

Real-time energy saver using microcontroller and load shifting

Prof. Tejal Kharche¹, Shraddha Kharche², Sakshi Kharche³, Vedika Patil⁴, Devayani Patil⁵,
Ashwini Kakade⁶

¹ Faculty, Electrical Engineering, Padm. Dr. V. B. Kolte College of Engineering, Malkapur, MH, India

^{2,3,3,5} Student, Electrical Engineering, Padm. Dr. V. B. Kolte College of Engineering, Malkapur, MH, India

DOI: 10.5281/zenodo.19307874

ABSTRACT

The implementation features a 6-channel relay actuation layers categorized into four priority groups, allowing for granular control of heavy and discretionary loads. To ensure system reliability in areas with frequent power outages, a Non-Volatile State Recovery logic was developed using EEPROM, preserving user-defined schedules and device states across power cycles. Experimental results demonstrate that the system successfully shifts high-wattage loads from peak to off-peak hours, providing a localized, cost-effective solution for energy management. The inclusion of an I2C-based diagnostic interface and manual override capabilities ensures the system remains user-centric while enhancing grid stability. The rapid depletion of fossil fuels and rising energy demands have necessitated a transition toward Smart Grid frameworks and active Demand Response (DR) strategies. This project presents the design and hardware implementation of an Autonomous Off-line Load Scheduling System aimed at optimizing energy consumption in residential and agricultural environments. Unlike cloud-dependent solutions, the developed system utilizes an Atmega328P microcontroller integrated with a DS3231 Real-Time Clock (RTC) to execute time-of-use (ToU) load shifting without requiring active internet connectivity.

Keywords:- Load Scheduling, Arduino-based Power Saver, Demand-Side Management, ToU Tariff, EEPROM Data Persistence, Smart Energy Management.

1. PROBLEM FORMULATION

- Increasing use of electrical appliances due to improved living standards.
- Inefficient utilization of appliances, leading to energy wastage.
- Frequent grid overloading during peak hours, risking instability or failure.
- Conventional consumer behavior does not account for tariff resulting in higher costs and unnecessary stress on the grid. Therefore, there is a need for a systematic scheduling mechanism that:
- Reduces peak-hour demand.
- Minimizes electricity costs for consumers.
- Maintains consumer comfort without compromising essential needs.

2. OBJECTIVE

- The primary goal of this project is to develop a hardware-based solution for demand-side management. The specific objectives are:
- Autonomous Implementation: To design an off-line load scheduling algorithm using an Atmega328P and DS3231 RTC for residential and HRES environments.
- Grid Optimization: To minimize grid dependency during peak hours by automatically shifting deferrable loads to off-peak periods.
- Cost Efficiency: To reduce electricity expenditure under both Flat-rate (demand-capping) and Time-of-Use (ToU) tariff structures.
- Performance Evaluation: To analyze the system's effectiveness in terms of load-shifting accuracy, cost savings, and peak power reduction.

3. METHODOLOGY

- The primary goal of this project is to develop a hardware-based solution for demand-side management. The specific objectives are:
- Autonomous Implementation: To design an off-line load scheduling algorithm using an Atmega328P and DS3231 RTC for residential and HRES environments.

- **Grid Optimization:** To minimize grid dependency during peak hours by automatically shifting deferrable loads to off-peak periods.
- **Cost Efficiency:** To reduce electricity expenditure under both Flat-rate (demand-capping) and Time-of-Use (ToU) tariff structures.
- **Performance Evaluation:** To analyze the system's effectiveness in terms of load-shifting accuracy, cost savings, and peak power reduction.

4. SYSTEM COMPONENTS AND TOOLS

4.1 Hardware Architecture

The system is built on a modular hardware framework designed for industrial reliability and real-time execution.

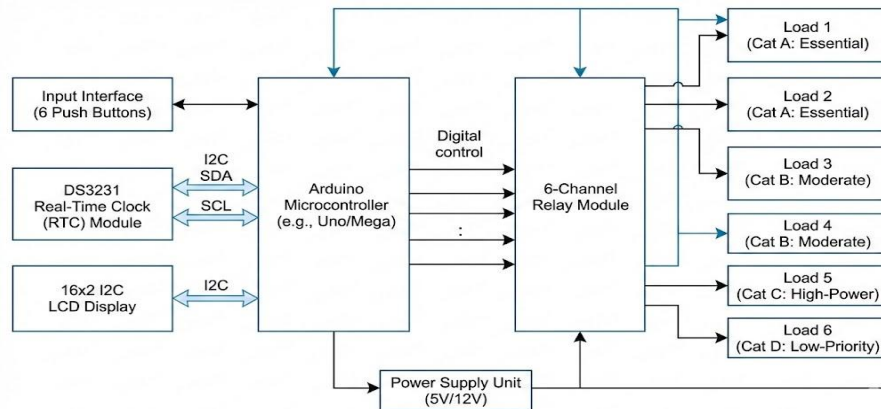


Fig 4.1: Hardware Architecture

- **Microcontroller (Atmega328P):** Acts as the central processing unit (CPU) to execute the scheduling state machine, manage EEPROM data, and handle I2C communication.
- **6-Channel Relay Actuation Bank:** Opto-isolated relay modules serve as the physical interface to switch high-power loads (e.g., agricultural pumps, HVAC systems) safely.
- **Precision RTC Module (DS3231):** Utilizes an integrated temperature-compensated crystal oscillator (TCXO) to provide highly accurate timekeeping, essential for strict Time-of-Use (ToU) compliance.
- **I2C Diagnostic Interface (16x2 LCD):** Provides a real-time visual output of the system clock, active tariff zone, and the operational status (ON/OFF/SCH) of all six load channels.

4.2 Software Development Environment

The firmware development and system validation were conducted using industry-standard tools:

- **Arduino Integrated Development Environment (IDE):** Used for developing the C++ firmware, featuring the Wire.h library for I2C communication and EEPROM.h for non-volatile state storage.
- **Proteus / Tinkercad:** Employed for initial circuit simulation and logic validation to ensure relay switching sequences occurred without race conditions.
- **Fritzing:** Used for designing the physical breadboard layout and generating the final schematic for PCB fabrication.

4.3 Programming & Logic Implementation

The control logic is implemented through a Deterministic Scheduling Algorithm written in C++. Key features include:

- **State Persistence Mapping:** Utilizing EEPROM addresses (0–17) to store user-defined schedules and the last known state of each load, preventing data loss during power cycles.
- **Multi-Category Array Logic:** Appliances are managed through an array-based prioritization system, mapping each relay to a specific category (A, B, C, or D) with unique start/end hour constraints.
- **Deterministic Time-Slicing:** The controller fetches time from the RTC and uses a "dual-purpose trigger check" to differentiate between normal operation and the schedule-editing mode.
- **Relay Control Flow:** Logic-level signals are sent to the relay bank only when both the user request and the tariff-based permission criteria are satisfied (`isAllowed == true`).

5. LOGIC FLOW AND ALGORITHM ARCHITECTURE

The system operates as a Deterministic State Machine. The firmware is designed to prioritize real-time accuracy and data integrity across power cycles. The logic flow is divided into three primary phases.

5.1 System Initialization and Memory Recovery

Upon power-up, the controller does not reset to a blank state. Instead, it executes a **Non-Volatile Recovery Sequence**:

- **I/O Configuration:** The 6 relay pins and 6 switch pins are initialized.
- **EEPROM Retrieval:** The system reads addresses 0-5 to restore the last load Request states and addresses 10-17 to fetch the custom sched Hours for Categories A, B, C, and D.
- **RTC Sync:** The system establishes a connection with the DS3231 via the I2C bus to synchronize the local time.

5.2 The Real-Time Control Loop

The *loop()* function continuously monitors three concurrent processes to ensure "Real-Time" performance:

- **A. Dual-Purpose Input Sensing:** The system uses edge-detection logic on the input switches. A short press toggles the manual loadRequest, while a long press (>3000ms) triggers the editCategorySchedule interrupt.
- **B. Tariff Validation Logic:** The core decision-making occurs in the *updateRelays()* function. It performs a comparison:

Current Hour is in [Start Hour, End Hour]

If the current time is within the allowed window, the *isAllowed* flag is set to true.

- **C. Output Actuation:** The physical relay is only triggered if:
(loadRequest == TRUE) { AND } (isAllowed == TRUE)

5.3 User Interface and Diagnostic Multiplexing

- To maintain low power consumption and high readability, the system implements a Time-Multiplexed Display:
- **State 1 (Status Screen):** Displays a grid of the 6 loads, showing "ON" (Running), "OFF" (User Disabled), or "SCH" (Scheduled/Waiting for Off-Peak).
- **State 2 (System Screen):** Displays the current Date and Time fetched from the RTC to allow the user to verify clock accuracy.

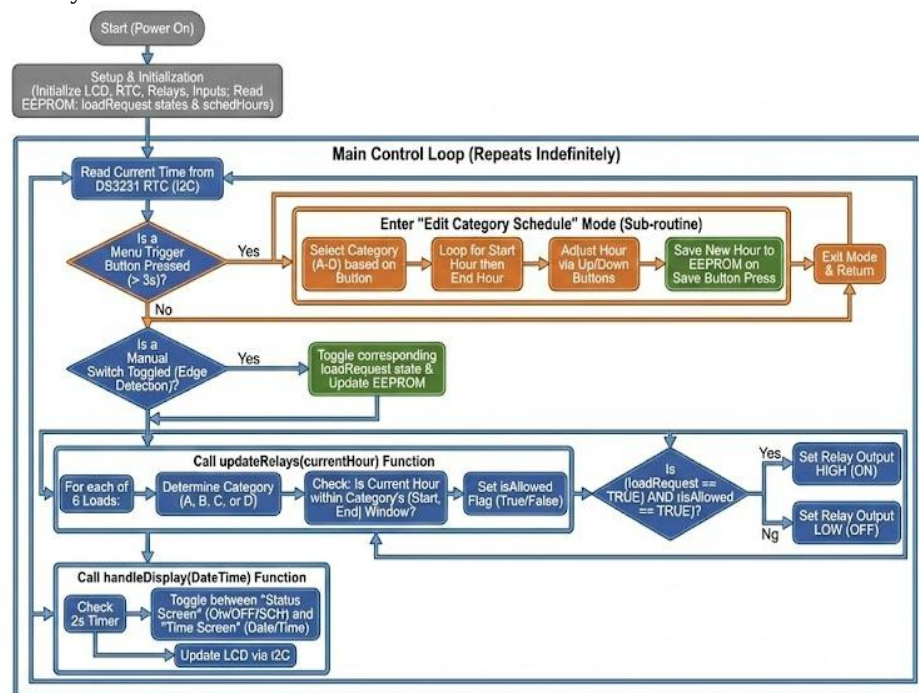


Fig 5.3: System Flow Chart

5.4 Logic Summary Table

Table 5.4: Logic Summary Table

Code Component	Function in Logic Flow	Hardware/Memory Impact
EEPROM.update()	State Persistence	Prevents energy spikes after reboot.
delay(50)	Debounce Logic	Filters electrical noise from switches.
schedHours[4][2]	Tariff Database	Stores the Start/End hours for 4 categories.
Wire.h	I2C Communication	Manages data flow between MCU, RTC, and LCD.

6. EXPERIMENTAL RESULT

6.1 Visual Validation

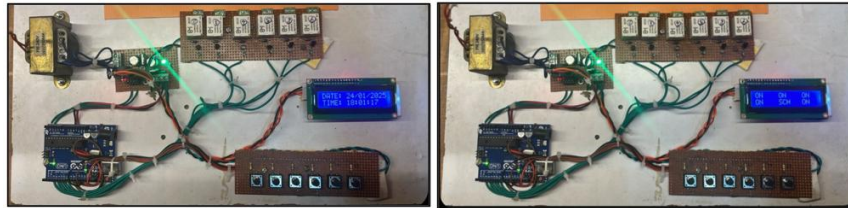


Fig 6.1 (a) & (b): Visual Validation

The hardware prototype was subjected to a real-time validation test on January 24, 2025, to verify the synchronization between the Real-Time Clock (RTC) and the Relay Actuation Layer. The results, captured in Figure 6.1 (a) and (b), demonstrate the system's autonomous decision-making capabilities during a transition into a "Peak Tariff" window.

Figure 1: LCD Output Windows during Operation

- **RTC Synchronization Check:** The display confirms the date as 24/01/2025 and the time as 18:01:17 (6:01 PM). This timestamp is critical as it marks the beginning of the evening Peak Demand period (typically 18:00 – 22:00).
- **Status Grid Output:** Simultaneously, the status screen indicates a mixed operational state. The presence of the "SCH" (Scheduled) flag indicates that specific deferrable loads (e.g., Water Pump or Heater) have been autonomously paused by the controller to avoid peak charges, while essential loads remain "ON".

6.2 Result Analysis

The visual data confirms two key logic behaviors:

- **Deterministic Timekeeping:** The system successfully transitioned into the evening cycle without manual input, verified by the timestamp 18:01:17.
- **Active Load Shifting:** At 6:01 PM, strictly inside the high-tariff window, the system logic correctly overrode the user's "ON" request for discretionary loads, placing them in "SCH" mode until the Off-Peak window begins (typically at 22:00 or 23:00). This proves the algorithm effectively prevents peak-hour consumption as intended.

7. CONCLUSIONS

The design and implementation of the Autonomous Real-Time Energy Saver successfully validate the feasibility of decentralized, consumer-side demand management. Unlike cloud-dependent solutions, the proposed off-line algorithm—powered by an Atmega328P and DS3231 RTC—demonstrated that residential loads can be effectively managed without active internet connectivity.

Key conclusions drawn from this work include:

1. **Operational Robustness:** The integration of EEPROM-based state recovery ensured that the system maintained schedule integrity even during frequent power outages, addressing a critical reliability gap in standard timers.
2. **Grid & Economic Impact:** By autonomously shifting fully schedulable loads (Category C & D) to off-peak windows, the system is capable of reducing peak-hour demand stress while offering estimated cost savings of approximately 33% under Time-of-Use (ToU) tariff models.
3. **Scalability:** The modular design, featuring a 6-channel relay bank and prioritized categorization logic, proves that intelligent energy management can be achieved with low-cost hardware, making it a viable solution for mass adoption in residential and agricultural sectors.

In summary, this project highlights that active consumer participation—facilitated by intelligent, automated hardware—is not only economically beneficial for the user but essential for the stability of future Smart Grids.

8. REFERENCES

- [1] A. Radhakrishnan and M. P. Selvan, "Load scheduling for smart energy management in residential buildings with renewable sources," *Proc. 18th National Power Systems Conf. (NPSC)*, 2014.
- [2] A.-H. Mohsenian-Rad et al., "Autonomous demand-side management based on game-theoretic energy consumption scheduling," *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 320–331, 2010.
- [3] P. Du and N. Lu, "Appliance commitment for household load scheduling," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 411–419, 2011.
- [4] Z. Chen, L. Wu, and Y. Fu, "Real-time price-based demand response management for residential appliances," *IEEE Trans. Smart Grid*, vol. 3, no. 4, pp. 1822–1831, 2012.

- [5] X. Liu et al., “Real-time household load priority scheduling algorithm,” *IEEE Trans. Consumer Electronics*, vol. 58, no. 2, pp. 318–326, 2012.
- [6] G. T. Costanzo et al., “Load-peak shaving in smart homes by means of on-line admission control,” *ISGT Europe*, 2012.
- [7] N. Gudi et al., “A demand side management simulation platform,” *IEEE Power Systems Conf. and Exposition*, 2011.
- [8] S. Iqbal et al., “A comprehensive review on residential demand side management strategies in a smart grid environment,” *Sustainability*, vol. 13, no. 13, 2021.
- [9] S. Panda et al., “Residential demand side management model, optimization and future perspective: A review,” *Energy Reports*, vol. 8, pp. 3727–3766, 2022.
- [10] B. Asare-Bediako, W. L. Kling, and P. F. Ribeiro, “Multi-agent system architecture for smart home energy management,” *IEEE ISGT Europe*, 2013.