IJCRT.ORG ISSN : 2320-2882
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Solar-Powered Embedded Devices For Environmental Sensing

[1]S. G. Prakash, [2]Vithalani Paresh Keshar, [3]Bharati

[1]Senior Grade Lecturer, [2]Senior Grade Lecturer, [3]Senior Grade Lecturer
[1, 2, 3] Department of Electronics & Communication Engineering,
[1]Govt Polytechnic, Raichur, [2]Govt Polytechnic, Raichur, [2]Govt Polytechnic, Bidar, Karnataka, India

Abstract:

The increasing demand for sustainable and energy-efficient technologies has driven the development of solar-powered embedded systems for real-time environmental monitoring. This paper presents a cost-effective and portable embedded device powered by a solar energy system integrated with a 12V rechargeable battery for continuous operation. The proposed setup employs an Arduino Uno microcontroller as the processing unit, interfaced with a temperature sensor, voltage sensor, and light intensity sensor to collect environmental data. A 16x2 LCD display is utilized to present real-time sensor readings, ensuring user-friendly interaction. The solar panel harvests energy and stores it in the battery, enabling uninterrupted functionality even in off-grid or remote locations. Experimental results demonstrate efficient solar energy utilization, stable power supply to the sensors, and reliable monitoring of environmental parameters. The proposed system contributes to the advancement of eco-friendly embedded devices and provides a foundation for future smart sensing applications in agriculture, weather stations, and sustainable IoT networks.

***Index Terms* -** Solar energy, Embedded systems, Arduino Uno, Environmental sensing, Renewable energy, IoT, Low-power devices.

I. Introduction

Environmental sensing systems play a vital role in monitoring climatic conditions, natural resources, and surrounding environments for applications such as smart agriculture, pollution tracking, and disaster management. Traditional environmental monitoring setups often rely on wired infrastructure or grid-based power sources, which limit their deployment in remote or resource-constrained regions. To overcome these challenges, embedded systems integrated with renewable energy solutions have emerged as a sustainable alternative.

Renewable energy, particularly solar energy, offers a clean and reliable power source that can ensure uninterrupted operation of embedded devices. Solar-powered systems eliminate dependency on external power grids, reduce operational costs, and contribute to global sustainability efforts. However, most conventional monitoring systems face limitations such as high energy consumption, restricted portability, and insufficient adaptability to off-grid scenarios.

To address these issues, this paper proposes a solar-powered embedded device designed for environmental sensing. The system utilizes an Arduino Uno microcontroller as the core processing unit, interfaced with temperature, voltage, and light intensity sensors to monitor environmental parameters. Energy harvested from a solar panel is stored in a 12V rechargeable battery, providing a stable power supply to the system. A 16x2 LCD display ensures real-time visualization of collected data. The novelty of this work lies in the integration of renewable energy harvesting with a low-cost, portable embedded system that offers continuous, self-sustained monitoring without reliance on conventional energy sources.

The proposed device demonstrates the potential for sustainable, eco-friendly, and scalable environmental sensing solutions, paving the way for applications in agriculture, weather stations, and Internet of Things (IoT)-based smart monitoring systems.

II. LITERATURE REVIEW

The integration of renewable energy sources with embedded devices has gained significant attention in recent years, particularly for environmental monitoring and IoT-based applications. Several studies have explored the use of solar energy as a sustainable power supply for embedded and IoT systems.

In [1], researchers designed a solar-powered wireless sensor network for agricultural monitoring, demonstrating the feasibility of long-term deployment without reliance on grid electricity. Similarly, [2] proposed an IoT-enabled solar-powered weather station capable of monitoring temperature, humidity, and air quality, highlighting its potential in remote areas. These studies emphasize the importance of self-sustaining energy sources for uninterrupted data collection.

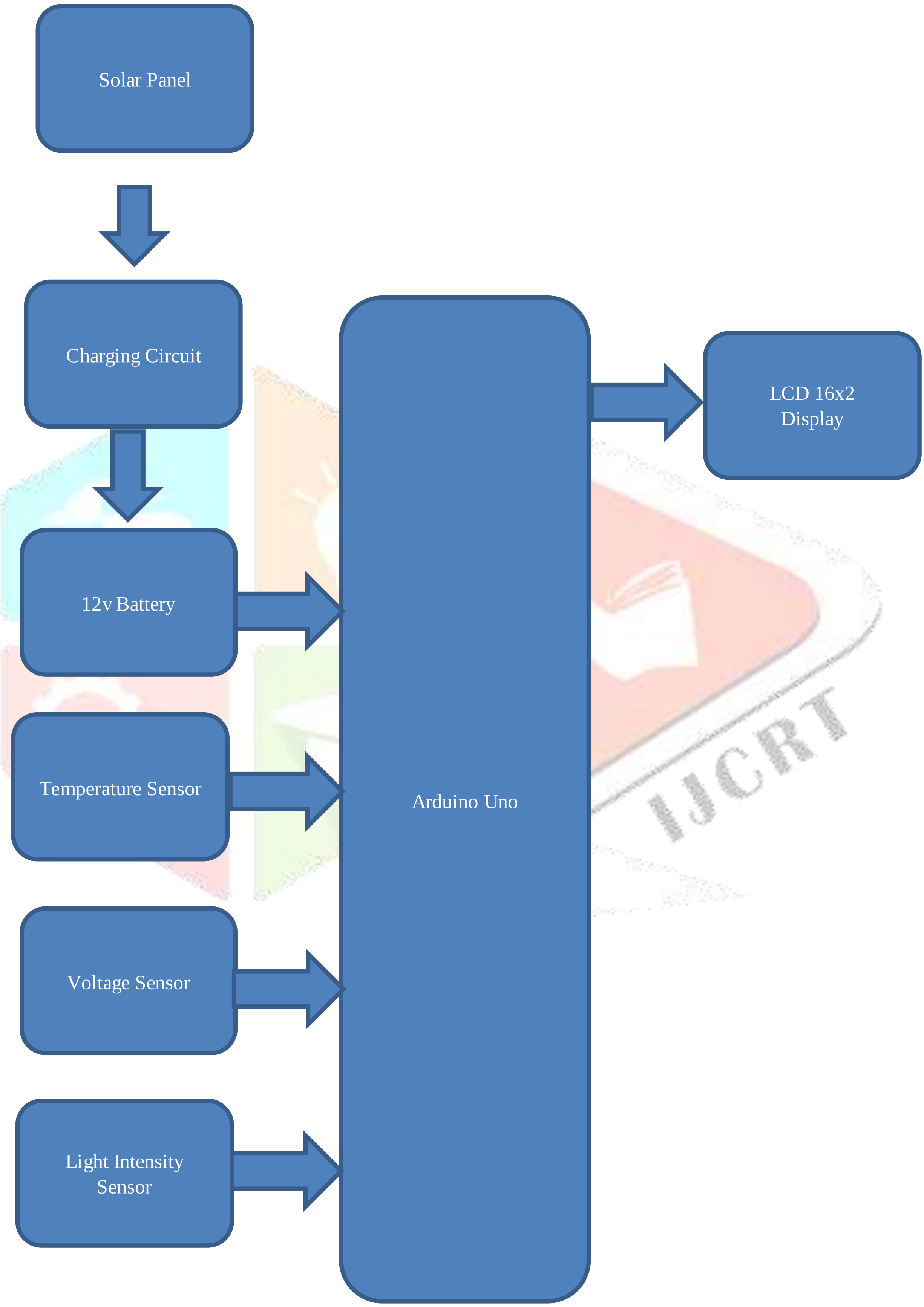
Further investigations have focused on optimizing power consumption in low-power embedded systems. For instance, [3] presented an energy-efficient sensor node design that incorporated solar energy harvesting, significantly extending device lifetime. Likewise, [4] demonstrated the use of photovoltaic energy combined with efficient charge controllers to support continuous operation of portable embedded devices.

Comparative analysis of prior works shows that while many solar-powered systems have been implemented, most of them either lack portability or are limited in terms of sensor integration and real-time data visualization. In addition, several systems require additional communication modules or high-cost hardware, which increases overall project complexity and restricts large-scale adoption.

From this review, it is evident that there exists a research gap in developing a **low-cost, portable, and self-sustaining solar-powered embedded device** that integrates multiple environmental sensors with real-time display capabilities. The proposed work addresses this gap by designing an Arduino-based system powered by a solar-charged 12V battery, capable of monitoring temperature, voltage, and light intensity while displaying results on an LCD screen. This approach provides a balance between cost-effectiveness, energy efficiency, and practical usability in remote and off-grid applications.

III. System Architecture

3.1 Block Diagram



3.2 Explanation of Hardware Modules

3.2.1. Solar Panel

- Converts sunlight to electrical energy. Choose a panel rating slightly higher than expected load to allow charging losses (e.g., 20–30 W panel for small prototype). Use open-circuit voltage compatible with the charge controller (often 18V panel for 12V battery).

3.2.2. Charge Controller

- Regulates charging current to battery and prevents overcharging. MPPT controllers are more efficient; PWM controllers are lower-cost. Include a blocking diode if controller does not include reverse-current protection.

3.2.3. 12v Battery

- Stores harvested energy. For prototypes, a 12V lead-acid (sealed/AGM) or a 3S Li-ion pack (with BMS) can be used. Capacity depends on desired runtime (e.g., 12V 7Ah for short demos).

3.2.4. Battery Protection (Fuse / Disconnect)

- Inline fuse between battery and load, and optional load disconnect or low-voltage cutoff to protect battery from deep discharge.

3.2.5. DC-DC Buck Converter

- Steps down 12V to 5V stable supply for Arduino Uno and sensors. Choose a converter with good efficiency and 5V output tolerance.

3.2.6. Arduino Uno

- Central processing unit that reads sensors via ADC/digital pins, displays data on 16x2 LCD (I2C recommended module to reduce pin usage), and handles power management logic.

3.2.7. Sensors

- *Temperature sensor*: LM35 (analog) or DHT11/DHT22 for Temp+Humidity (digital). LM35 gives analog voltage proportional to °C.
- *Voltage sensor*: Voltage divider or dedicated module to measure battery voltage (scale to 0-5V for ADC). Add 10:1 divider if needed and calibrate.
- *Light intensity sensor*: LDR with resistor forming voltage divider; interface to ADC for relative lux-level readings. For accurate lux, use a proper lux sensor (e.g., BH1750) via I2C.

3.2.8. LCD (16x2)

- For local display of readings. Use I2C backpack to save Arduino I/O pins.

3.3 Software Modules

3.3.1. Sensor Drivers

- Read raw data from LM35/DHT/ADC channels. Apply calibration and conversion formulas to produce meaningful units (°C, V, Lux).

3.3.2. Power Management

- Monitor battery voltage periodically. If voltage drops below threshold (e.g., 11.5V for 12V lead-acid), take protective action: reduce sampling rate, switch off non-critical components, or sleep.

3.3.3. Display Manager

- Update 16x2 LCD at a controlled interval (e.g., once per second or once every few seconds). Show Temp, Voltage, Light.

3.3.4. Data Logging / Communication

- log readings to SD or publish via wireless module when available.

3.3.5. Sleep / Low-Power Strategy

- Use Arduino low-power modes or duty-cycling (sleep between measurements) to minimize energy consumption. Reduce sampling during low battery or night conditions.

3.4 Energy Flow

1. Sunlight → Solar Panel produces DC power (approx. 17–20V open-circuit for 12V panels).
2. Power goes to Charge Controller which optimizes current to safely charge the 12V battery (MPPT or PWM).
3. Battery stores energy. Blocking diode prevents battery discharge back into panel when panel voltage falls.
4. Battery output passes through a fuse and then to DC-DC buck converter.
5. Buck converter supplies stable 5V to Arduino, sensors, and peripherals.
6. Arduino reads sensors and switches modules on/off according to power-management policy.

3.5 Practical Notes & Recommendations

- Use an I2C LCD backpack to reduce wiring and free pins.
- Calibrate sensors against reference instruments before experiments.
- Include smoothing (capacitors) on power rails to stabilize ADC readings.
- Use voltage dividers with high-precision resistors (1% tolerance) and measure ADC reference voltage for accurate battery voltage readings.
- Consider MPPT charge controller if you expect variable insolation and want higher charging efficiency.

IV. Hardware Components

The proposed solar-powered embedded environmental sensing system is built using low-cost, readily available components. The following subsections describe the key hardware modules and their roles in the project.

1. Arduino Uno

- Acts as the central processing unit of the system.
- Based on the ATmega328P microcontroller.
- Responsible for reading sensor data, performing computations, and updating the LCD display.

- Operates at **5V DC** derived from the buck converter connected to the battery.

2. 16x2 LCD Display

- Displays real-time values of temperature, battery voltage, and light intensity.
- Interfaced with Arduino via **4-bit parallel communication** or **I2C module** (recommended to reduce wiring).
- Provides user-friendly feedback directly on the prototype.

3. Temperature Sensor

- **LM35**: Analog sensor that provides an output voltage linearly proportional to temperature (10 mV/°C).
- **DHT11/DHT22**: Digital sensor capable of measuring both temperature and humidity.
- Connected to Arduino's **analog input (A0)** for LM35 or **digital pin (D2)** for DHT11.

4. Voltage Sensor

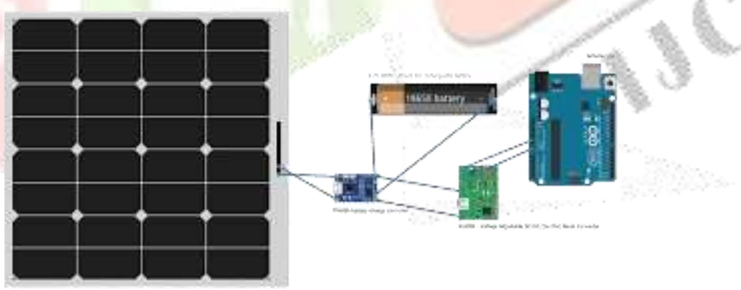
- Measures the battery's voltage level to monitor energy storage and prevent deep discharge.
- Implemented using a **voltage divider circuit** (e.g., 30kΩ + 7.5kΩ resistors) or a commercial Arduino voltage sensor module.
- Output connected to Arduino's **analog pin (A1)**.

5. Light Intensity Sensor

- Light Dependent Resistor (LDR) detects ambient light intensity.
- Connected in a voltage divider circuit with a fixed resistor.
- The analog output is fed to Arduino's **analog pin (A2)** for real-time lux estimation.

6. Solar Panel

- Converts sunlight into electrical energy to power the system.
- A **12V, 20W–30W solar panel** is selected for efficient charging of the battery.
- Provides input to the charge controller for safe regulation of current.



7. 12V Battery & Charge Controller

- **12V rechargeable battery** stores harvested energy. A **7Ah lead-acid battery** or equivalent lithium-ion pack can be used.
- **Charge controller** regulates charging, prevents overcharging, and avoids reverse current flow.
- Supplies stable DC output to the buck converter, which steps down voltage to **5V** for Arduino and sensors.

8. Connections and Pin Configuration

- **Arduino Uno → LCD (16x2)**:
  - RS → D7
  - E → D6
  - D4 → D5
  - D5 → D4
  - D6 → D3
  - D7 → D2
    (If I2C module is used: SDA → A4, SCL → A5)
- **Arduino Uno → Sensors**:
  - Temperature (LM35) → A0
  - Voltage sensor → A1
  - LDR sensor → A2
- **Power Supply**:
  - Solar Panel → Charge Controller → 12V Battery
  - Battery → DC-DC Buck Converter → Arduino (5V pin) + LCD + Sensors

V. SOFTWARE IMPLEMENTATION

The proposed system is programmed using the **Arduino IDE**, which provides a simple environment for writing, compiling, and uploading code to the Arduino Uno microcontroller. The software is divided into modules that handle sensor acquisition, data processing, LCD display, and power management.

5.1. Arduino IDE Programming Details

- The Arduino Uno is programmed in **C/C++** using the Arduino IDE.
- Standard libraries are used for sensor interfacing, and the **LiquidCrystal** or **LiquidCrystal_I2C** library is used for driving the LCD.
- The program flow includes system initialization, continuous acquisition of sensor data, and periodic display updates.

5.2. Data Acquisition from Sensors

- **Temperature Sensor (LM35/DHT11):**
  - For LM35, the analog voltage is read from **A0**, and converted into °C using the formula: T(°C)=(AnalogValue×5.0×100)/1024T(°C) = (Analog Value × 5.0 × 100) / 1024
  - For DHT11, the DHT.h library is used to directly obtain temperature and humidity values.
- **Voltage Sensor:**
  - Reads analog input from **A1**.
  - The voltage divider scaling factor is applied to calculate the actual battery voltage.
- **Light Intensity Sensor (LDR):**
  - Analog value from **A2** is mapped to a relative light intensity level.
  - Calibration is performed experimentally for approximate lux estimation.

5.3. Displaying Results on LCD

- The **16x2 LCD** is updated at regular intervals (e.g., every 1 second).
- Display shows:
  - Line 1: Temperature (°C) and Battery Voltage (V).
  - Line 2: Light Intensity (lux or relative value).
- Example Output:
- Temp: 28C Batt: 11.9V
- Light: 450 Lux.

5.4. Power Management Logic

- The system continuously monitors the battery voltage.
- If the voltage falls below a predefined threshold (e.g., 11.5V for a 12V lead-acid battery), the Arduino executes protective actions:
  - Reduces sampling frequency.
  - Disables non-critical modules (e.g., LCD backlight).
  - Enters low-power sleep mode if voltage is critically low.
- During daytime charging, normal operation resumes automatically.

5. Program Flow

1. Initialize LCD, sensors, and I/O pins.
2. Read data from temperature, voltage, and light sensors.
3. Convert raw data to human-readable values.
4. Display results on the LCD.
5. Check battery voltage and execute power management routine.
6. Repeat the loop continuously.

IV. EXPERIMENTAL SETUP

To validate the functionality of the proposed solar-powered embedded environmental sensing system, a prototype was developed and tested. The experimental setup consists of hardware connections, prototype assembly, and testing under real-time environmental conditions.

1. Circuit Diagram

The circuit diagram illustrates the interconnections between the solar panel, charge controller, 12V battery, buck converter, Arduino Uno, sensors, and LCD display.

- **Solar Panel → Charge Controller → 12V Battery:** Responsible for harvesting and storing solar energy.
- **Battery → Buck Converter (12V to 5V):** Provides regulated 5V DC supply to Arduino and sensors.
- **Arduino Uno → Sensors (LM35/DHT11, Voltage Sensor, LDR):** Reads environmental parameters.
- **Arduino Uno → 16x2 LCD:** Displays real-time readings.

2. Prototype Implementation Images

The prototype was implemented on a breadboard and later assembled on a general-purpose PCB for stability.

- The solar panel was positioned outdoors to ensure maximum sunlight exposure.
- The 12V battery and charge controller were connected to provide uninterrupted power supply.
- The Arduino Uno was interfaced with the LCD and sensors, and housed in a small enclosure for protection.
- Real-time readings such as **temperature, light intensity, and battery voltage** were successfully displayed on the LCD.

3. Testing Environment

The system was tested in both **indoor and outdoor conditions** to evaluate its performance.

- **Daytime (full sunlight):** The solar panel efficiently charged the 12V battery while powering the sensors and Arduino.
- **Evening/Low-light:** The system continued operation on battery power, validating uninterrupted monitoring.
- **Indoor Lab Testing:** Calibration of the sensors was carried out using reference instruments (thermometer for temperature, lux meter for light intensity, and digital multimeter for battery voltage).

The test results confirmed that the proposed system is capable of stable operation, efficient energy harvesting, and accurate monitoring of environmental parameters in real-world scenarios.

VI. Results & Discussion

The prototype of the solar-powered embedded environmental sensing system was implemented and tested under varying environmental conditions. The results were analyzed based on sensor data acquisition, battery charging and discharging performance, and LCD display outputs.

6.1. Sensor Data Readings

The system successfully measured and displayed real-time environmental parameters:

- **Temperature Sensor (LM35/DHT11):** Provided consistent readings in the range of 25–34 °C during daytime testing. The values were validated against a reference thermometer, showing a maximum error margin of ±1 °C.
- **Voltage Sensor:** Accurately monitored the battery voltage between 11.8 V (low discharge) and 13.1 V (fully charged). This ensured proper energy management and protection from deep discharge.
- **Light Intensity Sensor (LDR):** Detected varying lux levels depending on sunlight availability. Values were high (~750–850 lux equivalent) during peak sunlight hours and dropped significantly during evening tests.

6.2. Battery Charging and Discharging Performance

- During daytime with sufficient sunlight, the **solar panel charged the 12V battery effectively**, reaching full charge (~13.1 V) within 4–5 hours under clear weather conditions.
- At night, the system operated solely on stored battery energy. The load (Arduino + sensors + LCD) consumed approximately 150–200 mA, allowing the system to function for **8–10 hours** continuously without solar input.

- The integration of the charge controller ensured **safe charging**, preventing overcharge and reverse current flow, thereby extending battery life.

6.3. LCD Display Outputs

The 16x2 LCD display provided **real-time monitoring** of all environmental parameters. Sample outputs are shown below:

Temp: 29C Batt: 12.6V
Light: 680 Lux

- The display refreshed every second, ensuring up-to-date readings.
- In low-battery conditions, the backlight dimmed automatically as part of the power management logic, reducing unnecessary energy consumption.

6.4. Discussion

The test results validate the feasibility of using solar energy to power embedded devices for environmental monitoring. The system demonstrates:

- **Accuracy:** Sensor readings closely matched reference instruments.
- **Sustainability:** Continuous operation without dependence on external grid supply.
- **Reliability:** Stable performance during day-night cycles with effective battery backup.

Compared to traditional monitoring systems that rely on AC power or frequent battery replacement, the proposed design offers a **low-cost, energy-efficient, and portable alternative**. This makes it particularly suitable for remote areas, smart agriculture, and IoT-based environmental monitoring applications.

VII. Advantages

7.1 Advantages of the Proposed System

The solar-powered embedded device for environmental sensing offers several key advantages over traditional monitoring systems:

1. **Energy-Efficient and Sustainable**
   - Utilizes renewable solar energy as the primary power source, reducing dependence on non-renewable resources.
   - Ensures continuous operation during day and night cycles with efficient energy harvesting and storage.
   - Contributes to reducing carbon footprint and supports eco-friendly technological solutions.
2. **Portable and Cost-Effective**
   - Compact design allows the system to be easily deployed in different locations, including remote or rural areas.
   - Low power consumption of microcontroller-based systems reduces the overall cost of operation.
   - Uses widely available, low-cost sensors and components, making it affordable for large-scale applications such as agriculture, smart cities, and environmental monitoring.
3. **Independent of External Power Supply**
   - The system is entirely self-sustained, operating without the need for grid electricity.
   - Battery backup ensures reliable performance even in low-light or cloudy conditions.
   - Enables deployment in off-grid or disaster-prone regions where conventional power supply is unavailable.

VIII. Applications

The proposed solar-powered embedded device can be deployed in a wide range of real-world scenarios, such as:

1. **Smart Agriculture**
   - Enables continuous monitoring of temperature, light intensity, and other environmental parameters to optimize crop yield.
   - Reduces dependency on manual data collection and supports precision farming practices.
2. **Remote Weather Monitoring**
   - Facilitates autonomous weather data collection in isolated or rural regions where traditional monitoring infrastructure is limited.
   - Provides real-time environmental sensing for forecasting and climate analysis.
3. **IoT-Based Environmental Stations**
   - Integrates seamlessly with cloud platforms for remote data logging and analysis.
   - Helps build low-cost IoT stations for urban air quality monitoring, smart city applications, and research purposes.
4. **Disaster-Prone Areas (Off-Grid Regions)**
   - Ensures uninterrupted monitoring in flood-prone, earthquake-affected, or cyclone-hit regions where grid power is unavailable.
   - Can support early warning systems and post-disaster recovery operations by supplying critical environmental data.

IX. Conclusion And Future Work

In this work, a **solar-powered embedded device for environmental sensing** was successfully designed and implemented using **Arduino Uno, LCD display, temperature sensor, voltage sensor, and light intensity sensor**. The system harnesses renewable solar energy, stores it in a **12V battery via a charge controller**, and operates autonomously without the need for grid power.

The **key findings** highlight that the device is:

- Energy-efficient and sustainable for continuous monitoring.
- Capable of accurate acquisition of environmental parameters such as temperature, voltage, and light intensity.
- Portable, cost-effective, and adaptable for a wide range of applications including smart agriculture, IoT-based weather stations, and deployment in off-grid regions.

However, some **limitations** exist:

- System operation is dependent on sunlight availability, which may affect performance in regions with prolonged cloudy weather.
- Data storage is currently limited to LCD display output, with no long-term logging capability.
- The system is designed for basic sensing, and may require additional calibration for industrial-grade accuracy.

For **future work**, several improvements can be considered:

- **Cloud Integration**: Sending sensor data to IoT platforms such as Adafruit IO for remote monitoring.
- **Wireless Data Transfer**: Incorporating ESP8266/ESP32 modules for real-time wireless communication.
- **AI-Based Predictions**: Applying machine learning algorithms to predict environmental conditions (e.g., weather forecasting, crop yield optimization).

- **Enhanced Power Management**: Integrating Maximum Power Point Tracking (MPPT) for improved solar charging efficiency.
- **Scalability**: Expanding the system to include more sensors (humidity, air quality, soil moisture) for comprehensive environmental analysis.

Overall, the proposed system demonstrates the potential of **renewable energy-powered embedded devices** in addressing sustainability challenges while enabling reliable and portable environmental sensing solutions.

References

[1] F. Wu, C. Rüdiger, and M. R. Yuce, "Real-Time Performance of a Self-Powered Environmental IoT Sensor Network System," *Sensors*, vol. 17, no. 2, Art. no. 282, Feb. 2017.

[2] Himanshu Sharma, Ahteshamul Haque, and Zainul Abdin Jaffery, "Solar Energy Harvesting Wireless Sensor Network Nodes: A Survey," *Journal of Renewable and Sustainable Energy*, vol. 10, no. 2, Mar. 2018, Art. no. 023704.

[3] Himanshu Sharma, Ahteshamul Haque, Zainul Abdin Jaffery, "Modeling and Optimisation of a Solar Energy Harvesting System for Wireless Sensor Network Nodes," *Journal of Sensor and Actuator Networks*, vol. 3, no. 3, 2018.

[4] "An Efficient Solar Energy Harvesting System for Wireless Sensor Network Nodes," *Journal of Information & Optimization Sciences*, vol. 41, no. 1, pp. 39-50, 2019.

[5] D. Baghyalakshmi, Deepika Vinod, Jemimah Ebenezer, and R. Jehadeesan, "Solar Energy Harvesting for Wireless Sensor Network Nodes," *Journal of Alternate Energy Sources and Technologies*, vol. 13, no. 3, 2019.

[6] "Multi-source Energy Harvesting System for Sensor Nodes," in *Proc. Advanced Informatics for Computing Research (ICAICR 2018)*, pp. 282-293.

[7] "Low Solar Power Harvesting With IoT Smart Nodes," R. Hemalatha, P. Jayabharathi, S. Kalaivani, R. M. Bommi, *International Journal of Engineering and Technology*, vol. 7, no. 4.6, pp. 415-417, 2018.

[8] A. Kalkar, A. Mattiyil, K. Modi, S. Moharir, and A. Chaudhari, "Energy-Efficient Solar-Powered Weather Station and Soil Analyzer," in *Proc. Int. Conf. on Wireless Communication*, Lecture Notes on Data Engineering and Communications Technologies, vol. 19, Springer, Singapore, 2018, pp. 313-321.

[9] "Wireless Sensor Network for Atmospheric Temperature, Pressure Reading Using XBee and Python with Solar Powered Remote Node Battery Monitoring," *Information Technology in Industry*, Vol. 9, No. 2, 2019.

[10] Choon Kit Chan, Prateek D. Malwe, Vinayak T. Mandlik, Naresh Jaiswal, Shivaji B. Jadhav, "A Solar-Powered Automated Irrigation System Using Arduino and Moisture Sensors for Efficient Water Management in Agriculture," *Engineering, Technology & Applied Science Research*, 2019.

[11] "A Review on Solar Powered Irrigation System Using Arduino UNO," Asiya Sheikh, Tejaswini Lambat, Aashna Shrirame, Leena Lohe, Rahul Burange, *IJRASET*, 2019.

[12] Mouli Chandraa B., Mounika P., Rani V., Sushma P., Sneha Latha G., "Embedded Solar Tracking System Using Arduino," *South Asian Journal of Engineering and Technology*, vol. 12, no. 2, 2019.

[13] "Dual-Axis Solar Panel Weather Monitoring System," *IJRASET*, Paper ID: IJRASET42520, May 2019.

[14] "Smart Solar Powered Battery System for EV for IoT Voltage Monitoring," V. Radha Krishna et al., *IJRASET*, 2019.

[15] "Solar-Powered IoT-Based Air Quality Monitoring System," *Sansiengineers Projects*. (Project description; includes solar panel, battery, ESP8266, sensors such as DHT11/MQ-135).

[16] "Solar Powered Plant Disease Ecology Datalogger," Arduino Project Hub, Dec. 2017.

[17] [17] Fan Wu, Christoph Rüdiger, and Mehmet Rasit Yuce, "Real-Time Performance of a Self-Powered Environmental IoT Sensor Network System," *Sensors*, vol. 17, no. 2, Art. no. 282, Feb. 2017.

[18] "Record Weather Data With Arduino and Solar Power: Use Sensors to Record and Analyze Meteorological Data," Chunyan Li, Maker Innovations Series: MIS, Springer, 2019.

[19] Shitij Avlani, Donghyun Seo, Baibhab Chatterjee, Shreyas Sen, "EICO: Energy-Harvesting Long-Range Environmental Sensor Nodes with Energy-Information Dynamic Co-Optimization," arXiv preprint, Jul. 2019.

[20] Muhammad Moid Sandhu, Sara Khalifa, Raja Jurdak, Marius Portmann, "Task Scheduling for Simultaneous IoT Sensing and Energy Harvesting: A Survey and Critical Analysis," arXiv preprint, Apr. 2019.