Solar Battery Charger By Monitoring Via Smartphone Application

Thammakorn Krongtripop[1], Putthiphong Kirdpipat[2*], Jessada Konpang[2], Nattapong Intarawiset[3] and Kobkhun Chaiyawong[4]

[1]Department of Computer Engineering, Faculty of Engineering and Technology, Rajamangala University of Technology Isan, Nakhon Ratchasima

744 Nai Mueang, Mueang Nakhon Ratchasima, Nakhon Ratchasima, Thailand, 30000

[2]Department of Electrical and Telecommunication Engineering, Faculty of Engineering, Rajamangala University of Technology Krungthep

2 Nanglinchee Road, Tungmahamek, Sathorn, Bangkok, Thailand, 10120

[3] Department of Industrial Education and Technology, Faculty of Engineering, Rajamangala University of Technology Lanna

128 Huay Kaew Road, Muang, Chiang Mai, Thailand, 50300

[4]Department of Electrical Engineering, Faculty of Industry and Technology, Rajamangala University of Technology Isan, Sakhon Nakhon Campus

199 Phang Khon-Waritchaphum Road, Phang Khon District, Sakhon Nakhon, Thailand, 47160

[*]Corresponding Author: putthiphong.k@mail.rmutk.ac.th. Phone Number: 09-2615-2695

Received: 1 December 2022, Revised: 27 April 2023, Accepted: 27 April 2023

Abstract

Solar cell monitoring system using the Blynk application has some restrictions in crossing platforms for Android and iOS. Because the widget boxes of the Blynk application are limited in the energy usage to only 2,000 joules, the web dashboard also has complexity for the application. This paper proposes a solar battery charger monitoring mobile application for Android and iOS platforms with Firebase, which can operate cross-platform by using Ionic Framework. Cordova is mainly the management of Ionic Framework, which will be getting commands to form HTML, CSS, and Angular JS to each application of Android and iOS. We implement solar battery charger by using simple buck converter interfaces with Arduino board under Pulse Width Modulation to control the battery charger, and using ACS712 module and voltage divider circuit interface with Arduino to measure the current and battery's voltage while charging the battery. NodeMCU is interfaced with an Arduino module for transferring all data via WiFi to Android and iOS Smartphones. Those data will be transferred for the record on Firebase at Cloud Server. For the experimental results, the solar charger can fully charge the battery in 10-12 hours which may depend on the sun's irradiation each day. The average battery voltage after fully charged is 13.8V. On Smartphones of Android and iOS, our applications can control and display the operation correctly. In addition, the application displays the history of battery charging as well. Our research indicates that Firebase has a significant and necessary crossing platform capability on Smartphones that can operate for Android and iOS.

Keywords: Solar Cell, Monitoring System, Blynk Application, Solar Battery Charger

1. Introduction

Recently, the advance of Radio Frequency (RF)/ Solar energy is a significant subset of renewable energy. It is the fastest-growing source in every country, including THAILAND [1]. Solar energy may replace electrical power usage and help save electricity bills. Traditionally, electrical production may be caused by fossil fuel exhaustion and environmental pollution problems. However, solar energy has become clean energy for all environments. Many organizations in THAILAND have applied photovoltaic cells in their work, like at hospitals, office buildings, factories, etc. A photovoltaic cell, typically known as a solar cell, is a device that converts solar energy to electric power directly. The solar cell is usually made from semiconductor materials like Silicon (Si) [2], Gallium Arsenide (GaAs) [3], and Cadmium Telluride (CdTe) [4]. The efficiency of solar panels may depend on various factors, i.e., sun position, irradiation, temperature and humidity, dust and shade effect. Currently, several groups of researchers have been presented, such as solar panel tracking [5]-[9], the impact of the environment [10]-[14], solar application [15]-[19], solar battery charger [20]-[23] and solar monitoring system [24]. Solar panel tracking [5]-[7] applied LDR to detect and follow the sun's position to adjust the solar panel's tilt. The moving of the sun's position is caused by low voltage and low current in the solar cell during the daytime. The next research group is impacting the environment for the solar cell, which causes by temperature and humidity [10]-[11], [13], dust [11]-[12], and shade [11]-[12], [14]. Those of all factors may cause decreasing efficiency. Solar application [15]-[19] is hardware implementation using the photovoltaic power supply. Solar battery charger [20]-[23] charges the battery by photovoltaic during the day to support any application. The solar monitor system [24] monitors solar panels' voltage, current, and power during the day. For instance, the paper in [20] proposed a design of a battery charger that fed from two solar panels for an E-rickshaw. All authors used one solar panel mounted on the top and another at the backside of the E-rickshaw. Both solar panels can track the sun's position by an elevation angle of about 10-15 degrees with Maximum Power Point Tracking (MPPT). This method has more efficiency than a single solar panel. All authors in [21] proposed the designing and implementation of portable chargers for lithium-ion using the solar panel. Because the lithium-ion battery is widely used in electronic components, Arduino Nano is applied to process and control this battery charger. In the paper, [22] proposed the hybrid electric charging station by monitoring the voltage, current, and power via an IoT system. Wind and solar cells are combined to charge the electric vehicle's battery. A Monitor system has been proposed under the Blynk platform, which is managed by Raspberry Pi. The article in [23] proposed a portable solar mobile system with an IoT application. This prototype comprises many features such as the monitoring of battery charger, temperature, and humidity of power distribution box by showing via LCD and Blynk application on Smartphone. The research in [24] has also proposed a monitor system of solar panels which displays through the Blynk application on smartphones. The voltage, current, and temperature are monitored in this project. Therefore, Arduino Mega2560 is used for controlling the operation. The authors have proposed a monitor system for solar cells only without battery charging.

However, those previous papers have proposed solar battery chargers [20]-[23]. Some of the papers may be applied the monitor system under an IoT platform such as a Blynk application [22]-[23]. The Blynk platform still has some restrictions as follows: 1) Blynk application has many devices in widget boxes which are limited by energy usage at 2,000 joules only. For this reason, if we need more devices or the energy usage is over 2,000 joules, we must buy this option. 2) Blynk application can’t transfer variable value to the other platforms. This application means that it is difficult for cross-platform applications, and 3) the web dashboard in the Blynk application is not easy to use.

So, our paper presents a simple solar battery charger that monitors Mobile Applications without using Blynk, but we developed Mobile Application based on Firebase, which is more flexible. Firebase can cross-platform for both Android and iOS. We have several features, such as a battery charger, controlling and displaying, warning, and battery charging history. The battery charger is the initial voltage and current setting for solar cells and battery charging. Controlling and displaying are setting and showing the status of charging the battery via Smartphone, such as date and time in the present, voltage and current from the solar cell, voltage and current at the battery, status of battery charging, and quality of WiFi system. The warning alerts the battery charging fault if the voltage and current of the solar cell are different from the setting values. The history of battery charging shows all the information for the past month. Our work indicates that Firebase is significant in crossing platforms on Mobile Applications in Android and iOS.

2. Materials and Methods

2.1 General description of proposed system

Blynk application is a platform of IoT system for monitoring and controlling via Smartphone. Several articles have proposed this platform for Mobile Applications like smart farms [15]-[17] and smart homes [18], including solar battery chargers [22]-[23]. This platform can easily develop the hardware application if the user primarily knows the Arduino program. However, some options may be limited if we require many devices in widget boxes or if there are various Smartphones for each user. For this reason, we decided to cross-platform Smartphones by using Firebase to monitor solar battery chargers. Our proposed paper may help motivation for other applications. The solar battery charger system comprises a simple buck converter that adjusts the voltage to match each battery. Pulse Width Modulation (PWM) signal from the Arduino board will control the operation of the buck converter by switching ON and OFF continuously while charging the battery. The PWM method has OFF status for a short time in battery charging. This method may reduce the overcharge, improving the better chemical action in the battery. Our work comprises an Arduino board that controls all operations by measuring the voltage and current of solar cells and battery using the ACS712 module and display on LCD. To transfer all data to Mobile Application, we used NodeMCU to connect via the WiFi system by interfacing with the Arduino module. We implemented our work to monitor the solar battery charger using Firebase, which is more flexible than the Blynk platform. We also apply Ionic Framework for displaying on iOS.

2.2 Blackgroud and motivation

In this section, we explain the hardware components in our work that consists of several devices such as NodeMCU, Firebase, and Ionic Framework, respectively.

2.2.1 NodeMCU

NodeMCU is a microcontroller module that can transfer all data to a Smartphone application via a WiFi system. This node is more flexible than the Arduino board by supporting the Internet of Things. Several features are supported, i.e., 32 bits processor, memory flash 4 Mbs, low voltage operation, USB port, SPI port for SD card, and one analog to a digital channel, as shown in Fig. 1.

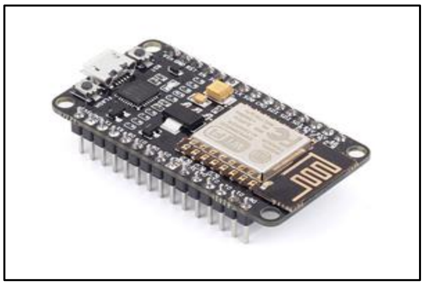


Figure 1 NodeMCU [15]

2.2.2 Firebase

Firebase [25] is a real-time database for any application, web application, and Mobile Application based on a Google platform service for IoT systems. This database is mainly core for interfacing and transferring all data between NodeMCU and Raspberry Pi to Android and iOS. It may support both Python and JavaScript languages. Firebase is a database in the form of NoSQL, which keeps all data in JSON form. There is rapid real-time synchronization with each device, and it supports an offline system. This real-time means that all data can keep in the locally until. After the system becomes online again, the synchronization will continuously update. There is also a security rule in which the user can assign the condition of accessing the data for reading and writing in Android and iOS. Firebase has a logo, as shown in Fig. 2. There are many tools and services, as follows:

Firebase

Figure 2 Firebase Logo [25]

Cloud Firestone: This feature is saving and synchronizing of data between the user and device based on NoSQL at the host on the Cloud Server. This synchronization is a real-time operation that may support offline systems, including investigating all data.

Authentication: This is managing the safety of Firebase authentication. There are various methods to confirm by using email and password. It may also combine with the third applications like Google, Facebook, and Twitter.

Hosting: This function is reducing the complexity of web hosting. When the user uploads all web application content, this service automatically sends all content to CDN worldwide. The user will get a certificate of SSL for safety and reliability.

Crashlytics: It is management and real-time reporting of each problem on Firebase, respectively, which helps the user to reduce the overall times of solving.

Performance Monitoring: This feature is an analysis of the problem on the application in the device and improves the performance of that application.

Test Lab: This function automatically tests the application on the device by Google, which uses the Firebase test lab during development to find the defect.

2.2.3 Ionic Framework

Ionic Framework [26] is a tool for developing Mobile Applications by applying for the basic program, i.e., HTML, CSS, and JavaScript languages for development which will send out the command code in the form of each application to Android, iOS, and Windows Mobile. There are many advantages, such as open source, community, one application to many platforms, component creation, comprehensive browser support, UI design, Web technology, and simplicity. In addition, it can use Angular Framework, which can support Web Applications by using Cordova for management. The structure of the Ionic Framework is shown in Fig. 3.

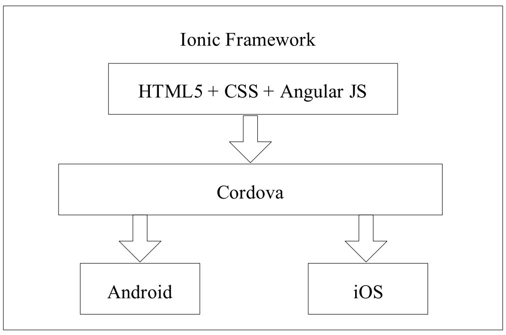


Figure 3 Ionic Framework [26]

2.3 Hardware implementation

Our proposed system is managed by an Arduino module which interfaces with several devices. We divide our circuit into many parts, i.e., battery charger by solar energy, voltage, and current measurements for battery and solar cell and Mobile Application, as shown in Fig. 4. Our concept supports Mobile Applications by transferring all data to Internet Network via a WiFi system using NodeMCU interfaces with an Arduino module. We consider that Firebase may be cross-platform for Android and iOS, which is more flexible than using the Blynk application for the monitoring system.

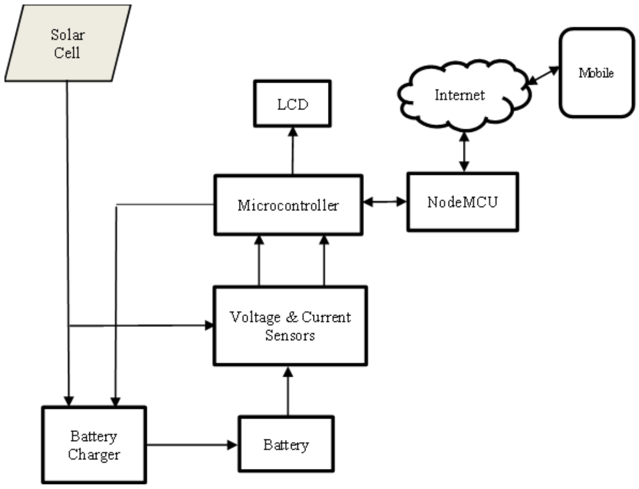


Figure 4 Block Diagram of Proposed System

Fig. 5 is illustrated all our circuits. The voltage and current of the battery and solar cell are measured by ACS712 modules and sent those values to the Arduino board for adjusting the duty cycle of Pulse Width Modulation (PWM), which corresponds with the voltage and current as user setting for each battery. PWM signal will be transferred to the buck converter circuit [27] to reduce the voltage and current for each suitable battery. This is because the voltage and current from the solar cell are commonly more than the voltage and current of the battery, which is improper for the charger. This is also a cause of degradation and provides a short lifetime. Arduino module will display the voltage and current while charging on LCD and transferring all information via WiFi by NodeMCU for Mobile Application.

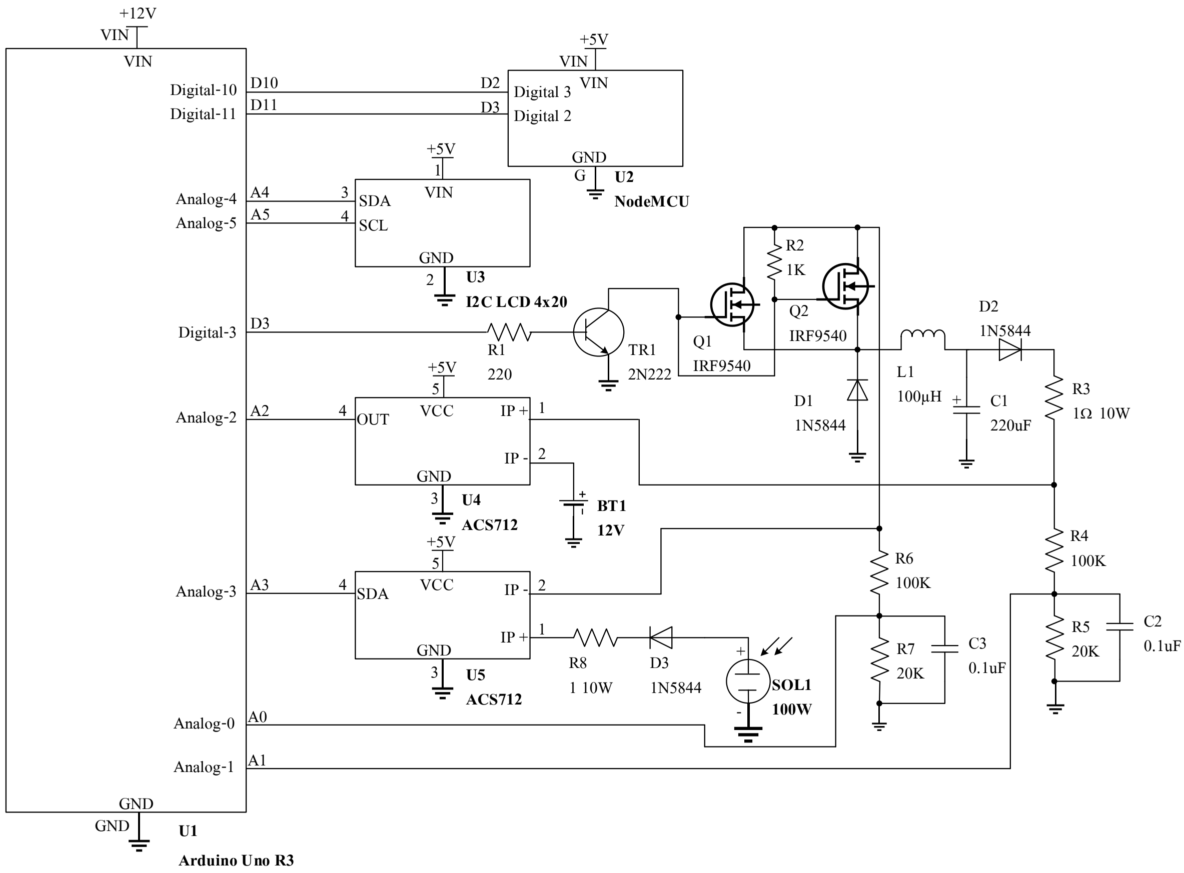


Figure 5 All Circuits of Solar Battery Charge

The buck converter comprises two MOSFETs, a capacitor, inductor, resistor, and diode, which get PWM signal from the Arduino board via pin-3 by using transistor 2N222 to drive the PWM signal. This signal will be sent to MOSFETs, switching to the discontinuous mode for charging the battery. Our circuit's limited voltage and current can be received at 12 V and 10 A, respectively.

Our program is divided into 3 parts. The first is a battery charging controller, the second transmits all data via NodeMCU, and the last displays the battery status on Mobile Application. The battery charger program has cooperated with voltage and current sensors, which composes of several functions, i.e., checking and displaying the sensor status, initial setting of voltage and current before battery charging, setting the time for charger, battery charging, and transferring all data to NodeMCU. Transmitting all data via NodeMCU is composed of several functions, such as connection to the Firebase function, getting data from the application, getting data from Arduino, transmitting all data to the application on Firebase, and so on. The last program shows the battery status on the Mobile Application, which consists of getting the battery status from Firebase and displaying functions, ON-OFF function, setting the voltage and current functions, displaying the history of battery charging function and fault of charging function, etc.

3. Results and Discussion

Our experiments are divided into 3 parts: battery charging, controlling and displaying via Mobile Application, and recording and showing the history of battery chargers, respectively.

This first experiment is battery charging. We have many parameters, i.e., deep cycle battery is 12 V at 7 A and solar panel is 100 W. We use the light meter model of TM-202, and the multimeter model is Amprobe AM-250. We test our battery charging under total experiments are 3 times by measuring the voltage and current 3 times to calculate the average values versus the average light intensity. The battery is 3 sets, and we use 1 set per 1 time experiment for 2 days. Each day's experiment is 6 hours, starting at 9 A.M. until 3 P.M. This is because the sun has less light intensity after 3 P.M., which may not be enough for battery charging. The total days of all experiments cover 6 days. We depict the details of the fully charged battery experiment in Table 1. Our work also implements the abnormal notification that may occur while charging the battery and will alert the application. However, the user must find out the possible cause: 1) Solar cell voltage is less than battery voltage which means no charge. 2) The battery's voltage is over 14.1 V, which means an overcharge. 3) Solar cell voltage is less than 15 V, charging in lower charge mode. In addition, we also record all information in Firebase on Cloud Server, which will be depicted in the history part.

Table 1 The details of charging the battery

No.	Day	Time	Status
1.	1-2	9 A.M. to 3 P.M. of each day	fully charged battery in 2 days
2.	3-4	9 A.M. to 3 P.M. of each day	fully charged battery in 2 days
3.	5-6	9 A.M. to 3 P.M. of each day	fully charged battery in 2 days
Average days of 3 times			2 days

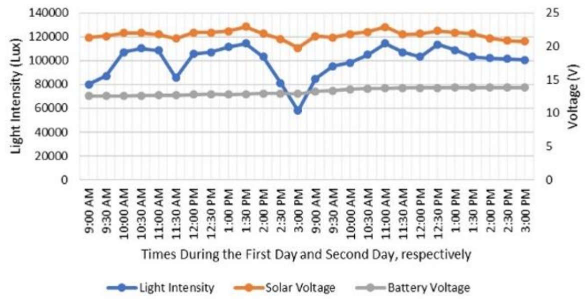


Figure 6 Average Voltages of Solar Cell and Battery versus Light Intensity during 2 Days

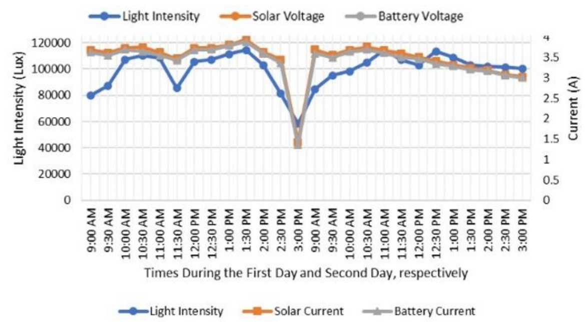
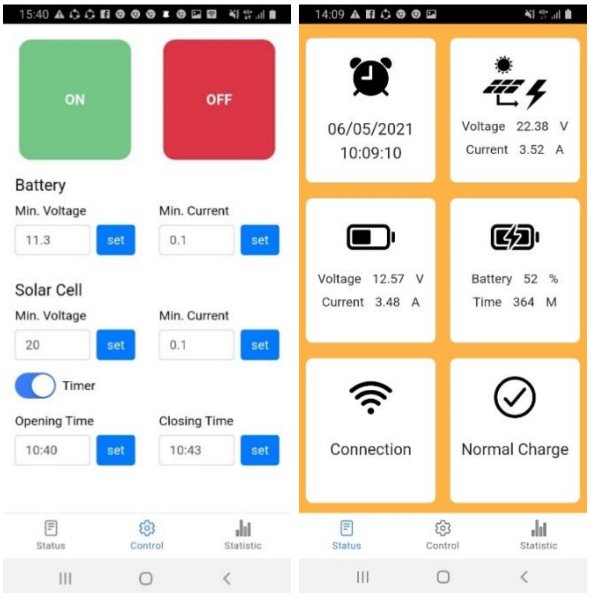


Figure 7 Average Currents of Solar Cell and Battery versus Light Intensity during 2 Days

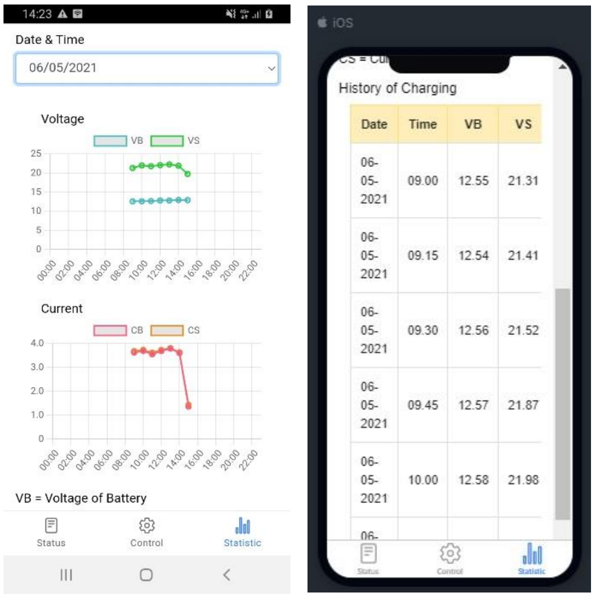
Figs. 6 and 7 are depicted the average voltage and current versus the average light intensity. We observe that the increase in light intensity will provide the average voltage and current of the solar cell are changed as well. However, the voltage of the battery is still fixed for charging. During the experiment, we can see that the average of current is dramatically low at 1.354 A of 58200 luxs because the radiation of the sun decreases after 3 P.M. on the first day, and we start our experiment continuously at 9 A.M. on the second day, which we got 84700 luxs for beginning at 3.595 A. We found that the time of fully charged approximately 10-12 hours which covers 2 days of each experiment. The total of experiments are 3 times in 6 days which will be considered in terms of average value as described previously. Typically, the fully charged time of the battery may also depend on the other parameters, i.e., humidity, temperature, and the moving of the sun's position each day. At the fully charged, we got a voltage equal to 13.8V.

The following experiment is controlling and displaying via Mobile Applications for Android and iOS. This experiment is based on a WiFi connection. We consider the function of ON/OFF controlling and setting the initial voltage and current values, including the other parameters, before charging. We test the function of each point 25 times so that it can operate and display correctly. Displaying fault means the user may set the voltage and current or some parameters incorrectly. All of the experiments areshown in Fig. 8.



(a) (b)

Figure 8 The Parameters Based on Android Smartphone, (a) Initial Setting, (b) Operation and Display



(a) (b)

Figure 9 History of Battery Charging, (c) Graph under Android, (d) Table under iOS

The last experiment is recording and displaying the history of the battery charger. The recording is done during battery charging in the first step. We test battery charging and also record the voltage and current every 15 minutes on the Firebase platform. The total results can plot the graph and table for displaying via Smartphone as shown in Fig. 9 (c) under Android and Fig. 9 (d) under iOS, respectively. As previous mention, we proposed the cross-platform based on firebase. We applied Ionic Framework for testing on the iOS platform. Our proposed technique can operate on both Android and iOS.

4. Conclusions

We have proposed a solar battery charger by monitoring both Android and iOS. This is because the solar cell monitoring system under the blynk application is limited for several items. It is difficult to monitor on various platforms, such as Android and iOS. Our work has implemented a solar battery charger system, thereby using the cross-platform method for a monitoring system. We have applied Firebase and Ionic Framework in crossing platform. Cordova is a significant core in Ionic Framework, which can manage by getting commands from HTML, CSS, and Angular JS for transferring and displaying on various platforms, i.e., Android and iOS. We applied Arduino and NodeMCU boards to process the simple buck convertor in the solar battery charger under the PWM controller. ACS712 module and voltage divider circuit are interfaced with Arduino to measure the current and voltage while charging the battery. NodeMCU is applied for relaying all information to Firebase on the Cloud Server, which is a database of this system. As the experimental results, we found that a solar battery charger can charge the battery for 10-12 hours during 2 days continuously. The fully charged battery can be achieved at 13.8 V. We have also recorded the battery charging history. In future work, we will apply the machine learning method to learn the abnormal light intensity each day to predict the time of fully charging the battery. The advantage of this technique is that it may help the user to plan battery usage correctly.

5. Acknowledgment

We thank Mr. Chittisan Wichito, Mr. Kritsanapong Phonok, and Miss. Supattra Chitsom from the Department of Computer Engineering, Faculty of Engineering and Technology, Rajamangala, University of Technology Isan, Nakhon Ratchasima, Thailand, which helps support and provide experiments of this research, has finished successfully.

6. References

[1] Yoomak S, Patcharoen T, Ngaopitakkul A. Performance and economic evaluation of solar rooftop systems in different regions of Thailand. Sustainability. 2019 Nov 25;11(23):6647.

[2] Siddula S, Gleeson CD. Solar Panel Position Control and Monitoring System For Maximum Power Generation. In2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE) 2020 Oct 9. pp. 169-174.

[3] Outes C, Seoane N, Almonacid F, Fernández EF, García-Loureiro AJ. Study of GaAs vertical solar cells at high concentration levels. In2018 Spanish Conference on Electron Devices (CDE) 2018 Nov 14. pp. 1-3.

[4] Pandey R, Drayton J, Gregory C, Kumar NM, Tyler K, King R, Sites J. Cadmium Telluride Cells on Silicon as Precursors for Two-Junction Tandem Cells. In2020 47th IEEE Photovoltaic Specialists Conference (PVSC) 2020 Jun 15.pp. 1326-1329.

[5] Hassan R, Abubakar B. Intelligent arduino based automatic solar tracking system using light dependent resistors (LDRs) and servo motor. Optics. 2020;9(2):13.

[6] Rosma IH, Putra IM, Sukma DY, Safrianti E, Zakri AA, Abdulkarim A. Analysis of single axis sun tracker system to increase solar photovoltaic energy production in the tropics. In2018 2nd International Conference on Electrical Engineering and Informatics (ICon EEI) 2018 Oct 16. pp. 183-186.

[7] Ghosh J, Dey N, Das P. Active solar tracking system using node mcu. In2019 International Conference on Computing, Power and Communication Technologies (GUCON) 2019 Sep 27. pp. 924-928.

[8] Jurj SL, Rotar R, Opritoiu F, Vladutiu M. White-box testing strategy for a solar tracking device using nodemcu lua esp8266 wi-fi network development board module. In2018 IEEE 24th International Symposium for Design and Technology in Electronic Packaging(SIITME) 2018 Oct 25. pp. 53-60.

[9] Surendhar PS, Tamilselvan S. Efficient solar tracking system using GPS. International Journal for Research in Applied Science and Engineering Technology (IJRASET), 2018, 6(4): 3842-3849.

[10] Samal KB, Bisoyi A. Investigation of enviromental effects on the performance of solar PV modules. International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET), 2020 July 10-11; Patna, India: p. 1-5.

[11] Mughees M, Sadaf M, Khan S, Rasheed A, Daniel K, Gelani HE. Impact of Shading, Dust Accumulation and Temperature rise on the Performance of Solar PV: The Case of Pakistan. In2020 International Symposium on Recent Advances in Electrical Engineering & Computer Sciences (RAEE & CS) 2020 Oct 20. Vol. 5, pp. 1-6.

[12] Phoolwani UK, Sharma T, Singh A, Gawre SK. IoT based solar panel analysis using thermal imaging. In2020 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS) 2020 Feb 22. pp. 1-5.

[13] Yoomak S, Jettanasen C, Ngaopitakkul A. Design of solar charger challenging various solar irradiance and temperature levels for energy storage. IJICIC. 2018 Dec 1;14(6):2071-90.

[14] Sunanda W, Gusa RF, Tiandho Y, Pratama EA. Impact of shading net on photovoltaic cells performance. Jurnal Teknik Elektro. 2019;11(2):56-60.

[15] Chieochan O, Saokaew A, Boonchieng E. Internet of things (IOT) for smart solar energy: A case study of the smart farm at Maejo University. In2017 international conference on control, automation and information sciences (ICCAIS) 2017 Oct 31. pp. 262-267.

[16] Shukla A, Jain A. Smart Automated Farming System using IOT and Solar Panel. Science and Technology Journal. 2019;7(2):22-32.

[17] Udutalapally V, Mohanty SP, Pallagani V, Khandelwal V. sCrop: A Internet-of-Agro-Things (IoAT) enabled solar powered smart device for automatic plant disease prediction. arXiv preprint arXiv:2005.06342. 2020 May 9.

[18] Manur A, Marathe M, Manur A, Ramachandra A, Subbarao S, Venkataramanan G. Smart solar home system with solar forecasting. In2020 IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy (PESGRE2020) 2020 Jan 2. pp. 1-6.

[19] Tabassum Z, Shylaja V. An IoT based garbage collection system with solar powered smart trash bin using NodeMCU. International Journal of Engineering Science and Computing (IJESC), 2021, 11(7): 28490-28494.

[20] Haldar I, Debnath D, Choudhury TR. Design of a battery charger fed from two solar panels arranged at different inclination for e-rickshaw and dc microgrid applications. In2021 1st International Conference on Power Electronics and Energy (ICPEE) 2021 Jan 2. pp. 1-6.

[21] Hudy W, Matviikiv O, Pytel K, Gumula S, Kozioł M, Bilski K. Experimental Solar-Based Rechargeable Battery Charger as an Example of Engineering Education. In2021 IEEE XVIIth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH) 2021 May 12. pp. 86-89.

[22] Madharshan A, Aravinth, Dheneshraajan, Gokul, Praveena P, Hybrid electric charging station using raspberry Pi. International Journal of Engineering and Advanced Technology (IJEAT), 2020, 10(1): 1-3.

[23] Mariadass SP, Al-Humairi SN, Daud RJ. Design a portable solar mobile system with an IoT integration-based. In2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC) 2021 Aug 7. pp. 40-45.

[24] Gusa RF, Sunanda W, Dinata I, Handayani TP. Monitoring system for solar panel using smartphone based on microcontroller. In2018 2nd international conference on green energy and applications (ICGEA) 2018 Mar 24. pp. 79-82.

[25] Firebase Logo. Available from: https://firebase.google.com.

[26] Waranashiwar J, Ukey M. Ionic framework with angular for hybrid app development. International Journal of New Technology and Research. 2018 May;4(5):263068.

[27] Arduino PWM Solar Charger. Available from: https//www.instructables.com/arduino-solar-charge-controller-pwm.