JJEE
Jordan Journal of Electrical Engineering

Volume 8, Number 1, March 2022
Pages 65-89
ISSN (Print): 2409-9600, ISSN (Online): 2409-9619

A Solar Energy System with Vehicle-to-Home and Vehicle-to-Grid Option for Newfoundland/Canada Conditions through Mozilla IoT

Raghul Suraj Sundararajan[1*], M. Tariq Iqbal[2]

[1, 2] Electrical Engineering Department, Memorial University of Newfoundland, St. John's, Newfoundland, Canada
E-mail: rssundararaj@mun.ca

Received: June 21, 2021 *Revised: September 20, 2021* *Accepted: September 29, 2021*

Abstract— The hardware implementation of a solar energy system with vehicle-to-home (V2H) and vehicle-to-grid (V2G) options for Newfoundland conditions through Mozilla IoT is discussed in this paper. To illustrate IoT, remote monitoring and control concepts, a prototype - is entirely a 12 V system - is created in the lab. To operate in multiple modes, the system checks the current and voltage parameters. The data is transmitted to the gateway using an ESP 32 microcontroller and the internet of things (IoT). Mozilla IoT is the platform that hosts the Raspberry Pi Things Gateway and serves as a dashboard to remotely control and monitor the system. The data that is transmitted is logged, and the logged data is shown as a graph. This paper presents the system design, details of demo experimental setup in addition to the test results which reveal that the successful implementation of proposed system - with V2H and V2G options - for Newfoundland/ Canada conditions.

Keywords— Solar energy; Photovoltaics; Vehicle-to-home; Vehicle-to-grid; Newfoundland/Canada; Internet of things; Mozilla IoT.

1. INTRODUCTION

Electric vehicles (EVs) are the next major advancement in the transportation industry. EVs offer several benefits, including being more powerful than vehicles with internal combustion engines and functioning as mobile power reservoirs. EVs must be charged, and renewable energy sources like solar energy systems can help satisfy this need while also reducing carbon emissions. A backup battery is required since renewable energy is naturally variable. Because EVs can also function as reservoirs, vehicle-to-home (V2H) and vehicle-to-grid (V2G) concepts can be implemented. We have already published sizing and system design in [1] and dynamic simulation was carried out with the help of MATLAB Simulink in [2]; 12 V lead acid battery is being considered for the in-house battery in order to provide continuous power to the system. Another lead acid battery is used in place of the Nissan Leaf's battery to accomplish the V2H concept. When the load's energy demand exceeds the photovoltaic (PV) generated power, the lead acid battery in place of the Nissan Leaf begins to power the load until the PV's produced power meets the load's energy demand; otherwise, PV is the primary source of power. If the PV power generated is inadequate to satisfy the energy requirement of the loads, the in-house battery is used to power them. When the system reaches 30% state of charge (SOC), it enters V2H mode [3], which means the Nissan Leaf's battery is discharging in order to satisfy the load's energy demand. In V2G mode, the car disconnects from the house and supplies electricity to the grid [4, 5].

In general, the internet of things (IoT) refers to a collection of physical objects (or things) embedded in electronics, sensors, actuators and connected devices that are linked

* Corresponding author
Article's DOI: 10.5455/jjee.204-1624302314

together in a network to allow data exchange with other interconnected devices in order to provide users with more benefits and services. In other words, it turns non-smart objects into smart ones [6]. In the realm of modern information and communication technology, the phrase "Internet of Things" is now widely used. This topic has sparked several arguments since its implementation in industry and services results in more effective action, but it also presents a number of concerns, such as privacy and interoperability. Mozilla IoT was considered because of its privacy and interoperability features in its Web of Things gateway.

In this research, we propose a method that tackles this challenge by integrating Mozilla IoT with the V2H and V2G concepts.

2. COMPONENTS USED

The components that were used in the implemented system are listed Table 1.

Table 1. Components used.

Component
Raspberry Pi Model B
ESP 32
PV Panel
Current Sensor (ACS 712)
Voltage divider
Inverter
Battery
Bulbs
Relay
Connecting wires

The Raspberry Pi 4 Model B is one the latest single board computers with 4 GB RAM, ARM Cortex CPU running at 1.5 GHz with Bluetooth and Wi-Fi capabilities. It is used to set up Mozilla Things Gateway.

The ESP 32 is a microcontroller which has Bluetooth and Wi-Fi capabilities and has a dual core CPU. It can be clocked up to 240 MHz, it also has an ultra-low power processor. It comes with 512 KB SRAM and 8 MB external memory. This is best suited for the considered - streaming data to the gateway. In this case, ESP 32 collects the data from sensors and sends that to the IoT server.

The Sun force 260 W solar kit (130 W + 130 W), 12 V is the PV panel considered for the proposed test setup.

ACS 712 – 20 A current sensor module is considered for current measurement in the implemented system. The sensor is used to measure PV current, Battery current and the load current. The generated data is sent to the ESP 32 microcontroller and then sent to the gateway with the help of IoT.

The voltage divider is a common circuit that takes higher voltage and converts it into a lower one by using a pair of resistors. This was used in the implemented system to measure the voltage across various measure points (12 V to 3.3 V, since ESP 32 needs less than 3.3 V as analog input).

The Sun force 200 W, 12 V modified sine wave inverter is considered for the implemented system. The output capacity of 120 V, 200 W is implemented.

For this application, the Panasonic LC-R 127R2P 12 V, 7.2 Ah battery was selected. This battery is used in place of in-house battery and the Nissan Leaf's battery.

Blubs are used as loads in the implemented test system. In this case, the house loads, Nissan Leaf as load and grid as load.

Relays are implemented to execute the switching operation between different system modes based on the control scheme.

3. METHODOLOGY

The PV panel, in-house battery, battery charger, inverter, Nissan Leaf battery and the loads are all part of the system. Fig. 1 depicts the proposed system's overall block diagram as published in our previous paper [1]. PV power is utilized to charge the in-house battery through a charger, and then an inverter converts the battery's stored power to AC in order to power the house load. In V2H mode, a Nissan Leaf's battery is also utilized to charge the in-house battery while also powering the house loads.

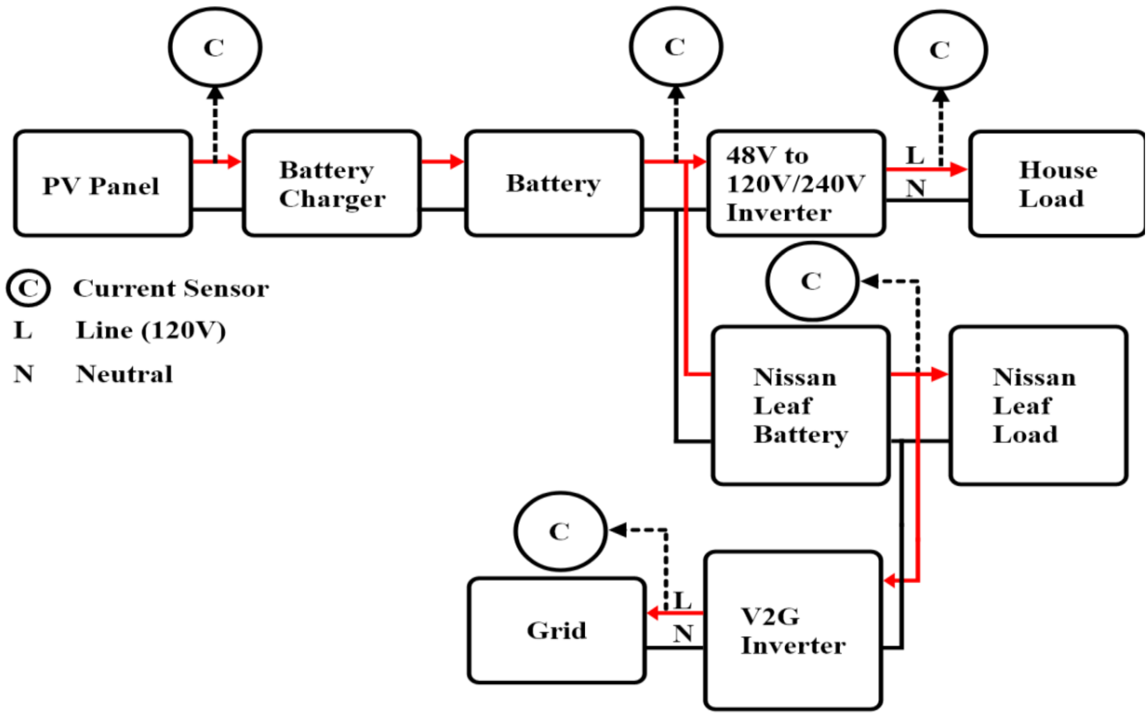


Fig. 1. Overall block diagram of the proposed system.

Fig. 2 depicts an overview of the implemented system's switching control logic. Charging and discharging the in-house battery, powering loads with the in-house battery, a 120 V inverter, V2H, overnight charging and V2G mode are all simulated in this system [7]. For the test setup, we are using only 12 V batteries to check the IoT and control logic. Furthermore, the hardware implementation was carried out incorporating eight modes representing different modes [8] of operation.

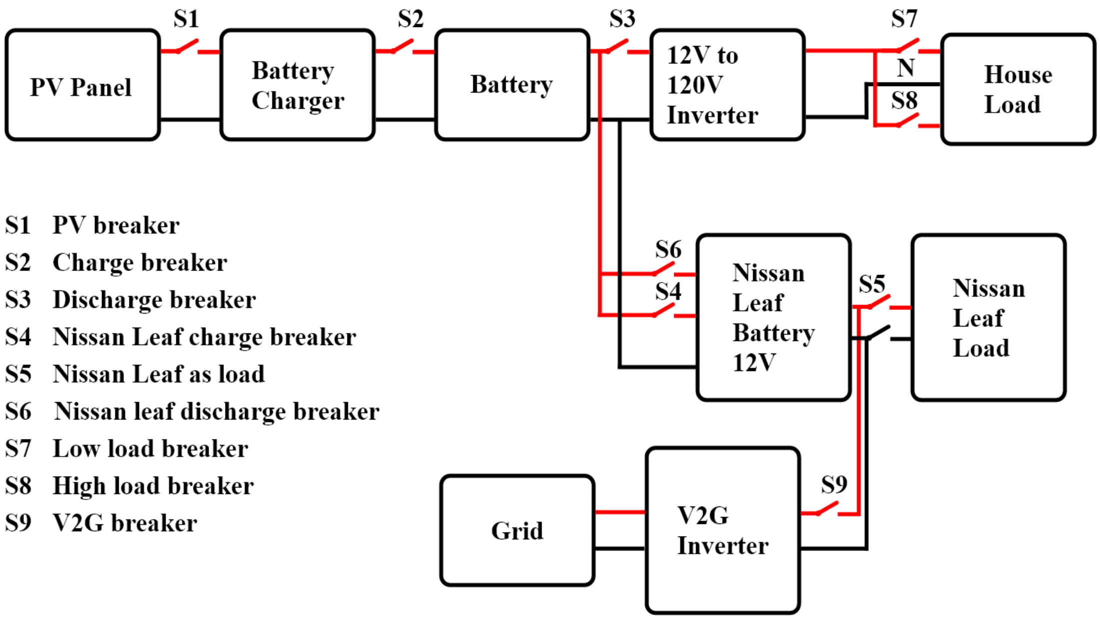


Fig. 2. Switching control logic of the implemented system.

3.1. Mode 1: In-House Battery Charging Mode

In this mode, the PV output is used to charge the in-house battery and power the loads. In this condition, the loads consume very little energy. By charging the battery, the excess energy is put to good use. Fig. 3 shows the control logic for charging the in-house battery. Mode 1 is implemented by setting switches S1, S2, S3, and S7 to HIGH, as shown in Fig. 3 and Table 2 (HIGH means break is ON).

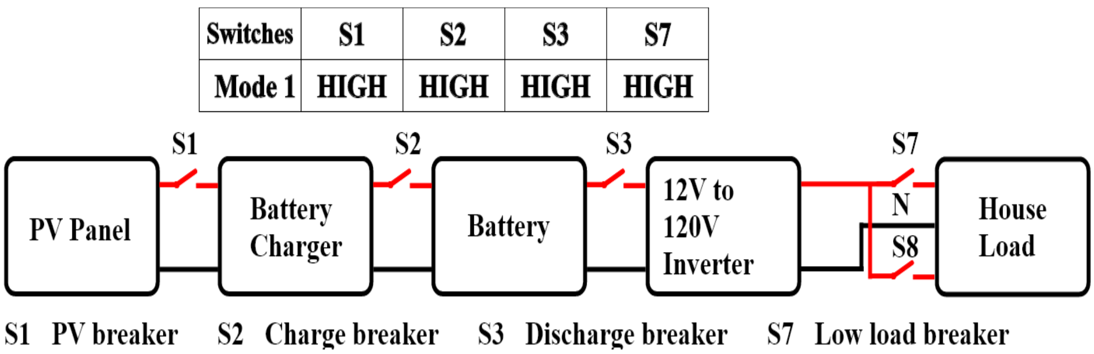


Fig. 3. Control logic for in-house battery charging.

Table 2. Switch control logic for the implemented system.

Mode	S1	S2	S3	S4	S5	S6	S7	S8	S9
Mode 1	HIGH	HIGH	HIGH	LOW	LOW	LOW	HIGH	LOW	LOW
Mode 2	LOW	LOW	HIGH	LOW	LOW	LOW	LOW	HIGH	LOW
Mode 3	HIGH	HIGH	HIGH	HIGH	LOW	LOW	HIGH	LOW	LOW
Mode 4	-	-	-	-	HIGH	-	-	-	-
Mode 5	LOW	LOW	HIGH	LOW	LOW	HIGH	HIGH	LOW	LOW
Mode 6	-	HIGH	HIGH	-	-	-	-	-	LOW
Mode 7	LOW	LOW	HIGH	HIGH	LOW	LOW	HIGH	LOW	LOW
Mode 8	-	-	-	-	-	-	-	-	HIGH

3.2. Mode 2: In-House Battery Discharge Mode

In this mode, PV production is low compared to the load's energy demand. In this case, the load is powered by the in-house battery. Fig. 4 shows the control logic for the in-house battery in discharge mode. An inverter with output voltages of 120 V is also utilized. Mode 2 is implemented by setting switches S3 and S8 to HIGH, as shown in Fig. 4 and Table 2.

Switches	S3	S8
Mode 2	HIGH	HIGH

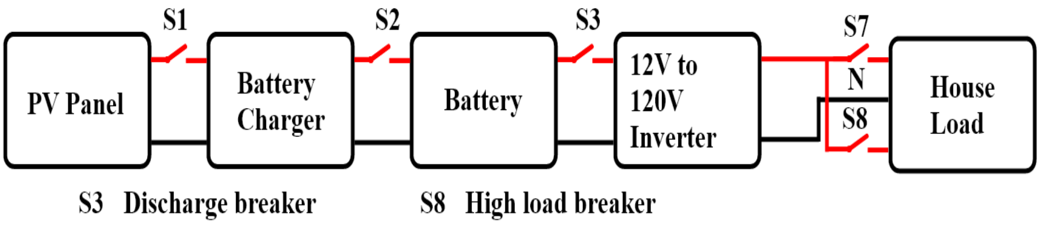


Fig. 4. Control logic for the in-house battery charging.

3.3. Mode 3: Nissan Leaf Charge Mode

The device will charge the Nissan Leaf's batteries when the PV output exceeds the load's energy requirement and the in-house batteries SOC surpass 60% [9, 10]. Fig. 5 shows the control logic for the Nissan Leaf charging. Mode 3 is implemented by setting switches S1, S2, S3, S4 and S7 to HIGH, as shown in Fig. 5 and Table 2.

Switches	S1	S2	S3	S4	S7
Mode 3	HIGH	HIGH	HIGH	HIGH	HIGH

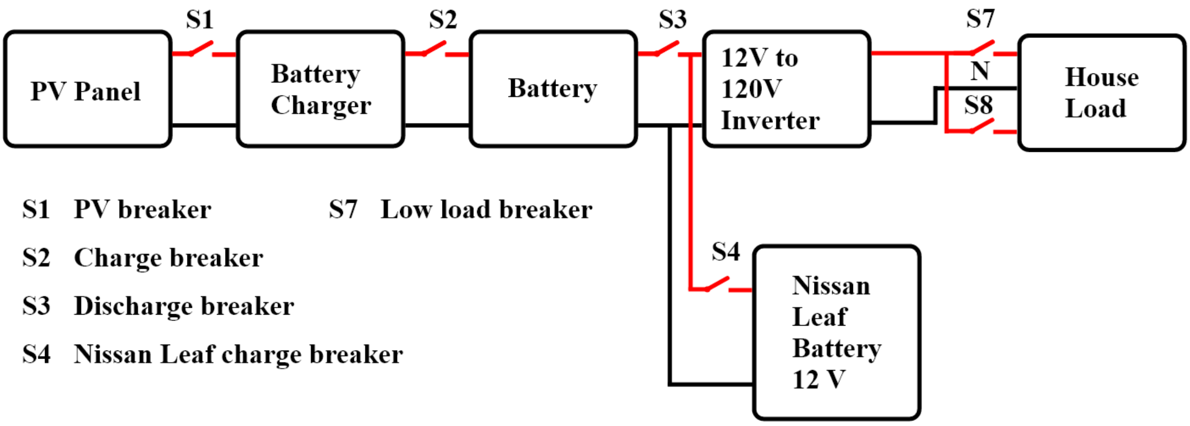


Fig. 5. Control logic for the Nissan Leaf charging.

3.4. Mode 4: Nissan Leaf as Load

The Nissan Leaf is considered as a load in this mode. Travel-related energy is estimated to account for 25% of the Nissan Leaf's battery depletion. The control logic for the Nissan Leaf as a load is shown in Fig. 6. As shown in Fig. 6 and Table 2, only switch S5 is set to HIGH to implement mode 4.

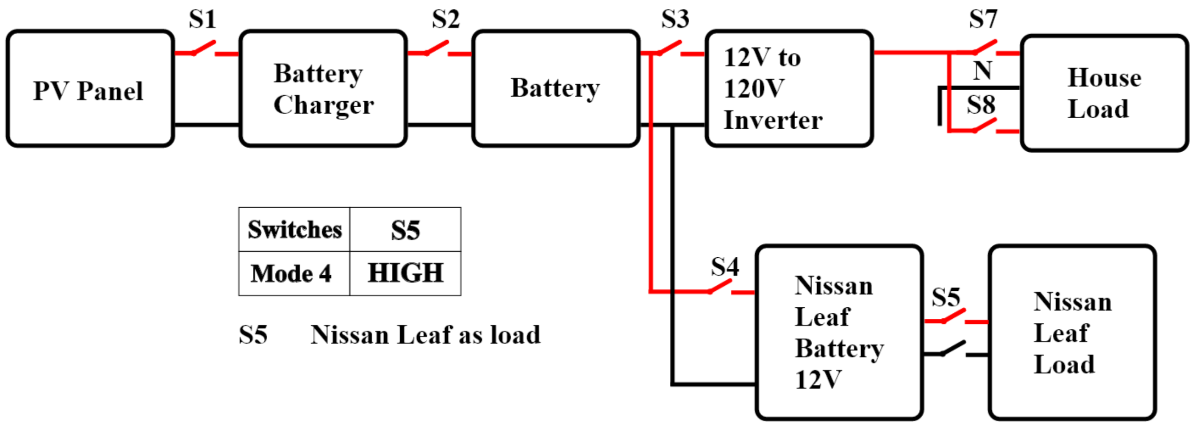


Fig. 6. Control logic for the Nissan Leaf as load.

3.5. Mode 5: Nissan Leaf Discharge Mode

Nissan Leaf reels out stored energy to the home to fulfill the load's energy demand when the PV output is less than the load's energy demand and the in-house batteries' SOC is less than 30% [8, 11]. Fig. 7 shows the control logic for the Nissan Leaf in discharge mode. Mode 5 is implemented by setting switches S3, S6, and S7 to HIGH, as shown in Fig. 7 and Table 2.

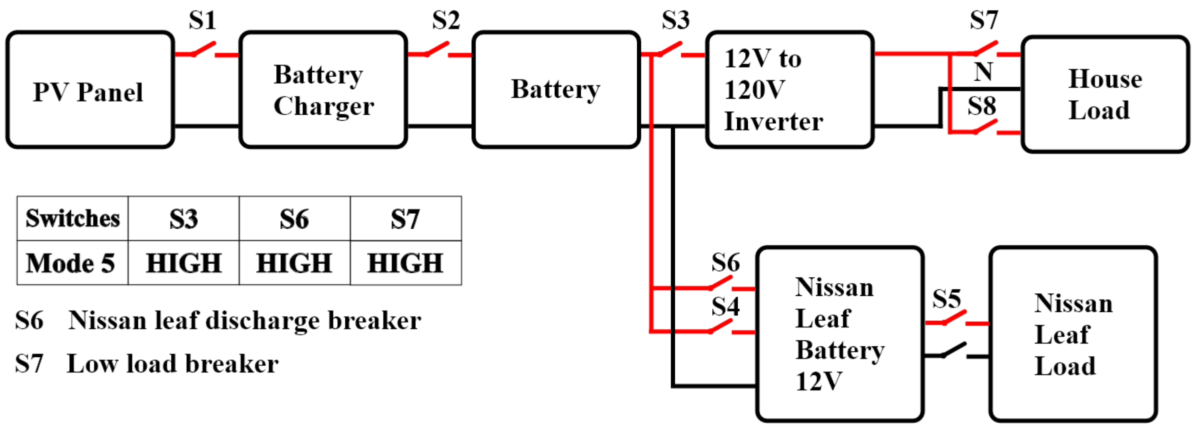


Fig. 7. Control logic for the Nissan Leaf discharge mode.

3.6. Mode 6: In-House Battery Protection Mode

This option is added to help save the in-house battery by tracking its SOC. The discharge breaker and the charge breaker will both go HIGH to charge the battery when the SOC of the in-house batteries falls below 30%. Mode 6 is implemented by setting switches S2 and S3 to HIGH, as shown in Table 2.

3.7. Mode 7: Nighttime Charging Mode

The energy stored in the in-house battery is utilized to charge the Nissan Leaf's battery at night, when the load's energy demand is lower and the in-house battery's SOC is more than 60% [12]. Fig. 8 shows the control logic for the evening charging mode. Mode 7 is implemented - by setting switches S3, S4 and S7 to HIGH - as shown in Fig. 8 and Table 2.

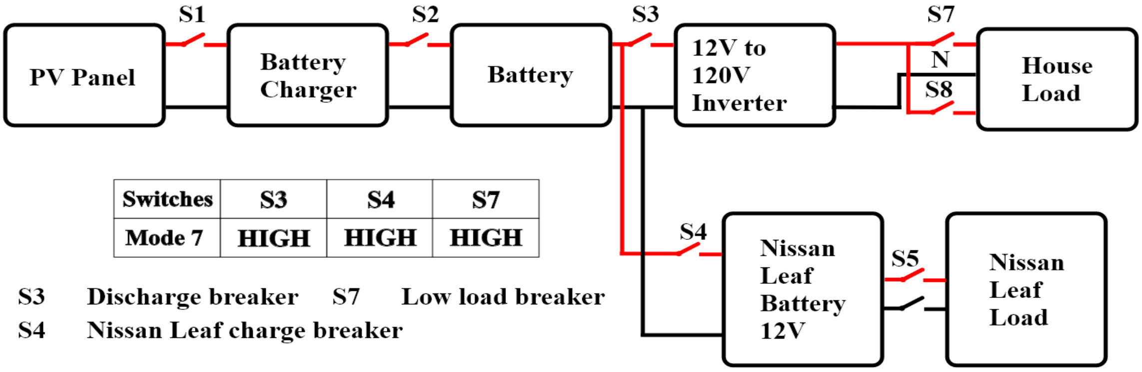


Fig. 8. Control logic for the nighttime charging mode.

3.8. Mode 8: V2G Mode

In this mode, the car disconnects from the house and connects to the grid [13]. The control logic for the V2G mode is shown in Fig. 9. As seen in Fig. 9 and Table 2, switch S9 is HIGH to implement mode 9.

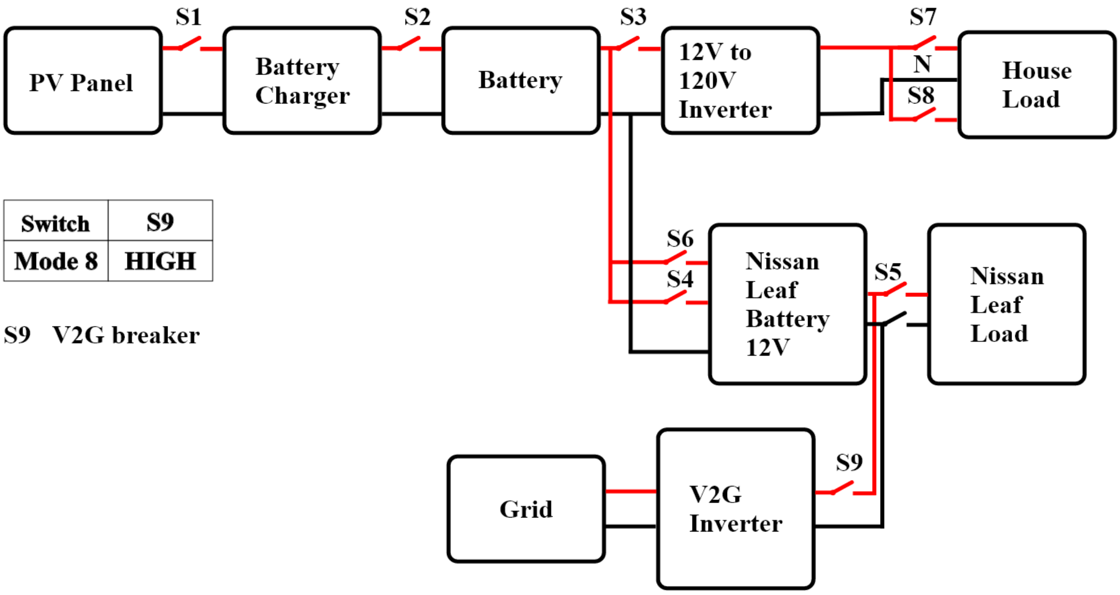


Fig. 9. Control logic for the V2G mode.

4. HARDWARE IMPLEMENTATION

The implemented system comprises ESP 32 microcontroller boards, Raspberry Pi, current sensor, voltage dividers, relays, PV panels, inverters and loads. The experimental setup is a 12 V based system. It was executed to demonstrate the IoT and control concepts. IoT is needed for control and data logging, and it functions as a basic SCADA system. The actual system has different voltage levels and involves multiple DC-DC converters. Fig. 10 illustrates the block diagram of the implemented system. The IoT part is handled by Mozilla IoT, and ESP 32 microcontroller is used for handling and updating the data to the gateway and also to control and monitor remotely using the dashboard. The real time outputs from the test setup are logged and can be seen as graphs. The research focused on the IoT part, where the concept of control and remote monitoring is established through IoT. Furthermore, the test results are plotted as graphs.

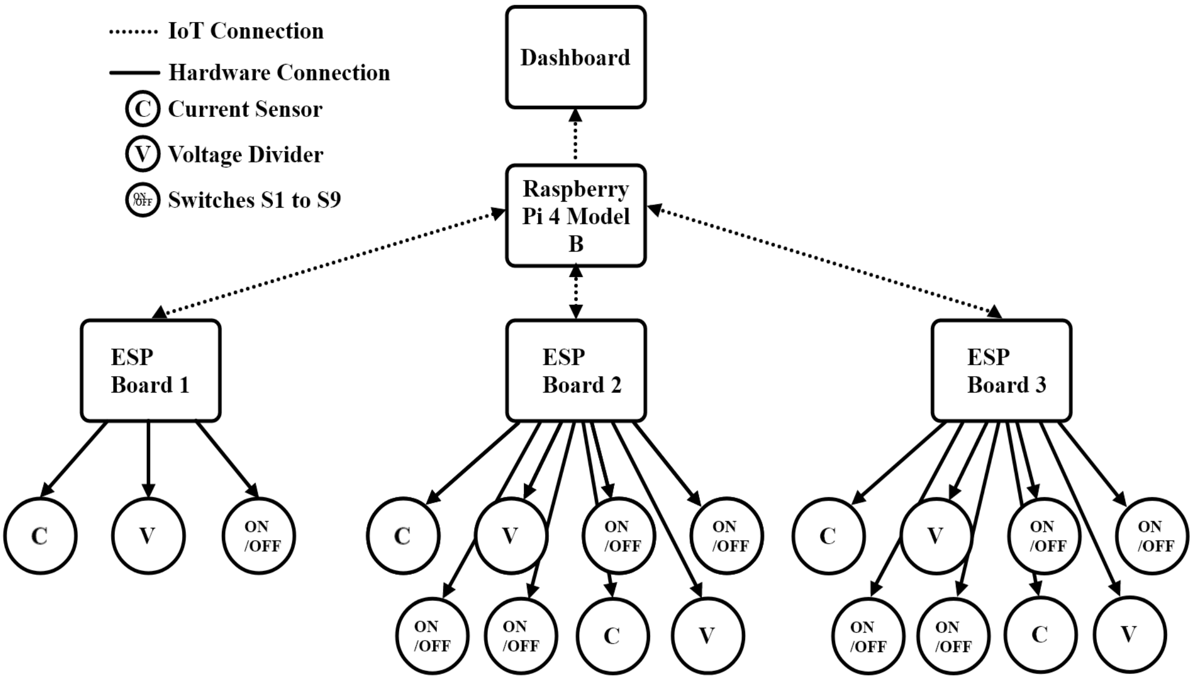


Fig. 10. Block diagram of the implemented system.

4.1. Mozilla IoT

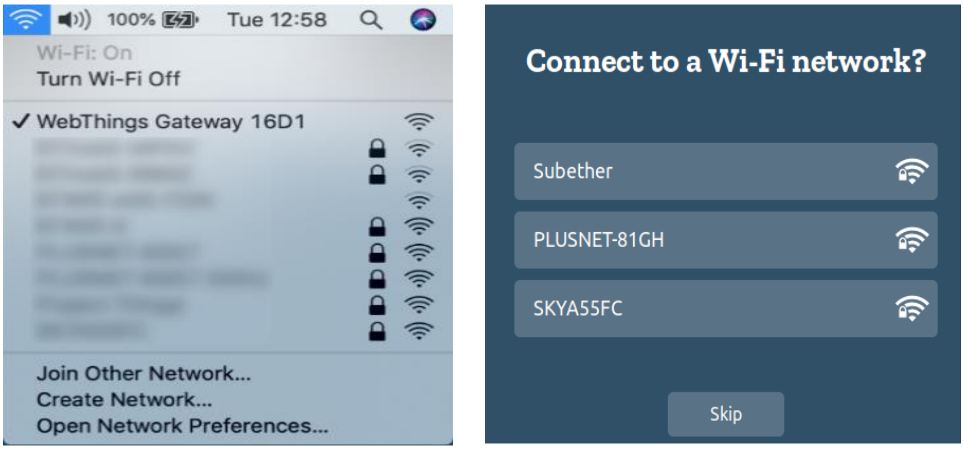
Because of its security, privacy, interoperability and overall value, Mozilla IoT was chosen for this study. In most IoT applications, the device sends data to the cloud, which is then monitored and controlled by a dashboard or a mobile application. However, in this scenario, the full web of things interoperability is carried out locally, with data shared with the cloud through a JWT or oAuth framework. The web of things aims to create a decentralized IoT by assigning URLs to objects on the internet, allowing them to be linked, discovered and interoperable [14].

4.1.1. Getting Started with Mozilla IoT – Webthings Gateway on Raspberry Pi

Webthings gateway is a smart home gateway software distribution that allows users to manage and monitor their devices directly over the internet, without the middleman. The gateway image is obtained from the Mozilla IoT website and flashed onto a microSD card with the downloaded image file. The microSD card is put into the microSD slot after flashing successfully, and the Raspberry Pi is booted.

4.1.2. Connecting WebThings Gateway to Wi-Fi

After successfully booting, the Raspberry Pi will function as a gateway, connecting to the internet and communicating with all devices through Wi-Fi. As illustrated in Fig. 11(a), the gateway generates a hotspot called "WebThings Gateway XXXX," with the XXXX being the four numbers from the Raspberry Pi's MAC address. Scanning and connecting to the network can be done with a computer or a mobile phone. As illustrated in Fig. 11(b), when you join, it creates a captive gateway page that displays neighboring Wi-Fi networks. When asked, select the preferred network and input the password. This takes you to the "Connecting to Wi-Fi..." page.



(a) (b)

Fig. 11. a) Gateway hotspot on network and internet setting; b) Gateway's connect to a Wi-Fi network page.

4.1.3. WebThings Gateway - Choosing Subdomain

After connecting the Raspberry Pi to Wi-Fi, make sure your computer or phone is connected to the same network and go to "http://gateway.local" in your browser. It takes you to a welcome screen, as shown in Fig. 12, where you may register a free domain to safely access the gateway via the internet using Mozilla's https tunneling, fill in the data, and click Create. This generates your subdomain, which you can load into "https://SUBDOMAIN.mozilla-iot.org" using a computer or a mobile phone.

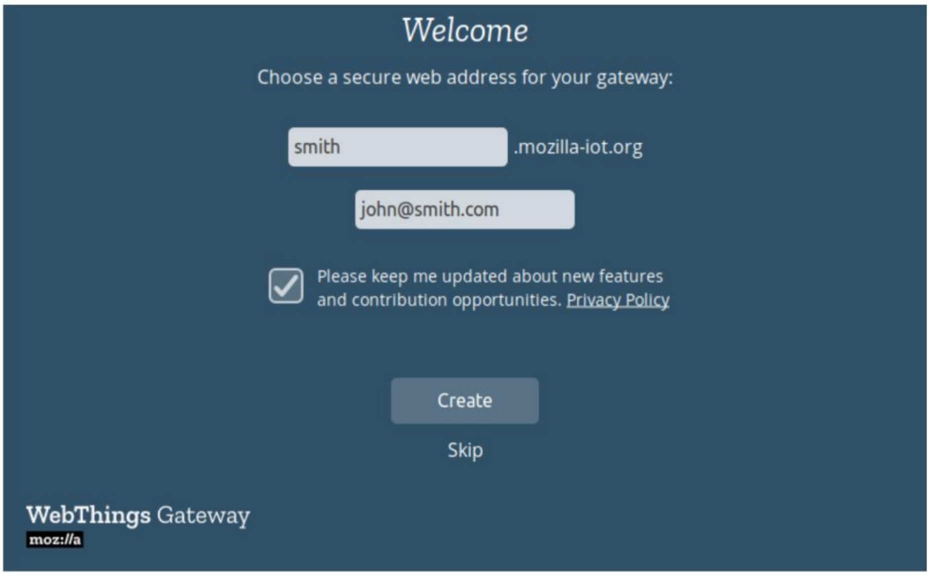
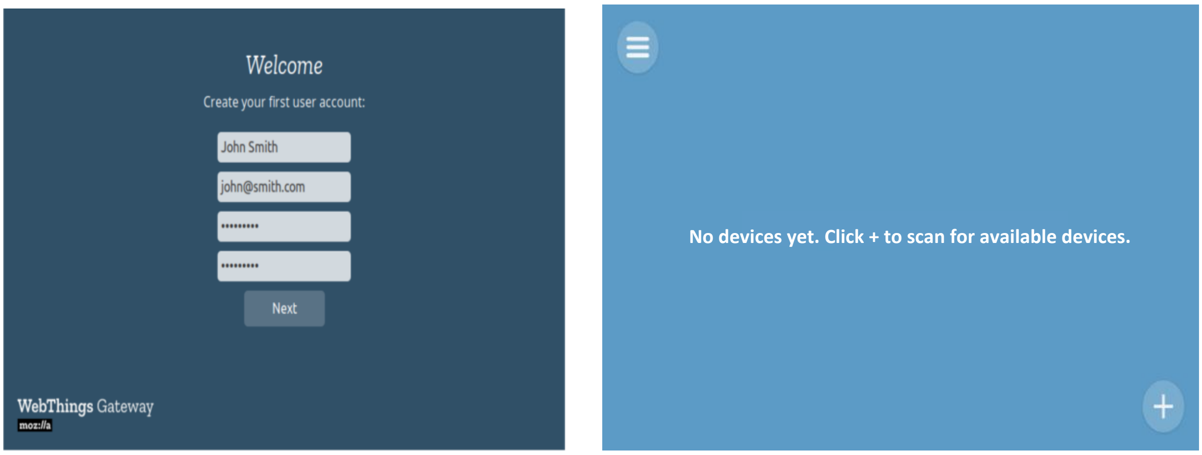


Fig. 12. Domain registration page.

4.1.4. WebThings Gateway – Creating a User Account

After completing the subdomain registration procedure, the user is redirected to the next stage of the setup process, which requires him to provide his name, email address and password (see Fig. 13(a)). This action establishes a user account in the gateway, which may be used to find, add, monitor and manage all connected devices. After successfully creating an account, the gateway sends the user to an empty Things screen where he may begin adding devices, as seen in Fig. 13(b).



(a) (b)

Fig. 13. Screen of WebThings Gateway: a) Account creation process; b) landing page.

4.1.5. WebThings Gateway – Programming Things

The microcontroller is an ESP 32 board, and programming is done using the Arduino programming language. The Webthings and Arduino JSON libraries are installed, and the code for the chosen use case is executed. The implemented system was executed using three ESP 32 microcontroller boards for PV measurement, battery measurement and V2G, and V2H measurements, as shown in Fig. 14. Fig. 14 depicts the implemented system lab setup, and Fig. 15 depicts the overall GUI for the WebThing dashboard. Six sets of PV panels are installed on the roof and connections are available in the lab for experiments.

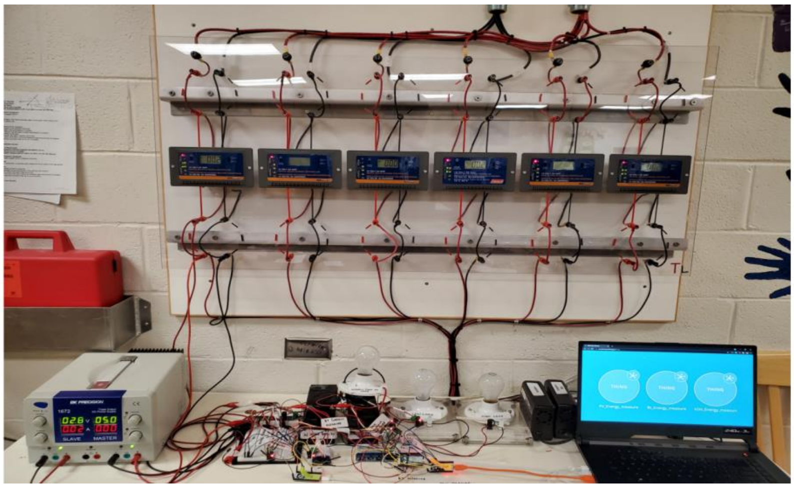


Fig. 14. Lab setup of the implemented system.

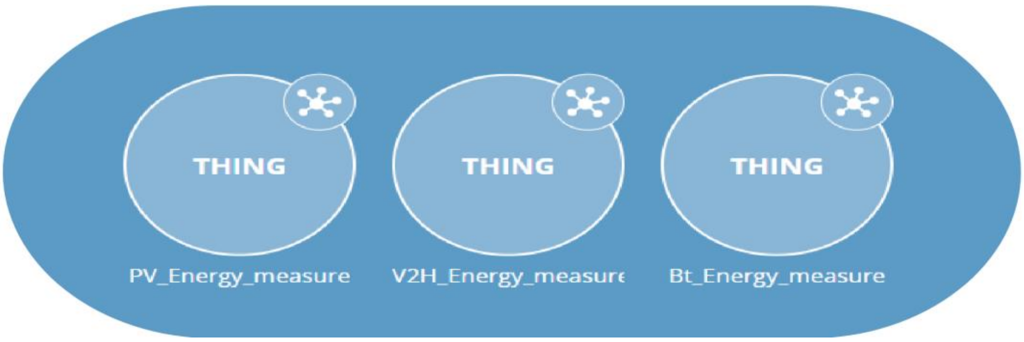


Fig. 15. GUI for Web Things gateway.

4.2. The Implemented System

As seen in Fig. 14, the entire system is powered by 12 V. The experimental system is being used to illustrate the concept of IoT, monitoring, and control. The following modes were simulated, and the results are displayed as graphs [15, 16]. All graphs show data for one minute, and there was no notable change in solar resource or battery voltage during that time. As a result, the plots are mostly straight lines. The graphs are generated in real time using data from the test setup that is sent to the gateway.

4.2.1. In-House Battery Charge Mode

PV serves as the primary source of energy in this mode [17-19], powering the load and charging the battery. The PV's overall GUI is exhibited in Fig. 16. Switch S1 is HIGH, as can be seen in Fig. 16, and 13 V is generated. The in-house battery is charged with 1 A of electricity. The recorded data of PV voltage and current is shown as a graph in Fig. 17. The PV panel produced 13 V and the battery was charged with 1 A of current. The voltage and current graph for the in-house batteries is shown in Fig. 18. The battery was being charged at 13 V and 1 A. Figs. 17 and 18 illustrate the test output for the in-house battery in charging mode where the output from PV was used for charging the in-house battery. This was achieved by switching ON the relays S1, S2, S3 and S7.

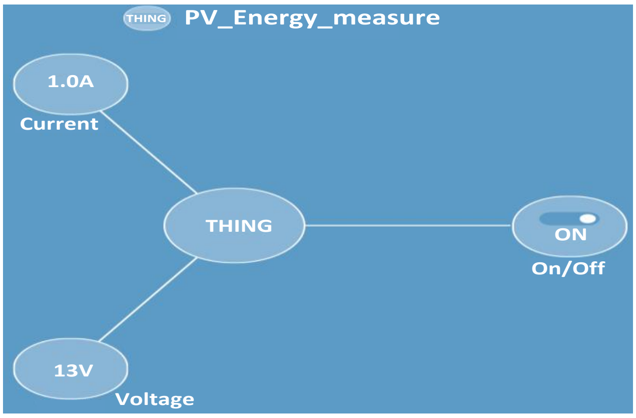


Fig. 16. Overall GUI for PV energy measure.

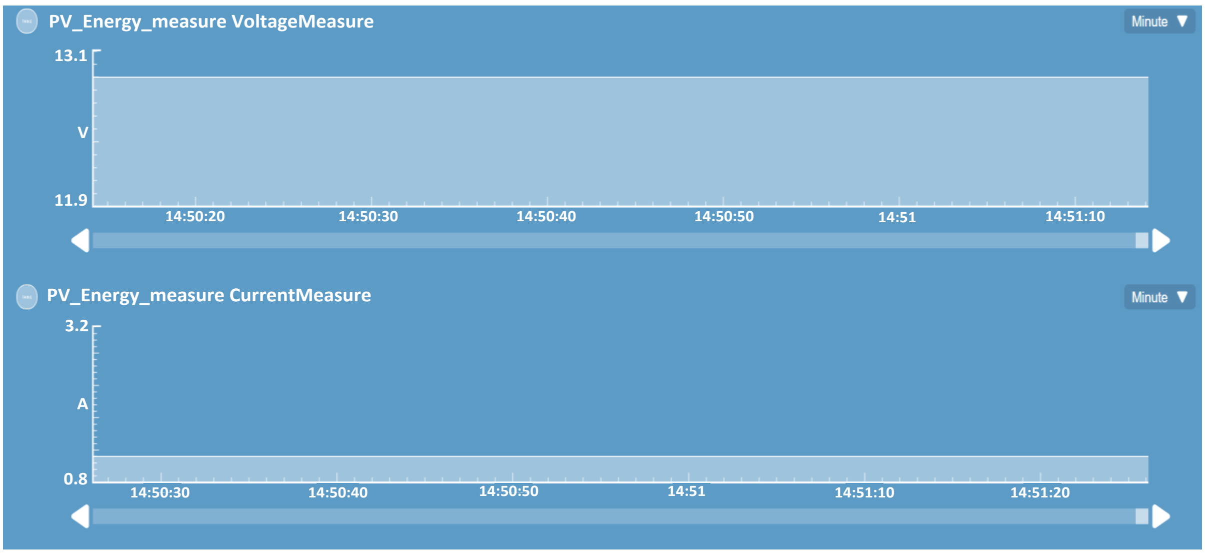


Fig. 17. The PV's measured voltage and current.

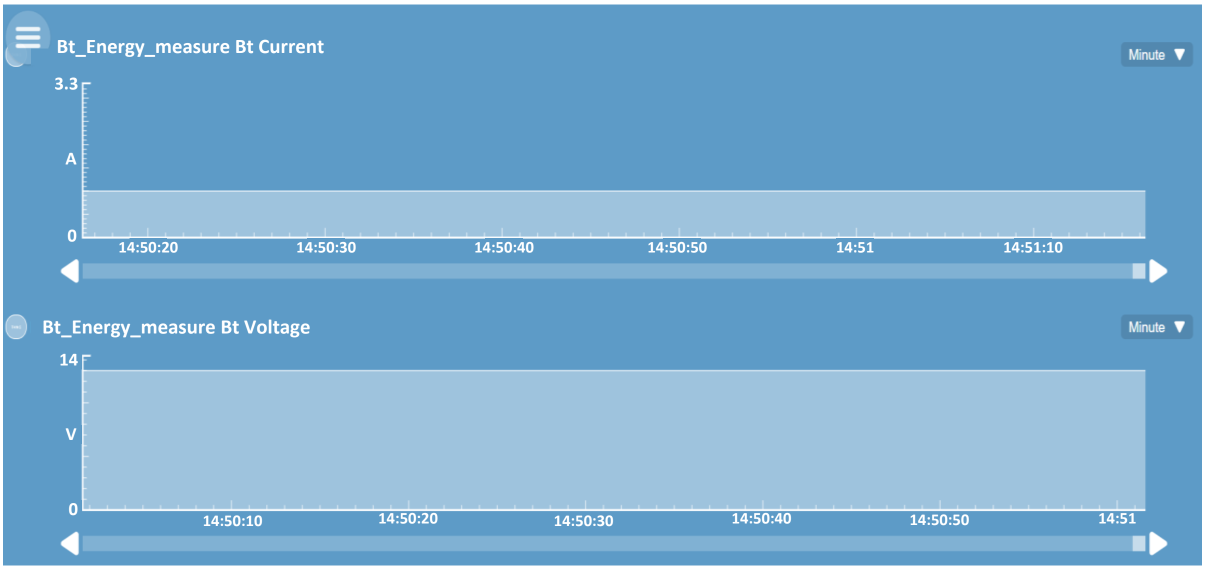


Fig. 18. The measured voltage and current for the in-house battery.

4.2.2. In-House Battery Discharge Mode

In this mode, the battery acts as the source of power that powers the load. Fig. 19 depicts the overall user interface for the in-house batteries. As shown in Fig. 19, switches S3 and S8 are HIGH and the battery is at a voltage of 13 V and discharging at 3 A to satisfy the energy requirement of the house load with the help of the inverter. The load is 120 V and draws 1.3 A; in this situation, the load is a light bulb. The in-house battery is draining at 13 V, 3 A, as shown in Fig. 20, while the load is running at 120 V, 1.3 A, as shown in Fig. 21. The experimental setup for an in-house battery in discharge mode is shown in Fig. 22. Figs. 20 and 21 illustrate the test output for the in-house battery in discharge mode. It can be seen from the graph that the in-house battery was discharging to power the load. This was achieved by switching ON the relays S3 and S8. Fig. 23 illustrates the in-house batteries SOC graph; it can be observed that the in-house battery was on discharge cycle between the time periods of 5 s and 15 s before starting to charge. This was simulated in MATLAB and details are provided in [1].

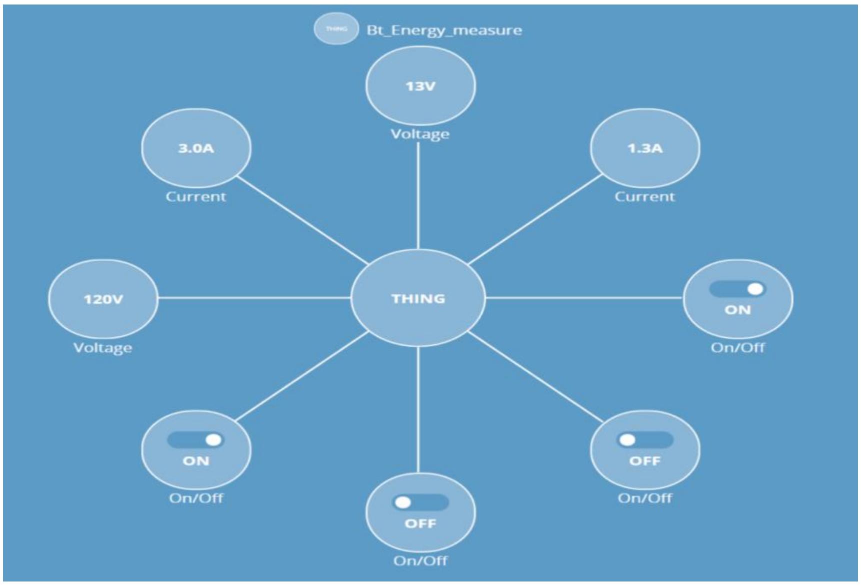


Fig. 19. Overall GUI for the in-house battery.

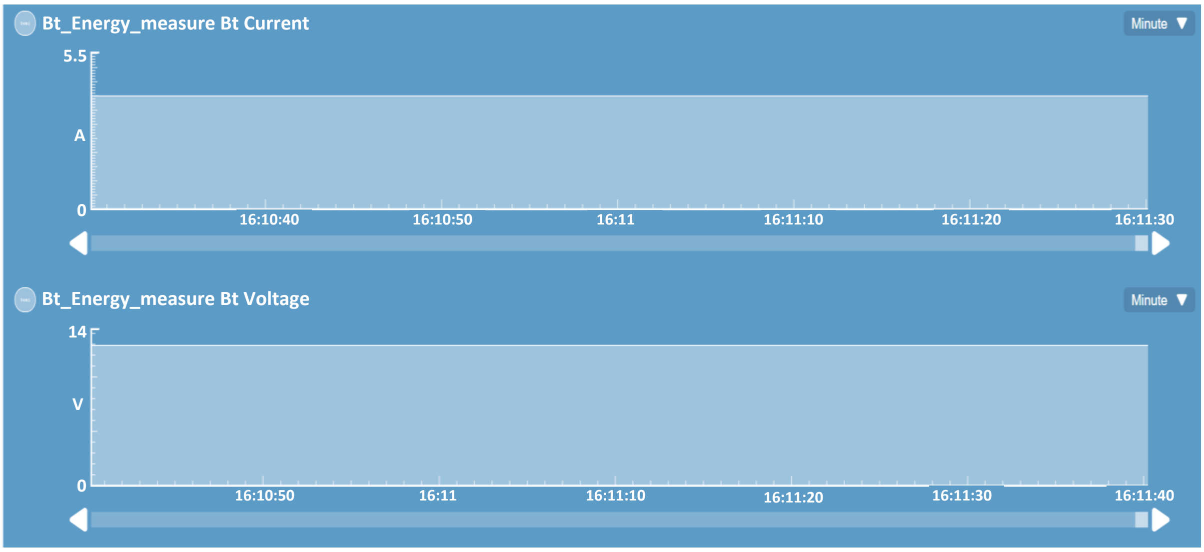


Fig. 20. The measured voltage and current of the in-house battery.

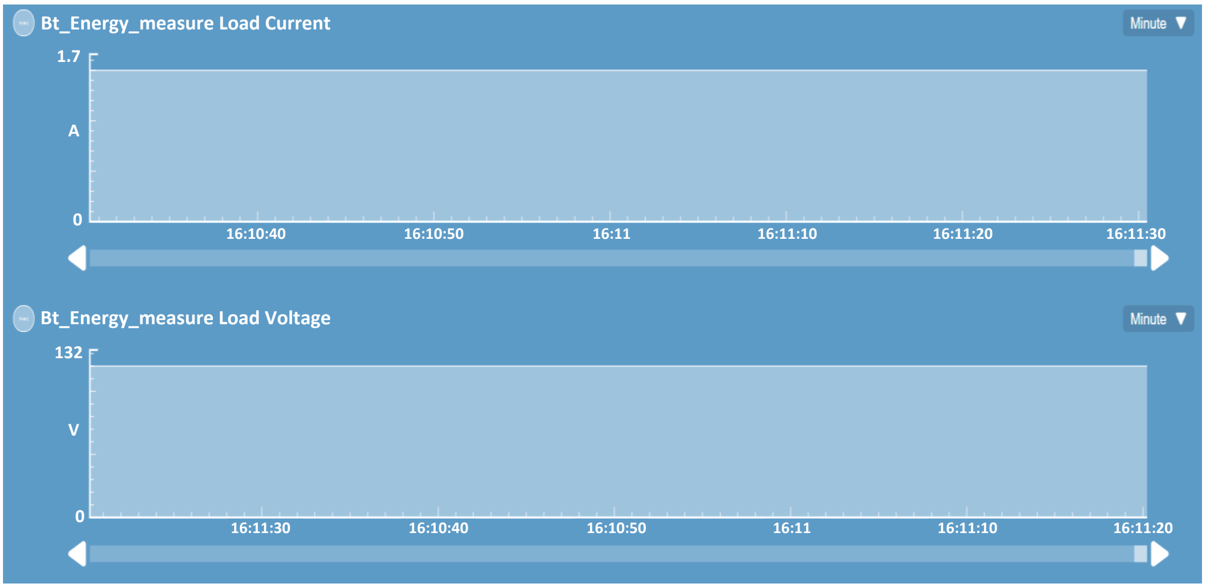


Fig. 21. The measured voltage and current of the load.

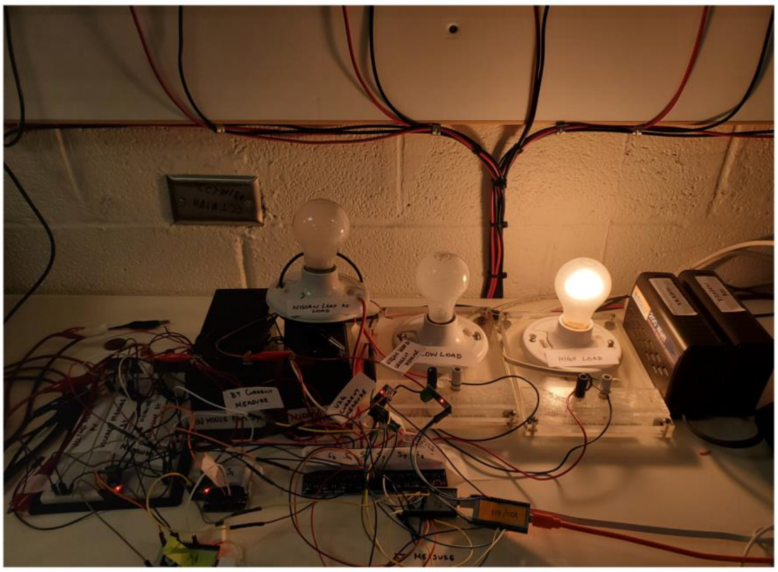


Fig. 22. Setup for measuring the load’s voltage and current.

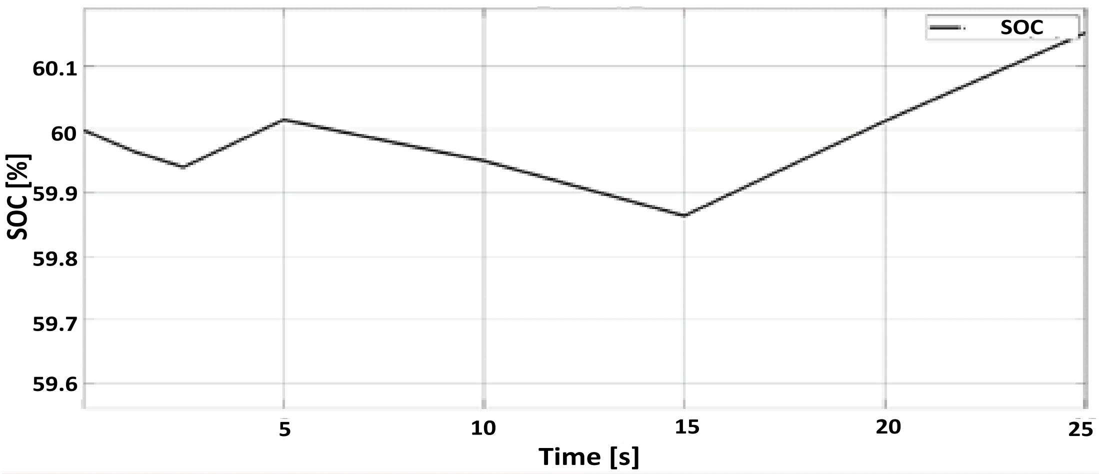


Fig. 23. SOC of the in-house battery.

4.2.3. Nissan Leaf in Charging Mode

In this mode, the remaining energy after satisfying the load's energy demand is utilized to charge the Nissan Leaf's 12 V lead acid battery [20]. PV is providing 13 V and simultaneously charging the battery and powering the load at 2 A; thus switches S1, S2, S3, S4, and S7 are HIGH. In this example, the load consumes 120 V at 1 A. The Nissan Leaf battery is also charged at 1 A and 13 V. Fig. 24 shows the overall PV GUI in (a), the overall in-house battery GUI in (b) and the overall V2H/V2G GUI in (c). From Fig. 25, it can be observed that the PV produces 13 V, and that 2 A of electricity is used to power the load and charge the in-house battery. From Fig. 26, the in-house battery is shown to be at 13 V, with 2 A of current going to the load and charging the in-house battery. The voltage and current measurement graph for the load is shown in Fig. 27. From Fig. 28, the Nissan Leaf battery is charging at 13 V and 1 A, as can be observed. Figs. 25, 26, 27 and 28 illustrate the test output for Nissan Leaf in charge mode. It can be seen from the graph that the output from PV is used for charging the in-house battery and simultaneously charging Nissan Leaf's battery. This was achieved by switching ON the relays S1, S2, S3 and S7. Fig. 29 illustrates the output from level 2 charging which can be observed between the time intervals of 5 s and 10 s, while the output from level 1 charging can be seen between the time intervals of 10 s and 15 s. This was simulated in MATLAB and details are provided in [1].

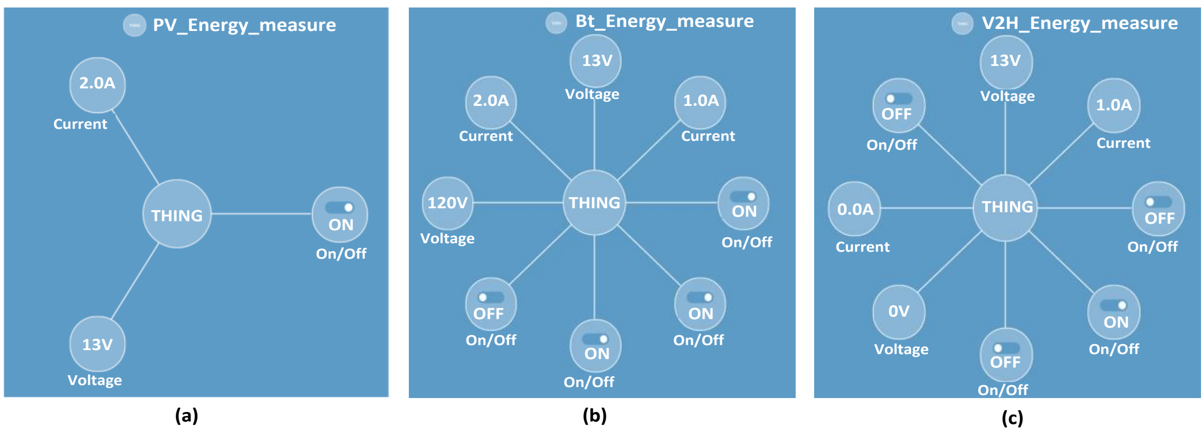


Fig. 24. Overall GUI for: a) PV; b) battery; c) V2H/V2G.

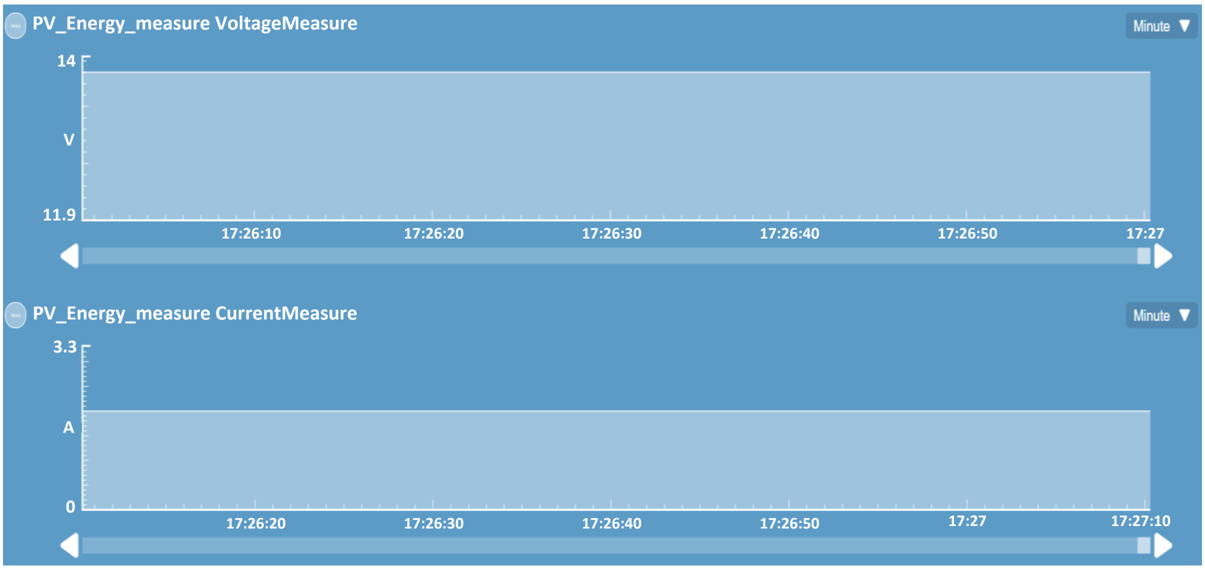


Fig. 25. The PV's measured voltage and current.

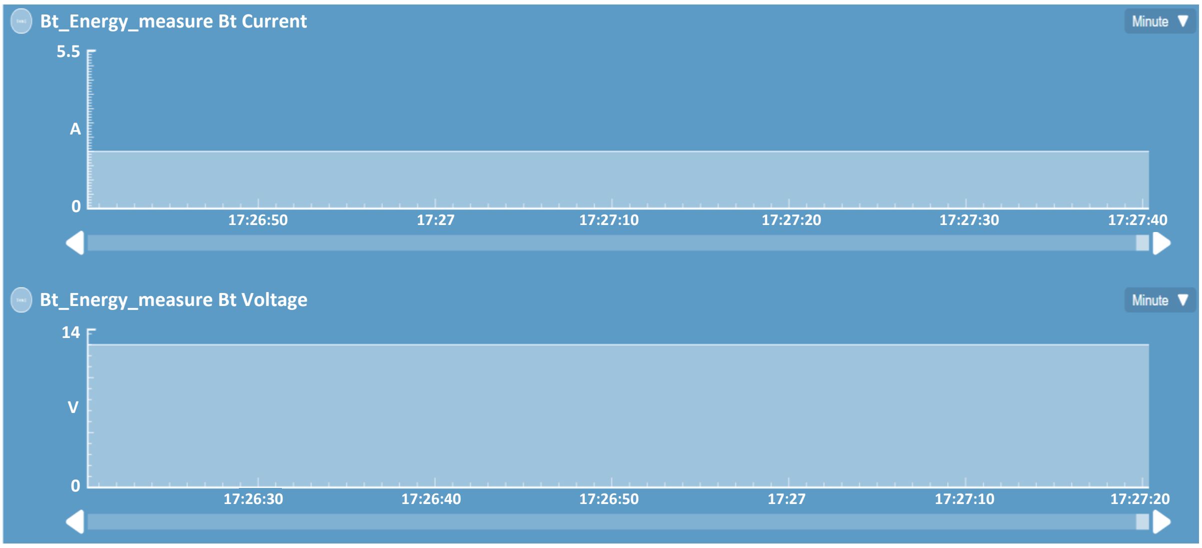


Fig. 26. The measured voltage and current for the in-house battery.

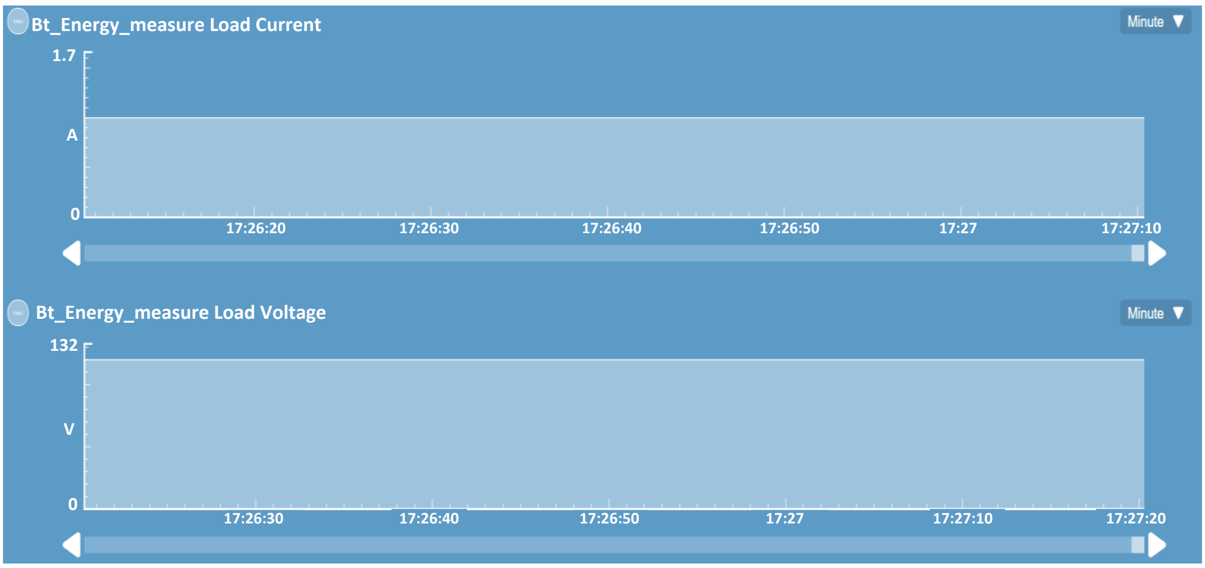


Fig. 27. The measured voltage and current for the load.

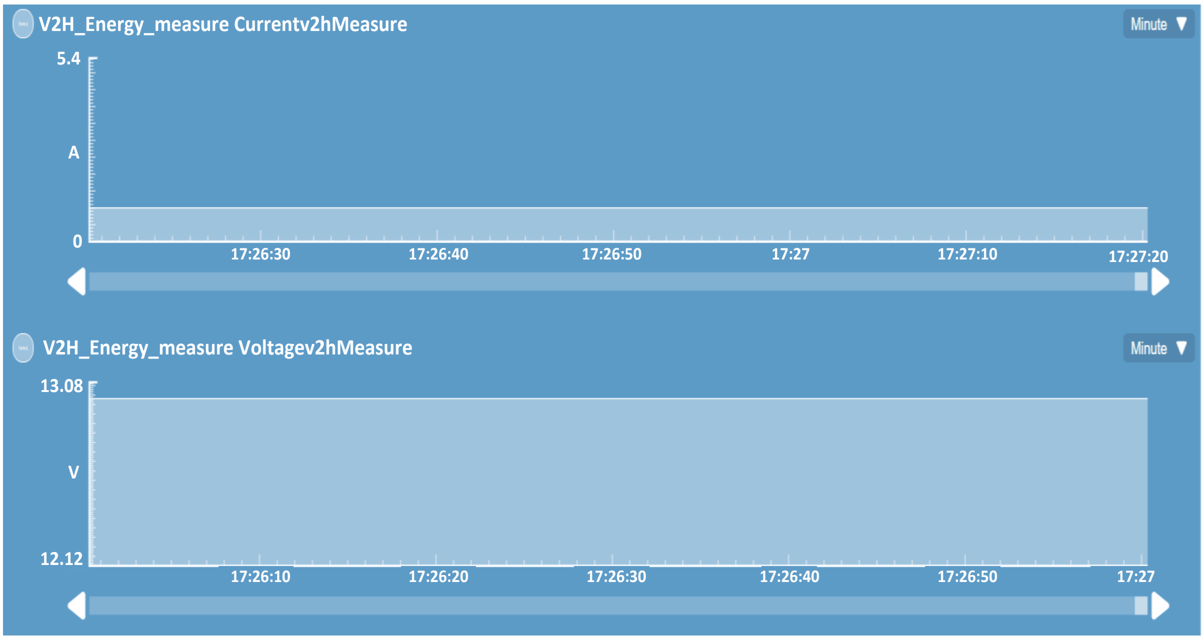


Fig. 28. The measured voltage and current for the Nissan Leaf.

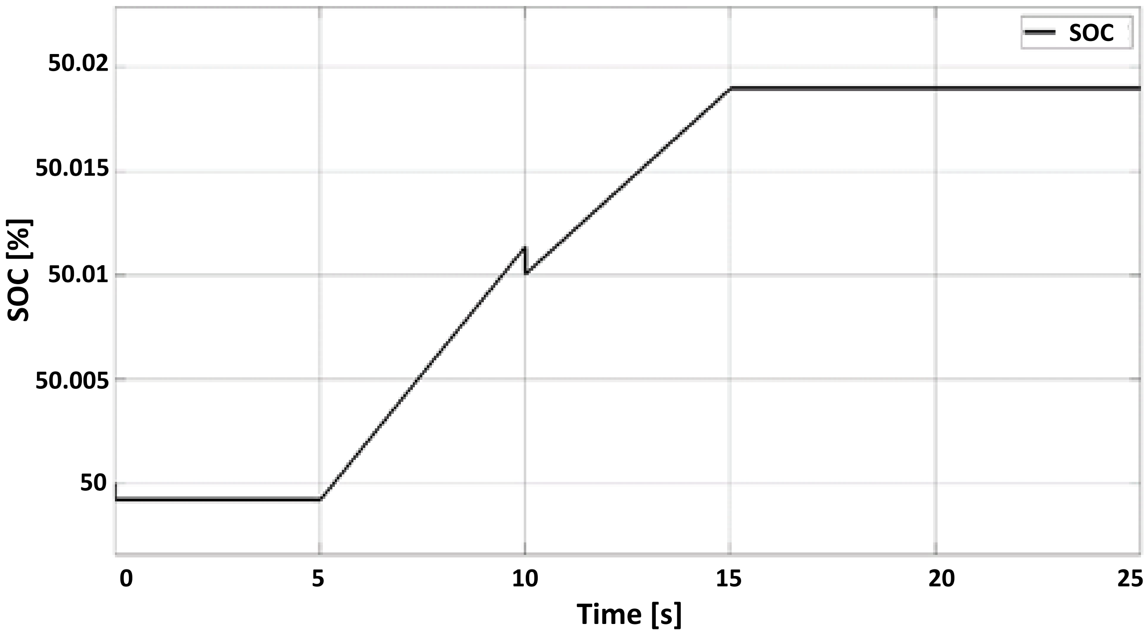


Fig. 29. SOC of the Nissan Leaf.

4.2.4. Nissan Leaf as Load

In this mode, the Nissan leaf serves as a load, and a DC light bulb is used to imitate the Nissan leaf as a load. The entire V2H/V2G GUI is exhibited in Fig. 30, which shows that Switch S5 is HIGH in this mode. The Nissan leaf battery is draining at a rate of 13 V and 5 A (see Fig. 29.) Fig. 31 illustrates the test output for Nissan Leaf as load, i.e. the measured voltage and current of the Nissan leaf battery. This figure reveals that the Nissan Leaf's battery was discharging to simulate the usage of EV. This was achieved by switching ON the relay S5. Fig. 32 depicts the experimental setup for demonstrating the working on the Nissan leaf as a load.

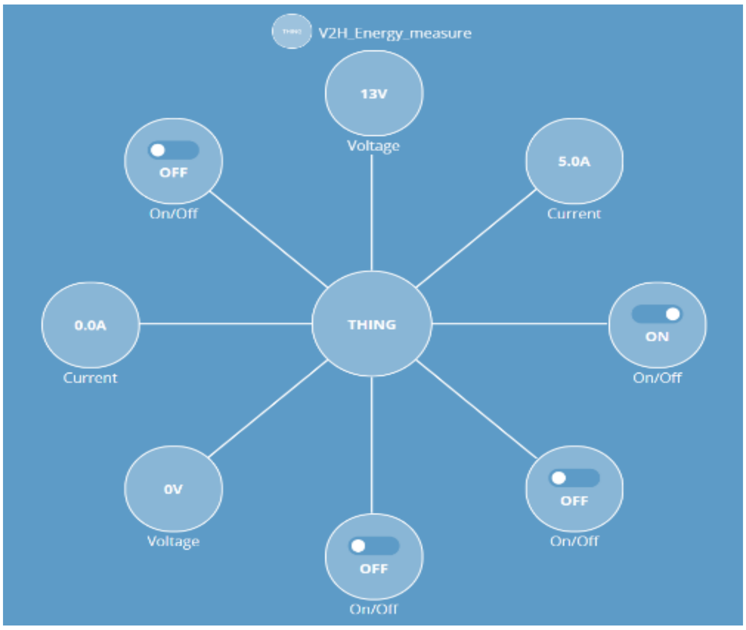


Fig. 30. Overall V2H/V2G GUI.

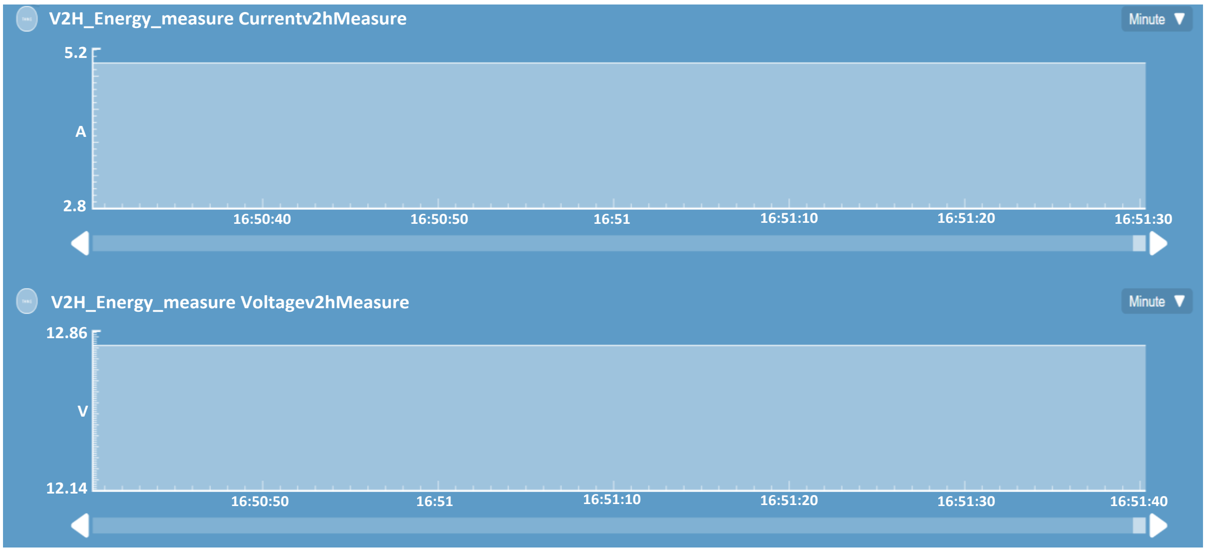


Fig. 31. The measured voltage and current for the Nissan Leaf battery.

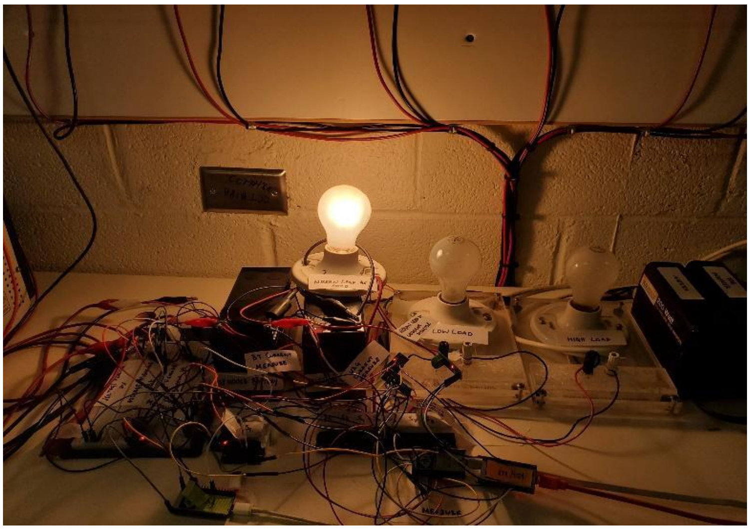
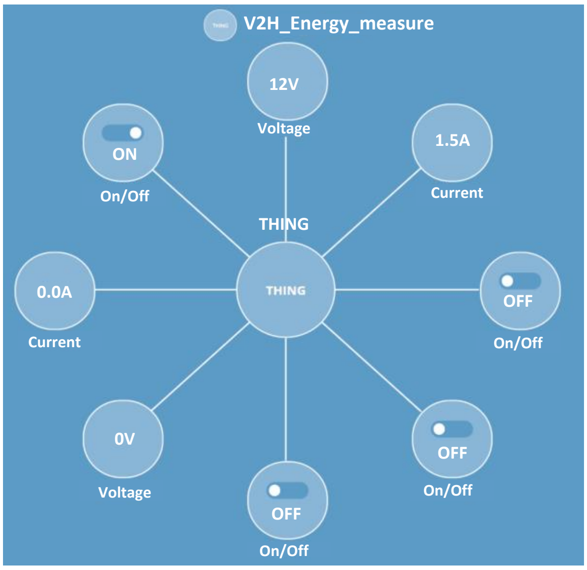
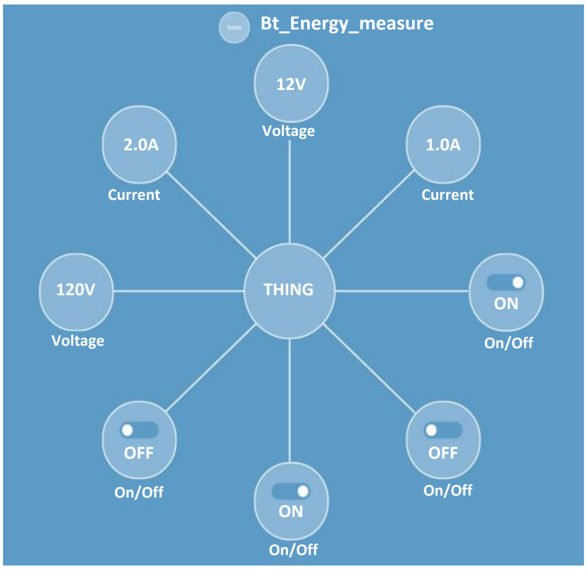


Fig. 32. Experimental setup for the Nissan Leaf as load.

4.2.5. Nissan Leaf in Discharge V2H Mode

When the power of the in-house batteries is low and the energy demand of the loads is high, this mode is used. Nissan Leaf reels the stored power to fulfill the load's energy requirement while also charging the in-house battery. Fig. 33 shows overall V2H/V2G GUI in (a) and overall in-house battery GUI in (b). It also shows that switches S6, S3, and S7 are HIGH. The Nissan Leaf battery is also discharged at 12 V and 1.5 A to power the load and charge the in-house battery, as shown in Fig. 33(a). The measured voltage and current for the in-house battery is shown in Fig. 34. The load is operating at 120 V and 1 A, as shown in Fig. 35. The voltage and current measurement graph for Nissan Leaf is shown in Fig. 36. Fig. 37 depicts the experimental setup for the V2H mode. Figs. 34, 35 and 36 illustrate the test output for Nissan Leaf in discharge mode. It can be seen from the graphs that the Nissan Leaf's battery was discharging to power the load and charge the in-house battery simultaneously. This was achieved by switching ON the relays S3, S6, and S7.





(a) (b)

Fig. 33. Overall GUI for: a) V2H/V2G; b) the in-house battery.

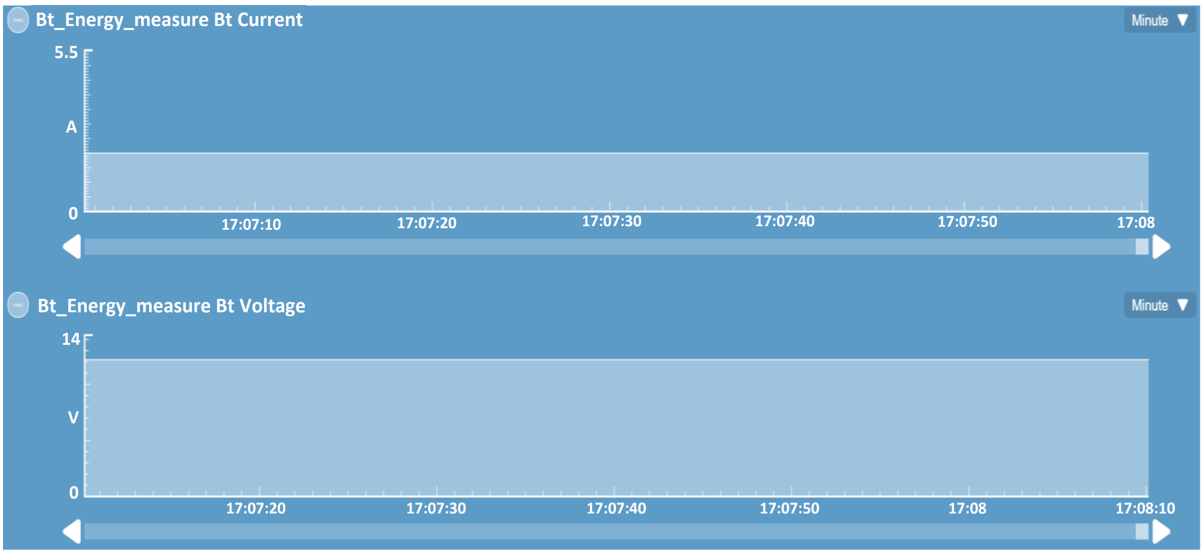


Fig. 34. The measured voltage and current for the in-house battery.

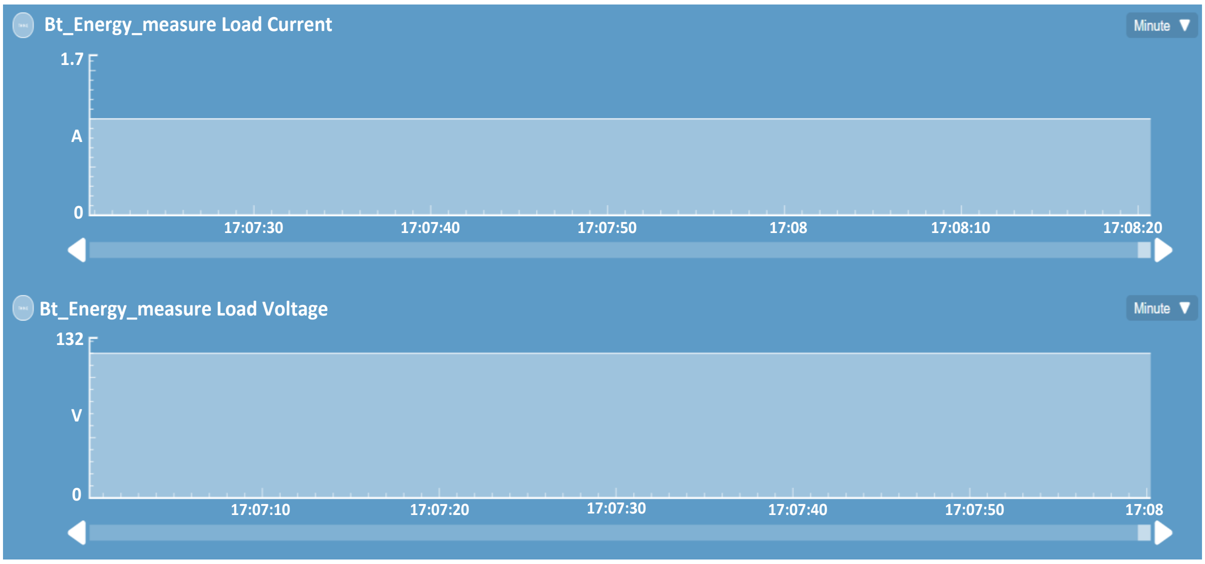


Fig. 35. Measured voltage and current for the load.

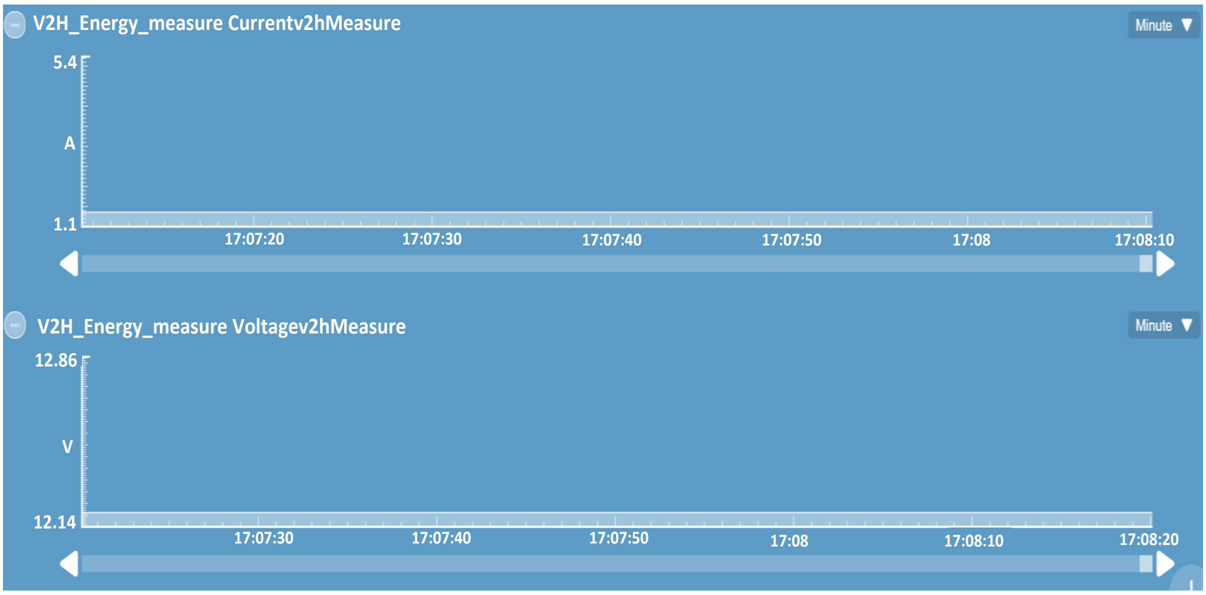


Fig. 36. Measured voltage and current for the Nissan Leaf battery.

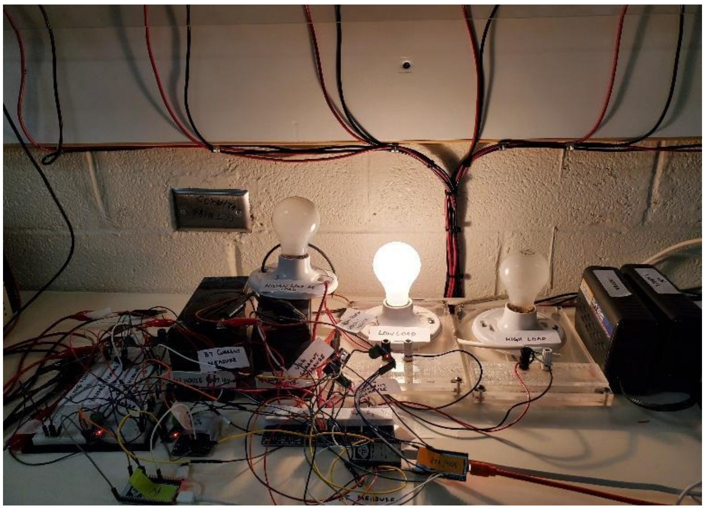
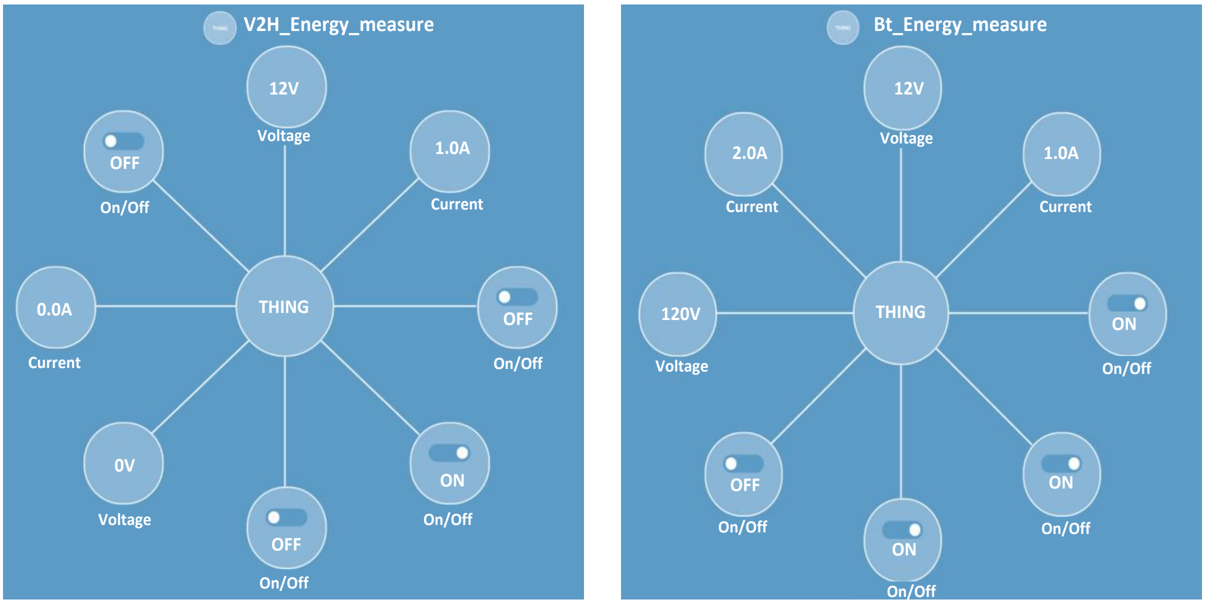


Fig. 37. Experimental setup for the V2H mode.

4.2.6. Nighttime Charging Mode

When the load energy demand is minimal, the in-house battery is utilized to charge the Nissan Leaf battery [21]. Fig. 38(a) shows overall V2H/V2G GUI, Fig. 38(b) shows overall in-house battery GUI; switches S3, S4, and S7 are HIGH in this mode. The voltage and current measurement graph for the in-house battery is shown in Fig. 39. The load is working at 120 V and 1 A, as shown in Fig. 40, while the Nissan Leaf battery is charging at 12 V and 1 A, as shown in Fig. 41. Figs. 39, 40 and 41 illustrate the test output for Nighttime charging mode. It can be seen from the graph that the in-house battery was discharging to charge Nissan Leaf's battery. This was achieved by switching ON the relays S3, S4 and S7.



(a) (b)

Fig. 38. Overall GUI for: a) V2H/V2G; b) the in-house battery.

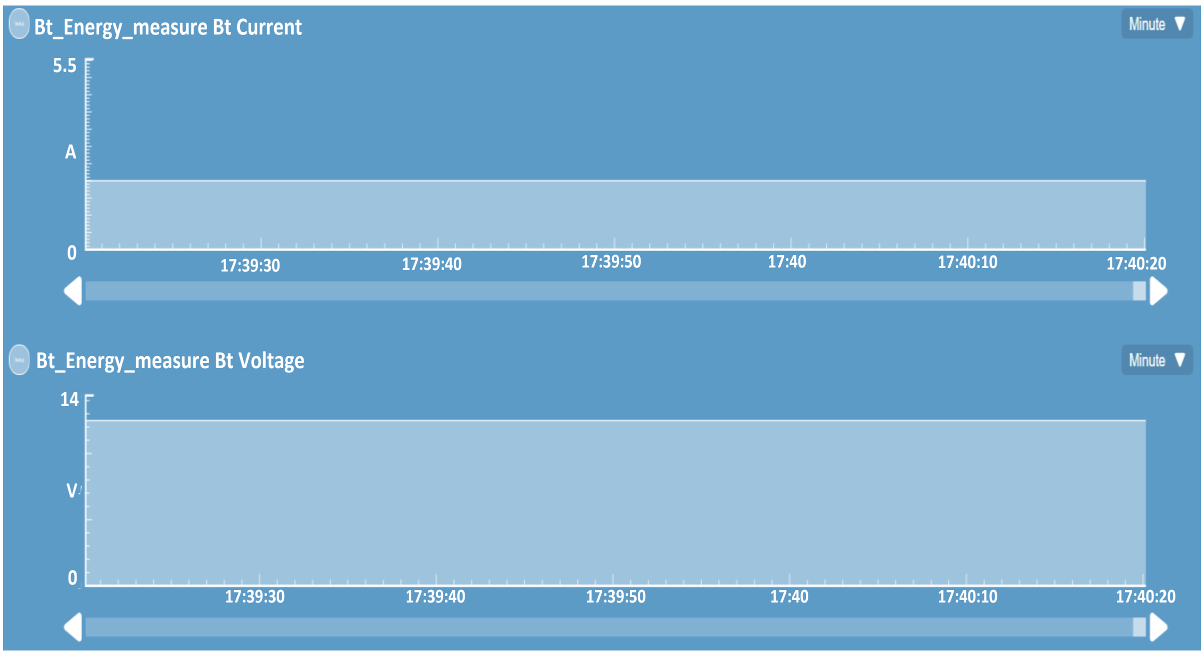


Fig. 39. Measured voltage and current for the in-house battery.

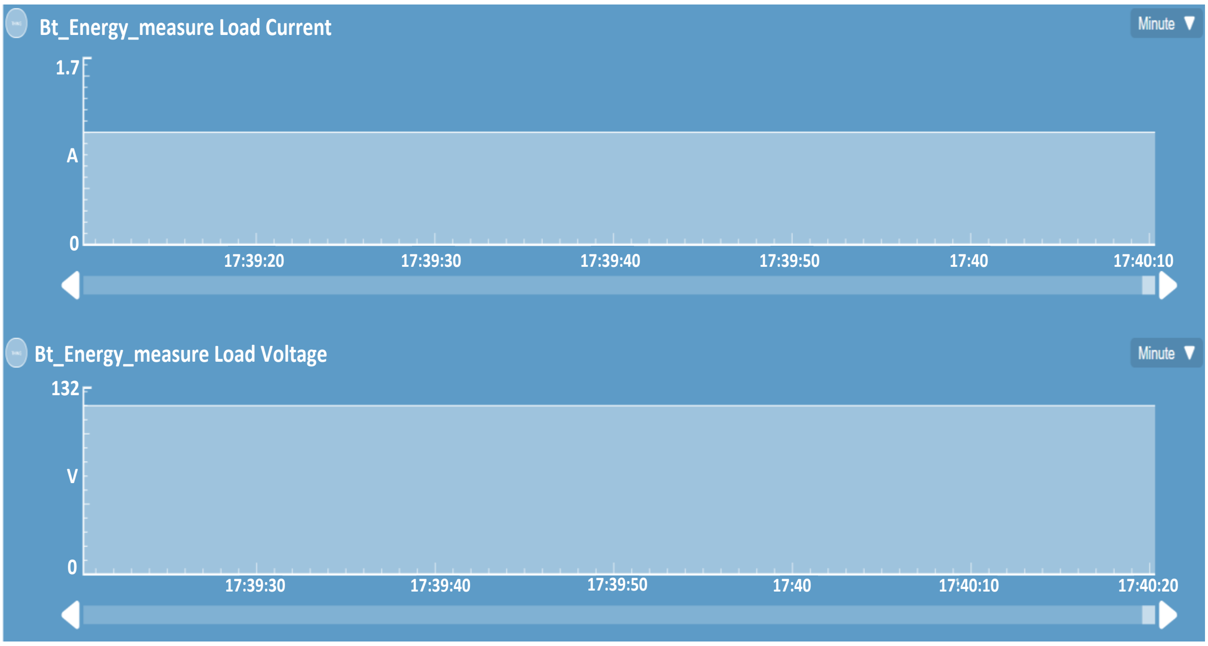


Fig. 40. Measured voltage and current for the load.

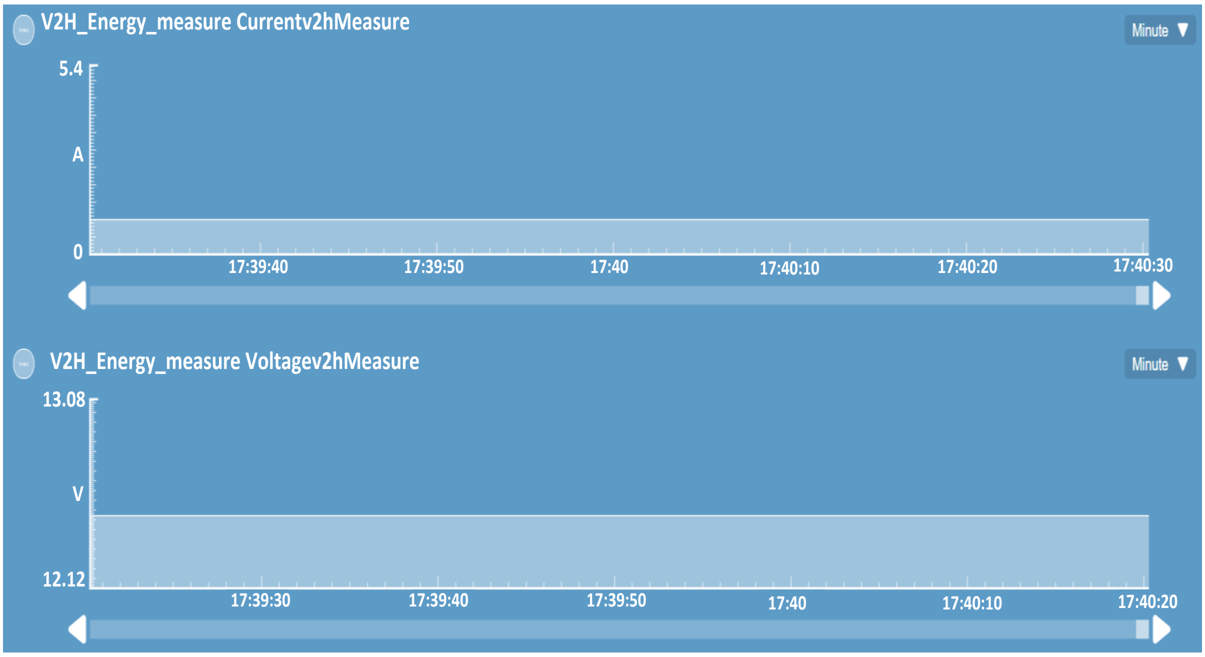


Fig. 41. Measured voltage and current for the Nissan Leaf battery.

4.2.7. V2G Mode

The car disconnects from the home in this mode, and the stored energy is redirected to the grid. A light bulb is used as a load in this example, and the load is powered by an inverter. Fig. 42 depicts the entire V2H/V2G GUI with switch S9 in the HIGH position. The Nissan Leaf's battery is discharging at 3 A and 12 V, as shown in Fig. 43, while the load is consuming 1 A and functioning at 120 V, as shown in Fig. 44. The experimental setup for the V2G mode is shown in Fig. 45. Figs. 43 and 44 illustrate the test output for V2G mode. It can be seen from the graph that the Nissan Leaf's battery was discharging to reel power the grid. This was achieved by switching ON the relay S9.

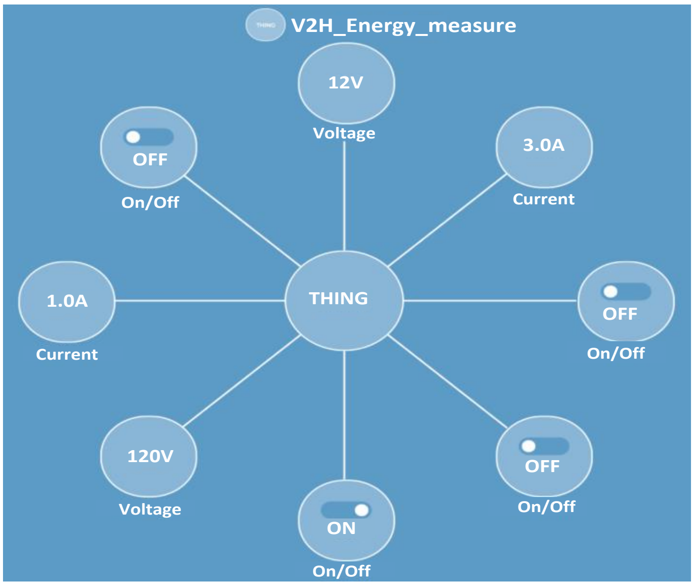


Fig. 42. Overall V2H/V2G GUI.

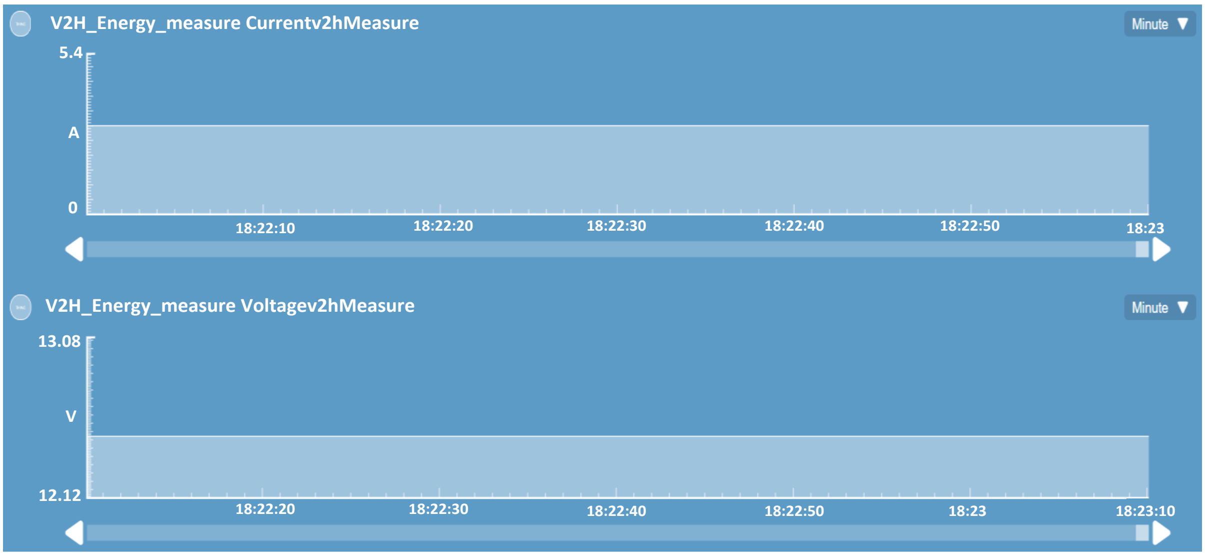


Fig. 43. Measured voltage and current for the Nissan Leaf battery.

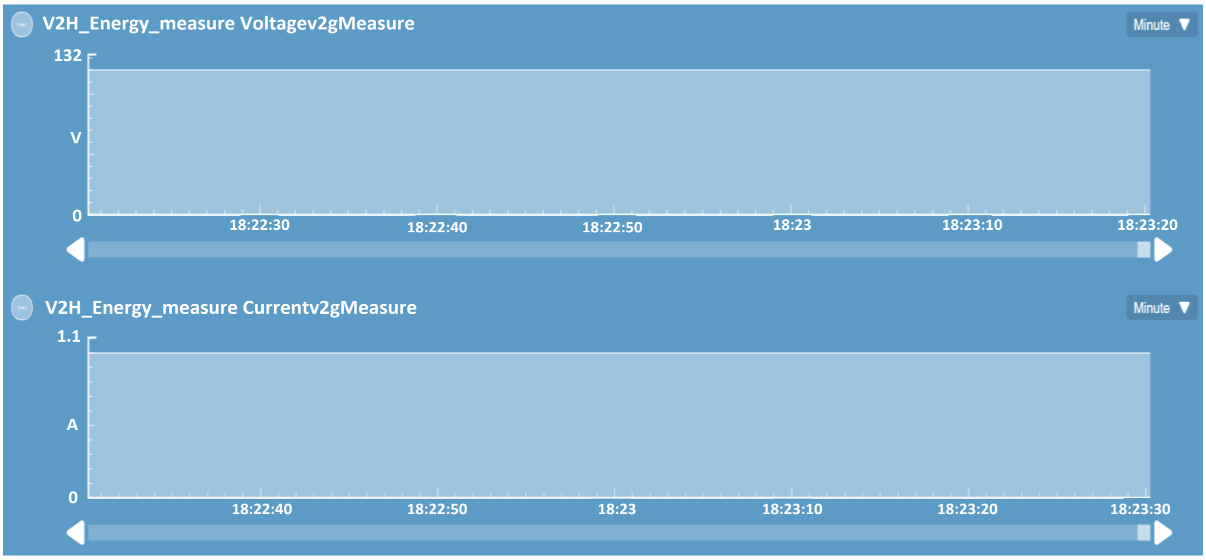


Fig. 44. Measured voltage and current for the grid as load.

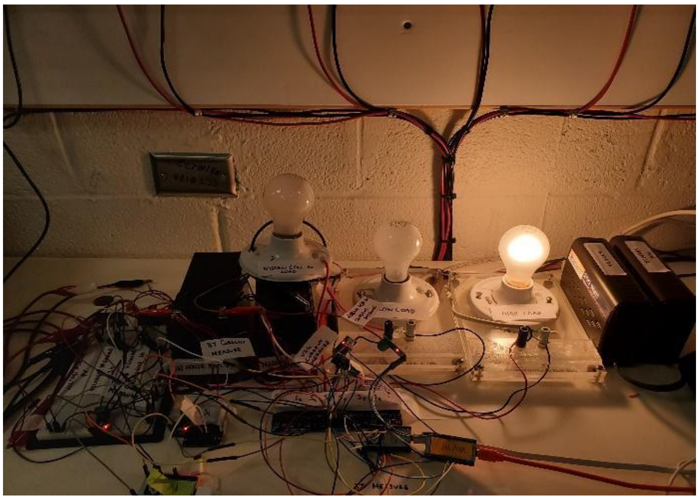


Fig. 45. Experimental setup for the V2G mode.

5. CONCLUSIONS

Through Mozilla IoT, a solar energy system supervisory control was successfully implemented with V2H and V2G options for Newfoundland conditions. The entire experimental system was powered by 12 V, and it was used to demonstrate the concept of IoT, as well as remote control and monitoring. The actual system, on the other hand, contains several DC-DC converters and a variety of voltage levels. The current and voltage sensors generate data, which the ESP 32 microcontroller collects and sends to the WebThings gateway hosted on the Raspberry Pi. Using Mozilla's https tunneling, the data transferred would be seen on the web. Because of its interoperability and privacy, Mozilla IoT was considered. The data is sent to the gateway through IoT, but it is not saved in the cloud; instead, it is kept locally. Three ESP 32 microcontrollers were used to run the experiment in eight different modes, with various parameters such as PV current and voltage, in-house batteries current and voltage, load current, and Nissan Leaf batteries current and voltage being monitored and logged. These parameters were transmitted in real time to the gateway. Because of WebThings, this system is scalable, open source, and interoperable, and it can also be implemented in production. Furthermore, because this research is scalable, it can be implemented in EV parking lots, connect several residences to form a microgrid, implement a V2X model, where cars are used to schedule loads [22, 23], power microgrids or enterprises, or implement V2V charging [24, 25].

Acknowledgement: The authors express their sincere gratitude to LUX Flavors PVT Ltd for funding this research.

REFERENCES

[1] R. Sundararajan, M. Iqbal, "Dynamic modelling of a solar energy system with vehicle to home and vehicle to grid option for newfoundland conditions," *European Journal of Electrical Engineering and Computer Science*, vol. 5, no. 3, pp. 45-53, 2021.

[2] R. Sundararajan, M. Iqbal, "Design of an IoT interface for a solar energy system with vehicle to home option for newfoundland conditions," *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference*, pp. 0597-0601, 2020.

[3] H. Shin, R. Baldick, "Plug-in electric vehicle to home (V2H) operation under a grid outage," *IEEE Transactions on Smart Grid*, vol. 8, no. 4, pp. 2032-2041, 2017.

[4] F. Shakeel, O. Malik, "Vehicle-to-grid technology in a micro-grid using DC fast charging architecture," *2019 IEEE Canadian Conference of Electrical and Computer Engineering*, pp. 1-4, 2019.

[5] H. Turker, I. Colak, "Multiobjective optimization of grid- photovoltaic- electric vehicle hybrid system in smart building with vehicle-to-grid (V2G) concept," *2018 7th International Conference on Renewable Energy Research and Applications*, pp. 1477-1482, 2018.

[6] S. Ramalingam, K. Baskaran, D. Kalaiarasan, "IoT enabled smart industrial pollution monitoring and control system using Raspberry Pi with BLYNK server," *2019 International Conference on Communication and Electronics Systems*, pp. 2030-2034, 2019.

[7] V. Monteiro, J. Pinto, J. Afonso, "Operation modes for the electric vehicle in smart grids and smart homes: present and proposed modes," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 3, pp. 1007-1020, 2016.

[8] J. Gupta, B. Singh, "A bidirectional home charging solution for an electric vehicle," *2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe*, pp. 1-6, 2019.

[9] J. Pinto, V. Monteiro, H. Gonçalves, B. Exposto, D. Pedrosa, C. Couto, J. Afonso, "Bidirectional battery charger with grid-to-vehicle, vehicle-to-grid and vehicle-to-home technologies," *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, pp. 5934-5939, 2013.

[10] M. Longo, W. Yaïci, F. Foiadelli, "Electric vehicles charged with residential's roof solar photovoltaic system: a case study in Ottawa," *2017 IEEE 6th International Conference on Renewable Energy Research and Applications*, pp. 121-125, 2017.

[11] G. De Lazari, M. Sperandio, "Vehicle-to-home evaluation in Brazil," *2019 IEEE PES Innovative Smart Grid Technologies Conference - Latin America*, pp. 1-6, 2019.

[12] Y. Wang, O. Sheikh, B. Hu, C. Chu, R. Gadh, "Integration of V2H/V2G hybrid system for demand response in distribution network," *2014 IEEE International Conference on Smart Grid Communications*, pp. 812-817, 2014.

[13] H. Chtioui, G. Boukettaya, "Vehicle-to-grid management strategy for smart grid power regulation," *2020 6th IEEE International Energy Conference*, pp. 988-993, 2020.

[14] E. Stark, F. Schindler, E. Kučera, O. Haffner, A. Kozáková, "Adapter implementation into Mozilla webthings IoT platform using Javascript," *2020 Cybernetics and Informatics*, pp. 1-7, 2020.

[15] T. Tavade, P. Nasikkar, "Raspberry Pi: data logging IOT device," *2017 International Conference on Power and Embedded Drive Control*, pp. 275-279, 2017.

[16] D. Bolla, J. Jijesh, S. Palle, M. Penna, Keshavamurthy, Shivashankar, "An IoT based smart e-fuel stations using ESP-32," *2020 International Conference on Recent Trends on Electronics, Information, Communication and Technology*, pp. 333-336, 2020.

[17] I. Allafi, T. Iqbal, "Design and implementation of a low cost web server using ESP32 for real-time photovoltaic system monitoring," *2017 IEEE Electrical Power and Energy Conference*, pp. 1-5, 2017.

[18] S. Biswas, M. Iqbal, "Solar water pumping system control using a low cost ESP32 microcontroller," *2018 IEEE Canadian Conference on Electrical and Computer Engineering*, pp. 1-5, 2018.

[19] U. Ashraf, M. Iqbal, "An open source SCADA for a solar water pumping system designed for Pakistani conditions," *2021 IEEE 11th Annual Computing and Communication Workshop and Conference*, pp. 1403-1408, 2021.

[20] M. Aachiq, T. Oozeki, Y. Iwafune, J. Fonseca Jr, "Reduction of PV reverse power flow through the usage of EV's battery with consideration of the demand and solar radiation forecast," *2013 IEEE International Electric Vehicle Conference*, pp. 1-3, 2013.

[21] H. Turker, "Optimal charging of plug-in electric vehicle (PEV) in residential area," *2018 IEEE Transportation Electrification Conference and Expo*, pp. 243-247, 2018.

[22] N. Breum, M. Joergensen, C. Knudsen, L. Kristensen, B. Yang, "A charging scheduling system for electric vehicles using vehicle-to-grid," *2019 20th IEEE International Conference on Mobile Data Management*, pp. 351-352, 2019.

[23] S. Das, P. Acharjee, A. Bhattacharya, "Charging scheduling of electric vehicle incorporating grid-to-vehicle (G2V) and vehicle-to-grid (V2G) technology in smart-grid," *2020 IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy*, pp. 1-6, 2020.

[24] D. Guo, P. Yi, C. Zhou, J. Wang, "Optimal electric vehicle scheduling in smart home with V2H/V2G regulation," *2015 IEEE Innovative Smart Grid Technologies - Asia*, pp. 1-6, 2015.

[25] A. Gautam, A. Verma, M. Srivastava, "A novel algorithm for scheduling of electric vehicle using adaptive load forecasting with vehicle-to-grid integration," *2019 8th International Conference on Power Systems*, pp. 1-6, 2019.