IOT BASED CENTRALIZED NET METER MONITORING AND MANAGEMENT SYSTEM FOR GRID CONNECTED SOLAR PV

By

MD. MUSTAQ MUJAHID
22121030
MD. ZUBAYET HOSSAIN ZUBORAZ
22121057
AJMAIN IQTIAD
22121017
MD. SHAKIL HASAN
22121067

A Final Year Design Project (FYDP) submitted to the Department of Electrical and Electronic Engineering in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering

Declaration

It is hereby declared that

1. The Final Year Design Project (FYDP) submitted is my/our own original work while completing degree at Brac University.
2. The Final Year Design Project (FYDP) does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The Final Year Design Project (FYDP) does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. I/We have acknowledged all main sources of help.

Student's Full Name & Signature:

MD. MUSTAQ MUJAHID **22121030**	**MD. ZUBAYET HOSSAIN ZUBORAZ** **22121057**
AJMAIN IQTIAD **22121017**	**MD. SHAKIL HASAN** **22121067**

Approval

The Final Year Design Project (FYDP) titled "IOT BASED CENTRAL NET METERING MONITORING AND MANAGEMENT SYSTEM FOR GRID CONNECTED SOLAR PV" submitted by

1. **MD. MUSTAQ MUJAHID (22121030)**
2. **MD. ZUBAYET HOSSAIN ZUBORAZ (22121057)**
3. **AJMAIN IQTIAD (22121017)**
4. **MD SHAKIL HASAN (22121067)**

of Fall, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of of Bachelor of Science in Electrical and Electronic Engineering on 05/01/2026.

Examining Committee:

Academic Technical Committee (ATC): (Chair)

Dr. Shahidul Islam Khan
Professor
Department of Electrical and Electronic Engineering

Final Year Design Project Coordination Committee: (Chair)

Dr. Mirza Rasheduzzaman
Associate Professor
Department of Electrical and Electronic Engineering

Department Chair:

17.02.26

Md. Mosaddequr Rahman, PhD
Professor and Chairperson
Department of Electrical and Electronic Engineering

Ethics Statement

We do make the following declaration:

1. The content is the exclusive work of the mentioned authors and it is compiled in the form of the original directing the abovementioned Academic Technical Committee panel members.
2. References have been appropriately credited and cited.
3. BRAC University Plagiarism Policy has been strictly followed.
4. The reported similarity index of this report, obtained using the standard plagiarism checking tool Turnitin is 15% .

Abstract / Executive Summary

The integration of rooftop solar photovoltaic (PV) systems through net metering has significant potential to improve energy sustainability in Bangladesh; however, the lack of intelligent monitoring and control often leads to inefficient utilization of installed systems. This project presents the design and hardware implementation of an IoT-based centralized net meter monitoring and management system for a grid-connected solar PV battery setup. The ESP32 microcontroller is a component of the system that measures real-time electrical parameters (voltage, current, power, energy import/export, and battery state-of-charge). Control logic based on relays is used to have automatic control over grid import, grid export and battery operation. The visualization of system data is provided with an IoT dashboard to give a clear understanding of net metering behavior. A hardware prototype was created and tested in order to confirm the system performance. The feasibility was also assessed by carrying out an economic analysis based on cost breakdown, payback period and sensitivity analysis. The outcomes showcase a better visibility of energy, efficiency in operations and realistic use as far as rooftop solar systems are concerned.

Keywords: Net metering; IoT-based energy monitoring; ESP32; Rooftop solar PV; Battery Energy Management; Smart grid.

Dedication

This is a project that is dedicated with love to our parents. Our academic life has been based on their unconditional support, sacrifice, encouragement and believing in our capabilities. Any success reached is the expression of their advice, patience and insatiable drive. It is a little salute to their undying love and devotion.

Acknowledgement

Without the advice, encouragement, and assistance of some people and organizations, the successful implementation of this project would have been impossible. To begin with, we would wish to mention that we would like to thank LTO Shafi who had been providing supervision, constructive feedback, and worthwhile insights throughout the duration of this project. His guidance played a crucial role in shaping the technical direction and ensuring the practical relevance of the work. We are also deeply thankful to Tareq Aziz Sir for his academic support, motivation, and thoughtful suggestions, which significantly contributed to the improvement of the project and the final report. We would like to acknowledge the Department of Electrical and Electronic Engineering at BRAC University for providing the necessary facilities and academic environment to carry out this work. Finally, we extend our appreciation to our peers and family members whose encouragement and support helped us complete this project successfully.

Table of Contents

List of Tables:

List of Figures :

List of Acronyms

Acronym	Description
ADC	Analog-to-Digital Converter
BERC	Bangladesh Energy Regulatory Commission
DER	Distributed Energy Resource
ESP32	Espressif 32-bit Microcontroller
IEA	International Energy Agency
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
INA226	Current and Power Monitoring IC
IRENA	International Renewable Energy Agency
LCD	Liquid Crystal Display
LDR	Light Dependent Resistor
MPPT	Maximum Power Point Tracking
OPC-UA	Open Platform Communications - Unified Architecture
PV	Photovoltaic
PWM	Pulse Width Modulation
PZEM	Power Measurement Module

SCADA	Supervisory Control and Data Acquisition
SOC	State of Charge
SREDA	Sustainable and Renewable Energy Development Authority
VPP	Virtual Power Plant

Chapter 1: Introduction

1.1 Introduction

Rapid urbanization, population growth, and industrial expansion have significantly increased electricity demand across the world, placing substantial stress on conventional power generation systems. The developing states like Bangladesh are especially sensitive because of the low production capacity, overreliance on foreign fossil fuels, and the rising energy prices. These issues have further increased the demand of sustainable, decentralized and cost effective energy solutions capable of meeting the long-term economic development besides minimizing environmental effects [1].

Solar photovoltaic (PV) systems have become one of the most viable technologies of renewable energy that can be deployed in urban areas. Solar PV atop roofs, specifically, allows producing electricity at the location of consumption and minimizes losses during transmission at the national grid, as well as relieves pressure in the peak hours. In urban areas with a high population density and a lack of land, rooftop solar is a viable option as a way to distributed generation and a more resilient grid [2].

Having realized this opportunity, Government of Bangladesh has implemented a number of regulatory schemes to hasten the adoption of rooftop solar. They are the Net Metering Guideline (2018) which enables consumers to sell excess electricity to the grid, the Renewable Energy Policy (2020), and the Rooftop Solar Mandate (2023) that obliges new constructions with more than a specific rooftop area to install solar on their roofs [3]. All these policies are meant to make consumers of electricity proactive energy producers besides decreasing the use of fossil fuel generation.

Nevertheless, the efficiency of the rooftop solar systems does not necessarily lie in their installation, but in the process of intelligent monitoring, control, and integration with the utility grid. Net metering systems involve proper determination of two-way flow of energy, precise decision making of both import and export business, and transparent visualization of activities of the system. In the absence of such possibilities, rooftop solar systems will remain underutilized resources instead of viable providers of national energy goals.

The latest developments in Internet of Things (IoT) technologies and embedded systems have been used to implement solutions that are low-cost and provide real-time energy monitoring and automated control. The IoT-based designs enable continuous data-gathering, remote accessibility, and smart power management through small microcontrollers and cloud applications. Such technologies can be used to address the gap between policy intent and practical implementation to make rooftop solar installations effective, reliable, and compliant with regulatory limits when applied to grid-connected solar PV battery systems [4].

1.1.1 Problem Statement

There is a severe discrepancy between the intended policy on renewable energy and the actual implementation of rooftop solar photovoltaic (PV) systems in Bangladesh. The Government of Bangladesh adopted the Rooftop Solar Mandate in October 2023 to address the increasing electricity demand, reliance on fuel imports, and environmental issues and mandate solar PV systems on new structures with rooftop areas larger than 92.2 m2. Although this policy is trying to speed up distributed renewable generation, the practical implementation of this policy has shown that it has systemic inefficiencies, which disempower the intended effect of this policy [5].

In large cities like Dhaka, where there is an excess of buildings (over 400,000) rooftop solar installations are beginning to be treated as a locality-driven obligation instead of an energy resource to be actively managed. Much of the installed PV systems is not used or not well monitored or even not connected to the grid. The existing estimates show that rooftop solar is less than 5 percent of urban electricity demand even though there is vast rooftop potential and that shows a wide disconnect between the potential capacity and energy used effectively [6].

Everything in this problem is based on the fact that there is no smart monitoring, control, and management infrastructure. The majority of rooftop solar systems do not have real-time access to energy production, use, battery storage, and grid connection. The user cannot know whether solar energy is being used locally, stored, or exported to the grid or it is going to waste. Moreover, traditional conventional net metering applications are dependent on manual billing systems and central electricity meters and provide no actionable feedback to consumers and no operational transparency [7].

The problem is also aggravated by grid integration. The distribution network in Bangladesh is not in its current form to support the bidirectional power flow in large scale that is supplied by the distributed generators. In the absence of smart control systems, exported solar may cause voltage swings, difficulties of reverse power flow, and uncertainty of operation on the part of the utility. Consequently, a good number of grid operators are still not keen on allowing net metering in totality and consumers have no faith in the ability to sell excess energy [8].

These technical constraints are enhanced by economic and institutional constraint. The initial cost of solar equipment is high, foreign sourcing of materials, unavailability of affordable funding and uneven application of the renewable energy regulations restrict the intentions to adopt it in greater numbers than the minimum requirements. Moreover, smaller digital monitoring platforms do not allow policymakers and utilities to measure the performance of rooftop solar efficiently, which undermines long-term planning and policy review [9].

As a result, solar PV installations in Bangladesh on rooftop are at risk of turning into symbolic elements instead of energy security and cost-reduction tools and emission mitigation. The goals of the Rooftop Solar Mandate, Net Metering Guideline, and Renewable Energy Policy can never be achieved without a credible, scalable, and user-friendly monitoring and management system. The proposed project fills this gap with the creation of an IoT-based centralized net meter monitoring and control system that will allow seeing energy in real-time, allow the grid to interact with solar cells on a rooftop, and make the use of solar cells on a rooftop practical.

1.1.2 Background Study

Energy is a pressing problem in Bangladesh at the moment, as the demand in electricity is growing at a very high rate, caused by urbanization, the development of industry, and population explosion. The Government of Bangladesh launched the Rooftop Solar Mandate (October 2023), to ensure that the country does not rely heavily on fossil fuels and instead encourages the use of clean energy. This is a requirement that applies to all new buildings whose areas on the rooftop are above 92.2 m2 and install solar photovoltaic (PV) systems. The policy states that connections of single-phase should at least install a 1 kW capacity and connections of three phases should at least install 3 kW capacity.

Such a project can revolutionize the power usage in cities. Over 200,000 roofs in Dhaka alone can have solar installed which can potentially provide a capacity of over 200 MW. This notwithstanding, rooftop solar is contributing less than 5 to the electricity demand of the city. This is mainly attributed to the fact that most of the installations are done as a form of compliance and not necessarily monitored, integrated and managed in terms of energy.

Key challenges include:

- **Absence of monitoring and control:** The majority of systems do not use digital means of monitoring generation, consumption, or battery storage, which results in the idle use of energy.
- **Weak grid integration:** The lack of proper bi-directional control systems stands in the way of the users having the facility to sell excess power to the grid effectively.
- **Lack of maintenance:** Faults are usually not detected because the real-time monitoring is not provided and it makes the process less efficient and shorter-lived.
- **Lack of knowledge:** In most cases, users lack the knowledge of how rooftop solar may cut down their utility bills by a large margin and enhance grid reliability.

Survey of Existing Research

To fill these gaps, the world has been undertaking numerous studies on the topic of IoT-enabled renewable energy system:

- **Bijoy et al. (2024)** proposed a smart net-metering system of distributed PV, which reveals the potential of feedback and billing with the help of the IoT.
- **Jamal et al. (2020)** proposed an IoT-based cost-effective metering device that is customized to suit residential users.
- **Rind et al. (2023)** highlighted that IoT is essential in smart metering especially regarding real-time data analytics and automation.
- **Pathare et al. (2024)** also modeled an IoT-enabled net-metering system in MATLAB, showing effective integration of renewable sources.

All these studies prove that IoT technologies may be used to enhance monitoring, efficiency, and demand-side control. However, gaps remain:

1. **Resilience:** The majority of designs are strongly dependent on cloud connection, which is not very reliable in locations where the internet is not stable.
2. **Compliance with the policies:** Limited implementations refer to the particular needs of Rooftop Solar Mandate (2023) in Bangladesh.
3. **Scalability:** Not all solutions can scale between small households and large commercial or industrial environments.
4. **User interaction:** Dashboards and interfaces are not necessarily user-intuitive in their visualization and actionable insights, but usually prioritize technical information.

Knowledge and Skills Acquired

The survey based on this background study and research has allowed the project team to become experienced in:

- **Solar PV and Microgrid Design:** Battery energy storage, bidirectional grid connection, and the nature of PV.
- **IoT and Data Communication:** The familiarity with microcontrollers, wireless communication, safe protocols and remote data management.
- **Modeling and Simulation:** The skill of using MATLAB/Simulink to test the system design and software to analyze data.
- **Data Analytics and Machine Learning:** The use of algorithms to forecast loads, predictive energy management, and optimization.
- **Regulatory and Safety Standards:** Knowledge of the national policies in Bangladesh (Net Metering 2018, Renewable Energy Policy 2020, Rooftop Solar Mandate 2023), as well as international standards (IEEE 1547 to connect to the grid and IEC 62109 to make PV systems safe).

With this knowledge base, the proposed system will help in solving the issue of technical inefficiencies as well as policy requirements that would bridge the gap between government requirements and the application of rooftop solar in a practical way.

1.1.3 Literature Gap

There is a considerable amount of literature on the use of Internet of Things (IoT)-based technologies in renewable energy systems, especially in the framework of smart metering and grid-linked solar photovoltaic (PV) systems. A number of studies show that the IoT-enabled sensors, microcontrollers, and cloud platforms can be used to monitor the energy generation and consumption in distributed PV systems.

A net-metering system was suggested by Bijoy et al. [10] which enables connecting solar PV systems to the grid, noting the efficiency of the IoT-based monitoring in terms of energy feedback and billing transparency. Jamal et al. [11] created a low-cost IoT-based energy metering chip targeted at the residential customers, with the focus on low costs and easy installation. In a similar manner, Rind et al. [12] showed IoT-based smart meters, which are used to acquire and visualize real-time data (in terms of consumption side). Pathare et al. [13] simulated in MATLAB a net-metering system based on IoT which confirmed the bidirectional power flow and integration of renewables.

Although these studies verify that the technical feasibility of IoT-based energy monitoring is achievable, some of its major constraints are not discussed in any way. First, the majority of the systems proposed are cloud-centric and are susceptible to the instability of the internet, and are inappropriate in areas where a reliable connection is a frequent issue in developing nations like Bangladesh. Without edge level intelligence, system resilience and local decision-making in the event of a network outage is limited.

Second, bidirectional energy control has not been developed in many cases. A lot of the available literature concentrates more on energy measurement and visualization as opposed to active management of power flow. This leads either to automated control of grid import, grid export, battery charging, and load prioritization being not implemented, or it being implemented in simple rule-based ways without system-level coordination.

Third, policy-specific implementation has not been examined much. Majority of literature takes care of generic net-metering ideas and does not match system architecture with national requirements. Specifically, the Rooftop Solar Mandate (2023), Net Metering Guideline (2018), and Renewable Energy Policy (2020) of Bangladesh have little attention. This is because the lack of policy-conscious system design constrains practical application and uptake by utilities and regulators.

Fourth, scalability and centralized monitoring is not well covered. Current solutions are normally configured to suit individual households or standalone developments. Limited literature suggests architectures that can achieve centralized monitoring and control in more than two buildings, campuses, or industrial sites, which is necessary to accomplish urban rooftop solar aggregation and smart city implementation.

Lastly, user interaction and actionable communication are poor. In the majority of cases, dashboards focus on the raw electrical values and do not include the intuitive visualization, predictive analytics, or the ability of the user to optimize energy consumption, battery behavior, or grid interaction.

As such, the research gap is the lack of a centralized, scalable and policy-conformant IoT-based net meter monitoring and management system that incorporates real-time sensing, two-way power control, battery energy control and robust communication architecture. This is the gap which needs to be addressed to turn rooftop solar installations into an active energy source, not merely a compliance tool, but in accordance with the national energy goals.

1.1.4 Relevance to current and future Industry

The suggested IoT-Based Centralized Net Meter Monitoring and Management System is directly related to the current changes in the needs of the power and energy industry, specifically in areas where the integration of renewable energy occurs rapidly. Since power systems are no longer centrally generated but exist in distributed forms, digitally controlled infrastructures, intelligent monitoring and control platforms are being necessary, not optional.

i) Relevance to Current Industry Practices

Currently, the energy sector is experiencing a massive transition to distributed energy resources (DERs), such as rooftop solar PV systems, and battery energy storage. The needs of utilities, industries, and commercial facilities are growing in the need of digital solutions that are able to offer them operational visibility and control over several points of the generation. To meet these short-term industry needs, the proposed system will make the following contributions:

Smart Grid Enablement:

The system facilitates two ways energy monitoring and control, which is an essential prerequisite of the contemporary grid-tied renewable systems that are deployed within the framework of net metering.

Industrial and Commercial Energy Management:

Monitoring of real time energy importation, exportation and storage will allow factories and commercial buildings to lower peak demand rates, enhance load scheduling, and make power more reliable.

Utility-Level Data Transparency:

Centralized monitoring gives utilities high quality time-stamped operational data to help them plan the grid better, reduce losses, and audited performance.

Compliance with Regulatory Mandates:

The solution ensures reduced compliance risk to the building owners since they can operate their systems in line with the national policies of net metering, the rooftop solar mandate, and facilitates easier regulatory implementation [14].

ii) Relevance to Emerging Industry Trends

In the future, the energy industry will be digitalized due to automatization, data analysis, and smart controls. The developed system is properly placed to accommodate the industry trends in the future:

Transition to Smart and Digital Grids:

Integration of IoT-based sensing and control is in line with the world internationalization to smart grids where real-time information and automated decision making forms are essential to stability and efficiency [15].

Scalable Aggregation of Distributed Resources:

Multiple rooftop solar systems can be monitored centrally to allow aggregation of several rooftop solar systems, resulting in virtual power plants (VPPs) and synchronized demand-response programs.

Integration of Edge Intelligence and AI:

An edge computing and machine learning algorithm load forecasting, fault detection, and predictive maintenance can be incorporated into the system architecture in the future.

Support for Smart Cities and Campuses:

Smart Cities, Smart Campuses may be supported by the government through local, regional, or national frameworks. Supporting Smart Cities and Campuses: The government can support Smart Cities and Campuses either at the local or regional or national levels. The suggested structure can be applied to the constructs of smart city, university campus, and industrial park implementations, where central energy monitoring is a requirement to manage sustainability.

Decarbonization and ESG Compliance:

Quality energy information and reporting can underpin corporate sustainability objectives, carbon reporting, and Environmental, Social, and Governance (ESG) reporting and compliance standards [16].

iii) Industry Impact Summary

As an industry solution, the proposed system will change the rooftop solar installations into isolated compliance-driven assets to actively managed and grid-interactive energy sources.

The solution helps to narrow the gap between policy needs, expectations of utility, and energy management at the user level, which leads to the modernization of power systems and long-term development of the sustainable energy infrastructure.

1.2 Objectives, Requirements, Specification and constant

1.2.1. Objectives

1. Monitor grid import and export energy separately
2. Estimate battery state-of-charge (SOC) and backup time
3. Implement automatic relay-based power flow control
4. Display real-time system data locally and via IoT

1.2.2 Functional and Nonfunctional Requirements

Functional Requirements

- Constant control of the generation of sunlight, storage, demand on loads, and grid communication.
- Automated power management in order to optimize the use of batteries.
- Smart net-metering with exportation of unused energy to the grid.
- Liveness with real-time visualization and alerts and web- and mobile-based dashboards.

Non-Functional Requirements

- Encryption and authentication of data communication.
- Universally applicable design to homes, organizations and factories.
- Hard and weatherproof hardware that can be used over the long term in the field.
- Remote support Remote maintenance such as firmware updates and diagnostics.
- A high degree of reliability, efficiency, and adherence to the international safety standards.

1.2.2 Specifications

Specifications

The given system will be made of four subsystems:

1. **Power Generation and Storage**
    - Energy generation rooftop solar PV modules.
    - Backup and peak load battery storage.
    - Two-way inverter in order to be grid tied safely.
2. **Control and Processing**
    - System management, data collection, and data control logic central processing unit.
    - Renault Resilience Local processing (edge computing) in case of internet outages.
3. **Sensing and Measurement**
    - Electrical parameters (voltage, current, flows of energy) are measured.
    - Surveillance of the environmental factors like solar irradiance and temperature.
    - To avoid over-discharge or over heat of batteries they monitor their health.
4. **Communication and Interface**
    - Remote connection by using secure wireless communication.
    - Long-term performance tracking storage/analytics in the clouds.
    - Local dashboards and mobile/web real time user interaction.

1.2.3 Technical and Non-technical consideration and constraint in design process

- **Technical Considerations**
    - The unstable internet connectivity in the rural locations can cause decreased trust in the cloud-only solutions.
    - The sensor errors could include inaccuracies in control and reporting.
    - In the absence of permanent connectivity, there will be delays when tracking real-time data to cloud dashboards, which will impact the user visibility.
- **Non-Technical Considerations**
    - Community-level projects may experience limitations of adoption due to budget constraints.

- Seldom a supply of skilled man power in installations and maintenance.
- Long-term implementation strategies can be influenced by possible policy changes.
- The reliance on imported equipment can escalate the expenses and slow down deployment.

1.2.4 Applicable compliance, standards, and codes

In a bid to achieve energy requirements and ensure safety and adaptable development, this project adopts different standards that have been introduced. The choices concerning selected standards or codes may influence the installation, configuration as well as the operation of the solar-powered microgrid setup.

A. National Standards & Policies

1. **Net Metering Guideline 2018 - Bangladesh Energy Regulatory Commission (BERC)**
   It explains how bi-directional energy metering may be executed on the grid-connected solar systems. It says what is eligibility, the maximum amount of power per device, how to construct a system, check calibration requirements, bill policies and payment schedules.

   - **Applicability to Project:**
     In our project, we will introduce a smart energy meter that will be able to measure imported and exported electricity. This guideline should be followed, particularly when the prototype is proven in a grid-connected urban environment. The system will operate on time-synchronized logging and smart meters, which will be in accordance with the accuracy and certification requirements.

   - **Relevant Sections:** Sections 4 (System Design), 5 (Interconnection and Metering) and Section 6 (Billing and Credit Mechanism).

2. **Renewable Energy Policy 2020 - Ministry of Power, Energy and Mineral Resources**
   The policy provides the strategic direction of renewable energy development in Bangladesh where 10 percent of the national electricity mix will be achieved by the year 2030 through renewable energy sources. It encourages off-grid and semi-urban off-grid and community-based solar systems.

- **Applicability to Project:**
  Our system directly relates to this by facilitating the process of local energy production, intelligent distribution and battery storage, which contributes to less reliance on the grid. It also promotes the use of rooftop solar in learning and commercial institutions.

3. **Rooftop Solar Mandate 2023 - Government of Bangladesh**
   This directive which came into effect in October 2023 requires the installation of rooftop solar in all new residential, commercial, educational, and industrial constructions with a rooftop area of more than 92.2 m2. It also gives minimum requirements of generation capacity depending on the type of building.

   - **Applicability to Project:**
     Our modular microgrid can be customized to different sizes of rooftops and load requirements. The project is a compliance enabler which enables building owners to comply with the law in real time monitoring and automated control features.

B. International Standards

4. **IEEE 1547-2018 - Interconnection and Interoperability of Distributed Energy Resources (DER)**

This standard establishes the technical specifications and testing standards of interconnections of DERs (such as solar PV) with utility grids. It consists of regulations regarding the calculation of voltage, frequency response, anti-islanding protection, and communication interface.

- **Applicability to Project:**
  We have a grid-tied mode which needs to be synchronized with utility voltage and frequency, fault handling and control of reactive power. Every inverter and energy transfer logic will be compliant with the IEEE 1547 standards of safe and standard grid interaction.

5. **IEC 62109-1 & IEC 62109-2 - Safety of Power Converters for Use in PV Systems**

The standards give fundamental safety considerations during the designing and the production of inverters and power converters utilized in the PV applications. Part 1 is concerned with general safety and Part 2 is concerned with specific requirements in inverter performance with abnormal conditions.

- **Applicability to Project:**
  The charge controller and inverter in our prototype will be used within these

standards in order to eliminate risks such as electric shock, fire hazards, over-voltage, thermal failure, and degradation of the components.

6. **IEC 62541 (OPC-UA) - Communication Protocol for Industrial Automation**

Unified Architecture (UA) OPC Unified Architecture (UA) is an industry-independent and platform-independent data transfer system. It is not directly applied to the prototype, but it is suggested to be applied in the future in terms of scalability and integration with industrial microgrid settings or smart city dashboards.

- **Applicability to Project:**
  In industrial application or large campus (Approach 2) cases, OPC-UA may be used to facilitate machine-to-machine interoperability, which supports edge analytics, SCADA interoperability, and compatibility with other existing facility infrastructure.

1.3 Systematic Overview/summary of the proposed project

The proposed project involves the design of the IoT-based centralized net meter monitoring and management system to be used with grid-connected rooftop solar photovoltaic (PV) systems and battery energy storage. The main reason why the system is needed is to make rooftop solar systems more than mere passive, compliance-driven assets, but an actively managed, grid-interactive energy resource due to real-time monitoring, intelligent control, and centralized data management.

The proposed system will combine four fundamental functional levels, i.e., energy generation and storage, sensing and measurement, control and decision-making, and communication and visualization. Rooftop solar PV modules produce electric power which is then controlled by a charge controller and saved in a battery system so as to provide backup operation and peak load control. When using a grid-tied inverter, it is possible to allow both input and output of energy as well as the reverse.

Electrical variables like voltage, current, power, and energy movement are constantly monitored at key locations of the system to identify the flow of grid import, grid export, solar generation and battery usage. It is based on these measurements that correct system monitoring and energy accounting is achievable. The state-of-charge of the battery (SOC) and availability of the backup is estimated to assist in making informed operational choices.

The central processing unit is an embedded control unit, which performs programmed control logic to operate the relay based power flow among the solar source, battery, grid, and load connected to it. Depending on the conditions of the system, the controller automatically decides on whether to consume energy locally, store it, get it in the grid or export it in the grid. This automated control can provide safe efficient and policy compliant operation without the need to use manual control.

The system uses an IoT based communication architecture to guarantee scalability and access. Web and mobile dashboards can be used to remotely visualize real-time operational data, which are being broadcast to a centralized monitoring platform. The centralized configuration enables the management and monitoring of several installations at one interface, and the system is appropriate in the residential buildings, commercial premises, and institutional installations.

On the whole, the proposed system has an integrated framework comprising of real-time visibility, intelligent control, and centralized management that help in solving the technical, operational, and regulatory issues related to rooftop solar PV implementation. The project provides a realistic basis of scalable and sustainable integration of rooftop solar in urban power systems by aligning the system functionality with national net metering policies and the emergent smart grid requirements.

1.4 Conclusion

This chapter provided the rationale, background and the principles upon which the proposed IoT-based centralized net meter monitoring and management system to connect grid-connected rooftop solar PVs installations was based. The introduction created the significance of the decentralized renewable energy in tackling the energy issues in Bangladesh, and the problem statement brought out the lack of connection between the policy goals of rooftop solar systems and the actual use of the system.

The background research and analysis of the gap in the literature indicated that the current solutions are rather concerning basic monitoring, but resilience, two-way control, scalability, and policy-conscious implementation are not discussed. To address these shortcomings, the applicability to both today and tomorrow industry focused on the necessity of smart, centralized, and digital enabled energy management systems in line with smart grid development.

Lastly, systematic knowledge of the intended project provided the integrated architecture and scope of the operations of the system, which provided a clear basis on which the further design, implementation, and validation phases would be executed. Collectively, these deliberations establish the problem space, the rationale of the given approach, and the preconditions of the detailed design methodology offered in the next chapter.

Chapter 2: Project Design Approach

2.1 Introduction

The technical, operational, and regulatory constraints have to be considered in the designing of an effective net meter monitoring and management system to be applied to grid-connected rooftop solar photovoltaic (PV) installations. As it was determined in Chapter 1, the main problem is not simply the implementation of solar systems on rooftops, but the lack of a proper monitoring, control, and integration system that would guarantee efficient use of the generated energy within the current net metering systems. The solution to this problem will require a systematic design method, which examines various system designs and then arrive at an ideal solution.

The suggested project presupposes the incorporation of distributed energy generation, battery energy storage, bidirectional interaction with the grid, and the Internet of Things (IoT)-based monitoring into one coherent system. This integration is achievable in various architectural designs with each having a different set of benefits and constraints in aspects of reliability, scalability, cost, dependency on communication and complexity in control. As a result, there can be no single design practice that can be presupposed to be the most appropriate one without a systematic consideration.

Various design strategies are discussed in this project depending on the control intelligence location, level of centralization and the reliance on cloud-based communication. These models extend to one that is entirely cloud-reliant in monitoring systems to architectures that utilize local (edge-level) decisions. All the design options will be evaluated based on the capability to enable real-time energy visibility, automated two-way power flow control, battery management, and adherence to the net metering and rooftop solar policies in Bangladesh.

This chapter aims at finding viable design strategies, outlining their structural and functional attributes, and evaluating their appropriateness against the requirements of the project in Chapter 1. By doing comparative analysis, the best design methodology is chosen to be implemented so that there can be a balance on the system performance, resilience, scale, and realistic deploy-ability of the urban rooftop solar PV systems.

2.2 Identify multiple design approach

To overcome the issues of centralized monitoring, two-way energy flow control, battery use, and regulatory compliance in the grid-connected rooftop solar photovoltaic (PV) systems, alternative system designs were considered in the conceptual design of this project. The aim of determining various design methods was to test the effects of various locations of control intelligence and communication dependency on the performance of the system, resilience, scalability, and ease of deployment.

The system suggested must be able to work in an environment with intermittent internet access, load profiles that are changing and changing net metering regulations. In these circumstances, the implementation of one strict architecture without the comparative analysis might impose some restrictions in terms of the feasibility to apply it in practice. Thus, two different ways of implementation of the design, but possible, were determined depending on the allocation of control, the location of data processing, and the access to cloud-based solutions.

Approach 1: Cloud-Based IoT Monitoring and Control Architecture

This implementation follows a centralized cloud-based implementation whereby the rooftop solar PV system serves as a data acquisition and actuation node. The voltage, current, energy flow and power are electrical parameters which are measured continuously in the local installation and sent to a remote IoT cloud platform. The cloud server is the main decision-making unit, it takes incoming data, compiles control commands, and logic on the system level.

The interaction between the user is enabled by web or mobile dashboards in which real-time visualization of the system performance is performed, such as grid import, grid export, solar generation, and battery status. Control commands e.g. relay switching decisions of energy flow control are provided on the cloud platform and implemented at the local hardware level. The focus here is on centralized control, the simplicity of remote access, and reduced hardware form design in the locals.

Approach 2: Edge-Assisted Monitoring with Local Decision-Making and Cloud Integration

The second one involves the introduction of local intelligence in which control and decision-making functions are deployed at the place of installation. This architecture uses an embedded controller to handle real-time sensor data on-the-fly and perform a set of predetermined control logic to control battery charging and discharge, grid import and export and load prioritization. The system can also operate in an autonomous manner without relying on the connectivity to the cloud on a regular basis.

In this scheme, cloud integration is not so much as a real-time which is centralized monitoring, data logging, and user visualization. It has periodical synchronization of operational data with the cloud platform and this facilitates the users and the system operators to monitor the systems performance remotely and at the same time, the systems run continuously on the local area even when there is an internet failure. This architecture is a hybrid between centralized visibility and scalability, and resilience and autonomy.

These two design solutions are opposite to each other and reflect the different ways of how an IoT-based centralized net meter monitoring and management system can be implemented. The former focuses on centralized control and ease of local processing whereas the latter lays stress on resilience in operations and distributed intelligence. The two methods were developed to be further elaborated and given a comparative analysis to produce the most appropriate solution to be implemented within the challenges that were realized in Chapter 1.

2.3 Describe multiple design approach

Approach 1: Cloud-Based IoT Monitoring System

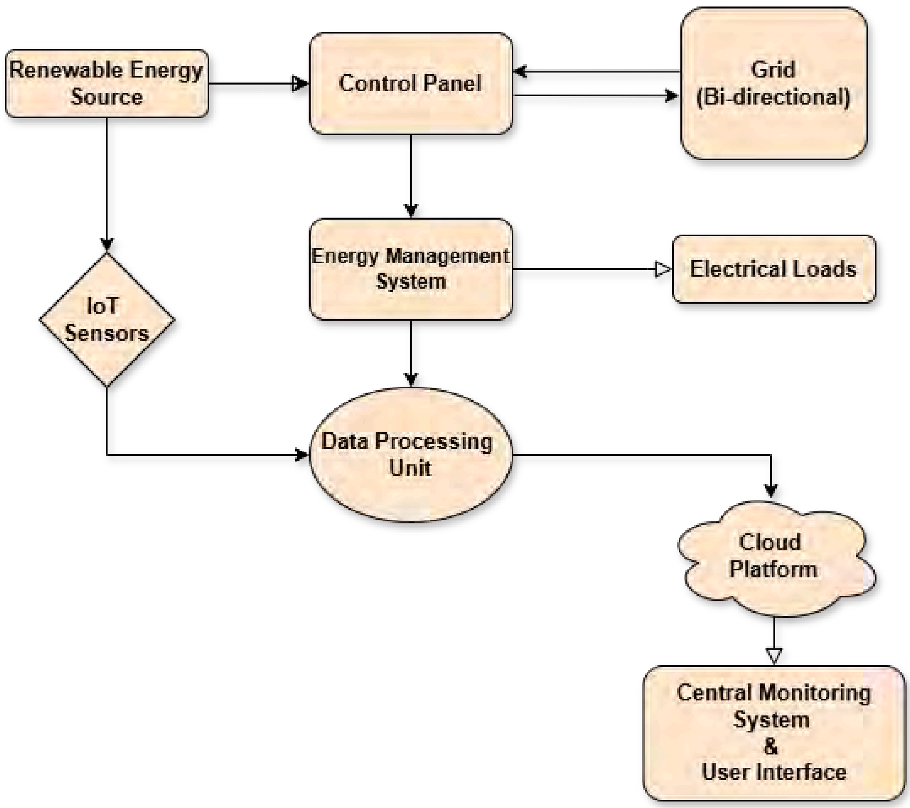

Figure 2.1: Schematic of the cloud-based IoT microgrid monitoring architecture.

The cloud-based prototype is based on the idea that every sensor node (with ESP32) periodically measures PV/battery voltage and current, environmental conditions, etc. and transmits this information via Wi-Fi to the cloud database (Blynk). The Blynk real-time database is a time-stamped measurements repository that is accessible in online dashboards or mobile applications. An Energy Management System is a central system running in the cloud, which processes incoming data to calculate metrics (e.g. net power, state of charge) and the control actions. The control commands (e.g. turn on/off loads, regulate inverter setpoint) may be pressed on the cloud to edge actuators using the same cloud channel. An example is the cloud server sends an ESP32 GPIO (physical pins on a microcontroller) or sends a command via MQTT (a lightweight communication protocol used in the internet of things to transmit data between devices) that in turn makes the local controller disconnect a load or change a relay. This is after published architectures in which sensor data is sent to a cloud server and commands sent in reverse to actuators.

The information in the cloud design will generally go through the following: Sensors → ESP32 (IoT node) → Internet (MQTT/HTTP) → Blynk (real-time DB) → User Interface/Analytics. Through the user interface (web or mobile), there is a direct connection to Blynk, and it displays live charts on power, loads, and battery status, which are provided by the capabilities of Blynk real-time sync. The protection is implemented through TLS/SSL(encryption protocols). The advantages of the cloud model are that it is easy to access remotely, and it has a strong cloud compute (e.g. machine learning analysis in future work) but it is also characterized by latency and reliance on an internet connection.

To sum up, the cloud prototype is based on the following workflow: ESP32 nodes send sensor measurements every periodically to Blynk via Wi-Fi (via MQTT/HTTP). The information is amalgamated and digested in the cloud server. End users (e.g. the operator of the microgrid) can also communicate with the system through a web/mobile application to check the status of the system and can provide commands. All big hardware selections (ESP32, Raspberry Pi, MPPT controller) are inexpensive and open-source receptive. The system will initially be simulated and tested in the lab and then on real hardware on a microgrid test-bed on a campus.

Approach 2: Edge Computing-Based Local Control System

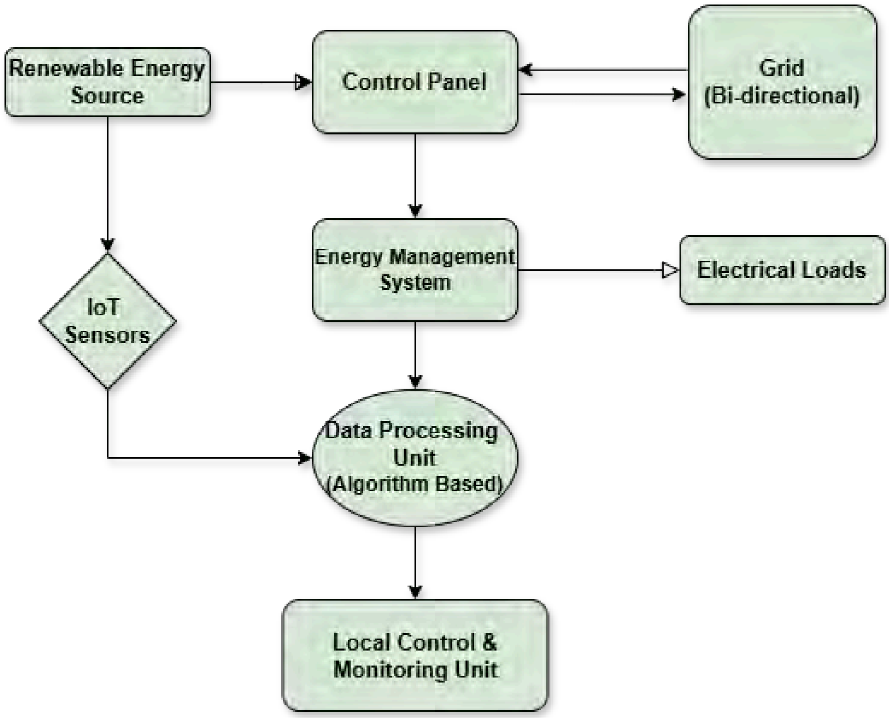

Figure 2.2 : Schematic of the edge computing-based local control architecture.

In the edge-based prototype, sensor nodes still gather PV, battery, and environmental data, but instead of sending raw data directly to the cloud, they report to a **local controller** (the Raspberry Pi) on the premises. The Raspberry Pi runs the core data processing and control algorithms. It maintains a local database or state vector of recent measurements and makes control decisions (for net-zero energy balance) in real time. A minimal summary of data (e.g. daily energy totals) may be uploaded to the cloud for backup and visualization, but primary decision-making resides at the edge.

This architecture is aligned with the edge-IoT paradigm: bringing compute closer to the sensors reduces latency and allows fast local response. As an example, in case of a critical load spike the Raspberry Pi is able to throttle loads on-the-fly, or tune an inverter output, without sending a round-trip message to the cloud. The Pi also has the ability to carry on with its operations even when there is a loss of Internet- connection - a frequent occurrence in rural Bangladesh. We will implement local control policies on the Pi (in Python/C++) that aim to match load with PV output. For instance, the Pi can switch on/off relays for non-essential loads based on battery state-of-charge or solar availability. In effect, the Pi acts as the **Energy Management System** for the microgrid.

Data flows in the edge design are: *Sensors → ESP32 (IoT node) → Local network → Raspberry Pi (edge server) → Control outputs/actuators*. The Raspberry Pi hosts a lightweight server (MQTT broker and/or HTTP REST API) that collects sensor data. Analysis and decision-making happen locally; a local dashboard (on the Pi or a connected display) visualizes current status. The Pi may periodically push aggregated data to Blynk so that remote monitoring is still possible. Edge computing in smart grids reduces how much you pay for computing, shortens the time to solve service requests, and is important for real-time IoT grids, as outlined in related literature.

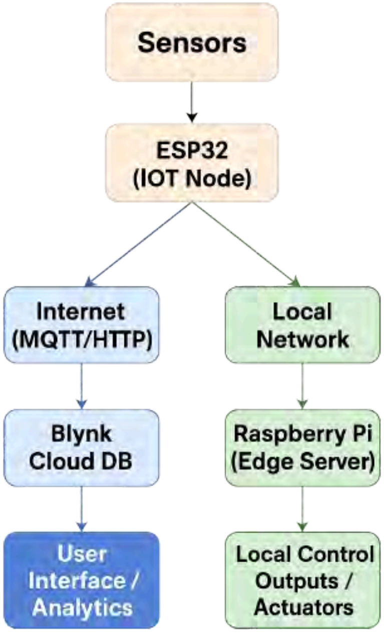

Figure 2.3: Dataflow diagram for Design approach 1 & approach 2

Compared to the cloud-only design, the edge system requires more local computing resources and software complexity, but it gains resilience and real-time performance. We will prototype this system by programming the Raspberry Pi to run the same control logic that the cloud server would, and interfacing it with the same sensors and actuators as in the cloud case. If needed, the Raspberry Pi can also run lightweight machine-learning or optimization routines offline to refine control over time. Overall, these two contrasting architectures - cloud-centric vs. edge-centric - will be systematically implemented and tested under identical conditions to ensure a fair comparison.

2.4 Analysis of multiple design approach

The project will develop prototype architectures for a solar-powered microgrid with net-zero energy management: (1) a **cloud-based IoT monitoring system** and (2) an **edge computing-based local control system**. Both prototypes will include solar PV panels, battery storage, power inverters/MPPT charge controllers, and loads connected to a bi-directional grid interface. The methodology touches upon system design, hardware/software development, data acquisition, analysis, visualization, and comparative evaluation. The design and the test of each system will be repeated until the objectives of the project are achieved by measuring the performance and fine-tuning the implementation. In the development of such systems, we capitalize on the discoveries that contemporary microgrids are based on the specialized sensors, communication systems, storage, and dynamic control mechanisms to offset intermittent renewables. We will show how each of the two, cloud and edge architecture, can be used to assist the essential workload of monitoring, control, and net-zero optimization, and which one of the two approaches is more suitable to facilitate scalable smart microgrids in the Bangladeshi setting.

System Components and Hardware Platforms

The major hardware constituents of both prototypes are:

- **Energy sources and conversion:** Solar PV panels feeding a maximum-power-point-tracking (MPPT) charge controller to charge the battery. MPPT controllers dynamically adjust load impedance to maximize PV power extraction under variable irradiance. A battery bank (e.g. lithium-ion or lead-acid) provides energy storage, and an inverter interfaces the DC system with the AC loads and grid.

- **Microcontrollers and single-board computers:** Low-cost microcontrollers (e.g. **ESP32** or ESP8266) will serve as sensor nodes and local data loggers, using their built-in Wi-Fi for connectivity. A **Raspberry Pi** (or similar SBC) is used as the local aggregator or edge server in the edge-based design. This follows proven practice in IoT microgrids, where "sensor-actuator nodes are embedded entities composed of a microcontroller, a communication device, and current/voltage sensors," and a single sink node includes a microcontroller plus a microcomputer. These off-the-shelf, open-source hardware elements keep costs low and simplify development.

- **Sensors:** Each PV panel and battery will have voltage sensors (e.g. voltage divider feeding an ADC) and current sensors (e.g. Hall-effect ACS712 modules) to measure electrical power flows. Environmental sensors (e.g. a light sensor or pyranometer for irradiance, a temperature sensor for panel/battery temperature) will also be included. The microcontrollers digitalize all sensor data and time stamp them.

- **Communication:** In the prototype provided on the cloud, the sensor nodes will be connected to the Internet through Wi-Fi (MQTT or HTTP). The prototype based on the edges can operate with Wi-Fi or even local wireless connection to transfer data to Raspberry Pi hub. Normative IoT protocols (MQTT, HTTPS, or websockets) will be applied. As an example, an ESP32 can send sensor data in the form of a JSON to a MQTT broker in the cloud or Firebase host.

- **Software and platforms:** In the case of the cloud-based system, we will rely on Blynk, which offers a user-friendly dashboard where real-time monitoring and control could be done. Blynk supports:

1. Mobile or web dashboard updates on real-time information.
2. Wi-Fi communication with microcontrollers (ESP32).
3. It will store sensor information on its own cloud storage.
4. Such controls as relay switching or passing commands to the devices via the app.
5. Safety of data management with encrypted traffic.

The sensor data will be transmitted with the help of ESP32 to either Blynk with its mobile SDK or MQTT API. The app interface allows users to see live values (such as power generation, battery level or load usage) and even control devices remotely.

The edge based system will have a lightweight database installed on the Raspberry Pi that will store sensor data locally. The data analysis, control decisions and real time responsiveness to local energy requirements will be done by custom Python scripts. A web interface or mobile dashboard will be developed in a simple form to display the data with the help of such tools as Node-RED or Grafana.

The two systems will enable us to monitor and manage the energy flow in real time either via the cloud or directly at the edge.

Data Acquisition, Analysis, and Control Strategy

In both prototypes the collection of data is continuous and time stamped. Sensor nodes will sample important signals (PV voltage/current, battery voltage/current, irradiance etc.) at rates of the order of once per second or similarly. The ADCs of the ESP32 convert the readings of analog sensors to digital values and send them in the form of JSON payloads. All data can be correlated to analyze it by time synchronization (with NTP or otherwise based on the receiving end). In the cloud case, Firebase will store data in JSON documents (one per sensor or per time slice) which can then be queried for analytics. In the edge case, data will be logged to a local time-series file or database (e.g. InfluxDB or SQLite).

Collected data will be post-processed to compute energy metrics: e.g. instantaneous power ($P=V\times I$) on PV and loads, cumulative energy in/out of the battery, and net energy balance. We will also log environmental context (solar irradiance, temperature) to correlate with PV performance. For visualization, a real-time dashboard will display trends of generation vs. consumption. For example, in the cloud prototype a web/mobile interface (leveraging Blynk's real-time sync) will plot live curves of PV output, load usage, and battery state-of-charge. In the edge prototype, similar plots can be generated on a local web server (e.g. using Grafana or a Python/Flask dashboard).

The control strategy aims to maintain *net-zero* energy: that is, over each 24-hour period, the microgrid should consume no more grid energy than it exports. To achieve this, the system will follow simple rule-based logic informed by the data analysis. For instance: if PV output exceeds local load, excess energy is stored in the battery. If load exceeds PV (and battery is low), non-critical loads are shed or delayed until PV production increases. If the battery is full, excess PV may be fed to the grid. These actions can be formulated as a priority scheme: essential loads always powered first, with discretionary loads curtailed as needed. The PV MPPT controller makes the charging efficient, whereas the code of the ESP32/RPi determines the load management. This sense, analyze, decide, actuate dynamic control loop is directly based on the microgrid control functions identified in the literature.

The trends and inefficiencies (such as whether the battery is often deep-discharged or whether there are big mismatches) will also be detected during data analysis. The analysis of historical data will be performed using Python (pandas, matplotlib) to calculate such performance indicators as the amount of renewable utilization percent, system efficiency, and reliability (uptime). The control parameters will be informed by these analytics. Google Cloud or Python scripts executing with Firebase could be used in the cloud design to apply advanced analytics (including machine learning). Analysis can also be easier in the edge design, although optimization routines can be periodically run locally. In their case, transparency and auditable algorithms are the focus to support the net-zero approach.

Comparative Evaluation Framework

The two prototypes will be compared with the help of a general assessment model. Some of the key performance criteria would be:

- **Latency and Responsiveness:** The delay of response between change of generation/load and system response (e.g. in shedding loads). Edge processing should be lower latency which we will measure (say a load step and time the control action).

- **Reliability and Resilience:** Test every system when there is loss or failure of a component in the network. Local control should remain in the edge system in case the Internet is unavailable, and in the cloud system, remote visibility would be lost. We will capture uptime, data loss events and capability to continue operations in the case of outages.

- **Data Bandwidth and Cost:** Monitor the data consumption (and costs of any cloud service) of each method. Each of the cloud prototypes will send all the raw data to Firebase (with bandwidth), whereas only the summaries will be sent through the edge prototype. We will measure the amount of data and comment on the implications of the data as regards to the connectivity costs in Bangladesh.

- **Scalability:** Think about how each design would scale to bigger microgrids including more sensors. Devices can easily be added to the cloud model because Firebase is horizontally scalable, however, this raises data flow. The edge model migrates load among more local controllers, which can also enhance scalability in a situation where each is covering a small region, but introduces complexity. We will do this qualitatively and through small-scale experiments (e.g. the addition of dummy sensor nodes).

- **Control Effectiveness:** Compare the effectiveness of each approach in keeping net-zero. Each system will be tested through a few days at the same load profile and total grid energy import/export will be measured. The time spent at net-zero (or net-export) is an important indicator that is expressed in percent.

- **Cost and Practicality:** Compare the operation cost and the cost of hardware of each system. The cloud solution needs an (free-tier)Subscription to an IoT platform and is reliant on internet penetration. The edge system requires more powerful local computer (RPi cost) but requires less cloud resources.

We will establish numerical measures or qualitative measures, in relation to each criterion. As an illustration, we will take sensor to actuator loop timestamps to calculate latency and we will compare daily energy balances. To test network outages or load spikes we can make use of automated tests. Current literature on the topic of IoT-edge-cloud systems presents the following trade-offs that we expect to face: edge computing "has high-computational capacity" near the devices, at the cost of heterogeneous management complexity. Such trade-offs we will record in comparison with each other.

All the tests will be conducted under conditions. In case it is feasible, test scripts will be used to collect data and compute the results in order to be fair. As an example, we can alternate between cloud and edge mode on the same hardware or with few modifications. Comparative report (a table or a chart) will be prepared formally to conclude which approach is more appropriate to achieve each goal.

Alignment with Bangladesh's Smart Microgrid Goals and Limitations

This study will be in line with the new development of interest in distributed renewable microgrids in Bangladesh. Bangladesh, similar to most of the developing areas, is experiencing increased energy demand, grid instability, and vast solar potential. Net-zero microgrids powered by the IoT can be used to better rural electrification and decrease the use of diesel generators. In this regard, their two approaches can be seen as complementary with the cloud solution providing a centralized monitoring solution alongside scalability in urban or well-connected locations, whereas the edge solution provides the freedom and resilience to off-grid or areas with low connectivity. Both scenarios involve cheap equipment and open platforms, which will make the solution cost-effective in Bangladesh.

Nevertheless, there are a number of shortcomings that should be admitted. First, small-scale prototyping is shown; in the real world, there may be more voltage in microgrids and hardy industrial equipment is needed. Second, cloud-based control relies on the consistent Internet that may not be accessible in rural areas in Bangladesh. On the other hand, edge-based control involves local maintenance and can not be as easily controlled by remote operators. Third, the security of the IoT is of interest: distributed IoT systems can be prone to attacks and as pointed out in previous surveys, secure means of accessing distributed nodes and data privacy is an inherent issue. We will alleviate this through encrypted communication (TLS/SSL) and authentication of each of these prototypes, but this project will not perform a comprehensive analysis of security.

Lastly, sensor information can contain noise or calibration errors and wireless connections can contain delays. We will also consider it by filtering data in software and create control hysteresis. The selection of approaches is based on the considerations of practicality: ESP32 and Raspberry Pi would enable fast prototyping with minimal time spent on the project and a wide range of community support, as it is demonstrated in the literature. Firebase was selected since it offers a real time database that is not very expensive to establish with high flexibility. Overall, the mentioned limitations notwithstanding, the suggested methodologies are quite adaptable to reach the project objectives: they bring together the academic best practices (IoT-edge-cloud continuum, MPPT to PV, sensor networks) and the practical limitations of the Bangladesh energy environment.

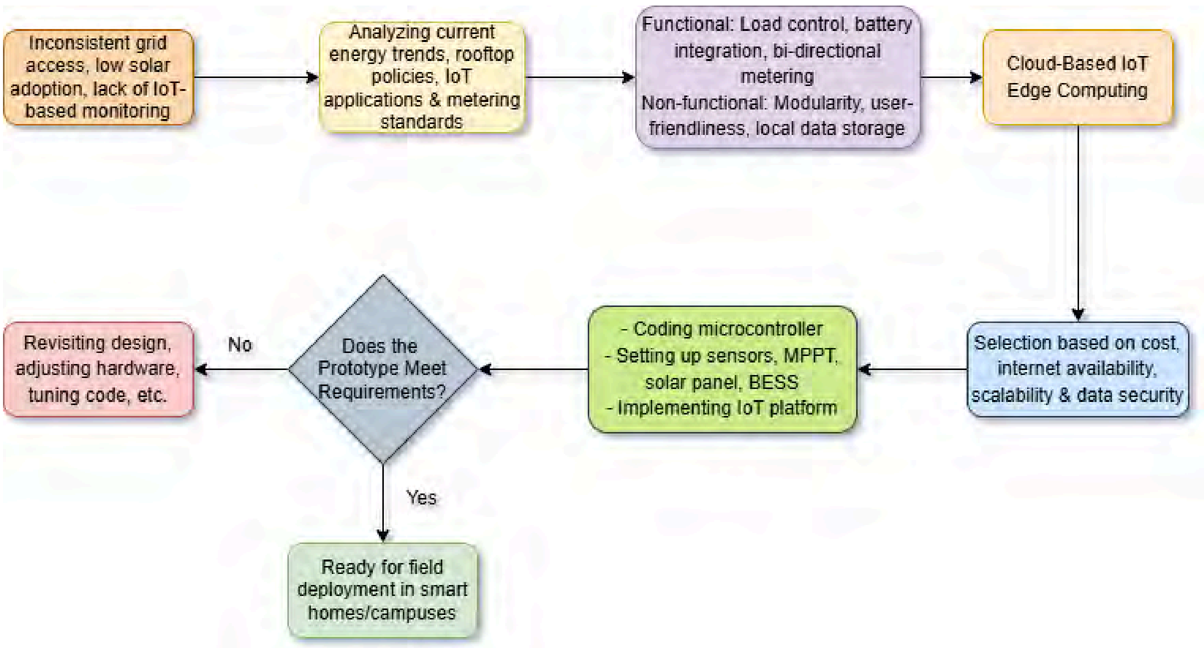

Figure 2.4 : Design Problem Flowchart

2.5 Conclusion

The chapter offered a systematic plan of creating an IoT based centralized net meter monitoring and management system to grid connected rooftop solar photovoltaic systems. Two workable system architectures were discovered and examined upon the variation in the location of control intelligence, dependency of communication, and resilience of operation.

Each of the design approaches had been outlined to point out the architectural makeup, functional coverage and working nature in the set up of the rooftop solar PV systems in the net metering set up. The comparison of these strategies creates a strong foundation of confined choices of centralized cloud-based management and edge-based local choice-making.

This methodical search preludes the chapter as it prepares the groundwork on the choice of the best design methodology that meets the technical, operational, and regulatory demands of the project. The knowledge acquired with the aid of this analysis determines the final system architecture that would be implemented, which is elaborated and justified in the following chapters.

Chapter 3: Use of Modern Engineering and IT Tool

3.1 Introduction

The centralized net meter monitoring and management system designed on an IoT based requires the efficient application of contemporary engineering and information technology (IT) tools. Because of the interdisciplinary character of proposed system, which implies the use of renewable energy systems, embedded control, data communication and intelligent decision-making, the use of analytical modeling and simulation is not enough with proper digital tools to verify the system, analyze data and visualize it.

The contemporary engineering and IT tools were used in the current project to facilitate the design, simulation, performance analysis, and comparative analysis of the suggested system architectures. These tools allowed to model the grid-connected solar photovoltaic (PV) system properly, check the functioning of the net metering in various conditions and show the concept of real-time monitoring and intelligent control. The presence of such tools ensured that the behavior of the system may be investigated prior to the implementation of the hardware, which reduced the risk of design and resulted in a higher level of reliability.

Moreover, the tools selected offered the opportunity to easily integrate the findings of the simulations with the monitoring platforms based on the IoT and machine learning environments. This facilitated the project to make a comparison of the cloud based and edge assisted design designs in a controlled and realistic fashion. The use of modern software environments and solution-based approach demonstrates through the project how modern engineering tools may be implemented to resolve the disconnect between system design theory and practice in smart energy system.

This chapter outlines the selection rationale, application, and input of each of the engineering and IT tools applied in the project and how they facilitated the successful accomplishment of precise system modeling, smart data processing, and successful visualization.

3.2 Select appropriate engineering and IT tools

The project design and simulation stage demanded assimilation of the current engineering and IT tools to model and analyze the suggested design strategies and validate them. The tools were selected to use due to the appropriateness of each one to the simulation of renewable energy systems, working with large datasets, application of machine learning analysis, and the visualization of IoT-based monitoring in real-time. The tools used and chosen were as follows:

1. MATLAB/Simulink

The main simulation tool to be used in modeling the grid-connected PV system with net metering was MATLAB Simulink. It was driven by the decision because of its progressive libraries in power electronics, renewable energy systems, and the dynamic system modeling. Simulink also allowed the real representation of the solar panel features, MPPT charge controllers, battery storage, and grid interaction. The platform allowed the project team to analyze key parameters such as PV output power, battery state of charge (SOC), grid import/export, and load demand under varying solar irradiance conditions.

MATLAB usage allowed to guarantee that the dynamics of the system could be analyzed in detail and results of the simulation could also be exported in form of structured files (CSV files) to be further processed. This is consistent with the need in the project of functional verification and is helpful in integrating with IoT dashboards.

2. Blynk IoT Platform

In Design Approach 1 (Cloud-based system), CSV data that was exported was used to create a Blynk IoT platform to illustrate real-time monitoring and visualization. Blynk was chosen as it will offer cloud connectivity, secure communication protocols (MQTT/HTTP withSSL/TLS) and provides user-friendly dashboards available in web and mobile applications. This tool is the implementation of the cloud-based workflow of the offered system and it confirms the possibility of sharing, monitoring, and controlling energy data remotely.

3. Google Colab (Machine Learning Environment)

In the case of Design Approach 2 (Edge-based system), Google Colab was adopted as the platform to design and test a Random Forest algorithm with the generated CSV data obtained after an MATLAB simulation. Colab has been chosen as it provides a free and cloud-based platform, and it is built to support Python libraries, including scikit-learn, pandas, and matplotlib. This made it easy to preprocess, train, and test machine learning models without the need to use more hardware resources. Random Forest algorithm was used to give suggestions on the best actions (load management, charging/discharging decision) to take depending on PV generation, load demand, and battery SOC.

4. Ngrok (Local Dashboard Hosting)

In Design Approach 2, the environment to simulate the edge-computing setup was to use Ngrok to run a local dashboard. This tool was able to map system data and machine learning suggestions in real-time using a safe local web interface. Ngrok was selected due to its ability to tunnel between local environments to public URLs and thus have the dashboard available without necessarily having a permanent cloud server. This coincides with the edge-based control philosophy which is such that the system can be operational when there is a limited access to the internet.

5. Supporting Tools

- **CSV Data Handling**: CSV file storage was adopted as a medium of data mobility between MATLAB simulation, Blynk dashboards, and machine learning models.
- **Python Libraries**: two libraries namely pandas and scikit-learn are used to manipulate data, to run the Random Forest model, and to have a visual representation

of the results, respectively. These libraries offered dependable and generally acknowledged frameworks of analysis of simulation outcomes.

Justification of Tool Selection

The integration of MATLAB / Simulink, Blynk, Google Colab, and Ngrok indicates the interdisciplinary character of the work, as it combines the simulation of renewable energy, its integration into the IoT, and intelligent decision-making. MATLAB was used to provide high fidelity system modeling and Blynk confirmed cloud based visualization. Google Colab was in favor of machine learning-based optimization, and Ngrok was used to simulate the local dashboard of edge computing. Collectively, these tools helped to make sure that the two design approaches are studied in a holistic fashion, and their performance would be compared in a controlled and realistic environment.

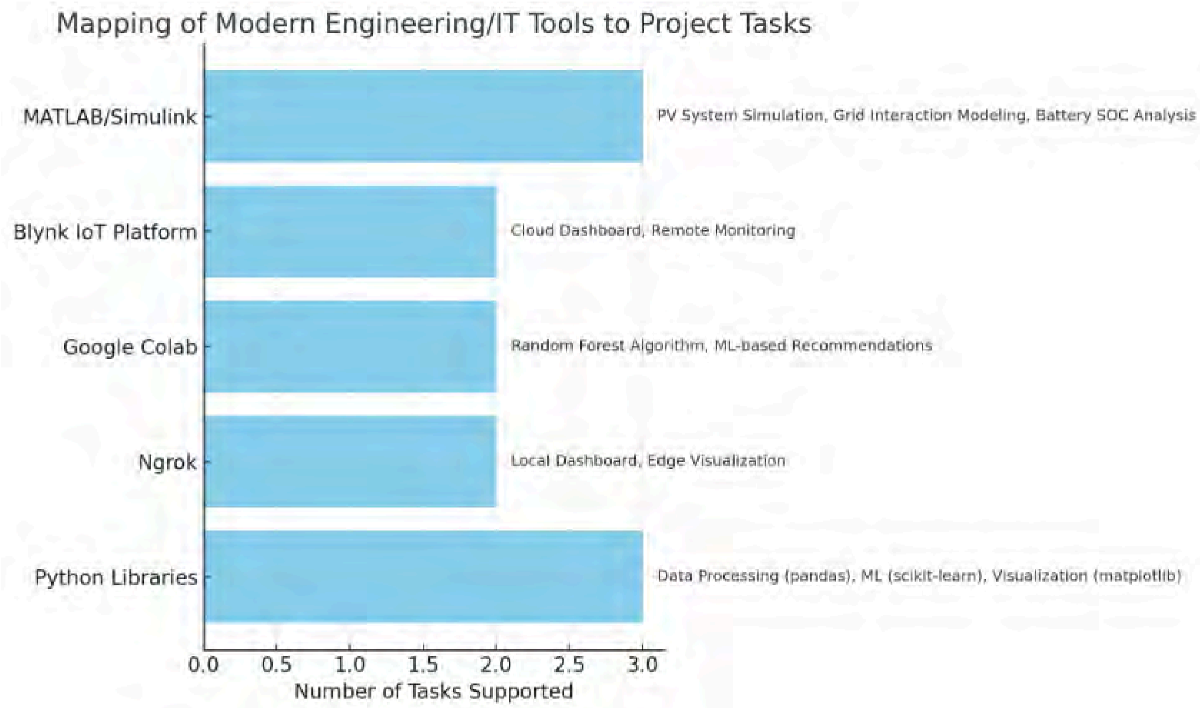

Fig. 3.1 : Mapping of modern Engineering tools.

3.3 Use of modern engineering and IT tools

IT-based tools and modern engineering were also actively employed during the design and simulating phase of the project to model the behavior of the system, analyze the performance of the system operationally, and determine the viability of the proposed methodology of net meter monitoring and management. These tools also allowed modeling the renewable energy systems, data-driven analysis, and visualization based on the IoT to be combined in a single workflow.

Use of MATLAB/Simulink for System Modeling and Simulation

The main platform upon which the modeling of the grid-connected rooftop solar photovoltaic (PV) system with net metering capability was carried out was MATLAB/Simulink. The simulation environment was designed in such a manner as to have an indication of the important system components such as solar PV modules, charge controllers, battery energy storage, inverters, loads and grid interaction. This enabled dynamic behavior of the system to be studied under different conditions of solar irradiance and load.

The operation parameters such as PV power, battery state of charge (SOC), grid import, and export power and load demand are important factors that were measured with the help of Simulink. The results of the simulation provided a good understanding of how the system worked under different conditions of operation such as surplus solar generation as well as battery aided process. The obtained output data were exported in structured CSV format to be able to further analyze them and be able to connect to other platforms.

Use of Blynk IoT Platform for Cloud-Based Monitoring

The data of the simulation output was obtained and integrated with the Blynk IoT platform in the case of the cloud-based design strategy, aimed at demonstrating the visualization and real-time monitoring. The interactive dashboards were developed using Blynk and used to show the parameters of the system parameters like energy flow, battery status, and grid interaction. These dashboards are simulations that would represent the real-time data presentation to users using a deployed IoT-based monitoring system via mobile or web interfaces.

Assessment of the cloud-centric workflow, such as data transmission, remote accessibility, and centralized monitoring, was possible with the use of Blynk. This showed that cloud-based supervision and control can be used in the net metering applications, especially where there is continuous internet connection.

Use of Google Colab for Machine Learning Analysis

In the case of the edge-assisted design approach, Google Colab was adopted as the machine learning development environment. Exported simulation data in MATLAB was imported into Colab as Python-based tools. Standard libraries were used to perform data preprocessing and feature selection and model training.

The analysis of the ties between PV generation and load demand, battery SOC and grid interaction, was carried out on a Random Forest algorithm. The recommendations about system operation, including charge or discharge, and load management were generated using the trained model, i.e., the choice of strategies. This showed the effectiveness of intelligent data-driven decision-making in improving the performance of a system rather than allowing simple rule-based control.

Use of Ngrok for Local Dashboard Hosting

In order to simulate an edge-computing system, Ngrok was used to run a local dashboard monitor. This provided the ability to visualize real-time system data and machine learning results using a locally hosted web interface which was accessible remotely by using secure tunneling. The functionality of Ngrok was used to show how monitoring and control of the system could be maintained, even without a dedicated cloud server.

This methodology favors situations in which the network connectivity is somehow limited or sporadic so that the local decision making and visualization can be handled at the same time external access is still available when it is needed.

Use of Supporting Tools and Data Handling Techniques

CSV file manipulation was taken as a middle-tier data exchange among the simulation, visualization and machine learning systems. The manipulation of data was performed with python packages like pandas, the construction of the Random Forest algorithm was done using scikit-learn, and system behavior was analyzed with the help of matplotlib. These tools offered credible and generally acceptable ways of processing and interpreting simulation data.

Altogether, the interaction of these modern engineering and IT tools allowed conducting a full assessment of both cloud-based and edge-assisted system structures, which helps to simulate accurately, conduct intelligent analysis, and visualize practically the offered net meter monitoring and management system.

3.4 Conclusion

This chapter has shown that the modern engineering and information technology tools have been used effectively to aid the design, simulation, and evaluation of the proposed net meter monitoring and management system. The use of MATLAB/Simulink facilitated the precise modeling of the grid-connected solar PV system and allowed to get quantitative information about the behavior of the system under different operating conditions. Intense combination of IoT settings and data-oriented settings enabled reasonable validation of cloud-based and edge-aided design ones.

Another example of the implementation of intelligent analysis and decentralized visualization to improve system resilience and flexibility in operation was the use of machine learning tools and local dashboard hosting. All these tools allowed full validation of system functionality to be done before implementation of the hardware, which lowers design uncertainty and provides aid in making informed engineering decisions. The results of this chapter provide a solid base to choose the best design architecture and move on to the final system implementation and validation in the further chapters.

Chapter 4: Optimization of Multiple Design and Finding the Optimal Solution

4.1 Introduction

In the previous chapters, various design methods of a centralized net meter monitoring and management system that is based on IoT were outlined and explained in detail. Although both designs are technically possible, not all designs present the same performance level when considered as a comparison to several practical parameters like reliability, scalability, and ability to withstand network volatility and the ability to deploy the design in the real world in Bangladesh. Hence an optimization and evaluation procedure is necessary and must be systematic to find out the best solution.

In this sense the term optimization is not only associated with the mathematical parameter tuning, but with the engineering optimization on the system level. This involves determining the suitability of each design strategy in meeting the requirements of Chapter 1, the functionality of the approach in relation to its ability to utilize resources provided, and the ability of the approach to work under different environmental and operational conditions. Dependency on the availability of internet connectivity, capability to conduct independent control, simplicity of integration, and long term maintainability are among factors that are deemed as vital in the selection of an ideal design.

In this chapter, the design approaches that have been identified are compared and optimized in a structured way. Both strategies are made based on set performance parameters applicable to the net metering operation, energy management efficiency and the resilience of the system. The analysis results in the optimal design approach that is the most optimal in terms of balancing between performance, reliability, scalability, and the implementation feasibility. Lastly, the chosen solution is reviewed to ensure that it fulfills the general project objectives and can proceed to hardware implementation and verification.

4.2 Optimization of multiple design approach

The PV generation and grid connection simulation in MATLAB simulink is also a basic element that is common to both designs. Consequently, this general methodology is addressed in the initial position and then Design Approach 1 and Design Approach 2 are described separately.

Introduction to the System Simulation

This project models a three-stage grid-connected solar PV system that has battery storage in MATLAB/Simulink. It is a realistic microgrid of renewable energy, comprising of solar generation and MPPT, a two-way battery storage, a three-phase inverter and grid synchronization, and variable load profiles. The simulation will be used to test the performance of the system in varying irradiance, the battery state of charge and loads, illustrating that the power is effectively used among the PV array, the battery, and the utility grid. Using the maximum solar is one of the critical results, and so is a DC bus stability, grid compliance, and intelligent energy management.

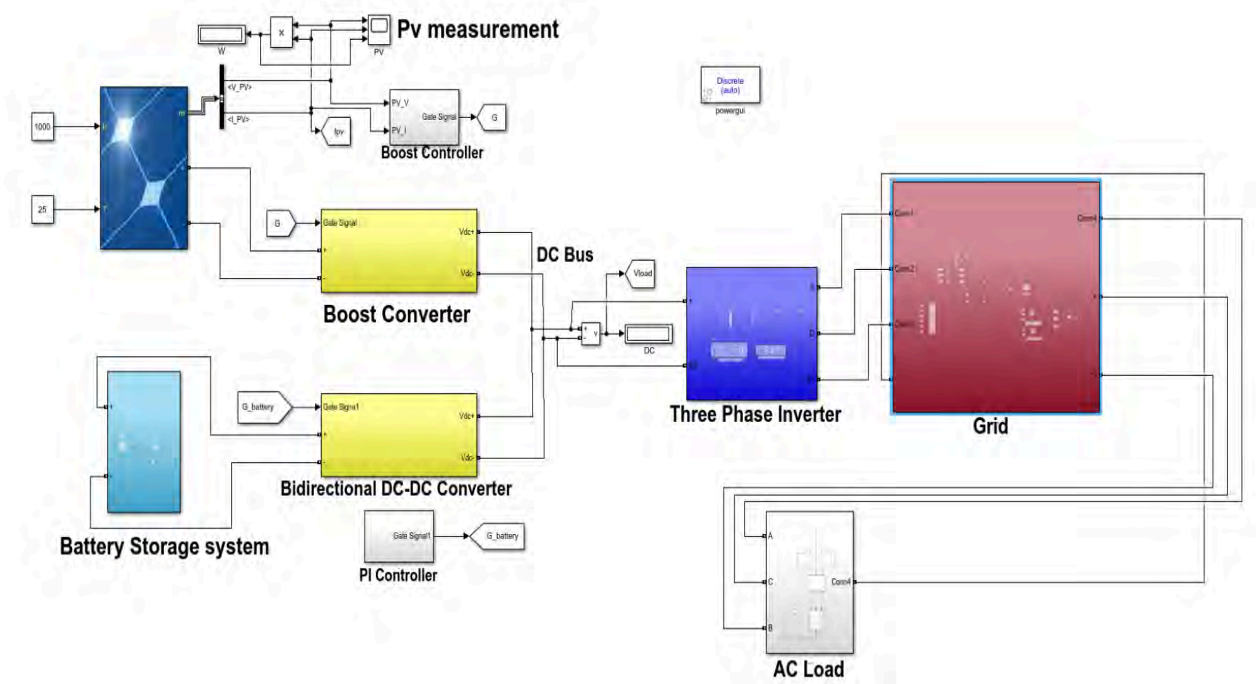

Fig. 4.1. Whole System with PV, Battery and Grid

1. Boost Converter (PV to DC Bus)

This boost converter increases the PV array voltage to keep the DC bus at 700 V. The capacitor (C) and the inductor (L) values are determined depending on the ripple requirements and switching frequency.

Required Equations:

1. Maximum Output Current

$$I_{out,\max} = \frac{P}{V_{out}} \quad \text{..................(i)}$$

2. Indicator Ripple Current

$$\Delta I_L = 0.01\, I_{out,\max} \left(\frac{V_{out}}{V_{in}} \right) \quad \text{..................................(ii)}$$

3. Output Voltage Ripple

$$\Delta V_{out} = 0.01\, V_{out} \quad \text{..(iii)}$$

4. Inductor

$$L = \frac{V_{in}\left(V_{out} - V_{in}\right)}{\Delta I_L \, f_s \, V_{out}} \quad \text{...(iv)}$$

5. Capacitor

$$C = \frac{I_{out,\max}\left(1 - \frac{V_{in}}{V_{out}}\right)}{f_s \, \Delta V_{out}} \quad \text{...(v)}$$

6. Load Resistance

$$R = \frac{V_{out}}{I_{out,\max}} \quad \text{...(vi)}$$

7. Duty Cycle

$$d = \frac{V_{out} - V_{in}}{V_{out}} \quad \text{...(vi)}$$

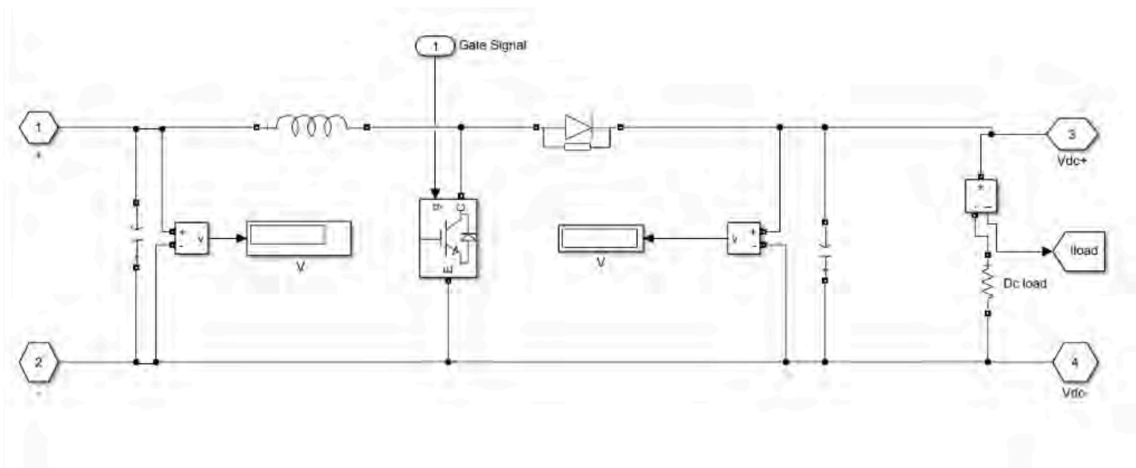

Fig. 4.2. Subcircuit of boost converter

The duty cycle of the converter is controlled by a Maximum Power Point Tracking (MPPT) controller based on the Incremental Conductance (IncCond) approach and makes sure a PV functions at its optimal operating point.

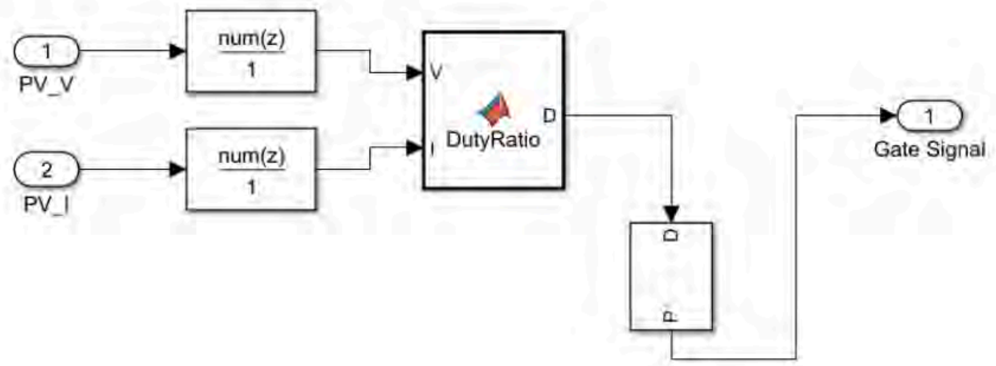

Fig. 4.3. Subcircuit of MPPT charge controller

2. Bidirectional DC-DC Converter

The battery pack is a 35 series lithium-ion battery with a nominal voltage of 48 V. The battery is connected to the DC bus by a bidirectional DC-DC converter that allows the converter to operate in two directions:

- **Charging:** Excessive power of the PV is sent to the battery during periods of high load demand at the converter.
- **Discharging:** In case of the PV generation is insufficient, the converter fills the DC bus with battery power.

Required Equations :

1. Maximum output current:

$$I_{out,\max} = \frac{P}{V_{out}} \quad \text{.......(viii)}$$

2. Inductor ripple current:

$$\Delta I_L = 0.01\, I_{out,\max} \frac{V_{out}}{V_{in}} \quad \text{.......(ix)}$$

3. Output voltage ripple:

$$\Delta V_{out} = 0.01\, V_{out} \quad \text{.......(x)}$$

4. Inductor value:

$$L = \frac{V_{in}(V_{out} - V_{in})}{\Delta I_L \, f_s \, V_{out}} \quad \text{..........(xi)}$$

5. Capacitor value:

$$C = \frac{I_{out,\max}\left(1 - \frac{V_{in}}{V_{out}}\right)}{f_s \, \Delta V_{out}} \quad \text{..........(xii)}$$

6. Load resistance:

$$R = \frac{V_{out}}{I_{out,\max}} \quad \text{..........(xiii)}$$

7. Duty cycle:

$$d = \frac{V_{out} - V_{in}}{V_{out}} \quad \text{..........(xiv)}$$

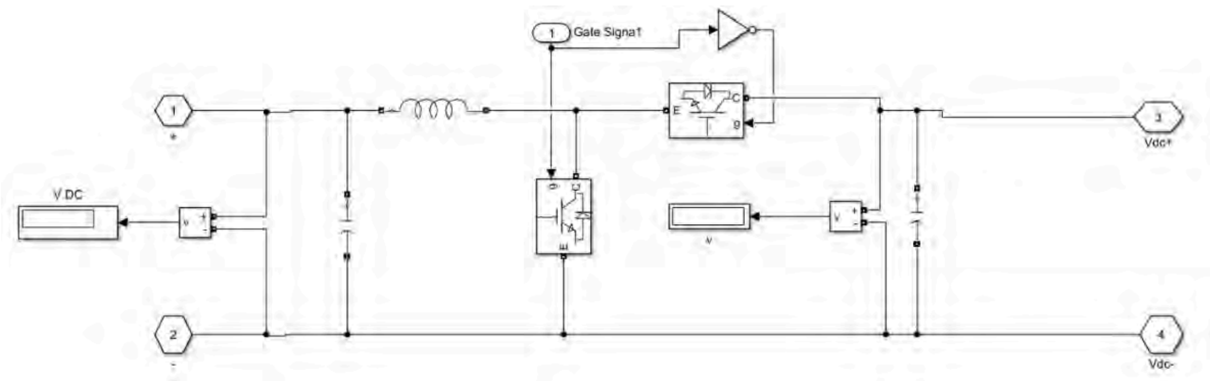

Fig. 4.4. Subcircuit of bidirectional DC-DC converter.

A PI controller is used to maintain the DC bus voltage at 700 V by controlling the current reference in the converter.

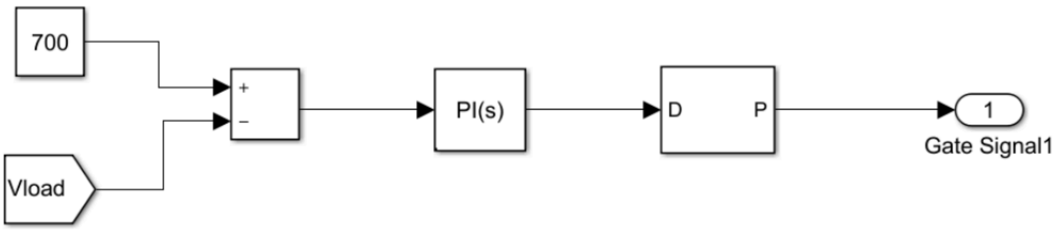

Fig. 4.5. PI controller.

3. Three-Phase Inverter

The three phase inverter will transform the regulated 700 V DC bus voltage into a three phase AC output that is completely synchronized with the utility grid. In order to obtain a high degree of control, the inverter utilizes the Sinusoidal Pulse Width Modulation (SPWM) algorithm alongside a phase-locked loop (PLL), the conversion between the abc-dq coordinates and a Phase-locked Loop (PLL). The transformation of the three-phase signals into a rotating reference frame of the three-phase AC currents using the abc-dq transformation simplifies the control of the AC currents, and the PLL provides a correct phase and frequency synchronization between the grid voltage and the AC currents. These methods combined allow the inverter to dynamically inject active power into the grid with the PV-battery system, support the quality of power and is responsive to changes in loads or generated power, and therefore integrate the grid reliably and efficiently.

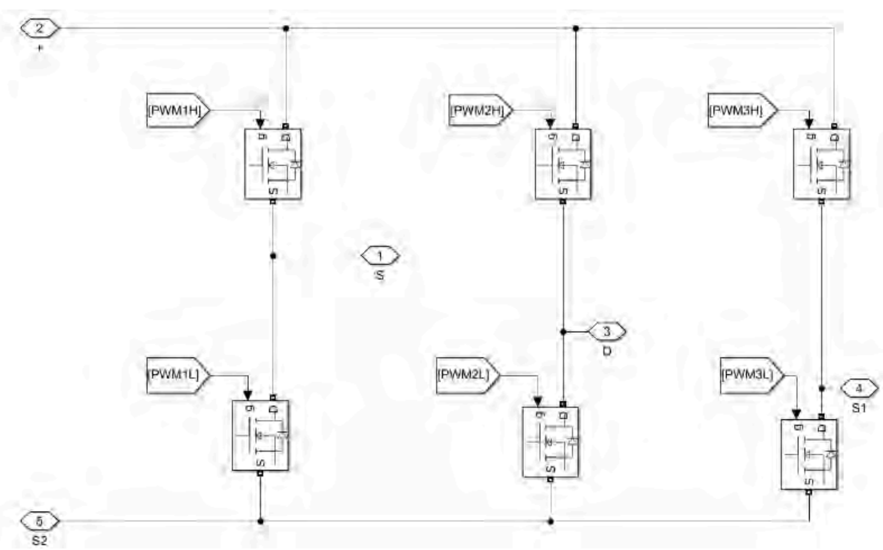

Fig. 4.6. Subcircuit of three phase inverter.

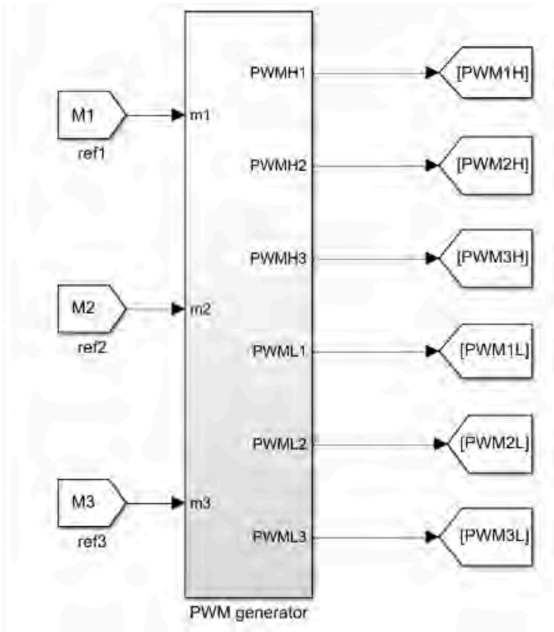

Fig. 4.7. PWM controller of inverter.

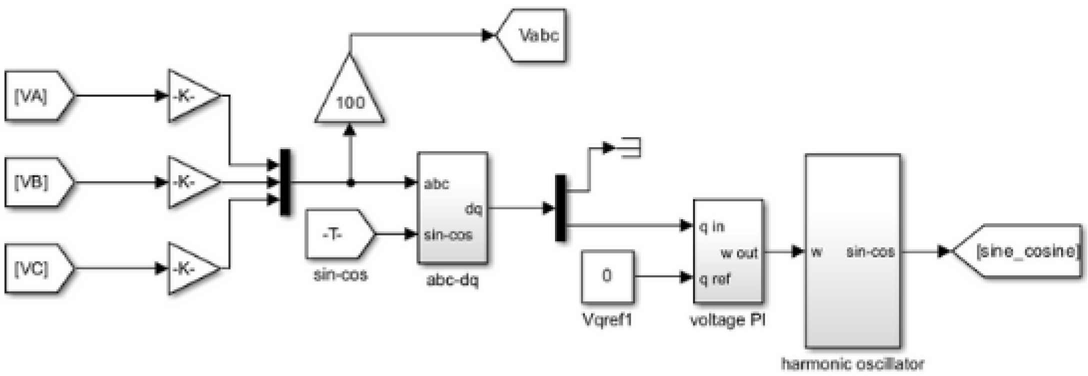

Fig. 4.8. PLL using DQ frame transformation for Sine Cosine ref generation.

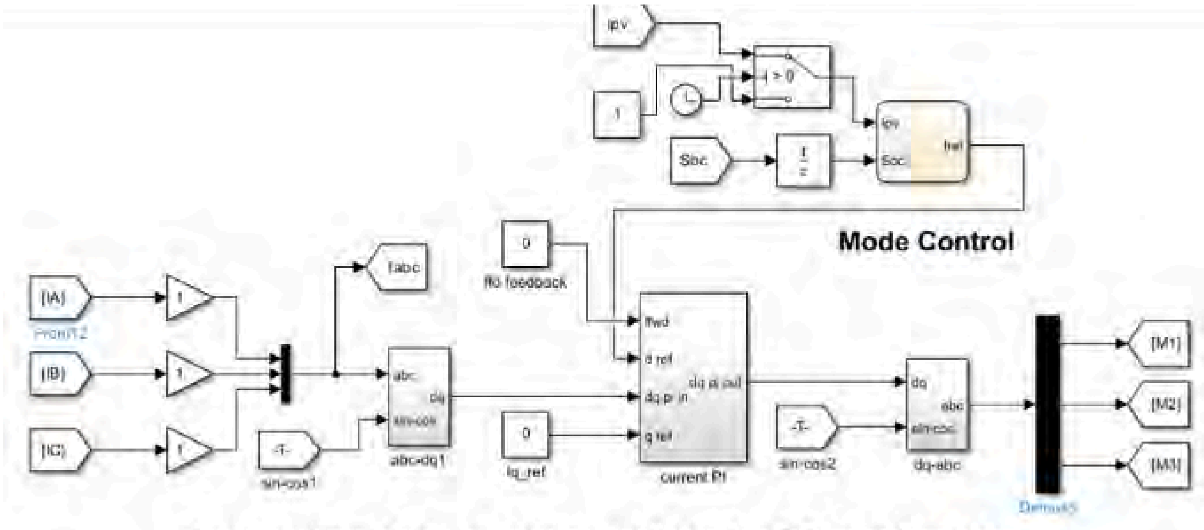

Fig. 4.9. Inverter three phase current controller using DQ transformation.

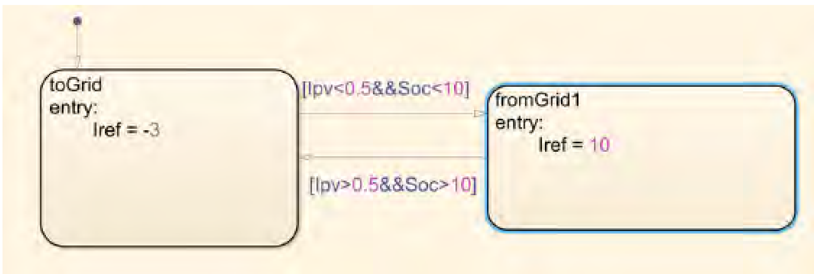

Fig. 4.10. Control unit.

4. Grid Connection

The system connects to a 400 V three-phase grid. Depending on PV generation, battery SOC, and load demand, power flow can be:

- Export: PV surplus is sent to the grid,
- Import: Grid supplies power when PV and battery are insufficient,
- Balanced Mode: PV matches load demand, grid exchange ≈ 0.

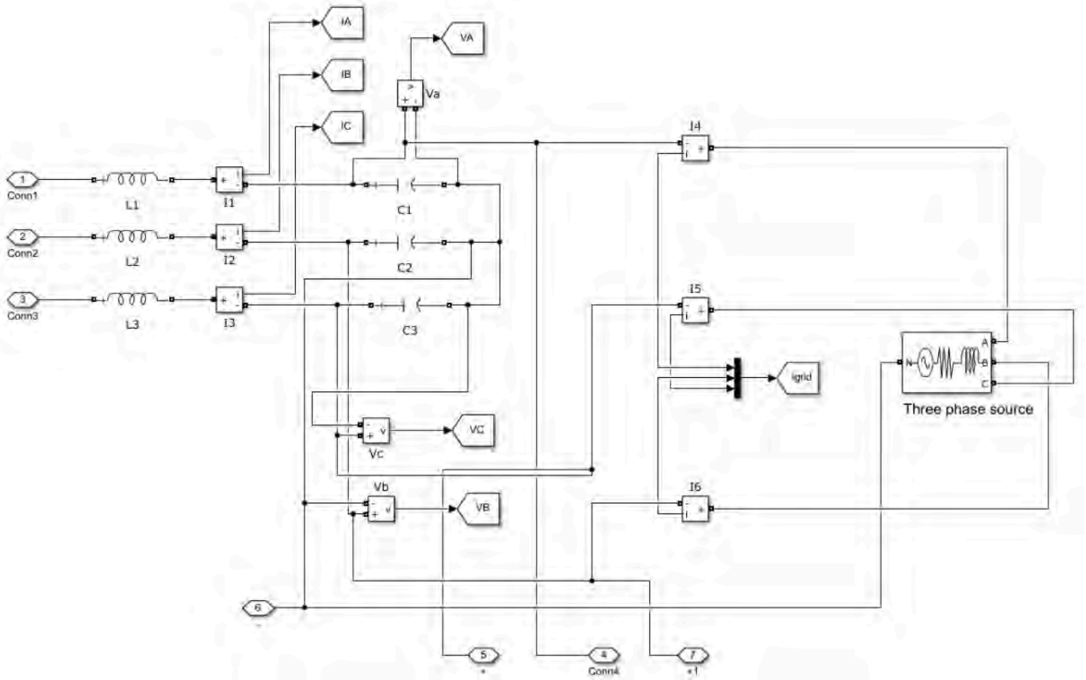

Fig. 4.11. Subcircuit of three phase grid.

5. LC Filter

An **LC filter** is connected at the inverter output to eliminate high-frequency switching harmonics, ensuring the delivered current and voltage waveforms are sinusoidal. Filter sizing is based on switching frequency and ripple attenuation requirements.

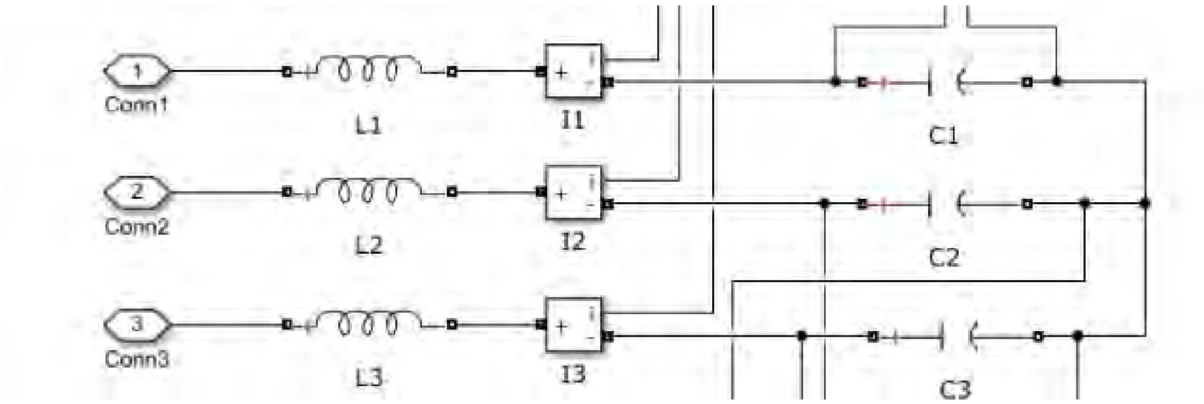

Fig. 4.12. LC filter

Required Equations :

1. Inductor Current:

$$I_L = \frac{P}{V_{dc}} \quad \text{...(xv)}$$

2. Maximum Ripple Current:

$$\Delta I_{\max} = 0.02\, I_L \qquad \text{(xvi)}$$

3. Inductor Value:

$$L = \frac{V_{dc}}{4\,\Delta I_{\max}\, f_{sw}} \qquad \text{(xvii)}$$

4. Capacitor Value:

$$C = \frac{1}{L}\left(\frac{10}{2\pi f_{sw}}\right)^2 \qquad \text{(xviii)}$$

6. Measurement and Monitoring Subsystem

Voltage, current, and power sensors are placed across all major blocks (PV, battery, load, inverter, and grid) to capture real-time performance. These measurements are used both for feedback control and for visualization of system operation in scopes/graphs.

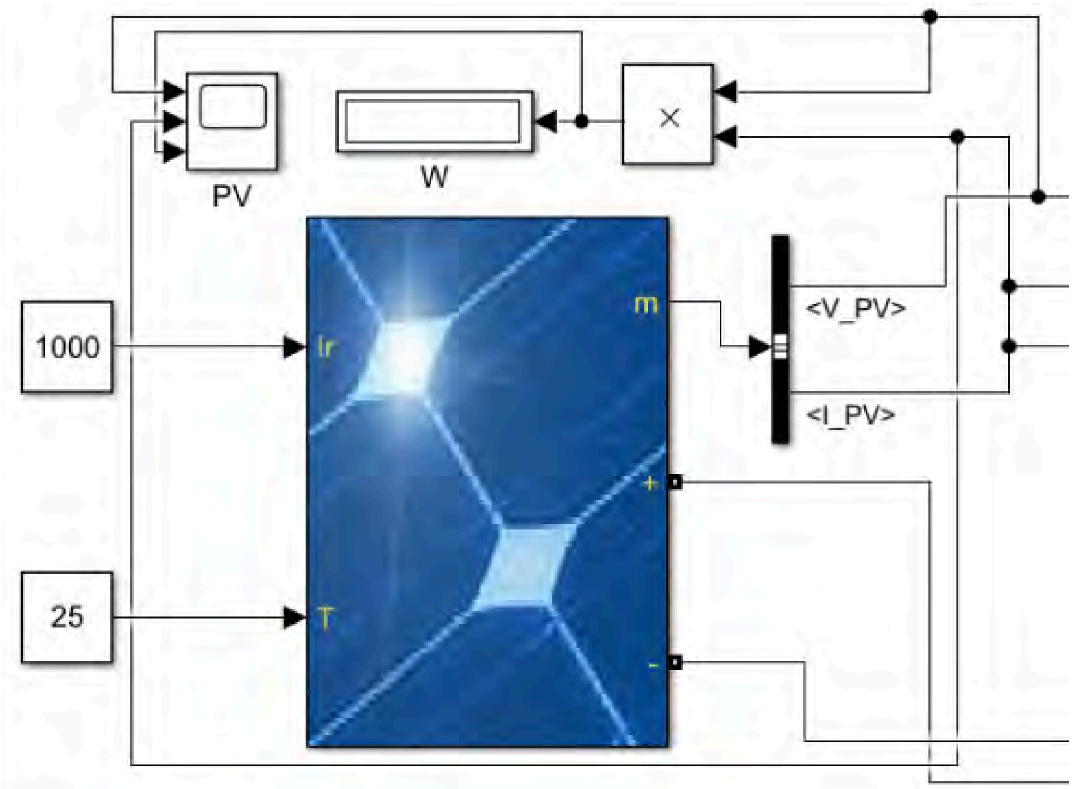

Fig. 4.13. Measurement and Monitoring Subsystem.

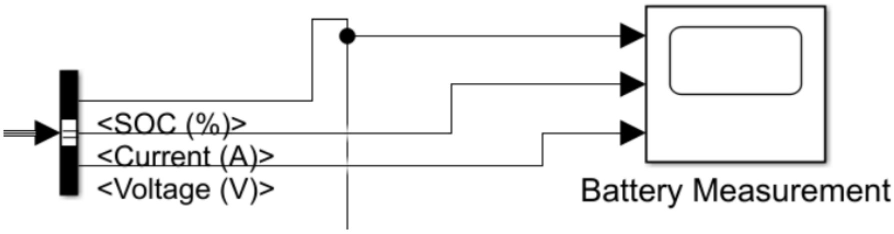

Fig. 4.14. Battery Measurement scope

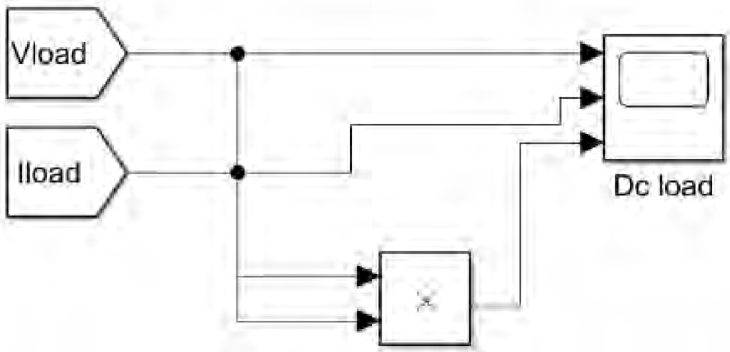

Fig. 4.15. DC Load measurement scope.

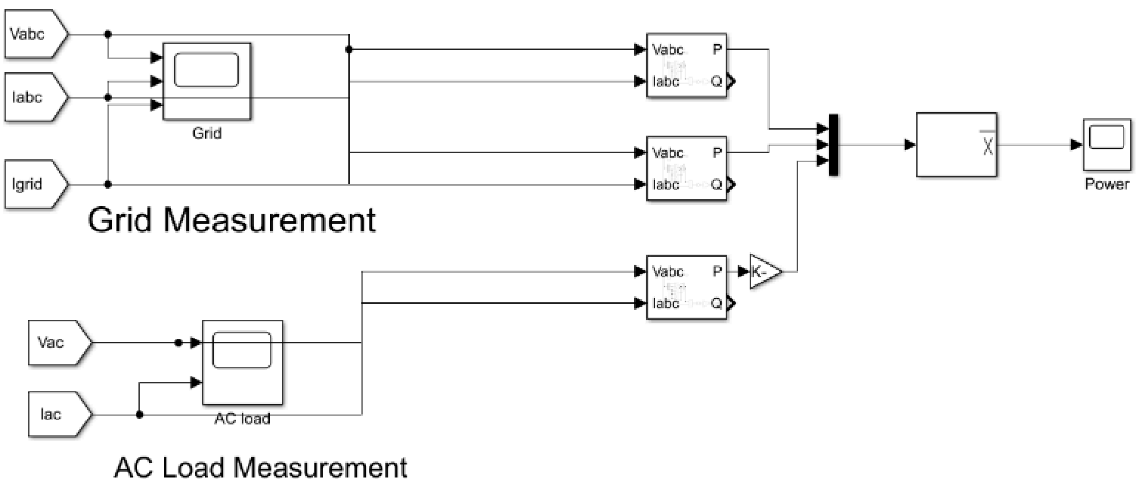

Fig. 4.16. Power measurement scope.

Design Approach 1: Cloud-Based IoT Monitoring System

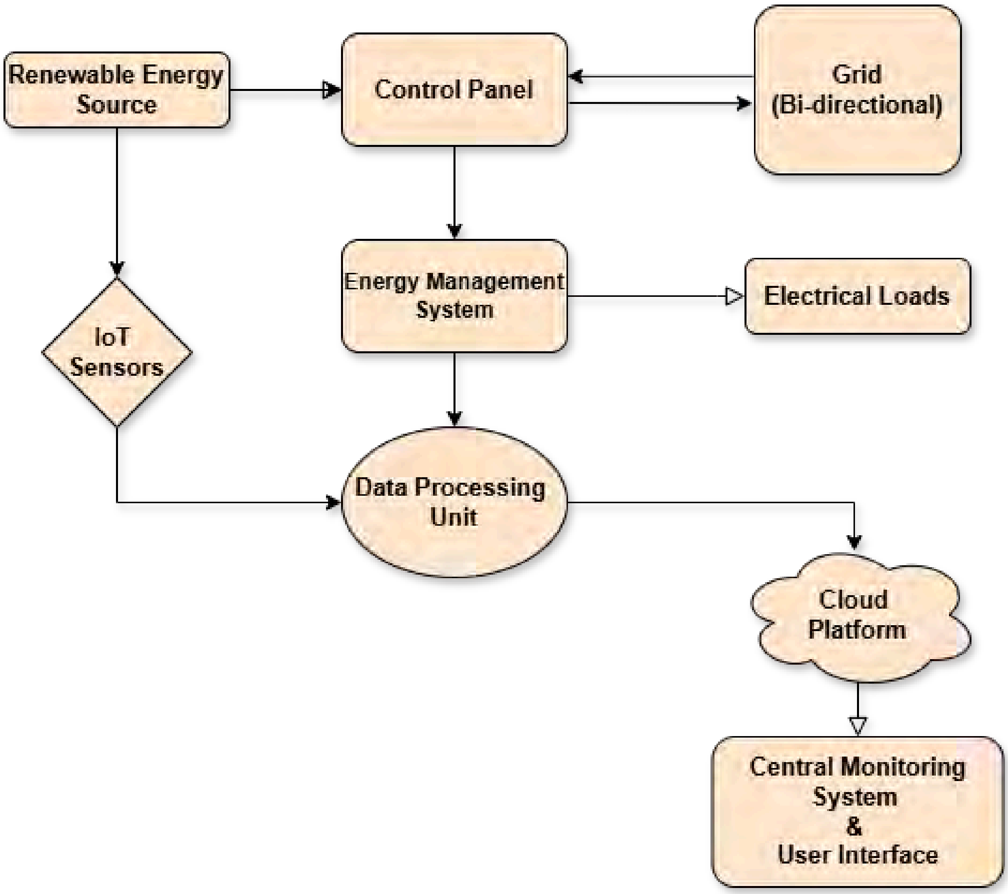

Fig. 4.17. Flowchart of Design approach 1

Design Process and Rationale

The former design implements the cloud-centric design of real-time monitoring and partial control of the solar-powered microgrid. The solution puts strong emphasis on remote access, scalability, and centralized processing and, therefore, it is adapted to urban or semi-urban regions where the internet connection is comparatively consistent.

Its design is grounded on the implementation of low-cost IoT nodes into the system to continuously monitor electrical and environmental parameters. These sensor nodes use their internal Wi-Fi capabilities to send the data to the Blynk IoT cloud platform where it is stored, processed and visualized. It is also possible to remotely give control commands (e.g., load switching, inverter regulation) via the same cloud channel, and this guarantees a two-way communication path.

This solution aligns with the project goals of net-zero energy management since it allows tracking the power generation, consumption, and storage constantly, and the user can use the dashboards to obtain practical insights.

Preliminary Design and System Flow

The cloud design has the following data flow:

1. **Sensors:** There are Voltage, current, irradiance and temperature sensors that measure real time values of PV panels, battery and loads.
2. **IoT Node:** Gathers sensor data, converts it to digital and packs it in JSON packets.
3. **Communication Layer:** It is MQTT/ HTTP with Wi-Fi serving as a means to transmit the measurements to the cloud.
4. **Cloud Platform (Blynk):** Sensors data is received, stored and structured. Offers APIs to real time visualization and remote control.
5. **User Dashboard:** Can be accessed through either mobile or web, and shows key performance indicators (PV output, battery SOC, grid interaction).
6. **Actuators:** Commands are sent back to the **IoT Node** to activate relays or switching devices for load control or inverter regulation.

Blynk IoT Dashboard - design, build steps, and section-by-section explanation

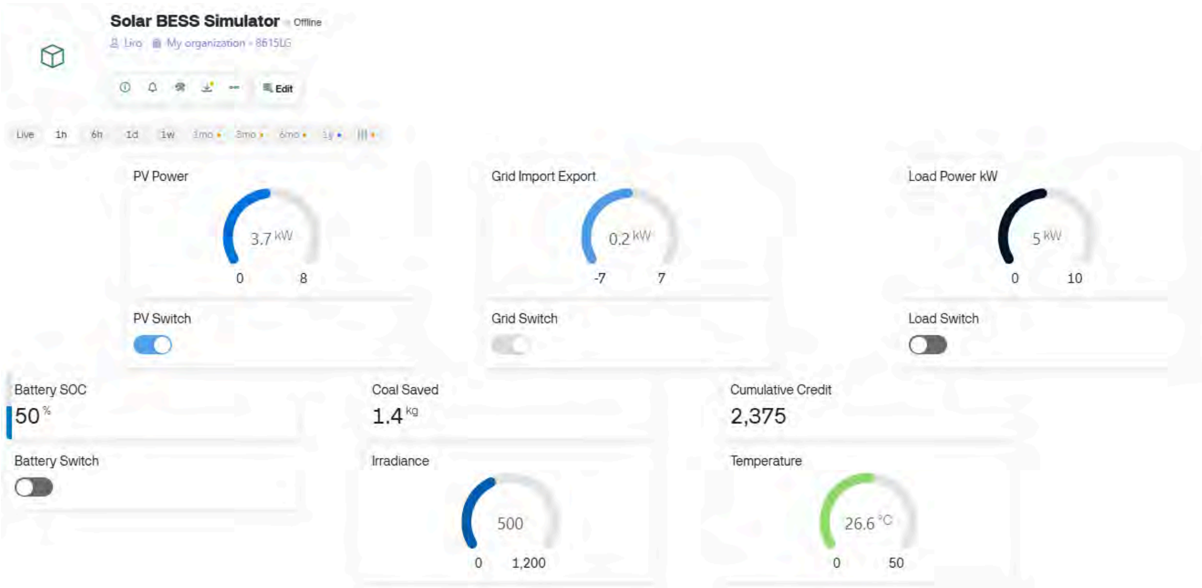

Fig. 4.18. Blynk Dashboard.

1. Design goals & Constraints

The main objective behind building the IoT dashboard was to provide a clear and reliable way to monitor the performance of our PV-based hybrid system in real time. We wanted to capture the most important performance indicators such as PV generation, grid interaction, battery state of charge, and load demand, and present them in a way that would be both visually appealing and easy to interpret. Another key objective was to ensure that the dashboard not only displayed the values but also allowed remote control of critical components such as the PV switch, battery switch, and load or grid switches.

At the same time, we had to work within a set of constraints. As far as the project is made under the free version of Blynk, the amount of the widgets and the frequency of data update are limited. The solution to this was to simply pick only the most useful KPIs and not attempt to stream raw or redundant data. The hardware constraints were also taken into consideration, as the system is based on ESP32 as an IoT node and, in certain scenarios, Raspberry Pi as an edge aggregator. This involved maintaining the design in a lightweight, bandwidth-saving and resilient manner to the intermittent internet connectivity.

2. Step-by-step: how we created the dashboard in Blynk

The initial stage in making the dashboard was to enter a new project into the Blynk IoT console. We have developed a new device profile, chosen the ESP32 as the main IoT node, and Wi-Fi as a connection system. After the project was developed, Blynk created a special authentication token, and it was subsequently utilized in the firmware to create safe communication between the hardware and the dashboard.

The second step was to create the dashboard design with the help of the widgets that would reflect the physical quantities in the most appropriate way that we wanted to monitor. In the case of PV power, power importation and exportation in grids, and load power, we used a circular gauge to provide a feeling of magnitude. In the case of such parameters as battery SOC, coal saved, cumulative credit, and temperature, we employed value displays that had the right units. Irradiance was also shown on a gauge, which has a direct relationship with the PV power output. Along with the monitoring widgets, we integrated such control elements as a PV, grid, battery, and load toggle switch.

Thereafter, the widgets were connected to a virtual pin. These pins acted as communication links between the application and the Blynk application. To transmit the real-time values to the appropriate virtual pins, the firmware, which is written to be run on the ESP32, was then programmed. The rate of data updates was done with a lot of consideration within the context of responsiveness versus bandwidth efficiency. The parameters like PV or load power were changed at the rate of several seconds, whereas such slow KPIs as energy yield and revenue were changed after every minute.

As well so as to make sure that the dashboard was not limited to monitor only, we set control functionality. As an example, the PV switch could be switched on or off in the case of testing the PV contribution, whereas the grid switch enabled us to create an effect of islanding. Alerts and notices were subsequently configured to alert the operator in instances where there was a critical condition like low SOC or deviant grid conditions. Then the system was tested using simulated data provided by Simulink to make sure that the widgets showed the right values and the control buttons managed to provide the desired actions.

3. Dashboard: section-by-section explanation

The key performance indicators at the top of the dashboard, including yield today, revenue today, and inverter rated power give a short glimpse of the entire system. The values can be used to quickly see the amount of energy generated by the system, financial return achieved in the day, and the proximity of the operation relative to the rated capacity of the system.

The PV power gauge displays the immediate power of the PV array. A switch is also provided with it so that the PV contribution can be enabled or disabled to test or simulate. The grid import/export gauge gives us an idea of whether the system is being fed by the grid or it is giving back the excess power to the grid. The grid switch has the capability of disconnecting the grid when required to a state of simulation of an islanding scenario.

The load power gauge displays the amount of energy the loads which are connected are consuming. A load switch enables the end user to switch off or switch on non-critical loads which makes it an interesting tool to test demand-side management strategies. The battery SOC display shows the percentage-based estimated level of charge of the battery in real-time and gives the operators a real-time feeling of the available storage capacity. This is combined with a battery switch whereby the battery could be switched on or off when testing.

Below are the environmental and secondary KPIs. Calculation of coal saved is made upon the production of PV energy and provides a reflection of the benefit to the environment. Cumulative credit shows the accrued amount of financial or net metering benefit over a period of time whereas the irradiance gauge shows the amount of solar radiation reaching the PV system directly. Lastly is the temperature gauge that shows ambient or battery temperature that is important in ensuring safe and reliable working condition.

Trend charts on energy and revenue are also placed on the dashboard. The charts are a graphical representation of the performance with the pass of time that makes the user identify any patterns like the high generation hours or an anomaly in the performance. The operators are able to assess daily and seasonal performance trends by viewing these charts.

4. How the dashboard is beneficial (practical outcomes)

The IoT dashboard is an efficient and viable method of monitoring and controlling hybrid PV system. The presence of all the KPIs allows the operator to make informed choices regarding the time to trust the grid, the time to trust the battery, and the time to sell energy. The control switches are flexible as they enable the system to be remotely reconfigured to suit other situations, especially during the testing and demonstrations.

The indicators related to environment (coal saved and CO_2 reduction) can be used to present the benefits of the system when it comes to the environmental issues and this can be applied to the stakeholders. The financial indicators, e.g., the cumulative credit and revenue today, are used to connect the technical work of the system and its financial results. The dashboard enables the monitoring, control, and reporting, which positively affect technical performance, and additionally, the environmental and financial gains are more visible, which makes it a comprehensive tool of operators and decision-makers.

Strengths of the Approach

- Simple to install using cheap equipment.
- Monitoring that can be done remotely.
- Scalable to larger systems having more nodes.
- Live monitoring using the mobile/web application of Blynk.

Limitations within Constraints

- Accountable on a fixed internet connection.
- Prone to network failure in rural settings.

Design Approach 2: Edge-Based Local Control with Random Forest

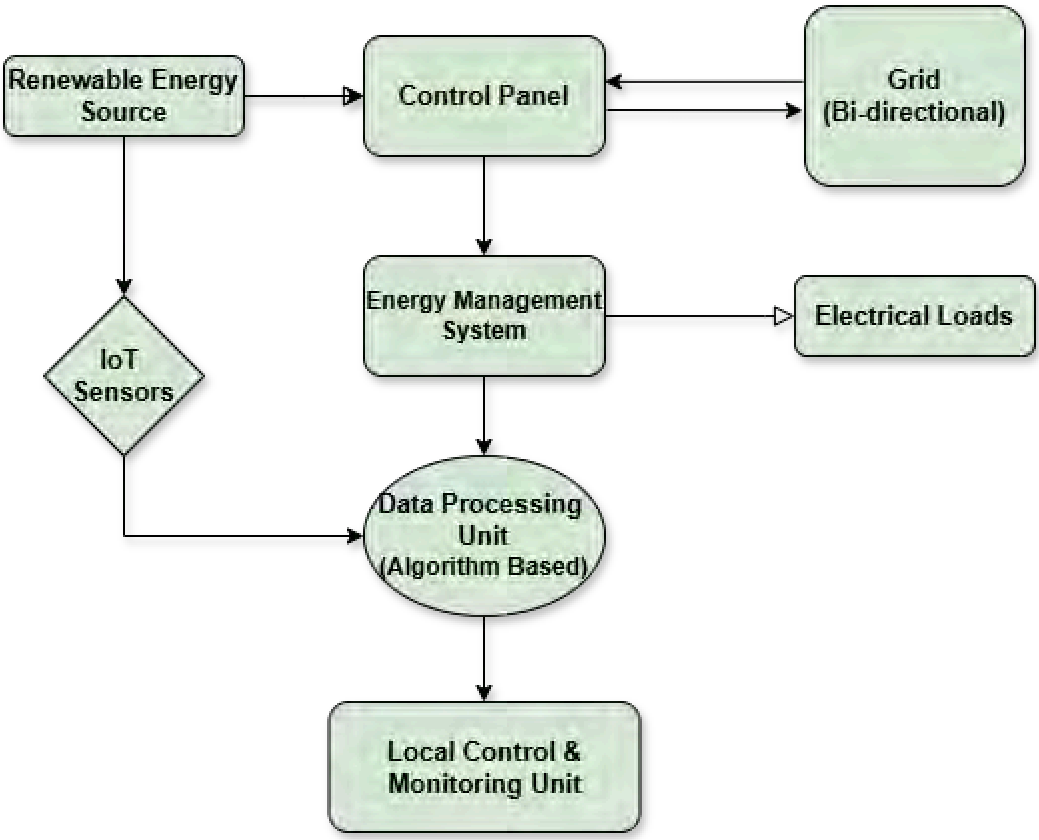

Fig.4.19. Flowchart of design approach 2

This other design creates a local edge-based design instead of cloud-based system of intelligence and visualization. The main reason as to why this design was chosen is because the PV microgrid monitoring system should be capable of working even when the internet quality is poor, as well as delving into the application of machine learning in improving decision-making. This method combines MATLAB/Simulink model, a data-driven approach that involves a Random Forest Classifier in Google Colab, and a real-time dashboard written in Streamlit and publicized with Ngrok.

1. Design Process

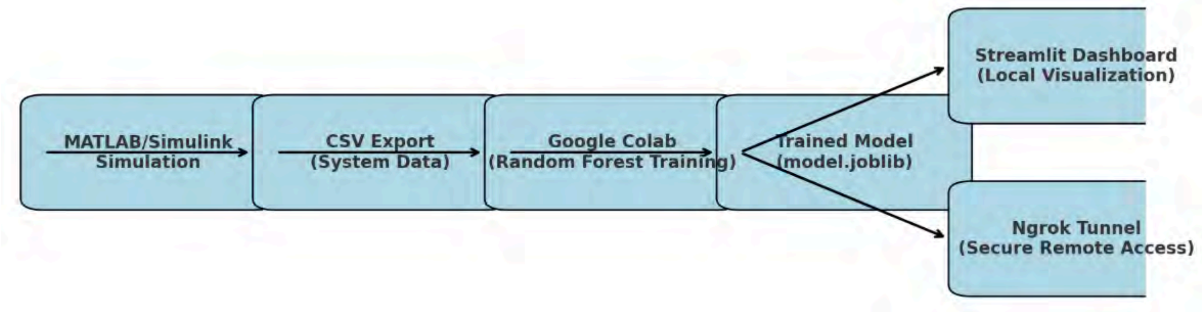

Fig. 4.20. Design process of design approach 2

The design procedure of this approach can be split into three key statements:

1. **Data Generation through Simulation (MATLAB/Simulink):**
   - A simulated PV system was a grid-based system that included solar panels, battery storage and variable load demand.
   - The major parameters that were measured included PV output power, load power, battery SOC, irradiance and temperature.
   - The CSV file of the simulation results was used as a dataset of machine learning.
2. **Machine Learning Integration (Google Colab - Random Forest):**
   - The CSV data were loaded into Google Colab using Python libraries (pandas, scikit-learn, joblib and plotly, streamlit).
   - The features used to train a Random Forest Classifier included Hour, Irradiance, Temperature, PV power, Load power and Battery SOC.
   - ActionCode is the target variable, which coded six possible operational states of the system:
     - 0: Normal (PV meets load directly)
     - 1: Charge battery

        - 2: Shift non-essential load
        - 3: Discharge battery
        - 4: Import from grid
        - 5: Export to grid

    - The data was divided into 80 percent training and 20 percent testing, which would guarantee that the model is generalized.
    - The model parameters were set: nestimators=200 (trees) and max-depth=12 which is a compromise between accuracy and computation efficiency.

3. **Visualization and Dashboard (Streamlit + Ngrok):**

    - In Streamlit, a dashboard was created and the dataset was read through a row by row iteration to simulate data flowing in real time.
    - The RF model made a prediction of the best action at every step and presented it to the user in the form of a recommendation.
    - KPIs like PV power, load, battery SOC and net grid power were shown in association with interactive gauges and time-series plots with Plotly.
    - An analysis section was also provided to present the distribution of suggested action as well as ranking of feature importance as provided by the Random Forest model.
    - Lastly, Ngrok was also set up to make the local Streamlit server accessible via a secure public URL, which would enable the controlled access to the dashboard.

2. Explanation of the Random Forest Algorithm

The Random Forest algorithm is the ensemble method of learning, which creates many decision trees in the course of training and provides the prediction as the most common one. Training each tree is done on a random sample of the data and on a random sample of the features so that there is variety amongst trees. Such randomness minimizes overfitting and makes it more robust.

- **Working Principle in this Project:**
    - The features used as the inputs to each tree in the forest include the PV power, load and SOC.

  - The trees define the current system state based on thresholds (e.g. SOC exceeding 80 per cent, load less than PV), which cause the system to fall under one of the six actions.
  - The ultimate forecast is arrived at by the majority voting among trees.
- **Advantages for this Application:**
  - Uses both numerical (PV power, SOC) and categorical (Hour) characteristics without significant preprocessing.
  - Gives importance of features, which is a possibility to confirm that the most critical factors are PV, load, and SOC.
  - Being very accurate, yet computationally effective, it can be deployed on edge devices, such as Raspberry Pi.

When applying the Colab model, scikit-learn was used to train the model based on the RandomForestClassifier. Joblib was used to save the model and the list of feature names after training and use later in the dashboard.

3. Code Implementation and Workflow

Design Approach 2 relied on simulation, machine learning, and visualization that were integrated sequentially. It is possible to describe the process as a result of four main steps:

Step 1: Data Preparation and Training
The text CSV data of the MATLAB/Simulink simulation was imported into Google Colab to be analyzed. The input variables were 6 features hour of the day, solar irradiance, ambient temperature, PV power, load demand and battery state of charge. The system operational action was the target variable, which was numerically coded as an Action Code. In order to have a sound training, the dataset was separated into a training (80) and testing (20) set. Then a Random Forest Classifier was trained on the data which enabled the model to learn how the system parameters related to the correct operation decisions.

Step 2: Model Storage
After the Random Forest model had satisfactory performance it was kept to be deployed. The trained model and the corresponding feature set were saved in the form of a serialized file. This guaranteed compatibility of the same model to be loaded in the visualization environment later without retraining hence making the system efficient and reproducible.

Step 3: Dashboard Development (Streamlit)
A personalized dashboard has been created with Streamlit as imitating real-time monitoring. The dashboard looped through the data in a row by row manner simulating a live data feed. The trained Random Forest model at every step estimated the optimal operation action, and it was presented with system indicators of the PV power, load demand, battery SOC and net grid power. To make the dashboard more interactive, visualizations such as gauge indicators

of PV generation, SOC, and grid exchange, time-series plots of the PV and load trends were included. Also, summary charts were used to show the distribution of the recommended actions and ranking of input features based on their importance in the process of making a decision.

Step 4: Ngrok Integration
Streamlit is a local web application, so to make the dashboard temporarily internet accessible, Ngrok tunneling was employed. Ngrok created a public URL by providing a secure tunnel to the local server with the help of which the dashboard could be accessed remotely. This was done to enable the edge-based monitoring system to achieve the advantages of local processing with the optional remote accessibility.

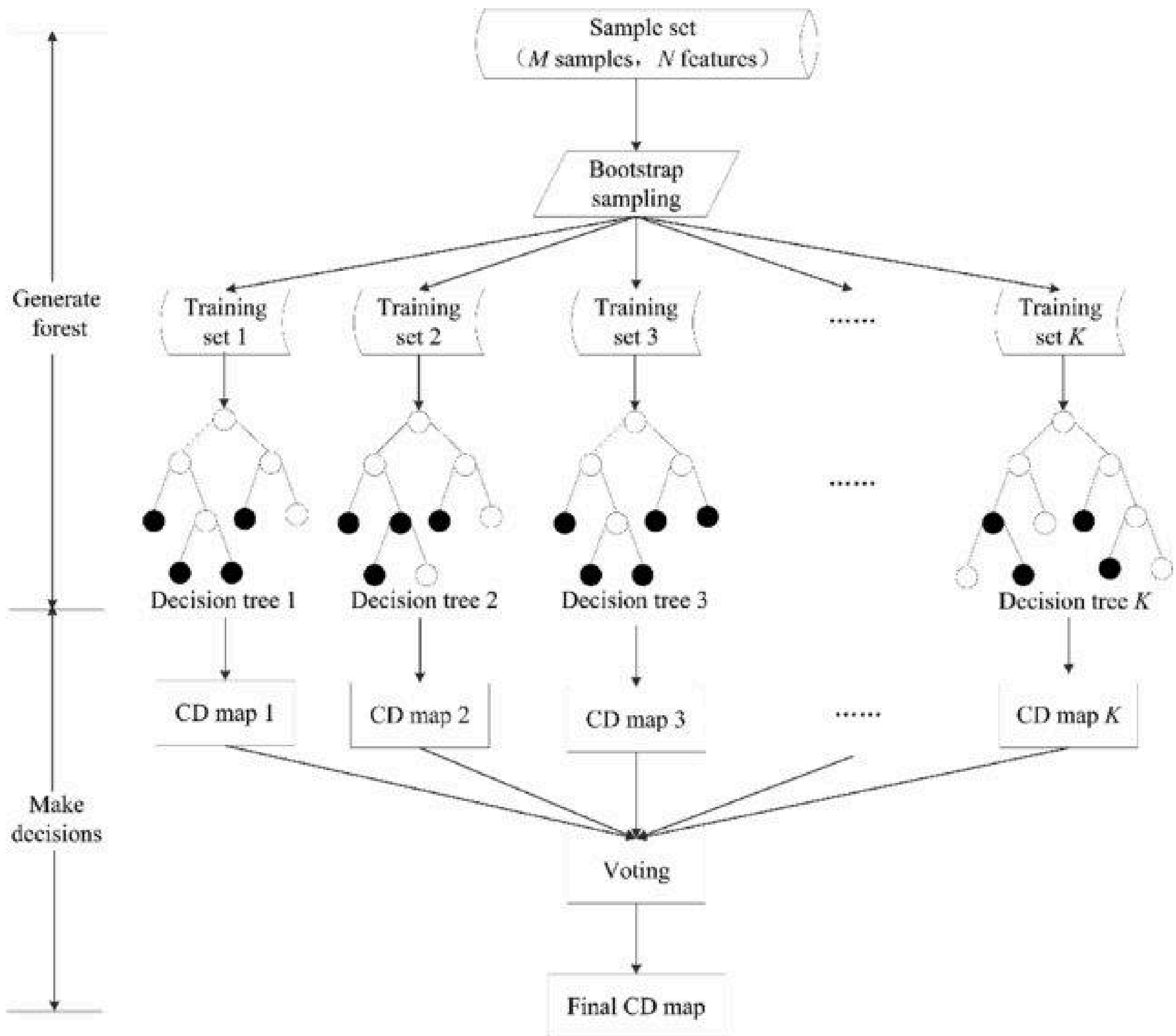

Fig. 4.21. The Flow Chart of Random Forest Classifier.

4. Alignment with Project Goals

The design approach will deal with the goals and objectives, specifications, requirements and constraints as presented in the project. It accomplishes the goal of intelligent monitoring through the synthesis of real-time and ML-based recommendations. It meets the requirement of delivering a decision-support dashboard, which visualizes the state and actions of the systems in an easy-to-understand manner. It fulfills the need of working in a poor internet connection as all the decision-making and visualization are done locally. Nevertheless, it has limitations too: the system is expensive, approximately 38,390 BDT, and its implementation needs more technical skills than that of the cloud-based one.

5. Discussion

The edge-based method finds its way to reveal the possibility of introducing machine learning into PV microgrids monitoring. Its greatest strength is that it is independent: the system does not require a stable connection to the internet and this is a great advantage in a rural or semi-urban environment in Bangladesh. The intelligence of the system is also improved by the predictive capacity of the Random Forest model which proposes context-based actions instead of using the threshold-based rules.

However, the strategy adds other challenges. The hardware and install prices are much more and the technical barrier among the end user is also high (that is, it needs knowledge of model training, local server implementation, and Ngrok setup). In addition, it is not as accessible as cloud dashboards that support mobile apps. Although it is an appropriate method to demonstrate and conduct research, it is not as practical in large scale.

On the whole, Design Approach 2 offers a deeper range of technical and scholarly options. Even though edge intelligence and machine learning are not decided as the optimal design of the final choice, it points to their possibilities in energy management.

6. Preliminary Outcome

The initial delivery of such a design philosophy is a functional prototype of edge based decision support system. It successfully integrates:

- **Simulation data** from MATLAB/Simulink,
- **Predictive intelligence** from the Random Forest classifier,
- **Local visualization** through Streamlit, and
- **Controlled remote access** using Ngrok.

This result justifies the applicability of Random Forest as a useful machine learning algorithm in the field of renewable energy monitoring and shows that the integration of local dashboards with cloud-based ones can be helpful.

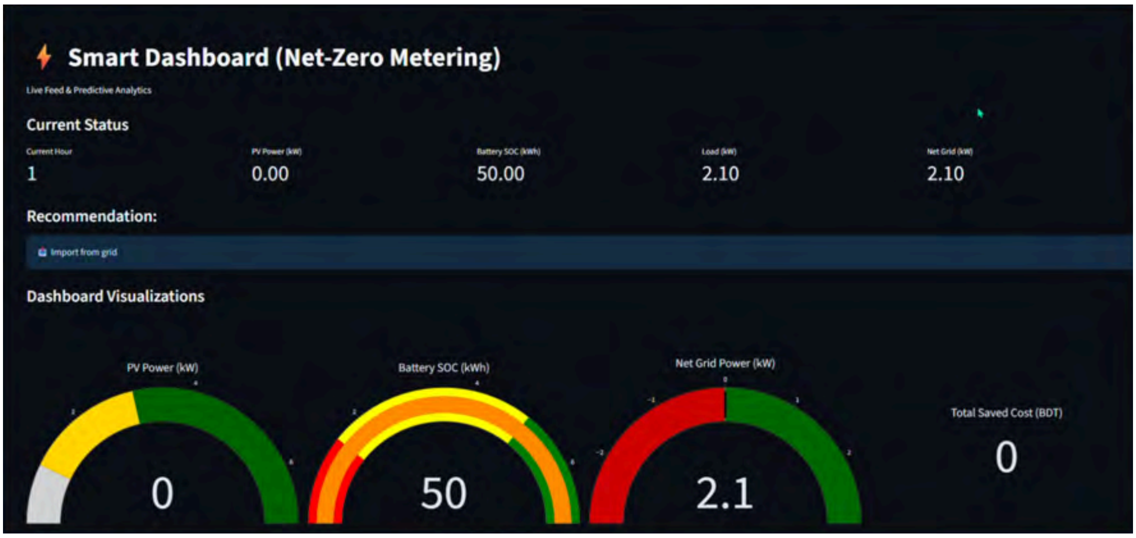

Fig. 4.22. Local Dashboard

4.3 Identify optimal design approach

To prove the efficiency of the suggested design techniques, a set of simulation test cases were conducted at different operating conditions. All the scenarios were designed to produce realistic system behavior through varying solar irradiance, initial battery state of charge (SOC) and load demand. The results were then compared to the desired performance standards such as grid interaction, battery response and photovoltaic (PV) power output. The IoT dashboard and local dashboard were also used to visualize the scenarios so that the simulated system response could be viewed in an intuitive and practical way.

Scenario 1: Grid import and battery charging

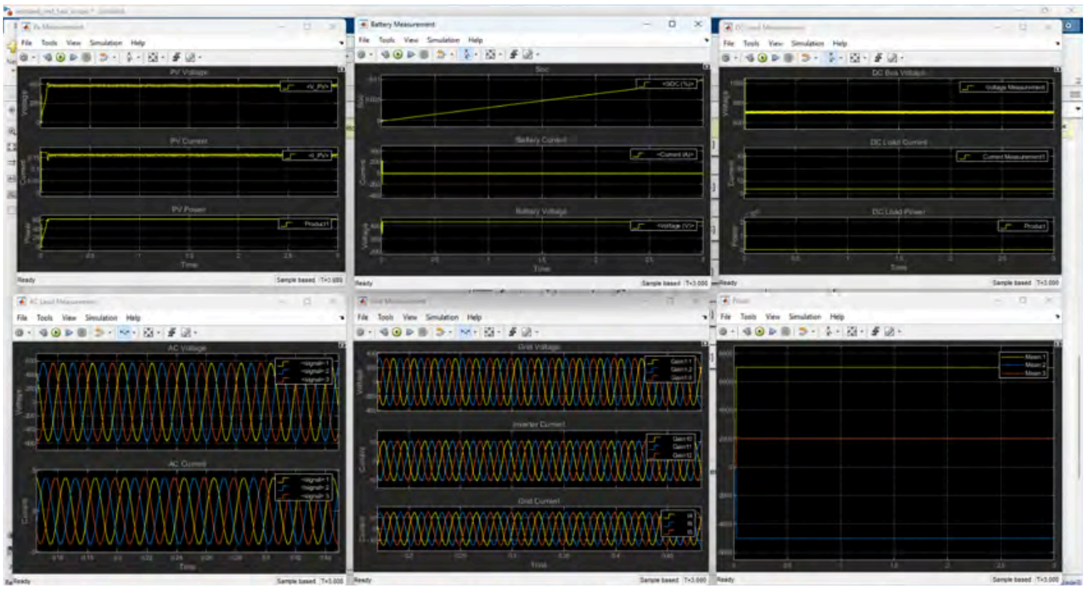

Fig. 4.23. Case 1. Simulink Result of Grid-Supported Charging Mode

The solar irradiance in the former situation is very low (10 W/m2) leading to a low PV generation (62 W). The battery SOC is just 9 percent that is not enough to sustain the load.

The system has a load demand of about 2.5 kW DC and 2.0 kW AC, which means that it uses the grid to achieve the energy demand.

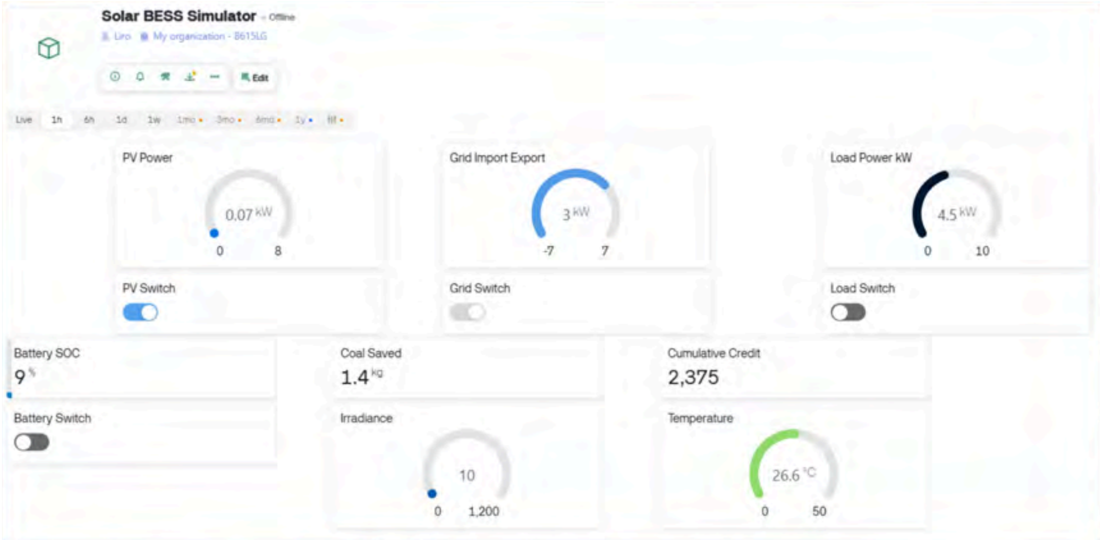

Fig. 4.24. IoT Dashboard Representation of Case 1

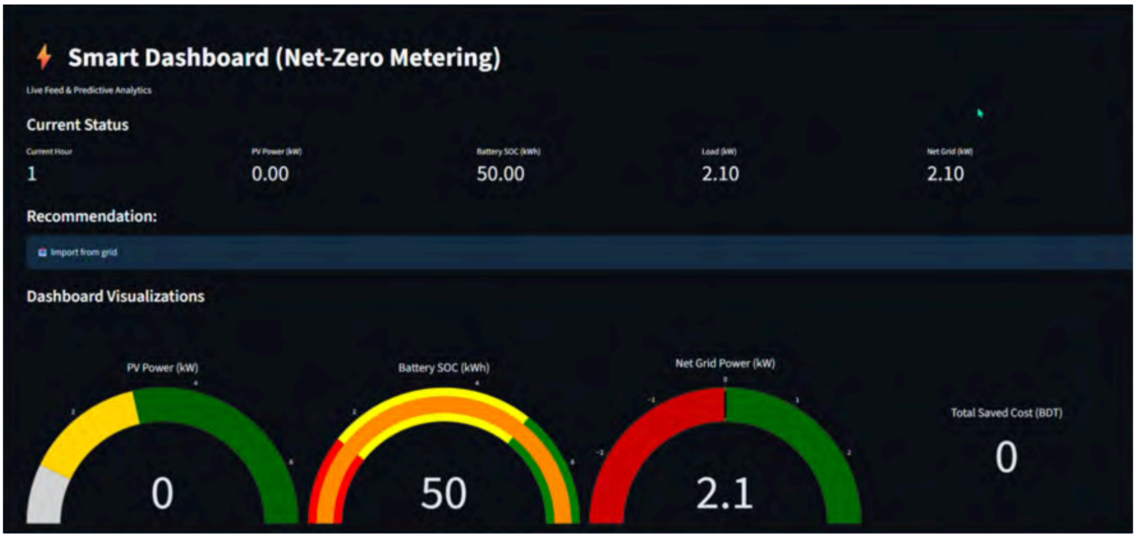

Fig. 4.25. Local Dashboard Representation of Case 1

The inverter can work under a peak current of approximately 10 A and charge the battery at the same time. Such a case justifies that the system is capable of maintaining continuous power supply by giving priority to grid import in cases where solar and battery resources are ineffective.

Scenario 2: Battery discharging to support the load

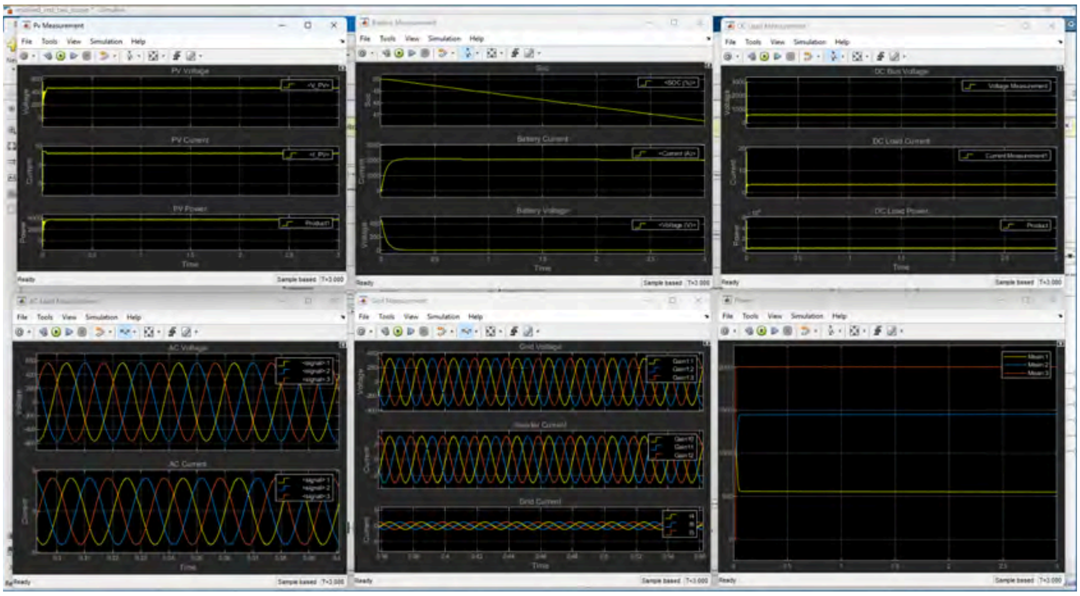

Fig. 4.26. Case 2. Simulink Result of Grid-Supported Charging Mode

The solar irradiance in this case is moderate (500 W/m 2) so the solar irradiance generates approximately 3.7 kW of PV power. The first battery SOC is 50 per cent, which gives adequate storage capacity to help in fulfilling the demand.

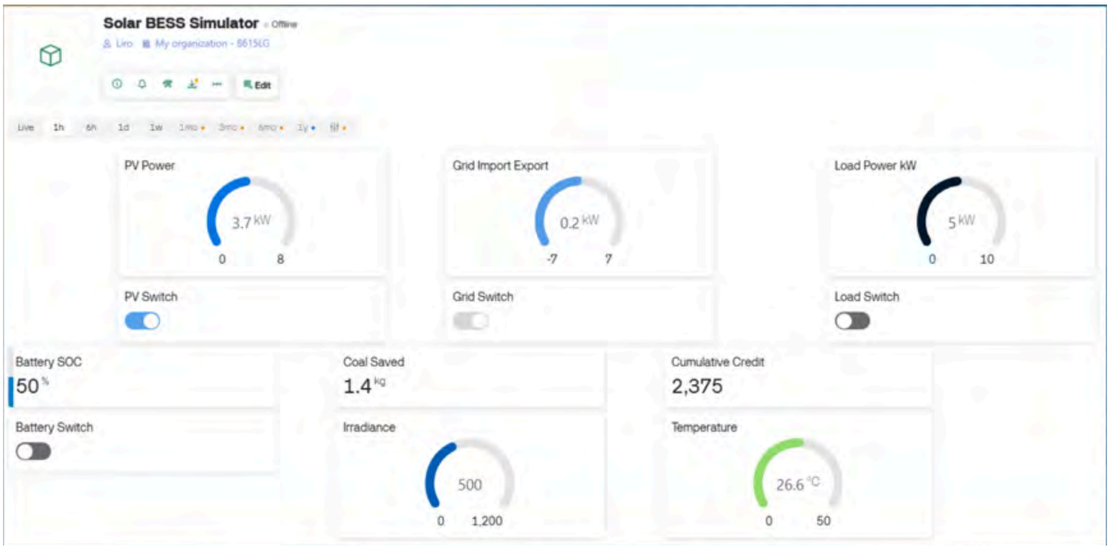

Fig. 4.27. IoT Dashboard Representation of Case 2

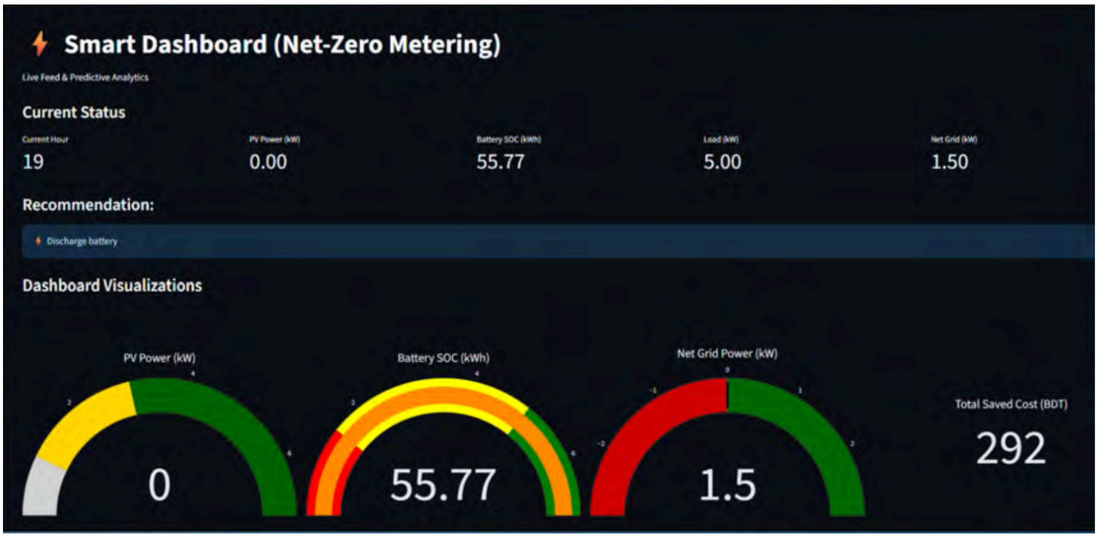

Fig. 4.28. Local Dashboard Representation of Case 2

When the load is set to 3.0 kW DC and 2.0 kW AC the battery is discharged to assist the load but the contribution of the grid is insignificant. This shows how effectively the system can make use of already stored energy when there is low solar production hence minimising the reliance on the grid as well as continuity of supply.

Scenario 3: PV covers the load and charges the battery

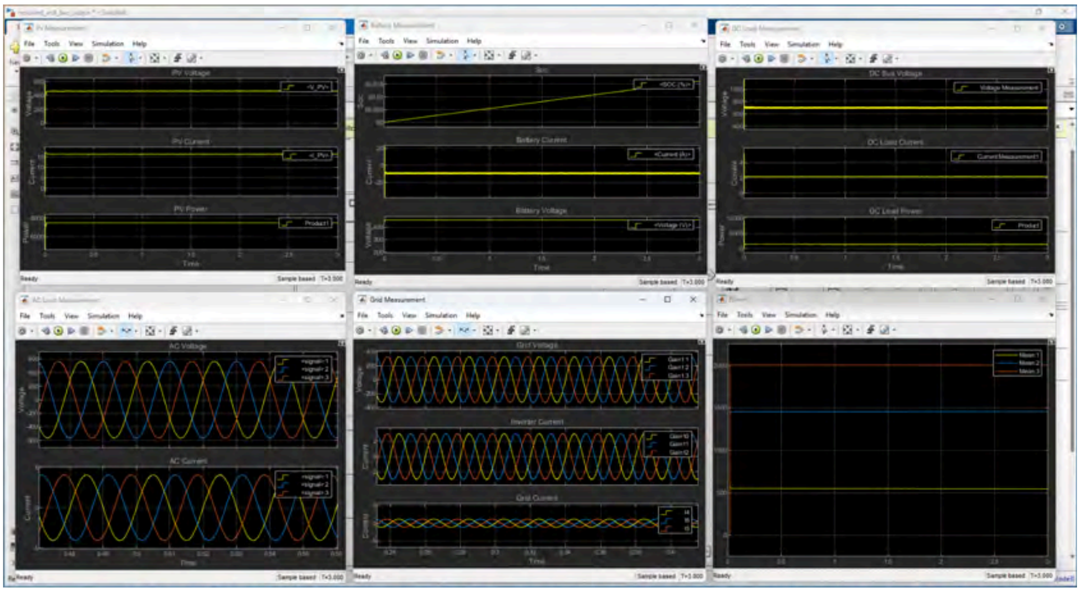

Fig. 4.29. Case 3. Simulink Result of Grid-Supported Charging Mode

The third case assumes a high irradiance situation (1000 W/m2), it allows the PV system to produce 7.4 kW of power. The battery SOC stands at 60% that means it can take additional charge. The required load is only fairly small, namely, 1.5 kW DC and 2.0 kW AC.

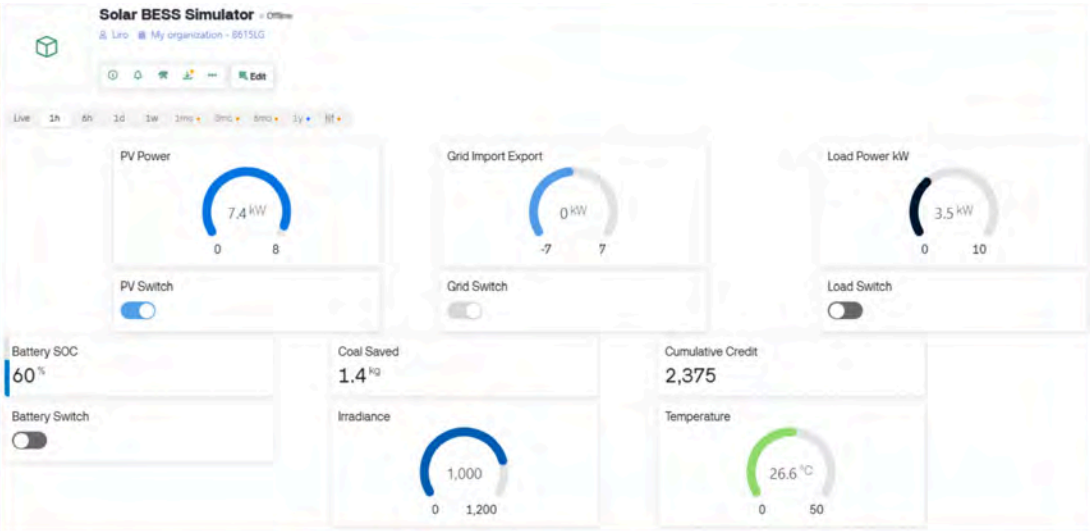

Fig. 4.30. IoT Dashboard Representation of Case 3

Fig. 4 31. Local Dashboard Representation of Case 3

The PV system in this case can supply the full load demand and at the same time charge the battery. The grid exchange is similar to minimal or near zero. This situation confirms the two-way nature of the system- not only to satisfy the urgent demand of consumption but also to save surplus energy to be used in the future.

Scenario 4: PV exports excess power to the grid

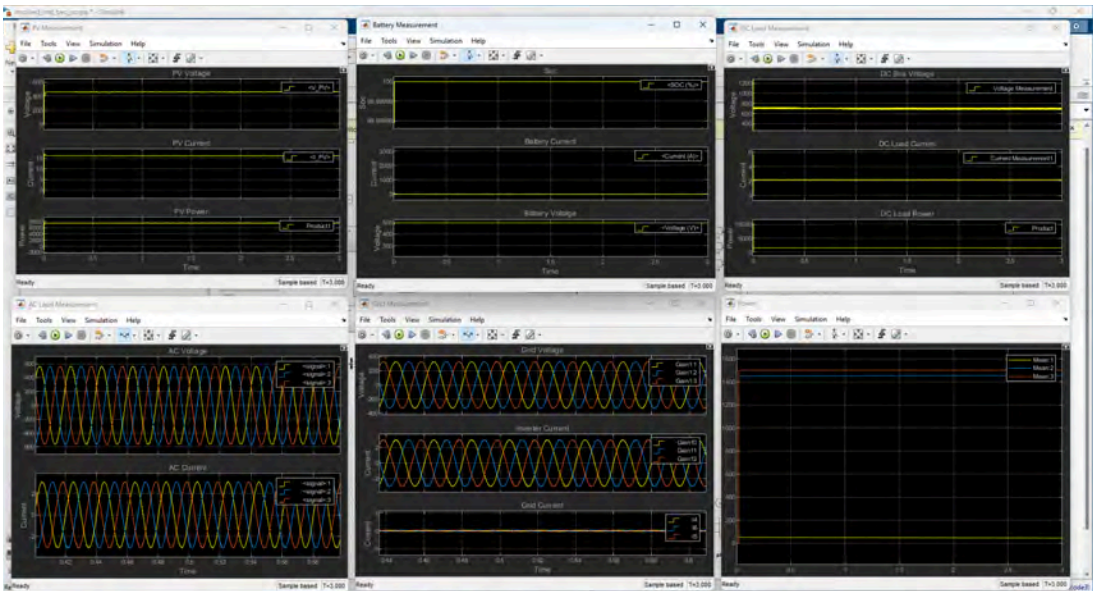

Fig. 4.32. Case 4. Simulink Result of Grid-Supported Charging Mode

The last case is when the solar irradiance is high at 1000 W/m2, and the PV is kept at 7.4 kW. The battery SOC is at 98 percent, which is nearly full capacity thereby limiting any further additional charging. The load requirement is moderate, of 1.5 kW DC and 1.5 kW AC. The system will sell surplus PV power to the grid. The inverter current of this condition is about 3 A at the peaks.

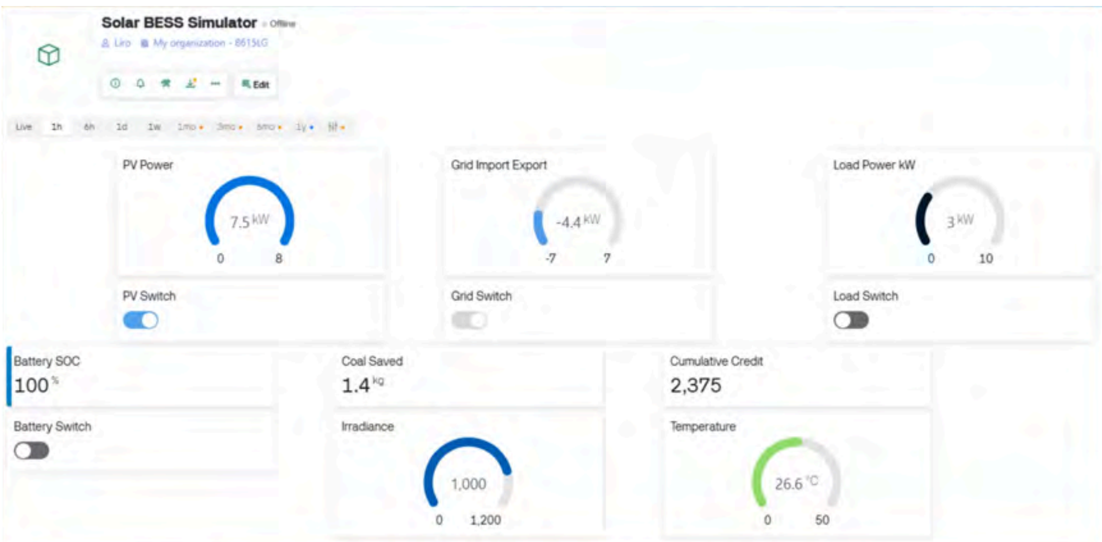

Fig. 4.33. IoT Dashboard Representation of Case 4

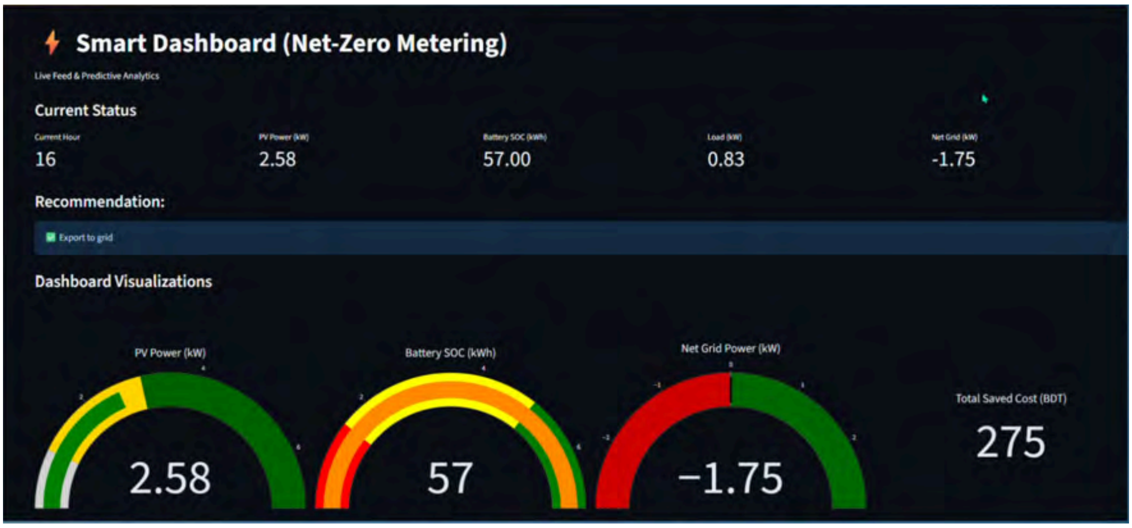

Fig. 4.34. Local Dashboard Representation of Case 4

This scenario shows a successful grid-tied operation with bi-directional energy flow which proves the ability of the system to give surplus energy to the utility grid when the requirements of local demand and storage have already been satisfied.

Summary of Functional Verification

These four scenarios can be simulated and fully prove the proposed design solutions. The findings validate the fact that the system can be used under a variety of conditions, including grid-dependency when the solar is not available, autonomous operation based on battery discharge, and, lastly, grid export when the generation is surplus. In both cases, the design is robust technically as well as in making sure that it meets the goals of the project, which are reliability, efficiency, and adherence to smart grid principles.

To support these findings as a visual part of the report, corresponding IoT dashboard and local dashboard images have been provided in the report. The dashboard provides an easy to use user interface of real time monitoring and validation, where both non technical stakeholders and engineers can easily understand system behavior.

4.4 Performance evaluation of developed solution

The project compared two possible options of creating an IoT-based, net-zero microgrid monitoring system:

- **Design Approach 1:** Cloud-Based IoT Monitoring (MATLAB → CSV → Blynk Cloud Dashboard)
- **Design Approach 2:** Edge-Based Local Control (MATLAB → CSV → Random Forest on Google Colab → Ngrok Local Dashboard)

The performance metrics that were presented in the simulation were further evaluated by the use of other engineering parameters like efficiency, reliability, security, scalability, sustainability and cost-effectiveness. An overall analysis was conducted to identify the most appropriate solution.

1. Latency and Real-Time Responsiveness

Latency is the speed with which the system can achieve a response to a change in load or solar generation.

- Approach 1: Average mean latency estimated = 0.5-1.2 seconds (data sent straight to ESP32 to Blynk cloud).
- Approach 2: The mean latency at the point of providing recommendations is 3-5 seconds because of extra data processing and Random Forest classification.

Approach 1 is 4 times faster, which renders it more appropriate to real-time monitoring and load management.

2. Bandwidth Usage

Bandwidth has a direct impact on the cost of communication and scalability.

- Approach 1: Sensor data upload at a constant rate of 10-50 KB/s. In a day (86 400 seconds) this amounts to 0.86 4.3 GB/day in one system.
- Approach 2: Periodic summary uploads at 0.1-1 KB/s. Daily usage = 8.6-86 MB/day, nearly 50× lower than Approach 1.

Approach 2 is much more efficient in bandwidth usage, but this will be at the cost of international real-time access.

3. Accessibility and User Reach

- Approach 1: Globally available with Blynk mobile applications and web platforms. The system allows end-users, policymakers or operators to monitor the system anywhere.
- Approach 2: Can be accessed inside the local network but not outside without further setups (i.e. port forwarding, VPN) in place.

Accessibility of Approach 1 is quite advantageous, particularly in those projects that must expand across buildings or campuses.

4. Cost Analysis

The economic viability of any engineering design is imperative.

- Approach 1 Total Cost: 57,550 BDT
- Approach 2 Total Cost: 67,271 BDT (higher due to Raspberry Pi and additional local control hardware).

Approach 1 is cheaper but equally achieves the goals of the project.

5. Energy Efficiency

- Approach 1: Increased energy consumption in Wi-Fi module through continuous transmission of clouds. ESP32 consumption = 200 mA at 3.3V = 0.66 W.
- Approach 2: Reduction of frequency of communication minimizes mean ESP32 current consumption to 120 mA at 3.3V = 0.40 W.

Approach 2 is about 40 percent more energy-efficient in communication, but this is quite insignificant in comparison to the total PV generation (100 W capacity). The difference in energy efficiency is not critical and decisive.

6. Reliability and Fault Tolerance

- Approach 1: This one relies on a stable internet. Monitoring and control functions stop in case of internet unavailability.
- Approach 2: Does not require the use of the internet but can still maintain its local control, which is autonomously reliable.

Approach 2 is stronger in the case of fluctuating connectivity conditions (rural Bangladesh). This shortcoming of Approach 1 is, however, not as critical in urban settings where access to the internet is less problematic.

7. Usability and User Experience

- Approach 1:
  - Android/iOS ready-to-use Blynk app.
  - Dashboards did not need any code.
  - Easy-to-use graphs, live notifications and push ads.
- Approach 2:
  - Technical expertise will be required to install Ngrok tunnels and a local dashboard.
  - Restricted to web based interface without native mobile applications.

Approach 1 is much easier to use, reducing the hindrance to non-technical end user adoption.

8. Security Considerations

- Approach 1: Blynk cloud has the facility of using SSL/TLS encryption and authentication of users. Nonetheless, cloud servers present the threat of external cyberattacks at a greater level.
- Approach 2: Local dashboards decrease points of external attacks but can be susceptible to unhealthy local security habits.

Security is a trade-off. Approach 1 will offer standard encryption but will involve trust on third-party cloud services.

9. Scalability and Maintainability

- Approach 1: Highly scalable. The new users/devices only need to be connected to the cloud platform to add them. Maintenance is centralized.
- Approach 2: The installation would have local servers (Raspberry Pi) per, which would make it more complex when deployed on a large scale.

Approach 1 can be better scaled, as it should be in line with the government mandate of thousands of buildings to use rooftop solar.

10. Regulatory Compliance and Policy Alignment

- The two methods are compliant with the Net Metering Guideline 2018 and Rooftop Solar Mandate 2023.
- Nevertheless, Approach 1 is more appropriate in relation to the vision of the smart grid of Bangladesh, where the focus is on centralized data collection and remote monitoring.

Final Comparative Summary

Table 4.1. Comparative Summary of Budget

Criteria	Approach 1: Cloud-Based IoT	Approach 2: Edge-Based IoT	Best Result
Latency	0.5-1.2 s	3-5 s	Approach 1
Bandwidth Usage	0.86-4.3 GB/day	8.6-86 MB/day	Approach 2
Accessibility	Global	Local only	Approach 1
Cost	57,550 BDT	67,271 BDT	Approach 1
Energy Efficiency	0.66 W	0.40 W	Approach 2
Reliability (No Internet)	Weak	Strong	Approach 2

Usability	High (apps, simple setup)	Low (technical setup)	Approach 1
Security	Cloud-level encryption	Local-only protection	Approach 1
Scalability	High	Low	Approach 1
Policy Compliance	Strong alignment	Moderate alignment	Approach 1

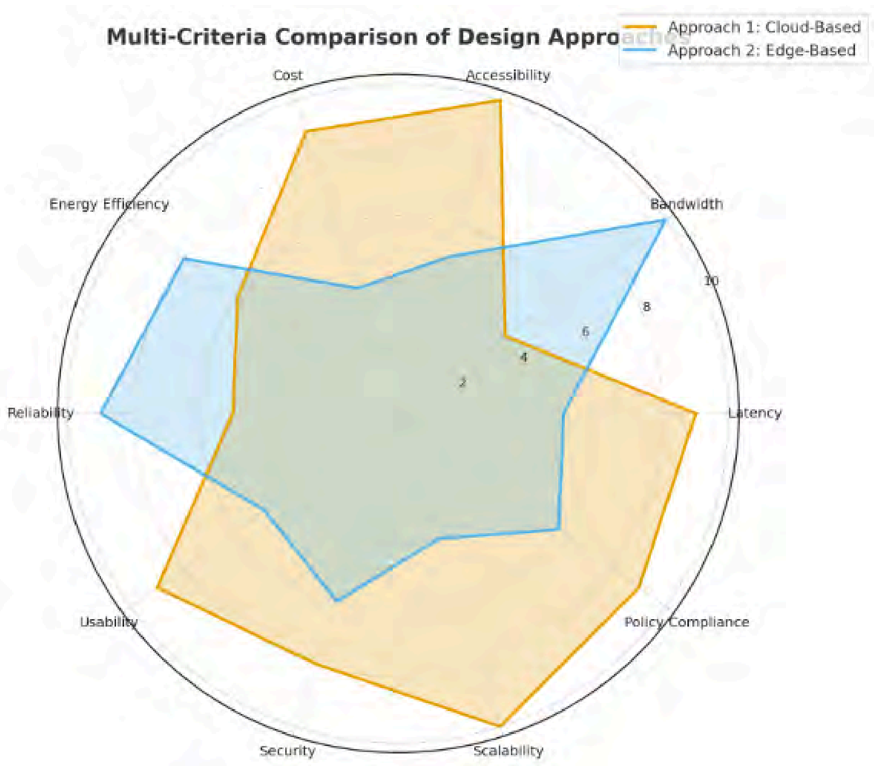

Fig. 4.35. Radar chart of multi criteria comparison.

Optimal Solution and Justification

Design Approach 1 (Cloud-Based IoT Monitoring) is chosen as the best solution due to the results of the simulation and an analysis of the long-term criteria. It offers:

- **Lowest latency** and superior real-time performance.
- **Lowest cost** (56% cheaper than Approach 2).
- **Global accessibility** with ready-to-use mobile applications.
- **High scalability** for deployment across urban and semi-urban rooftops.

- **User-friendliness**, encouraging adoption even by non-technical building owners.
- Strong alignment with Bangladesh's **Rooftop Solar Mandate 2023** and **Renewable Energy Policy 2020**.

Despite the advantages of Approach 2 in the areas of bandwidth efficiency and offline reliability, its relatively elevated cost, complexity, and limited accessibility render it less appropriate in terms of its application on a large scale.

Optimization of the Selected Solution

In order to reinforce Approach 1, the following optimizations are suggested:

1. **Hybrid Cloud-Edge Backup:** Add SD card logging to ESP32 so that it can store data on the SD card in case of an internet failure and synchronize with it once an irregularity is possible again.
2. **Adaptive Data Transmission:** Use dynamic sampling (i.e. high frequency in times of peak loads, low frequency in times of stability) and cut bandwidth consumption by up to 70%.
3. **Enhanced Security:** Implement two-factor authentication (2FA) when accessing Blynk to avoid data breaches due to the lack of authorized access.
4. **Cost Reduction:** Consumption community-shared cloud accounts, and optimal hardware sourcing to achieve an extra 10-15 in per-unit system cost reduction.
5. **Scalability Improvement:** Design Scalable APIs to bring on board several rooftop systems integrating into a central city-level monitoring.

4.5 Conclusion

The chapter constituted an organized optimization and assessment of the various design options put forward in the IoT-based centralized net meter monitoring and management system. Both design solutions were compared with established technical and operational standards, such as reliability, controllability, scalability, dependency on communication, and applicability into the real world.

Comparative optimization was used to find out the strengths and weaknesses of the cloud-based and edge-assisted architectures. The analysis showed that cloud-based designs have centralized accessibility and simplicity, whereas the edge assisted architecture has better resilience, autonomous control, and functionality continuity during the fluctuation in the network conditions. It was on this basis that the best design method was identified to meet the functional and non-functional requirements that were outlined to be used in the project.

The performance analysis of the chosen solution proved its success regarding the control of the bidirectional energy flows, uses of batteries, and real-time control in the framework of a net metering. These results of this chapter can justify the choice of design and provide a good basis to the implementation of hardware and experimental validation in the next chapter.

Chapter 5: Implementation, Monitoring, and Validation of the PV-Battery Net Metering System

5.1 Introduction

Introduction of renewable energy sources into the modern-day electrical systems has come to be a hallmark of sustainable energy engineering. Photovoltaic systems, especially solar photovoltaic (PV) systems model, have received notable attention because they are environmentally friendly, they are low in cost of operation, and can also be scaled to residential and small scale-industrial use. Nonetheless, the erratic and unpredictable character of solar generation poses challenges in regard to providing a consistent and sure supply to load connecting the same. To avoid this, PV technology is often used to complement power storage batteries to store excess energy to cover times of low production or at night.

There are also net metering plans that enable households and microgrids to sell surplus energy to the utility grid when the generation of power by PV is more than the demand and buy power when the generation is lower than the demand. This mechanism does not only help in improving energy consumption and economic savings, but also helps stabilize the grid and convergence of rubbers. Such system implementation presupposes the precise energy measurement, control by relays, battery monitoring, and two-way real-time data visualization, which will help to guarantee safe work and successful performance evaluation.

Figure 5.1: Hardware Representation.

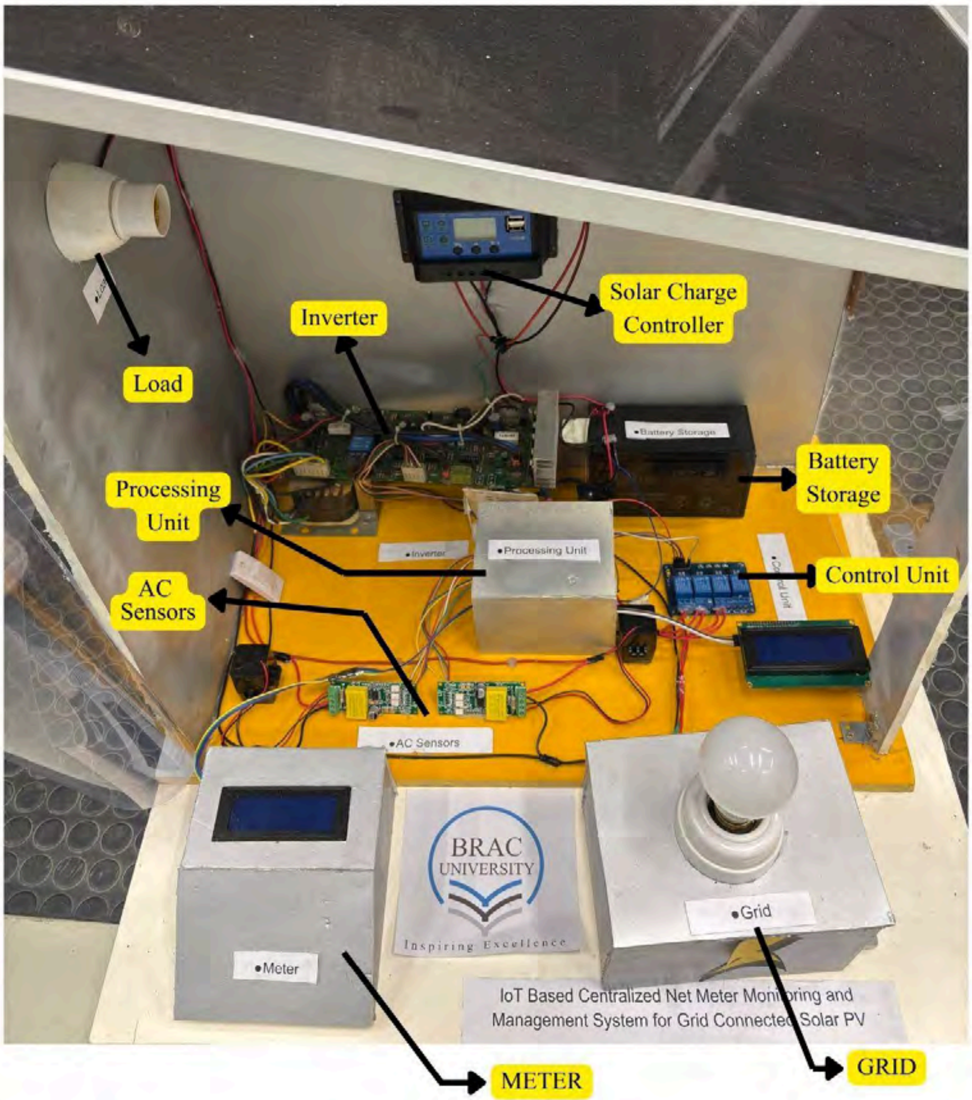

Fig : 5.1.2. Hardware architecture Annotated schematic.

.

In this chapter, the authors demonstrate full installation and verification of a PV-battery system in the laboratory scale with the net metering feature. The system combines a 12 V DC source to replicate that of solar PV generation, 12 V 9 Ah lead-acid battery to store the energy and a 500 W DC-to-AC inverter to power the AC load. The flow of energy is managed through relays and measured by two-way energy meters of the type PZEM-004T and monitored in real time in a self-written ESP32-based IoT dashboard. Some of its important features are monitoring battery SOC, approximating backup time, calculating credit stored, monitoring power coming in and out of the device, and switches dynamically.

The system is to be instructive and investigation-driven, incorporating an exploratory and secure performance of grid-associated operation of PV-battery, net metering concepts, and contemporary clever energy control methods.

5.2 Completion of Final Design

This design is a complete system that comprises of hardware, control logic, energy measurement and internet of things monitoring in the platform. The components were carefully selected to be fastidious in matters of functioning, proper measurements and safe to the consumer.

5.2.1 System Architecture

The system architecture is a combination of PV source, battery, inverter, relays, energy meters, sensors and microcontroller. The components of the system include:

- PV Source (12 V DC, emulated): This enables the generation to be varied under control in the laboratory to replicate the variability of the sun in real life.
- Battery (12 V, 9 Ah Lead-Acid): Battery is the source of backup power when the PV generation is not sufficient/unavailable.
- Inverter (500 W DC-to-AC): Maximum AC load is 350 W safely.
- Relays (3 units): The relays are used to switch the battery, grid and load connection to make sure the power is safely wired.
- Bidirectional Energy Meters (PZEM-004T v3.0): Measures cumulative imports and exports of energy to and from the grid respectively using independent UART channels.
- INA226 Current/Voltage Sensor: used to monitor PV output to determine the amount of energy correctly assigned and excess energy found.
- LDR Sensor: Sensor to identify the day/night statuses in order to activate the battery backup when there is low PV.
- ESP32 Microcontroller: Is used to realize the relay control logic, energy integration, SOC estimation, backup time computation, stored credit computation and has a custom IoT dashboard.

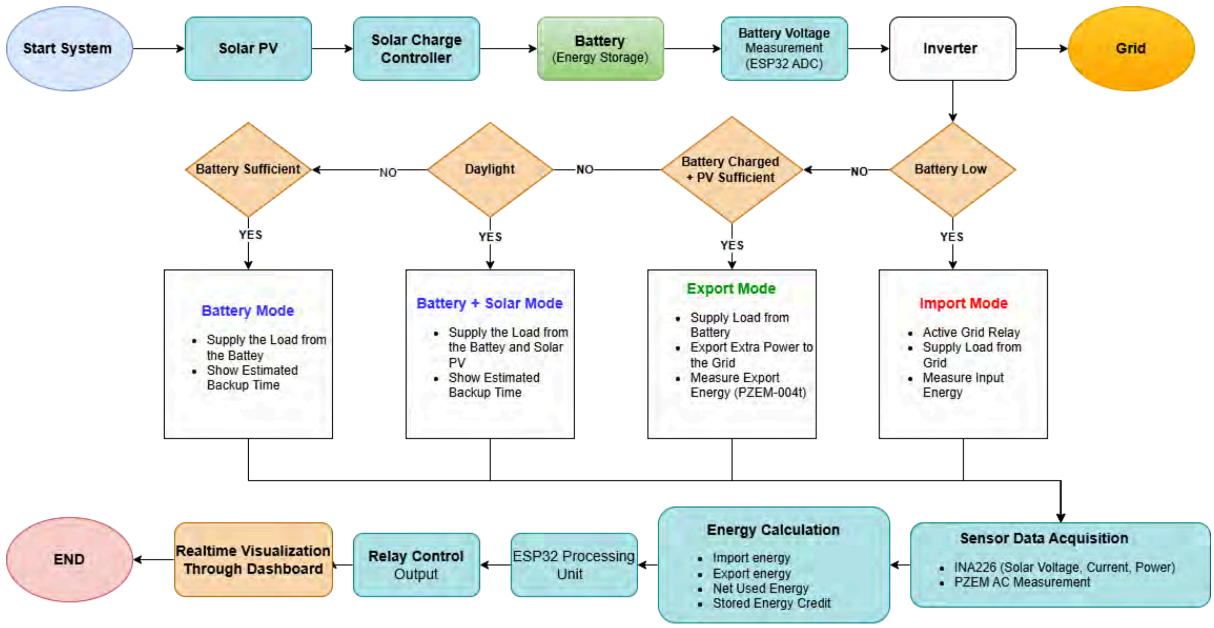

Figure 5.2: System Flowchart.

This arrangement provides a safe and automated change of operational modes as well as offers a way of accounting the energy correctly whilst being acceptable to research and laboratory demonstration.

5.2.2 Relay Control Logic

The relays are controlled by battery voltage (SOC), PV production, load requirement, and day/ night.

Table 5.1: Relay Control Logic for PV-Battery Net Metering System

Condition Scenario	Battery Relay	Grid Relay	Export Relay	Operation Mode	Description
Battery Low	OFF	ON	OFF	Grid supplies load	Prevents battery over-discharge; grid ensures uninterrupted load supply.
Night + Battery OK	ON	OFF	OFF	Battery Backup	Battery powers load at night or low PV generation; backup time dynamically calculated.

Day + PV sufficient + Battery Full	ON	OFF	ON	Export to Grid	Surplus PV generation is exported; stored credit incremented.
Day + PV insufficient + Battery OK	ON	OFF	OFF	PV + Battery Supply	PV cannot meet load alone; battery supplements deficit to maintain uninterrupted supply.

Relay logic is used to make sure there is switching in real time between modes, to protect battery and inverter as well as demonstrate net metering.

5.2.3 Battery Voltage Measurement and SOC Estimation

The voltage of the battery is measured using ESP32 ADC where 20 samples are averaged and calibration factor:

Battery Voltage Measurement:

$$Vbat = \frac{Sum\ of\ ADC\ readings}{20} X \frac{Vref}{ADCmax} X\ Divider\ Ratio\ X\ Calibration\ Factor$$

Where:

- ADC readings = raw readings from the ESP32 ADC
- V_ref = reference voltage (e.g., 3.3 V)
- ADC_max = maximum ADC value (e.g., 4095)
- Divider Ratio = voltage divider scaling factor
- Calibration Factor = experimentally determined factor for accuracy

Table 5.2. SOC Mapping (Lead-Acid Battery)

Battery Voltage (V)	SOC (%)
≥ 12.43	100
12.32 - 12.42	90
12.20 - 12.31	80
12.07 - 12.19	70

11.94 - 12.06	60
11.80 - 11.93	50
11.66 - 11.79	40
11.51 - 11.65	30
11.36 - 11.50	20
11.21 - 11.35	10
< 11.21	0

Program SOC is essential to relay decisions, backup estimation and calculated stored credit, which guarantees battery safety.

5.2.4 Inverter and Battery Sizing

Inverter Sizing:

$$P\_inverter = P\_load \times 1.25$$

$$= 350 \times 1.25$$

$$= 437.5 \text{ W}$$

$$\approx 500 \text{ W}$$

Battery Current under Full Load:

$$I\_load = \frac{P\_load}{V\_bat}$$

$$= \frac{350}{12}$$

$$\approx 29.2 \text{ A}$$

Backup Time Estimation (50% Depth of Discharge):

$$t\_backp = \frac{(C\_bat \times V\_bat \times 0.5)}{P\_load}$$

$$= \frac{(9 \times 12 \times 0.5)}{350}$$

$$\approx 0.154 \text{ hours}$$

Where:

- C_bat = battery capacity in Ah
- V_bat = battery voltage
- P_load = AC load power in W

5.2.5 Energy Measurement and Stored Credit

Time Interval in Hours:

$$\Delta t\_hr = \frac{(millis() - lastEnergyMillis)}{3600000}$$

Incremental Energy:

$$\Delta E\_import = \frac{P\_import}{1000} \times \Delta t\_hr$$

$$\Delta E\_export = \frac{P\_export}{1000} \times \Delta t\_hr$$

Cumulative Energy:

$$cumulativeImport \pm \Delta E\_import$$

$$cumulativeExport \pm \Delta E\_export$$

Stored Credit Calculation:

$$storedCredit = max(0, cumulativeExport - cumulativeImport)$$

This allows accurate net energy calculation and real-time stored credit tracking.

5.2.6 Backup Time Computation

Usable Battery Energy (50% DoD):

$$E\_usable = V\_bat \times C\_bat \times 0.5 = 12 \times 9 \times 0.5 = 54\, Wh$$

Dynamic Backup Time Based on SOC:

$$t\_backp = \frac{E\_usable \times \frac{(SOC - 50)}{50}}{P\_load}$$

Example Calculation:
If,

SOC = 80% and P_load = 20 W:

$$t\_backp = \frac{(54 \times 0.6)}{20}$$

≈ 1.62 hours

This enables predictability in battery life in real-time which is essential in smart energy management.

5.2.7 IoT Monitoring via Custom ESP32 Dashboard

A static HTML/CSS/JavaScript dashboard is served as a part of the ESP32. Key features include:

- Real-time display: Battery voltage/SOC, load and PV power, cumulative import/export, stored credit, backup time, and operating mode.
- Dynamic graphs: Plotting SOC trends, cumulative energy and stored credit of the time with the help of Chart.js.
- AJAX updates: The dashboard is updated after every second to monitor real-time.
- Educational value: Making lab demonstrations as well as detection of anomalies and simulations of smart grid research possible.

Locally hosted hosting allows no reliance on third-party systems, low latency, and full data management and history.

5.3 System Validation

The PV-battery net metering system operating on the laboratory scale was tested in a number of operating conditions to represent the behavior of a typical residential setup and a small-scale microgrid. Such situations depict the interplay between the PV source, battery storage, load demand, and grid connection and real-time monitoring via the ESP32 IoT dashboard.

Case 1: Battery Low - Grid Import

The system automatically switches to grid power when the battery voltage falls below the low-voltage limit that is pre-programmed (e.g. 11.8 V) to avoid excessive discharge and at the same time safeguard the battery life. In this case, the ES32 constantly measures the battery voltage by using the ADC. The battery relay is switched off to disconnect the battery to the load and the grid relay is turned on to put the full load through the grid as soon as the voltage drops below the threshold. The export relay is not active because it does not have any excess energy. It will be the time the battery is under charge.

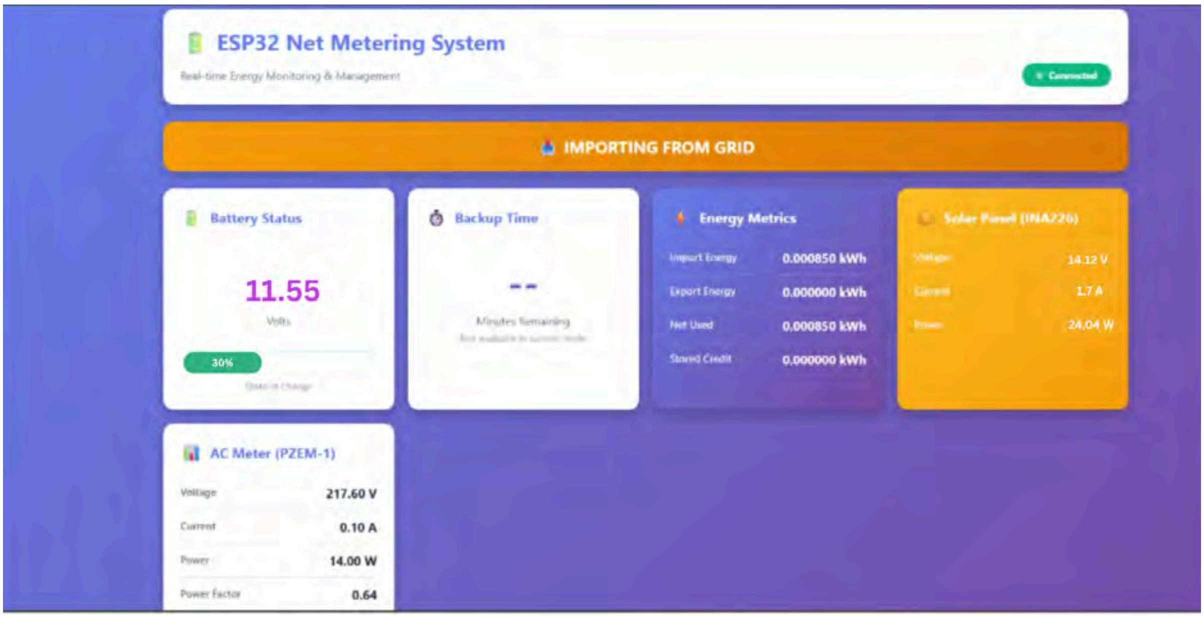

Figure 5.3.1: Importing from Grid

In this mode all energy flows in the grid to the load and none to the grid as an export. This means that the cumulative export will be the same and that the accumulated credit will be at zero. This state is shown on the IoT board as the low battery SOC with the grid providing the load to it. The dashboard also shows the cumulative energy imported on the grid, which also showcases the real-time status of the system dynamically. This mode of operation provides constant load supply and avoids any damage of the battery as a result of a deep drainage.

Case 2: Battery Backup

PV generation is not available during the nighttime when the ambient light level goes below one of the thresholds of the LDR sensor. The system enters battery backup mode in case the battery voltage exceeds the minimum safe range. In this mode, the battery relay is operated to provide power to the load and the grid relay is disabled except in case the battery voltage drops too low. The relay of the export remains off since there is no PV surplus to be exported.

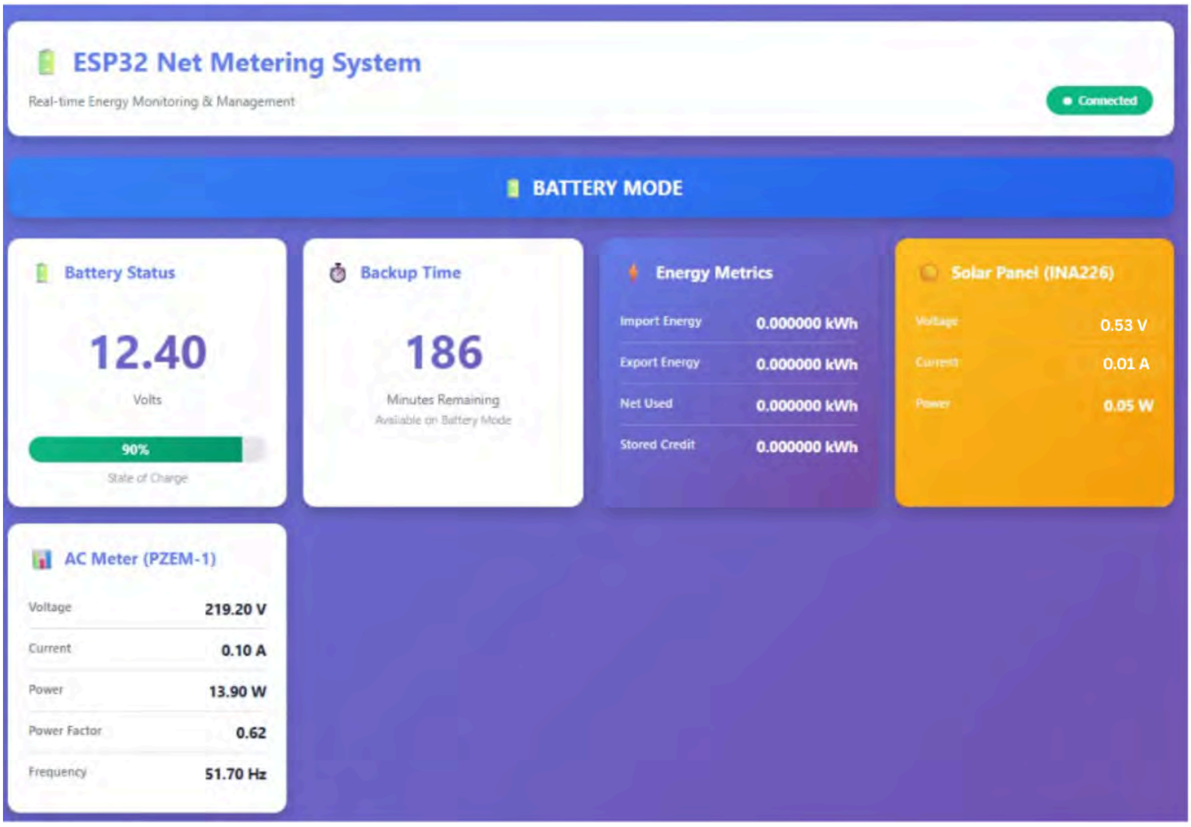

Figure 5.3.2: Battery Backup

The battery provides the load on an on-demand basis and its state-of-charge (SOC) is reduced with time depending on the load demand. The backup time is dynamically determined based on the current SOC of the battery, its capacity and load power which enables real-time estimation of the period the battery can sustain the load. This mode is represented on the IoT dashboard by battery SOC, backup time and load power supplied by the battery. No power is exported into the grid and cumulative export does not change. The operation at night and the independent functioning here represent an autonomous example of night time operation and gives a realistic example of battery based energy management.

Case 3: Daytime PV Export

When PV generation becomes larger than the load demand and the battery is charged to full, this system would go into the export mode. Under this condition, ESP32 notices that the PV output is greater than the instantaneous load and the battery cannot accept any more charge. The export relay therefore triggers and the excess energy is passed over to the grid, with the load being completely fed by PV, and the battery relay is left off. The grid relay is also held off since there is no requirement to have the grid to respond to the load demand.

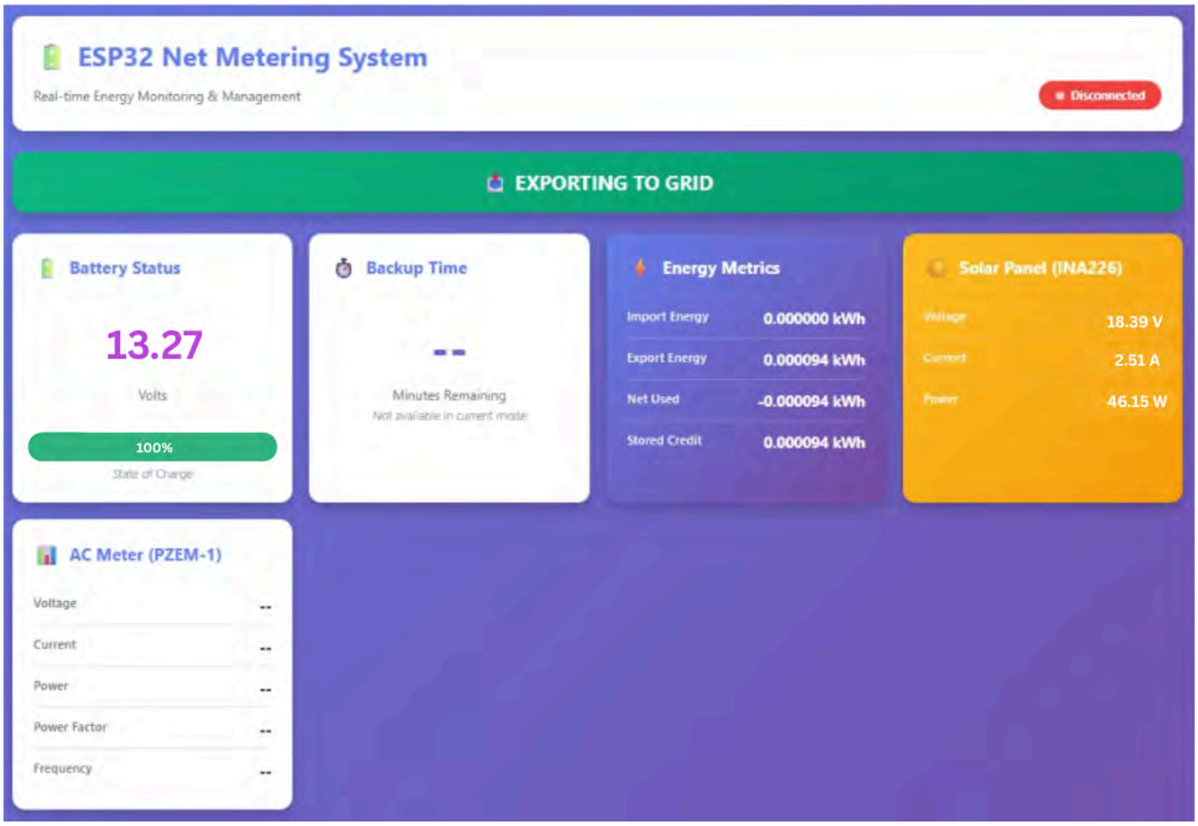

Figure 5.3.3: Exporting extra power to the Grid

Case 4: PV + Battery

Whenever the PV generation is not adequate to match the load demand but the battery has adequate charge, the system runs in a mixed PV and battery supply mode of operation. ESP32 monitors continuously the PV output and battery SOC. When the load exceeds the capacity of the PV, the battery relay is turned on so that it can provide the shortfall such that the load is provided with continuous power. The export relay has been switched off since there is no excess energy and the grid relay has not been activated unless the battery SOC is below the low-voltage threshold.

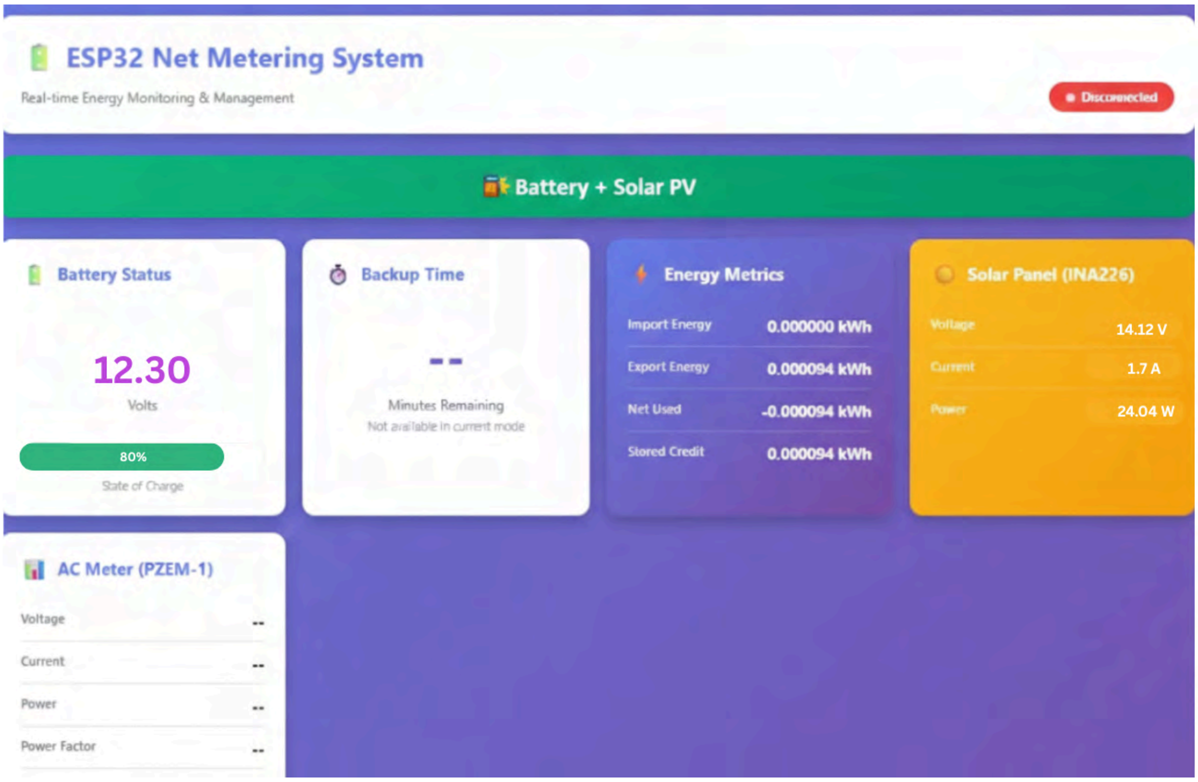

Figure 5.3.4: Battery + Solar PV supply

The mode of energy flow is a hybrid: PV will provide the maximum possible load, and the battery will cover the rest of the needs. The battery SOC also reduces over time in accordance with the deficit provided, and the IoT dashboard also shows real-time visualization of PV power, battery contribution, SOC, and total load that is provided. None of the energy is exported, and the accumulated credit does not change. Such a scenario indicates that the system is able to rationalize renewable generation and stored energy to ensure continuous load supply.

5.4 System Significance

The installed PV-battery net metering scheme proves to have a solid and workable draft to the contemporary renewable energy management. The importance of it is the automated control and precise energy monitoring along with the use of the IoT to monitor the data and visualize it, which will together represent a full educational and research platform. The major advantages of the system are:

- Safe Automated Energy Routing: The relay-controlled system will make sure that the energy flow is sent to the load, battery or grid correctly to avoid overcharging, deep discharge or overloading of components.

- Bidirectional Energy Measurement: The bidirectional energy meter allows the accurate calculation of imported and exported energy, which can be used when calculating the stored credit and the energy accounting.
- Dynamic Backup Computation: Predictive energy management is provided through dynamic calculation of battery backup time according to SOC, load demand and capacity, which is helpful to enhance the reliability of critical loads.
- Stored Credit Tracking: The system maintains a dynamic account of exported energy, being an accurate representation of credit accumulation, and is a simulation of real-world net metering setups.
- IoT-Based Monitoring and Remote Accessibility: ESP32-hosted dashboard allows monitoring the performance of the system, which includes load, PV generation, battery SOC, and energy values, thus improving transparency and simplifying the use.
- Research and Smart Grid Applications Scalability: The modular architecture can be scaled to larger systems, and supports experimentation, optimization research, and smart grid integration research.

Overall, the system serves as a complete educational and research tool, providing a practical demonstration of modern PV-battery-grid integration, net metering principles, and smart energy management strategies suitable for laboratory-based experimentation and study.

5.5 Conclusion

The deployed PV-battery net meter system proves to be an effective means of integrating the production of renewable energy, energy storage, and smart control to operate in a way that is safe, efficient, and autonomous. The system is a combination of relay-oriented control logic, battery state-of-charge (SOC) sensor, precise energy measurement, stored credit tracking, automatic computation of backup time, and internet-of-things (IoT)-based visualization, which makes it a reliable and research-ready framework.

As it was experimentally validated in various usage conditions, the system will be able to handle the energy flow with varying conditions such as low battery state, night backup, PV surplus export, and hybrid PV-battery load supply. The relay logic was consistent to limit the battery to over-discharge and provide un-interrupted load supply and to be able to export safe energy to the grid. Every imported and exported energy was correctly monitored in real time and the net energy contribution was calculated in stored credit, in accordance with practical net metering principles.

The system was also found to have dynamic backup estimation which gave predictive view of battery endurance within the SOC and the load demand. This aspect guarantees a better decision making to manage the load and make the system more practical in the case of the residential home, or microgrid researcher system. The ESP32-based IoT dashboard also enabled real-time monitoring, visualization, and analysis, which provided the user with real-time feedback on system performance, energy flow, and battery status, and helped carry out educational demonstrations and laboratory experiments.

In general, the system offers an all-encompassing educational and research environment to the analysis of PV-battery-grid integration, smart energy management, and optimization of renewable energy. It is very applicable in research works, laboratory experiments and smart grid simulations and is highly suitable due to its modular design, accurate energy accounting, and autonomy of control functions. The system is effective in validating CO8 and PO(c) goals, which is why it can be used in academic education and in the real world of renewable energy research.

Chapter 6: Project Sustainability and Impact Analysis

6.1 Introduction

The solutions to engineering problems in the modern world are supposed to not only be of technical performance but also of their extended influence on the society, environment, health, safety, legal regulations, and cultural acceptance. Projects in engineering which do not take into account these contextual factors are hardly accepted and well-maintained in the long-term. Thus, the impact analysis and sustainability assessment are two important elements of responsible engineering practice.

The chapter assesses the net effect of the recommended net zero metering and monitoring system to a grid-connected solar pv battery system based on the standpoints of society, health, safety, legal and cultural dimensions. Also, the environmental impacts and sustainability of the system in the long term are discussed. The evaluation makes sure that both the technical and non-technical problems are taken into account appropriately when coming up with a sustainable and socially responsible engineering solution.

6.2 Evaluation of the Solution Impact

6.2.1 Societal Impact

The impact of the proposed system on the society is positive since there is the encouragement of effective use of renewable energy resources. The system allows consumers to be much more aware of their energy consumption and promotes responsible action in their consumption by providing real-time access to monitor the solar power production and battery health, as well as grid interaction. Such awareness will result to the minimization of electricity wastage and improved energy management at household and community levels.

Moreover, the system encourages decentralization of power generation which can help in alleviating the pressure on the national grid during peak periods. Such systems have the potential to enhance energy resilience and reliability in areas with a low level of grid reliability. The project is also educational since it illustrates the concept of net metering thus adding to the knowledge sharing and skills acquisition among engineers and practitioners in the field of renewable energy.

6.2.2 Health Impact

Health wise, the system has an indirect positive influence on the health of the population since it promotes the use of clean energy. A decrease in air pollution due to a reduction in the use of electricity generated by the use of fossil fuel is linked to respiratory and cardiovascular diseases. The proposed system encourages healthier living conditions by encouraging people to use solar energy.

At operational level, the system aims at reducing health risks to the users. Wherever possible, low voltage parts are used and high voltage AC parts are appropriately isolated. Well defined operation indicators and regulated switching mechanism minimizes chances of exposure to dangerous electrical conditions accidentally.

6.2.3 Safety Impact

The design and implementation process considered safety issues. The Relay-based switching provides adequate electrical isolation between the battery and inverter connections and the grid connection, minimizing the short circuit, back-feed and electrical fault risks. The system design avoids the possibility of inadvertent and simultaneous unintended connections between grid supply and battery output which may cause damages to equipment or safety risk.

The selection of standard electrical components of the right ratings was done to make sure that it could operate safely under the anticipated load conditions. The modular design also enables ease of inspection, maintenance and troubleshooting, which further increases the safety of operation.

6.2.4 Legal Considerations

The introduction of the proposed system is conceptually consistent with the current net metering regulations and renewable energy policies in Bangladesh. Even though the prototype does not provide the use of utility-level protection devices and certified two-way meters, its logic of operation adheres to common net metering principles, including the regulated export and import of energy depending on the conditions of the system.

The project does not violate academic and ethical standards since it is created as an educational and research project but not directly implemented in the industry. Information gathered using the IoT platform will be restricted to performance indicators in a system, which will be in accordance to the data protection and privacy requirements.

6.2.5 Cultural and Ethical Aspects

The acceptance of renewable energy technologies is dependent on culture. The system proposed to be is straightforward, clear, and easy to use, which is applicable to users of different levels of technical skills. This ease helps to promote the culture of solar energy and intelligent monitoring devices.

Ethically, the project will focus on responsible engineering, such as user safety, information representation transparency, and truthful reporting of performance. The system promotes ethical use of energy by exposing users to the information about energy flow hence discouraging misuse and overuse.

6.3 Sustainability and Environmental Impact assessment

6.3.1 Environmental Impact

The suggested system also benefits the environmental sustainability as it supports the use of solar energy which is a renewable source of power that is also environmentally friendly. The system allows the unused solar energy to be released to the grid by means of enabling net metering. This will lead to the decrease in the total demand on the production of electricity using fossil fuel hence greenhouse gas emission and environmental pollution will be reduced.

The use of battery storage also enhances energy efficiency because any excess power of the solar energy is stored and can be used later, which eliminates the waste of unnecessary energy and increases the overall system performance.

6.3.2 Significance of Sustainable Engineering Solutions

There is a need to develop sustainable engineering solutions that should support the current energy demands without undermining the potential of the future generation to satisfy its needs. The solution to this need in the proposed system is a combination of renewable energy generation, intelligent monitoring, and efficient energy management.

The system shows how the small-scale renewable installations can be actively involved in the sustainable power systems. It underlines the significance of incorporation of monitoring and control to achieve maximum energy usage and long-term environmental payoff.

6.3.3 Sustainability Validation

The proposed system is sustainable with its design being low-cost, low-energy consumption, and low-adaptability. The ease of maintenance and replacement of the components can be guaranteed by the use of popular components like ESP32 microcontrollers, relay modules, and conventional electrical hardware, which can increase the life of the system operation.

Additionally, the system architecture can be expanded in the future with the introduction of new features, i.e., integration with smart meters, advanced data analytics, or other renewable sources. This scalability is important that the system is able to respond to changes in technology making its long-term viability.

6.4 Conclusion

In this chapter, the analysis of the proposed net zero metering and monitoring system impact and sustainability was done comprehensively. Social, health, safety, legal and cultural effects were evaluated, which proved that the system is socially positive, morally acceptable, and can be implemented at a laboratory level. The system environmental impact and sustainability were assessed, which proved that the solution promotes the renewable energy and sustainable engineering practices.

The proposed system, in general, can be seen as a responsible and sustainable engineering solution that can meet the current energy management needs and benefits the society and environment.

Chapter 7: Project Management in engineering

7.1 Introduction

Engineering project management is a systematic process that is aimed at delivering efficient, effective and within the required constraints in terms of time, resources and quality of engineering projects. Project management is essential in design based engineering projects, particularly in organizing the technical operations, risk management as well as in fostering co-ordination of team members. Monitoring and proper planning minimize uncertainty and enhance the chances of successful execution of a project.

The chapter outlines the project management approaches that will be implemented in the process of developing a proposed net zero metering and monitoring system of a grid-connected solar PV system. It describes the definition, planning and management of the project in various phases of academics. Besides, the chapter also compares project progress and planned schedule and talks about the personal and team inputs.

7.2 Engineering project definition, planning and management

7.2.1 Project Definition and Objectives

The project had a clear and measurable project definition, which was to design, simulate, implement, and validate an IoT based net zero metering and monitoring system. The purpose of the project was to prove the automated control of power flow between solar PV, battery storage, and grid connection and monitoring it in real-time via an IoT platform.

The scope of the project was well outlined to make it viable within the academic time frame. These were system modeling, simulation, hardware implementation, testing, data analysis and documentation. Such advanced features as utility-level grid synchronization, protection relays, and certified smart meters were purposely omitted to preserve academic focus and realistic feasibility.

7.2.2 Planning and Scheduling of projects

The project started off by creating a systematic project plan in accordance with the academic requirements and available resources. The project was broken down into systematic stages in line with the curriculum:

- **EEE499P (Proposal Phase):**
  Problem identification, literature, objective, system concept, and feasibility.

- **EEE499D (Design and Simulation Phase):**
  System architecture development, system simulation with use of suitable software tools, analysis of various design solutions and system optimization.

- **EEE499C (Final Implementation Phase):**
  Assembling hardware, creating the software, system integration, testing the system in the laboratory, creating the IoT dashboard, writing reports, and development.

An elaborate Gantt chart was drawn in order to schedule the time of activities of every task and milestone. It was found that task dependencies were necessary to prevent conflicts and provide a logical sequence of tasks. It was a well-organized planning that allowed efficient time and resources use.

7.2.3 Resource Management

Resource management was the process of identification and allocation of hardware or software and laboratory. The use of cost-effective components was made without interfering with the functionality and safety of the system. Simulation platforms and IoT services are software tools that were selected due to their availability and convenience.

The human resources were handled through task sharing among the team members on the basis of individual skills and interests. This guaranteed an equal balance in work loads and increased productivity.

7.2.4 Risk Identification and Mitigation

The identification of potential project risks was done at the stage of planning. These along with non-availability of components, hardware failures, software integration, and time schedule setbacks. In order to alleviate such risks, backup elements were pointed out, flexible schedules were followed, and regular tests were made to identify problems at an early stage. It was an active risk management method that reduced disruptions in the implementation of the project.

7.3 Project Progress Evaluation

7.3.1 Progress Monitoring Mechanism

This is the last mechanism, which involves an individual reviewing the current progress of the implementation in order to gauge the success of the implementation strategy employed.

Regular team meetings and milestone reviews were used to provide constant monitoring of the project progress. Progress reviews were done on weekly basis to compare the tasks done with the scheduled tasks. These reviews assisted in pointing out delays, technical and resource constraints at an early stage.

7.3.2 Schedule Compliance and Adjustments

The majority of the project activities were carried out in the planned time. The postponements at the implementation of hardware were minor because the hardware had to be tested through troubleshooting and verification of components. These slips were overcome by prioritizing work and working extra long hours without impacting on the completeness of the entire project.

7.3.3 Performance Assessment

All the academic stages such as proposal, design, simulation, and implementation are successfully completed, which testifies to successful project management. The last system was in compliance with functional requirements and worked as per the expectations in testing, which confirmed the success of the planning, coordination and implementation strategies.

7.4 Person and Group Performance

7.4.1 Responsibilities and Accountability at the individual level

The individual members of the team were given a certain set of roles and responsibilities according to the areas of expertise and areas of interest. There were literature review, simulation model, hardware configuration, firmware development, IoT configuration, testing, and documentation. Members of the team were accountable and professional.

7.4.2 Team Co-operation and Co-ordination

Team members in the various technical fields had to work consistently on the project. The frequent team meetings promoted the exchange of knowledge, group decision-making, and problem-solving. This joint strategy increased technical knowledge and led to general quality of the project. There should be effective communication among the participants.

7.4.3 Communication Effectiveness

There should be effective communication among the participants. Structured meetings, digital communication tools and shared documentation platforms were used in maintaining effective communication. Effective communication made every member of the team stay in line with project goals and schedules. Constant communication with the project supervisor gave advice and adherence to the academic standards.

7.4.4 Research, Documentation and Presentation Contributions

The members of the team were actively engaged in research and gathering information at the stage of the proposal and design. There were combined efforts in the preparation of simulation results, hardware documentation and final reports. Another aspect of teamwork involved the team members in the preparation of presentations and practice on the project defense provided there was a smooth flow of the technical information.

The team is effective, and fellow students can assess them.<|human|>7.5 Peer Evaluation and Team Effectiveness The team is effective, and other students can evaluate them.

Peer evaluation was also done to measure individual contributions, team work, communication skills and professionalism. The assessment plan promoted openness, impartiality, and respect towards one another in the team members. Positive feedback was used to enhance coordination and strengthen a positive team environment. Generally, there was a great deal of commitment, cooperation and shared responsibility among the team members during the project.

7.6 Conclusion

This chapter provided the detailed discussion of the engineering project management practices that were observed in the process of the given project implementation. The project has been successfully defined, planned, managed and assessed within the duration of time given and this is an indication of good project management skills. Also, group performance and individual performance, communication, and working together were considered. The project completion is a sign of the significance of organized project management and teamwork in the accomplishment of the complicated engineering goals.

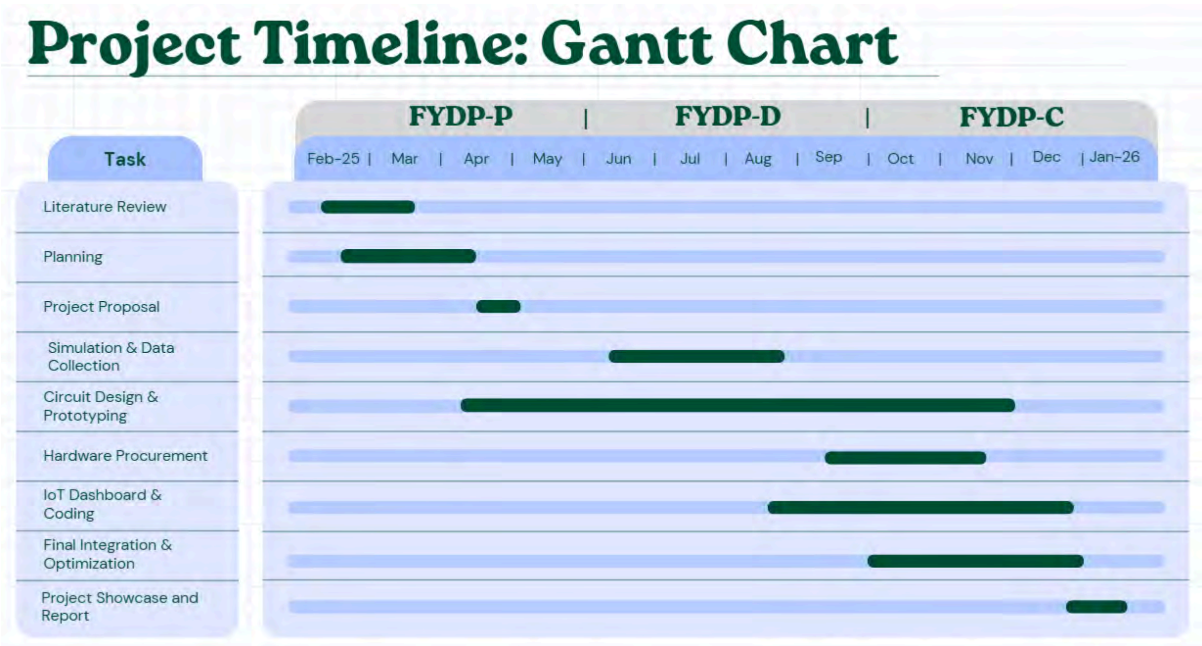

Fig 7.1. Project Gantt Chart

Chapter 8: Economical Analysis

8.1 Introduction

The economic viability of a grid-connected rooftop solar photovoltaic (PV) system is one of the key factors influencing the adoption and sustainability of the system in the long run. Although technical performance and system reliability are absolutely indispensable, such practical consideration as whether the proposed solution provides end users, utilities, and stakeholders with measurable financial returns is a critical determinant of practical deployment. Thus, an economic comparison is required to determine the cost and financial feasibility of the suggested centralized net meter monitoring and management system using IoT.

Considering the situation in Bangladesh, the rooftop solar systems are becoming a regulatory imperative and not an economic incentive. Consequently, most systems have not been fully exploited because there are no proper energy control and monitoring systems. The proposed system can enhance the value of the net metering participation by making the net metering process intelligent, enabling efficient use of the energy, and transparently monitoring the process to maximize the benefits of the return on investment (ROI).

In this chapter, the economic part of the proposed solution is assessed, which includes the analysis of the system costs, the advantages of operation, and the monetary results of the system lifecycle. The analysis has taken into account both the direct costs, i.e. the expenses of the hardware and installation, and indirect benefits, i.e. energy saving, less dependency on the grid, as well as, better utilization of the battery. The purpose of this chapter is to find out through the cost-benefit analysis and the financial evaluation the viability of the proposed system and its applicability into real-life usage in residential, commercial, and institutional applications.

8.2 Economic Analysis

The economic study of the proposed IoT-based net meter monitoring and management system is made on the basis of the actual cost of the components developed in the prototype development. In contrast to the massive commercial rooftop solar installations, the given project concentrates on the functional laboratory-scale prototype that will be used to prove the system architecture and control logic and verify the monitoring capability under the net metering conditions.

The overall project budget covers the cost of power electronics, sensing and measurement equipment, embedded controllers, energy storage, visualization equipment, structural material, and support that are needed to integrate the system and demonstrate its functionality. The calculation of all costs is done using local market prices in Bangladesh, which has unrealistic and verifiable financial evaluation.

8.2.1 Component-wise Cost Breakdown

Table 8.1. Component wise cost breakdown

Component Category	**Major Items Included**	**Cost (BDT)**
Power Conversion & Storage	PWM charge controller, inverter, lead-acid battery	3,740
Embedded Control & Processing	ESP32 DevKit, relays, headers	870
Energy Measurement & Sensors	PZEM-004T, INA226, LDR sensors	6,370
Display & Visualization	LCD display, poster materials	645
Structural & Mechanical	PCB board, acrylic board, clamps, screws	1,380
Wiring & Accessories	Wires, capacitors, tape, shrink tubes	605
Documentation & Presentation	Poster, brochure, printing	700
Total System Cost		**14,140 BDT**

Total Prototype Cost = 14,140 BDT

8.2.2 Cost Interpretation

Their distribution of costs indicates that measurement and sensing parts consume the most significant part of the overall cost. This is not surprising, since precise bi-directional monitoring of energy is an inalienable requirement of net metering systems. Granting grid import and grid export measurements separately made possible by the usage of the dual PZEM-004T modules and INA226 sensors is a big step toward the higher accuracy and reliability of the system.

The internal control and processing is not very expensive because the use of ESP32 microcontroller has been chosen as it combines processing power, wireless connectivity, and enough I/O interfaces into a unified platform. This proves the cost-efficiency of the IoT-based solutions to smart energy systems.

The inverter and battery add a significant part of the cost, but it is necessary to justify the bidirectional energy flow and the version of the operation at realistic conditions. The cost of structures and auxiliary costs are also reduced by means of locally available materials, so that the prototype itself can be used for academic and demonstration purposes.

8.2.3 Prototype Cost vs Real-World Deployment Perspective

It is noteworthy that this cost analysis is not a full-scale residential or commercial rooftop solar system, it is a prototype-scale. In real-world deployment:

- The larger capacity systems would share sensors and controllers.
- The unit cost would also be less as a result of bulk buying.
- Hardware cost monitoring is insignificant to PV modules cost.

Thus, the economic usefulness of the proposed system is not in the form of minimizing the cost of the hardware, but in the form of utilizing the installed solar capacity, avoiding energy losses, and having the opportunity to take part in net metering programs.

8.3 Cost Benefit Analysis

The cost benefit analysis will compare the system cost and the possible economic benefits realized by saving on the energy and selling to the grid. When applied in the real world, surplus solar energy into the grid via net metering can minimize the electricity bills.

The prototype assumes a simulated source of the sun and load on the grid, but the logic of operation resembles the behavior of net metering. The system aims at lessening reliance on grid electricity by focusing on solar and battery power during peak load periods. The excess energy is sold to the grid when there is excess power and this is a financial potential.

With time these savings may recoup the original investment cost. The realtime monitoring is also added to enable users to track the energy usage and generation which results in the better management of energy and additional cost savings.

Table 8.2. Cost Benefit Analysis

Economic Factor	Our System	Typical Commercial System (e.g., Huawei/Victron)	Key Advantage
1. Upfront Capital Cost	**~12,000 BDT** (Core monitoring & control unit)	50,000 - 500,000+ BDT (Full system import)	**>75% lower cost** makes smart technology accessible for local budgets.
2. Policy & ROI Alignment	Engineered for **Bangladesh's Net Metering Guideline 2018**, automating credit calculation.	Generic global product; may not automate local billing rules.	**Directly converts policy into savings**, ensuring faster and more reliable payback.
3. Operational Cost & Control	**No recurring cloud fees**; open-source local control.	Often requires vendor services, proprietary parts, or cloud subscriptions.	**Eliminates long-term vendor lock-in**, giving users full control and reducing lifetime cost.
4. Setup & Maintenance	**Modular design**; can be serviced by local technicians with standard components.	Professional installation required; often needs specialized vendor support.	**Builds local capacity** and reduces downtime with easier, cheaper repairs.
5. Future-Proofing & Scalability	**Hackable platform (ESP32/ML)** ready for P2P trading or grid-service upgrades.	Closed ecosystem; upgrades depend on vendors and are often costly.	**Protects investment** by adapting to future energy markets and tariff changes

8.4 Evaluation of Economic and Financial Aspects

Financially, the system proposed has long-term benefits even though the cost of implementation is moderate. Battery storage enhances reliability of the system when power is unavailable and when demand is at the peak which adds indirect economic value.

Recurrent costs are reduced due to the use of cheap embedded hardware and open-source application programs. Internet surveillance does not need costly proprietary systems and hence the cost of maintenance is insignificant.

Nevertheless, the price of changing the battery and the wear and tear of the components with time are important factors. These aspects influence long-term financial performance and have to be thought of in reality implementation. In spite of these shortcomings, the general financial prospect is good, particularly in those areas that are experiencing increased electricity charges and favorable net metering guidelines.

Financially, the proposed system is valuable, as the solar energy use will be enhanced with the aid of real-time monitoring and automated control of power flow in relieve with the solar energy and automated net metering-compatible energy tracking with the help of automated relays. Although the prototype cost will rely on the implementation at the laboratory level, it is possible to calculate the payback by mapping the monitoring/control unit to a roof top PV intervention.

8.4.1 Simple Payback Period Calculation

Following the economic evaluation method provided in the reference documentation, the **simple payback period** can be calculated as:

$$Simple\ Payback\ Period\ =\ \frac{\Delta P}{\Delta A}$$

Where,

- Δ P = Initial cost of the monitoring and management system (BDT)
- Δ A = Estimated annual savings achieved due to improved utilization and reduced grid import (BDT/year)

For this project, the total prototype cost is:

$$\Delta P = 14{,}140 \text{ BDT}$$

The annual savings will be determined by the size of the PV system and the extent to which supplementary usable energy is achieved as a result of improved monitoring/control (e.g. reduced wastage, better schedule of battery use, and effective tracking of export). Annual savings are calculated based on real environmental scenarios since the prototype employs a simulated source of solar.

Annual Savings (Deployment Mapping) Assumptions Made to ensure that the analysis remains conservative and justifiable:

- A 1 kW PV system on the rooftop in Bangladesh is estimated to generate about 4 kWh/day (average), producing:

$$E_{1KW} \approx 4 \times 365 = 1460 \text{ kWh/year}$$

- A 3 kW rooftop PV system produces approximately:

$$E_{3KW} \approx 3 \times 1460 = 4380 \text{ kWh/year}$$

- Average effective electricity value is taken as 10 BDT/kWh (mid-range representative of household slab tariffs).
- The proposed system enhances the use of solar energy capture (usable) by 15 percent due to the better management of energy flow and awareness by the users (conservative assumption in making the improvement monitoring-driven).

Thus, annual savings:

$$\Delta A = 0.15 \times E \times 10 \text{ BDT/kWh}$$

Table 8.3. Payback Period Estimation (Two Realistic Deployment Scenarios)

Scenario	**PV Capacity**	**Annual PV Energy, (E) (kWh/yr)**	**Improved Utilization (15%) (kWh/yr)**	**Annual Savings, (\Delta A) (BDT/yr)**	**Initial Cost, (\Delta P) (BDT)**	**Payback Period (\Delta P/\Delta A)**
Case 1 (Residential)	1 kW	1460	219	2,190	14,140	**6.46 years**
Case 2 (Mandate/3-phase)	3 kW	4380	657	6,570	14,140	**2.15 years**

The finding is that as the size of the system increases, payback period increases substantially. In a 1 kW residential rooftop, the monitoring and control system cost is paid back in one of about 6.5 years largely due to the fact that the annual energy offset and bill offset is modest. But on a 3 kW rooftop PV system (in keeping with the mandatory requirement of three-phase connection), the same monitoring system will have a payback of about 2.2 years, as higher yearly energy throughput will boost the monetary compensation of enhanced usage and export awareness.

Hence, the intended system will be economically more favorable in commercial/institutional installations and larger rooftop systems with high energy throughput and monitoring with quantifiable savings in operations.

8.4.2 Sensitivity Analysis of Payback Period

The proposed net meter monitoring and management system is economic because the level of real-time monitoring and intelligent control will enhance the use of solar energy. This improvement can depend upon user behavior, load patterns and system scale hence sensitivity analysis is used to determine the impacts of the increase in utilization improvement to the payback period.

Three realistic scenarios of improvement have been taken into consideration, namely, 10, 15 and 20 percent, which are conservative, nominal and optimistic cases, respectively. Simple payback period is determined with the help of the same formulation that was mentioned earlier:

$$Simple\ Payback\ Period\ =\ \frac{\Delta\,P}{\Delta\,A}$$

Where:

- Δ P = 14,140) BDT (initial system cost)
- Δ A varies according to improved energy utilization

Table 8.4: Sensitivity Analysis of Payback Period for Different Utilization Improvements

Scenario	**PV Capacity**	**Utilization Improvement**	**Additional Energy Utilized (kWh/year)**	**Annual Savings (BDT/year)**	**Payback Period (years)**
Residential	1 kW	10%	146	1,460	**9.68**
Residential	1 kW	15%	219	2,190	**6.46**
Residential	1 kW	20%	292	2,920	**4.84**
Mandated / Commercial	3 kW	10%	438	4,380	**3.23**
Mandated / Commercial	3 kW	15%	657	6,570	**2.15**
Mandated / Commercial	3 kW	20%	876	8,760	**1.61**

The sensitivity analysis indicates that payback period is greatly sensitive to the capacity of the system and level of energy consumption improvements that monitoring and control allows through the system. In residential systems of lower powers (1 kW), payback periods lie between 9.7 years at the conservative end to 4.8 years at the optimistic end. Conversely, in the case of larger rooftop installations (3 kW), the payback period is less than 3.5 years assuming a conservative set of assumptions and will decrease to about 1.6 years assuming higher use enhancements.

This discussion establishes the fact that the proposed system can be economically sound over a variety of practical operating scenarios. The monitoring and management system is financially feasible even with conservative assumptions, especially with commercial, institutional and required rooftop solar installations where throughput of energy is greater.

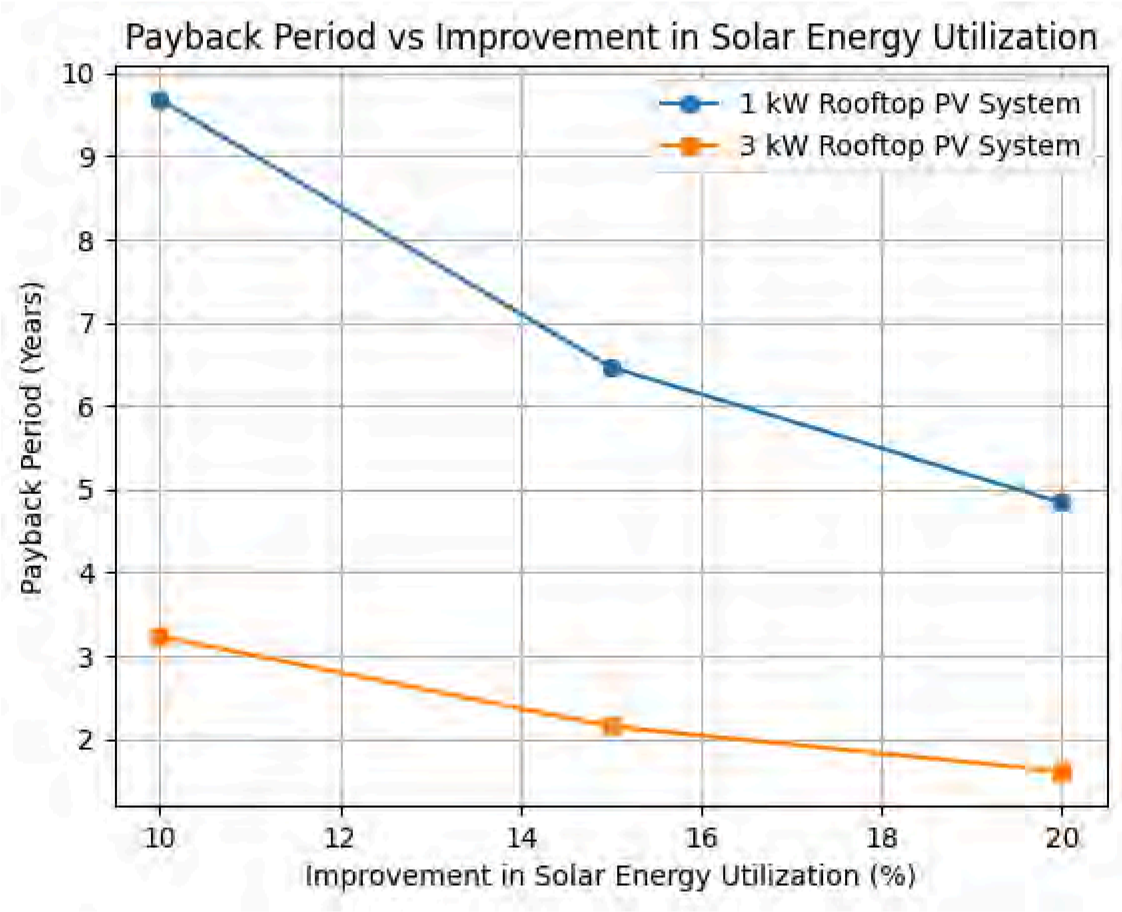

Figure 8.1. Payback period variation with improvement in solar energy utilization for 1 kW and 3 kW rooftop PV systems.

As shown in Figure 8.1, the payback period is reduced in a nonlinear fashion with the increase in the solar energy usage. The decrease is stronger with the larger capacity systems and this implies that the new monitoring and management system is more financially efficient in commercial and required rooftop PV systems.

8.5 Conclusion

The proposed net zero metering and monitoring system has an economical analysis and was explained in this chapter. It is revealed that small scale applications and education The system is cost-effective. The advancements in energy saving, decreased grid reliance, and enhanced monitoring are worth the investment. The economic assessment of the proposed system shows that the system is economically viable and replicable to practice despite it being an experimental version.

Chapter 9: Ethics and Professional Responsibilities

9.1 Introduction

Professional responsibility and ethics are a key part of the engineering practice, especially in the context of projects that involve electrical power systems and infrastructural elements of the population. The engineers are required to make sure that their designs are safe, reliable, environmentally friendly, and in line with the relevant standards and policies. The ethical factors in relation to grid-connected renewable energy systems are also viewed not only in the technical context of what is correct, but also within regulatory and legal frameworks, data representation and the effect on society.

In this chapter, the ethical issues and professional responsibilities related to the proposed net zero metering and monitoring system are discussed. It also outlines how pertinent national and global standards have been taken into account to make sure that the engineering practice is responsible.

9.2 Identification of Ethical Issues and Professional Responsibilities

There were a number of ethical concerns that were raised when designing and implementing this project. One of the greatest concerns is electrical safety because the system is associated with converting, storing, and interacting energy with the grid. Wrong management of level of voltages or poor isolation might lead to user and equipment hazards.

Another ethical issue is data accuracy and transparency. Misleading or inaccurate information, since the system offers users real-time monitoring of the imported and exported power, may impact the user decision-making and trust. The engineers should make sure that the data monitored should show the real system behavior and that limitations should be well explained.

The professional responsibility also demands that the regulatory and policy frameworks are adhered to. There are guidelines and mandates in Bangladesh that control the deployment of renewable energy and net metering. Such neglect might lead to non-conformity or unsafe system designs. Another aspect of ethics is environmental responsibility since the project disseminates the use of renewable energy and decreased fossil fuel reliance.

9.3 Application of Ethical Issues and Professional Responsibilities

9.3 Application of Ethical Issues and Professional Responsibilities

The principles of ethics and professional responsibilities were actively utilized during the project by balancing the system design and operation with the agreed standards and policies.

1. **National Standards and Policies**

The **Net Metering Guideline 2018 issued by the Bangladesh Energy Regulatory Commission (BERC)** was considered as a key reference. This standard establishes conditions of bi-directional energy metering, interconnection processes and billing systems. The project will focus on the implementation of an intelligent monitoring system that is able to measure imported and exported electricity which is directly related with Section 4, 5 and 6 of the guideline. Ethical responsibility in data reporting and transparency of users were prioritized by the time-synchronized logging and accurate sensing.

The **Ministry of Power, Energy and Mineral Resources** has developed the **Renewable Energy Policy 2020** that favors the decentralized generation and storage of renewable energy. The project contributes to the goal of this policy by making it possible to use renewable energy as a more significant part of the national energy mix by allowing local generation of solar energy along with battery storage and intelligent monitoring. This orientation demonstrates that there is ethical accountability to the goals of environmental sustainability and national development.

The **Rooftop Solar Mandate 2023** is another decision that has reinforced the relevance of this project. The requirement is the installation of rooftop solar in the new buildings where the size of the roof is adequate. Due to the scalable and modular structure of the suggested system, it is possible to adjust to various buildings and load constraints. As a professional responsibility, the project offers a sound technical solution that can assist the owners of the buildings to adhere to the legal requirements and still operate safely.

2. **International Standards**

At the global level, **IEEE 1547-2018** was taken into consideration as safe interconnection between distributed energy resources and the grid. Even though the prototype is a controlled environment, the relay logic, the synchronization method, and the ideas of fault-handling are based on the notion of this standard. Adherence to **IEEE 1547** facilitates ethical responsibility because it focuses on grid safety and interoperability.

IEC 62109-1 and **IEC 62109-2** that stipulate the safety of power converters refer to issues of electric shock, thermal failure, and abnormal operating conditions. Even in an academic context, the responsible practice in design is shown by the use of certified elements and safe voltage levels in the prototype.

IEC 62541 (OPC-UA) was also a possible communication standard to be used in the future. Although the current prototype does not extend to it, the presence of this element in the design roadmap is an indication of professional accountability to interoperability, cybersecurity, and system integration over the long term in either an industrial or smart campus setting.

9.4 Conclusion

In this chapter, the ethical considerations and professional liabilities that will be involved in the proposed net zero metering and monitoring system were discussed. The project is in line with the accepted ethical engineering practices by focusing on safety, accuracy of data, environmental sustainability, and adherence to the law. Professional integrity of the work is also enhanced by taking into consideration the national policies and international standards. On the whole, the given project shows positive engineering conduct and leads to sustainable and responsible implementation of renewable energy.

Chapter 10: Conclusion and Future Work

10.1 Conclusion

The project aimed at the design, simulation and hardware realization of a net zero metering and monitoring of a grid connected battery of solar photovoltaic (ESP32 microcontroller) system. The main aim was to show the basic functioning of net metering in a low-cost and basic and safe laboratory-level setting. The main concepts of the bidirectional power flow, the automated selection of energy sources, and real-time monitoring of the system were successfully proven with the help of the successful completion of the project.

There were two conditions under which the system was to work. The load was supplied by renewable sources and the surplus energy was sold to the grid when adequate solar and battery energy was accessible. In cases where there was a shortage of battery energy, the system would automatically revert to grid power without unwanted export of power. This operation logic was executed through the switching of relay-based switching that proved to be reliable in the course of experimental studies.

The ESP32 microcontroller was coupled with a locally hosted web page to attain real-time monitoring. This enabled users to visualize operating modes, relay functions and state of systems in real time. This type of monitoring enhances awareness in users on the energy flow and the behaviour of the system, which is lacking in small scale solar installations. Simulation as well as hardware testing results proved that the system achieves its goals and works properly within the scope as designed.

Although these have been achieved, a number of limitations were found when developing and testing. The significance of the greatest weakness is the use of DC power supply to model the solar PV source. The method fails to consider real world changes like variations in solar irradiance, temperature and partial shading that have a huge effect on the actual performance of the PV. Additionally, the system lacks grid synchronization, anti-islanding protection, certified bidirectional energy meters, all of which are required by utility-level grid-connected solar systems.

The relay modules that have been used in the demonstration were only for switching the power sources but they are not used in high-power or high-frequency switching. Moreover, the monitoring system will be based on constant internet access to send the data to the IoT platform. This reliance can lower the responsiveness of the system in places where network coverage is compromised. The estimation of state-of-charge of the battery is also simplified and does not take into account the long-term effects on the battery (aging), losses in its efficiency or the electrochemical behavior of the battery.

These restrictions were embraced with the purpose of being safe, affordable and viable in an academic environment. Although the system is not meant to directly be deployed to utility networks, it can be used as an efficient educational and experimental tool to learn about the concepts of net metering, smart energy management and IoT-based monitoring of the operation of renewable energy systems.

10.2 Future Work

This project can be developed in a number of directions in the future. The introduction of real solar PV panels to test the system performance under real environment would be one of the relevant improvements. This would enable better evaluation of the energy production, battery recharge pattern, and grid interchange.

The system would be more applicable to real world use by the use of certified and approved bidirectional energy meters and using the right grid synchronization techniques. A potential way to increase the speed, reliability and scalability of switching might be to replace relay-based switching with solid-state switching. Greater battery management methods, such as precise estimation of state-of-charge and battery healthness, can also be included.

Software wise, the system may be considered by creating a specific web or mobile application that will lessen the reliance on third-party IoT solutions. The features of data logging and analytics would be implemented to track the long-term energy use and export trends. Machine learning applications can also be used to predict energy production and make the best decisions on power flows.

Such improvements in future would remarkably boost the performance and practicability of the system, and would not lose the main goal of facilitating the integration of renewable energy and intelligent management of energy.

Chapter 11: Identification of Complex Engineering Problems and Activities

In this chapter, the features of complex engineering problems and complex engineering activities that the suggested net zero metering and monitoring system tackles are identified. The classification is according to the Washington Accord framework and depicts scope, depth, and practicality of the project.

11.1 Identification of Attributes of Complex Engineering Problems (EP)

Table 11.1 shows the selected attributes of complex engineering problems that are applicable to this project. Only the attributes marked with tick signs were considered relevant.

Table 11.1: Attributes of Complex Engineering Problems (EP)

Attributes	**Put tick (√) as appropriate**
P1: Depth of knowledge required	√
P2: Range of conflicting requirements	
P3: Depth of analysis required	√
P4: Familiarity of issues	√
P5: Extent of applicable codes	
P6: Extent of stakeholder involvement and needs	
P7: Interdependence	√

11.2 Reasoning on How the Project Addresses Selected EP Attributes

P1: Depth of Knowledge Required

The project involves expertise in various fields of engineering such as renewable energy systems, power electronics, embedded systems and communication over the internet of things. The combination of multiple subsystems, such as solar input, battery storage, inverter work and the monitoring with the ESP32, is a complicated process that requires more than just simple circuit design. This multidisciplinary need establishes that the issue entails high levels of depth of knowledge.

P3: Depth of Analysis Required

The technical analysis was required in detail in order to have proper system behavior under the varying operating conditions. Simulation and logical analysis was done to check power flow at charging, discharging, and grid interaction modes of the battery. Safety, efficiency, and functional reliability were taken together and that demanded the system level analysis instead of component level analysis.

P4: Familiarity of Issues

The issue that this project will focus on is similar to some of the challenges that continue to happen in the rooftop solar and net metering industry in Bangladesh. The lack of monitoring, ineffective use of energy, and the ignorance of users are typical problems in the existing systems. The emphasis on these practical and real issues makes the project highly in touch with established engineering issues and in need of careful technical solutions.

P7: Interdependence

The system is highly interdependent in terms of hardware and component control, IoT, and monitoring and control and regulatory concepts like net metering. The subsystems: relays, battery, inverter output and monitoring dashboard are closely interconnected. The performance of an overall system is directly influenced by a fault or a change in a given subsystem, which meets the interdependence attribute.

11.3 Identification of Attributes of Complex Engineering Activities (EA)

In table 11.2 shows the attributes of complex engineering activities which are applicable in this project are identified.

Table 11.2: Attributes of Complex Engineering Activities (EA)

Attributes	**Put tick (√) as appropriate**
A1: Range of resource	√
A2: Level of interaction	√
A3: Innovation	
A4: Consequences for society and the environment	√
A5: Familiarity	

11.4 Reasoning on How the Project Addresses Selected EA Attributes

A1: Range of Resource

The project required use of numerous hardware and software resources, such as simulation of solar supplies, battery, inverters, relay modules, sensors, microcontrollers and cloud-based IoT systems. The control of these various resources and the coordinated working of the same is an intricate engineering process and not an easy piece of work in a lab.

A2: Level of Interaction

It has a high level of interaction among the various system elements, which include power electronics hardware and embedded control software. Also, there is observation of interaction with the end-user requirements and utility grid concept, although the system is introduced on a prototype level. This interaction between layers is up to the engineering activity complexity.

A4: Consequences for Society and the Environment

The project would encourage efficient utilization of renewable energy resources and the concepts of net metering which may eliminate reliance on fossil fuels. Better monitoring will enable users to know their energy consumption and generation patterns resulting in economic and environmental gains. Such secondary implications warrant the social and environmental impact of the engineering operation.

11.5 Conclusion

This chapter discussed the specified and discussed the characteristics of complex engineering problems and activities that are covered by the proposed net zero metering and monitoring system. The attributes of EP and EA that have been selected are met by means of multidisciplinary application of knowledge, system level analysis, integration of resources and focusing on the societal impact. This analysis has established that the project is in good agreement with the Washington Accord structure and is a valuable and significant complex engineering project.

References:

[1] International Energy Agency, World Energy Outlook 2023, Paris, France: IEA, 2023.

[2] T. Ackermann, G. Andersson, and L. Söder, “Distributed generation: A definition,” Electric Power Systems Research, vol. 57, no. 3, pp. 195-204, 2001, doi: 10.1016/S0378-7796(01)00101-8.

[3] Bangladesh Energy Regulatory Commission (BERC), Net Metering Guideline 2018; Ministry of Power, Energy and Mineral Resources, Renewable Energy Policy 2020; Government of Bangladesh, Rooftop Solar Mandate, Oct. 2023.

[4] A. R. Al-Ali, I. A. Zualkernan, and M. Rashid, “A smart home energy management system using IoT and big data analytics,” IEEE Transactions on Consumer Electronics, vol. 63, no. 4, pp. 426-434, Nov. 2017, doi: 10.1109/TCE.2017.015014.

[5] Government of Bangladesh, Rooftop Solar Mandate for New Buildings, Ministry of Power, Energy and Mineral Resources, Oct. 2023.

[6] Sustainable and Renewable Energy Development Authority (SREDA), National Database of Renewable Energy, Dhaka, Bangladesh, 2023.

[7] Bangladesh Energy Regulatory Commission (BERC), Net Metering Guideline 2018, Dhaka, Bangladesh, 2018.

[8] International Renewable Energy Agency (IRENA), Integrating Distributed Solar PV in Urban Power Systems, Abu Dhabi, UAE, 2022.

[9] World Bank, Unlocking Rooftop Solar in South Asia: Barriers and Policy Solutions, Washington, DC, USA, 2021.

[10] M. Bijoy, S. K. Das, and A. H. Khan, “IoT-based smart net metering system for distributed solar PV,” IEEE Access, vol. 12, pp. 34521-34534, 2024, doi: 10.1109/ACCESS.2024.3374215.

[11] A. Jamal, M. R. Islam, and S. Ahmed, “Design and implementation of a low-cost IoT-enabled smart energy meter,” International Journal of Electrical Power & Energy Systems, vol. 118, pp. 105743, 2020, doi: 10.1016/j.ijepes.2019.105743.

[12] M. Rind, A. Jamal, and S. A. Khan, “Smart energy metering using IoT for real-time monitoring and analysis,” IEEE Access, vol. 11, pp. 14512-14525, 2023, doi: 10.1109/ACCESS.2023.3248759.

[13] P. Pathare, R. Kulkarni, and S. Patil, “Modeling and simulation of IoT-enabled net metering system for renewable energy integration,” Energy Reports, vol. 10, pp. 1123-1132, 2024, doi: 10.1016/j.egyr.2024.01.087.

[14] Bangladesh Energy Regulatory Commission (BERC), Net Metering Guideline 2018, Dhaka, Bangladesh.

[15] International Energy Agency, Digitalization and Energy, Paris, France: IEA, 2017.

[16] International Renewable Energy Agency (IRENA), Innovation Landscape for a Renewable-Powered Future: Digital Solutions for Energy, Abu Dhabi, UAE, 2019.

LOGBOOK (FYDP-P)

	Final Year Design Project (P) Spring 2025		
Student Details	**NAME & ID**	**EMAIL ADDRESS**	**PHONE**
Member 1	MD. Mustaq Mujahid	md.mustaq.mujahid@g.bracu.ac.bd	01627367246
Member 2	Md. Shakil Hasan	md.shakil.hasan@g.bracu.ac.bd	01648583510
Member 3	Ajmain Iqtiad	ajmain.iqtiad@g.bracu.ac.bd	01537458352
Member 4	Md. Zubayet Hossain Zuboraz	zubayet.hossain.zuboraz@g.bracu.ac.bd	01314027621
ATC Details:			
ATC 10			
Chair	Shahidul Islam Khan, PhD	shahidul.khan@bracu.ac.bd	
Member 1	Mohammad Zunaed	mohammad.zunaed@bracu.ac.bd	
Member 2	Saad Mahbub Chowdhury	saad.chowdhury@bracu.ac.bd	

Date/Time/ Place	**Attendee**	**Summary of Meeting Minutes**	**Responsible**	**Comment by ATC**
19.02.2025	Mustaq Shakil Ajmain Zubayet	Brainstormed project ideas and short-listed based on impact.	Mustaq	Narrower focus needed. Several ideas were too broad.
		Discussed feasibility of rooftop solar in urban areas.	Shakil	Good insight. Consider technical limitations early on.
		Finalized project title and approach direction.	Ajmain	Title is clear and aligns with national energy goals.
26.02.2025	Mustaq Shakil Ajmain Zubayet	Drafted the initial problem statement.	Mustaq	Needs stronger linkage to research gap.
		Began literature review assignment.	Ajmain	Make sure to cite peer-reviewed sources.
		Reviewed rooftop solar policy in Bangladesh.	Shakil	Good policy reference. Include official document links.
05.03.2025	Mustaq Shakil Ajmain Zubayet	Finalized scope and objectives.	Mustaq	Well-structured objectives. Add quantitative metrics.
		Compared cloud vs. edge architectures.	Ajmain	Add diagrams for clarity.
		Assigned team responsibilities for proposal sections.	Zubayet	Clear division of tasks. Ensure timely updates.

12.03.2025	Mustaq Shakil Ajmain Zubayet	Selected hardware components (ESP32, MPPT, sensors).	Shakil	Confirm datasheet specs before finalizing.
		Drafted system block diagram.	Zubayet	Incomplete. Lacks grid interface. Revise.
		Determined environmental sensors needed.	Ajmain	Good, but validate sensor range and accuracy.
17.03.2025	Mustaq Shakil Ajmain Zubayet	Finalized multiple design approaches (cloud/edge).	Mustaq	Strong structure. Evaluate trade-offs further.
		Began drafting system schematics.	Zubayet	Schematics still lack MPPT detail.
		Discussed microcontroller-pin mapping.	Shakil	Consider GPIO limitations of ESP32.
09.04.2025	Mustaq Shakil Ajmain Zubayet	Developed full methodology including control logic.	Ajmain	Strong work. Add test cases for logic validation.
		Identified software tools and libraries to be used.	Mustaq	Document license status (open source/paid).
		Decided dashboard layout structure.	Shakil	Mobile view not addressed. Revise UI wireframe.
17.04.2025	Mustaq Shakil Ajmain Zubayet	Progress Presentation	Zubayet Ajmain Shakil Mustaq	Rethink the complex Engineering attributes.

23.04.2025	Mustaq Shakil Ajmain Zubayet	Integrated all sections of proposal into a single document.	Ajmain	Proposal lacks standard citation format. Fix urgently.
		Cross-checked against rooftop mandate compliance.	Zubayet	Good compliance check. Attach supporting reference.
08.05.2025	Mustaq Shakil Ajmain Zubayet	Final Presentation	All Members	What are the benefits of the user.
14.05.2025	Mustaq Shakil Ajmain Zubayet	Final proposal submitted. Checklist completed.	All members	Submission confirmed.

LOGBOOK (FYDP-D)

	Final Year Design Project (D) Summer 2025		
Student Details	**NAME & ID**	**EMAIL ADDRESS**	**PHONE**
Member 1	MD. Mustaq Mujahid (22121030)	md.mustaq.mujahid@g.bracu.ac.bd	01627367246
Member 2	Md. Shakil Hasan (22121067)	md.shakil.hasan@g.bracu.ac.bd	01648583510
Member 3	Ajmain Iqtiad (22121017)	ajmain.iqtiad@g.bracu.ac.bd	01537458352
Member 4	Md. Zubayet Hossain Zuboraz (22121067)	zubayet.hossain.zuboraz@g.bracu.ac.bd	01314027621
ATC Details:			
ATC 10			
Chair	Shahidul Islam Khan, PhD	shahidul.khan@bracu.ac.bd	
Member 1	Mohammad Zunaed	mohammad.zunaed@bracu.ac.bd	
Member 2	Saad Mahbub Chowdhury	saad.chowdhury@bracu.ac.bd	

Date/Time /Place	Attendee	Summary of Meeting Minutes	Responsible	Comment by ATC
02.07.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Search for better sensors. 2. Divide in sub-systems for simulation purposes. 3. Simulate PV system in MATLAB.	Task 1: Zubayet Task 2: Ajmain Task 3: Mustaq , Shakil	Task 1: completed. Task 2: partially completed. Task 3: partially completed..
09.07.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Work parallely with different sub systems 2. Design and Simulate single PV array. 3. Add Inductive/capacitive load in the MATLAB simulation. 4. Try to simulate on Proteus.(2 diode model)	Task 1: Zubayet Task 2: Ajmain Task 3:Mustaq Task4: Shakil	Task 1: partially completed. Task 2: completed. Task 3: partially completed. Task4: Completed
16.07.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Single direction simulation 2. Backup battery (install proteus library) 3. Charge control(inside the smart inverter) 4. Blynk dashboard creation.	Task 1: Zubayet Task 2: Ajmain Task 3: Shakil Task4: Mustaq	Task 1: partially completed. Task 2: partially completed. Task 3: partially completed. Task4: partially completed
23.07.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Putting the calculated value and cross check the value. 2. Adding DC/AC load 3. Making block wise subsystem in Simulink	Task 1: Zubayet Task 2: Shakil Task 3: Ajmain, Mustaq	Task 1: completed. Task 2: completed. Task 3: completed.
29.07.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Single direction simulation 2. Backup battery (install proteus library) 3. Charge control(inside the smart inverter) 4. Blynk dashboard creation.	ask 1: Zubayet Task 2: Ajmain Task 3: Shakil Task4: Mustaq	Task 1: completed. Task 2: completed. Task 3: completed. Task4: completed
03.08.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Add Inductive/capacitive load in the MATLAB simulation.	Task 1: Zubayet, Mustaq Task 2: Ajmain, Shakil	Task1: completed Task2: completed

		2. Design and Simulate single PV array.		
06.08.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Complete validation of subsystem simulations. 2. Integrate PV and battery model. 3. Verify inverter efficiency.	Task 1: Zubayet Task 2: Shakil Task 3: Ajmain, Mustaq	Task 1: completed. Task 2: completed. Task 3: completed.
09.08.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Worke on Blynk dashboard visualization. 2. Conducte data logging tests. 3. Identifie bugs in Proteus simulation.	Task 1: Zubayet Task 2: Shakil Task 3: Ajmain, Mustaq	Task 1: completed. Task 2: partially completed. Task 3:completed.
13.08.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Debugge sensor calibration issues for simulations. 2. Improve MPPT algorithm. 3. Update simulation parameters with real data.	Task 1: Shakil Task 2: Zubayet Task 3: Ajmain	Task 1: completed. Task 2: completed. Task 3: completed.
20.08.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Conduct system-level integration. 2. Finalize contingency plan. 3. Prepare for presentation.	Task 1: Zubayet Task 2: Shakil, Task 3: Mustaq, Ajmain	Task 1: completed. Task 2:completed. Task 3:completed.
24.08.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Meeting before final presentation: suggested some minor changes.	Task1: All	Task:1 Completed
27.08.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Updates about the meetings: Prepare final report	Task 1: All	
10.09.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Updates of the final report and presentation slide.	Task 1: All	Task 1: completed

LOGBOOK (FYDP-C)

	Final Year Design Project (C) FAll 2025		
Student Details	**NAME & ID**	**EMAIL ADDRESS**	**PHONE**
Member 1	MD. Mustaq Mujahid (22121030)	md.mustaq.mujahid@g.bracu.ac.bd	01627367246
Member 2	Md. Shakil Hasan (22121067)	md.shakil.hasan@g.bracu.ac.bd	01648583510
Member 3	Ajmain Iqtiad (22121017)	ajmain.iqtiad@g.bracu.ac.bd	01537458352
Member 4	Md. Zubayet Hossain Zuboraz (22121067)	zubayet.hossain.zuboraz@g.bracu.ac.bd	01314027621
ATC Details:			
ATC 10			
Chair	Shahidul Islam Khan, PhD	shahidul.khan@bracu.ac.bd	
Member 1	Mohammad Zunaed	mohammad.zunaed@bracu.ac.bd	
Member 2	Saad Mahbub Chowdhury	saad.chowdhury@bracu.ac.bd	

Date/Time /Place	Attendee	Summary of Meeting Minutes	Responsible	Comment by ATC
07.10.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Reviewed approved FYDP-D design and simulation results 2. Mapped simulation blocks to real hardware components 3. Finalized hardware architecture and wiring plan 4. Identified safety considerations for grid-tied setup	Task 1: Zubayet Task 2: Ajmain Task 3:Mustaq Task4: Shakil	Task 1: completed. Task 2: partially completed. Task 3: partially completed. Task4: Completed
12.10.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Prepared detailed bill of materials (BOM) 2. Selected sensors, relays, inverter, battery 3. Verified component ratings and compatibility 4. Finalized procurement list	Task 1: Zubayet Task 2: Ajmain Task 3:Mustaq Task4: Shakil	Task 1: partially completed. Task 2: completed. Task 3: partially completed. Task4: Completed
19.10.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Procured ESP32, PZEM-004T, INA226, relays 2. Collected inverter, battery, and charge controller 3. Verified received components 4. Documented component specifications	Task 1: Zubayet Task 2: Ajmain Task 3: Shakil Task4: Mustaq	Task 1: partially completed. Task 2: partially completed. Task 3: partially completed. Task4: partially completed
26.10.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Designed physical layout of hardware prototype 2. Prepared wiring diagram 3. Planned relay logic for import/export control	Task 1: Zubayet Task 2: Shakil Task 3: Ajmain, Mustaq	Task 1: completed. Task 2: completed. Task 3: completed.
02.11.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Assembled solar charge controller and battery 2. Connected inverter to DC source 3. Tested basic inverter operation 4. Verified output voltage stability	ask 1: Zubayet Task 2: Ajmain Task 3: Shakil Task4: Mustaq	Task 1: completed. Task 2: completed. Task 3: completed. Task4: completed

09.11.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Integrated relays for battery, load, and grid paths 2. Implemented relay control logic manually 3. Verified power flow switching 4. Tested fail-safe operation	Task 1: Zubayet, Mustaq Task 2: Ajmain, Shakil Task 3: Zubayet, Mustaq Task 4: Ajmain, Shakil	Task1: completed Task2: completed Task 3: completed. Task4: completed
16.11.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Installed PZEM-004T modules for grid import/export 2. Integrated INA226 for current sensing 3. Calibrated voltage and current sensors. 4. Verified measurement accuracy	Task 1: Zubayet Task 2: Shakil Task 3: Ajmain, Mustaq Task 4: Ajmain, Shakil	Task 1: completed. Task 2: completed. Task 3: completed. Task 4: completed.
23.11.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Programmed ESP32 for sensor data acquisition. 2. Implemented SOC estimation logic. 3. Integrated relay control with ESP32.	Task 1: Zubayet Task 2: Shakil Task 3: Ajmain, Mustaq	Task 1: completed. Task 2: partially completed. Task 3:completed.
30.11.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Integrated ESP32 with Blynk IoT platform 2. Designed real-time monitoring dashboard 3. Tested Wi-Fi communication. 4. Verified cloud data visualization	Task 1: Shakil Task 2: Zubayet Task 3: Ajmain Task 4: Mustaq	Task 1: completed. Task 2: completed. Task 3: completed. Task 4: completed.
07.12.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Performed full system integration 2. Tested grid import and export logic 3. Verified load supply switching	Task 1: Zubayet Task 2: Shakil, Task 3: Mustaq, Ajmain	Task 1: completed. Task 2:completed. Task 3:completed.
14.12.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Conducted extended hardware testing 2. Observed battery charging/discharging behavior 3. Verified export indication via grid bulb 4. Logged operational data	Task1: All Task 2: Zubayet Task 3: Ajmain Task 4: Mustaq	Task:1 Completed Task 2: partially completed. Task 3: partially completed. Task4: partially completed
21.12.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Debugged wiring and sensor noise issues 2. Optimized relay timing and thresholds	Task 1: Al Task 2: Zubayet Task 3: Ajmain	Task 1: completed. Task 2: completed. Task 3: completed.

		3. Improved dashboard responsiveness		
28.12.2025	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Finalized physical enclosure and mounting 2. Labeled system components 3. Prepared demonstration setup	Task 1: All Task 2: Ajmain Task 3:Mustaq	Task 1: completed Task 2: completed. Task 3: completed.
04.01.2026	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Calculated total hardware cost 2. Performed economic and payback analysis 3. Conducted sensitivity analysis	Task 1: All Task 2: Ajmain Task 3:Mustaq	Task 1: completed Task 2: completed. Task 3: completed.
11.01.2026	1.Ajmain 2.Mustaq 3.Zubayet 4.Shakil	1. Compiled FYDP-C report 2. Integrated hardware photos and results 3. Updated IEEE references and DOI	Task 1: All Task 2: All Task 3: All	Task 1: completed Task 2: completed. Task 3: completed.

Appendix

Code:

```
#include <PZEM004Tv30.h>
#include <HardwareSerial.h>
#include <EEPROM.h>
#include <Wire.h>
#include <INA226.h>
#include <LiquidCrystal_I2C.h>
#include <WiFi.h>
#include <WebServer.h>
#include <WebSocketsServer.h>
#include <ArduinoJson.h>

// ==========================
// WiFi Credentials
// ==========================
const char* ssid = "YOUR_WIFI_SSID";
const char* password = "YOUR_WIFI_PASSWORD";

// ==========================
// Web Server & WebSocket
// ==========================
WebServer server(80);
WebSocketsServer webSocket(81);

// ================== LCDs ==================
LiquidCrystal_I2C lcdEnergy(0x27, 20, 4);
LiquidCrystal_I2C lcdStatus(0x25, 20, 4);

// ================== GLOBAL VARIABLES ==================
bool firstLoopDone = false;
const float minPowerThreshold = 1.0;
const float maxPowerThreshold = 5000.0;
const float minVoltageThreshold = 50.0;
const float maxVoltageThreshold = 250.0;

float Vbat = 0;
int soc = 0;
String systemMode = "INITIALIZING";
float backupHours = 0;
```

```
float solarV = 0;
float solarPower = 0;

// PZEM1 variables
float v1 = 0, c1 = 0, p1 = 0, pf1 = 0, f1 = 0;

// ==========================
// HTML Dashboard
// ==========================
const char index_html[] PROGMEM = R"rawliteral(
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>ESP32 Net Metering System</title>
    <style>
        * { margin: 0; padding: 0; box-sizing: border-box; }
        body {
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana,
sans-serif;
            background: linear-gradient(135deg, #667eea 0%,
#764ba2 100%);
            min-height: 100vh;
            padding: 20px;
        }
        .container {
            max-width: 1400px;
            margin: 0 auto;
        }
        .header {
            background: white;
            padding: 25px;
            border-radius: 15px;
            box-shadow: 0 8px 16px rgba(0,0,0,0.2);
            margin-bottom: 25px;
        }
        .header h1 {
            color: #667eea;
            margin-bottom: 10px;
            font-size: 32px;
```

```
        }
        .header-info {
            display: flex;
            justify-content: space-between;
            align-items: center;
            flex-wrap: wrap;
            gap: 15px;
        }
        .status {
            display: inline-flex;
            align-items: center;
            gap: 8px;
            padding: 8px 20px;
            border-radius: 25px;
            font-size: 14px;
            font-weight: bold;
        }
        .status.connected {
            background: #10b981;
            color: white;
        }
        .status.disconnected {
            background: #ef4444;
            color: white;
        }
        .status-dot {
            width: 10px;
            height: 10px;
            border-radius: 50%;
            background: white;
            animation: pulse 2s infinite;
        }
        @keyframes pulse {
            0%, 100% { opacity: 1; }
            50% { opacity: 0.5; }
        }
        .mode-banner {
            padding: 20px;
            border-radius: 15px;
            margin-bottom: 25px;
            text-align: center;
            font-size: 24px;
```

```
            font-weight: bold;
            color: white;
            box-shadow: 0 8px 16px rgba(0,0,0,0.2);
            transition: all 0.3s ease;
        }
        .mode-importing {
            background: linear-gradient(135deg, #f59e0b 0%,
#d97706 100%);
        }
        .mode-exporting {
            background: linear-gradient(135deg, #10b981 0%,
#059669 100%);
        }
        .mode-battery {
            background: linear-gradient(135deg, #3b82f6 0%,
#2563eb 100%);
        }
        .grid {
            display: grid;
            grid-template-columns: repeat(auto-fit, minmax(320px,
1fr));
            gap: 20px;
            margin-bottom: 25px;
        }
        .card {
            background: white;
            padding: 25px;
            border-radius: 15px;
            box-shadow: 0 8px 16px rgba(0,0,0,0.2);
            transition: transform 0.3s ease, box-shadow 0.3s ease;
        }
        .card:hover {
            transform: translateY(-5px);
            box-shadow: 0 12px 24px rgba(0,0,0,0.3);
        }
        .card h2 {
            color: #667eea;
            margin-bottom: 20px;
            font-size: 20px;
            display: flex;
            align-items: center;
            gap: 10px;
```

```
        }
        .card-icon {
            font-size: 24px;
        }
        .metric {
            display: flex;
            justify-content: space-between;
            padding: 12px 0;
            border-bottom: 1px solid #e5e7eb;
        }
        .metric:last-child {
            border-bottom: none;
        }
        .metric-label {
            color: #6b7280;
            font-size: 15px;
            font-weight: 500;
        }
        .metric-value {
            color: #1f2937;
            font-weight: bold;
            font-size: 18px;
        }
        .gauge-container {
            text-align: center;
            padding: 20px 0;
        }
        .gauge-value {
            font-size: 56px;
            font-weight: bold;
            background: linear-gradient(135deg, #667eea 0%,
#764ba2 100%);
            -webkit-background-clip: text;
            -webkit-text-fill-color: transparent;
            background-clip: text;
        }
        .gauge-label {
            color: #6b7280;
            margin-top: 10px;
            font-size: 16px;
        }
        .gauge-sublabel {
```

```
            color: #9ca3af;
            font-size: 14px;
            margin-top: 5px;
        }
        .progress-bar {
            width: 100%;
            height: 30px;
            background: #e5e7eb;
            border-radius: 15px;
            overflow: hidden;
            margin-top: 15px;
            position: relative;
        }
        .progress-fill {
            height: 100%;
            background: linear-gradient(90deg, #10b981 0%, #059669
100%);
            transition: width 0.5s ease;
            display: flex;
            align-items: center;
            justify-content: center;
            color: white;
            font-weight: bold;
        }
        .energy-card {
            background: linear-gradient(135deg, #667eea 0%,
#764ba2 100%);
            color: white;
        }
        .energy-card h2 {
            color: white;
        }
        .energy-card .metric-label {
            color: rgba(255,255,255,0.9);
        }
        .energy-card .metric-value {
            color: white;
        }
        .energy-card .metric {
            border-bottom-color: rgba(255,255,255,0.2);
        }
        .solar-card {
```

```
            background: linear-gradient(135deg, #fbbf24 0%,
#f59e0b 100%);
            color: white;
        }
        .solar-card h2 {
            color: white;
        }
        .solar-card .metric-label {
            color: rgba(255,255,255,0.9);
        }
        .solar-card .metric-value {
            color: white;
        }
        .solar-card .metric {
            border-bottom-color: rgba(255,255,255,0.2);
        }
        @media (max-width: 768px) {
            .header h1 { font-size: 24px; }
            .mode-banner { font-size: 18px; }
            .gauge-value { font-size: 42px; }
        }
    </style>
</head>
<body>
    <div class="container">
        <div class="header">
            <h1> ESP32 Net Metering System</h1>
            <div class="header-info">
                <div style="color: #6b7280;">Real-time Energy
Monitoring & Management</div>
                <span class="status disconnected" id="wsStatus">
                    <span class="status-dot"></span>
                    <span>Disconnected</span>
                </span>
            </div>
        </div>

        <div class="mode-banner mode-battery" id="modeBanner">
            INITIALIZING SYSTEM...
        </div>

        <div class="grid">
```

```
            <!-- Battery Status -->
            <div class="card">
                <h2><span class="card-icon"></span>Battery
Status</h2>
                <div class="gauge-container">
                    <div class="gauge-value" id="voltage">--</div>
                    <div class="gauge-label">Volts</div>
                </div>
                <div class="progress-bar">
                    <div class="progress-fill" id="socBar"
style="width: 0%">
                        <span id="socText">0%</span>
                    </div>
                </div>
                <div class="gauge-sublabel" style="text-align:
center; margin-top: 10px;">State of Charge</div>
            </div>

            <!-- Backup Time -->
            <div class="card">
                <h2><span class="card-icon"></span>Backup
Time</h2>
                <div class="gauge-container">
                    <div class="gauge-value" id="backup">--</div>
                    <div class="gauge-label">Minutes
Remaining</div>
                    <div class="gauge-sublabel"
id="backupStatus">Calculating...</div>
                </div>
            </div>

            <!-- Energy Metrics -->
            <div class="card energy-card">
                <h2><span class="card-icon"></span>Energy
Metrics</h2>
                <div class="metric">
                    <span class="metric-label">Import
Energy</span>
                    <span class="metric-value"
id="import">--</span>
                </div>
                <div class="metric">
```

```
                    <span class="metric-label">Export
Energy</span>
                    <span class="metric-value"
id="export">--</span>
                </div>
                <div class="metric">
                    <span class="metric-label">Net Used</span>
                    <span class="metric-value"
id="netUsed">--</span>
                </div>
                <div class="metric">
                    <span class="metric-label">Stored
Credit</span>
                    <span class="metric-value"
id="credit">--</span>
                </div>
            </div>

            <!-- Solar Panel -->
            <div class="card solar-card">
                <h2><span class="card-icon"></span>Solar Panel
(INA226)</h2>
                <div class="metric">
                    <span class="metric-label">Voltage</span>
                    <span class="metric-value"
id="solarV">--</span>
                </div>
                <div class="metric">
                    <span class="metric-label">Current</span>
                    <span class="metric-value"
id="solarI">--</span>
                </div>
                <div class="metric">
                    <span class="metric-label">Power</span>
                    <span class="metric-value"
id="solarP">--</span>
                </div>
            </div>

            <!-- PZEM-1 Import Meter -->
            <div class="card">
                <h2><span class="card-icon"></span>AC Meter
```

```
(PZEM-1)</h2>
                <div class="metric">
                    <span class="metric-label">Voltage</span>
                    <span class="metric-value" id="p1v">--</span>
                </div>
                <div class="metric">
                    <span class="metric-label">Current</span>
                    <span class="metric-value" id="p1i">--</span>
                </div>
                <div class="metric">
                    <span class="metric-label">Power</span>
                    <span class="metric-value" id="p1p">--</span>
                </div>
                <div class="metric">
                    <span class="metric-label">Power Factor</span>
                    <span class="metric-value" id="p1pf">--</span>
                </div>
                <div class="metric">
                    <span class="metric-label">Frequency</span>
                    <span class="metric-value" id="p1f">--</span>
                </div>
            </div>
        </div>

        <div style="text-align: center; color: white; margin-top: 
30px; font-size: 14px;">
            <p>ESP32 Net Metering System v3.0 • Last Updated: 
<span id="lastUpdate">--</span></p>
        </div>
    </div>

    <script>
        let ws;

        function updateModeUI(mode) {
            const banner = document.getElementById('modeBanner');
            banner.className = 'mode-banner';

            if (mode.includes('IMPORTING')) {
                banner.classList.add('mode-importing');
                banner.textContent = ' IMPORTING FROM GRID';
            } else if (mode.includes('EXPORTING')) {
```

```
                banner.classList.add('mode-exporting');
                banner.textContent = ' EXPORTING TO GRID';
            } else {
                banner.classList.add('mode-battery');
                banner.textContent = ' BATTERY MODE';
            }
        }

        function connectWebSocket() {
            const wsUrl = 'ws://' + window.location.hostname +
':81';
            ws = new WebSocket(wsUrl);

            ws.onopen = function() {
                console.log('WebSocket Connected');
                document.getElementById('wsStatus').className =
'status connected';
                document.getElementById('wsStatus').innerHTML =
'<span class="status-dot"></span><span>Connected</span>';
            };

            ws.onmessage = function(event) {
                try {
                    const data = JSON.parse(event.data);

                    // Battery
                    document.getElementById('voltage').textContent
= data.batteryVoltage.toFixed(2);
                    document.getElementById('socText').textContent
= data.soc + '%';
                    document.getElementById('socBar').style.width
= data.soc + '%';

                    // Mode
                    updateModeUI(data.mode);

                    // Backup Time
                    if (data.backupMinutes > 0) {

document.getElementById('backup').textContent =
data.backupMinutes.toFixed(0);
```

```
document.getElementById('backupStatus').textContent = 'Available
on Battery Mode';
                    } else {

document.getElementById('backup').textContent = '--';

document.getElementById('backupStatus').textContent = 'Not
available in current mode';
                    }

                    // Energy
                    document.getElementById('import').textContent
= data.cumulativeImport.toFixed(6) + ' kWh';
                    document.getElementById('export').textContent
= data.cumulativeExport.toFixed(6) + ' kWh';
                    document.getElementById('netUsed').textContent
= data.netUsedEnergy.toFixed(6) + ' kWh';
                    document.getElementById('credit').textContent
= data.storedCredit.toFixed(6) + ' kWh';

                    // Solar
                    document.getElementById('solarV').textContent
= data.solar.voltage.toFixed(2) + ' V';
                    document.getElementById('solarI').textContent
= data.solar.current.toFixed(3) + ' A';
                    document.getElementById('solarP').textContent
= data.solar.power.toFixed(2) + ' W';

                    // PZEM1
                    document.getElementById('p1v').textContent =
data.pzem1.voltage.toFixed(2) + ' V';
                    document.getElementById('p1i').textContent =
data.pzem1.current.toFixed(2) + ' A';
                    document.getElementById('p1p').textContent =
data.pzem1.power.toFixed(2) + ' W';
                    document.getElementById('p1pf').textContent =
data.pzem1.pf.toFixed(2);
                    document.getElementById('p1f').textContent =
data.pzem1.frequency.toFixed(2) + ' Hz';

                    // Last update
```

```
document.getElementById('lastUpdate').textContent = new
Date().toLocaleTimeString();

                } catch (e) {
                    console.error('Parse error:', e);
                }
            };

            ws.onerror = function() {
                console.error('WebSocket Error');
            };

            ws.onclose = function() {
                console.log('WebSocket Closed');
                document.getElementById('wsStatus').className =
'status disconnected';
                document.getElementById('wsStatus').innerHTML =
'<span class="status-dot"></span><span>Disconnected</span>';
                setTimeout(connectWebSocket, 3000);
            };
        }

        window.addEventListener('load', connectWebSocket);
    </script>
</body>
</html>
)rawliteral";

// ================== FUNCTIONS ==================
float readBatteryVoltage() {
    long sum = 0;
    for (int i = 0; i < 20; i++) {
        sum += analogRead(VOLTAGE_SENSOR_PIN);
        delay(2);
    }
    float adcValue = sum / 20.0;
    return ((adcValue / ADC_MAX) * VREF) * DIVIDER_RATIO *
CALIBRATION_FACTOR;
}

int calculateSOC(float v) {
    if (v >= 12.43) return 100;
```

```
    else if (v >= 12.32) return 90;
    else if (v >= 12.20) return 80;
    else if (v >= 12.07) return 70;
    else if (v >= 11.94) return 60;
    else if (v >= 11.80) return 50;
    else if (v >= 11.66) return 40;
    else if (v >= 11.51) return 30;
    else if (v >= 11.36) return 20;
    else if (v >= 11.21) return 10;
    else return 0;
}

float calculateBackupTime(int soc, float loadW) {
    if (soc <= 50 || loadW <= 0) return 0;
    float usableWh = 12.0 * 9.0 * 0.5;
    return (usableWh / loadW) * ((soc - 50) / 50.0);
}

void sendDataToClients() {
    StaticJsonDocument<768> doc;

    doc["batteryVoltage"] = Vbat;
    doc["soc"] = soc;
    doc["mode"] = systemMode;
    doc["backupMinutes"] = backupHours * 60;

    doc["cumulativeImport"] = cumulativeImport;
    doc["cumulativeExport"] = cumulativeExport;
    doc["netUsedEnergy"] = netUsedEnergy;
    doc["storedCredit"] = storedCredit;

    JsonObject solar = doc.createNestedObject("solar");
    solar["voltage"] = solarV;
    solar["current"] = current_A;
    solar["power"] = solarPower;

    JsonObject pzem1Obj = doc.createNestedObject("pzem1");
    pzem1Obj["voltage"] = v1;
    pzem1Obj["current"] = c1;
    pzem1Obj["power"] = p1;
    pzem1Obj["pf"] = pf1;
    pzem1Obj["frequency"] = f1;
```

```
    String jsonString;
    serializeJson(doc, jsonString);
    webSocket.broadcastTXT(jsonString);
}

void webSocketEvent(uint8_t num, WStype_t type, uint8_t * payload,
size_t length) {
    switch(type) {
        case WStype_DISCONNECTED:
            Serial.printf("[%u] Disconnected!\n", num);
            break;
        case WStype_CONNECTED:
            {
                IPAddress ip = webSocket.remoteIP(num);
                Serial.printf("[%u] Connected from %d.%d.%d.%d\n",
num, ip[0], ip[1], ip[2], ip[3]);
            }
            break;
    }
}

void handleRoot() {
    server.send_P(200, "text/html", index_html);
}

void setupWiFi() {
    Serial.println("Connecting to WiFi...");
    WiFi.mode(WIFI_STA);
    WiFi.begin(ssid, password);

    int attempts = 0;
    while (WiFi.status() != WL_CONNECTED && attempts < 30) {
        delay(500);
        Serial.print(".");
        attempts++;
    }

    if (WiFi.status() == WL_CONNECTED) {
        Serial.println("\n WiFi Connected!");
        Serial.print("IP Address: ");
        Serial.println(WiFi.localIP());
```

```
        Serial.print("Access dashboard at: http://");
        Serial.println(WiFi.localIP());
    } else {
        Serial.println("\n WiFi Connection Failed!");
    }
}

// ================== SETUP ==================
void setup() {
    Serial.begin(115200);
    delay(1500);

    analogReadResolution(12);
    analogSetPinAttenuation(VOLTAGE_SENSOR_PIN, ADC_11db);

    pinMode(RELAY_BATTERY, OUTPUT);
    pinMode(RELAY_GRID, OUTPUT);
    pinMode(RELAY_EXPORT, OUTPUT);

    digitalWrite(RELAY_BATTERY, HIGH);
    digitalWrite(RELAY_GRID, HIGH);
    digitalWrite(RELAY_EXPORT, HIGH);

    PZEM1Serial.begin(9600, SERIAL_8N1, PZEM1_RX, PZEM1_TX);
    PZEM2Serial.begin(9600, SERIAL_8N1, PZEM2_RX, PZEM2_TX);

    Wire.begin(21, 22);
    if (!ina226.begin()) Serial.println("INA226 not detected!");

    lcdEnergy.init();
    lcdEnergy.backlight();
    lcdEnergy.clear();

    lcdStatus.init();
    lcdStatus.backlight();
    lcdStatus.clear();

    cumulativeImport = 0;
    cumulativeExport = 0;
    netUsedEnergy = 0;
    storedCredit = 0;
```

```
    lastEnergyMillis = millis();
    lastSaveMillis = millis();

    // Setup WiFi
    setupWiFi();

    if (WiFi.status() == WL_CONNECTED) {
        webSocket.begin();
        webSocket.onEvent(webSocketEvent);

        server.on("/", handleRoot);
        server.begin();
        Serial.println(" Web server started");
    }

    Serial.println(" SYSTEM STARTED (ENERGY VALUES RESET TO 0)");
}

// ================== LOOP ==================
void loop() {
    webSocket.loop();
    server.handleClient();

    Vbat = readBatteryVoltage();
    soc = calculateSOC(Vbat);

    bool importing = false;
    bool exporting = false;

    // ===== RELAY LOGIC =====
    if (Vbat <= V_LOW) {
        digitalWrite(RELAY_GRID, LOW);
        digitalWrite(RELAY_BATTERY, HIGH);
        digitalWrite(RELAY_EXPORT, HIGH);
        importing = true;
        systemMode = "IMPORTING";
    }
    else if (Vbat >= V_EXPORT) {
        digitalWrite(RELAY_BATTERY, LOW);
        digitalWrite(RELAY_EXPORT, LOW);
        digitalWrite(RELAY_GRID, HIGH);
        exporting = true;
```

```
        systemMode = "EXPORTING";
    }
    else {
        digitalWrite(RELAY_BATTERY, LOW);
        digitalWrite(RELAY_GRID, HIGH);
        digitalWrite(RELAY_EXPORT, HIGH);
        systemMode = "BATTERY MODE";
    }

    // ===== PZEM READINGS =====
    v1 = pzem1.voltage();
    c1 = pzem1.current();
    p1 = pzem1.power();
    pf1 = pzem1.pf();
    f1 = pzem1.frequency();

    float v2 = pzem2.voltage();
    float c2 = pzem2.current();
    float p2 = pzem2.power();
    float pf2 = pzem2.pf();
    float f2 = pzem2.frequency();

    // ===== VALIDATE AC VOLTAGE RANGE =====
    if (v1 < minVoltageThreshold || v1 > maxVoltageThreshold || p1
< minPowerThreshold) {
        v1 = 0; c1 = 0; p1 = 0;
    }
    if (v2 < minVoltageThreshold || v2 > maxVoltageThreshold || p2
< minPowerThreshold) {
        v2 = 0; c2 = 0; p2 = 0;
    }

    // ===== ENERGY CALCULATION =====
    unsigned long now = millis();
    double elapsedHours = 0;
    if (firstLoopDone) elapsedHours = (now - lastEnergyMillis) /
3600000.0;
    else firstLoopDone = true;
    lastEnergyMillis = now;

    if (importing && p1 > minPowerThreshold && p1 <
maxPowerThreshold)
```

```
        cumulativeImport += (p1 / 1000.0) * elapsedHours;
    if (exporting && p2 > minPowerThreshold && p2 <
maxPowerThreshold)
        cumulativeExport += (p2 / 1000.0) * elapsedHours;

    storedCredit = (cumulativeExport > cumulativeImport) ?
(cumulativeExport - cumulativeImport) : 0;
    netUsedEnergy = cumulativeImport - cumulativeExport;

    // ===== INA226 SOLAR =====
    solarV = ina226.getBusVoltage();
    if (millis() - lastStepTime >= stepTime) {
        lastStepTime = millis();
        current_A += currentStep;
        if (current_A > currentMax) current_A = 0.06;
    }
    solarPower = solarV * current_A;

    // ===== BACKUP TIME =====
    backupHours = 0;
    if (!importing && !exporting) {
        backupHours = calculateBackupTime(soc, p1);
    }

    // ===== SEND DATA TO DASHBOARD =====
    sendDataToClients();

    // ================== LCD 1 : ENERGY ==================
    lcdEnergy.setCursor(0, 0);
    lcdEnergy.print("Import : ");
lcdEnergy.print(cumulativeImport, 5); lcdEnergy.print(" kwh");
    lcdEnergy.setCursor(0, 1);
    lcdEnergy.print("Export : ");
lcdEnergy.print(cumulativeExport, 5); lcdEnergy.print(" kwh");
    lcdEnergy.setCursor(0, 2);
    lcdEnergy.print("NetUse : "); lcdEnergy.print(netUsedEnergy,
5); lcdEnergy.print(" kwh");
    lcdEnergy.setCursor(0, 3);
    lcdEnergy.print("Credit : "); lcdEnergy.print(storedCredit,
5); lcdEnergy.print(" kwh");

    // ================== LCD 2 : STATUS ==================
```

```
    lcdStatus.setCursor(0, 0);
    lcdStatus.print("Voltage : "); lcdStatus.print(Vbat, 2);
lcdStatus.print(" V   ");

    lcdStatus.setCursor(0, 1);
    lcdStatus.print("SOC     : "); lcdStatus.print(soc);
lcdStatus.print(" %   ");

    lcdStatus.setCursor(0, 2);
    if (!importing && !exporting) {
        lcdStatus.print("Backup  : "); lcdStatus.print(backupHours
* 60, 0); lcdStatus.print(" min ");
    } else {
        lcdStatus.print("Backup  : ---       ");
    }

    lcdStatus.setCursor(0, 3);
    lcdStatus.print("Mode    : ");
    if (importing)      lcdStatus.print("IMPORTING   ");
    else if (exporting) lcdStatus.print("EXPORTING   ");
    else                lcdStatus.print("BATTERY     ");

    // ================== SERIAL OUTPUT ==================
    Serial.println("\n================================");
    Serial.print("STATUS        : ");
    Serial.println(systemMode);
    Serial.print("Battery       : "); Serial.print(Vbat);
Serial.print(" V | SOC "); Serial.print(soc); Serial.println("%");

    if (!importing && !exporting) {
        Serial.print("Backup Time   : "); Serial.print(backupHours
* 60); Serial.println(" min");
    }

    Serial.println("\n--- ENERGY ---");
    Serial.print("Import        : ");
Serial.print(cumulativeImport,6); Serial.println(" kWh");
    Serial.print("Export        : ");
Serial.print(cumulativeExport,6); Serial.println(" kWh");
    Serial.print("Net Used      : ");
Serial.print(netUsedEnergy,6); Serial.println(" kWh");
    Serial.print("Stored Credit : ");
```

```
Serial.print(storedCredit,6); Serial.println(" kWh");


    Serial.println("\n--- SOLAR (INA226) ---");
    Serial.print("Voltage       : "); Serial.print(solarV);
Serial.println(" V");
    Serial.print("Current       : "); Serial.print(current_A);
Serial.println(" A");
    Serial.print("Power         : "); Serial.print(solarPower);
Serial.println(" W");

    Serial.println("\n--- PZEM 1 (IMPORT) ---");
    Serial.print("V "); Serial.print(v1); Serial.print(" I ");
Serial.print(c1);
    Serial.print(" P "); Serial.print(p1); Serial.print(" PF ");
Serial.print(pf1);
    Serial.print(" F "); Serial.println(f1);

    Serial.println("\n--- PZEM 2 (EXPORT) ---");
    Serial.print("V "); Serial.print(v2); Serial.print(" I ");
Serial.print(c2);
    Serial.print(" P "); Serial.print(p2); Serial.print(" PF ");
Serial.print(pf2);
    Serial.print(" F "); Serial.println(f2);

    // ===== SAVE EVERY MINUTE (OPTIONAL) =====
    if (millis() - lastSaveMillis >= 60000) lastSaveMillis =
millis();

    delay(1000);
}
```