the most important one because as mentioned in this paper the CO gas has no smell so that it can be very dangerous, and this sensor measures the concentration of CO. This sensor has a wide detection range.

4.2. Software

1) Raspbian: A free and open source Operating System that is based on Linux Debian. It is optimized for the Raspberry Pi, making it light yet powerful.
2) Python: A high-level programming language that lets you work more rapidly and integrate your systems more effectively. It is the main method that used to interact with the GPIO (General Purpose Input Output) port. Like the Raspbian, Python is free and open source software.
3) Dependencies Software:
   a) Pip: Allows the installation of Python modules easily, reducing time waste and complexity.
   b) Git: A version control software that allows access to Git Hub and downloads files.
   c) RPi-GPIO: Software that allows the Pi to interface with the GPIO ports.
   d) Python-smbus: Let Python use the Pi's I2C interfaces

5. IMPLEMENTATION

1) Configuring the Raspberry Pi:

The Raspberry Pi comes preloaded with an image of many Linux-based operating systems. Raspbian is used in this project because it is optimized for the Raspberry Pi and it both light and powerful. After installation was complete, Apt-Get was used to install as well as update most of the required programs and their dependencies. Then, it finished off by using Git to download the needed Python code from GitHub.

2) Assembling the Hardware:

Sensors, resistors, ADC and T-Cobbler were assembled into the breadboard. Authors have followed steps provided online or checked the Data Sheets provided by the manufacturer.

In Figure 1, the Raspberry Pi is connected to the breadboard through a ribbon cable and a T-Cobbler. The sensors were attached to the breadboard and then to the T-cobbler for completing the connection. An ADC was also attached to the breadboard, and some sensors were connected to it. Male and female jumper wires were used to connect all the parts on the breadboard to each other, and to get the readings of the sensors.

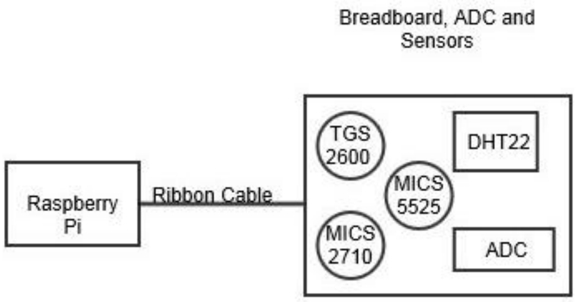

Figure1. Assembling Hardware

3) Algorithm (Coding)

Many steps were done to finish the coding will be explained thoroughly. Also, it will be shown clearly in Figure 2.

a) Settings and preparation:

The program begins with object initializer that is used as a blueprint for objects; then it prepares the rounding function and gathers basic settings. The program then checks if it can access the directory for the configuration file (AirPi.cfg), if it was unable, the program would display an error and end. Otherwise, it will gather all the basic settings. At this point, it will be defined a new function that sends emails; this function will be used within the loop.

Then, the program will check the ADC settings and tries to find which ADC is used. If the settings are incorrect, the program will end displaying an error. If all goes well to this point, the program will check to see which sensors are enabled and gets their settings. After that, the program would set up the excel objects and readies them for the loop, as shown in Figure 2.

b) Data Gathering:

The program works inside an infinite loop that breaks only through keyboard interrupt. It starts by getting the readings of individual sensors and writes them to the excel objects, then, it compares them to the danger level readings and sends an email using the function if the readings are too high or too low.

Finally, it appends the data to display on the screen later. This is repeated for all the sensors and once all that is done the data is displayed on the screen.

c) Saving and ending:

Once the program detects a keyboard interrupt it will start writing all the formula needed to find the minimum, maximum and average next to all the readings using a loop. When that is done, the excel objects are saved into individual excel files.

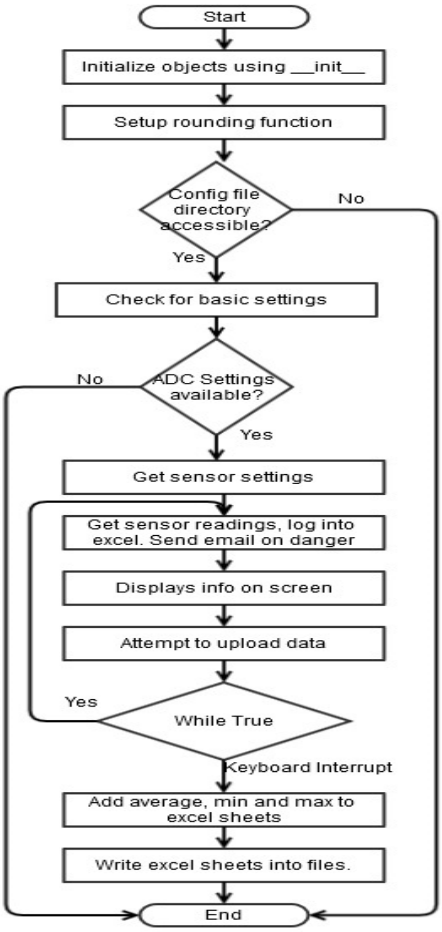

Figure 2. Flowchart for Writing and Plotting Data to Excel and Email

6. RESULT AND DISCUSSION

The system has been tested to detect pollutant gases, temperature, and humidity as mentioned in this paper. For each sensor, there are two factors: the y-axis is for the sensor's readings, and the x-axis is for the time. These readings were exported to excel, and line charts were made for them, these line charts display the actual recordings as well as minimum, maximum, and the average of that figure. The following readings have been taken over a period of around six days.

Figure 3 shows the recorded temperature of the room, two factors are there, the y-axis that represents the temperature of the room, and the x-axis is the time per 60 seconds. The average was 18.43447878, the maximum was 20.9, and the minimum was 16.9. The program was set to warn if the temperature ever gets above 30 or below 15.

Figure 4 shows the humidity readings; the sensor was set to warn if humidity levels got above 60%. In this line chart the humidity of the room which is signified by the y-axis, and the x-axis is the time per 60 seconds. The average was 49.44858164, the maximum was 59.4, and the minimum was 34.

Figure 5 shows the CO readings. The y-axis shows the CO concentration in the room while the x-axis shows the time per 60 seconds. The average of the Co was 40.41416671, the maximum was 40.45428014, and the minimum was 36.33883391. It is set to warn at above 50ppm.

Figure 6 shows the readings of NO2 which is set to warn above 2ppm. The average that got was 0.1446938, while the maximum was 0.189887086 and the minimum was 0.094311863. Same with all readings, the y-axis demonstrates the NO2 concentration in the room whereas the x-axis demonstrates the time per 60 seconds.

In figure 7, since the air quality sensor (TGS2600) gave readings for a mixture of gases. The y-axis shows the reading of these gases and the x-axis shows the time per 60 seconds. The average was 7107.803264, maximum was 12749.48903, and the minimum was 5751.422858. It was hard to find a warning level for this sensor, and so it set to send an email at every reading for testing purposes. More research needs to be made on this sensor to set the levels.

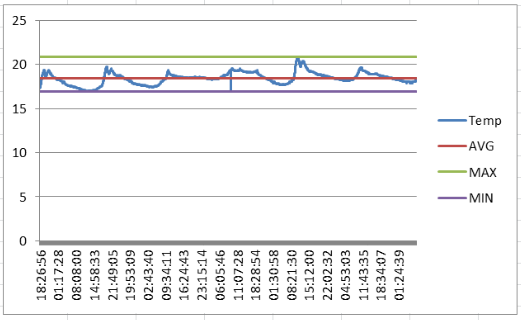

Figure 3. The Results for Temperature

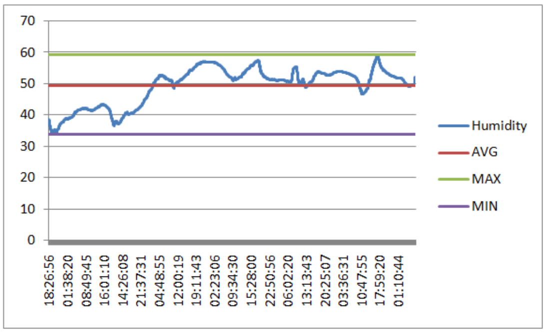

Figure 4. The Results for Humidity

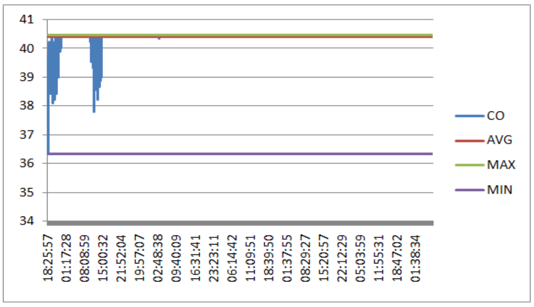

Figure 5. The Results for CO

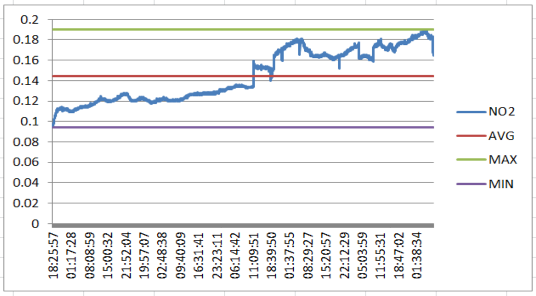

Figure 6. The Results for NO

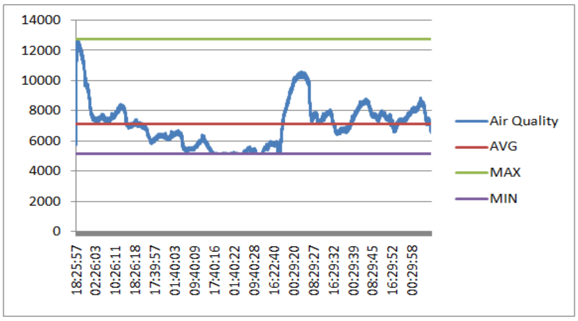

Figure 7. The Results for Air Quality

7. CONCLUSION

The work presented in the Raspberry Pi project is used mostly to discover harmful gases inside buildings (houses specifically) and monitoring air quality such as smoke and other gases. The system monitors the environment and logs the data, and informs people by email if any of the readings are over the specified limit.

The work required integrating different types of sensors with the Raspberry Pi, control and manage those using python. The entire code is open source, so it can be used with other sensors and can be used to expand into different measurements and experiments. Since it also uses Linux, it can easily be ported to other devices that can run an operating system and have input/output ports. The author hoped to add more sensors, but time and shipping problems did not allow, so they leave these plans for the future, where can add many more sensors and readings, as well as better analysis and information delivery.

REFERENCES

[1] Kulkarni, G. H., and P. G. Waingankar. "Fuzzy logic based traffic light controller." Industrial and Information Systems, 2007. ICIIS 2007. *International Conference on. IEEE*, 2007.

[2] Alkandari, A., Alnasheet, M., Alabduljader, Y., & Moein, S. (10-12 July 2012). "Water Monitoring System using Wireless Sensor Network (WSN): Case Study of Kuwait Beaches." Paper presented at Digital Information Processing and Communications (ICDIPC), *2012 Second International Conference on. Klaipeda City*. DOI 10.1109/ICDIPC.2012.6257270. (pp. 10 - 15).

[3] Hussein, M. (2012, 3-5 July 2012). "Design and implementation of a cost effective gas pollution detection system." Paper presented in Computer and Communication Engineering (ICCCE), *2012 International Conference on. Kuala Lumpur*. DOI 10.1109/ICCCE.2012.6271142. (pp. 7 - 12).

[4] Arizaga, J., Calleja, J., Hernandez, R., & Benitez, A. (2012, 27-29 Feb. 2012). "Automatic Control for Laboratory Sterilization Process based on Arduino Hardware". Paper presented in Electrical Communications and Computers (CONIELECOMP), *2012 22nd International Conference on. Cholula, Puebla*. DOI 10.1109/CONIELECOMP.2012.6189895. (pp. 130 - 133).

[5] Sayantani, B., Sridevi, S., &Pitchiah R. (2012, 18-21 Dec. 2012). "Indoor Air Quality Monitoring using Wireless Sensor Network." Paper presented at Sensing Technology (ICST), 2012 *Sixth International Conference on. Kolkata.* ISSN 2156-8065. DOI 10.1109/ICSensT.2012.6461713. (pp. 422-427).

[6] Preethichandra, D. (6-9 May 2013). "Design of a Smart Indoor Air Quality Monitoring Wireless Sensor Network for Assisted Living." Paper presented at Instrumentation and Measurement Technology Conference (I2MTC), *2013 IEEE International. Minneapolis*, MN. ISSN 1091- 5281. DOI 10.1109/I2MTC.2013.6555624. (pp. 1306 – 1310).

[7] Ferdoush, Sheikh, and Xinrong Li. "Wireless Sensor Network System Design Using Raspberry Pi and Arduino for Environmental Monitoring Applications." *Procedia Computer Science* 34 (2014), 103-110. doi:10.1016/j.procs.2014.07.059.

[8] Sonali, D., Venkatasubramanian, K. A (2016). "Raspberry-PI based IOT system for measuring the environmental parameters to monitor the pollution level using IBM Bluemix," *Pakistan Journal of Biotechnology*, 13, pp. 231-235.

[9] Balasubramaniyan, C., Manivannan, D. (2016). "IoT enabled Air Quality Monitoring System (AQMS) using Raspberry Pi." Indian Journal of Science and Technology, 9 (39), art. no. 90414.Jadhav, Gaurav, Kunal Jadhav, and Kavita Nadlamani. "Environment Monitoring System using Raspberry-Pi." *International Research Journal of Engineering and Technology (IRJET)*3, no. 4 (April 2016).

[10] Jadhav, Gaurav, Kunal Jadhav, and Kavita Nadlamani. "Environment Monitoring System using Raspberry-Pi." *International Research Journal of Engineering and Technology (IRJET)* 3, no. 4 (April 2016).

[11] Rohani, Mohd F., Noor A. Ahmad, Shamsul Sahibuddin, and Salwani M. Daud. "OPC Protocol Application for Real-Time Carbon Monitoring System for Industrial Environment." *International Journal of Electrical and Computer Engineering (IJECE)* 7, no. 2 (April 2017), 1051. doi:10.11591/ijece.v7i2.pp1051-1059.

[12] Susanto, Dwi, Kudang B. Seminar, Heru Sukoco, and Liyantono Liyantono. "Parallel Processing Implementation on Weather Monitoring System for Agriculture." *Indonesian Journal of Electrical Engineering and Computer Science* 6, no. 3 (June 2017), 682. doi:10.11591/ijeecs.v6.i3.pp682-687.

[13] Mekki, Marwa, and Osman Abdallah. "Development of a Wireless Sensors Network for Greenhouse Monitoring and Control." *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)* 5, no. 3 (September 2017), 270-274.

[14] Abd Allah, Osman, Mohammed Abdalla, Suliman Abdalla, Amin Babiker, and Alaa Awad Allah. "Universal Data Logger System for Environmental Monitoring Applications." *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)* 5, no. 2 (June 2017), 131-136.

[15] Baharun, A. N., N. A. Murad, and N. N. Malik. "Harmful Gases Profiling in Meru Menora Tunnel using SICK Sensor." *TELKOMNIKA (Telecommunication Computing Electronics and Control)* 15, no. 2 (June 2017), 888-894.

[16] Montanaro, Teodoro, Fulvio Corno, Carmelo Migliore, and Pino Castrogiovanni. "SmartBike: an IoT Crowd Sensing Platform for Monitoring City Air Pollution." *International Journal of Electrical and Computer Engineering (IJECE)* 7, no. 6 (December 2017), 3602. doi:10.11591/ijece.v7i6.pp3602-3612.

[17] Mitchell, G.. "The Raspberry Pi single-board computer will revolutionize computer science teaching." *Engineering & Technology*. ISSN 1750-9637. Vol. 7, Issue -3. (April 2012), p. 26.