e-ISSN: 2278-8875
p-ISSN: 2320-3765

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

Volume 13, Issue 3, March 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.317

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.317|| A Monthly Peer Reviewed & Referred Journal |

||Volume 13, Issue 3, March 2024||

|DOI:10.15662/IJAREEIE.2024.1303035 |

Internet of Things-Based Photovoltaics Parameter Monitoring System Using Node MCU

Mrs. A Mary sumitha[1], Antroyons X [2], Karthikraja A[3], Prabakar M V[4], Thomson Q[5]

Associate professor, Department of EEE, Francis Xavier Engineering College, Tirunelveli, India[1]

Student, Department of EEE, Francis Xavier Engineering College, Tirunelveli, India[2]

Student, Department of EEE, Francis Xavier Engineering College, Tirunelveli, India[3]

Student, Department of EEE, Francis Xavier Engineering College, Tirunelveli, India[4]

Student, Department of EEE, Francis Xavier Engineering College, Tirunelveli, India[5]

ABSTRACT: This project proposes an Internet of Things-based photovoltaic parameter monitoring system using the Node MCU. A solar module is used to evaluate the suggested system's ability to measure voltage, current, temperature, and humidity. An Arduino-based system is used to monitor the parameters of a 10-watt solar panel. This technology uses the internet to send the power output to the IOT system while continuously monitoring the solar panel. The voltage is observed by the voltage and current sensors, respectively. The temperature and humidity of the atmosphere are measured using a DHT11 sensor. This PV monitoring system is developed by a smart Wi-Fi-enabled Arduino microcontroller with an ESP8266 processor that communicates and uploads the data to a cloud platform with the Blynk application.

KEYWORDS: Internet of Things (IoT),Photovoltaics (PV),NodeMCU, Monitoring System, Real-time Data, Sensor Integration, Cloud Connectivity, Data Transmission, Alert Mechanisms, Remote Access, Energy Optimization

I. INTRODUCTION

Throughout history, innovations and novelties have fuelled the development and progress of technology, which resulted in even further energy and time saved for even further improvements. One such example is the invention of electricity. Nowadays, electricity is not a luxury as it was previously, but an everyday necessity. More electricity demand resulted in the creation of new technologies for generating electrical energy. In order to provide sufficient energy for the public and reduce pollution, the attention shifted from conventional fossil fuels to renewable sources of energy. The list of renewable energy sources includes wind, tides, rain, and sunlight. The name “renewable” means that it comes from natural resources, thus, it is naturally replenished.

1.1 PROBLEM STATEMENT

- Generating electricity from solar radiation is considered as a clean way to produce power required as compared to conventional electricity generation by using fossil fuels.
- Every time the combustion reaction of fossil fuels and oxygen takes place, it emits greenhouse gases into atmosphere.
- Producing photovoltaic (PV) modules, on the other hand, is the only time when carbon footprint is established in solar electricity generation.
- One of the reasons is the cost and fragility of solar panels.
- The hardware circuit will be designed to provide the physical presence to the project.

1.2 OBJECTIVE

- To use the Internet to create a wireless solar panel monitoring system.
- To create a web page that will collect the data that has been transmitted and show it visually.
- To enable remote access to the monitoring data and provide control functionality.

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.317|| A Monthly Peer Reviewed & Referred Journal |

||Volume 13, Issue 3, March 2024||

|DOI:10.15662/IJAREEIE.2024.1303035 |

- To accurately measure and monitor the current flowing through the photovoltaic system.
- Enable manual control for navigation to fire sites.

1.3 SCOPE AND STUDY

This project will be focusing on monitoring of solar electricity generating system (SEGS). In order to do so, initially a literature review will be conducted on solar power and solar generating systems, as well as on available monitoring systems to understand the requirements of the future work. Moreover, initial assessment and experiments regarding output characteristics of the PV array under different conditions will be performed to study the factors affecting the performance of solar panels. The results of experiments will provide author with information needed to select the sensors to monitor the desired outputs. The hardware circuit will be designed to provide the physical presence to the project. Atmel microcontroller based Arduino Uno development board is planned to be employed in the circuit. After that, the software program will be created in order to control the hardware circuit. The program will be written in C language. Tests and simulations will be carried on to verify readings. And finally, the device will be implemented in the SEGS.

II. COMPONENTS

COMPONENTS AND SPECIFICATIONS:

- Photovoltaics (PV)
- NodeMCU
- Monitoring System
- Sensor NodeMCU
- Alert Mechanisms
- Cloud Connectivity
- Remote Access
- Real-time Data
- Energy Optimization
- Renewable Energy
- Internet of Things (IoT)

Photovoltaics (pv)

Photovoltaics refers to the technology that converts sunlight directly into electricity using solar cells. These cells are typically made of semiconductor materials, such as silicon, and when sunlight hits the cells, it generates an electric current. Photovoltaic systems are commonly used to harness solar energy for various applications, including powering homes, businesses, and even satellites.

Node MCU

NodeMCU is an open-source firmware and development board based on the ESP8266 Wi-Fi module. It allows for easy programming and deployment of IoT (Internet of Things) projects, as it combines the capabilities of the ESP8266 with a Lua-based firmware and a convenient development environment. NodeMCU boards are widely used for prototyping IoT applications, home automation, sensor monitoring, and more due to their low cost, built-in Wi-Fi connectivity, and ease of use.

Monitoring system

A monitoring system is a tool or set of tools designed to observe, track, and analyze various parameters, events, or activities within a specific environment. These systems can range from simple setups to complex networks of sensors, data collection devices, and software platforms. Monitoring systems are used in various fields such as:

- Environmental Monitoring: Tracking air quality, water quality, weather conditions, and ecological parameters.

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.317|| A Monthly Peer Reviewed & Referred Journal |

||Volume 13, Issue 3, March 2024||

|DOI:10.15662/IJAREEIE.2024.1303035 |

- Infrastructure Monitoring: Monitoring the health and performance of buildings, bridges, roads, and other structures to ensure safety and efficiency.

- Industrial Monitoring: Monitoring manufacturing processes, equipment performance, and industrial systems to optimize production and prevent downtime.

- Health Monitoring: Monitoring vital signs, patient conditions, and medical equipment in healthcare settings to improve patient care and safety.

- Security Monitoring: Monitoring surveillance cameras, access control systems, and alarms to enhance security and prevent unauthorized access or incidents.

Sensor nodemcu

Using sensors with NodeMCU (or any ESP8266-based board) is a common practice in IoT projects. Here's a general overview of how you can use sensors with

- NodeMCU: Selecting Sensors: Choose sensors that are compatible with NodeMCU and suitable for your project requirements. Common types include temperature, humidity, light, motion, gas, and proximity sensors. Wiring: Connect the sensor to the appropriate GPIO pins on the NodeMCU board. Refer to the datasheets or documentation of both the sensor and the NodeMCU board for pin mappings.

- Programming: Write code to read data from the sensor using the appropriate communication protocol (e.g., I2C, SPI, analog). You can use Arduino IDE, PlatformIO, or Lua (NodeMCU firmware supports Lua) for programming NodeMCU.

- Data Processing: Process the data obtained from the sensor as needed. This may involve converting raw sensor readings to meaningful values, performing calculations, or applying algorithms.

- Communication: Transmit the sensor data to a central server, cloud platform, or other devices using Wi-Fi (built-in on NodeMCU) or other communication protocols (e.g., MQTT, HTTP). Integration: Integrate the sensor data with your IoT application, dashboard, or automation system for visualization, analysis, or control.

Alert mechanicsm

An alert mechanism is a system designed to notify users or stakeholders about important events, conditions, or anomalies in real-time or near real-time. Here's how you can implement an alert mechanism in an IoT project using NodeMCU and sensors

- Threshold Monitoring: Set thresholds for sensor readings beyond which an alert should be triggered. For example, if you're monitoring temperature, you might set upper and lower limits for acceptable temperature levels. Continuous Monitoring: Continuously monitor sensor data at regular intervals using the NodeMCU board. This can be done by reading sensor values and comparing them to predefined thresholds.

- Alert Generation: If sensor data exceeds or falls below the predefined thresholds, generate an alert. This can be achieved by activating a buzzer, flashing an LED, sending an email, or triggering a notification to a mobile device.

- Alert Transmission: Transmit the alert information to the appropriate recipient(s). You can use Wi-Fi connectivity on the NodeMCU to send alerts via MQTT, HTTP requests, or other communication protocols to a central server or cloud platform. Alert Acknowledgement: Implement a mechanism for acknowledging alerts

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.317|| A Monthly Peer Reviewed & Referred Journal |

||Volume 13, Issue 3, March 2024||

|DOI:10.15662/IJAREEIE.2024.1303035 |

to prevent duplicate notifications. This could involve sending acknowledgment messages back to the NodeMCU or updating the alert status in a central database. Fallback

- Mechanism: Implement a fallback mechanism in case of network connectivity issues or hardware failures. For example, store alert information locally on the NodeMCU and resend it once connectivity is restored.

- Remote Configuration: Allow users to configure alert thresholds and notification preferences remotely. This can be done through a web interface or a dedicated mobile app. By implementing these steps, you can create an effective alert mechanism in your IoT project using NodeMCU and sensors, ensuring timely notifications of important events or condition

Cloud connectivity

Cloud connectivity in IoT projects allows devices like NodeMCU to communicate with cloud-based platforms for data storage, analysis, and management. Here's how you can establish cloud connectivity with NodeMCU

Remote access

Remote access in IoT projects allows users to monitor and control devices from a distance, typically over the internet. Here's how you can implement remote access with NodeMCU: Enable Remote Connectivity: Ensure that your NodeMCU board is connected to the internet via Wi-Fi. NodeMCU has built-in Wi-Fi capabilities, so you can connect it to your local Wi-Fi network. Port Forwarding (Optional): If your NodeMCU is behind a router or firewall, you may need to set up port forwarding to allow external access to the device. This involves configuring your router to forward incoming traffic on a specific port to the internal IP address of the NodeMCU. Dynamic DNS (Optional): If your internet connection uses a dynamic IP address, consider using Dynamic DNS (DDNS) to map a domain name to your changing IP address. This allows you to access your NodeMCU using a consistent domain name instead of an IP address. Secure Communication: Implement secure communication protocols to ensure the confidentiality and integrity of data transmitted between your NodeMCU and remote devices. Use protocols like HTTPS or MQTT with TLS/SSL for encrypted communication.

Real-time data

Real-time data in IoT refers to information that is collected, processed, and transmitted instantaneously, allowing for immediate analysis and action. Here's how you can handle real-time data with NodeMCU in IoT projects: Sensor Data Acquisition: Use NodeMCU to read sensor data at regular intervals or in response to events. This can include temperature, humidity, motion, light, and other environmental parameters. Data Processing: Process the raw sensor data on the NodeMCU board to convert it into meaningful information. This may involve filtering, averaging, or aggregating sensor readings to reduce noise and improve accuracy. Real-time Transmission: Transmit the processed sensor data to a central server or cloud platform in real-time using protocols like MQTT, WebSocket, or HTTP. NodeMCU's built-in Wi-Fi capability allows for seamless communication with internet-connected services.

Energy optimization

Energy optimization in IoT projects involves minimizing energy consumption while maintaining or improving the functionality and performance of connected devices. Here's how you can achieve energy optimization with NodeMCU in IoT projects: Low-power Hardware Selection: Choose energy-efficient components and sensors for your NodeMCU-based devices. Opt for sensors with low power consumption and sleep modes to conserve energy when not in use. Sleep Modes: Utilize sleep modes on the NodeMCU board to minimize power consumption during idle periods. NodeMCU supports various sleep modes, including deep sleep, which disables most of the board's components to conserve power. Wake-up Sources: Configure wake-up sources such as timers, interrupts, or external triggers to wake the NodeMCU from sleep mode only when necessary. This allows the device to remain in low-power mode for extended periods and wake up only when there is data to be collected or processed.

Renewable energy

Renewable energy refers to energy derived from naturally replenished sources that are continuously available and sustainable over the long term. These sources include sunlight, wind, water (hydroelectric), geothermal heat, and biomass. Here's how renewable energy is harnessed and its significance: Solar Energy: Solar energy is harnessed using photovoltaic (PV) cells to convert sunlight directly into electricity, or through solar thermal systems that use sunlight to heat water or air for various applications, including electricity generation and heating. Wind Energy: Wind turbines capture the kinetic energy of wind and convert it into electricity. Wind farms, located onshore or offshore, generate large amounts of clean energy to power homes, businesses, and communities.

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.317|| A Monthly Peer Reviewed & Referred Journal |

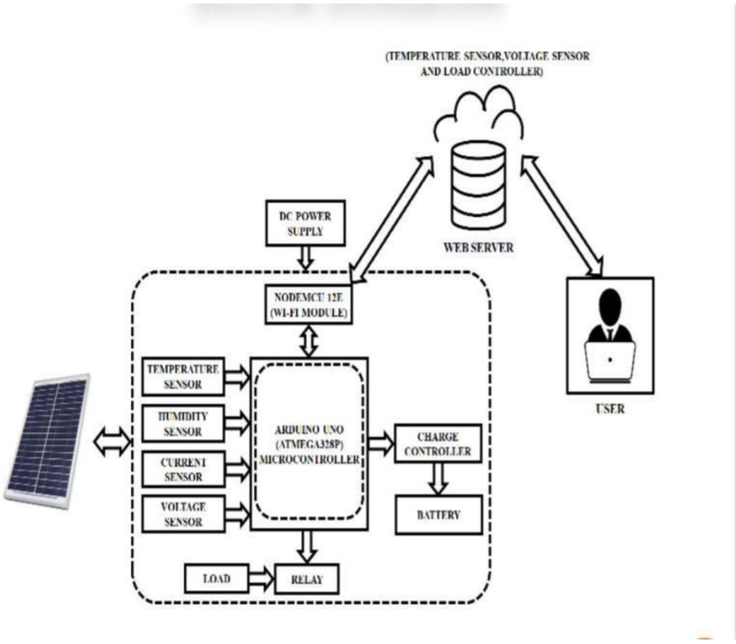
||Volume 13, Issue 3, March 2024||

|DOI:10.15662/IJAREEIE.2024.1303035 |

Internet of things (IOT)

The Internet of Things (IoT) is a network of interconnected devices, sensors, actuators, and other objects that communicate and exchange data over the internet. These "smart" devices collect, transmit, and receive data, enabling them to interact with each other and with centralized systems. Here's a breakdown of key aspects of IoT Connectivity: IoT devices are typically connected to the internet or to local networks via wired or wireless connections such as Wi-Fi, Bluetooth, Zigbee, or cellular networks. This connectivity enables devices to exchange data and interact with each other and with cloud-based services. Sensors and Actuators: IoT devices are equipped with sensors to collect data from the environment, such as temperature, humidity, motion, light, and more. Actuators enable devices to perform actions based on this data, such as controlling lights, thermostats

BLOCK DIAGRAM



Working

The IoT-based solar power monitoring system is designed to continuously monitor and display the electrical power generated, temperature, and light intensity of a solar panel. The system works by measuring the current, voltage, and temperature of the solar panel using sensors and transmitting the data wirelessly to data visualization interfaces such as an LCD display, a mobile application, and a computer screen.The working of the system can be divided into three parts: sensing, processing, and visualization. Sensing: The first step in the system is sensing, where the current, voltage, temperature, and light intensity of the solar panel are continuously measured using the corresponding sensors. The voltage and current sensors measure the electrical power generated by the solar panel, while the temperature and light sensors measure the temperature and light intensity respectively.Processing: Once the data is sensed, the ESP32 microcontroller processes the data using the pre-installed software. The software on the microcontroller analyses the data to calculate the power generated by the solar panel, the temperature of the panel, and the light intensity. The microcontroller then compares the calculated values with the preset thresholds and generates alerts if any of the parameters exceed the set limits. Visualization: The last step in the system is the visualization of data, which involves displaying the real-time data on the LCD display, mobile application, and computer screen. The LCD display shows the real-time data on the electrical power generated, temperature, and light intensity. The mobile application and computer screen also display the real-time data, allowing the user to monitor the system remotely. The data can be saved in a database for further analysis and comparison. Overall, the system continuously measures the parameters of the solar panel, processes the data, and generates alerts and real-time data visualization. The user can monitor the system's performance and make necessary adjustments, resulting in better performance and longer lifespan of the solar panel.

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.317|| A Monthly Peer Reviewed & Referred Journal |

||Volume 13, Issue 3, March 2024||

|DOI:10.15662/IJAREEIE.2024.1303035 |

III. RESULT AND DISCUSSIONS

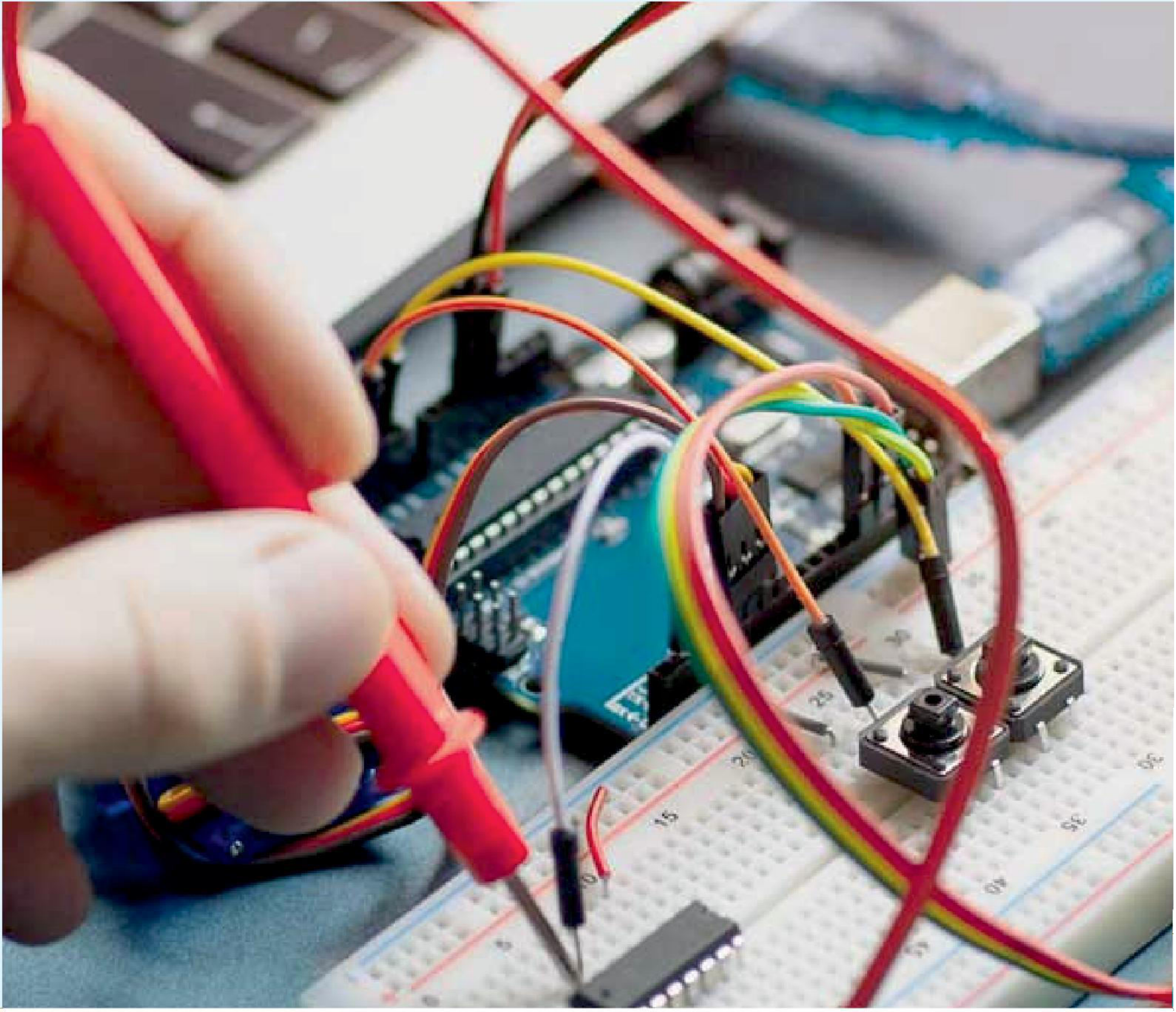
The PV is used as a power generator and the laptop is used to view graphic data obtained via the ThingSpeak platform. The electrical measurement device is used to verify the accuracy of the electric sensor, the breadboard is used to attach the sensor to the mainboard, the box contains the mainboard and the PWM solar charge controller, and the 12 V battery serves as an energy backup system. The graphical data was obtained from the ThingSpeak platform. The PV parameter measurements, include solar irradiation level, ambient temperature, PV output voltage, and current, as well as the PV output power. The linear relationship between solar irradiation, PV output current, and PV output power values indicates that the sensor readings and data collecting system are working appropriately. In around 30 seconds, the sensor collects, analyses, transmits, and stores the data in a cloud server. Meanwhile, the cost comparison of several key controllers used in the IoT-based PV monitoring system. This is to show that the system's development costs are the lowest compared to others.

IV. CONCLUSION

An IoT-based PV parameter monitoring system was introduced in this paper. The parameters such as solar irradiance, ambient temperature, PV output voltage, PV output current, and PV output power are all measured by the system. Moreover, solar irradiance, ambient temperature, PV output voltage, and PV output current are measured using a photodiode, DHT22, impedance divider, and ACS712, respectively. Besides, PV output power is obtained from the product of PV voltage and PV current. The main controller of the system is the NodeMCU V3 ESP8266, which also has the ability to create a WiFi-enabled gateway. As a result, the calculated parameters can be transferred wirelessly from the sensor to the cloud without the need for an external WiFi module. Finally, the acquired parameters are tracked on the Cloud server using ThingSpeak. The selected components for the system are low-cost and readily available on the market. The reliability of each component also a factor to consider. Finally, laboratory experiments are used to verify the system. The outcome clearly shows that the system offers accurate results in monitoring the PV parameters. It is important to test the system under a variety of environmental conditions over a prolonged period of time for future work.

REFERENCES

1) Hashem Mortazavi, Hasan Mehrjerdi, Maarouf Saad, Serge Lefebvre, Dalal Asber, and Laurent Lenoir, "An Impedance-Based Method for Distribution System Monitoring", IEEE Transactions on Smart Grid, vol.9, no.1, pp.220 – 229, 2018.
2) Younes Seyedi, Houshang Karimi, Santiago Grijalva, "Irregularity Detection in Output Power of Distributed Energy Resources Using PMU Data Analytics in Smart Grids", IEEE Transactions on Industrial Informatics, vol.15, no.4, pp.2222 – 2232, 2019.
3) Hamid Shaker, Hamidreza Zareipour, David Wood, "A Data-Driven Approach for Estimating the Power Generation of Invisible Solar Sites", IEEE Transactions on Smart Grid, vol.7, no.5, pp. 2466 – 2476, 2016.
4) Hamid Shaker, Hamidreza Zareipour, David Wood, "Estimating Power Generation of Invisible Solar Sites Using Publicly Available Data", IEEE Transactions on Smart Grid, vol.7, no.5, pp. 2456 – 2465, 2016.
5) Qunlong Duan, Nguyen Vu Quynh, Heba M. Abdullah, "Optimal Scheduling and Management of a Smart City Within the Safe Framework", IEEE Access, vol.8, pp.161847 – 161861, 2020.
6) Gi-Ren Liu, Phone Lin, Yuguang Fang, Yi-Bing Lin, "Optimal Threshold Policy for In-Home Smart Grid with Renewable Generation Integration", IEEE Transactions on Parallel and Distributed Systems, vol.26, no.4, pp.1096 – 1105, 2015.
7) Sol Moon, Sung-Guk Yoon, Joung-Hu Park, "A New Low-Cost Centralized MPPT Controller System for Multiply Distributed Photovoltaic Power Conditioning Modules", IEEE Transactions on Smart Grid, vol.6, no.6, pp.2649 – 2658, 2015



INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.317

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

International Journal of Advanced Research

in Electrical, Electronics and Instrumentation Engineering