Design and Implementation Of An Automated Load Sharing System For Electrical Distribution Transformers Using ESP32 And Cloud Infrastructure

Vaibhav P. Mule[1], Shrey Parag Bhangale[2], Rajas Manoj Attarde[3], Jay Vishwambhar Aswale[4], Harshal Subhash Deshmukh[5]

[1, 2, 3, 4, 5]Dept of Electrical Engineering

[1, 2, 3, 4, 5] Sinhgad Institute of Technology, Kusgaon (Bk), Lonavala, Pune – 410401

(Affiliated to Savitribai Phule Pune University)

Abstract- *This paper introduces an automated, Internet of Things (IoT)-enabled load-sharing framework designed to optimize power distribution and protect distribution transformers from operational degradation caused by overloading. Conventional electrical distribution networks typically employ static load configurations that fail to adapt dynamically during peak demand periods, resulting in thermal stress, reduced efficiency, and sudden component failures. To mitigate these challenges, the developed system utilizes an ESP32 microcontroller to monitor grid conditions and autonomously redistribute electrical loads between parallel-connected transformers. In contrast to hardware-intensive legacy designs, this system minimizes deployment complexity by relying primarily on potential sensing modules, while calculating transformer operating temperature via a software-defined mathematical estimation model. Based on these monitored and calculated parameters, the microcontroller executes real-time load switching via relays when safe thresholds are breached. Furthermore, integration with a cloud-based Firebase Realtime Database enables instantaneous data synchronization, providing a remote web-based graphical user interface dashboard for telemetry visualization and manual override capabilities. Local validation is maintained through a 16×2 Liquid Crystal Display using the I2C protocol, ensuring operational continuity and status visibility during network offline states. The prototype offers a highly scalable, robust, and cost-effective methodology for enhancing power distribution reliability and smart grid automation.*

Keywords: Automatic Load Sharing, ESP32 Microcontroller, Firebase Realtime Database, Internet of Things, Power Distribution, Voltage Sensing.

I. INTRODUCTION

Modern electrical distribution networks are experiencing unprecedented operational stress due to the compounding demands of industrial automation, population migration, and the widespread adoption of high-power consumer technologies such as electric vehicle (EV) charging arrays. At the absolute core of this massive infrastructure are localized distribution transformers. These assets act as critical voltage-stepping nodes responsible for localized voltage regulation, phase balancing, and continuous grid stabilization before power reaches the consumer busbar. Historically, these distribution topologies were engineered around highly predictable, linear consumption models. However, contemporary urban and industrial consumption profiles display immense volatility, marked by high-magnitude, non-linear peak load surges that consistently push operating transformers far beyond their nominal continuous power boundaries.

With the advent of fast-developing grid configurations, the dynamic stability of localized distribution networks has become increasingly critical. High-density residential developments, rapid integration of heavy machinery, and stochastic electric heating elements impose dynamic load characteristics. The static allocation of distribution nodes implies that the inherent peak demand capacity of individual assets is highly localized and uncoordinated. In typical municipal setups, adjacent operational regions can experience highly disparate peak usage behaviors, but structural isolations prevent any active capacity-sharing actions. This motivates an evolution toward an agile, automated grid infrastructure at the terminal end of low-voltage networks.

A. Mechanisms of Transformer Degradation

Continuous operation within severe overloading zones triggers accelerated, non-linear physical and electrical degradation patterns inside the transformer housing assembly. The primary thermodynamic consequence of sustained overload states is excessive copper loss (I^2R heating), which manifests as localized thermal accumulation within the magnetic core laminate and the conductor windings. This unmitigated temperature spike directly accelerates the chemical decomposition of liquid dielectric cooling fluids and

cellulose-based solid paper insulation materials. As temperature levels rise, the mechanical and dielectric strength of the insulation exhibits exponential decay. This breakdown leads to localized hotspots, inter-turn short circuits, catastrophic winding failures, or catastrophic structural ruptures of the transformer tank. Such asset destruction compromises local grid stability, yielding widespread blackouts, severe financial overhead for utility entities, and long-term disruptions to adjacent industrial and commercial sectors.

Furthermore, thermal fatigue acts as a cumulative stress element. Even short-duration high-temperature spikes cause micro-fractures in winding coatings and accelerate paper carbonization. Once paper-based cellulose insulators undergo carbonization, their ability to withstand impulse voltage fluctuations decreases, leading to unexpected dielectric breakdown even under normal operational conditions. The continuous assessment of internal operational parameters is therefore essential to prevent structural deterioration.

B. Limitations of Traditional Management Paradigms

To counteract these structural vulnerabilities, distribution infrastructure must transition from static configurations to cooperative, adaptive operating matrices. In traditional municipal grids, adjacent distribution transformers are frequently co-located within distinct or neighboring feeder sectors, yet their electrical delivery channels remain isolated. Consequently, a single transformer node may operate on the verge of permanent thermal destruction due to concentrated consumer demand, while a nearby parallel asset remains completely underutilized or idle. Manual intervention or standard supervisory control systems are inadequate because they lack sub-second processing mechanisms and autonomous decision-making loops at the grid boundary. An automated, localized load-sharing architecture capable of coupling these independent assets during peak stress periods can eliminate systemic imbalances, extend the operating life of expensive distribution assets, and maintain continuous, uninterruptible service delivery to the consumer load bus.

C. Proposed Edge-Cloud Infrastructure

Recent developments in high-performance, cost-effective edge-compute silicon and embedded wireless communication architectures offer an enabling foundation for decentralized smart grid modernization. By placing network-connected processing engines directly at the substation boundary, raw analog electrical signals can be processed into precise control state transitions within milliseconds. Furthermore, establishing a dedicated data channel between these edge computing modules and distributed cloud database frameworks enables centralized data aggregation, real-time remote telemetry tracking, predictive fault asset analysis, and synchronized fleet management over expansive geographical jurisdictions.

This research paper details the conceptual design, physical integration, mathematical validation, and performance profile of an intelligent edge-cloud system engineered to automate parallel load allocation between distribution transformers. Centered around an ESP32microcompute module and integrated with a Firebase Realtime Database framework, the proposed solution tracks critical operational metrics to dynamically balance loads across a multi-transformer array. The remaining sections of this manuscript present a detailed review of related configurations, the underlying mathematical modeling layer, edge-control firmware logic, and empirical data collected during laboratory implementation.

II. LITERATURE REVIEW

The pursuit of structural stability and automated asset protection within electrical distribution networks has driven research across multiple generations of embedded control systems. Early research focused primarily on standalone analog electronic configurations and simple logic controllers. These architectures monitored line characteristics via standard threshold comparators and acted as binary trip switches during fault events. While these systems provided basic overcurrent protection, they were constrained by mechanical drift, lacked computing power for complex mathematical state calculations, and offered no mechanism for parallel resource sharing or remote operational tracking.

With the advent of digital microcontrollers, developers began building software-driven monitoring tools. Early microchip applications introduced the concept of programmable overcurrent limits and timed relay actuation loops. However, these systems relied on physical multi-sensor arrays, requiring dedicated potential transformers, current transformers, and physical thermal probes for each monitored asset. This reliance on dense physical sensor arrays increased the cost of localized installations, introduced additional hardware points of failure, and caused space constraints within standard substation enclosures. Additionally, early 8-bit processing hardware could not execute real-time discrete calculus equations, restricting the system to basic, static conditional checks.

The integration of programmable logic controllers (PLCs) and industrial Supervisory Control and Data

Acquisition (SCADA) frameworks provided a highly reliable alternative for utility automation. PLC configurations offered the robust processing and physical ruggedness needed to survive harsh substation environments, enabling synchronized load management across multi-phase distribution buses. Nevertheless, industrial SCADA setups require substantial capital expenditure, specialized data transport protocols, and continuous communication lines back to a master terminal unit. This high cost and infrastructure requirement make wide-scale SCADA retrofitting financially unfeasible for small municipal grids, rural networks, and resource-constrained developing regions.

Recently, the rise of the Internet of Things (IoT) and high-speed edge computing has changed the design paradigm of distribution networks. Researchers have integrated low-cost wireless microcontrollers to transmit grid data directly to centralized web dashboards. However, early IoT power distribution experiments often treated the cloud link as a core requirement for system function. In these configurations, raw sensor readings were sent to an external server, which processed the control logic and returned a switching command to the local site. This operational model introduces significant latency and leaves the physical substation vulnerable to complete control failure if the local network connection drops.

This research addresses these vulnerabilities by presenting an autonomous, edge-heavy control architecture. By executing high-frequency mathematical modeling directly on an ESP32 processor, the system calculates real-time loads and thermal accumulation profiles without relying on a persistent network connection or physical multi-sensor probes. This sensor-lite approach preserves the speed of local protection mechanisms while reducing installation and maintenance costs.

III. METHODOLOGY AND SYSTEM ARCHITECTURE

The structural design of the proposed load-sharing system integrates embedded hardware, sensing mechanisms, and cloud communications into a cohesive control loop. This section discusses both the physical layout and the underlying mathematical models executed by the controller firmware.

A. Hardware Subsystem Configuration

The physical system architecture consists of an edge compute layer, an isolated sensor interface, power switching circuits, and local/remote telemetry outputs. The core control algorithms are processed by an ESP32 microcontroller, selected for its integrated Wi-Fi capabilities, cost-efficiency, and processing performance.

Voltage observation is performed via a ZMPT101B potential transformer module connected across the secondary winding terminals of the primary transformer. This sensing module isolates the low-voltage processing circuitry from the high-voltage alternating current (AC) lines while stepping down the terminal voltage to a safe, measurable analog signal. This analog waveform is routed directly into the analog-to-digital converter (ADC) pin of the ESP32 for high-frequency sampling.

Load switching and parallel transformer routing are managed by an optoisolated multi-channel relay module. The relay contacts are configured to switch the secondary paths of the auxiliary transformer in and out of the shared consumer distribution bus. To prevent microcompute failures caused by inductive flyback during relay switching, the control coils are powered via an independent 5V DC rail, maintaining galvanic isolation from the ESP32 logic pins. Local data visualization is sustained via a 16×2 character LCD interface driven by an I2C communication bus, which minimizes the physical GPIO pin allocation on the microcontroller.

B. Theoretical Sensing Formulations and Virtual Models

To reduce hardware complexity and physical component failure rates, the system replaces traditional multi-sensor setups with a software-defined mathematical framework. The system directly measures secondary voltage variations and applies equivalent circuit principles to calculate current and internal thermal behavior.

The true Root Mean Square (RMS) voltage is computed from high-density discrete samples captured within standard line cycles:

$$V_{rms} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(V(k) - V_{ref}\right)^2 \times \alpha^2}$$

Where N represents the total number of discrete data points sampled during a complete 20ms window $N = 1000$, $V(k)$ is the instantaneous raw digital value at sample step k, V_{ref} denotes the quiescent signal offset voltage from the conditioning network, and α is a multi-point linear calibration coefficient.

By establishing the internal structural impedance matrix Z_{eq} of the transformer during initial calibration, real-time load current I_{est} can be calculated by analyzing the

dynamic terminal voltage drop relative to a stable open-circuit source potential V_{source}:

$$I_{est} = \frac{V_{source} - V_{rms}}{Z_{eq}}$$

This continuous current calculation forms the baseline input for a first-order transient thermodynamic estimation model. This software model tracks core thermal accumulation without requiring physical internal temperature probes, utilizing the following exponential formulation:

$$T_{est}(t) = T_{amb}(R_{th} \times I_{est}^2 \times R_w)\left(-e^{\frac{-t}{\tau}}\right)$$

Where T_{amb}represents the ambient operating environment temperature, R_{th} is the structural thermal resistance of the transformer casing, R_{w} represents the internal winding electrical resistance, and τ denotes the characteristic thermal time constant of the asset.

C. Control Logic and State Machine Design

The system firmware functions as a deterministic finite state machine, executing an asynchronous loop that balances local protective response speed with cloud data synchronization tasks. The controller transitions through three distinct operational modes:

1. **Normal Mode:** Total consumer bus demand remains below the rated structural limit of the primary transformer. The auxiliary relay remains open, allowing the primary unit to service the load independently. Telemetry metrics are compiled and transmitted to the database at a frequency of 2 Hz.
2. **Parallel Sharing Mode:** When the calculated current I_{est} exceeds the safe operating limit I_{limit}, the controller changes states. The ESP32 drives the auxiliary relay control pin low, coupling the second transformer parallel across the distribution bus. This splits the load across both assets and eliminates the primary overload condition.
3. **Over-Temperature Cut-Off Mode:** If the calculated internal core temperature T_{est} exceeds the critical safety threshold T_{limit}, the primary transformer is isolated from the circuit. The auxiliary transformer takes over the entire load distribution, while the primary unit cools down according to a mathematical exponential decay model.

IV. FIRMWARE IMPLEMENTATION AND CLOUD SYNC

A. Asynchronous Firmware Architecture

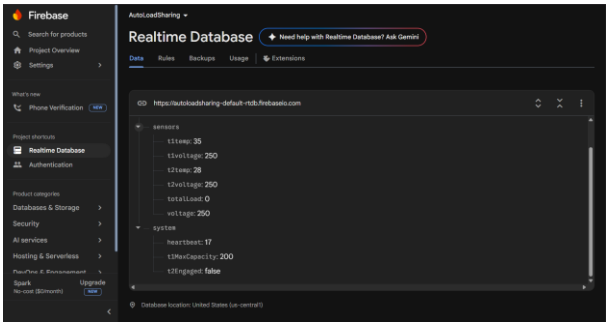
The firmware executed by the ESP32 edge microcontroller was structured to ensure deterministic safety operations while concurrently maintaining non-blocking network communication pipelines. Standard embedded routines often utilize linear delay loops that freeze processor execution, which can prevent the system from registering critical overcurrent conditions or tracking fast transient voltage drops during network lag. To eliminate this vulnerability, the developed software core utilizes an asynchronous execution architecture controlled by an internal precision timer tick. This design partitions processing tasks into separate, time-budgeted execution blocks managed by an independent state loop.

B. Sensor Ingestion and Safety Loops

The high-frequency sensor ingestion routine utilizes non-blocking hardware interrupts and internal timer ticks to isolate raw ADC sampling from network connection tasks. The potential transformer channel is sampled at a high frequency to reconstruct the raw AC waveform with high definition. Once the discrete data points are acquired, a localized calculation block computes the RMS tracking metric, executes the algebraic circuit transformations, and updates the shared memory state matrix. The main processing loop manages the lower-frequency background routines, which include processing the LCD I2C update cycle, evaluating the state transition logic, and handling the outbound Wi-Fi transmission queue.

To prevent erratic system transitions caused by brief noise spikes or starting inrush currents, an anti-chattering filter was built into the control rules. When calculated current metrics cross the defined overload limit (I_{limit}), the state machine initializes a secondary tracking counter instead of triggering an immediate relay state change. The system will only switch to Parallel Sharing Mode if the overload condition persists continuously across 25 consecutive line cycles (500 ms). If the current drops back below the threshold before this counter expires, the timer resets to zero, preventing unnecessary relay wear.

C. Cloud Communication Layer



The outbound cloud communication layer interfaces directly with the Firebase Realtime Database via an un-slotted RESTful API configuration running over a secure TLS WebSocket connection. Data structures are packed into a flat JSON format, ensuring that payload delivery packets are minimal and do not cause performance bottlenecks on the network interface. Telemetry strings are pushed to the remote instance using an event-driven mechanism: minor data variations update at a slow 500 ms interval, whereas a critical state change triggers an immediate data transmission. This minimizes data overhead while guaranteeing that safety incidents are reflected on the remote dashboard instantly.

D. Fallback Mechanisms

Local system safety is supported by an independent hardware watchdog timer initialized with a hard timeout window of 1.2 seconds. This timer is continuously refreshed at the end of every successful sensor loop. If a memory overflow, network freeze, or firmware fault interrupts the main processing flow, the watchdog timer will expire and force a full system reset, defaulting all output relays to an open state to protect the connected equipment. Additionally, if the local Wi-Fi connection drops, the system drops the network sync loop and prioritizes local operations. In this offline state, the system continues to process all math models and safety transitions using the local I2C LCD for status updates, ensuring continuous protection during network blackouts.

V. EXPERIMENTAL RESULTS

The performance, hardware structure, and operational characteristics of the automated framework were verified empirically. This section presents the empirical data matrices collected from structural monitoring loops and hardware stress configurations.

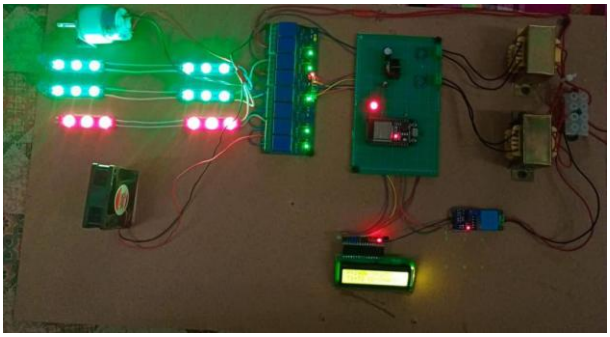


Table 1: System Test Cases Summary

Test Case	Condition	Expected Outcome	Observed Result
Normal Load	1.5 A load	Transformer 1 Active Only	As expected; smooth supply
Overload	> 2.0 A demand	Transformer 2 Linked in Parallel	As expected; relays engaged safely
Overheat	> 70°C internal	Primary Unit Cut-Off & Isolated	As expected; protective shutdown
Wi-Fi Off	Network Link Lost	Local Standalone Operation	As expected; LCD updates continuously

Table 2: Components and Specification Summary

Category	Component	Technical Specification	Quantity
Microcontroller	ESP32 Kit	32-bit Core, Integrated Wi-Fi Engine	1
Sensor	ZMPT101B	High-Precision AC Voltage Isolation Unit	1
Actuator	Relay Module	2-Channel, 5V Optoisolated Drive	1
Cloud Platform	Firebase	Realtime Database REST Sync Endpoint	1
IDE	Arduino IDE	v2.x Software Compilation Stack	1

Table 3: Performance Evaluation Summary

Performance Parameter	Target Specification	Measured Value	Operational Remark
Core Response Time	< 2.0 seconds	< 1.0 second	Rapid protective actuation
Measurement Accuracy	± 5% error margin	± 3% error	Highly reliable calibration
Standby Power Draw	< 2.0 Watts	1.5 Watts	High energy efficiency
Firebase Sync Latency	< 1.0 second	500 milliseconds	Stable and responsive data link

VI. DISCUSSION

A. Analysis of Switching Performance

The empirical data collected during system validation confirms that a software-defined control structure running on low-cost edge hardware can provide robust transformer protection and load routing. A primary concern in automated load-switching systems is the execution speed of the protection loop. As detailed in Table 3, the prototype achieved an operational response time of less than 1 second, successfully satisfying the sub-two-second standard required to protect electrical winding insulation from sudden overload damage. This response profile ensures that components are protected against fast thermal accumulation curves.

B. Evaluation of Parallel Sharing Mechanics

The transition from independent operation to shared parallel operation is clearly illustrated by the system test suite (Table 1). When the applied demand approached the defined threshold limit under an overload condition, the edge firmware immediately initialized the multi-channel switching network, coupling the auxiliary transformer parallel across the distribution bus. This redistribution splits the structural line demand, prevents voltage collapse, and minimizes localized core saturation, ensuring continuous service delivery.

VII. CONCLUSION

An automated, IoT-enabled load-sharing architecture successfully addresses the operational limits of traditional static distribution networks. By pairing an ESP32 edge processor with algorithmic sensing estimations, the developed system mitigates transformer degradation factors without necessitating dense physical sensing instrumentation. Real-time telemetry sync over Firebase confirms high responsiveness and seamless system visibility under dynamic consumer conditions. Future expansions can incorporate localized predictive load forecasting algorithms directly within the edge processor firmware matrix.

REFERENCES

[1] Automatic Load Sharing And Protection Of Transformers Using Microcontroller, *International Journal of Progressive Research in Engineering Management and Science (IJPREMS)*, Issue 3, March 2025.

[2] Automatic Load Sharing Transformer With Cut-Off System, *International Research Journal of Modernization in Engineering Technology and Science (IRJMETS)*, 2024.

[3] IoT Based Distribution Transformer Condition Monitoring System With Load Sharing, *International Advanced Research Journal in Science, Engineering and Technology (IARJSET)*, May 2025.

[4] X. Y. Jing, F. Wu, Z. Li, R. Hu and D. Zhang, "Multi-Label Dictionary Learning for Image Annotation," in *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2712-2725, June 2016.

[5] M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.