e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:06/June-2023 Impact Factor- 7.868 www.irjmets.com

RASPBERRY PI PICO INDUSTRIAL SENSOR MONITORING SYSTEM WITH GSM & WEB APP/IOT TECHNOLOGY

Mr. Kondeti Chirunadh[*1], Mr. G. Sai Varaprasad Goud[*2], Miss. G. Tejaswini[*3], Mr. B. Srinivasa Rao[*4]

[*1,2,3]Student, Dept. Of Electronics And Communication Engineering, ACE Engineering College, Hyderabad, Telangana, India.

[*4]Associate Professor, Dept. Of Electronics And Communication Engineering, ACE Engineering College, Hyderabad, Telangana, India.

DOI : https://www.doi.org/10.56726/IRJMETS42308

ABSTRACT

Our objective is to design an Industrial monitoring system using IOT. Surveillance is most important security systems in home, industrial, office and public places. In this security system is based on the embedded system along with Microcontroller and sensor networks. The human movement is detected using the PIR sensors. In this time, the system triggers an alarm detecting the presence of person in a specific interval of time and simultaneously sends the how many persons are intruder. When the security system is activated, the PIR Sensor is activated. This highly reactive approach has low computational requirement. Therefore, it is well suited for Industrial surveillance system. This surveillance security system implemented using Microcontroller and sensors. Industrial security systems have grown in popularity in recent years, a Industrial owners look for ways to protect their personal space and enhance their Industrial values. It is necessary for every Industrial owner to considering adding a industrial security and monitoring system, as burglaries, thefts and murders have become routine in big cities. This paper demonstrates a Industrial machine control system that allows the user to monitor the data through mobile phone. A desktop PC is used to run the server software. It can automatically turn off the main motors and turn on a motors at a specified time.

I. INTRODUCTION

Modern industries have been introduced to a broad range of manufacturing processes to ease reconfigurability and enhance flexibility while retaining the high throughput of the quality of products. Such systems provide real time data acquisition, enabling the monitoring of the actual condition of the manufacturing process. IOT facilitates real time monitoring and optimization of the fabricating systems, reduces the time for maintenance by instantaneously taking necessary corrective measures. IOT technology can generate an added value to logistics. The embedded software architecture offers a reliable solution to eliminate communication latency and provides real-time response to acquired information. Real-time (data) monitoring is the delivery of continuously updated information streaming at zero or low latency. Monitoring involves collecting data periodically throughout an organization's environment from on-premises hardware and virtualized environments to networking and security levels, into the application stack including those in the cloud and out to software UIs. From this data, we can analyze system performance, flaganomalies and resolve issues. Real-time monitoring ups the ante by providing a continuous low- latency stream of relevant and current data from which administrators can immediately identify serious problems. Alerts can be more quickly routed to appropriate staff or even to automated systems. By tracking real- time monitoring data over time, organizations can reveal and predict trends and performance.

II. METHODOLOGY

1. Requirement Analysis:

Define the specific features and functionalities you want to include in the system, such as Sensors Operation and functionality , GSM SIM 800A , RASPBEERY PI PICO,NODE MCU and notifications, analysis of IOT Platform.

Determine the communication range and reliability required for the IoT module based on the distance between the NODE MCU and the monitoring device (mobile phone or PC). Consider any specific constraints or limitations, such as budget, power supply availability, or space requirements.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:06/June-2023 Impact Factor- 7.868 www.irjmets.com

2. Hardware Setup:

Connect the Fire sensor, LM35 Sensor, PIR Senor and Gas Sensor to the Node MCU. Connect an LED to the Raspberry Pi Pico. Connect the GSM module to the Node MCU for SMS functionality. Now interface the Node MCU

2. Software Setup:

Set up the development environment for the Raspberry Pi Pico and Node MCU .Install the necessary libraries for the sensors, LED, and GSM module.

3. Sensor Integration:

Write code to read data from the Fire sensor, LM35 sensor, Gas sensor and PIR Sensor. Define appropriate thresholds to detect abnormal readings indicating a potential hazard.

4. GSM Integration:

Implement code to send SMS messages using the GSM module. Define the mobile number(s) to which the SMS will be sent. Compose the SMS message to include relevant information about the detected hazard.

5. Integration with Thing Speak IoT Platform:

Create an account on Thing Speak (https://thingspeak.com/) and set up a new channel. Configure the Raspberry Pi Pico or Node MCU to send sensor data to the Thing Speak channel . Send the sensor data to Thing Speak using the appropriate API or library provided by Thing Speak.

6. Analysis using Thing Speak:

Use the Thing Speak platform to analyze and visualize the sensor data. Set up appropriate MATLAB code on Thing Speak to generate alerts or trigger actions based on predefined rules. Configure Thing Speak to send notifications or trigger events when abnormal sensor readings are detected.

7. Testing and Deployment:

Test the entire system by simulating sensor outputs and verifying if the SMS is sent and LED is functioning correctly.

Deploy the system in the desired location and ensure proper power supply, connectivity, and security measures.

8. Monitoring and Maintenance:

Regularly monitor the system for any issues or false alarms. Perform routine maintenance, such as checking sensor functionality, replacing batteries, and updating software if necessary.

III. MODELING AND ANALYSIS

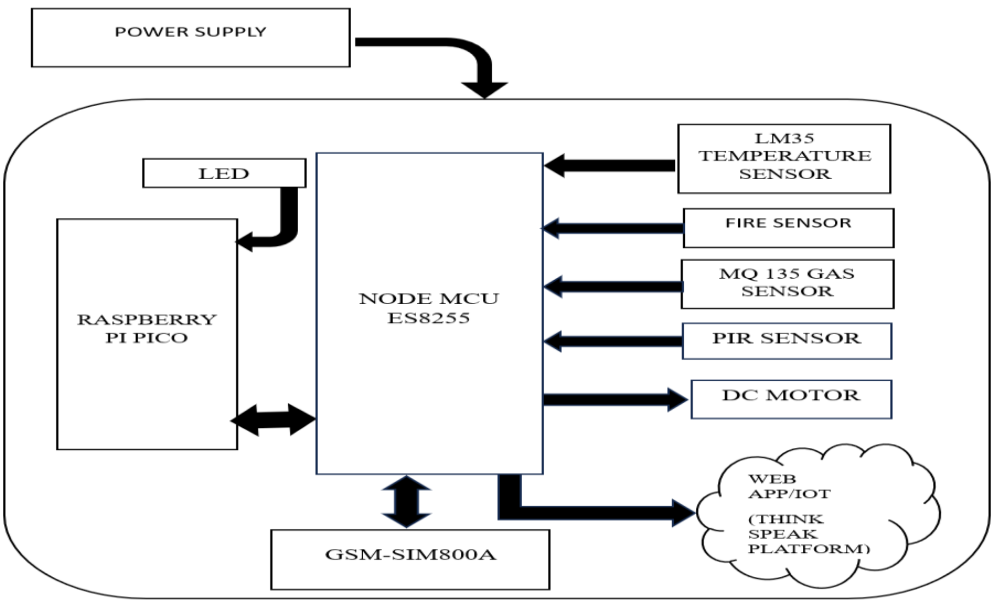


Figure 1: Block Diagram

IRJMETS
e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:06/June-2023 Impact Factor- 7.868 www.irjmets.com

In this section, we present the models and materials used in our research work. The following table provides an overview of the key components and specifications of the hardware and software utilized in our study.

System Architecture:

The architecture of the fire detection system consists of the following components:

1. Fire Sensor: Detects the presence of fire or significant rise in temperature.
2. Flame Sensor: Detects the presence of flames.
3. Gas Sensor: Detects the presence of harmful gases.
4. Pir Sensor: Detects the infrared radition emitted by objects in its field of view.
5. Raspberry Pi Pico: Microcontroller board used to interface with the Node MCU and control the LED.
6. Node MCU: Wi Fi-enabled microcontroller used for Internet connectivity and communication with the Thing Speak platform and transfers the data from sensors to the Raspberry Pi pico.
7. GSM Module: Enables sending SMS alerts to a designated mobile number.
8. LED: Visual indicator controlled by the Raspberry Pi Pico.
9. Thing Speak IoT Platform: Cloud-based platform for data logging and analysis.

System Workflow:

1. Sensor Detection: The fire, flame, and gas sensors continuously monitor their respective environments.
2. Sensor Output: When any of the sensors detect a fire hazard, they send a signal to the Node MCU and Node MCU transfers and receives the data from Raspberry Pi PICO
3. SMS Alert: The Raspberry Pi Pico triggers the GSM module to send an SMS alert containing relevant information (e.g., sensor type, timestamp, location) to the designated mobile number by tasmitting signals to GSM through Node MCU.
4. Thing Speak Integration: The Node MCU communicates with the Thing Speak platform, sending sensor data for logging and analysis purposes.

Think Speak IoT Platform Connectivity:

1. Thing Speak Setup: Create a Thing Speak account and set up necessary channels for data logging.
2. Node MCU Integration: Configure the Node MCU with the appropriate W iFi credentials and Thing Speak API keys.
3. Data Logging: The Node MCU periodically sends sensor data to the Thing Speak platform, which records and stores the data in the designated channels.
4. Data Analysis: Utilize Thing Speak's built-in visualization and analysis tools to monitor sensor data trends, generate alerts based on predefined thresholds, and gain insights into fire incidents.

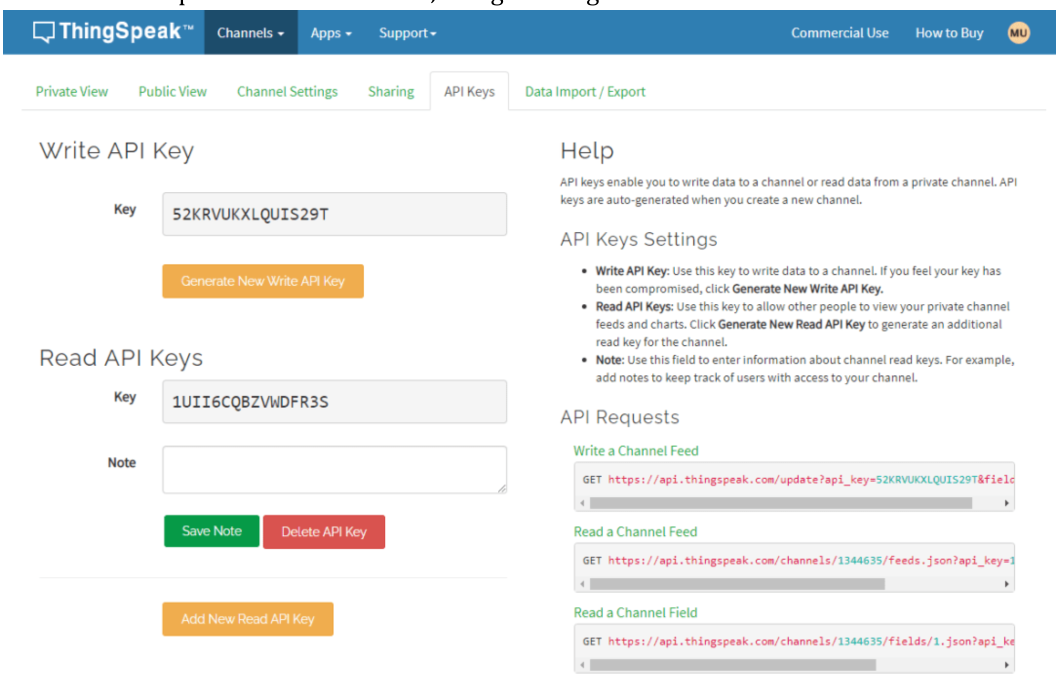


Figure 2: Think Speak channel Setup/Connectivity

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:06/June-2023 Impact Factor- 7.868 www.irjmets.com

IV. RESULTS AND DISCUSSION

The system will send an SMS notification to a designated mobile number as soon as any of the sensors detect an abnormality. This allows the recipient to be promptly informed about the potential hazard, enabling them to take appropriate action. The SMS notification will indicate the type of alarm triggered, whether it is related to fire, flame, or gas. This information helps the recipient to identify the nature of the potential threat accurately. The system continuously monitors the sensor outputs, ensuring constant surveillance for potential hazards.

It offers a proactive approach to detecting and responding to fire, flame, or gas incidents, enhancing safety and security. Since the system utilizes GSM technology, it can operate remotely and send SMS alerts to any mobile number, providing flexibility and convenience for monitoring hazardous situations from a distance.

The project allows for customization of sensor thresholds and SMS notification messages. This feature enables users to tailor the system to their specific requirements, adapting it to different environmental conditions and sensitivities. The project can be expanded to incorporate additional sensors or integrate with other security and automation systems. This scalability enables the system to be extended to cover broader safety concerns or integrated into larger-scale projects. By providing real-time alerts, the project contributes to improved safety measures and reduces response time in critical situations. This can be crucial in preventing accidents, minimizing damage, and potentially saving lives.

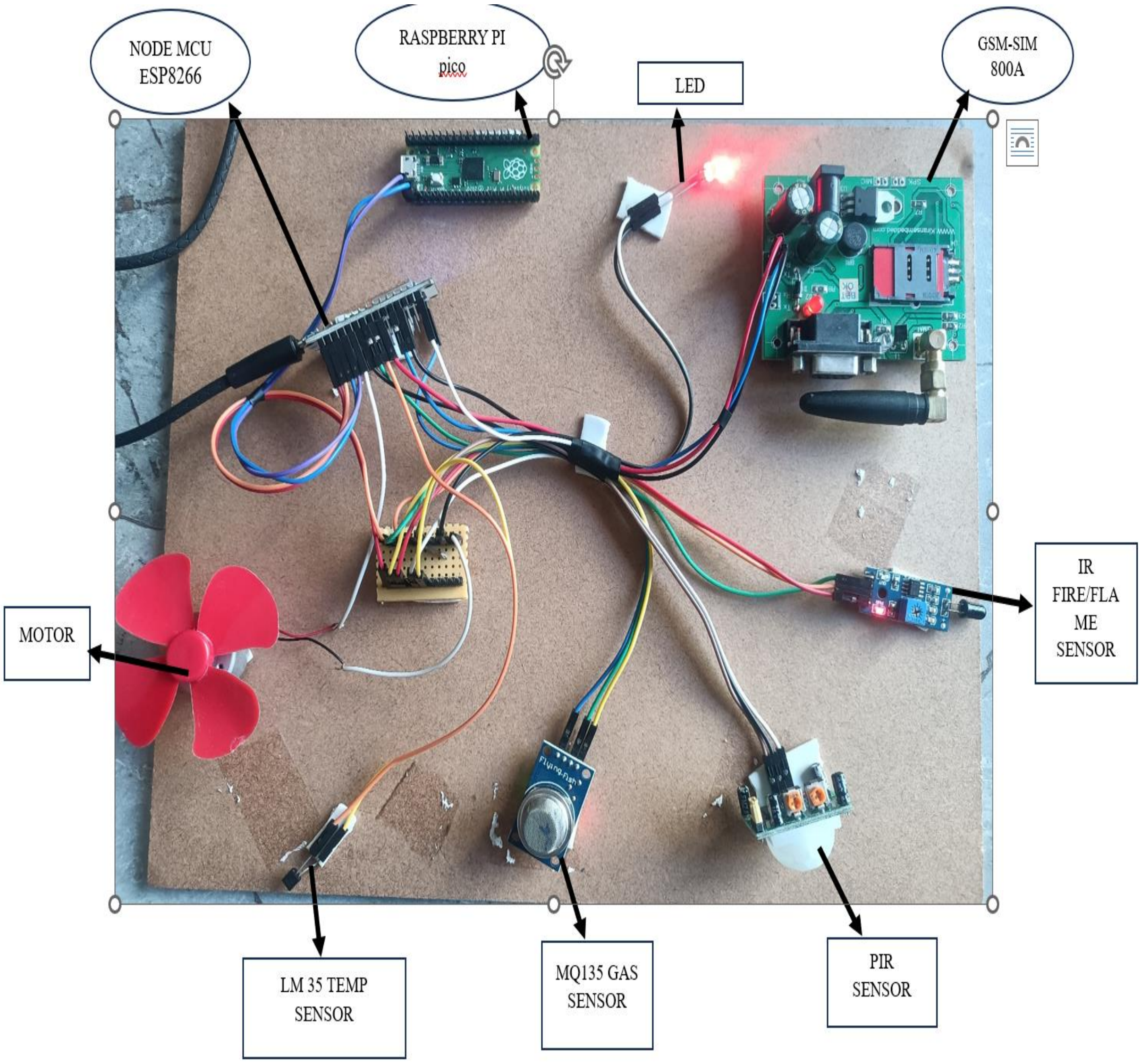
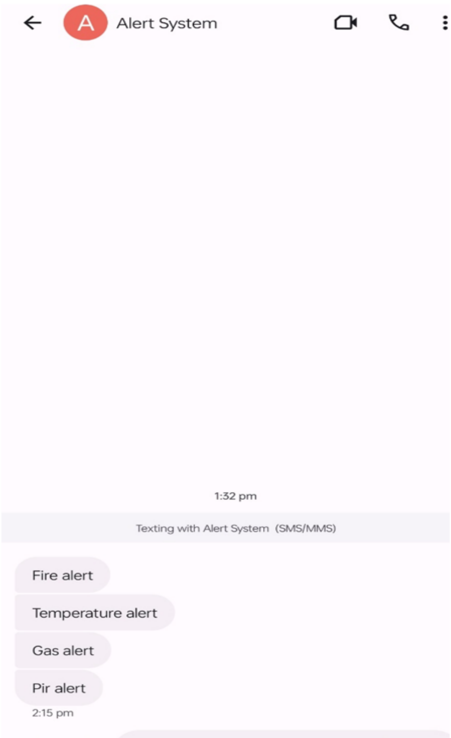


Figure 1: Working Model

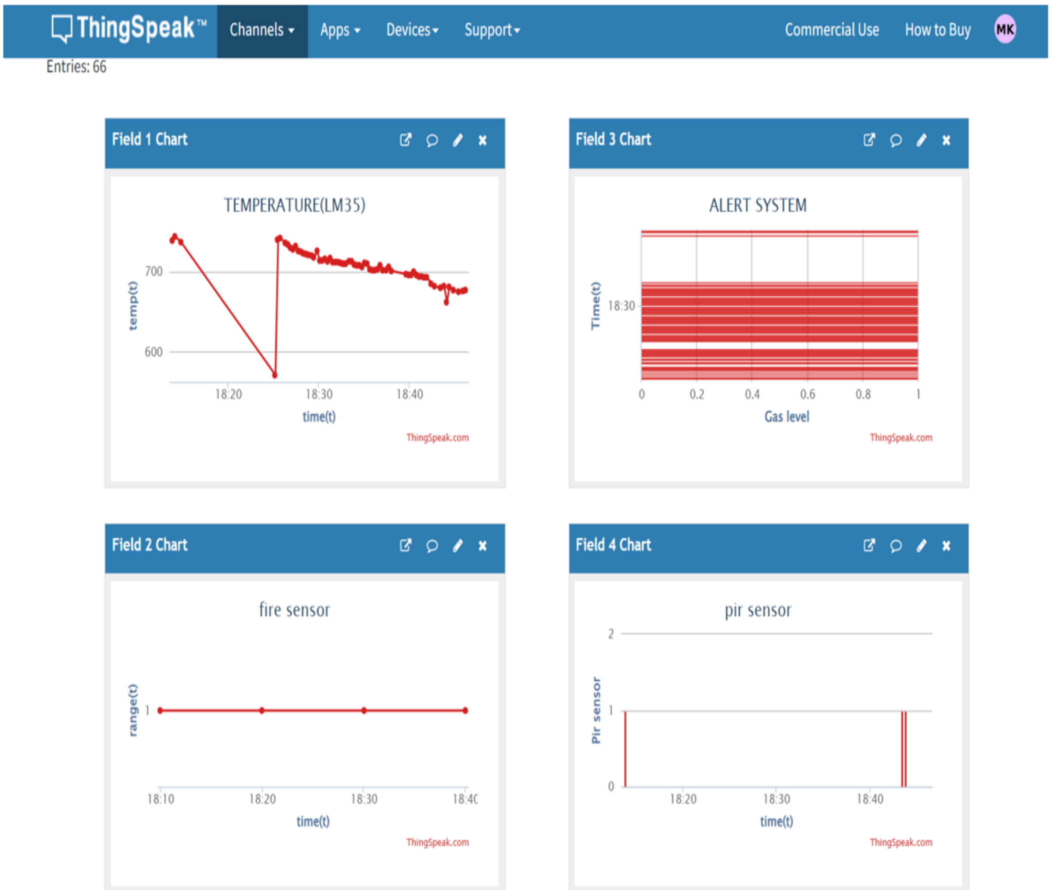
e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:06/June-2023 Impact Factor- 7.868 www.irjmets.com



Input: When the sensors are triggered ,the status of sensors wil be sent to a Node MCU and Raspberry Pi Pico.

Output: The Raspberry Pi Pico sends the Alert message to the designated Mobile Number ,through GSM -Sim 800A.

Figure 2: Result.1 -MESSAGE DELIVERED TO A DESIGNATED NUMBER



Fields:

x-axis:-time(t)

y-axis:-range of sensing data

The graph sheets indicates the data from the sensors output.

The x-axis indicates the time and date.

The y-axis indicates the data in range of numbers accordingly as output of the sensors.

Figure 3: Result on WEB APP/ IOT platform.

V. CONCLUSION

In conclusion, this project successfully integrates multiple sensors, including a Fire sensor, LM 35 Sensor, PIR Senosr and Gas sensor, with a Raspberry Pi Pico, Node MCU, and GSM technology to detect potential fire or gas-related incidents. When any of these sensors detect an output, the system triggers an LED and sends an SMS to a designated mobile number, ensuring immediate notification and response to potential hazards.

Furthermore, the project utilizes the Think Speak IoT platform for connectivity and data analysis. By leveraging the platform's capabilities, the system can securely transmit sensor data to the cloud, enabling real-time monitoring and analysis. This integration allows for advanced data visualization, trend analysis, and the ability to set up alerts or notifications based on predefined thresholds.

The combination of sensor integration, Raspberry Pi Pico and Node MCU controllers, GSM technology for SMS alerts, and the Think Speak IoT platform for data connectivity and analysis makes this project a comprehensive solution for fire and gas detection. It enhances safety measures by promptly alerting the designated individuals and facilitating data-driven decision-making for preventive measures and system improvements

VI. FUTURE SCOPE

Expanding the system's capability by integrating various sensors. This will provide comprehensive coverage. By analyzing data from multiple sensors simultaneously, the system can accurately identify and differentiate between various types of emergencies, such as a fire outbreak or a gas leak. Introduce customizable alert levels to prioritize different situations. Users should be able to set specific thresholds for each sensor, defining the severity of the detected hazard. Develop a centralized monitoring system that provides real-time updates on the sensor readings, alarm status, and SMS notifications. Create dedicated mobile applications for iOS and Android platforms to enhance user accessibility and control. Integrate geolocation services and mapping APIs to enhance the system's functionality.

VII. REFERENCES

[1] Internet of Things-based Hydrocarbon Sensing for Real- time Environmental Monitoring, Ali Yavari, Dimitrios.

[2] Georgakopoulos, Paul R. Stoddart, Mahnaz Shafiei, Faculty of Science, Engineering and Technology, Swinburne University of Technology, Melbourne, Australia – 2019 IEEE 5th World Forum on Internet of Things (WF- IoT)

[3] Low Cost IoT Sensor System for Real- time Remote Monitoring, Matteo DAloia, Annalisa Longo, Gianluca Guadagno, Mariano Pulpito, Paolo Fornarelli, Pietro Nicola Laera, Dario Manni.

[4] Design of an Industrial IoT-Based Monitoring System for Power Substations, Long Zhao Igor Matsuo Yuhao Zhou* Wei-JenLee, University of Texas at Arlington.

[5] Industry 4.0 LabVIEW Based Industrial Condition Monitoring System for Industrial IoT system, 2019 Developments in eSystems Engineering (DeSE).

[6] https://circuitdigest.com

[7] https://raspberrypi.cc