2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)

Development of an IoT Based Open Source SCADA System for PV System Monitoring

Lawrence O. Aghenta
ECE, Faculty of Engineering, Memorial University
St. John's, NL, Canada
Email: loaghenta@mun.ca

M. Tariq Iqbal
ECE, Faculty of Engineering, Memorial University
St. John's, NL, Canada
Email: tariq@mun.ca

***Abstract*—This paper presents the development of a low cost, open source Supervisory Control and Data Acquisition (SCADA) system for solar photovoltaic (PV) system monitoring and remote control. The proposed SCADA system is based on the Internet of Things (IoT) SCADA architecture which incorporates web services with the conventional SCADA for a robust supervisory control and monitoring. It comprises of analog Current and Voltage sensors for acquiring the desired data from the solar PV system, Arduino Uno micro-controller which serves as a Remote Terminal Unit to receive the acquired sensor data, Raspberry Pi with a Node-RED programming tool for parsing the acquired data (Communication Channel), and Emoncms Local Server IoT Platform for data storage, monitoring and remote control (Master Terminal Unit). The developed SCADA system was set up to monitor and control a 260W, 12V solar PV panel in the Electrical and Computer Engineering Laboratory at Memorial University, and some of the created Dashboards and Charts showing the acquired data on Emoncms server where an operator can monitor the data in the cloud using both a computer with internet access, and Emoncms mobile app are presented in the paper.**

***Index Terms*—Low Cost, Open Source SCADA, EMONCMS, Arduino Uno, Raspberry Pi, Node-RED, Instrumentation and Control.**

I. INTRODUCTION

SCADA is an acronym formed from the first letters of the term "Supervisory Control and Data Acquisition". It is a technology that enables a user to collect data from one or more distant facilities and/or send limited control instructions to those facilities. The major function of SCADA is for acquiring data from remote devices such as batteries, valves, pumps, transmitters, etc. and providing overall monitoring and control remotely from a SCADA host software platform [2] [12]. As in the energy industry across the globe, power system assets, such as inverters are usually distributed over large geographical areas, sometimes in harsh environments. While it may be necessary to have local means of managing the operations of these assets, it is equally important to have a reliable, flexible, cost effective and sophisticated coordinated control. SCADA system is the perfect solution for this task. SCADA makes it unnecessary for an operator to be assigned to stay at or frequently visit these remote locations when the facilities are operating normally. Essentially, a SCADA system performs four basic functions; Data Acquisition, Networked Data Communication, Data Presentation, and Remote Monitoring and Supervisory Control.

SCADA performs these essential functions using its four basic elements; Field Instrumentation devices such as sensors and actuators connected to the systems being managed, Remote Terminal Units (RTUs) such as single board computers for acquiring remote data from field instrumentation devices, Master Terminal Units (MTUs) for handling data processing and human machine interactions, and lastly SCADA Communication channels for connecting the RTUs to the MTUs [2].

Generally, there are two classes of SCADA hardware and software; Proprietary and Open Source. In a proprietary system, all major components are from a single manufacturer and the standards are often specific to that system, and developed by the manufacturer. With a proprietary SCADA system, the responsibility for system reliability and security rests solely with the manufacturer, which leaves the user vulnerable to a single manufacturer/supplier as the manufacturer/supplier could be slow to respond to technological changes in a subsystem of the SCADA system. The customer is also at risk if the manufacturer/supplier goes out of business, and the solution is largely expensive. There is also the problem of flexibility with the already existing devices and network. An Open source system allows a user to "mix and match" components and choose the most appropriate from several suppliers. This means that with an open source system, no single supplier is responsible for overall system performance. An open source solution also represents the most cost effective solution as the user is not beholden to a single supplier. In an open source SCADA system, the major components adhere to certain standards which allow them to be interchanged with similar components manufactured by others to the same standards [12]. Therefore, in this paper, an open source SCADA system solution is proposed.

II. LITERATURE REVIEW

There are four generations of SCADA system architecture and they include; the first generation (Monolithic), the second generation (Distributed), the third generation (Networked), and the fourth generation (Internet of Things based SCADA architecture) [12]. The Internet of Things concept has to do with connecting physical objects with embedded electronics, software, sensors and connectivity to enable data interchange between these devices and an operator over a common network or the web [1] [3] [4] [5].

Various researchers across the globe have in the past designed SCADA systems based on the IoT architecture. In [4], a form of IoT based SCADA system is implemented using Raspberry Pi3 as the sensor gateway, DHTII temperature and humidity sensors to acquire the desired data, and IBM Bluemix cloud platform to receive, visualize and manage the acquired sensor data over the web while using Node-Red and Web Socket Protocol for data exchange and communication between the cloud platform and the Raspberry Pi connected sensors. [3] presented an IoT based urban climate monitoring system using Arduino Nano, Raspberry Pi2 and Adafruit IO IoT web server. Elsewhere, [6] presented the implementation of a web-based monitoring and control system for real-time electrical data measurement in a hybrid wind/PV/battery system using web-based InTouch for graphical user interface. In [5], the proposed IoT based SCADA system uses Raspberry Pi3 along with Intel Edison board for sensor inputs and the acquired sensor data is sent to Amazon Web Services (AWS) IoT platform using MQTT brokers in Node-Red. At the AWS IoT platform, various monitoring and control schemes are initiated using Amazon's Voice Service called Alexa. In another development, [2] developed a low cost SCADA system for CO2 Enhanced Oil Recovery based on the IoT SCADA architecture using sensors and actuators connected to wellheads, Arduino Yun, Ubiquiti NanoStation wireless Ethernet/IP addressable radios physically connected to the Arduino Yun, and a web enabled central server running MySQL

978-1-7281-0319-8/19/$31.00 ©2019 IEEE

DBMS database for Graphical User Interface. In a recent development, [1], presented the design of an IoT based sensing and solar house monitoring and automation system using NodeMCU combined with ESP8266 micro-controller as the sensor gateway for communication and data acquisition, and a combination of Blynk and EmonCMS web server for collecting and visualizing the acquired data, and for remote control of home appliances and devices. Also, authors in [7], [8], [9], and [10] have implemented various forms of IoT based SCADA systems.

In most of the above mentioned papers, the IoT platform, like the Amazon Web Services presented in [5], the IBM Bluemix in [4], Adafruit IO in [2], and EmonCMS web server in [1] for data storage, monitoring and control is hosted in the cloud, which leaves the stored data vulnerable to web attacks. However, security in a SCADA system is a serious issue both from the operational and economic points of view as the resultant unavailability of the critical infrastructure being monitored in the events of attacks can disrupt the related operations which could cause a huge loss. Therefore, in this paper, local EmonCMS server is used as the IoT platform for data acquisition, storage, monitoring and control. The hardware is installed in a private Linux based machine and is managed by the user. The local EmonCMS server has all the functionalities of the web-based EmonCMS server, with additional data security as the user is able to manage the server locally [11]. The local EmonCMS server solution is also less expensive as the user only buys the hardware on a one-off basis while the user on the web-based EmonCMS server continuously pays for data storage [11]. Also, the power consumption of the local EmonCMS server is minimal [11]. Furthermore, the data communication modes between the sensor gateways and the IoT platforms presented in some of the literatures are complex and require advanced programming skills like in the case of Python Script in [2]. To address this shortcoming, the simple secured Node-Red sensor wiring is proposed for data transfer from the gateway to EmonCMS server.

The remaining part of this paper is dedicated to the design , experimental setup and testing of the proposed IoT based SCADA system.

III. THE PROPOSED SCADA SYSTEM DESIGN

The proposed low cost, open source SCADA system is based on the Internet of Things (IoT) SCADA architecture. The schematics of the system design configuration is shown in Fig. 1. The system is made up of analog Voltage and Current sensors, Arduino Uno micro-controller, Raspberry Pi2 single-board computer with Node-Red programming tool for data transmission, and EmonCMS local server IoT Platform for sensor data monitoring and remote control.

A. Sensors

Sensors are the Field Instrumentation devices in the proposed SCADA system as they are connected directly to the PV system being managed to acquire the desired data [2]. Three analog sensors are used in our setup; one ACS 712 Hall Effect Current Sensor, and two MH Electronic Voltage Sensor modules. The properties of these sensors and their usage in this project are described below:

1) ACS 712 Hall Effect Current Sensor: The ACS 712 Hall Effect Current Sensor is manufactured and supplied by Allegro MicroSystems, LLC. It is a low cost, fully integrated, Hall Effect-based linear current sensor IC with a low-resistance current conductor. In this project, the 30A model is used to measure the DC current from the solar PV system. To achieve this, its VCC pin is powered with the 5V on the Arduino Uno board, the OUT pin is connected to Analog pin A0 on the Arduino, and its GND pin is connected to the GND pin on the Arduino while the two Input pins are connected in series to the PV system to measure the DC current flowing through the system.

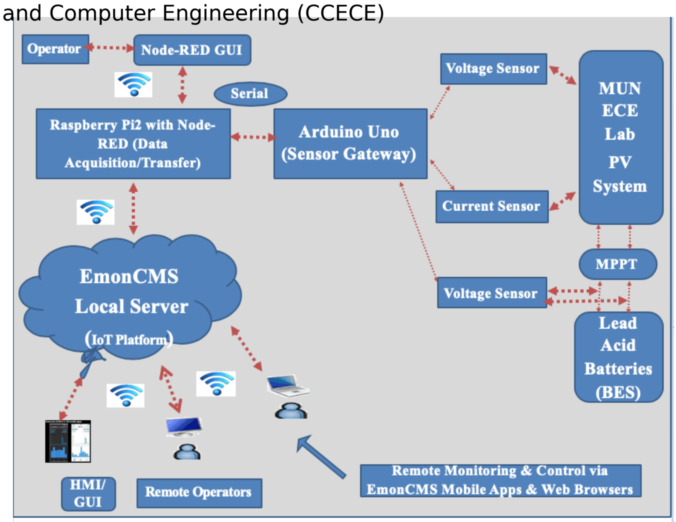

Fig. 1. Block Diagram of the Proposed IoT-based SCADA System.

2) MH Electronic Voltage Sensor module: This low cost voltage sensor uses the concept of voltage divider to measure the supply voltage across which it is connected. In this project, two voltage sensors are used. One voltage sensor is connected in parallel to the PV system to measure the voltage across it while a second voltage sensor is connected in parallel across the lead acid battery system to measure the stored battery voltage. For the first voltage sensor, PIN S is connected to Analog PIN A1 on the Arduino Uno board, PIN – is connected to a GND pin on the Arduino while its GND and VCC pins are connected in parallel across the PV panel output to measure the voltage across the PV system (PIN + on the sensor is not used). The second sensor is connected to the battery in a similar fashion using Analog PIN A2 on the Arduino Uno board.

B. Arduino Uno Board

Arduino Uno is a basic, low cost micro-controller board based on the ATmega328P. It has 14 digital input/output pins, 6 analog inputs (A0 to A5), a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button [10]. In this project, the current and voltage sensors are connected to the Arduino board as described in the sensor section above. First, an Arduino sketch to measure the voltage and DC current from the PV system, and calculate the PV power output from the voltage and current values, as well as to separately measure the voltage stored in the battery via the sensors is written in the Arduino IDE and uploaded to the board. Secondly, having uploaded the program to the Arduino board, the Arduino board is connected through its USB cable to a USB port on the Raspberry Pi2 to power the Arduino, and to parse the measured and calculated sensor data to the Raspberry Pi using the specified Baud Rate.

C. Raspberry Pi Board

The Raspberry Pi2 model B used in this project is a 85*56mm single board computer device with BCM2836 quad core ARMv7 processor [3]. In this application, the Node-RED programming tool needed to send the acquired data to EmonCMS IoT platform is installed in the Raspberry Pi. The Raspberry Pi is connected to MUN network using an Ethernet cable which means that any other machine on the network with the right authorization, the EmonCMS server in this case, can communicate with the Node-RED installed on the Raspberry Pi.

1) Node-RED: Developed by IBM, Node-RED is an open source programming tool for wiring together hardware devices, APIs and online services in a smart way [4]. It can be installed in a Linux based Platform and it provides a browser-based editor that makes it easy to wire together flows using various nodes in the palette that can be deployed to its runtime in a single-click [4], [5]. In this application, the Node-RED is installed on the Raspberry Pi. A Node-RED flow is written to acquire the sensor data from the Arduino Uno Serial port connected to the Raspberry Pi, and post the acquired data to EmonCMS local server using its IP address, input API Key and Node ID. As shown in Fig. 2 below, the Node-RED flow used consists of an Arduino Uno Serial block which receives sensor data from the Arduino board, a Function block with "functions" for parsing the acquired sensor data, an Emoncms Push block containing Emoncms server parameters for posting the sensor data to the platform, and msg.payload block for debugging. The acquired sensor data being posted to Emoncms IoT platform is displayed on the Node-RED display window as shown in the figure (right).

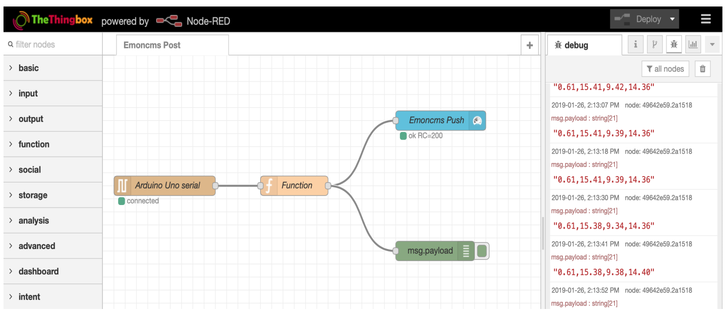

Fig. 2. Node-RED Flow for EmonCMS Data Logging.

D. EMONCMS Local Server IoT Platform

EMONCMS is a powerful open-source web-app for processing, logging and visualizing energy, temperature and other physical/environmental data [11]. It is a low cost IoT platform (under CAD 500), highly flexible, and it is a part of the OpenEnergyMonitor.org project. It has a Local Server option where a user purchases (one-off) the hardware and installs it on a standalone or networked Windows or Linux based machine for proper management. It also has a web-based server option which can be accessed using its URL just like every other web application. Both the web-based and local server options have IoT capability as the data stored in them can be accessed remotely with an internet enabled computer or with an internet enabled phone via EMonCMS mobile app or phone browser [11]. However, the local server EmonCMS option is more secured as the user has a better control of the server and stored data. A free open source version is available on Github.com.

EmonCMS IoT platform allows the operator to create all kinds of visualization dashboards and events for remote monitoring and supervisory control. It also allows an operator to create customized reports in the form of charts, data logs and alarms which can be viewed either locally, via email notifications, via web browsers, and EmonCMS mobile apps [11].

In this project, the EmonCMS local server receives the PV voltage, PV current, measured PV power, and stored battery voltage from the Node-RED program installed on the Raspberry Pi. At EmonCMS Inputs, the Node-RED program identifies the specified Node ID and displays the data. This Input data is logged automatically, as set up by the operator, to EmonCMS Feeds in order to maintain a history of the received data. The data at both the Inputs and Feeds is stored, while remote monitoring, and control access/commands are initiated via Dashboards and Events created by the operator.

E. MUN ECE Laboratory PV System Overview

The Memorial University Electrical and Computer Engineering Laboratory PV system is a 260W, 12V PV panel with Maximum Power Point Tracking (MPPT) system. Electrical batteries are connected to the MPPT to store the energy from the solar panels for use during prolonged extreme weather conditions. In this project, the SCADA system is set up to acquire the PV voltage, current and power, and the stored battery voltage for remote monitoring and supervisory control.

IV. EXPERIMENTAL SETUP OF THE PROPOSED SCADA SYSTEM

The proposed SCADA system is set up in MUN ECE Laboratory as shown in Fig. 3. On the wall is the PV System being monitored, and for testing purposes, one module is used. The inputs of the current and voltage sensors (sensors shown in the figure) are connected to the PV module and the battery system (batteries are below the table in the figure) using electrical cables, and the sensor outputs are connected via cables to an Arduino Uno board (left). The Arduino is connected to a Raspberry Pi (right) via a USB cable, and the Node-RED program installed on the Raspberry Pi, connected to MUN network using an RJ45 Ethernet cable (the blue cable in the figure) sends the acquired data to EmonCMS server.

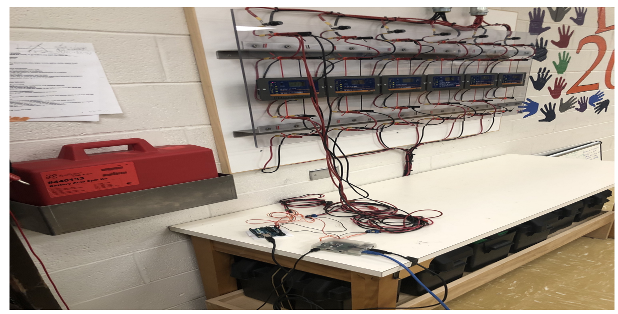

Fig. 3. Experimental Setup of The Proposed SCADA System.

V. TESTING, RESULTS AND DISCUSSIONS

The proposed SCADA system was set up in MUN ECE Laboratory to acquire data from the PV system, and post it, using Node-RED to EmonCMS local server for remote monitoring and supervisory control. Fig. 4 shows the data flow from the PV system to Emoncms IoT Platform.

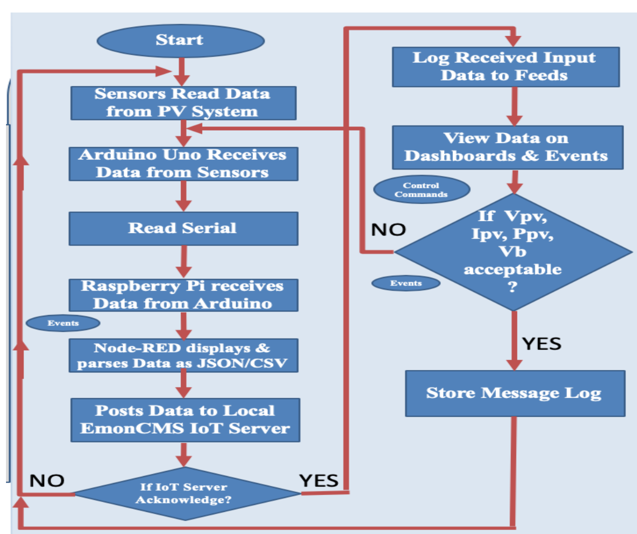

Fig. 4. Flow Chat of the Proposed SCADA System Data Acquisition.

A. Results

The system was tested for two weeks. The data received at EmonCMS Input window was automatically logged to the Feeds window. Having received the PV system data, customized dashboards and reports in the form of charts, data logs and alarms were created on EmonCMS IoT Platform. These dashboards, events and reports which could be viewed either locally, through email notifications, web browsers, and EmonCMS mobile apps help an operator to monitor the stored data, view trends in the stored data, and initiate supervisory controls remotely. Figures 5 and 6 show two of the created dashboards for Realtime data visualization and Raw data visualization respectively. In figure 5, real time values of PV Voltage, Current, and Power, and Battery Voltage are being visualized respectively, and the time intervals and time stamps for each data received are displayed. In figure 6, raw values for PV Voltage, Current, and Power, and Battery Voltage are being visualized respectively at various time intervals shown in the figure. The vibrations of the raw data show the changes in the values of the received data due to the fluctuating environmental conditions affecting the PV System. Although the recorded voltage and current values are low at this time due to the snow conditions in St. John's at this time of the year, the voltage and current sensors are capable of measuring up to 25V and 30A respectively. However, the acquired sensor data match the values measured locally using conventional current and voltage digital multimeters. Aside figures 5 and 6 shown, EmonCMS Mobile App was also connected to EmonCMS local server IoT Platform by scanning the barcode of the platform using the Mobile App for a more flexible remote monitoring. Other Scheduling and Visualization features such as bargraphs, histgraphs, and timecompare are available on EmonCMS IoT Platform [11].

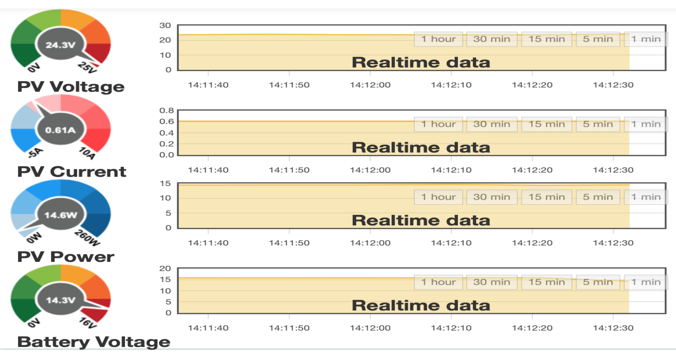

Fig. 5. Created EmonCMS Dashboard showing Realtime Data Visualization.

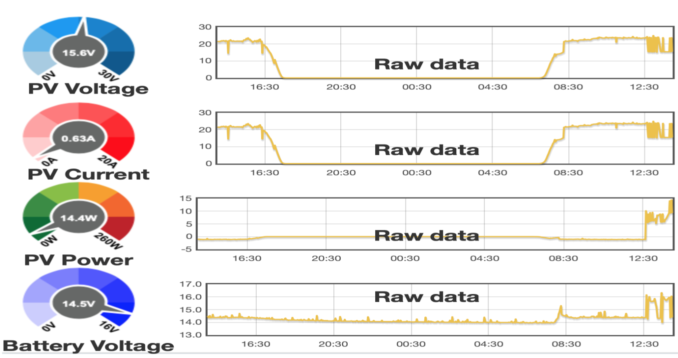

Fig. 6. Created EmonCMS Dashboard showing Raw Data Visualization.

B. Discussions

Some of the key features of the developed IoT based SCADA system are enumerated below:

- Internet of Things based SCADA System: It is based on the Internet of Things SCADA architecture and it has the four basic elements of a SCADA system listed earlier in this paper.
- Low Cost, Open Source: All the components of the proposed system are manufactured and supplied by different manufacturers (mix and match) which is one of the key characteristics of an open source system.
- Data Acquisition and Historic Storage: The SCADA system stores the received data and maintains data history. It also supports adjustable data log and storage rate.
- Remote Monitoring and Supervisory Control: It enables an operator to create Events and Dashboards for remote monitoring and supervisory control via web browsers and EmonCMS Mobile App.
- Reporting: It presents reports to the operator and key decision makers in the form of charts, data logs, and alarms (local, email, web, and mobile app).

VI. Conclusions

In this paper, the development of a low cost, open source Internet of Things based SCADA system has been presented. The developed SCADA system has all the four basic elements needed in a SCADA system including Field Instrumentation Devices, Remote Terminal Units, Master Terminal Units, and SCADA Communication Channel. The experimental setup of the developed SCADA system was carried out in MUN ECE Laboratory and it was used to acquire, and remotely monitor and control a 260W, 12V solar PV panel system. The system has been tested and upon testing, the system was able to carry out the desired functions of a SCADA including Data Acquisition, Networked Data Communication, Data Presentation, Remote Monitoring and Supervisory Control. The developed SCADA system can also be applied in other industries to remotely monitor and control critical infrastructures such as Electric power generation, transmission and distribution systems, Buildings, facilities and environments, Oil and Gas production facilities, Mass transit systems, Water and sewage systems, and Traffic signal systems.

Acknowledgements

The authors would like to thank the Natural Sciences and Engineering Research Council (NSERC) of Canada Energy Storage Technology Network (NESTNet) for funding this research.

References

[1] M. Al-Kuwari, A. Ramadan, Y. Ismael, L. Al-Sughair, A. Gastli and M. Benammar, "Smart-home automation using IoT-based sensing and monitoring platform," 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018), Doha, 2018, pp. 1-6.

[2] Xie Lu. "Supervisory Control and Data Acquisition System Design for CO2 Enhanced Oil Recovery", Master of Engineering Thesis, Technical Report No. UCB/EECS-2014-123. EECS Department, University of California at Berkeley, May 21, 2014.

[3] R. Shete and S. Agrawal, "IoT based urban climate monitoring using Raspberry Pi," 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, 2016, pp. 2008-2012.

[4] M. Lekić and G. Gardašević, "IoT sensor integration to Node-RED platform," 2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH), East Sarajevo, 2018, pp. 1-5.

[5] A. Rajalakshmi and H. Shahnasser, "Internet of Things using Node-Red and alexa," 2017 17th International Symposium on Communications and Information Technologies (ISCIT), Cairns, QLD, 2017, pp. 1-4.

[6] Li Wang and K. Liu, "Implementation of a Web-Based Real-Time Monitoring and Control System for a Hybrid Wind-PV-Battery Renewable Energy System," 2007 International Conference on Intelligent Systems Applications to Power Systems, Toki Messe, Niigata, 2007, pp. 1-6.

[7] N. P. Kumar and R. K. Jatoth, "Development of cloud based light intensity monitoring system using raspberry Pi," 2015 International Conference on Industrial Instrumentation and Control (ICIC), Pune, 2015, pp. 1356-1361.

[8] V. Sandeep, K. L. Gopal, S. Naveen, A. Amudhan and L. S. Kumar, "Globally accessible machine automation using Raspberry pi based on Internet of Things," 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Kochi, 2015, pp. 1144-1147.

[9] S. Chanthakit and C. Rattanapoka, "MQTT Based Air Quality Monitoring System using Node MCU and Node-RED," 2018 Seventh ICT International Student Project Conference (ICT-ISPC), Nakhonpathom, 2018, pp. 1-5.

[10] J. C. B. Lopez and H. M. Villaruz, "Low-cost weather monitoring system with online logging and data visualization," 2015 International Conference on Humanoid, Nanotechnology, Information Technology,Communication and Control, Environment and Management (HNICEM), Cebu City, 2015, pp. 1-6.

[11] [Online]: https://emoncms.org/. Accessed January 2019.

[12] [Online]: https://www.abbey.co.nz/papers.html. Accessed January 2019.