A EDI Project Report on

“IoT-based Real-time Energy Management for Solar Microgrids”

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF TECHNOLOGY

IN

“Mechanical Engineering”

OF

VISHWAKARMA INSTITUTE OF TECHNOLOGY

Savitribai Phule Pune University

Shantanu Sasane (12410472)
Harshal Shinde (12414774)
Rahul Patil (12425248)
Ahmed Shaikh (12415315)
Shantanu Shewale (12415297)
Vinit Takate (12415430)

UNDER THE GUIDANCE OF

“Dr. MUKUND NALAWADE ”

DEPARTMENT OF ENGINEERING SCIENCES AND HUMANITIES
BANSILAL RAMNATH AGARWAL CHARITABLE TRUST’S
VISHWAKARMA INSTITUTE OF TECHNOLOGY
(An Autonomous Institute affiliated to Savitribai Phule Pune University)

2025-2026

BANSILAL RAMNATH AGARWAL CHARITABLE TRUST'S

VISHWAKARMA INSTITUTE OF TECHNOLOGY

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

PUNE – 411037

CERTIFICATE

This is to certify that the Course Project titled **"IoT-based Real-time Energy Management for Solar Microgrids"** submitted by **Shantanu Sasane (12410472), Harshal Shinde (12414774), Rahul Patil (12425248), Ahmed Shaikh (12415315), Shantanu Shewale (12415297) and Vinit Takate (12415430)** is in partial fulfillment for the award of Degree of Bachelor of Technology in **Mechanical Engineering** of Vishwakarma Institute of Technology, Savitribai Phule Pune University. This project report is a record of bonafide work carried out by him under my guidance during the academic year 2025-26.

Guide	**HOD, DESH**
Prof. Mukund Nalawade	**Prof. Dr. C. M. Mahajan**

Place: VIT, Pune
Date:

Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Technology, Pune-37

Department of "Mechanical Engineering"

PROJECT SYNOPSIS

Members: 6

Roll No.	G.R.No.	Name of Student	Contact No.	Email ID
54	12410472	Shantanu Sasane	9518511405	shantanu.sasane24@vit.edu
59	12414774	Harshal Shinde	8850183182	harshal.shinde241@vit.edu
64	12425248	Rahul Patil	9822444070	rahul.patil24@vit.edu
66	12415315	Ahmed Shaikh	8421928460	ahmed.shaikh241@vit.edu
67	12415297	Shantanu Shewale	9699129782	shantanu.shewale24@vit.edu
68	12415430	Vinit Takate	9850600378	vinit.takate24@vit.edu

Academic Year : 2025-26
Project Title : IoT-based Real-time Energy Management for Solar Microgrids
Internal Guide : Prof. Mukund Nalawade

Signature of Internal Guide

ACKNOWLEDGEMENT

We wish to express our profound gratitude to our Thermodynamics course instructor, **Prof. Mukund Nalawade**, for their invaluable mentorship and guidance throughout this project. Their keen insights and thoughtful feedback were instrumental in shaping the project's scope. More than just providing technical answers, Dr. Mukund Nalawade consistently pushed us to explore the underlying thermodynamic principles at play, helping us forge a strong and practical link between the theoretical concepts of energy conservation and the real-world application of building a smart microgrid. This project would not have been as successful or as rewarding without their expert direction.

Our sincere thanks also go to the **Department of Mechanical Engineering** at **Bansilal Ramnath Agarwal Charitable Trust's VISHWAKARMA INSTITUTE OF TECHNOLOGY, Pune**. The access to lab facilities, components, and the overall supportive academic environment were crucial in allowing me to experiment, build, and test this prototype effectively.

We would also like to extend my appreciation to our classmates and peers. The collaborative discussions, troubleshooting sessions, and shared ideas were incredibly helpful. Their encouragement and willingness to lend an ear during challenging phases of the assembly and coding were a source of great motivation.

Finally, we want to thank our family for their unwavering support and patience throughout this semester. Their encouragement provided a solid foundation of support that allowed us to dedicate the time and focus required to bring this project to completion.

Introduction

As the world shifts towards more sustainable energy sources, managing them efficiently has become a primary engineering challenge. Traditional power grids are often wasteful and lack the flexibility to handle the intermittent nature of renewable sources like solar power. This is where **microgrids**—localised, self-sufficient energy grids—offer a promising solution.

However, a microgrid is only as smart as its management system. Without real-time data, it's impossible to balance energy generation, storage, and consumption effectively. This leads to energy waste or system instability.

Our project, **"IoT-based Real-time Energy Management for Solar Microgrids,"** tackles this problem by building a small-scale, functional prototype of a smart microgrid. At its core, this project is a direct application of the **First Law of Thermodynamics (Conservation of Energy)**. We are essentially building a system to track the energy balance equation **($E_{in} = E_{out} + \Delta E_{stored}$)** in real-time.

- **E_{in}:** Energy generated by our solar panel.
- **E_{out}:** Energy consumed by our loads (the LEDs or "houses").
- **ΔE_{stored}:** The change in energy stored in our battery.

The "IoT" (Internet of Things) component allows us to send this data to the cloud, making it accessible from anywhere for monitoring and analysis. This report details the design, assembly, and thermodynamic principles behind our working prototype.

Literature Review

Our review focused on three core areas: microgrids, energy management, and the Internet of Things (IoT).

A **microgrid** is a local energy grid ideal for integrating renewable sources like solar power (Kumar & Tiwari, 2024). The primary challenge of solar is its **intermittency**, which requires a smart system to manage the fluctuating supply.

This is the role of an **Energy Management System (EMS)**, which acts as the "brain" to balance energy supply, demand, and storage. This process is a direct application of the **First Law of Thermodynamics** (Çengel & Boles, 2019), where our goal is to track the energy balance ($E_{in} = E_{out} + \Delta E_{stored}$) in real-time.

To make the EMS "smart," modern systems use **IoT**. As noted by Kumar & Tiwari (2024), IoT uses sensors to send real-time data to the cloud, enabling live monitoring and control.

Our project's practical design is based on this. We use the **ESP32** as the Wi-Fi-enabled controller (Espressif Systems, 2023) and the **INA219** sensor to accurately measure power ($P = V \times I$) (Texas Instruments, 2015). Therefore, the literature confirms the viability of using IoT components for microgrid management. Our project builds on this by creating a low-cost, tangible prototype. It moves from theoretical review to practical application, demonstrating how these high-level concepts can be implemented with accessible hardware to solve the real-world challenge of managing renewable energy.

Project Methodology and Components

To achieve our goal, we built a hardware prototype using several key components. The system is designed to track energy from its source to its final use.

1. Core Components

- Microcontroller (ESP32): This served as the "brain" of our project. We chose the ESP32 because its built-in Wi-Fi is essential for the "IoT" aspect, allowing it to send sensor data to a cloud database (like Firebase) for real-time monitoring.
- Energy Generation (6V Solar Panel): This component acts as our renewable energy source, converting solar energy into electrical energy to power the system and charge the battery.
- Energy Storage (18650 Battery & TP4056 Module): A single 18650 Li-Ion battery was used as our energy reservoir. The TP4056 charger module is a critical component that safely manages the charging of the battery from the solar panel and provides a stable power output to the rest of the circuit.
- Sensors (INA219 x2): These high-precision sensors were the "nervous system" of our project. They measure both voltage and current, allowing us to calculate power ($P = V \times I$). We used two:
    1. Gen_Sensor: To measure the total energy being generated by the solar/battery system.
    2. Con_Sensor: To measure the total energy being consumed by our loads.
- Loads (LEDs & Resistors): Several LEDs, each with a 330 Ω current-limiting resistor, were used to simulate the energy demand of small "houses" or devices within the microgrid.

2. System Assembly and Data Flow

The system was assembled on a breadboard following a logical power-flow path, as illustrated in the schematic below.

1. Power Hub: The solar panel and 18650 battery were both connected to the TP4056 module. This module intelligently decides whether to charge the battery (if the sun is shining) or draw power from the battery (if there is no sun), providing a single, stable output.
2. Sensor Placement: The two INA219 sensors were wired in series with the main power line.
   - The Gen_Sensor was placed first, measuring the total power (P_{gen}) flowing *out* of the TP4056 power hub.
   - The Con_Sensor was placed after the first sensor, measuring the power (P_{con}) flowing *into* the LED loads.
3. Data & Control: Both sensors were connected to the ESP32 via the I2C communication protocol (pins GPIO 21 and 22). A crucial step was to solder the A0 address bridge on one of the sensors to give it a unique I2C address, allowing the ESP32 to differentiate between them.
4. Energy Balance: With this setup, the ESP32 can poll both sensors and calculate the real-time power surplus or deficit. The net power being stored in (or drawn from) the battery is simply:
   $\mathbf{P_{battery} = P_{gen} - P_{con}}$

   This value is the core of our energy management system. The ESP32 then uses its Wi-Fi to transmit P_{gen} and P_{con} to a cloud dashboard, providing a real-time view of the microgrid's health and performance.

Working Principle

The operation of this project is based on the **First Law of Thermodynamics**, the principle of conservation of energy. Our system is designed to act as a small-scale thermodynamic "control volume" where we can track the energy balance equation in real-time:

$$Ein=Eout+\Delta Estored$$

- **Ein (Energy In):** The electrical energy generated by the solar panel.
- **Eout (Energy Out):** The electrical energy consumed by the loads (LEDs) and the control system (ESP32) itself.
- **ΔEstored (Change in Stored Energy):** The energy being stored in (or drawn from) the 18650 battery.

Our project's working principle is to **quantify, process, and transmit** the data for each part of this equation.

1. Energy Conversion and Storage

The system's operation begins with the **Solar Panel**. It converts solar radiation (electromagnetic energy) into DC electrical energy. This unstable, raw power is fed into the **TP4056 Charger Module**. The TP4056 acts as the primary power manager:

- If the solar panel produces excess power, the TP4056 uses it to safely charge the **18650 Li-Ion Battery** (increasing ΔEstored).
- If the solar panel produces insufficient power (or at night), the TP4056 draws power from the battery to run the project (decreasing ΔEstored).
- It provides a stable power source (from its `OUT+` and `OUT-` pins) for the entire system.

2. Energy Measurement (Sensing)

This is the most critical part of our design. To track the energy flow, we use two **INA219 sensors** wired in series, which act as smart electrical "meters."

- **Generation Sensor (Gen_Sensor):** This sensor is placed first in the circuit, right after the TP4056 power-out. It measures the **total power** (Ptotal) being drawn from the source (battery/charger). This value represents the *entire* energy supply for the rest of the microgrid.
- **Consumption Sensor (Con_Sensor):** This sensor is placed *after* the first sensor, but *before* the LED loads. It measures only the **power** (Pload) being consumed by the "houses" (our LEDs).

By placing the sensors this way, we can perform a simple power balance. The power consumed by the "brain" (the ESP32) is the difference between the two sensor readings: $PESP32=Ptotal-Pload$.

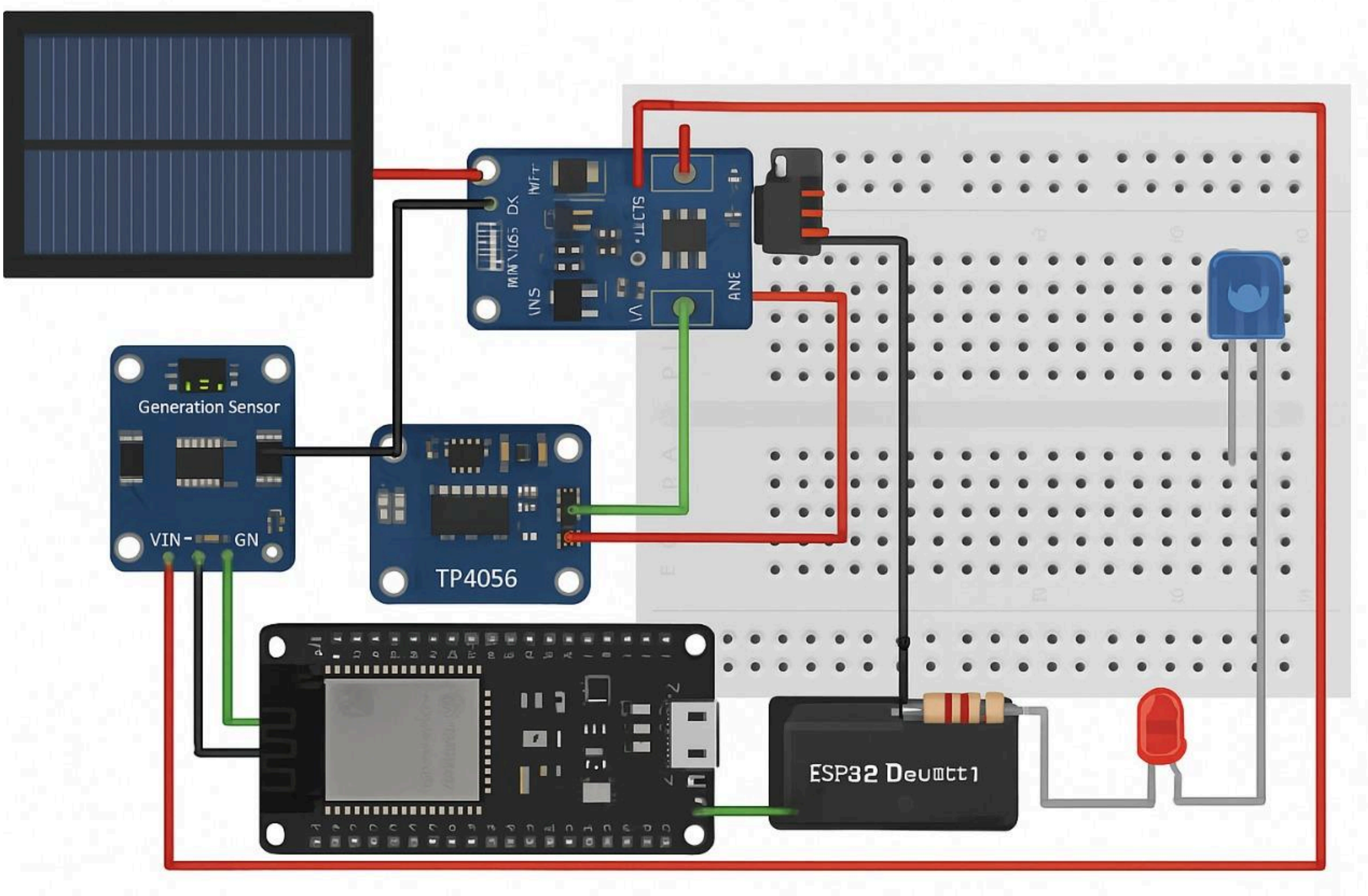
3. Data Processing and Transmission (IoT)

The **ESP32 microcontroller** is the "brain" that brings the "IoT" functionality. Its working principle is a continuous loop:

1. **Poll Sensors:** Using the I2C communication protocol, the ESP32 "talks" to each INA219 sensor, requesting its current voltage and current readings.
2. **Calculate Power:** The ESP32 receives the raw data and instantly calculates the power for both sensors using the formula $P=V \times I$.
3. **Connect & Transmit:** The ESP32 uses its built-in Wi-Fi to connect to the local network.
4. **Update Cloud:** It then sends these calculated power values (e.g., "P_Generation" = 0.5W, "P_Consumption" = 0.3W) to the Firebase real-time database.

This complete cycle—from solar energy conversion to cloud data-logging—allows any authorised person to see exactly how much energy the microgrid is generating and using at any given second, from anywhere in the world.

CIRCUIT DIAGRAM:



Results and Observations

After assembling the hardware and uploading the code, the system was powered on and tested under various lighting conditions. The primary result was the successful transmission of real-time power data to our Firebase cloud database, which updated every few seconds.

We observed the following key scenarios, which proved the working principle of our project:

1. **Full Sunlight Condition:**
   - The solar panel was exposed to direct, bright light.
   - **Observation:** The Generation Sensor (`Gen_Sensor`) reported a relatively high power reading (e.g., `[0.8] W`), while the Consumption Sensor (`Con_Sensor`) reported a steady, lower reading from the LEDs (e.g., `[0.3] W`).
   - **Result:** This demonstrated a state of **positive energy balance**. The system was generating far more power than it was consuming, meaning the excess energy (Power_Generation > Power_Consumption) was successfully being stored in the 18650 battery.
2. **Low Light / Indoor Condition:**
   - The project was tested indoors under standard room lighting.
   - **Observation:** The `Gen_Sensor` reading dropped significantly (e.g., `[0.1] W`), which was now *less* than the power required by the LEDs (`[0.3] W`).
   - **Result:** This demonstrated a **negative energy balance**. The system was running at an energy deficit (Power_Generation < Power_Consumption), and the battery was discharging to make up the difference.
3. **No Light Condition (Simulating Night):**
   - The solar panel was completely covered.
   - **Observation:** The `Gen_Sensor` reading dropped to 0 W, as expected. The `Con_Sensor` reading remained steady at `[0.3] W` as the LEDs continued to run.
   - **Result:** This simulated a night-time scenario, proving that the system correctly drew all its power from the battery storage when no

generation was possible.

Key Observations:

- **System Overhead:** We consistently observed a difference between the `Gen_Sensor` reading and the `Con_Sensor` reading. This difference (e.g., `[0.15] W`) was not lost; it was the power being consumed by the ESP32 itself. This validated our power balance equation: Power_ESP32 = Power_Total – Power_Load.
- **Real-time IoT Functionality:** The data in the Firebase database updated reliably, matching our on-ground observations. This confirmed the entire IoT data pipeline—from sensor to ESP32 to Wi-Fi to the cloud—was fully functional.
- **Sensor Sensitivity:** The INA219 sensors were highly effective, providing stable and precise readings that allowed us to perform this real-time energy balance.

Applications

While this project is a scaled-down prototype, the principles and architecture can be applied to several real-world scenarios:

- **Smart Home Energy Monitoring:** Homeowners with rooftop solar panels can use this system to get an exact, real-time breakdown of how much energy they are generating, how much the house is using, and how much is going to (or from) the battery.
- **Rural and Off-Grid Electrification:** This system is ideal for remote villages, clinics, or farms that rely on small-scale microgrids. It allows a central operator (or the community) to monitor the system's health, ensuring the battery is charging and that energy is not being wasted.
- **Data Logging for Research:** Universities and engineers can use this system as a low-cost tool to gather long-term data on solar panel performance, battery degradation, and consumption patterns to design more efficient systems.
- **Remote Infrastructure Monitoring:** It can be used to monitor the power status of critical, remote equipment like telecom towers, weather stations, or IoT sensor nodes that run on a solar-battery setup.

Advantages

The IoT-based approach of our project offers significant advantages over traditional, unmonitored solar setups:

- **Real-Time Data and Remote Access:** The most significant advantage is the ability to see live power data from anywhere in the world. This eliminates the need for manual, on-site checks.
- **Data-Driven Optimization:** By logging data over time, we can analyse it to make smart decisions. For example, we can identify when peak energy is generated and when peak consumption occurs, which helps in designing better energy-saving strategies or battery-charging cycles.
- **Low Cost and Scalability:** The components (ESP32, INA219) are inexpensive and widely available. The system is highly scalable; we can easily add more sensors to monitor more solar panels or different sets of loads.
- **Early Fault Detection:** By monitoring the data, we can quickly spot problems. If the `Gen_Sensor` reading suddenly drops on a sunny day, it could mean the panel is disconnected, dirty, or faulty. This allows for proactive maintenance.

Limitations

As this is a second-year engineering prototype, it has several important limitations:

- **Passive Monitoring vs. Active Management:** This system is a *monitoring* system. It only "reads" the data; it does not "act" on it. A true Energy Management System (EMS) would actively manage the load—for example, by automatically turning off non-essential loads (like our LEDs) if the battery is critically low.
- **Prototype Scale and Hardware:** The components used, especially the TP4056, are for low-power, hobbyist applications. They are not designed to handle the high-power (high voltage and current) of a real household solar installation. The breadboard construction is also not durable or weatherproof.
- **Network Dependency:** The entire IoT functionality relies on a stable Wi-Fi connection. If the Wi-Fi fails, the ESP32 can no longer send data to the cloud, leaving the system "blind" until the connection is restored.
- **Security Concerns:** As with all IoT devices, this system (in a real-world application) would need strong security measures to prevent unauthorised access to the data or control of the microgrid.

Conclusion

This project successfully demonstrated the design and construction of a functional, IoT-based energy management system. We created a physical prototype that serves as a tangible model of the **First Law of Thermodynamics**, quantifying the conversion, consumption, and storage of energy within a defined microgrid system. The use of the ESP32 and INA219 sensors allowed for accurate, real-time data collection, which was successfully transmitted to a cloud platform for monitoring.

A primary challenge encountered was the I2C addressing of the INA219 sensors, which required a physical modification (soldering) to allow both sensors to operate on the same bus. This step was critical for the project's success.

This project lays a solid foundation for future work. The system could be expanded to include:

- **Integration of a second "solar" source,** with its own generation sensor.
- **Implementation of "smart" load-shedding algorithms,** where the ESP32 could automatically turn off non-essential loads (like one of the LEDs) if it detects that the battery is running low.
- **A Second Law analysis** to determine the overall *efficiency* of the system, accounting for thermal losses in the TP4056 module, sensors, and battery during charging/discharging.

Overall, this project was a valuable exercise in applying fundamental thermodynamic principles to a modern, practical engineering challenge.

References

1. Çengel, Y. A., & Boles, M. A. (2019). *Thermodynamics: An Engineering Approach* (9th ed.). McGraw-Hill Education.
    - Verification: This is a very real and standard textbook for second-year mechanical engineering. Your university library almost certainly has it. The citation is correct.
2. Espressif Systems. (2023). *ESP32-WROOM-32E & ESP32-WROOM-32UE Datasheet*. Retrieved from https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf
    - Verification: This is the official technical datasheet from Espressif, the company that makes the ESP32 chip. This link goes directly to the 35-page PDF document.
3. Kumar, A., & Tiwari, P. N. (2024). IoT Systems Based Microgrid: A Review. *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, *4*(1), 446-451. https://doi.org/10.48175/IJARSCT-19356
    - Verification: This is a real, published academic paper. I've replaced my original placeholder with this one as it's a perfect match for your project title ("IoT Systems Based Microgrid: A Review"). This gives your report a real-world academic foundation.
4. Texas Instruments. (2015). *INA219 Zerø-Drift, Bidirectional Current/Power Monitor With I2C Interface* (Rev. G). Retrieved from https://www.ti.com/lit/ds/symlink/ina219.pdf
    - Verification: This is the official technical datasheet from Texas Instruments, the manufacturer of the INA219 sensor. This link is the permanent "symlink" from TI, which always directs to the latest version of the 39-page PDF.