



International Journal of Research Publication and Reviews

Journal Homepage: www.ijrpr.com ISSN 2582-7421

IoT-Based Monitoring and Control of Smart Inverter System

Dr. S. Arun¹, Gowrishankar A², Sivamani A³, Ranjith AR⁴

¹Associate Professor and Project Guide, Department of Electrical and Electronics Engineering, P. A. College of Engineering and Technology (Autonomous), Pollachi – 642 002, Tamil Nadu, India

^{2,3,4}UG Students, Department of Electrical and Electronics Engineering, P. A. College of Engineering and Technology (Autonomous), Pollachi – 642 002, Tamil Nadu, India

DOI : <https://doi.org/10.55248/gengpi.07.1026.3232>

ABSTRACT

Most small inverters tell the user very little about what is going on inside them. A falling battery voltage, a heavy current draw or an overheating transformer usually goes unnoticed until the inverter trips or fails. This paper presents a smart inverter that measures, displays, protects and reports its own condition using an ESP32 microcontroller. Battery voltage, battery current, AC output voltage and inverter temperature are read from a voltage-divider module, an ACS712 current sensor, a ZMPT101B module and a DHT11 sensor. The readings appear on a 16×2 I²C LCD and are sent over Wi-Fi to a Blynk dashboard. A two-channel relay switches the inverter and a 12 V cooling fan: the fan starts when the temperature goes above 35 °C, and the inverter is isolated when the battery, current or AC reading leaves its safe range. The inverter can also be switched ON or OFF from the dashboard. A prototype with a 12 V, 4 Ah battery and a 300 W inverter was tested with a 40 W bulb, and its readings were logged on the dashboard. The unit gives local display, remote monitoring and automatic protection in one low-cost arrangement.

Keywords: IoT, smart inverter, ESP32, Blynk, relay protection, temperature-controlled cooling, remote monitoring

1. Introduction

Inverters keep homes, shops and small offices running during power cuts by turning the DC stored in a battery into AC for ordinary loads. They are used every day, yet the user usually sees only a few indicator lights or a buzzer. Battery voltage, load current and the temperature of the unit are not shown, and the owner has to stand next to the inverter to judge its condition [3], [5].

This matters because the battery and the inverter are stressed by exactly the quantities that stay hidden. A deeply discharged battery, a load that draws more current than expected or poor ventilation can shorten the life of the equipment, and by the time a basic protection circuit reacts the user has had no warning. Remote access is rarer still. In most basic inverters there is no way to see the readings from another room, or to switch the inverter off without walking up to it.

The Internet of Things (IoT) offers a low-cost answer [10], [11]. Energy-monitoring systems built around the ESP32 already send electrical readings to cloud dashboards [1], [2], and a few recent works extend the idea to inverter batteries [3] and to inverter control [5]. This work follows the same direction but keeps everything in one small unit: four measured quantities, a local LCD, a Blynk dashboard, a temperature-controlled fan and a relay that can isolate the inverter, either automatically or on a command from the phone.

The main objectives of the proposed work are:

- To measure battery voltage, battery current, AC output voltage and inverter temperature continuously using suitable sensors.
- To show the readings on a local LCD and send them over Wi-Fi to a Blynk dashboard for remote monitoring.
- To protect the inverter automatically by switching a cooling fan ON at high temperature and isolating the inverter through a relay on abnormal readings, with remote ON/OFF control from the dashboard.

The rest of the paper is arranged as follows. Section 2 reviews related work. Section 3 describes the proposed system, its circuit and working. Section 4 gives the hardware and software implementation. Section 5 presents the results and discussion, and Section 6 concludes the paper.

2. Literature Survey

Anusha built an energy monitor in which an ESP32 reads voltage and current sensors, shows the values on an LCD and uploads them to ThingSpeak [1]. The work shows that ThingSpeak is a workable place to store and plot electrical measurements. El-Khozondar et al. also used the ESP32 to

measure voltage, current, active power and accumulated energy, and sent the data to Blynk and WhatsApp [2]. Both systems are general energy monitors; neither is designed around an inverter.

Inverter-specific work is closer to the present project. Ben-Festus and Falode monitored inverter battery voltage, current and temperature and passed the data to a cloud server and a mobile application, reporting battery voltages of 11.6–14.4 V and temperatures of 25–36 °C during charging tests [3]. Pandya et al. looked at transient power behaviour in solar inverters through IoT monitoring [4]. Akinola et al. went a step further and built a smart inverter that monitors battery and load voltage and current and ambient temperature, and also allows wireless switching of the connected loads [5].

On the power-stage side, Aditya et al. and Gopinath and Vijayakumar studied nine-level boost inverter topologies for EV and grid-connected PV use [6], [7]. Barbosa et al. earlier applied a boost current multilevel inverter to single-phase grid-connected PV systems [15]. These papers deal with how the AC waveform is produced, not with watching the inverter while it runs.

Table 1 compares some of the reviewed systems with the proposed one. Most of them cover either monitoring or control, and few combine a local display, a cloud dashboard, temperature-based fan control and relay isolation in a single unit. The proposed system is built to bring these together.

Table 1 - Comparison of existing systems and the proposed system

Ref.	Main technology	Output / function
[1]	ESP32, voltage and current sensing, LCD, ThingSpeak	Remote view of energy readings
[2]	ESP32, Blynk, WhatsApp	Voltage, current, power and energy monitoring
[3]	Battery voltage, current and temperature sensing, cloud server, mobile app	Inverter battery monitoring
[5]	Sensors, wireless communication, load switching	Monitoring with wireless load control
Proposed	ESP32, FZ0430, ACS712, ZMPT101B, DHT11, relay module, I ² C LCD, Blynk	Local LCD, Blynk dashboard, automatic fan control, relay isolation, remote ON/OFF

3. Proposed System

3.1. Existing system and its limitations

A conventional inverter converts battery DC to AC and relies on built-in switching and protection circuits. The user mostly depends on indicator lights or on looking at the unit. Its limitations are:

- important parameters are not available together on a single display;
- no continuous remote monitoring, so the inverter has to be checked in person;
- temperature monitoring and automatic cooling control are limited; and
- remote ON/OFF control and stored operating data are generally not available.

3.2. Proposed system

The proposed system adds an ESP32 controller, four sensing units, an I²C LCD, a two-channel relay and Wi-Fi communication to a 12 V battery and DC-to-AC inverter. The ESP32 reads the sensors, compares each value with its programmed limit, shows the result on the LCD, uploads it to Blynk and, when needed, operates the relays. Monitoring, protection, cooling and remote control therefore sit on one platform.

3.3. Block diagram

The block diagram is shown in Fig. 1. The battery feeds the inverter, which supplies the AC load. The voltage, current and temperature sensors measure the battery side and the AC voltage sensor measures the inverter output. All signals go to the ESP32, which acts as the central control unit. It drives the LCD, sends data through its built-in Wi-Fi to the IoT cloud, and controls the inverter relay and the fan relay.

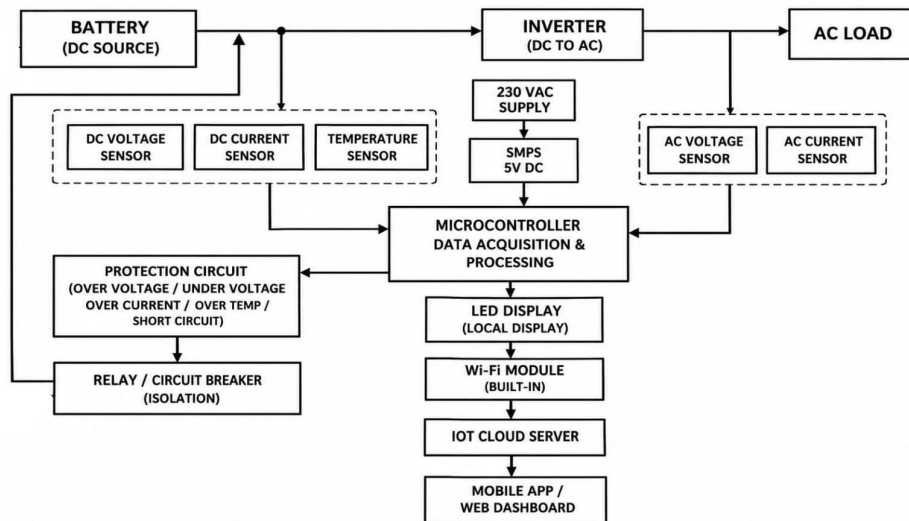


Fig. 1 - Block diagram of the proposed smart inverter monitoring and control system.

3.4. Circuit diagram

The complete circuit is given in Fig. 2. The 12 V, 4 Ah battery supplies the inverter and an LM2596 buck converter. The converter gives the regulated 5 V needed by the relay module, the sensor modules and the LCD, while the ESP32 itself works at 3.3 V logic. A common ground is kept across all sections. The battery voltage module is connected to GPIO34, the ACS712 current sensor to GPIO35, the ZMPT101B module to GPIO32 and the temperature sensor to GPIO4. The inverter relay is driven from GPIO26, the fan relay from GPIO27, and the LCD is connected on the I²C lines GPIO21 (SDA) and GPIO22 (SCL).

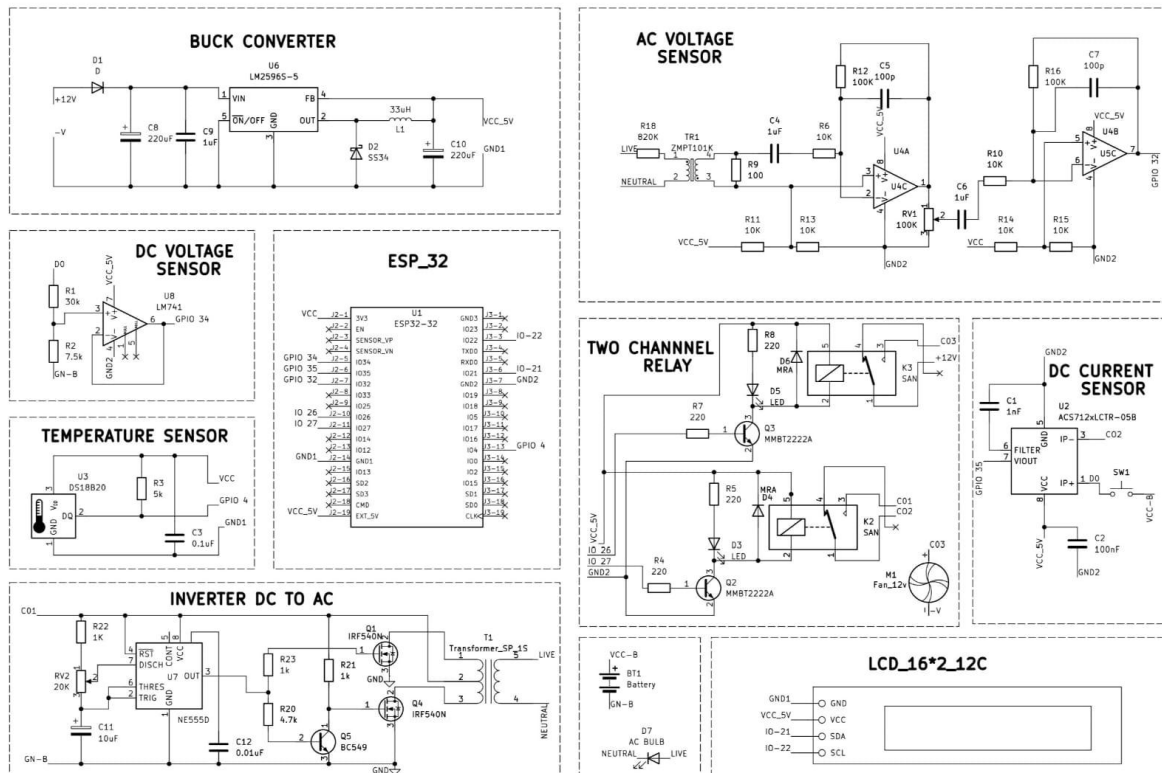


Fig. 2 - Circuit diagram of the entire system.

3.5. Working principle

When power is applied, the ESP32 sets up the sensors, LCD and relays and joins the configured Wi-Fi network. The battery then supplies the inverter, and the sensors keep measuring battery voltage, battery current, AC output voltage and temperature. The ESP32 converts these signals to engineering units, updates the LCD and uploads the values to the Blynk dashboard.

If the temperature rises above 35 °C the fan relay is switched ON, and it is switched OFF again when the temperature falls back. If the battery voltage, battery current or AC output voltage moves outside its allowed range, the inverter relay is opened, the inverter is isolated from the battery and the dashboard shows a fault. The user can also send an ON or OFF command from the dashboard; an ON command is ignored while a fault is active.

3.6. Sensor signal scaling

The ESP32 ADC accepts at most 3.3 V, so the sensor outputs must be scaled before they are read. The FZ0430 battery voltage module uses a resistive divider with $R_1 = 30 \text{ k}\Omega$ and $R_2 = 7.5 \text{ k}\Omega$, which gives a 5:1 ratio:

$$V_{out} = V_{in} \times R_2 / (R_1 + R_2) \quad (1)$$

For a fully charged 12 V battery, $V_{out} = 12 \times 7.5 / 37.5 = 2.4 \text{ V}$, which is safely inside the ADC range. The ACS712 current sensor has a sensitivity S of 100 mV/A and an output of 2.5 V at zero current, so the current is

$$I = (V_{out} - V_{zero}) / S \quad (2)$$

An output of 2.9 V therefore corresponds to $I = (2.9 - 2.5) / 0.1 = 4 \text{ A}$. For a 12-bit ADC reading, the voltage seen at the pin is

$$V_{adc} = (ADC \times 3.3) / 4095 \quad (3)$$

3.7. Flowchart

The program flow is shown in Fig. 3. After initialization the ESP32 reads the four sensors, updates the LCD and the dashboard, and then checks the temperature and the fault limits. The loop repeats every 3 s, so the inverter is monitored continuously.

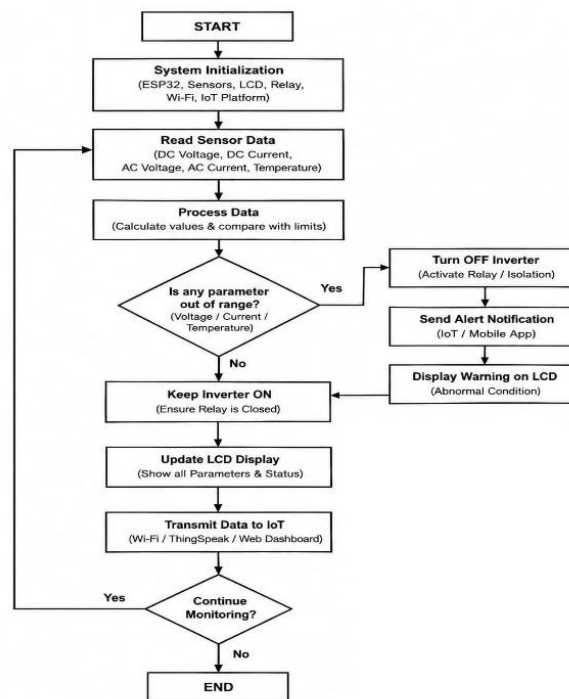


Fig. 3 - Flowchart of the proposed system.

4. Hardware and Software Implementation

4.1. Hardware components

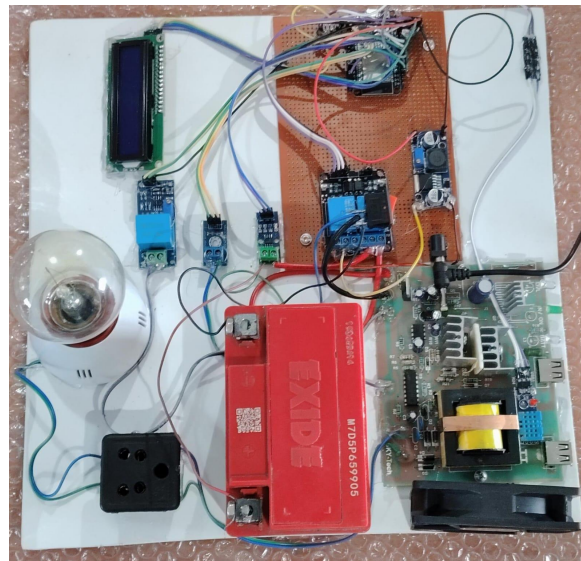
The components used and their main specifications are listed in Table 2. Each was chosen for its operating voltage, measuring range and compatibility with the ESP32. The ESP32-WROOM-32 has a dual-core processor running up to 240 MHz with built-in Wi-Fi and Bluetooth [8], which is enough to handle the sensors, the LCD, the relays and the Blynk connection. The ratings of the ACS712, LM2596 and DHT11 follow their datasheets [12]–[14].

Table 2 - Components used in the system

Component	Specification	Function
ESP32 controller	ESP32-WROOM-32, 32-bit dual core, 240 MHz, 12-bit ADC, Wi-Fi and BLE	Main controller
Battery	12 V, 4 Ah rechargeable	DC source for the inverter
Inverter	12 V DC to about 230 V AC, 300 W, 50 Hz, square-wave output	Supplies the AC load
DC voltage sensor	FZ0430, 0–25 V input, 5:1 divider (30 k Ω / 7.5 k Ω)	Battery voltage
DC current sensor	ACS712-20A, Hall effect, ± 20 A, 5 V supply	Battery current
AC voltage sensor	ZMPT101B, up to 250 V AC, 50/60 Hz, isolated output	AC output voltage
Temperature sensor	DHT11, 3.3–5.5 V, 0–50 °C, ± 2 °C, single-wire	Inverter temperature
Relay module	Songle SRD-05VDC-SL-C, two channels, 5 V coil, 10 A contacts	Inverter and fan switching
Buck converter	LM2596, 4.5–40 V input, adjustable output, 2 A	Regulated 5 V supply
LCD	16 $\times$ 2, HD44780 with PCF8574 I ² C backpack	Local display
Cooling fan	12 V DC brushless axial fan	Temperature control
AC load	40 W, 230 V incandescent bulb	Test load
Charging supply	12 V DC adapter, 1–2 A, 100–240 V AC input	Battery charging

4.2. Hardware prototype

The prototype was assembled on a flat board, as shown in Fig. 4. The ESP32, relay module and buck converter are mounted on a perforated board, and the LCD and sensor modules are placed next to it. The battery and the inverter board with its cooling fan are fixed on the same base, and the 40 W bulb is connected as the AC load.

**Fig. 4 - Hardware prototype of the smart inverter system.**

4.3. Software implementation

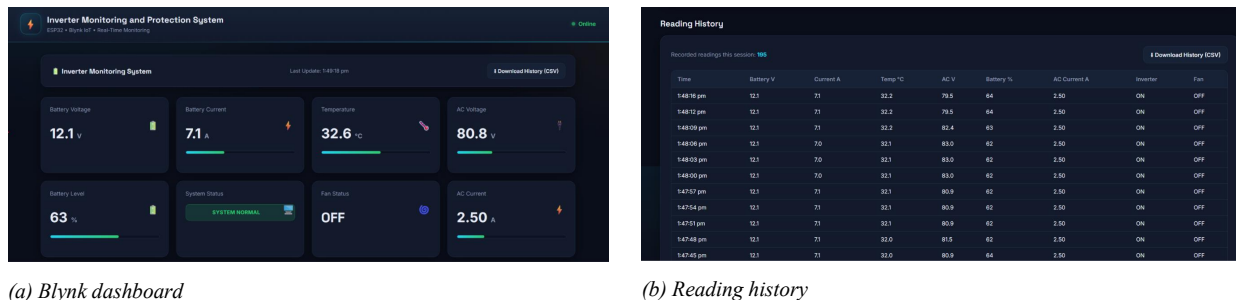
The firmware is written in Embedded C using the Arduino IDE and uploaded to the ESP32. It uses the WiFi, Blynk, Wire and LiquidCrystal_I2C libraries [9]. The program has two main functions: setup(), which runs once to start the serial port, the relay pins, the LCD and the Blynk connection, and loop(), which runs the Blynk client and a timer. The software is organized into the following blocks.

- **Sensor reading:** each analog pin is read with the 12-bit ADC and converted to volts using Eq. (3), then scaled to battery voltage, battery current, temperature and AC voltage.
- **Battery level:** the battery percentage is estimated linearly between 10.5 V (0 %) and 12.6 V (100 %).
- **Fault check:** the inverter relay is opened when the battery voltage falls below 10.5 V, the battery current goes above 20 A, or the AC voltage lies outside 150–250 V. The fan relay is switched ON when the temperature is above 35 °C.
- **Local display:** the LCD shows battery voltage and temperature for one second and then AC voltage and current for one second.
- **Cloud monitoring:** every 3 s the readings, the battery level, the fan and inverter states and a status message are written to Blynk virtual pins.
- **Remote control:** two Blynk virtual pins carry the inverter and fan commands; an inverter ON command is accepted only when no fault is active.

5. Results and Discussion

5.1. Hardware output

The prototype was tested with the 40 W bulb connected to the inverter output. The Blynk dashboard in Fig. 5(a) shows the readings at one instant: battery voltage 12.1 V, battery current 7.1 A, temperature 32.6 °C, AC voltage 80.8 V and battery level 63 %. The system status was SYSTEM NORMAL and the fan was OFF, since the temperature stayed below 35 °C. The history table in Fig. 5(b) holds 195 readings for the session, logged at 3 s intervals, and can be downloaded as a CSV file.



(a) Blynk dashboard

(b) Reading history

Fig. 5 - Hardware output of the system.

Fig. 6 plots ten consecutive readings from the history. The battery voltage stayed at 12.1 V, the battery current moved between 7.0 A and 7.1 A, and the temperature crept from 32.0 °C to 32.2 °C. The AC voltage varied between about 79.5 V and 83.0 V. The inverter state stayed ON throughout.

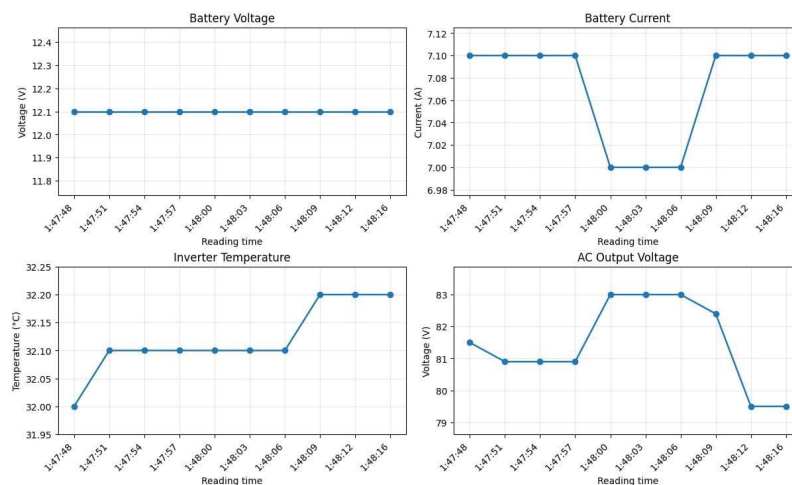


Fig. 6 - Monitoring graphs of the inverter parameters.

The AC voltage displayed is well below the 230 V rating of the inverter output, so the scaling of the ZMPT101B channel still has to be calibrated against a reference meter. Until then the 150–250 V limit for the AC channel should not be relied on.

5.2. System response

The response programmed for each event is summarized in Table 3.

Table 3 - Response of the system

Event	Response
All readings normal, temperature $\leq 35^{\circ}\text{C}$	Inverter keeps running, fan OFF, values shown on LCD and dashboard
Temperature $> 35^{\circ}\text{C}$	Fan relay ON; fan OFF again when the temperature falls
Battery voltage low, current excessive or AC voltage out of range	Inverter relay OFF; dashboard shows FAULT DETECTED
Remote ON / OFF command	Inverter relay switched; ON command ignored while a fault is active

5.3. Advantages

- **Local and remote monitoring:** the same four readings are available on the LCD and on the phone.
- **Automatic cooling:** the fan runs only when the temperature needs it.
- **Protection:** the relay isolates the inverter without waiting for the user.
- **Remote switching:** the inverter can be turned ON or OFF from the dashboard.
- **Low cost:** common modules and a single microcontroller are used.

5.4. Limitations

Remote monitoring and control need a working Wi-Fi connection. The DHT11 covers only $0\text{--}50^{\circ}\text{C}$ with $\pm 2^{\circ}\text{C}$ accuracy, and the ESP32 ADC is not perfectly linear, so the readings depend on careful calibration, as the AC voltage channel shows. The prototype is small, with a 12 V, 4 Ah battery and a 300 W square-wave inverter, and has not been tried on larger or solar-fed systems.

5.5. Applications

The system can be used with home and shop inverters, small offices, laboratories and battery-backed solar setups where the owner wants to see the battery and temperature without being next to the unit.

6. Conclusion and Future Scope

A smart inverter monitoring and control system using an ESP32, four sensors, an PC LCD, a two-channel relay and a Blynk dashboard has been designed and built as a hardware prototype. It measures battery voltage, battery current, AC output voltage and temperature, shows them locally and remotely, starts a cooling fan above 35°C , isolates the inverter on abnormal readings and accepts ON/OFF commands from the phone. Tests with a 40 W bulb showed stable readings and a continuous log on the dashboard.

In future, the system can be improved with the following features: (i) calibration of the AC voltage channel with a true-RMS measurement and the addition of AC current and power readings; (ii) a better battery state-of-charge estimate and a full battery management function; (iii) cloud storage of the logged data with alerts sent to the phone; and (iv) a printed circuit board and testing on larger and solar-fed inverters.

Acknowledgements

The authors thank Dr. S. Vijayabaskar, Professor and Head, Department of Electrical and Electronics Engineering, and the project coordinators Mr. P. Prakash and Mr. M. Vignesh Kumar of P. A. College of Engineering and Technology, Pollachi, for their support, and Dr. S. Arun, Associate Professor, for his technical guidance throughout the project work.

REFERENCES

- [1] M. Anusha, "Internet of Things (IoT) based energy monitoring with ESP32 and using ThingSpeak," 2024.

-
- [2] H. J. El-Khozondar et al., "A smart energy monitoring system using ESP32 microcontroller," 2024.
- [3] B. N. Ben-Festus and O. A. Falode, "IoT-enabled monitoring system for battery management in inverter systems," 2026.
- [4] V. Pandya, J. Patel, D. Shah, P. Papaniya, and D. Senghani, "IoT-based transient power monitoring and measurement system for solar inverters," 2025.
- [5] A. Akinola, S. O. Adigun, I. M. Oyerinde, and O. Y. Obansola, "Design and implementation of an IoT-enabled smart inverter system for real-time monitoring and control," 2025.
- [6] K. Aditya, Y. Suresh, B. Shiva Naik, B. Nageswar Rao, and A. K. Panda, "A single-source nine-level boost inverter with new optimal switching scheme for EV applications," *Int. J. Circuit Theory Appl.*, vol. 52, no. 2, pp. 954–972, 2024.
- [7] N. P. Gopinath and K. Vijayakumar, "Common ground nine-level boost inverter for grid-connected PV applications," *Front. Energy Res.*, vol. 10, Art. no. 922786, 2022.
- [8] Espressif Systems, "ESP32-WROOM-32 datasheet," ver. 3.4, 2023. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf
- [9] Blynk Inc., "Blynk IoT platform." [Online]. Available: <https://blynk.io>
- [10] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 2015.
- [11] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [12] Allegro MicroSystems, "ACS712: Fully integrated, Hall effect-based linear current sensor IC," datasheet.
- [13] Texas Instruments, "LM2596 SIMPLE SWITCHER power converter 150-kHz 3-A step-down voltage regulator," datasheet.
- [14] Aosong Electronics, "DHT11 humidity and temperature sensor," datasheet.
- [15] P. G. Barbosa, H. A. C. Braga, M. D. C. B. Rodrigues, and E. C. Teixeira, "Boost current multilevel inverter and its application on single-phase grid-connected photovoltaic systems," *IEEE Trans. Power Electron.*, vol. 21, no. 4, pp. 1116–1124, 2006.